

密行列固有値解法の最近の発展 (I)

- Multiple Relatively Robust Representations アルゴリズム -

山本有作

名古屋大学大学院工学研究科計算理工学専攻

Recent Developments in Algorithms for Solving Dense Eigenproblems (I)

- Algorithm of Multiple Relatively Robust Representations -

Yusaku Yamamoto

Department of Computational Science & Engineering, Nagoya University

Abstract. The Algorithm of Multiple Relatively Robust Representations (MR³) is a new algorithm for the symmetric tridiagonal eigenvalue/eigenvector problem proposed by I. Dhillon in 1997. It has attracted much attention because it can compute all the eigenvectors of an $n \times n$ matrix in only $O(n^2)$ work and is easy to parallelize. In this article, we survey the papers related to the MR³ algorithm and try to present a simple and easily understandable picture of the algorithm by explaining, one by one, its key ingredients such as the relatively robust representations of a symmetric tridiagonal matrix, the dqds algorithm for computing accurate eigenvalues and the twisted factorization for computing accurate eigenvectors. Limitations of the algorithm and directions for future research are also discussed.

1 はじめに

実対称行列または複素エルミート行列の固有値・固有ベクトルを求める問題は、科学技術計算において中心的な問題の一つであり、分子軌道法による化学計算、統計計算、情報検索 [3] などにおいて多くの応用がある。特に分子軌道法の分野では、近年、たんぱく質などの巨大分子の解析が行われるようになり、10 万元以上の密行列の固有値・固有ベクトル計算が必要な場面も生じている。

$n \times n$ の対称密行列 A の固有値・固有ベクトルを計算する場合、通常は、まず行列 A を直交行列 Q により対称三重対角行列 $T = Q^t A Q$ に変換し、 T の固有値・固有ベクトルを求める¹。したがって、三重対角行列の固有値・固有ベクトルを求める方法が重要となる。このためのアルゴリズムとしては、QR 法 [23][31]、二分法・逆反復法 [42][47]、分割統治法 [9][27] などのすでに確立した方法があり、広く使われている。しかし、これらは次章で述べるように、計算量が多い、並列化が困難、一部の固有値・固有ベクトルのみを求めることが困難などの欠点がある。そのため、数万元以上の大規模問題に対して適用するには十分でない。

¹ただし、精度においては三重対角化を経由しないヤコビ法が上回るという報告もある。[12]、[20] を参照。

これに対し、1997年に I. S. Dhillon らによって提案された Multiple Relatively Robust Representations (MR³) アルゴリズムと呼ばれる新しい手法がある [13]. この手法は (ある条件の下では) 次のような画期的な特長を持ち、注目されている.

- 全固有ベクトルの計算が $O(n^2)$ で可能
- 固有ベクトルの直交化が不要であり、各固有ベクトルは独立に計算可能
- k 本だけの固有ベクトルを求める場合、計算量は $O(kn)$

この手法は、三重対角行列の Relatively Robust Representation, 固有値を高精度に求めるための dqds 法, 固有ベクトルを高精度に求めるための twisted 分解など, さまざまな新開発の技法の組み合わせにより構成されているが, これらの技法や, その収束性, 精度を保証する定理などの重要な結果は, 10 年以上に渡る多数の論文に分散されて発表されている. また, MR³ 法はまだ発展段階にあるアルゴリズムであり, あらゆる入力行列に対して正しい解を与えることが理論的には必ずしも保証されておらず, 現在も研究が続けられている.

そこで本論文では, MR³ アルゴリズムに関する現在までの研究成果を整理し, アルゴリズムの骨格と収束性や精度の保証に用いられる定理とを, なるべく簡潔な形で紹介することを目的とする. また, アルゴリズムに関する性質のうち, 既存の定理でカバーできている部分とカバーできていない部分とをなるべく明らかにする形で紹介する.

以下では, 2 章で, 従来の固有値・固有ベクトル計算手法とその問題点について触れる. 3 章では理想的な固有値ソルバの持つべき特徴を挙げ, MR³ アルゴリズムの目標を導出する. その過程で, 中心的な役割を果たす Relatively Robust Representation の概念を導入する. 4 章で, MR³ アルゴリズムを解説する. 5 章では今後の発展方向について述べ, 最後に 6 章でまとめを行う.

2 従来の固有値・固有ベクトル計算手法とその問題点

対称な三重対角行列 T の固有値・固有ベクトルを計算するために従来広く使われている手法としては, QR 法, 二分法・逆反復法, 分割統治法がある. 本章ではそれぞれの長所・短所について, 簡単に述べる. なお, ここで取り上げる各解法のより詳しい比較については, [13][10][4] を参照されたい.

2.1 QR 法

QR 法は 1960 年代初めに Francis[23] と Kublanovskaya[31] により独立に提案された手法であり, T に対して直交行列による相似変換を繰り返すことにより, 対角行列に近づけてゆく方法である. 収束証明があり [36], かつ長い使用実績があるため, 信頼性のあるアルゴリズムとして認められている. 計算量は, 固有値のみを求める場合は $O(n^2)$, 固有ベクトルも求める場合は約 $6n^3$ である [10][24]. QR 法では, 単位行列に対して直交変換を繰り返すことに

より固有ベクトルを計算するため、固有値の分布にかかわらず、得られる固有ベクトルの直交性が高いという特長がある。また、固有ベクトル行列の計算は各行ごとに独立に行え、並列性が高い。

一方、短所としては、固有ベクトルの計算量が $6n^3$ と多いこと、一部の固有値・固有ベクトルのみを求めることが困難であることなどが挙げられる。また、固有値を求める部分については、並列化が困難であることが知られている²。そのため、分散メモリ型並列計算機向けの行列計算ライブラリ ScaLAPACK[8] の並列 QR 法では、固有値計算の部分は全プロセッサが重複して同じ処理を行い、固有ベクトル計算の部分のみプロセッサ間で分担して計算を行う方式が採用されている。

2.2 二分法・逆反復法

この方法では、Sylvester の慣性則に基づく二分法で固有値を求め、得られた近似固有値に基づき、逆反復法により固有ベクトルを求める [10][24][42]。なお、固有値を求める部分は二分法の代わりに QR 法を用いてもよい。二分法は、 k 個の固有値を求める場合、演算量は $O(kn)$ であり、多分法にすることにより、容易に並列化が可能である。逆反復法は、固有値が互いに離れている場合は、各固有値に対する固有ベクトルを独立に求めることができ、 k 本求める場合の演算量は $O(kn)$ と小さい。

しかし、近接固有値がある場合には、固有ベクトルの直交性を確保するため、修正 Gram-Schmidt 法などによる直交化を行う必要がある。直交化を行うか否かの基準としては、隣接する固有値間の距離が $10^{-3} \|T\|$ 以下の場合にそれらの固有値が同じクラスターに属すると判定し、同じクラスター内に属する全固有ベクトルの間で直交化を行うという Peters-Wilkinson の方法 [42] が広く採用されている。しかし、この方法を採用すると、1000 元以上の問題では、多くの場合、固有値のほとんどが 1 つのクラスターに属してしまうことが知られている [13]。この場合、直交化による演算量が $O(n^3)$ (k 本のみを求める場合は $O(k^2n)$) と増大する。さらに、直交化処理により各固有ベクトルの計算の並列性が失われてしまうという問題も生じる。

これらの問題を解決するため、直交化すべき固有ベクトルの組を削減して演算量を削減し、かつ並列性を抽出できるように直交化の計算順序を変更する方法 [35]、直交化のために並列性の高い古典的 Gram-Schmidt 法 [29] あるいは Householder 変換を使う方法 [51][52] などが提案されてきた。しかし、これらの方法はいずれも、固有値の分布によっては直交性が従来法より落ちる、あるいは演算量が $O(n^3)$ となるという短所があり、決定的な方法とはなっていない。

逆反復法では、上記のような並列化の際の問題点に加え、特殊な三重対角行列に対しては収束しない場合もあることが指摘されている [13][14]。

²ただし、これに対しても研究があり、いくつかの方法が提案されている [44][30][2]。

Table 1: Comparison of various algorithms for solving tridiagonal eigenvalue problems

解法	演算量	一部の固有ベクトルのみの計算	並列性	作業領域	LAPACK ルーチン
QR法	$6n^3$	×	固有値 Δ 固有ベクトル \odot	$O(n)$	DSTEQR
二分法・ 逆反復法	$O(n^2)$ $\sim O(n^3)$	$\circ (O(k^2n))$	Δ	$O(n)$	DSTEBZ + DSTEIN
分割統治法	$O(n^2)$ $\sim O(n^3)$	×	\circ	n^2	DSTEDC
MR ³ (目標)	$O(n^2)$	$\odot (O(kn))$	\odot	$O(n)$	DSTEGR

2.3 分割統治法

分割統治法は、固有値問題を半分のサイズの三重対角行列の固有値問題2個に分割し、各部分問題を解き、その結果を合成することで、元の固有値問題の解を得る方法である。各部分問題はさらに半分のサイズの問題に分割して解き、この操作を再帰的に繰り返す。Cuppen[9]により提案されたが、そのアルゴリズムは安定性に問題があり、後に Gu & Eisenstat により安定なアルゴリズム [26][27] が提案された。この方法は、問題を部分問題に分割して解くので並列性が高く、並列計算機上での実装と評価も行われている [19][46]。また、計算途中でデフレーションと呼ばれる処理を行うが、このために、固有値が密集している場合には収束が速くなり、 $O(n^2)$ の計算量で全固有値・固有ベクトルが求まる場合もある [10]。

しかし、計算量は固有値の分布に大きく依存し、最悪の場合は $O(n^3)$ となる。また、一部分の固有値・固有ベクトルのみを求めることは困難である。さらに、作業領域として n^2 の大きさの領域が必要である (QR法や逆反復法では $O(n)$ で十分) ため、大規模問題では不利となる。

2.4 まとめ

以上で述べた各手法の特徴を表1にまとめる。表では、各手法に対し、それを実装した LAPACK[1] ルーチンの名称も載せた。また、次章で述べる MR³ アルゴリズムについても、その目標とする特徴を併せて掲載した。

3 MR³ アルゴリズムの目標

以上を踏まえて、理想的な対称三重対角行列向けの固有値・固有ベクトル計算法が持つべき性質を挙げると、次のようになる。

- (a) 計算量が少ないこと。ただし、全固有ベクトルを求める場合、計算すべき要素の数は

n^2 だから, 少なくとも $O(n^2)$ 回の計算は必要である. そこで, $O(n^2)$ の計算量を目標とする.

- (b) 固有ベクトルのうち k 本のみが必要な場合, 計算量は $O(kn)$ にできること.
- (c) 粒度の大きな並列計算ができること. 特に, 各固有ベクトルの計算が独立に行えること.
- (d) 計算された固有ベクトルの精度が高いこと.

上記の目標のうち, (a), (b), (c) の3つについては説明は不要と思われるが, (d) については, より精密化することが必要である. 特に, 浮動小数点での演算を行うことを前提とした場合, そもそも実現可能な目標とは何かを考えなくてはならない.

以下では, 入力三重対角行列 T は既約, すなわち副対角成分に 0 を含まないと仮定する. この仮定が満たされない場合, T は複数の三重対角行列の直和となるから, この仮定により一般性が失われることはない. また, この仮定の下では, T は重複固有値を持つことはない [36] ことに注意する.

固有ベクトルの精度に関する目標としてもっとも望ましいのは, 真の固有ベクトルを \mathbf{v} , 計算された固有ベクトルを $\hat{\mathbf{v}}$ とするとき, \mathbf{v} と $\hat{\mathbf{v}}$ とのなす角の正弦 $\sin \angle(\mathbf{v}, \hat{\mathbf{v}})$ について, その最大値をマシンイプシロン ϵ の定数倍程度の微小値で抑えることであろう³. しかし, 後退誤差解析と相対的摂動論 [21][32][33] とを組み合わせた標準的な誤差評価方法 [10] に基づくと, これは一般には不可能である. なぜなら, 後退誤差解析により示せるのは $\hat{\mathbf{v}}$ が T の各成分に小さな相対的摂動を加えた行列 \hat{T} の厳密な固有ベクトルになっているということであるが, T に対する小さい相対的摂動が, 固有値・固有ベクトルに対する大きな相対的摂動, すなわち固有ベクトルの角度の大きなずれを引き起こす例が知られている [13][37] からである.

一方, 対称三重対角行列 T は, 対角要素が 1 の下二重対角行列 L と対角行列 D とにより, LDL^t と表現することもできる. この場合, もしこの行列が正定値, すなわち D の要素がすべて正ならば, L と D の要素に対する小さい摂動は, 固有値・固有ベクトルの小さな相対的摂動しか引き起こさないことが知られている [11]. 一般に, このような性質を持つ三重対角行列の表現を Relatively Robust Representation (RRR) と呼ぶ. なお, 正定値でない LDL^t であっても, ある固有値・固有ベクトルの組 (λ, \mathbf{v}) に対しては RRR となることがある. RRR のより正確な定義は, 4.3.2 項で述べる.

そこで, MR³ アルゴリズムでは, T を正定値になるように適当にシフトした行列 $T - \sigma I$ の LDL^t 分解を入力とし, その高精度な固有値・固有ベクトルを求めることを目標とする. LDL^t 分解が無限精度で計算された場合, LDL^t の固有値は T の固有値をシフトしたもの, 固有ベクトルは T の固有ベクトルそのものとなる. 実際には T から LDL^t を求める部分は有限精度で行うため, LDL^t は $T - \sigma I$ に対して微小な相対的摂動を加えた行列 $\hat{T} - \sigma I$ の修正コレスキー分解となる. そのため, $\hat{T} - \sigma I = LDL^t$ の固有ベクトルと $T - \sigma I$ の固有ベクトルとは大きな角度のずれがある可能性もあるが, MR³ アルゴリズムではそのことは不問とする.

³以下, 本論文では, 2 本のベクトルのなす角度 $\angle(\mathbf{v}, \hat{\mathbf{v}})$ を鋭角にとると約束し, $|\sin \angle(\mathbf{v}, \hat{\mathbf{v}})|$ の代わりに $\sin \angle(\mathbf{v}, \hat{\mathbf{v}})$ を用いる.

さらに、固有ベクトルの精度を保証する指導原理として、次の $\sin \Theta$ 定理 [36] を用いることにする。

定理 1 (sin Θ 定理) 実対称行列 LDL^t の近似固有値を $\hat{\lambda}$ 、対応する近似固有ベクトルを \hat{v} ($\|\hat{v}\|=1$) とする。また、 $\hat{\lambda}$ にもっとも近い LDL^t の真の固有値を λ とし、対応する真の固有ベクトルを v とする。このとき、

$$(3.1) \quad \sin \angle(v, \hat{v}) \leq \frac{\|LDL^t \hat{v} - \hat{v} \hat{\lambda}\|}{\text{gap}(\hat{\lambda})}$$

が成り立つ。ただし、 $\text{gap}(\hat{\lambda})$ は、 μ を λ 以外でもっとも $\hat{\lambda}$ に近い真の固有値とすると、

$$(3.2) \quad \text{gap}(\hat{\lambda}) = |\mu - \hat{\lambda}|$$

と定義される。

いま、マシンイプシロンを ϵ とするとき、 $\hat{\lambda}$ 、 \hat{v} を

$$(3.3) \quad \|LDL^t \hat{v} - \hat{v} \hat{\lambda}\| = O(n\epsilon)|\hat{\lambda}|$$

を満たすように計算できたとしよう [37]。これは、左辺に現れる $\hat{\lambda}$ が浮動小数点数であり、一般に $|\hat{\lambda}| \epsilon$ 程度の誤差を不可避免的に持つことを考慮すると、(因子 n は別として) 残差の値として達成できる最小の限界である。これを式 (3.1) に代入すると、

$$(3.4) \quad \sin \angle(v, \hat{v}) \leq \frac{O(n\epsilon)|\hat{\lambda}|}{\text{gap}(\hat{\lambda})} = \frac{O(n\epsilon)}{\text{relgap}(\hat{\lambda})}$$

ただし、

$$(3.5) \quad \text{relgap}(\hat{\lambda}) = \frac{|\mu - \hat{\lambda}|}{|\hat{\lambda}|}$$

が成り立つ。

式 (3.4) は、 $\hat{\lambda}$ の相対的ギャップ $\text{relgap}(\hat{\lambda})$ が大きい場合には、 \hat{v} が v の極めて精度の高い近似であることを示している。なお、真の固有ベクトルは直交性を持つから、相対的ギャップが大きい複数の固有値に対して、計算された固有ベクトルどうしは高い直交性をもつこともわかる。一方、 $\text{relgap}(\hat{\lambda})$ が小さい場合には、 $\sin \Theta$ 定理を用いる限り、計算された固有ベクトルと (LDL^t の) 真の固有ベクトルとのなす角を小さく抑えることはできない。しかしその場合でも、残差 $\|LDL^t \hat{v} - \hat{v} \hat{\lambda}\|$ が $O(n\epsilon) + O(\text{spdiam}[LDL^t]\epsilon)$ 程度で、かつ計算された他の固有ベクトルと高い直交性を持つ (内積が $O(n\epsilon)$ 程度となる) 固有ベクトルを計算することは可能であることがわかっている [16]。ここで、 $\text{spdiam}[LDL^t]$ は、 LDL^t の最大固有値と最小固有値との差 $\lambda_{\max}[LDL^t] - \lambda_{\min}[LDL^t]$ を表す。以上の考察より、固有ベクトルの精度に関する目標は次のように立てるのが適切であることがわかる。

(d') 固有値の相対的ギャップが大きいときは、 LDL^t の真の固有ベクトル v とのなす角度が式 (3.4) で抑えられるようなベクトル \hat{v} を求める。相対的ギャップが小さいときには、 $\|LDL^t \hat{v} - \hat{v} \hat{\lambda}\| = O(n\epsilon) + O(\text{spdiam}[LDL^t]\epsilon)$ 、かつ他の計算された固有ベクトルとの内積が $O(n\epsilon)$ 程度となる固有ベクトルを求める。

(a), (b), (c), (d') が MR³ アルゴリズムの目標となる。

4 MR³アルゴリズムの詳細

4.1 全体の流れ

前章で述べたように, MR³アルゴリズムでは, 式 (3.3) を満たすような $\hat{\lambda}$, \hat{v} が重要となる. そのためには, まず固有値を相対精度の意味で精度良く求める必要がある. このために用いる dqds 法を 4.2 節で説明する. 次に, 式 (3.3) を満たすベクトル \hat{v} を求める. このためには twisted 分解と呼ばれる手法を使う. これを 4.3 節で説明する. 以上で, 相対的ギャップが大きい場合については, 式 (3.4) を満たす高精度な固有ベクトルが求められることになる. 一方, 相対的ギャップが小さい場合については, LDL^t に対して単位行列の定数倍を加え, 固有値スペクトルの原点シフトを行うことにより, 相対的ギャップを拡大する. そして, その上で, 相対的ギャップが大きい場合のアルゴリズムを適用する. これについて, 4.4 節で説明する.

4.2 dqds 法による固有値の高精度計算

LDL^t が正定値の場合, その固有値は dqds (differential quotient-difference with shift) 法 [22][41] により相対誤差の意味で高精度に計算できる. 本節では, [22] に従って dqds 法を導出し, その誤差解析と収束性について述べる.

4.2.1 dqds 法の導出

まず, LR 法 [24] を対称正定値行列用に特化した方法として, LL^t 法を考える. 正定値な対称三重対角行列 T の固有値を求める場合, LL^t 法では $T^{(0)} \equiv T$ を $T^{(0)} = L^{(0)}L^{(0)t}$ とコレスキー分解し, $T^{(1)} = L^{(0)t}L^{(0)}$ とおく. さらに $T^{(1)} = L^{(1)}L^{(1)t}$ とコレスキー分解し, $T^{(2)} = L^{(1)t}L^{(1)}$ とする. この操作を繰り返してゆくと, $T^{(k)}$ は対角行列に収束する. なお, $L^{(k)}$ はすべて二重下三角行列であることは容易にわかる.

ここで, $T^{(k+1)}$ を陽的に作るのをやめ, $L^{(k)}$ から直接 $L^{(k+1)}$ を作ることを考える [22]. 定義より $L^{(k)t}L^{(k)} = L^{(k+1)}L^{(k+1)t}$, すなわち $(L^{(k)}L^{(k+1)-t})^t = (L^{(k)}L^{(k+1)-t})^{-1}$ だから $L^{(k)}L^{(k+1)-t}$ はある直交行列 Q に等しく, $L^{(k)} = QL^{(k+1)t}$ と書ける. いま, 二重下三角行列 $L^{(k)}$ に対して Givens 回転 [24] を繰り返すことにより上三角行列への変形を行うとすると, QR 分解の一意性より, 得られる上三角行列は $L^{(k+1)t}$ そのものである. したがって, 次のように Givens 回転を繰り返すことにより $L^{(k)}$ から直接 $L^{(k+1)t}$ を求めるアルゴリズムが得られる. これを orthogonal qd 法 (oqd 法) と呼ぶ. なお, アルゴリズム中では, L の対角成分を a_1, a_2, \dots, a_n , 副対角成分を b_1, b_2, \dots, b_{n-1} に格納するとする. これらの配列は上書きされる.

[アルゴリズム 1: oqd 法]

```

 $\bar{a} := a_1$ 
for  $i=1, n-1$ 
   $\hat{a}_i := \sqrt{\bar{a}^2 + b_i^2}$ 
   $\hat{b}_i := b_i * (a_{i+1}/\hat{a}_i)$ 
   $\bar{a} := \bar{a} * (a_{i+1}/\hat{a}_i)$ 
end for
 $\hat{a}_n := \bar{a}$ 

```

oqd 法において、すべての代入式の両辺を 2 乗し、 $q_i = a_i^2$, $e_i = b_i^2$ とおくと、次のように平方根を取り除いたアルゴリズムができる。これを differential qd 法 (dqd 法) と呼ぶ。

[アルゴリズム 2: dqd 法]

```

 $d := q_1$ 
for  $i=1, n-1$ 
   $\hat{q}_i := d + e_i$ 
   $\hat{e}_i := e_i * (q_{i+1}/\hat{q}_i)$ 
   $d := d * (q_{i+1}/\hat{q}_i)$ 
end for
 $\hat{q}_n := d$ 

```

アルゴリズム 2 は dqd 法の 1 反復であるが、この反復を繰り返すことにより、 q_i は $T = LDL^t$ の第 i 番目の固有値に収束する。oqd 法、dqd 法の著しい特徴は、減算がないことである。しかも、定義より、 q_i , e_i の初期値はすべて非負の値である。したがって、桁落ちによる精度悪化は生じず、すべての固有値が極めて高い精度で計算できる [22]。

実際には、収束を加速するため、原点シフトを用いる。シフト量を τ^2 とするとき、シフト付きのアルゴリズムは次のようになる。これを differential qd with shift 法 (dqds 法) と呼ぶ。

[アルゴリズム 3: dqds 法]

```

 $d := q_1 - \tau^2$ 
for  $i=1, n-1$ 
   $\hat{q}_i := d + e_i$ 
   $\hat{e}_i := e_i * (q_{i+1}/\hat{q}_i)$ 
   $d := d * (q_{i+1}/\hat{q}_i) - \tau^2$ 
end for
 $\hat{q}_n := d$ 

```

高い精度を保つため、シフト τ^2 は最小固有値より小さく取るのが望ましいとされている。

4.2.2 誤差解析

dqds 法の 1 反復を無限精度で行うと, 入力行列 $L^{(k)}L^{(k)t}$ の固有値 $\lambda_i^{(k)}$ と出力行列 $L^{(k+1)}L^{(k+1)t}$ の固有値 $\lambda_i^{(k+1)}$ とは, $\lambda_i^{(k+1)} = \lambda_i^{(k)} - \tau^2$ の関係にある. 実際の計算は有限精度で行うため, その影響で固有値がどの程度ずれるか, 誤差解析を行う必要がある. 誤差解析の方法としては, 後退誤差解析がよく知られているが, dqds 法においては, この解析を行うことは困難である. そこで, 代わりに混合型誤差解析という方法を用いる. まず, 次のことが成り立つ.

定理 2 ([22], Theorem 4) 下二重対角行列 \bar{L} に対して浮動小数点演算による dqds 法の 1 反復を適用して得られる行列を \hat{L} とする. また, 計算の途中でオーバーフロー, アンダーフローは起きないとする. このとき, \bar{L} に微小な相対的摂動を加えて得られる下二重対角行列 \tilde{L} , および微小な相対的摂動を加えることにより \hat{L} になる下二重対角行列 \check{L} が存在し, \check{L} は \tilde{L} に正確な dqds 法の 1 反復を施して得られる行列になっている. より詳しくは, \tilde{L} は \bar{L} の各対角要素を最大 3ulps, 副対角要素を最大 1ulps 変化させることによって得られ, \hat{L} は \check{L} の各要素を最大 2ulps 変化させることによって得られる.

$$\begin{array}{ccc} \bar{L} & \xrightarrow{\text{dqds in floating point arithmetic}} & \hat{L} \\ \downarrow & & \uparrow \\ \tilde{L} & \xrightarrow{\text{dqds in exact arithmetic}} & \check{L} \end{array}$$

ここで, 1ulps の変化とは, 浮動小数点数の仮数部の最後の 1 ビット分が変化することであり, 倍率で言うと $1 + \eta$ 倍 ($|\eta| \leq \epsilon$) の変化に相当する.

次に, この (\tilde{L} および \check{L} に対する) 相対的摂動が固有値に与える影響を考える. LL^t の固有値は L の特異値の 2 乗であることを考慮し, 定理 2 と下二重対角行列に対する微小な相対的摂動は特異値に微小な相対的摂動しか及ぼさないという Demmel-Kahan の定理 [11] を用いると, 次の結果が得られる.

定理 3 ([22], Corollary 1) $\bar{L}\bar{L}^t$, $\tilde{L}\tilde{L}^t$, $\check{L}\check{L}^t$, $\hat{L}\hat{L}^t$ の特異値をそれぞれ $\bar{\lambda}_i$, $\tilde{\lambda}_i$, $\check{\lambda}_i$, $\hat{\lambda}_i$ とすると, これらの間には次の関係が成り立つ.

$$\begin{aligned} \tilde{\lambda}_i &= \bar{\lambda}_i \exp(4(n-1)\epsilon_1^{(i)}) \\ \check{\lambda}_i &= \tilde{\lambda}_i - \tau^2 \\ \hat{\lambda}_i &= \check{\lambda}_i \exp(4(n-1)\epsilon_2^{(i)}) \end{aligned}$$

ただし, $|\epsilon_1^{(i)}| \leq \epsilon$, $|\epsilon_2^{(i)}| \leq \epsilon$ である.

これより, 本来成り立って欲しい関係 $\hat{\lambda}_i = \bar{\lambda}_i - \tau^2$ は, 高い相対精度で成り立つことがわかる. 定理 3 は 1 反復における固有値の関係式であるが, dqds 法は QR 法と同様, 適切なシフトを行えば定数回で 1 個の固有値を求めることができるため, 定理 3 の示す誤差の定数倍

程度の誤差で固有値を計算することが原理的には可能である。ただし全固有値を求める場合には、QR法と同様、1個の固有値が求まるたびにデフレーションを行い、合計で $O(n)$ 回の dqds 反復を行うため、最後に求まる固有値の相対誤差は、最初に求まる固有値の相対誤差に比べて最大で n 倍程度悪くなると考えられる。

QR法で下二重対角行列の特異値を求める場合でも、シフトなしの場合は最小特異値を高い精度で求められることが知られている [11]。これに対して dqds 法では、全特異値を高い相対精度で求めることができ、かつ、収束を速めるためのシフトを行ってもその性質が失われない。これが dqds 法の大きな利点である。ただし、このような高精度計算が可能なのは [22] で提案された differential form の場合に限られており、Rutishauser によって提案されたオリジナルの qd 法 [43] はこのような優れた性質を持たないことが知られている。

4.2.3 収束性

oqd 法の収束性については、シフトがない場合の収束定理が次のように与えられている。

定理 4 ([22], Theorem 5) 全要素が正の下二重対角行列 $L = L^{(0)}$ を入力としてシフトなしの oqd 法を無限精度演算で行うと、得られる下二重対角行列の列 $\{L^{(k)}\}_{k=0}^{\infty}$ は $k \rightarrow \infty$ のとき対角行列に収束し、その対角成分には L の特異値が大きい順に並ぶ。

oqd 法と dqd 法とは無限精度演算では等価であるから、この定理は dqd 法に対しても成り立つ。ただし、実用上重要なシフト付きの dqds 法に対する収束証明は、[22] では与えられていない。また、収束に対する有限精度演算の影響は論じられているが、厳密な言明はなされていない。

4.2.4 実装上の注意など

dqds 法の実装に当たっては、シフトの選び方、デフレーションの判定基準、オーバーフロー／アンダーフローの扱いなどが重要である。これらについては、[41] に詳しく述べられている。

また、 LDL^t が正定値でない場合、dqds 法では高い精度で固有値を求められる保証がないが、この場合にも、dqds 変換を用いた二分法 [15] を使えば、

$$(4.1) \quad |\lambda - \hat{\lambda}| \leq O(n\epsilon\hat{\lambda})$$

を満たす固有値 λ の近似値 $\hat{\lambda}$ を求めることが可能である。

なお、[22] では、dqds 法は並列化が可能であると述べられているが、そこで提案されている並列化手法はいわゆる parallel prefix による並列化であり、数値的不安定を引き起こす恐れがあるので実用的ではない。dqds 法の優れた性質を保つ並列化手法の開発は、まだ今後の課題と思われる。

4.3 twisted 分解による固有ベクトルの高精度計算 (相対的ギャップが大きい場合)

いま, 相対的ギャップが大きな LDL^t の固有値に対して, その近似固有値が式 (4.1) を満たすように高い相対精度で求められたと仮定する. このとき, 式 (3.3) を満たす近似固有ベクトル \hat{v} を計算することを考える. まず無限精度で演算が可能な場合のアルゴリズムを述べ, 次に有限精度の影響を考える.

4.3.1 無限精度演算の場合

固有ベクトルの計算には, 逆反復法 [24][42] をベースとした方法を用いる. 通常の逆反復法と異なるのは, 残差が小さくなるように右辺ベクトルを適切に選ぶことと, 方程式を解くのに LU 分解の代わりに twisted 分解と呼ばれる方法を使うことである. 具体的には, 第 k 要素 (k の選び方は後述) のみが定数, 他は 0 であるベクトルを右辺とし,

$$(4.2) \quad (LDL^t - \hat{\lambda}I) \mathbf{z}^{(k)} = \mathbf{e}_k \gamma_k$$

を解く [39][15]. ここで, \mathbf{e}_k は第 k 要素が 1, その他が 0 のベクトルであり, $\mathbf{z}^{(k)}$ が求める近似固有ベクトルである. 得られる近似固有ベクトルは k の値によって異なるため, 上付き添字 (k) を付けた. また, γ_k は \mathbf{z}_k の第 k 成分 $z^{(k)}(k)$ が 1 になるようにするための定数である. $LDL^t - \hat{\lambda}$ の逆行列の要素を用いると, $\gamma_k = (LDL^t - \hat{\lambda})_{kk}^{-1}$ と書けることに注意する.

このとき, 近似固有ベクトルの 2 ノルムが 1 になるように規格化を行った場合の残差は次のように書ける.

$$(4.3) \quad \frac{\| (LDL^t - \hat{\lambda}I) \mathbf{z}^{(k)} \|}{\| \mathbf{z}^{(k)} \|} = \frac{|\gamma_k|}{\| \mathbf{z}^{(k)} \|}$$

さらに, λ に属する真の固有ベクトル \mathbf{v} の第 k 成分を $v(k)$ と書くと, 右辺は $|\lambda - \hat{\lambda}|/|v(k)|$ で抑えられることが示せる ([15], Theorem 11). したがって, $|v(k)|$ が最大になる k を選べば残差の上界を最小にできる. $\| \mathbf{v} \| = 1$ より絶対値最大の要素は明らかに $1/\sqrt{n}$ より大きいから, このように k を選ぶと

$$(4.4) \quad \frac{\| (LDL^t - \hat{\lambda}I) \mathbf{z}^{(k)} \|}{\| \mathbf{z}^{(k)} \|} \leq \sqrt{n} |\lambda - \hat{\lambda}|$$

が成り立つ.

実際には真の固有ベクトル \mathbf{v} はわからないから, 最適な k を選ぶことは困難であるように思えるが, $\hat{\lambda} \rightarrow \lambda$ のとき, 任意の添字 k, l に対して

$$(4.5) \quad \frac{\gamma_k}{\gamma_l} \rightarrow \frac{v(l)^2}{v(k)^2}$$

が成り立つことが示せる ([39], Lemma 11). したがって, $|v(k)|$ を最大化する代わりに $|\gamma_k|$ を最小化する k を選ぶことができる. γ_k ($1 \leq k \leq n$) の値は, $LDL^t - \hat{\lambda}I$ の LDL^t 分解と

UDU^t 分解

$$(4.6) \quad LDL^t - \hat{\lambda}I = L_+ D_+ L_+^t$$

$$(4.7) \quad LDL^t - \hat{\lambda}I = U_- D_- U_-^t$$

が求めれば, その成分を使って簡単な式により計算することができる [39]. ここで, [39][15] の記法に従い, LDL^t 分解により得られる量には下付き添字 $+$, UDU^t 分解により得られる量には上付き添字 $-$ を付けた. なお, L_+, D_+, U_-, D_- は上式の左辺を計算してから LDL^t 分解あるいは UDU^t 分解を行うという通常の方法ではなく, 相対的摂動に弱い三重対角形式を避けて高精度を保つため, $L, D, \hat{\lambda}$ から直接 dstqds 変換 (differential stationary qd with shift 変換; L_+, D_+ の場合) あるいは dqds 変換 (U_-, D_- の場合) によって計算する [15]. 以上により, 最適な k を選ぶ指針が得られた.

k を決めた後で $\mathbf{z}^{(k)}$ を計算するには, twisted 分解を用いる. いま, 二重下三角行列 L_+ , 二重上三角行列 U_- の副対角要素をそれぞれ $l_i \equiv (L_+)_{i+1, i}$, $u_i = (U_-)_{i, i+1}$ と書き, 第 1 列目から第 $k-1$ 列目までは L_+ , 第 $k+1$ 列目から第 n 列目までは U_- に等しく, 第 k 列目は (k, k) 成分にのみ 1 を持つような行列

$$(4.8) \quad N_k \equiv \begin{pmatrix} 1 & & & & & & & & \\ & l_1 & & & & & & & \\ & & \ddots & & & & & & \\ & & & \ddots & & & & & \\ & & & & 1 & & & & \\ & & & & & l_{k-1} & 1 & u_{k+1} & \\ & & & & & & & 1 & \ddots \\ & & & & & & & & \ddots & u_{n-1} \\ & & & & & & & & & & 1 \end{pmatrix}$$

を定義する. twisted 分解とは, この N_k を用いて $LDL^t - \hat{\lambda}I$ を

$$(4.9) \quad LDL^t - \hat{\lambda}I = N_k D_k N_k^t$$

と分解することである. ここで, D_k は対角行列であり, D_+, D_- の要素と γ_k を用いて

$$(4.10) \quad D_k = \text{diag}(D_+(1), \dots, D_+(k-1), \gamma_k, D_-(k+1), \dots, D_-(n))$$

と表される. このような分解が可能であること, および twisted 分解の性質については, [39][15] を参照されたい. twisted 分解を使うと, 式 (4.2) は

$$(4.11) \quad N_k D_k N_k^t \mathbf{z}^{(k)} = \mathbf{e}_k \gamma_k$$

となる. さらに, $N_k \mathbf{e}_k = \mathbf{e}_k$, $D_k \mathbf{e}_k = \mathbf{e}_k \gamma_k$ を使うと, この方程式は

$$(4.12) \quad N_k^t \mathbf{z}^{(k)} = \mathbf{e}_k$$

と簡単化できる. N_k^t の形より, 式 (4.12) は第 k 番目の式から始めて上下両方向に向かって解くことができる. このとき, 必要な演算は乗算のみであるため, 計算は極めて高精度に行える.

以上をまとめると、近似固有ベクトル計算のアルゴリズムは次のようになる。このアルゴリズムは、[15]において Getvec と呼ばれている。

[アルゴリズム 4: twisted 分解による固有ベクトルの計算 (Getvec)]

1. dstqds 変換により、分解 $LDL^t - \hat{\lambda}I = L_+ D_+ L_+^t$ を行う。
2. dqds 変換により、分解 $LDL^t - \hat{\lambda}I = U_- D_- U_-^t$ を行う。
3. L_+, D_+, U_-, D_- より γ_k ($1 \leq k \leq n$) を計算し、 $|\gamma_k|$ が最小になる k を選ぶ。この k に対して $LDL^t - \hat{\lambda}I$ の $N_k D_k N_k^t$ 分解を構成する。
4. $N_k^t \mathbf{z}^{(k)} = \mathbf{e}_k \gamma_k$ を解き、近似固有ベクトル $\mathbf{z}^{(k)}$ を求める。

Getvec により得られたベクトル $\mathbf{z}^{(k)}$ は、 $\hat{\lambda}$ が λ に十分近ければ、式 (4.4) の不等式を満たす。

4.3.2 有限精度の場合

前項では演算が無限精度で行われる場合を考えたが、次に、演算が有限精度で行われることによる影響を考える。このため、入力行列 LDL^t から twisted 分解 $N_k D_k N_k^t$ を計算する部分について、4.2.2 項で導入した混合型誤差解析を行う。その結果、次の定理が得られる。

定理 5 ([15], Fig. 5) 入力行列 LDL^t に対して浮動小数点演算による dtwqds 変換⁴を適用して得られる twisted 分解を $\hat{N}_k \hat{D}_k \hat{N}_k^t$ とする。このとき、 L, D にそれぞれ微小な相対的摂動を加えて得られる \bar{L}, \bar{D} 、および微小な相対的摂動を加えることにより \hat{N}_k, \hat{D}_k になる行列 \bar{N}_k, \bar{D}_k が存在し、 $\bar{L} \bar{D} \bar{L}^t - \hat{\lambda}I = \bar{N}_k \bar{D}_k \bar{N}_k^t$ が厳密に成り立つ。より詳しくは、 $\bar{L}, \bar{D}, \hat{N}_k, \hat{D}_k$ はそれぞれ $L, D, \bar{N}_k, \bar{D}_k$ の各要素を $3 \sim 3\frac{1}{2}$ ulps, $1 \sim 4$ ulps, $3 \sim 4\frac{1}{2}$ ulps, 2 ulps 変化させることによって得られる。

$$\begin{array}{ccc} LDL^t & \xrightarrow{\text{dtwqds in floating point arithmetic}} & \hat{N}_k \hat{D}_k \hat{N}_k^t \\ \downarrow & & \uparrow \\ \bar{L} \bar{D} \bar{L}^t & \xrightarrow{\text{exact dqds: } \bar{L} \bar{D} \bar{L}^t - \hat{\lambda}I = \bar{N}_k \bar{D}_k \bar{N}_k^t} & \bar{N}_k \bar{D}_k \bar{N}_k^t \end{array}$$

この定理を使って誤差解析を行うため、計算された固有ベクトルの誤差が生じる原因を次の3つに分けて考える [15].

- (a) LDL^t の厳密な固有ベクトル \mathbf{v} と $\bar{L} \bar{D} \bar{L}^t$ の厳密な固有ベクトル $\bar{\mathbf{v}}$ との間の差。
- (b) $\bar{L} \bar{D} \bar{L}^t$ の厳密な固有ベクトル $\bar{\mathbf{v}}$ と、 $\bar{L} \bar{D} \bar{L}^t - \hat{\lambda}I$ に対して 4.3.1 項のアルゴリズムを無限精度で適用して得られるベクトル $\bar{\mathbf{z}}^{(k)}$ との間の差。

⁴twisted 分解を計算するため、 L_+, D_+ の部分を dstqds 変換、 U_-, D_- の部分を dqds 変換を用いて計算することを、[15] では dtwqds 変換と呼んでいる。

(c) $\bar{N}_k^t \bar{z}^{(k)} = \mathbf{e}_k$ を無限精度で解いて得られるベクトル $\bar{z}^{(k)}$ と $\hat{N}_k^t \hat{z}^{(k)} = \mathbf{e}_k$ を有限精度で解いて得られるベクトル $\hat{z}^{(k)}$ との差 (\bar{N}_k と \hat{N}_k の違いによる影響と, 有限精度で解く際の丸め誤差の影響の両方を含む).

このうち, (b) については, 無限精度での計算であることから 4.3.1 項の結果がそのまま使え, $\hat{\lambda}$ が $\bar{L}\bar{D}\bar{L}^t$ の固有値 $\bar{\lambda}$ の十分良い近似値であるならば, 式 (4.3) の後で述べたことより

$$(4.13) \quad \frac{\|(\bar{L}\bar{D}\bar{L}^t - \hat{\lambda}I)\bar{z}^{(k)}\|}{\|\bar{z}^{(k)}\|} \leq \frac{|\bar{\lambda} - \hat{\lambda}|}{|\bar{v}^{(k)}|}$$

が成り立つ. したがって, 式 (3.1) より,

$$(4.14) \quad \sin \angle(\bar{z}^{(k)}, \bar{\mathbf{v}}) \leq \frac{|\bar{\lambda} - \hat{\lambda}|}{|\bar{v}^{(k)}| \text{gap}(\hat{\lambda})}$$

が言える. また, (c) については, 4.3.1 項で述べたように $\hat{N}_k^t \hat{z}^{(k)} = \mathbf{e}_k$ が乗算のみを用いて解けることより簡単に誤差解析を行うことができ,

$$(4.15) \quad \sin \angle(\hat{z}^{(k)}, \bar{z}^{(k)}) \leq \frac{(1 + \epsilon)^{5(n-1)} - 1}{(1 - \epsilon)^{5(n-1)}}$$

が示せる ([15], Corollary 14).

もっとも難しいのは (a) である. \bar{L} , \bar{D} はそれぞれ L , D の要素に微小な相対的摂動を加えることにより得られるが, 一般にはこの摂動により, LDL^t の固有値 λ , 固有ベクトル \mathbf{v} は大きな相対的摂動を受ける可能性がある. そこで, 次のように Relatively Robust Representation の概念を導入する [40][15].

定義 1: Relatively Robust Representation LDL^t の固有値・固有ベクトルの組を (λ, \mathbf{v}) とする. LDL^t が (λ, \mathbf{v}) に対して Relatively Robust Representation (RRR) であるとは, L の副対角要素 $l_i \equiv L_{i+1,i}$, D の対角要素 $d_i \equiv D_{ii}$ に微小な相対的摂動 $l_i \rightarrow l_i(1 + \eta_i)$, $d_i \rightarrow d_i(1 + \delta_i)$ ($|\eta_i| < \xi$, $|\delta_i| < \xi$) を加えたとき, 固有値, 固有ベクトルの変動が次のように抑えられることである.

$$(4.16) \quad \frac{|\delta\lambda|}{|\lambda|} \leq K_1 n \xi, \quad \lambda \neq 0$$

$$(4.17) \quad \sin \angle(\mathbf{v}, \mathbf{v} + \delta\mathbf{v}) \leq \frac{K_2 n \xi}{\text{relgap}(\lambda)}$$

ここで, K_1 , K_2 は小さい (高々 100 程度の) 定数とする.

アルゴリズム Getvec では, 入力 LDL^t が計算したい固有値・固有ベクトルの組 (λ, \mathbf{v}) に対し, RRR となっていることを要請する. このとき, L , D から \bar{L} , \bar{D} への摂動は高々 4ulp 程度であることを考慮すると, 式 (4.17) における ξ は ϵ の数倍程度と小さい. また, 固有値の相対的ギャップは大きいと仮定しているから, 分母の $\text{relgap}(\lambda)$ は大きい. したがって, 式 (4.17) より, (a) の差について,

$$(4.18) \quad \sin \angle(\bar{\mathbf{v}}, \mathbf{v}) = O(n\epsilon)$$

が成り立つ。

それでは、 LDL^t が組 (λ, \mathbf{v}) に対して RRR であるというのは、容易に満たせる条件なのだろうか。MR³ アルゴリズムでは、3章で述べたように、まず T を正定値になるように原点シフトした行列 $T - \sigma I$ に対して LDL^t を計算する。この LDL^t は正定値であるから、任意の固有値・固有ベクトルの組に対して RRR である [11]。したがって、最初の LDL^t において相対的ギャップが大きいと判定され、その表現を使って計算される固有値・固有ベクトルの組については問題がない。一方、相対的ギャップが小さいと判定された固有値に対しては、4.4 節に述べるように、 LDL^t に対して再度適当な大きさ σ_c の原点シフトを行うことにより、その固有値に対する相対的ギャップを拡大する。ところが、この結果得られる新しい表現 $L_c D_c L_c^t = LDL^t - \sigma_c I$ は一般に正定値ではないため、問題とする固有値・固有ベクトルの組に対して RRR であるという保証がない。

実際には、相対的ギャップを拡大するような σ_c の値にはある程度の任意性があるため、ある σ_c に対する $L_c D_c L_c^t$ が RRR でなくても、 σ_c の値を変えることにより RRR が得られるという可能性はある。また、Dhillon と Parlett は [16] において、「 $L_c D_c L_c^t$ が特異に極めて近い行列ならば、それは 0 に近い固有値に対して RRR となっている」という予想を述べ、RRR を見つけるのがそれほど困難ではないと主張している。しかし理論的には、ある固有値・固有ベクトルの値に対し、相対的ギャップが大きく、かつ RRR となるようなシフトの値が必ず存在するかどうか、また、存在するならばそれをどうやって見つけるかは未解決であることに注意する必要がある。

なお、上記の RRR の定義は、ある LDL^t がある (λ, \mathbf{v}) に対して RRR であるかどうかを計算機上で容易に判定できる形にはなっていない。そこで [15] では、 LDL^t が微小な相対的摂動を受けたときに固有ベクトル \mathbf{v} がどの程度変化するかを指標である相対的条件数 $\text{relcond}(\mathbf{v})$ を次のように定義している。

いま、 $L|D|^{1/2}$ ($|D|$ は D の要素の絶対値を要素とする行列を表す) の双曲的特異値分解 [15][38] を $L|D|^{1/2} = V\Sigma P^t$ とし、

$$(4.19) \quad \Sigma = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_n)$$

$$(4.20) \quad P = (\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_n)$$

とする。 σ_i は LDL^t の固有値 λ_i と $\sigma_i = \sqrt{|\lambda_i|}$ の関係にあるため、 $\sigma_j = \sqrt{|\lambda|}$ となる j が存在する。そこで、この j に対して

$$(4.21) \quad \delta_{ji} \equiv \frac{|\lambda_j - \lambda_i|}{\sigma_j + \sigma_i} \quad (i = 1, \dots, n)$$

$$(4.22) \quad \text{rgap}_j \equiv \min_{i \neq j} \frac{\delta_{ji}}{\sigma_j}$$

$$(4.23) \quad \|\mathbf{m}_j\|^2 \equiv \sum_{i \neq j} \left(\frac{\|\mathbf{p}_i\|^2}{\delta_{ji}/\sigma_j} \right)^2$$

とおき、 \mathbf{v} の相対的条件数を

$$(4.24) \quad \text{relcond}(\mathbf{v}) \equiv 2 \|\mathbf{p}_j\| \|\mathbf{m}_j\| + \frac{1}{2}$$

により定義する.

この $\text{relcond}(\mathbf{v})$ を用いると, (a) による差を

$$(4.25) \quad \sin \angle(\bar{\mathbf{v}}, \mathbf{v}) \leq \frac{\epsilon_* \text{relcond}(\mathbf{v})}{1 - \epsilon_*}$$

と抑えることができる. ただし, $\epsilon_* \equiv (1 + \epsilon)^{6n-1} - 1$ である.

$\text{relcond}(\mathbf{v})$ を正確に計算しようとする, $L|D|^{1/2}$ の双曲的特異値分解が必要となって計算量が大きいため, [15] では次のような上界を導いている.

$$(4.26) \quad \text{relcond}(\mathbf{v}) \leq \frac{\text{cond}_F(L|D|^{1/2})}{\text{rgap}_j} + \frac{1}{2}.$$

ここで, $\text{cond}_F(L|D|^{1/2})$ はフロベニウスノルムによる $L|D|^{1/2}$ の条件数である. また, rgap_j は, LDL^t の固有値 $\{\lambda_i\}_{i=1}^n$ がわかっているならば, 関係式 $\sigma_i = \sqrt{|\lambda_i|}$ と式 (4.21), (4.22) を用いて容易に計算可能である.

式 (4.14), (4.15), (4.25) より次の定理が得られる.

定理 6 ([15], Theorem 15) LDL^t の真の固有ベクトルを \mathbf{v} , アルゴリズム Getvec により有限精度の計算で得られた固有ベクトルを $\hat{\mathbf{z}}^{(k)}$ とする. このとき, \mathbf{v} と $\hat{\mathbf{z}}^{(k)}$ のなす角は次のように抑えられる.

$$(4.27) \quad \sin \angle(\hat{\mathbf{z}}^{(k)}, \mathbf{v}) \leq \frac{(1 + \epsilon)^{5(n-1)} - 1}{(1 - \epsilon)^{5(n-1)}} + \frac{|\bar{\lambda} - \hat{\lambda}|}{|\bar{v}(k)| \text{gap}(\hat{\lambda})} + \frac{\epsilon_* \text{relcond}(\mathbf{v})}{1 - \epsilon_*}$$

いま, (i) $\text{relgap}(\lambda) = O(1)$, (ii) $|\bar{\lambda} - \hat{\lambda}| = O(n\epsilon\hat{\lambda})$, (iii) $\text{relcond}(\mathbf{v}) = O(1)$ の 3 つの条件が成り立つとする. このとき, 式 (4.27) 右辺の第 1 項は約 $5n\epsilon$, 第 2 項は $O(n\sqrt{n\epsilon})$, 第 3 項は $6n\epsilon$ 程度となる. したがって, 右辺は $O(n\sqrt{n\epsilon})$ となり, 有限精度の Getvec は, $\sin \angle(\hat{\mathbf{z}}^{(k)}, \mathbf{v}) = O(n\sqrt{n\epsilon})$ という精度で固有ベクトルを計算できることがわかる. 真の固有ベクトルは直交するから, これより, 異なる固有値に属する計算された固有ベクトルどうしは, 内積が $O(n\sqrt{n\epsilon})$ 程度と高い直交性を持つこともわかる. したがって, 直交化は不要である. なお, Dhillon と Parlett は [15][16] において, 式 (4.27) の第 2 項が残りの項よりも小さく, 右辺全体が $O(n\epsilon)$ となる場合も多いこと, $O(n\sqrt{n\epsilon})$ という評価は第 2 項の複数の因子の上界を別々に求めているため, 過大評価である可能性もあることを述べている.

4.4 相対的ギャップが小さい場合の固有ベクトル計算

LDL^t の (計算された) 固有値の 1 つを $\hat{\lambda}$ とし, $\text{relgap}(\hat{\lambda})$ があるしきい値 tol より小さいとする⁵. このとき, $\hat{\lambda}$ はある固有値のクラスター (正確な定義は 4.4.1 項) に属する. $\hat{\lambda}$ に対応する固有ベクトルを求める場合, 直接 Getvec を使ったのでは, 式 (4.27) 右辺の第 2 項が大

⁵[16] では, しきい値の例として $\text{tol} = 10^{-3}$ を挙げている.

きくなるため、小さな上界が得られない。そのため、計算された固有ベクトルが他の固有ベクトルと直交することも期待できない。

そこで、この場合は LDL^t に対して適当な大きさ σ_c の原点シフトを行い、対象とする固有値が大きな相対的ギャップを持つような、新しい表現 $L_c D_c L_c^t = LDL^t - \sigma_c I$ を求める。そして、 $L_c D_c L_c^t$ の固有値 $\hat{\lambda} - \sigma_c$ に属する固有ベクトルを Getvec により求める。なお、 $L_c D_c L_c^t$ は、twisted 分解の場合と同様、 LDL^t から直接 dstqds 変換により計算する。このように複数の表現を用いて固有ベクトルの計算を行うことが、Multiple Relatively Robust Representations アルゴリズムの名前の由来となっている。

3章で述べた目標 (d') を鑑みると、このアルゴリズムにおいて、新しい表現から計算した固有ベクトルが元の表現に対して小さい残差を持つかどうか、また、以前に計算した固有ベクトルと高い精度で直交するかどうかは重要である。そこで以下では、まず固有値のクラスターを定義し、クラスターに対する RRR の概念を導入する。次に、それを用いて、新しい表現から計算した固有ベクトルの直交性と残差に関する理論的結果を紹介する。最後に、新しい表現の下でなお固有値のクラスターが残る場合、固有ベクトルの計算を系統的に行うための representation tree について述べる。

4.4.1 固有値のクラスター

いま、 LDL^t の固有値を小さい順に $\lambda_1, \lambda_2, \dots, \lambda_n$ とし、 $\hat{\lambda}_j$ を λ_j の (高い相対精度で計算された) 近似値とする。このとき、 $2 \leq j \leq n$ について、 $|\hat{\lambda}_j - \hat{\lambda}_{j-1}| / |\hat{\lambda}_j| < \text{tol}$ のとき、 $\hat{\lambda}_{j-1}$ と $\hat{\lambda}_j$ とは同じクラスターに属すると定義する。これにより、 n 個の固有値の集合は、いくつかのクラスターの直和となる。クラスターの要素数が 1 の場合、そのクラスターに属する固有値を孤立固有値と呼ぶ。孤立固有値は相対的ギャップが大きい固有値であるから、対応する固有ベクトルは 4.3 節のアルゴリズムで計算できる。そこで、要素数が 2 以上のクラスターのみを考えればよい。

いま、1つのクラスターを考え、それに属する固有値の番号の集合を Γ とする。以下では Γ をこのクラスターと同一視し、このクラスターのことを Γ と呼ぶことにする。また、 Γ に対応する LDL^t の固有ベクトル $\{\mathbf{v}_j\}_{j \in \Gamma}$ の張る部分空間を次のように定義する。

$$(4.28) \quad S_\Gamma^{LDL^t} = \text{span}(\mathbf{v}_j), \quad j \in \Gamma$$

さらに、表現 LDL^t の下での Γ の相対的ギャップを次のように定義する。

$$(4.29) \quad \text{relgap}(\Gamma; LDL^t) = \min_{j \notin \Gamma, i \in \Gamma} |\lambda_i - \lambda_j| / |\lambda_i|$$

この定義とクラスターの定義より、 $\text{relgap}(\Gamma; LDL^t)$ は tol 程度以上の大きな値であることに注意する。

これらを用いて、固有値クラスターに対する Relatively Robust Representation の概念を次のように導入する [16].

定義2: 固有値クラスターに対する Relatively Robust Representation LDL^t の固有値クラスターの1つを Γ とする. LDL^t が Γ に対して Relatively Robust Representation (RRR) であるとは, L の副対角要素 $l_i \equiv L_{i+1,i}$, D の対角要素 $d_i \equiv D_{ii}$ に微小な相対的摂動 $l_i \rightarrow l_i(1+\eta_i)$, $d_i \rightarrow d_i(1+\delta_i)$ ($|\eta_i| < \xi$, $|\delta_i| < \xi$) を加えて得られる行列を $\bar{L}\bar{D}\bar{L}^t$ とするとき, Γ に属する固有値の変動と, 部分空間 $S_\Gamma^{LDL^t}$ の変動が次のように抑えられることである.

$$(4.30) \quad \frac{|\delta\lambda_j|}{|\lambda_j|} \leq K_1 n \xi, \quad j \in \Gamma, \quad \lambda_j \neq 0$$

$$(4.31) \quad \sin \angle(S_\Gamma^{LDL^t}, S_\Gamma^{\bar{L}\bar{D}\bar{L}^t}) \leq \frac{K_3 n \xi}{\text{relgap}(\Gamma; LDL^t)}$$

ここで, K_1, K_3 は小さい (高々100程度の) 定数とする. また, 2つの部分空間のなす角度は, それらの正準角 [24] のうち最大の角度と定義する.

定義1と比較すると, 定義2の式(4.31)では, 各固有ベクトルの変動が小さいことを要請しない点で条件を緩めているが, その代わりに右辺分母の $\text{relgap}(\Gamma; LDL^t)$ が tol 程度以上の大きな値であるため, 右辺は小さくなっている. 正定値の LDL^t は, 任意の固有値クラスター Γ に対して RRR となっていることが示せる [21].

4.4.2 表現の変換と固有ベクトルの計算

いま, LDL^t のある固有値クラスター Γ_c に属する固有値 λ_j に対して, 固有ベクトル \mathbf{v}_j を計算することを考える. ただし, LDL^t は Γ_c に対して RRR であるとする. 3節で述べた通り, MR³ アルゴリズムでは最初の表現は正定値であるから, 前項の最後で述べたことより, この条件は最初の表現については満たされている.

固有ベクトルの計算に当たっては, まず LDL^t に対して適当な大きさ σ_c の原点シフトを行った行列 $LDL^t - \sigma_c I$ を作り, これを dstqds 変換により分解して新しい表現 $L_c D_c L_c^t$ を求める. シフト量 σ_c は, 次の条件を満たすように決める [16]⁶.

(A) $L_c D_c L_c^t$ における固有値 $\lambda_j - \sigma_c$ の相対的ギャップが tol 以上となる.

(B) $L_c D_c L_c^t$ が固有値・固有ベクトルの組 $(\lambda_j - \sigma_c, \mathbf{v}_j)$ に対して RRR となっている.

(C) $L_c D_c L_c^t$ が Γ_c に対して RRR となっている.

(D) $L_c D_c L_c^t$ 分解における要素の増大が, ある式により抑えられる. 具体的には, Γ_c に対応する任意の計算された固有ベクトル $\hat{\mathbf{z}}_j$ ($j \in \Gamma_c$) に対し, 次の2つの式が成り立つ.

$$(4.32) \quad \|D_c \hat{\mathbf{z}}_j\| \leq C \cdot \text{spdiam}[LDL^t]$$

$$(4.33) \quad \|\dot{L}_c D_c \dot{L}_c^t \hat{\mathbf{z}}_j\| \leq C \cdot \text{spdiam}[LDL^t]$$

ここで, $\dot{L}_c = L_c - I$ であり, C は10程度の定数である.

⁶実際には, 条件(D)が満たされているか否かは固有ベクトルの計算後でないとわからないため, まず(A), (B), (C)のみを満たす σ_c に対して固有ベクトルを計算し, 条件(D)については事後的に調べる.

条件 (A), (B) は, 新しい表現 $L_c D_c L_c^t$ において固有値 $\lambda - \sigma_c$ に対応する固有ベクトルが Getvec により計算できるための条件である. 一方, 条件 (C), (D) は計算した固有ベクトルの直交性と残差を保証するための条件であり, 4.4.3 項と 4.4.4 項で用いる.

$L_c D_c L_c^t$ が求められたら, 近似固有値 $\lambda_j - \sigma_c$ を相対誤差の意味で高精度に計算し直す⁷. そして, これを用いて Getvec により対応する固有ベクトル \hat{z}_j を計算する.

4.4.3 計算された固有ベクトルの直交性

Getvec の性質より, こうして計算されたベクトル \hat{z}_j は新しい表現に対して精度の良い固有ベクトルになっている. しかし, 元の表現に対しては, 一般に精度の良い固有ベクトルとはなっていない. そのため, 元の表現から計算した (Γ に属さない固有値に対する) 固有ベクトルとの直交性を調べる必要がある.

このため, LDL^t から $L_c D_c L_c^t$ を計算する部分について, 混合型誤差解析を行う. その結果, 次の定理が得られる.

定理 7 ([16], Fig. 6) 入力行列 $LDL^t - \sigma_c I$ に対して浮動小数点演算による dstqds 変換を適用して得られる分解を $L_c D_c L_c^t$ とする. このとき, L, D にそれぞれ微小な相対的摂動を加えて得られる \bar{L}, \bar{D} , および微小な相対的摂動を加えることにより L_c, D_c になる行列 \bar{L}_c, \bar{D}_c が存在し, $\bar{L}\bar{D}\bar{L}^t - \sigma_c I = \bar{L}_c \bar{D}_c \bar{L}_c^t$ が厳密に成り立つ. より詳しくは, $\bar{L}, \bar{D}, L_c, D_c$ はそれぞれ $L, D, \bar{L}_c, \bar{D}_c$ の各要素を 3ulps, 1ulps, 3ulps, 2ulps 変化させることによって得られる.

$$\begin{array}{ccc} LDL^t & \xrightarrow{\text{dstqds in floating point arithmetic}} & L_c D_c L_c^t \\ \downarrow & & \uparrow \\ \bar{L}\bar{D}\bar{L}^t & \xrightarrow{\text{exact dstqds: } \bar{L}\bar{D}\bar{L}^t - \sigma_c I = \bar{L}_c \bar{D}_c \bar{L}_c^t} & \bar{L}_c \bar{D}_c \bar{L}_c^t \end{array}$$

いま, 元の表現 LDL^t から計算した固有ベクトルの 1 つを \hat{z}_i とする. 上の定理を使って \hat{z}_j と \hat{z}_i の直交性を調べるため, \hat{z}_j と \hat{z}_i のなす角を次の 6 つの部分に分けて考える [16].

- Getvec により計算された固有ベクトル \hat{z}_i と LDL^t の厳密な固有ベクトル v_i とのなす角.
- v_i と部分空間 $S_{\Gamma_c}^{LDL^t}$ とのなす角.
- 部分空間 $S_{\Gamma_c}^{LDL^t}$ と部分空間 $S_{\Gamma_c}^{\bar{L}\bar{D}\bar{L}^t}$ とのなす角.
- 部分空間 $S_{\Gamma_c}^{\bar{L}\bar{D}\bar{L}^t}$ と部分空間 $S_{\Gamma_c}^{\bar{L}_c \bar{D}_c \bar{L}_c^t}$ とのなす角.
- 部分空間 $S_{\Gamma_c}^{\bar{L}_c \bar{D}_c \bar{L}_c^t}$ と部分空間 $S_{\Gamma_c}^{L_c D_c L_c^t}$ とのなす角.
- 部分空間 $S_{\Gamma_c}^{L_c D_c L_c^t}$ と Getvec により計算された固有ベクトル \hat{z}_j とのなす角.

⁷ $\hat{\lambda}_j$ が LDL^t の相対誤差の意味で高精度な固有値であっても, $\hat{\lambda}_j - \sigma_c$ が $L_c D_c L_c^t$ の相対誤差の意味で高精度な固有値であるとは限らないので, 再計算あるいは精度向上が必要である.

このうち, (a), (f) は, 定理 6 で述べたように, $O(n\sqrt{n\epsilon})$ 程度 (多くの場合は $O(n\epsilon)$ 程度) とできる. また, (b) については, 対称行列の異なる固有値に属する固有ベクトルが直交することより, \mathbf{v}_i と部分空間 $S_{\Gamma_c}^{LDL^t}$ は直交する. (d) は, $\bar{L}_c \bar{D}_c \bar{L}_c^t$ が $\bar{L} \bar{D} \bar{L}^t$ を正確にシフトした行列になっていることより, 厳密に 0 である. さらに, (c) は $\bar{L} \bar{D} \bar{L}^t$ が LDL^t に高々 3ulps 程度の相対的摂動を加えた行列であることと LDL^t が Γ_c に対して RRR であることから高々 $3K_3 n\epsilon / \text{relgap}(\Gamma_c; LDL^t)$ 程度, (e) も $L_c D_c L_c^t$ が $\bar{L}_c \bar{D}_c \bar{L}_c^t$ に高々 3ulps 程度の相対的摂動を加えた行列であることと $L_c D_c L_c^t$ が Γ_c に対して RRR であることから高々 $3K_3 n\epsilon / \text{relgap}(\Gamma_c; L_c D_c L_c^t)$ 程度となる.

以上をまとめると, 条件 (A), (B), (C) が成り立つならば, 新しい表現から計算したベクトル $\hat{\mathbf{z}}_j$ は元の表現から計算したベクトル $\hat{\mathbf{z}}_i$ と高い精度で直交する (内積が $O(n\epsilon)$ 程度に抑えられる) ことが示せる. なお, $L_c D_c L_c^t$ から計算した固有ベクトルどうしも Getvec の性質より高い直交性を持つから, 両方の表現から計算した固有ベクトル群は高い直交性を持つ.

4.4.4 計算された固有ベクトルの残差

ベクトル $\hat{\mathbf{z}}_j$ は, Getvec の性質より, 行列 $L_c D_c L_c^t$ に対しては小さい残差を持つ. しかし, 元の行列に対する残差 $(LDL^t - \hat{\lambda}_j I) \mathbf{z}_j$ が小さいかどうかは自明ではないため, これを評価する必要がある.

そこで, この残差を次のように分解して評価する.

$$\begin{aligned}
 \|(LDL^t - \hat{\lambda}_j I) \mathbf{z}_j\| &= \|(L_c D_c L_c^t - (\hat{\lambda}_j - \sigma_c) I) \mathbf{z}_j \\
 &\quad + (\bar{L}_c \bar{D}_c \bar{L}_c^t - L_c D_c L_c^t) \mathbf{z}_j \\
 &\quad + (\bar{L} \bar{D} \bar{L}^t - \bar{L}_c \bar{D}_c \bar{L}_c^t - \sigma_c I) \mathbf{z}_j \\
 &\quad + (LDL^t - \bar{L} \bar{D} \bar{L}^t) \mathbf{z}_j\| \\
 &\leq \|(L_c D_c L_c^t - (\hat{\lambda}_j - \sigma_c) I) \mathbf{z}_j\| \\
 &\quad + \|(\bar{L}_c \bar{D}_c \bar{L}_c^t - L_c D_c L_c^t) \mathbf{z}_j\| \\
 &\quad + \|(LDL^t - \bar{L} \bar{D} \bar{L}^t) \mathbf{z}_j\|
 \end{aligned}
 \tag{4.34}$$

ここで, 最後の不等式では, $\bar{L} \bar{D} \bar{L}^t - \sigma_c I = \bar{L}_c \bar{D}_c \bar{L}_c^t$ が厳密に成り立つことを用いた.

いま, $\|\mathbf{z}_j\| = 1$ と規格化して考えると, 式 (4.34) の最右辺の第 1 項は Getvec の性質より $O(n\sqrt{n\epsilon}(\hat{\lambda}_j - \sigma_c))$ 程度 (多くの場合は $O(n\epsilon(\hat{\lambda}_j - \sigma_c))$ 程度) となる. また, 条件 (D) を使うと, 第 2 項, 第 3 項は $O(\text{spdiam}[LDL^t]\epsilon)$ 程度であることが示せる [16].

以上をまとめ, かつ通常は $|\hat{\lambda}_j - \sigma_c| \ll 1$ であることを考慮すると, 新しい表現から計算したベクトルの元の行列に対する残差 $\|(LDL^t - \hat{\lambda}_j I) \mathbf{z}_j\|$ は $O(n\epsilon) + O(\text{spdiam}[LDL^t]\epsilon)$ 程度に抑えられることがわかる.

4.4.5 representation tree

以上では、 Γ_c に属する固有ベクトルのうち、 $L_c D_c L_c^t$ から Getvec により計算可能なものについて、直交性と残差に関する性質を示した。しかし一般に、 Γ_c に属する固有値の中には、新しい表現 $L_c D_c L_c^t$ においても十分大きな相対的ギャップを持たず、Getvec により固有ベクトルを計算できないような固有値が存在する。このような固有値の集合は、 $L_c D_c L_c^t$ の（一般には複数の）クラスターとなる。

これらの固有値に対して固有ベクトルを計算するため、MR³ アルゴリズムでは各クラスターに対して $L_c D_c L_c^t$ の再度の原点シフトを行い、クラスターに属する固有値の少なくとも1個が大きな相対的ギャップを持つような表現 $L_d D_d L_d^t = L_c D_c L_c^t - \sigma_d I$ を求めてから、Getvec を適用する。このように、クラスターの同定と新しい表現の生成という過程が、どの固有値も孤立固有値と見なせるまで、再帰的に続けられる。このとき、ある表現から原点シフトにより生成した表現を元の表現の子供と見なすことにより、表現の集合は最初の表現 LDL^t を根とする木構造をなす。これを representation tree と呼ぶ [16]。

4.4.3 節と 4.4.4 節で扱ったのは親と子の2つの表現しかない場合の解析であるが、同様の解析を representation tree の枝をたどって行うことにより、representation tree を用いて計算した任意の固有ベクトルどうしの直交性、および任意の固有ベクトルの最初の表現に対する残差を評価することができる。representation tree の高さを ndepth とするとき、[16] では計算された固有ベクトルの直交性および残差が次の不等式で抑えられることが示されている。

定理 8 ([16], Theorem 3) MR³ アルゴリズムで計算した第 j 固有値に属する固有ベクトルを \hat{z}_j とするとき、次の式が成り立つ。

$$(4.35) \quad \max_{j \neq k} |\hat{z}_k^t \hat{z}_j| \leq 2[G + (\text{ndepth} - 1)R]n\epsilon.$$

ここで、 G , R はある定数である。

定理 9 ([16], Theorem 4) \hat{z}_j の残差については次の式が成り立つ。

$$(4.36) \quad \|LDL^t \hat{z}_j - \hat{z}_j\| \leq Gn\epsilon + 9(\text{ndepth} - 1) \left(2C + \frac{1}{2}\right) \epsilon \text{spdiam}[LDL^t] + O(n\epsilon^2).$$

ここで、 G は定理 8 と同じ定数である。また、 C は式 (4.32), (4.33) に出てくる定数である。

4.5 まとめ

以上で、MR³ アルゴリズムの概要を説明した。最後に、3章で述べたアルゴリズムの目標と照らし合わせ、どこまで達成できているかを確認したい。

計算量 まず、求めたい固有値・固有ベクトルの組すべてについて相対的ギャップが大きい場合は、最初に作った正定値の LDL^t 分解を用いてアルゴリズム Getvec により各固有ベクトルを求めればよい。各固有ベクトルの計算量は $O(n)$ だから、全固有ベクトルの計算量は $O(n^2)$

となり、目標が達成できている。なお、従来の逆反復法でもすべての固有値が $10^{-3} \|T\|$ 以上離れている場合は同じオーダーの計算量が達成できるが、MR³ アルゴリズムでは、これを相対的ギャップが大きい場合へと緩和したことが特徴である。一方、相対的ギャップが小さい固有値がある場合は、表現の変換 $LDL^t \rightarrow L_c D_c L_c^t$ が必要となる。これによる演算量の増加は、条件 (A) ~ (D) を満たすシフトが何回の試行錯誤で見つかるか、また、条件を満たしているかどうかの判定にどの程度の演算量が必要かにより異なるが、それぞれ定数回、 $O(n)$ の演算量 [16] と仮定すると、オーダーとしては固有ベクトル 1 本あたり $O(n)$ で、相対的ギャップが大きい場合と変わらない。したがって、目標は達成できていると言える。

一部の固有ベクトルのみの計算 上で述べた通り、固有ベクトル 1 本あたりの計算量は相対的ギャップが大きい場合でも小さい場合でも $O(n)$ だから、 k 本を求めるための演算量は $O(kn)$ となる。

並列性 ある表現が与えられたとき、その下で Getvec を用いて各固有ベクトルを計算できるかどうかの判定、および Getvec による実際の固有ベクトル計算は、固有ベクトルごとに独立にできる。したがって、並列粒度は非常に大きい。

精度 求めたい固有値・固有ベクトルの組すべてについて相対的ギャップが大きい場合は、RRR であることが保証されている初期の正定値の LDL^t 分解によりすべての固有ベクトルが計算できる。したがって、近似固有値さえ高い相対精度で求めておけば、4.3.2 項の最後で述べたように、最初の目標 (d') が (正確には n を $n\sqrt{n}$ に変えた形で) 達成できる。一方、相対的ギャップが小さい固有値がある場合には、最初の表現 LDL^t に対してシフトを行って得られる新しい表現 $L_c D_c L_c^t$ を用いる必要がある。しかし、この場合、シフト σ_c を決めるために満たすべき条件は 4.4.2 項の (A), (B), (C), (D) と多く、かつ、条件を満たすような σ_c を必ず与える手続きも存在しない。したがって、この場合については目標 (d') が十分達成できたとはまだ言えず、今後さらに研究が必要であると思われる。

5 今後の発展方向

5.1 理論の精密化と新しいアルゴリズムの開発

前章の説明で明らかなように、MR³ アルゴリズムは非常に優れた特徴を持つ新しいアルゴリズムであるが、現在までに構築されている理論では、十分にその性質が解明されていない部分がある。たとえば、正定値でない LDL^t に対する RRR の存在の問題や、式 (4.27) における右辺第 2 項のより良い評価の問題などである。また、固有値を求める部分の dqds 法でも、シフトがある場合の収束証明は与えられていない。

この方向への研究として、中村・岩崎らによる mdLVs (modified discrete Lotka-Volterra with shit) 法の研究 [28][34][45] がある。mdLVs 法は、可積分系である離散 Lotka-Volterra 方程式を用いた新しい特異値計算アルゴリズムであり、dqds 法と同等以上の精度で特異値を計

算できるという特徴を持つ。さらに、シフト付きのアルゴリズムについても収束証明を持つという、dqds法にはない特長がある。最近では、特異ベクトルの計算もdqds変換に代えてLotka-Volterra系を用いて行おうという研究が進められており、今後が期待される。

5.2 密行列固有値解法への組み込み

一方、MR³アルゴリズムにより三重対角行列の固有値・固有ベクトルが高速に求められるとなると、密行列から三重対角行列への変換・逆変換の部分の実行時間が問題となってくる。実際、Bientinesiらの論文[4]では、64,000元の密行列固有値問題を64ノードのPentium 4 (2.4GHz) クラスタで解く場合、MR³アルゴリズムによる三重対角行列の固有値・固有ベクトル計算には406秒しか要しないのに対し、三重対角行列への変換には9,638秒、逆変換には2,846秒の時間を要することが報告されている。これは、三重対角化で使用されているDongarraのアルゴリズム[18][17]において、キャッシュ利用効率が低く、かつ細かな通信が多発することによって考えられる[49]。

これに対し、帯行列を経由して三重対角化を行うことにより、三重対角化において $O(n^3)$ の演算量を持つ部分をすべてlevel-3 BLAS (行列積) の形で行うことができるBischofのアルゴリズム[6][7][5]、およびそれを改良したWuのアルゴリズム[48]が提案されており、最近のマイクロプロセッサ上でDongarraのアルゴリズムの2倍以上の性能を達成できることが示されている[49]。さらに、並列の場合には、この差が5倍近くにまでに開く[50]。そこで、BischofまたはWuのアルゴリズムとMR³アルゴリズムとを組み合わせることが、効率の良い密行列固有値ソルバの開発のために有望である。BischofあるいはWuのアルゴリズムの効率的な実装では、固有ベクトルを帯行列から求めるため[49]、MR³アルゴリズムとの組み合わせ方は自明ではないが、検討の価値があると考えられる。

5.3 応用

MR³アルゴリズムの固有値計算以外の分野への応用として、特異値分解への応用が研究されている[25][15]。上二重対角行列 L^t の特異値分解 $L^t = U\Sigma V^t$ を計算する場合、 U の列ベクトルの直交性、 V の列ベクトルの直交性の両方を満たすことが必要であり、かつ満たすべき残差の式も $L^t v - u\sigma = 0$, $Lu - v\sigma = 0$ (ただし σ は特異値、 u , v は対応する特異ベクトル) の2つがあるため、固有値問題に比べてより複雑な問題となる。しかしこの場合でも、MR³アルゴリズムを利用してこれらすべての式を精度良く満たす解を求めるアルゴリズムが提案されている[25]。

6 おわりに

本稿では、対称行列の新しい固有値・固有ベクトル計算法として注目されているMR³アルゴリズムについて、その骨格と収束性や精度の保証に用いられる定理とを簡潔な形で紹介す

ることを目的として、サーベイを行った。また、アルゴリズムにおいて現在までの理論でカバーできていない点を明らかにするとともに、今後発展が期待される方向についても簡単に述べた。密行列向け固有値計算の分野では、この他にも、マルチシフト QR 法や超高精度の固有値計算法など、最近 10 年程度の間大きな発展があった。機会があれば、これらについても別のサーベイ論文で報告したい。

謝辞 本論文の執筆を勧めて下さった東京大学大学院情報理工学研究科の杉原正顯教授と筑波大学大学院システム情報工学研究科の櫻井鉄也助教授に感謝いたします。また、本論文の内容を応用数学会研究部会「行列・固有値問題の解法とその応用」で発表した際にご討論下さった皆様と、本論文に対して有益なご指摘を下さった査読者の皆様に感謝いたします。なお、本研究は名古屋大学 21 世紀 COE プログラム「計算科学フロンティア」および科学研究費補助金若手研究 (B) (課題番号 16760053) の補助を受けている。

参考文献

- [1] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov and D. Sorensen, *LAPACK Users' Guide*, 2nd ed., SIAM, Philadelphia, 1995.
- [2] I. Bar-On and B. Codenotti, A fast and stable parallel QR algorithm for symmetric tridiagonal matrices, *Linear Algebra and its Applications*, Vol. 220 (1995), 63–95.
- [3] M. W. Berry and M. Browne, *Understanding Search Engines*, SIAM, Philadelphia, 1999.
- [4] P. Bientinesi, I. S. Dhillon and R. A. van de Geijn, A parallel eigensolver for dense symmetric matrices based on multiple relatively robust representations, to appear in *SIAM Journal on Scientific Computing*.
- [5] C. Bischof, B. Lang and X. Sun, Parallel tridiagonalization through two-step band reduction, Technical Report 17, *PRISM Working Note*, 1994. <http://www-unix.mcs.anl.gov/prism/lib/tech.html>.
- [6] C. Bischof and C. F. van Loan, The WY representation for products of Householder matrices, *SIAM Journal on Scientific and Statistical Computing*, Vol. 8, No. 1 (1987), s2–s13.
- [7] C. Bischof, M. Marques and X. Sun, Parallel bandreduction and tridiagonalization, Technical Report 8, *PRISM Working Note*, 1993. <http://www-unix.mcs.anl.gov/prism/lib/tech.html>

- [8] L. S. Blackford, J. Choi, A. Cleary, E. D’Azevedo, J. Demmel, I. Dhillon, J. Dongarra, S. Hammarling, G. Henry, A. Petitet, K. Stanley, D. Walker and R. C. Whaley, *ScaLAPACK Users’ Guide*, SIAM, Philadelphia, 1997.
- [9] J. J. M. Cuppen, A divide and conquer method for the symmetric tridiagonal eigenproblem, *Numerische Mathematik*, Vol. 36 (1981), 177–195.
- [10] J. Demmel, *Applied Numerical Linear Algebra*, SIAM, Philadelphia, PA, 1997.
- [11] J. Demmel and W. Kahan, Accurate singular values of bidiagonal matrices, *SIAM Journal on Scientific and Statistical Computing*, Vol. 11, No. 5 (1990), 873–912.
- [12] J. Demmel and K. Veselic, Jacobi’s method is more accurate than QR, *SIAM journal on Matrix Analysis and Applications*, Vol. 13 (1992), pp. 1204–1246.
- [13] I. S. Dhillon, *A New $O(n^2)$ Algorithm for the Symmetric Tridiagonal Eigenvalue/Eigenvector Problem*, Ph. D. Thesis, Computer Science Division, University of California, Berkeley, California, May 1997.
- [14] I. S. Dhillon, Current inverse iteration can fail, *BIT*, Vol. 38, No. 4 (1998), 685–704.
- [15] I. S. Dhillon and B. N. Parlett, Orthogonal eigenvectors and relative gaps, *SIAM Journal on Matrix Analysis and Applications*, Vol. 25, No. 3 (2004), 858–899.
- [16] I. S. Dhillon and B. N. Parlett, Multiple representations to compute orthogonal eigenvectors of symmetric tridiagonal matrices, *Linear Algebra and its Applications*, Vol. 387, No. 1 (2004), 1–28.
- [17] J. J. Dongarra and R. A. van de Geijn, Reduction to condensed form for the eigenvalue problem on distributed architectures, *Parallel Computing*, Vol. 18, No. 9 (1992), 973–982.
- [18] J. J. Dongarra, S. J. Hammarling and D. C. Sorensen, Block reduction of matrices to condensed forms for eigenvalue computations, *Journal of Computational and Applied Mathematics*, Vol. 27 (1989), 215–217.
- [19] J. J. Dongarra and D. Sorensen, A fully parallel algorithm for the symmetric eigenproblem, *SIAM Journal on Scientific and Statistical Computing*, Vol. 8, No. 2 (1987), 139–154.
- [20] Z. Drmac and K. Veselic, On the singular value decomposition of matrices generated by finite elements, preprint.
- [21] S. Eisenstat and I. Ipsen, Relative perturbation techniques for singular value problems, *SIAM Journal on Numerical Analysis*, Vol. 32, No. 6 (1995), 1972–1988.

- [22] K. Fernando and B. N. Parlett, Accurate singular values and differential qd algorithms, *Numerische Mathematik*, Vol. 67, No. 2 (1994), 191–229.
- [23] G. J. F. Francis, The QR transformation, Parts I and II, *Computer Journal*, Vol. 4 (1961-62), 265–271, 332–345.
- [24] G. H. Golub and C. F. van Loan, *Matrix Computations*, 3rd ed., Johns Hopkins University Press, 1996.
- [25] B. Grosser and B. Lang, An $O(n^2)$ algorithm for the bidiagonal SVD, *Linear Algebra and its Applications*, Vol. 358 (2002), 45–70.
- [26] M. Gu and S. C. Eisenstat, A stable and efficient algorithm for the rank-1 modification of the symmetric eigenproblem, *SIAM Journal on Matrix Analysis and Applications*, Vol. 15, No. 4 (1994), 1266–1276.
- [27] M. Gu and S. C. Eisenstat, A divide-and-conquer algorithm for the symmetric tridiagonal eigenproblem, *SIAM Journal on Matrix Analysis and Applications*, Vol. 16, No. 1 (1995), 172–191.
- [28] M. Iwasaki and Y. Nakamura, On a convergence of solution of the discrete Lotka-Volterra system, *Inverse Problems*, Vol. 18 (2002), 1569–1578.
- [29] 片桐孝洋, 金田康正, 並列固有値ソルバーの実現とその並列性の改良, 並列処理シンポジウム JSPP98 論文集, 223–230, 1998.
- [30] L. Kaufman, A parallel QR algorithm for the symmetric tridiagonal eigenvalue problem, *Journal of Parallel and Distributed Computing*, Vol. 23 (1994), 429–434.
- [31] V. N. Kublanovskaya, On some algorithms for the solution of the complete eigenvalue problem, *Zh. Vychisl. Mat.*, Vol. 1 (1961), 555–570.
- [32] R. -C. Li, Relative perturbation theory: I. Eigenvalue and singular value variations, *SIAM Journal on Matrix Analysis and Applications*, Vol. 19 (1998), 956–982.
- [33] R. -C. Li, Relative perturbation theory: II. Eigenspace and singular subspace variations, *SIAM Journal on Matrix Analysis and Applications*, Vol. 20 (1998), 471–492.
- [34] 中村佳正, 岩崎雅史, シフトつき可積分特異値分解アルゴリズムについて, 日本応用数学会 2004 年度年会講演予稿集, 400-401, 2004.
- [35] K. Naono, Y. Yamamoto, M. Igai, H. Hirayama and N. Ioki, A multi-color inverse iteration for a high performance real symmetric eigensolver, in B. Ludwig and K. Wismuller (eds.), *Proc. Of Euro-Par 2000*, Lecture Notes in Computer Science 1900, Springer-Verlag, 527-531, 2000.

- [36] B. N. Parlett, *The Symmetric Eigenvalue Problem*, SIAM, Philadelphia, PA, 1997.
- [37] B. N. Parlett, For tridiagonals T replace T with LDL^t , *Journal of Computational and Applied Mathematics*, Vol. 123 (2000), 117–130.
- [38] B. N. Parlett, A bidiagonal matrix determines its hyperbolic SVD to (varied) relative accuracy, *SIAM Journal on Matrix Analysis and Applications*, submitted.
- [39] B. N. Parlett and I. S. Dhillon, Fernando's solution to Wilkinson's problem; an application of double factorization, *Linear Algebra and its Applications*, Vol. 267 (1997), 247–279.
- [40] B. N. Parlett and I. S. Dhillon, Relatively robust representations of symmetric tridiagonals, *Linear Algebra and its Applications*, Vol. 309 (2000), 121–151.
- [41] B. N. Parlett and O. A. Marques, An implementation of the dqds algorithm (positive case), *Linear Algebra and its Applications*, Vol. 309 (2000), 217–259.
- [42] G. Peters and J. H. Wilkinson, The calculation of specified eigenvectors by inverse iteration, contribution II/18, Vol. II of *Handbook of Automatic Computation*, Springer-Verlag, 418–439, 1971.
- [43] H. Rutishauser, *Lectures on Numerical Mathematics*, Birkhauser, Boston, 1990.
- [44] A. H. Sameh and D. J. Kuck, A parallel QR algorithm for symmetric tridiagonal matrices, *IEEE Transactions on Computing*, Vol. C-26 (1977), 147–153.
- [45] 高田雅美, 岩崎雅史, 木村欣司, 中村佳正, ロトカ・ボルテラ系による特異値分解アルゴリズムの実装と性能評価, HPCS2005 論文集, 2005.
- [46] F. Tisseur and J. Dongarra, A parallel divide and conquer algorithm for the symmetric eigenvalue problem on distributed memory architectures, *SIAM Journal on Scientific Computing*, Vol. 20, No. 6 (1999), 2223–2226.
- [47] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*, Oxford University Press, Oxford, 1965.
- [48] Y. -J. J. Wu, P. A. Alpatov, C. H. Bischof and R. A. van de Geijn, A parallel implementation of symmetric band reduction using PLAPACK, *Proc. Scalable Parallel Libraries Conference*, 1996.
- [49] 山本有作, キャッシュマシン向け対称密行列固有値解法の性能・精度評価, 情報処理学会論文誌: コンピューティングシステム, Vol. 46, No. SIG 3 (ACS8) (2005), 81–91.
- [50] Y. Yamamoto: Performance of fully BLAS-3 based tridiagonalization algorithms on modern SMP machines, in preparation.

- [51] 山本有作, 猪貝光祥, 直野健, 共有メモリ型並列計算機向けの高並列固有ベクトル解法とSR8000での評価, 情報処理学会論文誌, Vol. 42, No. 4 (2001), 771-778.
- [52] Y. Yamamoto, M. Igai and K. Naono, A new BLAS-3 based parallel algorithm for computing the eigenvectors of real symmetric matrices, in L. T. Yang and Y. Pan (eds.), *High Performance Scientific and Engineering Computing - Hardware/Software Support*, Kluwer Academic Publishers, 2004.

山本有作 (正会員) 〒464-8603 愛知県名古屋市千種区不老町

1992年 東京大学大学院工学系研究科 物理工学専攻 修士課程終了。同年 (株) 日立製作所入社。現在, 名古屋大学大学院工学研究科 計算理工学専攻 講師。並列数値計算アルゴリズムとその応用の研究に従事。博士 (工学)。

(2005年3月26日受付)

(2005年5月16日最終稿受付)