数理解析研究所講究録
1198巻 2001年 28-32

固有値問題の数値解の高速精度保証

大石進一 (Shin'ichi Oishi)
早稲田大学理工学部情報学科
oishi@oishi.info.waseda.ac.jp

1 はじめに

連立一次方程式，固有値問題，特異値問題など数値線形代数の問題を数値計算で解いたとき，得られた数値解の近くに真の解が存在するかや存在するとしたら，真の解と数値解の差はどれ位あるかを，数値計算で厳密に評価すること (これを精度保証という) が，多くの問題について数値解を求める手間と同程度で実行できることを著者は最近示してきた。本報告ではこのような手法に基づく固有値問題の高速精度保証法について議論する。研究会当日は文献 [1] に従って，固有値問題の数値解の精度保証が，数値解を計算する時間と同程度かそれより短い時間で計算できることを述べた。研究会当日の報告の内容は，この論文に詳細に記述されているので，ここでは，その補足になるような事柄についてまとめておきたい。

2 IEEE754

高速な数値計算というときには現行では倍精度浮動小数点数を使った計算を考えなければいけない。本節はその標準について記述する。本報告全体の準備である。

現在の浮動小数点演算はIEEE754規格に基ずくものが標準的である。これは優れた数値計算学者であるKahanらの努力により1985年に制定された。その内容については、当事者であるW. Kahanの解説 (http://www.cs.berkeley.edu/~wkahan/ ieee754status/) を読むのがよいであろう。制定の歴史を語った文書もある (An Interview with the Old Man of Floating-Point (http://http.cs.berkeley.edu/~wkahan/ ieee754status/754story.html))。現在ではIntelのPentium CPUを含むほとんどのマシンがIEEE754に従っているが、どのようなCPUが例外かなどはInterval FAQの記録 What machines support IEEE754?(http://studsys.mscs.mu.edu/~georgec/ IFAQ/casares1.html) が参考になる。また、IEEE754にしたがってどのように浮動小数点演算が実装されているかの例は、例えば、IntelのPentiumIIIのマニュアル

(http://developer.intel.com/design/intarch/pentiumiii/Manual.htm)

などを見るとよい。

IEEE754によって標準化されているのは2進の倍精度浮動小数点数である。浮動小数点数を利用した精度保証付き計算にはIEEE754の丸め (以下に示すものの中で上と下への丸め) を利用する。$\mathbb{R}$ を実数の集合，$\mathbb{F}$ を倍精度浮動小数点数の集合，c を実数 ($c \in \mathbb{R}$) と

する。

(1) 上向きの丸め (round upward) c 以上の浮動小数点数の中で最も小さい数に丸める。これを $\triangle : \mathbb{R} \to \mathbb{F}$ と表す。プログラム中では **up** と表すことにする。

(2) 下向きの丸め (round downward) c 以下の浮動小数点数の中で最も大きい数に丸める。これを $\triangledown : \mathbb{R} \to \mathbb{F}$ と表す。プログラム中では **down** と表すことにする。

これ以外にも近似計算のデフォールトの丸めである最近点への丸め (round to nearest) とチョッピング (切り捨て)(round toward 0) の丸めがある。IEEE754 では浮動小数点数演算 ($\mathbb{F}$ 上での四則演算) は丸めとの関係によりつぎのように定義されている。$\cdot \in \{+, -, \times, /\}$, $\bigcirc \in \{\triangle, \triangledown\}$ のとき

$$x \bigodot y = \bigcirc(x \cdot y) \text{ (任意の } x, y \in \mathbb{F} \text{ について)} \tag{1}$$

この式は，左辺の浮動小数点数の四則演算の結果 $x \bigodot y$ は，右辺の数学的に正しい (実数としての) 四則演算の結果 $x \cdot y$ を指定された丸めを行って得られた数 $\bigcirc(x \cdot y)$ に一致するように計算する規格を表している。また，平方根も

$$(\sqrt{x})_{fp} = \bigcirc(\sqrt{x}) \text{ (任意の } x \in \mathbb{F} \text{ について)} \tag{2}$$

と，浮動小数点数演算によって計算された平方根 $(\sqrt{x})_{fp}$ は，正確な実数演算で計算された平方根 $\sqrt{x}$ を指定された丸めの方向へ丸めた数となる規格である。注意すべきことは，指数関数や三角関数などはこのような規格をみたすようには規定されていないことである。この事に関しては初等関数などの実装の研究者 J.-M. Muller の The Table Maker's Dilemma:our search for worst cases (http://www.ens-lyon.fr/~jmmuller/Intro-to-TMD.htm) が参考になる。

基礎は内積計算である。n 次元ベクトル $u = (u_1, u_2, \cdots, u_n)^t$ と $v = (v_1, v_2, \cdots, v_n)^t$ を考える。ただし、u_i, v_i は $i = 1, 2, \cdots, n$ に対して倍精度浮動小数点数であるとする。このとき、u と v の内積 z を計算することを考えよう。よく考えると

down;
$zd = u' * v$;
up;
$zu = u' * v$;

とすると $z \in [z_d, z_u]$ となることがわかる。ただし、以上の計算は Matlab 的に計算したものとする。Matlab ではベクトル u, v に対して $u' * v$ はベクトルの内積として演算子多重定義されており、内部では LAPACK(場合により LINPACK) のベクトルの内積のライブラリ関数が呼び出されて計算される。もちろん、最高に高速化を望む場合は、C や FORTLAN でプログラムを書いて LAPACK と速い BLAS の組み合わせで、LAPACK の内積などの計算関数を呼び出して計算する。丸めの制御計算方式とは、従来の高速パッケージ関数の関数を呼び出して精度保証計算を進める方式で、従来の高速パッケージでは使用されていなかった丸めの制御命令が、関数呼び出しの前と後に位置し、パーケージの呼び出し (ひ

いては高速化) の妨げにならないような計算方式のことを指すことにした。以上の内積の精度保証計算が、通常の近似内積計算の 2 倍の計算速度で実行できることはプログラムを見てすぐにわかる。内積計算が高速に精度保証できると、いわゆる数値線形代数のアルゴリズムは高速に精度保証することができる。

3 固有値の精度保証

ここでは，次の固有値問題を考える。

$$Bx = \lambda x \tag{3}$$

ただし，B は $n \times n$ 複素行列で，λ は固有値，x は n 次元ベクトルである。本節では Bauer-Fike の摂動定理により，計算した固有値の精度保証をすることを考える。

3.1 Bauer-Fike の定理の応用

行列 B に対して，固有値と固有ベクトルを数値計算したとする。計算したすべての固有値を対角に配した $n \times n$ 対角行列 D とし，対応する固有ベクトルを列ベクトルとして並べた $n \times n$ 行列を P とする。このとき，数値計算誤差がなく，真の固有値，真の固有ベクトルが計算できたならば

$$BP = PD \tag{4}$$

が成立するが，実際には式 (4) は近似的にしか成立しない。ここで，P は正則と仮定し，

$$A = PDP^{-1} \tag{5}$$

と定義する。次の Bauer-Fike の定理が成立する。

Theorem 1 (Bauer-Fike) $D = diag(\lambda_1, \lambda_2, \cdots, \lambda_n)$ を $n \times n$ 対角行列，P を $n \times n$ 正則行列とする。$A = PDP^{-1}$ とする。このとき，$n \times n$ 行列 B の固有値を λ とすると，

$$\min_{1 \leq k \leq n} |\lambda - \lambda_k| \leqq \|P\| \, \|P^{-1}\| \, \|B - A\| \tag{6}$$

が成立する。ただし，$\|\cdot\|$ は $\|\cdot\|_p$, $(p = 1, 2, \cdots, \infty)$ のいずれかとする。

この定理から，$\epsilon = \|P\| \, \|P^{-1}\| \, \|B - A\|$ とするとき

$$\Lambda = \bigcup_{1 \leq k \leq n} U_k, \; U_k = \{z \in \mathbb{C} | \; |z - \lambda_k| \leq \epsilon\} \tag{7}$$

に B のすべての固有値が含まれることがわかる。

ここで，$t \in [0, 1]$ に対して

$$B_t = (1 - t)A + tB \tag{8}$$

と定義する。このとき，B_t の固有値を λ_t とすると Bauer-Fike の定理から

$$\min_{1 \leq k \leq n} |\lambda_t - \lambda_k| \leqq t\|P\| \, \|P^{-1}\| \, \|B - A\| \tag{9}$$

が成立する。したがって，λ_t も Λ の中に含まれる。また，$t=0$ のときを考えると，$B_0=A$ より，U_k は $\lambda_{0_k}=\lambda_k$ を含んでいる。ここでもし，$0\leq m<n$ に対して $\bigcup_{1\leq i\leq m}U_{k_i}$ が他の残りの U_k と交わらなければ，B_t の固有値が t の連続関数になることから $\bigcup_{1\leq i\leq m}U_{k_i}$ の中に m 個の B の固有値が存在することがわかる。

以下，$\epsilon=\|P\|\,\|P^{-1}\|\,\|B-A\|$ を精度保証付きで計算する方法を示そう。以下，簡単のため，最大値ノルムを考える。まず，P は与えられているので，$\|P\|_\infty$ は精度保証付きで計算することができる。L を P^{-1} を数値計算で求めた結果の近似行列であるとする。$\|I-LP\|_\infty<1$ であれば，P が正則であることがわかる。また，

$$\|P^{-1}\|_\infty\leq\frac{\|L\|_\infty}{1-\|I-LP\|_\infty} \tag{10}$$

が成立するので，これにより，$\|P^{-1}\|_\infty$ の上限を精度保証つきで計算することができる。一方，

$$\begin{aligned}\|A-B\|_\infty &= \|PDP^{-1}-B\|_\infty \\ &\leq \|PD(P^{-1}-L)+PDL-B\|_\infty \\ &\leq \|PD\|_\infty\,\|P^{-1}-L\|_\infty+\|PDL-B\|_\infty\end{aligned} \tag{11}$$

ここで，

$$\|P^{-1}-L\|_\infty\leq\|P^{-1}\|_\infty\|I-LP\|_\infty \tag{12}$$

以上をまとめて，

$$\begin{aligned}\min_{1\leq k\leq n}|\lambda-\lambda_k|\leqq\epsilon &\leqq \|P\|_\infty\|P^{-1}\|_\infty\left\{\|PD\|_\infty\|P^{-1}-L\|_\infty+\|PDL-B\|_\infty\right\} \\ \|P^{-1}\|_\infty &\leq \frac{\|L\|_\infty}{1-\|I-LP\|_\infty} \\ \|P^{-1}-L\|_\infty &\leq \|P^{-1}\|_\infty\|I-LP\|_\infty\end{aligned} \tag{13}$$

を得る。この式に現れる $\|P\|_\infty$, $\|L\|_\infty$, $\|PD\|_\infty$, $\|I-LP\|_\infty$, $\|PDL-B\|_\infty$ は文献 [1] に示された方法で，高速に精度保証付きで上限を求めることができるので，ϵ の上限を精度保証付きで高速に求めることができる。

この方法は理論的にはすっきりしている。また，200×200 のランダム行列のすべての固有値と固有ベクトルの近似を Pentium 750M のマシンで Matlab6 を用いて求めると約 5 秒で，精度保証がこの方法だと 2 秒で終わる。しかし，随所に過大評価しているので，考えた例題の場合，誤差の上限が 10^{-3} 程度とあまりシャープには得られない。同じ行列に対して，文献 [1] の方法は 10^{-9} 程度の精度があると出力するので，文献 [1] の与える結果の方がずっとシャープである。

3.2 シャープな評価

そこで，過大評価を減少させることを考える。記号は前小節を継承しているものとする。Bauer-Fike の定理は実は次の形で成立することが知られている。

Theorem 2 (Bauer-Fike) $D = diag(\lambda_1, \lambda_2, \cdots, \lambda_n)$ を $n \times n$ 対角行列，P を $n \times n$ 正則行列とする。$A = PDP^{-1}$ とする。このとき，$n \times n$ 行列 B の固有値を λ とすると，

$$\min_{1 \leq k \leq n} |\lambda - \lambda_k| \leqq \|P^{-1}(B - A)P\| \tag{14}$$

が成立する。ただし，$\|\cdot\|$ は $\|\cdot\|_p$, $(p = 1, 2, \cdots, \infty)$ のいずれかとする。

ここで，

$$\begin{aligned} \|P^{-1}(B - A)P\| &\leqq \|P^{-1}BP - D\| \\ &\leqq \|LBP - D\| + \|(L - P^{-1})BP\| \\ &= \|LBP - D\| + \|(I - (LP)^{-1})LBP\| \\ &\leqq \|LBP - D\| + \|(LP)^{-1}\| \, \|I - LP\| \, \|LBP\| \end{aligned} \tag{15}$$

に注意すれば，Theorem 2 の条件下で

$$\min_{1 \leq k \leq n} |\lambda - \lambda_k| \leqq \|LBP - D\| + \|(LP)^{-1}\| \, \|I - LP\| \, \|LBP\| \tag{16}$$

が成立することがわかる。D が B の近似固有値からなる対角行列，P が対応する近似固有ベクトル (列ベクトル) を並べた $n \times n$ 行列，L を P の近似逆行列とする。これらの近似の精度が十分よい場合，$\|LBP - D\|$ はほぼゼロ行列，LP はほぼ単位行列であるから，$\|(LP)^{-1}\|$ は 1 程度の量，$\|I - LP\|$ はほぼゼロ行列，$\|LBP\|$ は $\|D\|$ 程度の量であるから，式 (16) はシャープな評価を与える。実際，前小節の 200×200 のランダム複素行列で実験したところ，精度が 10^{-9} 程度で，文献 [1] の与える精度とほとんど同じ精度が得られた。ちなみに，実行時間は前節の環境下で 200×200 の行列のすべての固有値と固有ベクトルの近似を求めると約 5 秒で，精度保証が約 4 秒で終わる。

$$U_k = \{z \in \mathbb{C} | \ |z - \lambda_k| \leq \epsilon\}, \ \epsilon = \|LBP - D\| + \|(LP)^{-1}\| \, \|I - LP\| \, \|LBP\| \tag{17}$$

としたとき，もし，$0 \leq m < n$ に対して $\bigcup_{1 \leq i \leq m} U_{k_i}$ が他の残りの U_k と交わらなければ $\bigcup_{1 \leq i \leq m} U_{k_i}$ の中に m 個の B の固有値が存在することが保証される。このように，理論的にもすっきりし，精度も文献 [1] と同程度に出るようになった。別の言葉でいえば，前小節の結果では P の条件数 $\|P\| \, \|P^{-1}\|$ が最終結果 (6) に出てしまっていた。これは，スケーリングによって形式的にいくらでも悪条件にできることが知られているなど，望ましいことではなかった。これに対して，式 (16) では P の条件数が表に出ないようになったことなどにより，シャープな結果が得られる様になったといえる。

参考文献

[1] Shin'ichi OISHI:"Fast Enclosure of Matrix Eigenvalues and Singular Values Via Rounding Mode Controlled Computation", Linear Algebra and its Applications, **324** (2001) pp.133-146