Implementation details of an extended oqds algorithm for singular values

Sho Araki¹, Kinji Kimura¹, Yusaku Yamamoto² and Yoshimasa Nakamura¹

¹ Graduate School of Informatics, Kyoto University, Yoshidahonmachi, Sakyoku, Kyoto 606-8501, Japan
² The University of Electro-Communications, Chohugaoka, Chohu, Tokyo 182-8585, Japan

E-mail araki@amp.i.kyoto-u.ac.jp

Received June 20, 2014, Accepted November 5, 2014

Abstract

We introduce an extended oqds algorithm for singular values of lower tridiagonal matrix which is a condensed form of inputted full matrix. Reduction to the lower tridiagonal matrix is able to be performed using cache-efficient block Householder method based on BLAS 2.5 routines. In this letter, we describe the implementation details of the latter algorithm such as the shift strategy and criteria for deflation and splitting. The effectiveness of our approach is demonstrated by numerical experiments.

Keywords singular value computation, extended oqds algorithm, shift strategy, convergence criteria

Research Activity Group Algorithms for Matrix / Eigenvalue Problems and their Applications

1. Introduction

Singular value decomposition is one of the most important matrix operations in numerical linear algebra and it is applied to many fields in engineering. Nowadays, it is a major challenge to develop high performance singular value computation solver for large scale problems.

In this letter, we describe implementation details of extended oqds (orthogonal quotient-difference with shifts) algorithm for singular values of lower tridiagonal matrices. If we adopt the extended oqds algorithm to singular value problem of dense full matrix, the input matrix is first reduced to a lower tridiagonal form by the block Householder method [1]. Then, its singular values are computed with the extended oqds algorithm for lower tridiagonal matrices [2]. The advantage of using the lower tridiagonal form is that we can use the block Householder method, which is based on the BLAS Level 2.5 (L2.5) and is a cache-efficient algorithm. The disadvantage is that computing the singular values of a lower tridiagonal matrix with the extended oqds algorithm is much slower than computing the singular values of a bidiagonal matrix with the dqds algorithm [3]. However, since most of the computation time is spent in the former (preprocessing) step, we expect that the advantage is larger and the total computation time can be shortened if we implement the extended oqds efficiently and scalably.

To improve the performance and scalability of the extended oqds algorithm, we adopt Algebraic shift which computes lower bound of the singular values of lower tridiagonal matrices. Moreover, we design two types of convergence criteria by estimating \( p \)-norms of perturbation matrices to reduce iterations of the oqds algorithm. The behavior of the convergence criteria changes by the choice of norms of perturbation matrices.

At last, we show some result of numerical experiments and demonstrate the advantages of the proposed block Householder lower tridiagonalization plus the extended oqds algorithm and discuss the effectiveness of Algebraic shift and \( p \)-norm convergence criteria.

2. Improvements of Cash Efficiency of Householder Reduction

2.1 Basic Householder method

The basic Householder method is used in the preprocessing step to reduce the input full matrix to a bidiagonal matrix, so that bidiagonal singular values algorithms such as the dqds algorithm can be applied. It performs bidiagonalization by eliminating all off-diagonal elements except for the lower subdiagonal ones by Householder transformations.

In the Householder method, matrix-vector multiplications and rank-2 updates occupy almost all of computation time. However, this method is cache inefficient because these operations require \( O(n^2) \) memory access for \( O(n^3) \) operations if the size of the input matrix is \( n \).

2.2 Block Householder method

The block Householder method is one of improved methods which is based on cash efficient BLAS L2.5 routines. This algorithm is originally proposed as the first step of multistep tridiagonalization method by Bischof, and Wu, et al. [1, 4], and reduces dense full matrices to narrow band matrices.

Imamura applies the divide and conquer eigensolver to the narrow band matrices [5] and gets higher performance than classical tridiagonalization plus divide and conquer method. Imamura also reports that on the K
computer, maximum performance can be obtained by reducing the full matrix to a pentadiagonal matrix. In this letter, we use the block Householder method to reduce the full matrix to a lower tridiagonal matrix so that an extended oqds algorithm can be applied.

3. The oqds Algorithm for Lower Tridiagonal Matrices

3.1 Oqds algorithm

The oqds algorithm is one of singular value computation algorithms for lower bidiagonal matrices proposed by von Matt in [6]. The algorithm is based on Cholesky-LR method. Cholesky-LR method computes the singular values of lower triangular matrices by the iteration of the following form of Cholesky decomposition with shift \( \tau(i) \)

\[
L^{(i+1)}(i+1)^T = L(i)^T L(i) - \tau(i)^2 I.
\]

By the iteration, the matrix \( L(i) \) converges to a diagonal matrix and its diagonal elements approach to singular values of matrix \( L(0) \). However, explicit Cholesky-LR method requires a lot of iterations which causes serious decline in performance and accuracy. The oqds algorithm is specialized form of the Cholesky-LR method which is formulated practically and implements suitable shift strategy for bidiagonal matrices. The extended oqds algorithm is specialized form for lower tridiagonal matrices.

4. Shift Strategy

In the Cholesky-LR method, proper choice of the shift value \( \tau(i) \) significantly accelerates convergence of the oqds algorithm. The shift value \( \tau(i) \) must be smaller than the minimum singular value of the matrix \( L(i) \) to keep the positive-definiteness of \( L(i+1)L(i+1)^T \). It is reported in [6] that the oqds algorithm is accelerated if we set the shift value \( \tau(i) \) close to the minimum of singular values of matrix \( L(i) \). Therefore, we need a method to estimate a lower bound of the eigenvalues of lower tridiagonal matrix \( L(i) \) or the minimum eigenvalue of \( L(i)L(i)^T \). In this section, we introduce the so-called Algebraic shift for lower tridiagonal matrices.

4.1 Algebraic shift

We apply Algebraic shift proposed by Yamashita et al. in [7] to accelerate the extended oqds algorithm for lower tridiagonal matrices. Algebraic shift is composed of three shifts; the generalized Newton, Laguerre and Kato-Temple shifts obtained from the trace of inverse matrix and it is reported by Yamashita that for bidiagonal matrices, the shift delivers an extremely sharp lower bound. In this subsection, we consider the application of Algebraic shift to lower tridiagonal matrices.

4.2 Differential method for obtaining the trace of inverse of pentadiagonal matrices

For a positive-definite symmetric matrix \( A \) and an arbitrary positive integer \( p \), the value of \( \text{Tr}((A^{-p})^{-1}) \) is a lower bound of the eigenvalues of \( A \). The lower bound is known as the generalized Newton shift. Other two types of shift, Laguerre and Kato-Temple shifts are also obtained from the value of \( \text{Tr}(A^{-p}) \).

Then, finding the value of \( \text{Tr}((L^T L)^{-p}) \) for symmetric pentadiagonal matrix \( (L^T L)^{-p} \), we get a lower bound of singular values of lower tridiagonal matrix \( L \). We adapt the differential method proposed in [2] of computing the value of \( \text{Tr}((L^T L)^{-1}) \) and \( \text{Tr}((L^T L)^{-2}) \) for the Algebraic shift.

5. Convergence Criteria

In this section, we introduce deflation and splitting for the oqds algorithm and design convergence criteria for these operations.

5.1 Deflation and splitting

Deflation and splitting are familiar stopping method for qd algorithm. In the next subsection, we consider the situation that we can perform the deflation or splitting properly where the values of subdiagonal and second-subdiagonal elements of lower tridiagonal matrices are so small. Specifically, we estimate the perturbation of eigenvalues of \( L^T L \) and \( LL^T \) by Weyl’s theorem.

5.2 1-norm convergence criteria

We prefer \( p \)-norm estimation because we have to assess the perturbation precisely to avoid extra iteration of extended oqds step which has higher computational complexity than that of qdqs step. We consider the case where \( p = 1 \) at first.

Let us write

\[
L = \begin{bmatrix}
\alpha_1 & \beta_1 & \alpha_2 \\
\gamma_1 & \beta_2 & \alpha_3 \\
& \ddots & \ddots \\
& & \gamma_{n-2} & \beta_{n-1} & \alpha_n
\end{bmatrix}
\]

and

\[
\hat{L} := L - \beta_k e_k e_k^T
\]

which is the matrix equal to \( L \) except for zero at \((k + 1, k)\)-entry. Then

\[
L^T L = \hat{L}^T L + E_1,
\]

\[
LL^T = \hat{L}L^T + E_2
\]

hold, where \( E_1 \) and \( E_2 \) are perturbation matrices defined by

\[
E_1 := \beta_k^2 e_k e_k^T + \alpha_{k+1}\beta_k (e_k e_{k+1}^T + e_{k+1} e_k^T)
+ \beta_k \gamma_{k-1} (e_{k-1} e_k^T + e_k e_{k-1}^T),
\]

\[
E_2 := \beta_k^2 e_k e_k^T + \alpha_{k+1}\beta_k (e_k e_{k+1}^T + e_{k+1} e_k^T)
+ \beta_k \gamma_k (e_k e_{k+1}^T + e_{k+1} e_k^T).
\]

Theorem 1 (Weyl’s monotonicity theorem [8, 9])

For an \( n \)-by-\( n \) positive-definite matrix \( A \), let \( \lambda_i (A) \) denote the \( i \)-th largest eigenvalue of \( A \). Then, there exist real numbers \( u_i \) and \( v_i \) such that

\[
\lambda_i (L^T L) = \lambda_i \left( \hat{L}^T \hat{L} \right) + u_i \| E_1 \|_p,
\]

\[
\lambda_i (LL^T) = \lambda_i \left( \hat{L} \hat{L}^T \right) + v_i \| E_2 \|_p
\]
where $|u_i| \leq 1$, $|v_i| \leq 1$. It should be noted that $\|\cdot\|_p$ is $p$-norm.

From the definitions (1) and (2) of $E_1$ and $E_2$, we have

$$\|E_1\|_1 = \|E_1\|_{\infty} = |\beta_k| \left( |\alpha_{k+1}| + |\beta_k| + |\gamma_{k-1}| \right),$$
$$\|E_2\|_1 = \|E_2\|_{\infty} = |\beta_k| \left( |\alpha_k| + |\beta_k| + |\gamma_k| \right).$$

By Weyl's monotonicity theorem, we thus get the numerical criterion to regard the subdiagonal element $\beta_k$ as zero:

$$T + |\beta_k| \left( |\beta_k| + \min\{ |\alpha_{k+1}| + |\gamma_{k-1}|, |\alpha_k| + |\gamma_k| \} \right) \simeq T,$$

where `$\simeq$' means that the left-hand side and the right-hand side are numerically equal and $T$ is the square summation of shift values previously applied written as follows:

$$T := \sum \tau^{(i)}.$$ 

We assume that $\beta_k$ is so small and negligible provided that (5) holds numerically.

Similarly, we get the numerical criterion for neglecting a second-subdiagonal element $\gamma_k$. On the setting of

$$\hat{L} := L - \gamma_k e_{k+2} e_k^T,$$

the perturbation matrices are given by

$$E_1' := \gamma^2 e_k e_k^T + \alpha_{k+2} \gamma_k (e_{k+2} e_k^T + e_k e_k^T) + \beta_{k+1} \gamma_k (e_{k+1} e_k^T + e_k e_k^T),$$
$$E_2' := \gamma^2 e_k e_k^T + \alpha_k \gamma_k (e_{k-2} e_k^T + e_k e_k^T) + \beta_k \gamma_k (e_{k-1} e_k^T + e_k e_k^T).$$

Then, by evaluating the 1- and $\infty$-norms of these matrices, we obtain the criterion for neglecting a second-subdiagonal element $\gamma_k$ as follows:

$$T + |\gamma_k| \left( |\gamma_k| + \min\{ |\alpha_{k+2}| + |\beta_{k+1}|, |\alpha_k| + |\beta_k| \} \right) \simeq T.$$

5.3 2-norm convergence criteria

In the previous subsection, we design convergence criteria by estimating the perturbation of equation (3) and (4) with 1-norm. Although the choice of norm dimension $p$ is arbitrary, we can assess the convergence more sharply if we obtain 2-norm of perturbation matrices $E_1$ and $E_2$. In the case of lower tridiagonal matrices, characteristic polynomial of $E_1$ is

$$f(E_1) = x^3 - x^2 \beta_k^2 - (x \beta_k^2 \gamma_{k+1} + \beta_k^2 \alpha_{k+1}^2),$$

and hence we can factorize and reduce the cubic characteristic polynomial to quadratic polynomial as follows

$$f(E_1) = x^2 - x \beta_k^2 - (\beta_k^2 \gamma_{k+1} + \beta_k^2 \alpha_{k+1}^2).$$

Then, we obtain the following 2-norm convergence criterion

$$T + \frac{1}{2} |\beta_k| \times \left( |\beta_k| + \sqrt{\beta_k^2 + \min\{ 4 \alpha_{k+1}^2 + 4 \gamma_{k+1}^2, 4 \alpha_k^2 + 4 \gamma_k^2 \} } \right) \simeq T.$$ 

Similarly, for $E_2$, we obtain the criterion

$$T + \frac{1}{2} |\gamma_k| \times \left( |\gamma_k| + \sqrt{\gamma_k^2 + \min\{ 4 \alpha_k^2 + 4 \beta_k^2 + 4 \alpha_{k+1}^2 + 4 \beta_{k+1}^2 \} } \right) \simeq T.$$ 

By using these 2-norm criteria, the oqds algorithm may not run faster because 2-norm criteria require square root computation though the criteria allow to perform deflation and splitting faster.

6. Numerical Experiments

Some numerical experiments were performed for following three purposes;

(1) Confirm the effectiveness of Algebraic shift.
(2) Benchmark of proposed block Householder lower tridiagonalization plus extended oqds algorithm.
(3) Observe the behavior of the oqds algorithm with two convergence criteria.

The numerical experiments were performed on a Linux PC with Intel Core i7 920 (Nehalem) 2.66GHz and DDR3-1066 12GB memory. Each program is compiled by Intel C/C++ compiler with -fast and -mkl option.

First, we execute the extended oqds for $1\times1\times1$ matrix and random matrix $\alpha_i \in [1,3]$, $\beta_i \in [-1,1]$ and $\gamma_i \in [-1,1]$ of size 1000 with Gerschgorin [10] and Algebraic shift. We define $1\times1\times1$ matrix as lower tridiagonal matrix all whose nonzero elements are set to one. We observe the convergence of $\gamma_{n-2}$ which is the bottom of second-subdiagonal element and generally converges to zero at first. Absolute values of $\gamma_{n-2}$ are plotted in Fig. 1. The abscissa of the the graph represents the iteration count, and the ordinate represents the absolute value of $\gamma_{n-2}$.

Next, we compare the computation time of conventional bidiagonalization and lower tridiagonalization. We apply the classical Householder reduction method to bidiagonalization while apply the block Householder method to lower tridiagonalization, respectively. The DGEBD2 subroutine on LAPACK [11] is used for bidiagonalization while hand-coded reduction program with BLAS L2.5 is used for lower tridiagonalization. Table 1 shows the computation time of each algorithm. The first
row shows the size of matrices. The second and the third rows show the computation time taken by the classical Householder bidiagonalization and the block Householder lower tridiagonalization, respectively.

We compare the time of singular value computation by dqds for bidiagonal matrices and oqds for lower tridiagonal matrices. We adopt the DBDSQR routine in LAPACK as dqds algorithm while the oqds is our hand-coded program with 1-norm convergence criteria. Table 2 shows the computation time of each algorithm. The first row shows the size of matrices. The second and the third row show the computation time of the dqds algorithm and the extended oqds algorithm, respectively.

We perform one more experiment to observe the difference of behavior between the oqds with 1-norm convergence criteria and the oqds with 2-norm convergence criteria. Table 3 shows the computation times and number of deflations and splitting executed in oqds with two types of convergence criteria for a 20000 by 20000 lower tridiagonal random matrix.

7. Discussion and conclusions

We improve the performance and scalability of the extended oqds algorithm for singular values of lower tridiagonal matrices through the adoption of Algebraic shift and $p$-norm convergence criteria. As a result, we can adapt the block Householder lower tridiagonalization plus the extended oqds algorithm for lower tridiagonal matrices to singular value computation of dense full matrices. Though computing the singular values of lower tridiagonal matrices still takes longer time than computing the singular values of bidiagonal matrices (Table 2), the increment is small in absolute value. On the other hand, the preprocessing time, which accounts for most of the total computation time, is significantly reduced thanks to the use of the cache-efficient block Householder method (Table 1). As a result, the total computation time of the proposed algorithm is less than half of the conventional method.

For acceleration of the extended oqds algorithm, we adopt Algebraic shift which is proposed as shift strategy for dqds algorithm. We apply a differential method to compute the trace for pentadiagonal matrices to adopt Algebraic shift to the extended oqds algorithm and then, the convergence speed of the extended oqds algorithm is accelerated as Fig. 1 shows. The Algebraic shift accelerates the convergence of the extended oqds algorithm and gives the algorithm scalability for specific type of matrices which is incompatible for Gerschgorin shift.

Moreover, we design two types of convergence criteria for deflation and splitting required for the implementation of the extended oqds algorithm. 1-norm convergence criteria has less complexity and takes less time. On the other hand, 2-norm convergence criteria performs sharper test and split matrices more frequently, as Table 3 shows. We would take additional research for the two types of criteria in aspects of performance, accuracy and suitability for parallel computer.

As a result, the performance and scalability of the extended oqds algorithm for singular values of lower tridiagonal matrices are improved with the adoption of Algebraic shift and $p$-norm convergence criteria.

As a future work, we have to parallelize the oqds algorithm to compute extremely large matrices by using massively parallel system. Furthermore, exact error analysis should be made and we ought to check out the accuracy of the algorithm after improving the implementation and setting proper test matrices which have known singular values.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number 24360038.

References

[1] C. H. Bischof, B. Lang and X. Sun, The SBR toolbox - software for successive band reduction, ACM TOMS, 26 (2000), 602-616.
[2] S. Araki, H. Tanaka, K. Kimura and Y. Nakamura, Implementation of the orthogonal QD algorithm for lower tridiagonal matrices, in: Proc. of PDPTA 2013, Vol. I, pp. 161-167, Las Vegas, 2013.
[3] K. V. Fernando and B. N. Parlett, Accurate singular values and differential qd algorithms, Numer. Math., 67 (1994), 191-229.
[4] Y. J. Wu, P. A. Alpatov, C. H. Bischof and R. A. van De Geijn, A parallel implementation of symmetric band reduction using PLAPACK, in: Proc. of SPLC 1996, PRISM Working Note 35, Missippi, 1996.
[5] T. Imamura, S. Yamada and M. Machida, Narrow-band reduction approach of a DRSM eigensolver on a multicore-based cluster system, Adv. Parallel Comput., 19 (2010), 91-98.
[6] U. von Matt, The orthogonal QD algorithm, SIAM J. Sci. Comput., 18 (1997), 1163-1186.
[7] M. Takata, H. Toyokawa, H. Ishigami, K. Kimura, T. Yamasita, M. Iwasaki and Y. Nakamura, New shift strategy in the dqds algorithm and the m2dLVs algorithm for singular values (in Japanese), ACS, 6 (2013), 94-107.
[8] B. N. Parlett, The Symmetric Eigenvalue Problem, Prentice Education Canada, Englewood Cliffs, 1989.
[9] J. H. Wilkinson, The Algebraic Eigenvalue Problem, Clarendon Press, Oxford, 1995.
[10] Gerschgorin, Uber die abgrenzung der digenwerte einer matrix, Izv. Akad. Nauk., USSR Otd. Fiz.-Mat. Nauk, 6 (1931), 749-754.
[11] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov and D. Sorenson, LAPACK User’s Guide, 2nd ed., SIAM, Philadelphia, 1995.