

適応的ピボット選択付き二重対角化を用いた行列の特異値分解

細田 陽介[†] 鉦田 雅之^{††} 長谷川 武光[†]

通常、行列の特異値分解は、分解すべき行列を二重対角行列に変換し、次に二重対角行列の特異値分解を行うことにより求められる。このうち、二重対角化は行列のランクを考慮していないため無駄な計算をともなう。本論文において我々は、適応的ピボット選択付き二重対角化法を提案する。本方法は必要なときのみピボット選択による行置換を行う。これにより、ランク落ちした行列に対しては高速に、フルランクに近い行列に対しては従来の方法と同程度の速度で、行列の二重対角化および特異値分解を行うことができる。

An Algorithm with Adaptive Pivoting Bidiagonalization for Singular Value Decomposition

YOSUKE HOSODA,[†] MASAYUKI HOKODA^{††}
and TAKEMITSU HASEGAWA[†]

We propose a new method for bidiagonalizing a matrix in a singular value decomposition (SVD) algorithm. Our method incorporates a strategy of adaptive row pivoting into the Householder transformation. Numerical experiments show that for rank-deficient matrices the present method obtains the SVD faster than the conventional method, while for matrices of nearly full rank our method and the conventional method compute equally the SVD.

1. はじめに

我々の目的は、行列 A の特異値分解

$$A = U\Sigma V^T \quad (1)$$

を求めることにある。ここで、分解すべき行列 A は実 $m \times n$ 行列であり、 $m \geq n$ とする。また、 U, V は列正規直交行列、 Σ は対角行列であり、 Σ の対角成分を特異値、 U, V の列ベクトルをそれぞれ左、右特異ベクトルという。

特異値分解は適用範囲の広い有用な数値計算法である。線形方程式の最小二乗問題をはじめ、悪条件方程式、信号処理、画像処理、天文学など、さまざまな分野で用いられている^{1),3),5)}。しかし、特異値分解は、行列の固有値問題の解法を用いて求められるため、計算量が多いという欠点を持つ。

特異値分解は通常

- 直交変換による行列の二重対角化
- 二重対角行列の特異値分解

の順序で計算される。行列の二重対角化は、左右から

の Householder 変換、もしくは、 $m \geq 5/3n$ の場合は Householder 変換による QR 分解と高速 Givens 変換により求められる^{1),2)}。いずれの方法も直接法であり、計算量は分解すべき行列のサイズに依存する。二重対角行列の特異値分解は一般的に陰のシフト QR アルゴリズムが用いられることが多い^{1),2),5)}。ここで我々は、行列の二重対角化、特にランク落ちした行列に対する二重対角化の高速化を考察する。なお、本論文では左右からの Householder 変換による二重対角化法を比較対象とする。

従来の二重対角化法は行列のランク落ちを考慮していないため、ランク落ちした行列に対しては無駄な計算をともなう。これに対して我々はピボット選択付き Householder 変換による二重対角化法を提案した⁴⁾。この方法はランク落ちした行列に対しては高速ではあるが、常時ピボット選択による行置換を行うため、フルランクに近い行列には従来の方法より二重対角化に時間がかかる。一般に行列のランクは分解を行うまでは不明であることが多いため、この方法の常用には問題が残る。

これに対して我々は、適応的ピボット選択付き二重対角化法を提案する。本方法は必要なときのみ、すなわち二重対角行列の対角成分がゼロとなるときだけ、

[†] 福井大学工学部

Faculty of Engineering, Fukui University

^{††} 福井大学大学院工学研究科

Graduate Course in Engineering, Fukui University

残りの要素のゼロ判定およびピボット選択による行置換を行う．そのため、ランク落ちした行列に対しては、ランク落ちを検知して無駄な計算を省き、フルランク行列に対しては、行置換は行われず、従来の二重対角化法と同じ結果が得られる．本方法は、ランク落ちした行列には従来の方法および文献 4) の方法よりも高速で、フルランクに近い行列には従来の方法と同程度の時間で二重対角化が行えることが数値実験により検証された．

本論文の構成は以下のとおりである．次章で我々の提案する適応的ピボット選択付き二重対角化法について述べる．3 章で数値実験およびその考察を行い、4 章でまとめる．

2. 適応的ピボット 選択付き二重対角化法

本章では、我々の提案する適応的ピボット選択付き二重対角化法について述べる．

いま、分解すべき行列 A およびゼロ判定を行うための定数 $\varepsilon_a > 0$ が与えられ、 $k - 1$ 段階までの二重対角化

$$A_{k-1} := \begin{bmatrix} d_1 & f_1 & & & & \\ & \ddots & \ddots & & & \\ & & \ddots & \ddots & & \\ & & & d_{k-1} & f_{k-1} & \\ & & & & a_{kk}^{(k-1)} & \cdots & a_{kn}^{(k-1)} \\ & & & & \vdots & \ddots & \vdots \\ & & & & a_{mk}^{(k-1)} & \cdots & a_{mn}^{(k-1)} \end{bmatrix}$$

が得られたものとする．ただし、 $A_0 := A$ である．このとき、上式右辺の右下 $(m - k + 1) \times (n - k + 1)$ 小行列を $A^{(k-1)}$ とおき、 k 段階目の二重対角化を以下の手順で行う．

Step 1 $A^{(k-1)}$ の第 1 列のユークリッド ノルムを計算し、 α_k とする．

Step 2 α_k が ε_a より大きければ、左からの Householder 変換を行い d_k を求め、Step 6 へ．このとき、Householder 変換を行列 H_k で表せば

$$H_k A_{k-1} = \begin{bmatrix} d_1 & f_1 & & & \\ & \ddots & \ddots & & \\ & & \ddots & \ddots & \\ & & & d_{k-1} & f_{k-1} \\ & & & & \tilde{A}^{(k)} \end{bmatrix}$$

となる．ただし

$$\tilde{A}^{(k)} := \begin{bmatrix} d_k & \tilde{a}_{k,k+1}^{(k)} & \cdots & \tilde{a}_{kn}^{(k)} \\ & \vdots & \ddots & \vdots \\ & \tilde{a}_{m,k+1}^{(k)} & \cdots & \tilde{a}_{mn}^{(k)} \end{bmatrix}$$

である．

Step 3 α_k が ε_a 以下ならば、 d_k を 0 とおき、 $A^{(k-1)}$ の第 2 列目以降で絶対値が最大となる要素を探す．

Step 4 絶対値最大の要素が ε_a 以下ならば二重対角化終了．

Step 5 絶対値最大の要素が ε_a より大きければ、その要素を含む行と $A^{(k-1)}$ の第 1 行目を入れ替える．

Step 6 右からの Householder 変換を行い f_k を求める．

Step 7 $k = n$ ならば終了．そうでなければ $k = k + 1$ として Step 1 へ．

ただし、 $k = n$ では Step 3~7 は不要である．また、Step 6 における右からの Householder 変換を \tilde{H}_k で表せば、

$$H_k A_{k-1} \tilde{H}_k = A_k := \begin{bmatrix} d_1 & f_1 & & & \\ & \ddots & \ddots & & \\ & & \ddots & \ddots & \\ & & & d_k & f_k \\ & & & & a_{k+1,k+1}^{(k)} & \cdots & a_{k+1,n}^{(k)} \\ & & & & \vdots & \ddots & \vdots \\ & & & & a_{m,k+1}^{(k)} & \cdots & a_{mn}^{(k)} \end{bmatrix}$$

となる．ここで、 d_k がゼロのときは H_k は行置換行列となる．

アルゴリズム中、Step 1 のノルム計算は従来の二重対角化法でも必要である．そのため、行列がフルランクの場合は、残りの要素のゼロ判定およびピボット選択による行置換は行われず、従来の方法に比べ増加する処理は、Step 2 における条件評価 n 回のみである．

もし、行列がランク落ちしているならば、いずれかの d_k がゼロとなる．このとき、 $A^{(k-1)}$ の第 2 列目以降の要素がすべてゼロであると判断されれば、以降の計算が省略でき、高速に二重対角化が行える．

いま、 $k = p + 1 \leq n$ で上のアルゴリズムが打ち切られたとすると、得られた二重対角化は

$$A = \tilde{U} B \tilde{V}^T + O(\varepsilon_a) \quad (2)$$

と表せる．ただし、 \tilde{U} は左からの Householder 変換およびピボット選択による行置換行列をまとめた列正規直交行列、 \tilde{V} は右からの Householder 変換をまと

めた列正規直交行列, B は

$$B := \begin{bmatrix} d_1 & f_1 & & & \\ & \ddots & \ddots & & \\ & & \ddots & \ddots & \\ & & & d_p & f_p \end{bmatrix}$$

である. このとき, d_p と f_p が同時にゼロとはならない. さらに, B に対し, 陰的シフト QR アルゴリズムなどの二重対角行列に対する特異値分解法を適用することにより特異値分解 (1) が得られる.

本方法もしくは文献 4) の方法を用いる場合, 分解 (2) の \tilde{U}, \tilde{V} ならびに特異値分解 (1) の U, V は, 上のアルゴリズムの打ち切り回数に応じた列ベクトルのみ計算することが可能である. このとき, 行列 B, \tilde{U}, \tilde{V} のサイズはそれぞれ $p \times (p+1), m \times p, n \times (p+1)$ となる. 従来の方法では打ち切りは行わないので, \tilde{U}, \tilde{V} のサイズはそれぞれ $m \times n, n \times n$ である. もし, ゼロ特異値に対応した特異ベクトルが不要であるならば, \tilde{U}, \tilde{V} は打ち切り回数に応じた列のみ求めればよく, 二重対角化および特異値分解の計算を高速化できる.

また, 線形方程式 $Ax = b$ の最小二乗問題などは, 分解 (1) の U を陽に求める必要はなく, $U^T b$ が求まればよい. ベクトル $U^T b$ は, 二重対角化および二重対角行列の特異値分解法における左からの直交変換を, A と同様に b にも施すことにより求められる. このとき, ピボット選択による行置換は単に要素を入れ換えるだけでよく, 行置換による計算量の増加は抑えることができる.

ただし, 本方法により得られた二重対角行列 B は対角成分としてゼロをとりうるため, 打ち切り回数 p が必ずしも A の数値的なランクを表さないことに注意する. 同様に, U, V を打ち切り回数に応じた列のみ求めた場合, それらにゼロ特異値に対応した特異ベクトルが含まれることもある.

3. 数値実験

本章では我々の提案する方法の有効性を数値実験を用いて検証する. ここでは, 一様乱数により生成した行列に対する実験と, 典型的な悪条件問題である第 1 種フレドホルム型積分方程式を離散化した線形方程式に対する実験を示す.

数値実験環境は以下のとおりである. プログラムは C 言語で作成し, 計算はすべて倍精度実数演算で, 富士通 GP7000F モデル 900 の 1CPU において富士通の C コンパイラを用いて行った. 行列の二重対角化

法は, 従来の Householder 変換を 2 回行う方法, 文献 4) の方法と我々の提案する適応的ピボット選択付き二重対角化法を用いた. ゼロ判定のためのしきい値は $\varepsilon_a = 1.0 \times 10^{-14}$ とした. 特異値分解は, 得られた二重対角行列に対して文献 2) の Algorithm 8.6.2 を適用して求めた.

実験 1

まず, 一様乱数を用いて作成した行列に対する実験について述べる. テスト行列は, $[-1, 1]$ 区間の一様乱数を用いてランク個数の列ベクトルを作成し, それ以降の列ベクトルは, それらランク個数の列ベクトルの線形和により作成した. 線形和の係数も同様に一様乱数を用い, 最後に乱数により列の入れ換えを行った. サイズが 200×200 の行列を, ランクが 10 から 200 まで 10 刻みでそれぞれ 10 個, 乱数の初期値を変えて作成した. これらの行列に対して, 二重対角化, 特異値分解に要した時間を測定し, それぞれのランクについて平均をとった.

それぞれの方法を用いての二重対角化に要した時間を図 1 および図 2 に示す. ただし, 図中の Householder2, H3 method, present method はそれぞれ従来の方法, 文献 4) による方法, 我々の提案する方法を表す. また, 図 1 は分解 (2) における \tilde{U}, \tilde{V} を打ち切り回数 p に応じた列のみ求めた場合, 図 2 は \tilde{U}, \tilde{V} を 200 列すべて求めた場合の結果である. 従来の方法は計算量がランクには依存しないため一定の計算時間を要する. また, 打ち切りは行わないため, どちらにおいても同じ結果である. 文献 4) の方法は, ランクがサイズに比べて小さいときは高速であるが, フルランクに近くなると従来の方法に比べて計算時間がかかる. 本方法は, つねに文献 4) の方法よりも高速で, フルランクに近い行列に対しては, 従来の方法と同程度の時間で行列の二重対角化が行える. また, 二重対角化における打ち切り回数 p はいずれの行列に対しても行列のランクと一致し, ピボット選択による行置換は行われなかった.

二重対角化を含めた特異値分解全体の計算に要した時間を図 3 および図 4 に示す. 図 3 は U, V を打ち切り回数に応じた列のみ求めた場合, 図 4 は U, V を 200 列すべて求めた場合の結果である. 二重対角化の場合と同様に, フルランクに近い行列に対しては文献 4) の方法は, 従来の方法よりも時間がかかるが, 本方法は, ランクが小さい行列に対しては他の方法よりも高速に, フルランクに近い行列に対しては従来の方法と同程度の時間で特異値分解が求められることが分かる.

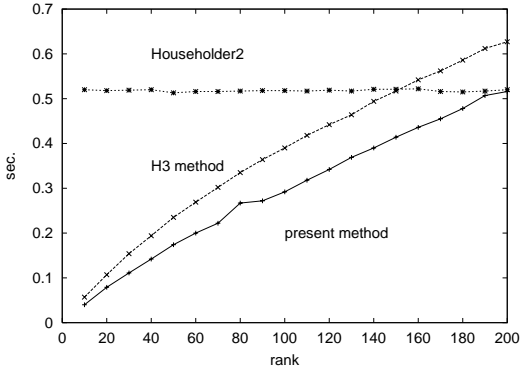


図1 従来の方法，文献4)の方法および本方法による二重対角化に要した計算時間の比較． \tilde{U} ， \tilde{V} を打ち切り回数 p に応じた列のみ求めた場合

Fig. 1 CPU-times in obtaining the bidiagonal form by using three methods with only p and $p + 1$ column vectors of \tilde{U} and \tilde{V} , respectively.

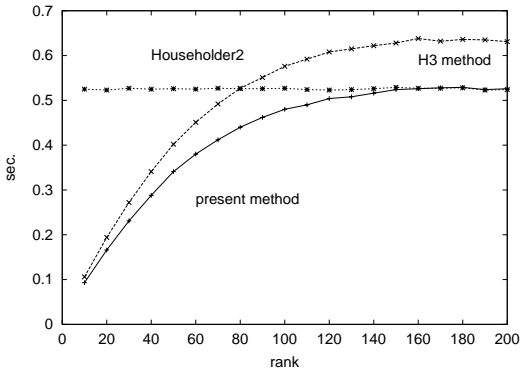


図2 従来の方法，文献4)の方法および本方法による二重対角化に要した計算時間の比較． \tilde{U} ， \tilde{V} の列をすべて求めた場合

Fig. 2 CPU-times in obtaining the bidiagonal form by using three methods with all column vectors of \tilde{U} and \tilde{V} .

二重対角行列の特異値分解は反復計算であるため，計算時間の一般的な比較は難しいが，本方法ではランク落ちした行列に対しては元の行列よりも小さいサイズの二重対角行列が得られることが多いため，その特異値分解に要する時間は従来の方法に比べ少なくなる。

なお，特異値および非ゼロな特異値に対応する特異ベクトルは，いずれの方法を用いても，その差は 1.0×10^{-13} 程度であった。

実験 2

次に典型的な悪条件問題である非退化核を持つ第1種フレドホルム型積分方程式

$$\int_0^1 K(s,t)f(t)dt = g(s), \quad 0 \leq s \leq 1 \quad (3)$$

に対する数値実験結果を示す。まず， n 点 Gauss-

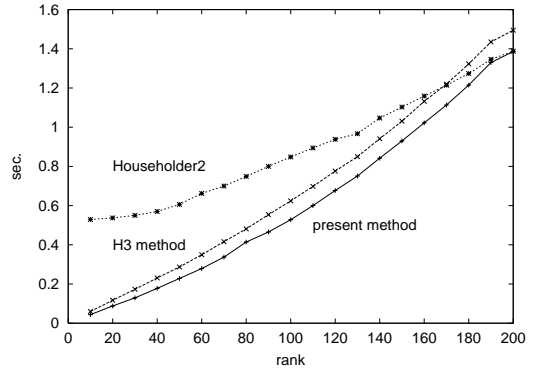


図3 従来の方法，文献4)の方法および本方法を用いての特異値分解に要した計算時間の比較．打ち切り回数 p に応じた特異ベクトルのみ求めた場合

Fig. 3 CPU-times in computing the SVD by using three methods with only p singular vectors.

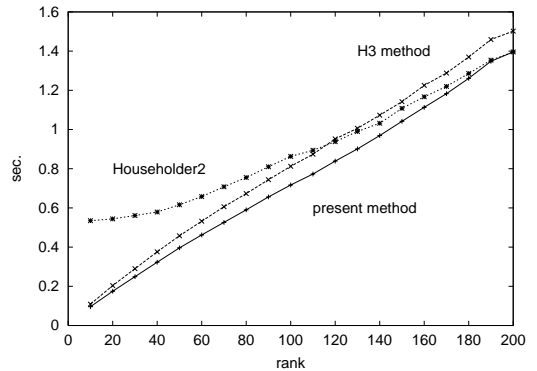


図4 従来の方法，文献4)の方法および本方法を用いての特異値分解に要した計算時間の比較．すべての特異ベクトルを求めた場合

Fig. 4 CPU-times in computing the SVD by using three methods with all singular vectors.

Legendre 数値積分則を用いて積分方程式 (3) を

$$\begin{aligned} A &= (a_{ij}), & a_{ij} &= K(t_i, t_j)\sqrt{w_j}, \\ \mathbf{b} &= (b_i), & b_i &= g(t_i), \end{aligned} \quad (4)$$

$$1 \leq i, j \leq n$$

と離散化し線形方程式 $Ax = b$ を得る。ただし， $\{t_j\}$ ， $\{w_j\}$ はそれぞれ数値積分則の分点と重みであり，解 x は

$$\mathbf{x} = (x_j), \quad x_j = f(t_j)\sqrt{w_j}, \quad 1 \leq j \leq n$$

である。テスト問題として我々は

$$\begin{aligned} K(s,t) &= \sqrt{s^2 + t^2}, \\ g(s) &= \frac{(1+s^2)^{\frac{3}{2}} - s^3}{3} \end{aligned}$$

を用いた。この問題の真の解は

$$f(t) = t$$

表1 悪条件問題 (4) に対する特異値分解の計算時間の比較
Table 1 CPU-times in computing the SVD of the matrix (4) by using three methods.

	CPU-times (sec.)
Householder 2	0.285
H3 method	0.154
Present method	0.129

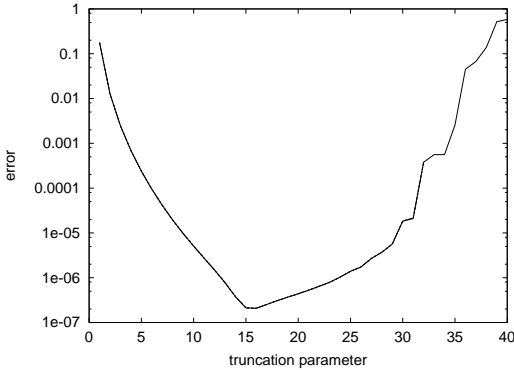


図5 従来の方法, 文献4)の方法および本方法を用いた打ち切り特異値分解法による近似解の打ち切り項数と, その誤差
Fig.5 Truncation parameter vs. errors of approximation solutions by the TSVD method with three methods.

である。また, $n = 100$ とした。

上により得られた行列をそれぞれの方法を用いて特異値分解を得るために要した時間を表1に示す。計算時間は, それぞれの方法において分解を100回行い, その平均をとった。二重対角化における打ち切り回数は文献4)の方法, 本方法とも $p = 32$ であった。また, 本方法ではピボット選択による行置換は行われなかった。この問題においても本方法が最も早く分解が得られることが分かる。

図5に, それぞれの方法を打ち切り特異値分解法³⁾(TSVD法)に適用して得られた近似解の誤差を示す。横軸はTSVD法の近似解

$$x_k := \sum_{j=1}^k \frac{u_j^T b}{\sigma_j} v_j$$

の打ち切り項数 k を, 縦軸は近似解 x_k と真の解とのユークリッドノルムによる誤差を常用対数尺度で表したものである。ただし, 上式の σ_j は行列 A の特異値, u_j, v_j はそれぞれ σ_j に対応した左, 右特異ベクトルである。また, σ_j は大きい順に並んでいるものとする。

図5は3つの方法を用いたTSVD法の各打ち切り項数における近似解の誤差が描かれているが, これらはほとんど重なっている。これより, いずれの方法を

用いてもほぼ同じ精度の近似解が得られることが分かる。

4. おわりに

我々は特異値分解を前提とした, 適応的ピボット選択付き二重対角化法を提案した。本方法は, 二重対角行列の対角成分がゼロとなる時のみ残りの成分のゼロ判定を行う。もし非ゼロ要素が含まれていれば行置換を行い, そうでなければ以降の処理を打ち切る。この打ち切り回数が行列のランクに近いほど高速に二重対角化を行うことができる。また, フルランク行列に対しては従来の方法に比べ増加する処理は対角成分のゼロ判定のみである。我々の行った数値実験では上記の打ち切り数が行列の階数と一致しなかった例はなく, そのため計算時間は, ランクが十分小さい場合には従来の方法ならびに文献4)の方法よりも高速であり, フルランクに近い行列に対しては従来の方法と同程度であった。また, 十分な精度で分解が行え, さらに, 典型的な悪条件問題である第1種フレドホルム型積分方程式に対しても従来の方法と同精度の近似解が得られることが確認された。

謝辞 有益なコメントをいただいた査読者に感謝します。

参考文献

- 1) Björck, Å.: *Numerical Methods for Least Squares Problems*, SIAM, Philadelphia (1996).
- 2) Golub, G.H. and Van Loan, C.F.: *Matrix Computation* —3rd ed., Johns Hopkins University Press, Baltimore (1996).
- 3) Hansen, P.C.: *Rank-Deficient and Discrete Ill-Posed Problems*, SIAM, Philadelphia (1998).
- 4) 細田陽介, 銚田雅之, 長谷川武光: ランク落ちした行列に対する特異値分解アルゴリズムについて, 情報処理学会論文誌, Vol.43, No.10, pp.3235-3238 (2002).
- 5) 中川 徹, 小柳義夫: 最小二乗法による実験データ解析: プログラム SALS, 東京大学出版会, 東京 (1982).

(平成15年2月12日受付)

(平成15年5月6日採録)



細田 陽介 (正会員)

1965年生。1994年名古屋大学大学院博士課程修了。広島市立大学情報科学部助手，富山県立大学工学部助手を経て，現在福井大学工学部情報・メディア工学科講師。博士(工学)。数値解析，特に悪条件な線形方程式の数値解法の研究に従事。日本応用数理学会会員。



銚田 雅之

1979年生。2002年福井大学工学部情報工学科卒業。現在福井大学大学院工学研究科情報工学専攻博士前期課程在学中。悪条件線形方程式の数値解法の研究に従事。



長谷川武光 (正会員)

1944年生。1972年名古屋大学大学院博士課程単位取得退学。工学博士。福井大学工学部情報・メディア工学科教授。主たる研究テーマは数値解析，数学ソフトウェアおよびインターネット上での数値計算環境の構築。日本応用数理学会，日本物理学会，AMS，IEEE(Computer)各会員。

