数理解析研究所講究録 1362 巻 2004 年 31-37

特異値分解アルゴリズムにおける 適応的ピボット選択を用いた行列の二重対角化

福井大学大学院工学研究科 鉾田 雅之 (Masayuki Hokoda)
Graduate Course in Engineering, University of Fukui
福井大学工学部 細田 陽介 (Yohsuke Hosoda)
Faculty of Engineering, University of Fukui
福井大学工学部 長谷川 武光 (Takemitsu Hasegawa)
Faculty of Engineering, University of Fukui

1 はじめに

我々の目的は，行列 A の特異値分解

$$A = U\Sigma V^T \tag{1}$$

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，$m \geq n$ として一般性を失わない．また，U, V は列正規直交行列，Σ は対角行列であり，そ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，悪条件問題，信号処理，画像処理など，さまざまな分野で用いられている [1, 3, 6]．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もつ．

通常，特異値分解は

- 直交変換による行列に二重対角化
$$A = \tilde{U}B\tilde{V}^T$$
- 二重対角行列の特異値分解
$$B = \hat{U}\Sigma\hat{V}^T$$

の手順で計算される．ただし，B は二重対角行列，$\tilde{U}, \hat{U}, \tilde{V}, \hat{V}$ は列正規直交行列であり，

$$U := \tilde{U}\hat{U}, \qquad V := \tilde{V}\hat{V}$$

とおくことにより特異値分解 (1) が求まる．

行列の二重対角化は，左右から相互にハウスホルダー変換を行う方法，もしくは，ハウスホルダー変換による QR 分解を行った後に高速ギブンス変換を用いる方法がある．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的シフト QR アルゴリズムが用いられることが多い．本論文では，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について考察する．なお，本論文では左右からのハウスホルダー変換による二重対角化法を比較の対象とする．

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伴う．これに対して我々はピボット選択付きハウスホルダー変換による二重対角化法を提案した [4]．この方法は行列のサイズに比べ十分に小さなランクの行列に対しては高速であるが，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対しては従来の方法より

も計算時間がかかるという欠点をもつ．一般に行列のランクは分解を行うまでは不明であるため，この方法の常用は問題が残る．

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 [5]．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，残りの要素のゼロ判定ならびに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以降の無駄な計算を省き，フルランク行列に対しては，行置換を行わず，従来の方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ならびに文献 [4] 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．次節では文献 [4] の方法ならびに我々の提案するう適応的ピボット選択付き二重対角化法の算法の詳細を述べる．3 節で数値実験およびその考察を行い，4 節でまとめる．

2 行列の二重対角化

本節では，ピボット選択を用いた行列の二重対角化法について述べる．まず，常時ピボット選択付きハウスホルダー変換を用いた二重対角化法 [4] について説明し，その問題点を指摘する．次に我々の提案する適応的ピボット選択付き二重対角化法 [5] について述べる．

ただし，以下の 2 通りの算法において，簡単のため行列などに同じ記号を用いるが，異なるものであることに注意する．

いま，分解すべき行列 A と，行列要素のゼロ判定を行うための定数 $\varepsilon_a > 0$ が与えられ，$k-1$ 段階目までの二重対角化

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

が得られたものとする．ただし，$A_0 := A$ である．上式右辺の右下 $(m-k+1) \times (n-k+1)$ 小行列を $A^{(k-1)}$ とおく．常時ピボット選択付き二重対角化法は，k 段階目の二重対角化を以下の手順で行う．

算法 1．（常時ピボット選択付き二重対角化法）

Step 1 $A^{(k-1)}$ の要素の中で，絶対値が最大となる要素を探す．もし，その値が ε_a 以下ならば二重対角化終了．

Step 2 絶対値最大の要素を含む行と $A^{(k-1)}$ の第 1 行とを入れ換える．

Step 3 $A^{(k-1)}$ の第 1 列において，絶対値が最大となる要素を探す．もしその値が ε_a 以下ならば $d_k = 0$ とおき Step 6 へ．

Step 4 $A^{(k-1)}$ の第 1 列の第 2 項以下を 0 とするようなハウスホルダー変換を左から $A^{(k-1)}$ に行う．

Step 5 上の操作で更新された $A^{(k-1)}$ の第 1 行において，第 2 項以降で絶対値が最大となる要素を探す．もし，その値が ε_a 以下ならば $f_k = 0$ とおき Step 7 へ.

Step 6 $A^{(k-1)}$ の第 1 行の第 3 項以降を 0 とするようなハウスホルダー変換を右から行い，更新された $A^{(k-1)}$ の第 1 行第 2 項を f_k とおく.

Step 7 $k = k + 1$ として Step 1 へ.

ただし，$k = n - 1$ のときは Step 4~6 が，$k = n$ のときは Step 2 が不要である.

ここで，Step 2 の行の入れ換えを置換行列 Π_k，Step 4 および Step 6 でのハウスホルダー変換をそれぞれ行列 $\tilde{U}_k$, $\tilde{V}_k$ で表すと，上の操作は

$$A_k = \tilde{U}_k \Pi_k A_{k-1} \tilde{V}_k$$

と行列表記することができる．この操作を $A^{(k-1)}$ の要素がすべて ε_a 以下になるまで，もしくは $k = n$ まで行うことにより二重対角行列が得られる．いま，上の算法での打ち切り回数が $k = p + 1 \leq n$ であったとすると，正規直交行列 $\tilde{U}$, $\tilde{V}$ をそれぞれ

$$\tilde{U} := \Pi_1 \tilde{U}_1 \cdots \Pi_p \tilde{U}_p, \quad \tilde{V} := \tilde{V}_1 \cdots \tilde{V}_p$$

とおくことにより，得られた二重対角行列 B は

$$\begin{aligned} B &= \tilde{U}^T A \tilde{V} + O(\varepsilon_a) \\ &= \begin{bmatrix} d_1 & f_1 & & \\ & \ddots & \ddots & \\ & & d_p & f_p \\ & & & \end{bmatrix} \end{aligned} \tag{2}$$

と書ける．この B に対して，二重対角行列の特異値分解法を適用することにより，行列 A の特異値分解が得られる.

この方法では，$A^{(k-1)}$ のすべての要素に対して，絶対値が最大となる要素を探し，ランク落ちを判定し，そして，ピボット選択による行の入れ換えを行う．ランク落ちと判定されれば，以降の二重対角化の手間が省略できるため，高速化が期待できる．反面，常時行置換を行うことになり，ランクが n よりも十分に小さい場合には高速であるが，フルランクもしくはランクが n に近い場合には従来の方法よりも計算の手間が増えるという欠点がある．本来行列のランクは分解を行う前は不明であるため，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 [5] を提案する．以下に本方法の算法を示す.

算法 2． (適応的ピボット選択付き二重対角化法)

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，左からのハウスホルダー変換を行い d_k を求め Step 6 へ.

このとき，ハウスホルダー変換を行列 $\tilde{U}_k$ で表せば

$$\tilde{U}_k A_{k-1} = \begin{bmatrix} d_1 & f_1 & & & & & \\ & \ddots & \ddots & & & & \\ & & d_{k-1} & f_{k-1} & & & \\ & & & d_k & \tilde{a}^{(k)}_{k,k+1} & \cdots & \tilde{a}^{(k)}_{kn} \\ & & & & \vdots & \ddots & \vdots \\ & & & & \tilde{a}^{(k)}_{m,k+1} & \cdots & \tilde{a}^{(k)}_{mn} \end{bmatrix}$$

となる．

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 行目を入れ換える．

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 $\tilde{V}_k$ で表せば，

$$A_k = \tilde{U}_k A_{k-1} \tilde{V}_k$$

と表すことができる．ただし，d_k が 0 のときは，H_k は行置換行列となる．

算法 2 の中で，Step 1 のノルムの計算は従来の二重対角化法においても必要な操作である．そのため，行列がフルランクの場合は，$A^{(k-1)}$ の第 2 列目以降のゼロ判定ならびに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 における条件判定 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，ここで二重対角化が終了し，以降の計算が省略できるため，高速に二重対角化が行える．

いま，上の算法での打ち切り回数が $k=p+1\le n$ であったとすると，

$$\tilde{U} := \tilde{U}_1 \cdots \tilde{U}_p, \quad \tilde{V} := \tilde{V}_1 \cdots \tilde{V}_p$$

とおくことにより二重対角化 (2) が得られる．このとき，d_p と f_p は同時にはゼロとならない．さらに，B に二重対角行列のための特異値分解アルゴリズムを適用することにより，A の特異値分解が得られる．

算法 1 もしくは算法 2 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，式 (2) の行列 $B,\tilde{U},\tilde{V}$ 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れないので，$B,\tilde{U},\tilde{V}$ のサイズはそれぞれ $n\times n, m\times n, n\times n$ 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ベクトルのみを求めればよく，その分，二重対角化ならびに特異値分解での計算を高速化することができ

る．さらに，A がランク落ちしているならば B のサイズも従来の方法に比べ小さくなるため，二重対角行列の特異値分解においても計算量を軽減することができる．

線形方程式 $Ax = b$ の最小二乗問題などに特異値分解を用いるときは，分解 (1) の U を陽に求める必要はなく，$U^T b$ のみが必要となる場合が多い．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アルゴリズム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，U を構築するよりも少ない手間で求めることができる．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を抑えることができる．

ただし，算法 1 ならびに算法 2 により求め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わ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する特異ベクトルが含まれることもある．

3 数値実験

本節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．

数値実験は以下の手順で行った．テスト行列は，まず $[-1, 1]$ 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．

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 1 CPU 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，常時ピボット選択による行置換を行う算法 1，ならびに我々の提案する算法 2 の適応的ピボット選択付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．

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 に，特異値分解全体の計算に要した時間を図 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，算法 1，ならびに我々の提案する方法である算法 2 を表す．また，図 1，2 の左図は，分解 (2) の $\tilde{U}, \tilde{V}$ もしくは分解 (1) の U, V を打ち切り回数 p に応じた列のみ求めた場合．右図は 200 列すべてを求めた場合の結果である．

二重対角化において，従来の方法は計算量がランクに依存しないため，一定の計算時間を要する．また，打ち切りは行われないため，左右どちらの図においても同じ結果である．算法 1 は，ランクが行列のサイズに比べて十分に小さいときには高速であるが，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．算法 2 は，常に算法 1 よりも高速で，かつ，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ことがわかる．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，算法 2 におけるピボット選択による行置換は行われなかった．

特異値分解全体に要した計算時間においても二重対角化と同様に，フルランクに近い行列に対しては算法 1 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

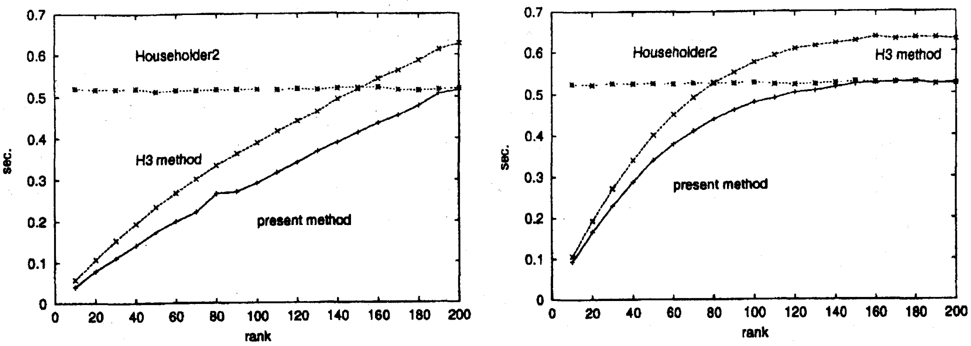

図 1: 従来の方法，算法 1，ならびに我々の提案する方法による二重対角化に要した計算時間の比較．左図は $\tilde{U}, \tilde{V}$ を打ち切り回数 p に応じた列のみ求めた場合．右図は $\tilde{U}, \tilde{V}$ の列をすべて求めた場合．

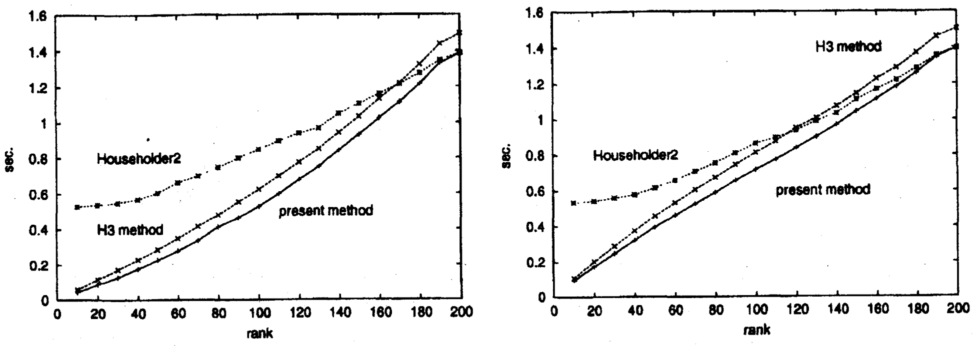

図 2: 従来の方法，算法 1，ならびに我々の提案する方法を用いての特異値分解に要した計算時間の比較．左図は打ち切り回数 p に応じた数のみの特異ベクトルを求めた場合．右図はすべての特異ベクトルを求めた場合．

られることがわかる．

二重対角行列の特異値分解アルゴリズム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，算法 1 および算法 2 を用いると，ランク落ちした行列に対しては元の行列よりもサイズが小さい二重対角行列が得られるため，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，特異値および非零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．

4 おわりに

我々は行列の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ときのみ，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

記の打ち切り回数が行列のランクと一致しなかった例はなく，そのため計算時間は，ランクが十分に小さい場合には従来の方法ならびに算法1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ることも確認した．

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, 鉾田雅之, 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] 細田陽介, 鉾田雅之, 長谷川武光: 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, 情報処理学会論文誌, Vol. 44, No. 7, pp. 1649–1654 (2003).

[6] 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).