並列固有ベクトル計算における強制対角ブロック化の効果

片　桐　孝　洋[†,☆]

本論文では実対称三重対角行列の固有ベクトル計算時において，強制的に副対角要素を零にすることで対角ブロック行列化し，問題レベルの並列性を増加させる強制対角ブロック化の効果を検証する．HITACHI SR8000/MPP 128PE，Fujitsu VPP800/63 32 ノードを用い，理論上強制対角ブロック化の効果が期待できるように作成した試験行列を用いた数値実験の結果，演算量と通信時間を削減することで 78,000 倍程度の速度向上が達成できる例があることを確認した．

Effect on Endiagonal-blocked Method for Parallel Eigenvector Computations

Takahiro KATAGIRI[†,☆]

In this paper, we examine the effect of endiagonal-blocked method (EBM) for computing eigenvectors with real symmetric tridiagonal matrices. By using the EBM, small values of sub-diagonal elements are enforced with zero elements. This changes the matrices into block diagonal matrices, and hence the parallelizm of problem is dramatically increased. For the results of using 128 PEs of the HITACHI SR8000/MPP and 32 nodes of the Fujitsu VPP800/63, we observed a case that more than 78,000 times speed-up was obtained by reducing the computation complexity and communication time, when we applied a test matrix which was suitable for EBM theoretically.

1. はじめに

量子力学などの物理学における分野において，エルミート行列の固有値・固有ベクトルが必要となる場合がある．応用分野では，このエルミート行列の全固有値・全固有ベクトルを計算することをスペクトル分解とよぶ．エルミート行列 A のスペクトル分解を行う場合，以下の 3 種の方法が利用されることが多い：

[方法 (i)] 複素 Householder 変換を用いる方法

[方法 (ii)] 実対称行列に帰着させる方法

[方法 (iii)] エルミート行列 A を実部行列 A_R と虚部行列 A_I に分け，A_R と A_I のそれぞれに三重対角化を利用する方法[1)]

この 3 手法の性能について比較検討することは興味深いが，本論文ではあえて方法 (ii) にのみ焦点をあてる．この理由は方法 (ii) を用いることでエルミート行列の固有値・固有ベクトル計算が，方法 (i) に比べ演算量を変えることなく実対称行列用の固有値ソルバを利用できるからである[☆☆]．

また方式 (ii) を採用する大きな理由は，本論文で検証する強制対角ブロック化を適用できる点にある．ここで本論文で用いる強制対角ブロック化という概念は新しいものではなく，逐次計算機を対象に作られた固有値ソルバである EISPACK の固有ベクトル計算ルーチンにおいて従来から用いられている技法である．ところが従来の強制対角ブロック化は主に演算量の削減のみを考慮したものであり，並列性の向上という観点から実装されたものではない．そこで本論文では並列性の向上という点に注目し，実験的にその効果を検証することを目的にした．

本論文の構成は以下の通りである．まず強制対角ブロック化を説明する前にエルミート行列のスペクトル分解において，強制対角ブロック化が理論上利用できることを第 2 章で説明する．第 3 章で，本論文のテーマである強制対角ブロック化について述べる．第 4 章は HITACHI SR8000/MPP 128 PE (Processing Element)，および Fujitsu VPP800/63 32 ノードを用いて強制対角ブロック化の効果の上限を評価するため，その効果が期待できる人工的に作られた試験行列による性能評価を行う．最後に本論文で得られた知見をまとめる．

2. 強制対角ブロック化の適用例：エルミート行列のスペクトル分解

ここではまず，エルミート行列の固有値計算に関する定義や数学的性質の説明を行う．

† 科学技術振興事業団 さきがけ研究２１(情報基盤と利用環境)領域 “Information Infrastructure and Applications”, PRESTO, Japan Science and Technology Corporation(JST)

☆ 現在，電気通信大学大学院情報システム学研究科 Presently with Graduate School of Information Systems, The University of Electro-Communications

☆☆ さらに実対称行列用の様々な高速化技法を活用できる．欠点は，方法 (i) に比べて余分な記憶領域が必要な点である．

2.1 エルミート行列の定義

エルミート行列 (Hermitian Matrix) とは，以下の性質をもつ行列である．

$$A = A^H \tag{1}$$

ここで，$A \in \Im^{n\times n}$ ，A^H は行列 A の共役転置行列である．

いま，複素行列 A の実部からなる行列を $A_R \in \Re^{n\times n}$ ，虚部からなる行列を $A_I \in \Re^{n\times n}$ とおく．すなわち，

$$A = A_R + iA_I \tag{2}$$

とする．エルミート行列の定義 (1) から，以下の性質がある．

$$A_R^T = A_R \tag{3}$$
$$A_I^T = -A_I \tag{4}$$

すなわち実数行列 A_R は対称行列，実数行列 A_I は歪対称行列となる．

2.2 エルミート行列の性質

エルミート行列は以下の性質をもつ[2)]．

性質 1：エルミート行列のすべての固有値は実数である．

性質 2：エルミート行列の固有ベクトルは，もしこれらが異なる固有値に対応していれば，互いに直交している．

2.3 方法 (ii) によるスペクトル分解アルゴリズム[3)]

この方法は以下のように計算する方法である．いまエルミート行列 A を $A = A_R + iA_I$ ，複素固有ベクトル x を $x = u + iv$ とすると，複素固有値問題 $Ax = \lambda x$ は，

$$A(u+iv) = \lambda(u+iv)$$
$$(A_R + iA_I)(u+iv) = \lambda u + i\lambda v$$
$$(A_R u - A_I v) + i(A_I u + A_R v) = \lambda u + i\lambda v$$

となる．実部，虚部にまとめて行列表記すると，

$$\left[\begin{array}{c|c} A_R & -A_I \\ \hline A_I & A_R \end{array}\right] \left[\begin{array}{c} u \\ \hline v \end{array}\right] = \lambda \left[\begin{array}{c} u \\ \hline v \end{array}\right]$$
$$\hat{A}\hat{x} = \lambda\hat{x} \tag{5}$$

となる．いま，

$$\hat{A} \equiv \begin{bmatrix} A_R & -A_I \\ A_I & A_R \end{bmatrix}, \quad \hat{x} \equiv \begin{bmatrix} u \\ v \end{bmatrix} \tag{6}$$

とおいた．ここで，エルミート行列の性質である式 (3)(4) から $\hat{A}^T = \hat{A}$ である．すなわち $\hat{A}$ は対称行列である．したがってエルミート行列 A の固有値問題は，対称行列 $\hat{A}$ の固有値問題に帰着された (性質 1,2 も実対称性により満たす .)

また，以下のことがいえる．

定理 1：$A = A_R + iA_I$ をエルミート行列とする．このとき A の固有値 λ に対する複素固有ベクトル x は，対称行列 $\hat{A}$ の固有ベクトル $\hat{x}$ の要素から構成される複素ベクトル $u + iv$ および $-v + iu$ になる[3)]．

定理 1 から，対称行列 $\hat{A}$ に対する固有値問題では，ある固有値に対する固有ベクトルは 2 つ以上存在することを示している．すなわちエルミート行列 A の固有値が全て異なっていても，対称行列 $\hat{A}$ の固有値は同一の固有値が 2 個ずつあることになる．したがって理論上，重複固有値が必ず存在するわけである．

3. 強制対角ブロック化

エルミート行列を実対称行列に帰着させた行列 $\hat{A}$ のように重複固有値が理論上必ず存在するような対称行列を三重対角化する場合，副対角要素が零，もしくは零とみなせるような小さな値になることがある．すなわち三重対角化された行列 T は

$$T = \begin{bmatrix} t_{11} & t_{21} & & & & \\ t_{21} & \ddots & \ddots & & & \\ & \ddots & t_{k,k} & \epsilon & & \\ & & \epsilon & t_{k+1,k+1} & t_{k+2,k+1} & \\ & & & t_{k+2,k+1} & \ddots & \end{bmatrix},$$

となる場合がある．ここで ϵ は零とみなせる値である．この三重対角行列 T に逆反復法などで固有ベクトルを求めるとき，現在計算している固有値 λ が $t_{k,k}$ に非常に近い場合 ($\lambda \approx t_{k,k}$) ，$T - \lambda I$ の行列が特異に近くなり演算精度の著しい低下を引き起こすと推察される．

この状況を避けるため，もし副対角要素が小さくなった場合は，$T_1 \in \Re^{k\times k}$ ，$T_2 \in \Re^{n-k\times n-k}$ の小行列に分ける．すなわち

$$T = \begin{bmatrix} T_1 & \epsilon \\ \epsilon & T_2 \end{bmatrix}$$

とし，各対角ブロック行列 T_1 ，T_2 ごとに独立に固有ベクトルを計算する．ここで固有ベクトルに関しては，T_1 に対応する固有ベクトルは $(x_1, ..., x_k, 0, ..., 0)^T$ ，T_2 に対応する固有ベクトルは $(0, ..., 0, x_{k+1}, ..., x_n)^T$ のようにとる．

以上の実装上の工夫により，逆反復法中の演算精度，すなわち固有ベクトルの直交精度の改善が期待される．また小行列ごとに分割することで，固有ベクトル計算の計算量も削減できる．さらに並列化する場合，計算自体の並列性の向上，直交化に関する通信量も削減できることになる．

本論文では，この副対角要素を零とみなし強制的に対角ブロック行列に分割して計算精度の向上，演算量の削減，および並列性の向上を目指す方法を強制対角ブロック化(Endiagonal-Blocked Method，以降 EBM) とよぶ．

4. 性 能 評 価

4.1 実 験 環 境

ここでは EBM の効果を検証するため，東京大学情報基盤センタの HITACHI SR8000/MPP，および京都大学学術情報メディアセンタの Fujitsu VPP800/63

を用いてその効果を調べる．

HITACHI SR8000/MPP は分散メモリ型の並列計算機であり，メッセージ交換により通信を行うことができる．本実験で使用した SR8000/MPP は，東京大学情報基盤センタが所有する 144 ノード構成のものである．SR8000/MPP の各ノードは 8 つの PE を有し，各ノードにおける理論性能は 14.4GFLOPS，各ノードのメモリは 16 GB である．また通信網のトポロジは 3 次元ハイパークロスバ網であり，最大通信性能は片方向で 1.6 Gbytes/s，双方向で 3.2Gbytes/s である．本実験では，HITACHI Optimized Fortran90 V01-04 コンパイラが使われている．またコンパイラオプションは *-opt=ss -parallel=0* である．通信ライブラリとして，日立製作所が提供している MPI (Message Passing Interface) を用いている．

Fujitsu VPP800/63 はベクトル並列型の並列計算機であり，メッセージ交換により通信を行うことができる．本実験で使用した VPP800/63 は，京都大学学術情報メディアセンタが所有する 63 ノード構成のものである．VPP800/63 の各ノードは 8GFLOPS のベクトルユニット，および 1GOPS のスカラユニットからなり，各ノードのメモリは 8 GB である．また通信網のトポロジはクロスバ網であり，最大通信性能は 3.2Gbytes/s である．本実験では，Fujitsu UXP/V Fortran/VPP V20L20 コンパイラが使われている．またコンパイラオプションは *-O5 -X9* である．通信ライブラリとして，富士通が提供している MPI を用いている．

4.2 実 験 対 象

EBM の効果を検証するため，逆反復法 (Inverse Iteration Method，IIM) を用いた固有ベクトル計算[3] にこの機能を実装した．IIM のアルゴリズムの概略は以下の通りである：

- 対象行列：実対称三重対角行列 T
- 計算対象：標準固有値問題 $Tx_i = \lambda_i x_i$ の全固有ベクトル $x_i \in \Re^n$, $(i = 1, 2, ..., n)$
- 方法の詳細： 二分法であらかじめ全固有値を計算した上で，Rayleigh 商による固有値改良付き IIM で全固有ベクトルを計算
- 各固有ベクトル計算の最大反復回数：40
- 解の残差 $||Tx_i - \lambda_i x_i||_2$, $(i = 1, 2, ..., n)$ に対する要求誤差：$||T||_1 \times \epsilon$ (マシンε)
- 固有ベクトルの直交化：重複固有値に対する固有ベクトルを計算時に，計算済の重複固有値に対する固有ベクトルに対し Gram-Schmidt 直交化法を用いて再直交化

我々は既に，従来手法よりも通信量を削減できる効率の良い並列固有値ソルバ[4),5)] を開発している．IIM はこのソルバに採用されているので，新たに EBM の機能を組み込んで実行時間を測定する．

固有ベクトル計算時の再直交化には，以下の 5 種類の再直交化手法を実装した．

- MG-S：修正 Gram-Schmidt 法(Modified Gram-Schmidt Method, MG-S 法)
- CG-S(1)： PE 内データに MG-S 法を用いた古典 Gram-Schmidt 法 (Classical Gram-Schmidt Method, CG-S 法)
- CG-S(2)：PE 内データに CG-S 法を用いた CG-S 法
- IRCG-S：反復改良 CG-S 法 (Iterative Refinement CG-S Method, IRCG-S 法)[6)]
- NoOrt.：再直交化なし

4.3 試 験 行 列

試験行列として，Glued Wilkinson W_g^+ 行列 ($\delta = 10^{-14}$) を用いた．Glued Wilkinson 行列 W_g^+ は以下のように定義される．

$$W_g^+ \equiv \begin{bmatrix} W_{21}^+ & \delta & & & & \\ \delta & W_{21}^+ & \delta & & & \\ & \delta & \ddots & \ddots & & \\ & & \ddots & W_{21}^+ & \delta & \\ & & & \delta & W_{21}^+ & \end{bmatrix},$$

ここで行列 $W_{21}^+ \in \Re^{21\times 21}$ は

$$W_{21}^+ \equiv \begin{bmatrix} 10 & 1 & & & & & \\ 1 & 9 & 1 & & & & \\ & 1 & \ddots & \ddots & & & \\ & & \ddots & 0 & \ddots & & \\ & & & \ddots & \ddots & 1 & \\ & & & & 1 & 10 & \end{bmatrix}$$

である．

ここでは W_g^+ に対して EBM を適用しない / する，で以下の 2 種の行列に分けて性能評価を行う．

- 行列 (1)：EBM を適用しない
  - 次元 n：10,000
  - 重複しているとみなす固有値の距離 ： 64.5
  - 重複固有ベクトルのグループ数：1
  - 最大の重複固有ベクトル数 (再直交化が必要な固有ベクトル数): 10,000
- 行列 (2)：EBM を適用する
  - 次元 n：10,000
  - 重複しているとみなす固有値の距離 ： 0.129
  - 重複固有ベクトルのグループ数：13
  - 最大の重複固有ベクトル数 (再直交化が必要な固有ベクトルの最大数): 2
  - 対角ブロック行列の個数：477

行列 (2) では，EBM により複数の 21 次元の正方対角行列 W_{21}^+ に分解する．このことで分解された小行列 W_{21}^+ ごとに IIM を独立して適用できる点が行列

(2) の特徴である．

さらに重複しているとみなされる固有値に対する固有ベクトル計算時には，再直交化が必要であることに注意する．再直交化に必要な数は，重複している固有値に付随する固有ベクトル数と一致する．この理由から行列 (2) は行列 (1) に対し，$10,000/2 = 5,000$ 倍もの個数の再直交化を必要とする．

4.3.1 データ分散と並列アルゴリズムの概略

本実験で用いた IIM の行列のデータ分散について説明する．

いま PE 番号を $myid = 0, 1, ..., nprocs - 1$ とする．このとき，各データを所有する PE 番号は以下のようになっている．

- 実対称三重対角行列 T： 全 PE が全データを所有しているが，EBM を適用する場合は各 PE が担当する固有ベクトルに対応する対角ブロックのみ計算対象になる
- 固有値 λ_i： $i/(n/nprocs)$ 番 PE が所有
- 固有ベクトル x_i：$i/(n/nprocs)$ 番 PE が所有

ここで $n/nprocs$ は割り切れるものとした．

次に我々の並列 IIM ルーチン[4),5)] では，大まかには図 **1** のような流れで固有値計算を行う．

```
⟨1⟩ do   i = np*myid + 1, np*(myid + 1)
⟨2⟩   if ( λ_i は重複固有値でない ) then
⟨3⟩     IIM による x_i の計算;
⟨4⟩   else
⟨5⟩     i 番より前の重複固有値 λ_j (j < i)
            に対応する固有ベクトル x_j (j < i)
            全てに x_i を再直交化しながら
            IIM による x_i の計算;
⟨6⟩   endif
⟨7⟩ enddo
```

図 **1** 並列 IIM の概略．なお $np \equiv n/nprocs$ とおいた．

図 1 中の行 ⟨5⟩ では，当然自分の PE 内に固有ベクトルが無い場合も再直交化するが，この時は通信を必要とする処理となる．この再直交化に伴う通信量や通信時間は，再直交化の手法により大きく異なる．再直交化に関する並列アルゴリズムの詳細については紙面の都合上言及できないが，一つだけ注意すべき点は，このデータ分割を前提とする場合 MG-S 法の並列性が理論上皆無となる点である．すなわち，MG-S 法が本実験で用いた 5 種の再直交化手法の内で，最も時間がかかる処理になる．なお各並列再直交化の詳細は，文献 7) を参考にされたい．

EBM がなされる場合においても，基本的には図 1 のように処理が進む．具体的には，以下の手順で処理すべき固有ベクトルが無くなるまで処理を続ける．

(1) 強制対角ブロック化された対角ブロックを左上から順に番号をつける．
(2) 各 PE が計算すべき対角ブロックのうち，番号が最も若いものを処理対象とする．
(3) その対角ブロック内で重複固有値を調べ，図 1 の要領で処理を行う．

図 **2** は，この処理の一例を示している．

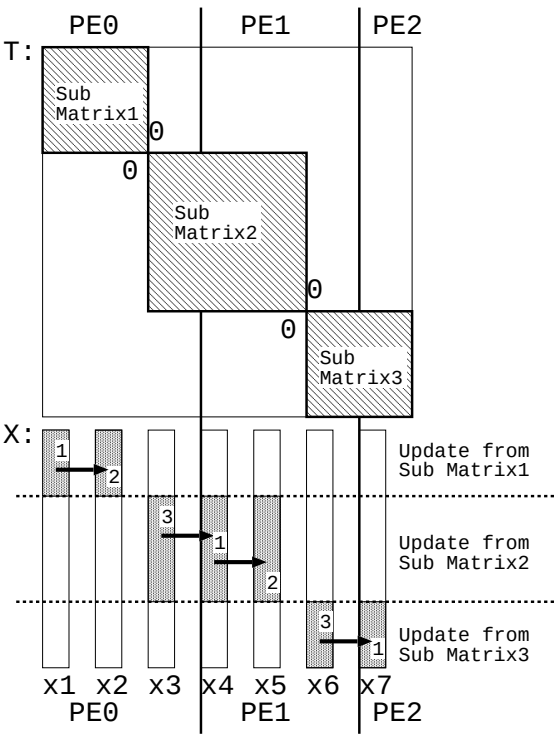

図 **2** 強制対角ブロック化された行列における固有ベクトル計算順序の例．ここで Sub Matrix1 は固有ベクトル x_1,x_2，Sub Matrix2 は固有ベクトル x_3-x_5，Sub Matix3 は x_6,x_7 に対応している．また固有値 λ_1,λ_2，固有値 λ_3-λ_5，そして固有値 λ_6,λ_7 は重複しているとする．すなわち，$\lambda_1 \approx \lambda_2$，$\lambda_3 \approx \lambda_4 \approx \lambda_5$，$\lambda_6 \approx \lambda_7$．

図 2 の例では，PE1 に関して固有ベクトル x_4 を計算する前に固有ベクトル x_6 を計算する方が全体の処理時間は減少するものと思われる．しかしながら，このような処理のスケジューリングは現在実装されておらず，今後の課題の一つである．

4.4 実 験 結 果

4.4.1 実 行 時 間

[HITACHI SR8000/MPP]： 表 **1** は，HITACHI SR8000/MPP での EBM 付きの IIM における実行時間を示している．また東京大学のスーパーコンピュータ利用環境の制約から，$\lfloor 64/\lceil p/8 \rceil \rfloor$ 時間以上のジョブは，自動的に実行が中断される．ここで p は PE 数である．

表 **2** は，表 1 における行列 (1) と行列 (2) との実行時間を割ることにより評価した EBM の効果である．

表 1　HITACHI SR8000/MPP による IIM を用いたルーチンの実行時間．単位は秒．ここで記号 > は，実行時間が利用環境による制限時間を越えたことを意味する．

(a) 行列 (1): EBM なし

#PE	MG-S	CG-S(1)	CG-S(2)	IRCG-S	NoOrt.
8	>	43,005	43,188	>	103
16	>	22,714	22,534	44,798	50
32	>	13,306	13,280	25,313	27
64	>	9,866	10,108	19,378	13
128	>	13,796	13,990	>	7

(b) 行列 (2): EBM あり

#PE	MG-S	CG-S(1)	CG-S(2)	IRCG-S	NoOrt.
8	0.51	0.55	0.56	2.49	0.59
16	1.92	0.55	0.69	0.61	0.33
32	0.97	0.36	0.82	0.85	0.20
64	0.52	0.53	0.45	0.37	0.38
128	0.31	0.27	0.22	0.25	0.07

表 2　HITACHI SR8000/MPP における EBM の効果 (行列 (1) の実行時間) / (行列 (2) の実行時間) で速度向上を算出．

#PE	MG-S	CG-S(1)	CG-S(2)	IRCG-S	NoOrt.
8	–	78,190	77,121	–	174
16	–	41,298	32,657	73,439	151
32	–	36,961	16,195	29,780	135
64	–	18,615	22,462	52,372	34
128	–	51,096	63,590	–	100

[Fujitsu VPP800/63]:表 **3** は，Fujitsu VPP800/63 での EBM 付きの IIM における実行時間を示している．

表 **4** は，表 3 における行列 (1) と行列 (2) との実行時間を割ることにより評価した EBM の効果である．

表 **3**　Fujitsu VPP800/63 による IIM を用いたルーチンの実行時間．単位は秒．

(a) 行列 (1): EBM なし

#N	MG-S	CG-S(1)	CG-S(2)	IRCG-S	NoOrt.
4	26,802	11,011	11,006	21,165	259
8	26,715	6,049	6,046	11,243	130
16	26,848	3,634	3,634	6,418	65
32	27,109	2,590	2,588	4,292	32

(b) 行列 (2): EBM あり

#N	MG-S	CG-S(1)	CG-S(2)	IRCG-S	NoOrt.
4	1.30	1.30	1.34	1.37	1.25
8	0.70	0.70	0.71	0.74	0.67
16	0.71	0.69	0.71	0.75	0.43
32	0.46	0.46	0.46	0.49	0.33

4.4.2　直 交 精 度

[HITACHI SR8000/MPP]： 表 **5** は，計算された固有ベクトルにおける直交精度を示している．

ここでこの精度は計算された固有ベクトルからなる行列を X とすると，行列 $(X^TX - I)$ の Frobenius ノルムによって評価するものとする．$n \times n$ 行列 $A = (a_{ij})$，$i, j = 1, ..., n$ の Frobenius ノルム $||A||_F$ 表 **4**　Fujitsu VPP800/63 における EBM の効果．(行列 (1) の実行時間) / (行列 (2) の実行時間) で速度向上を算出．

#N	MG-S	CG-S(1)	CG-S(2)	IRCG-S	NoOrt.
4	20,616	8,470	8,213	15,448	207
8	38,164	8,641	8,515	15,193	194
16	37,814	5,118	5,118	8,557	151
32	58,932	4,630	5,626	8,759	96

は以下のように定義される：

$$||A||_F \equiv \sqrt{\sum_{i,j=1}^{n} a_{ij}^2}\ . \qquad (7)$$

しがって $||A||_F \leq n$ であり，k 本の固有ベクトルが直交化しない場合の上限は $||A||_F \leq \sqrt{k(k-1)}$ 程度である．なお EBM を適用した行列 (2) については，強制対角ブロック化された各副対角行列に対応する固有ベクトルの直交精度の合計になる点に注意する．

表 **5**　HITACHI SR8000/MPP での IIM における固有ベクトルの直交精度．単位は Frobenius ノルム．

(a) 行列 (1): EBM なし

#PE	MG-S	CG-S(1)	CG-S(2)	IRCG-S	NoOrt.
8	–	2.26E-11	4.66E-11	–	166
16	–	1.34E-5	1.34E-4	1.34E-4	163
32	–	5.12	5.12	5.12	179
64	–	11.5	11.5	11.5	218
128	–	22.9	22.9	–	278

(b) 行列 (2): EBM あり (MG-S 法における最大残差: $max_i||Tx_i - \lambda x_i||_2 = 4.47\text{E-}8$)

#PE	MG-S	CG-S(1)	CG-S(2)	IRCG-S	NoOrt.
8	3.96E-8	3.96E-8	31.5	3.96E-8	31.5
16	3.96E-8	3.96E-8	31.5	3.96E-8	31.5
32	3.96E-8	3.96E-8	31.3	3.96E-8	31.5
64	3.96E-8	3.96E-8	31.3	3.96E-8	31.5
128	3.96E-8	3.96E-8	31.0	3.96E-8	31.5

[Fujitsu VPP800/63]：表 **6** は，計算された固有ベクトルにおける直交精度を示している．

4.5　考　　察

4.5.1　速度向上に関して

表 2，表 4 からこの行列の場合は，EBM の効果が非常に大きいことがわかる．特に何らかの再直交化を行う場合，速度向上が約 4,600-78,000 倍と極めて大きく，非常に効果的であることがわかる．この一方で再直交化を行わない場合は速度向上が約 30-190 倍であり，再直交化する場合と比べて少ない．このことから並列 IIM では再直交化に関する通信時間の占める割合が多く，再直交化の回数を減らすことが重要であるといえる．

表 3 から EBM を適用しない場合，MG-S は並列処理の効果 (台数効果) がほとんど得られないが，CG-S を利用すると台数効果が得られることがわかる．これ

表 6 Fujitsu VPP800/63 での IIM における固有ベクトルの直交精度．単位は Frobenius ノルム．

(a) 行列 (1): EBM なし (MG-S 法における最大残差: $max_i||Tx_i - \lambda x_i||_2$ = 2.32E-11)

#N	MG-S	CG-S(1)	CG-S(2)	IRCG-S	NoOrt.
4	1.92E-12	8.62E-12	4.94E-11	8.42E-12	161
8	1.94E-12	2.26E-11	3.47E-11	2.28E-11	156
16	1.91E-12	1.38E-5	1.41E-5	1.48E-5	171
32	1.88E-12	5.09	5.09	5.09	167

(b) 行列 (2): EBM あり (MG-S 法における最大残差:$max_i||Tx_i - \lambda x_i||_2$ = 4.47E-8)

#N	MG-S	CG-S(1)	CG-S(2)	IRCG-S	NoOrt.
4	3.96E-8	3.96E-8	31.5	3.96E-8	31.5
8	3.96E-8	3.96E-8	31.5	3.96E-8	31.5
16	3.96E-8	3.96E-8	31.5	3.97E-8	31.5
32	3.96E-8	3.97E-8	31.3	3.97E-8	31.5

は MG-S は理論上並列化できないことによる．またそのことから MG-S で EBM の効果が大きい．

4.5.2 演算精度に関して

表 5，表 6 から，EBM を適用すると直交精度が悪化する．この結果は，当初想定はしていなかった．なぜならば強制対角化することで切り離された副対角行列に付随する固有ベクトル同士は理論上も実際も完全に直交化しているので，副対角行列内の直交化を高い精度で行うことで，EBM を適用しない結果よりも高精度で計算できると推察できるからである．この原因は，(1) 各対角ブロック行列の直交化を高い精度で行えない；(2) 強制対角化したことで IIM 反復の結果に影響する重大な誤差が生じた；などの理由が考えられるが，詳しい解析は今後の課題である．もちろん EBM を適用した結果の精度で十分であるならば，実行速度の観点から EBM を適用するメリットは絶大といえる．

一方再直交化方式の観点では，CG-S を利用すると MG-S に比べ直交精度が悪化する結果が得られた．この結果は MG-S は理論上 CG-S よりも精度が高い手法であるので，この理論上の精度差が顕著になったものと推察される．

また EBM を適用した場合，CG-S(2) での直交精度が破綻している．この理由は PE 内に所有する固有ベクトルも CG-S で直交化したためと推察される．したがって CG-S を利用する場合でも，各 PE が所有する固有ベクトルについては MG-S で再直交化すべきである．同様に CG-S(1) でも PE 数が増加するにつれて直交精度が悪化していく理由は，CG-S の精度の悪さを補える MG-S が適用された PE 内固有ベクトルの数が，PE 数が増加するにつれ減少したためと推察される．

5. おわりに

本論文では，従来から利用されていた EBM を並列固有ベクトル計算に適用することで，最大で 78,000 倍もの速度向上が達成できる場合があることが明らかになった．ただし数値実験で用いた行列は EBM が極めて有効となるように作られた行列であり，ある意味 EBM で得られる効果の上限を評価したものであるといえる．したがって今後，実際の応用分野で用いられているエルミート行列等で性能評価をする必要がある．

実験結果から EBM を適用することにより直交精度を低下させる可能性がある．したがって直交精度の観点からの再評価，もしくは直交精度を改善する手法の研究をする必要がある．すなわち EBM の閾値をどこまで小さくできるか，直交精度の観点から追求することが今後の課題である．また EBM を適用してとりあえず計算し，その結果を利用して EBM を適用しないで問題を再度解き解の精度を改良する手法も，有用な方法となる可能性がある．

本数値実験の結果は，単純な並列処理の実装上の工夫やアイデアから数万倍の速度向上が得られる可能性を示唆している．したがって，並列数値計算においては実装の研究も軽視できない一例になっているといえる．

なお EBM 機能を付加した並列固有値ソルバは，ソースコードなどを含め ABC-LIB プロジェクトホームページ (`http://www.abc-lib.org/`) 上で公開準備中である．

参考文献

1) Bar-on, I. and Ryaboy, V.: Fast Diagonalization of Large and Dense Complex Symmetric Matrices, with Applications to Quantum Reaction Dynamics, *SIAM J. Sci. Comput.*, Vol. 18, No. 5, pp. 1412–1435 (1997).
2) Strang, G.: 線形代数とその応用, 産業図書, pp. 242–245 (1976). 山口昌哉 監訳，井上昭 訳.
3) Parlett, B. N.: *The Symmetric Eigenvalue Problem*, SIAM, pp. 7–11 (1998).
4) Katagiri, T. and Kanada, Y.: A Parallel Implementation of Eigensolver and its Performance, *IPSJ SIG Notes, 97-HPC-69*, pp. 49–54 (1997).
5) Katagiri, T.: A Study on Large Scale Eigensolvers for Distributed Memory Parallel Machines, *Ph.D Thesis, Department of Information Science, The University of Tokyo* (2000).
6) Balay, S., Gropp, W., McInnes, L. C. and Smith, B.: PETSc 2.0 Users Manual (1995). ANL-95/11 - Revision 2.0.24, http://www-fp.mcs.anl.gov/petsc/.
7) Katagiri, T.: Performance Evaluation of Parallel Gram-Schmidt Re-Orthogonalization Methods, *Proceedings of High Performance Computing for Computational Science (VECPAR2002)*, pp. 71–83 (2002).