Differential qd algorithm for totally nonnegative band matrices: convergence properties and error analysis

Yusaku Yamamoto¹ and Takeshi Fukaya¹

Department of Computational Science and Engineering, Nagoya University, Furo-cho, Chikusa-ku, Nagoya, 464-8603, Japan
E-mail yamamoto@na.cse.nagoya-u.ac.jp

Received May 27, 2009, Accepted August 13, 2009

Abstract

We analyze convergence properties and numerical properties of the differential qd algorithm generalized for totally nonnegative band matrices. In particular, we show that the algorithm is globally convergent and can compute all eigenvalues to high relative accuracy.

Keywords eigenvalue, totally nonnegative, band matrix, qd algorithm, error analysis

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

1. Introduction

The dqds (differential quotient-difference with shift) algorithm [1] is one of the most successful algorithms for computing the eigenvalues of a symmetric positive-definite tridiagonal matrix. It is mathematically equivalent to the LR algorithm, but instead of working on the tridiagonal matrix \( T \), it uses the elements of its bidiagonal factor \( B \), where \( T = B^T B \), as basic variables and performs the LR step \((B^{(n+1)})^T B^{(n+1)} = B^{(n)} (B^{(n)})^T\) without forming \( B^{(n)} (B^{(n)})^T \) explicitly. The dqds algorithm has a unique feature that it can compute all the eigenvalues to high relative accuracy and is used as one of the key ingredients in the MR³ algorithm for the symmetric tridiagonal eigenproblem [2].

The high relative accuracy of the dqds algorithm is made possible thanks to the following two properties:

(i) The algorithm involves only positive variables and uses no subtractions except in the introduction of origin shifts. Thus the rounding errors arising in the computation of \( B^{(n)} \to B^{(n+1)} \) are kept small in the sense of relative error.

(ii) Small relative errors in the elements of \( B^{(n)} \) cause small relative perturbation in the eigenvalues.

It is therefore interesting to investigate if the dqds algorithm can be extended to other types of matrices while retaining these useful properties.

In this paper, we consider a class of totally nonnegative band matrices. A matrix is called totally nonnegative (TN) if all of its minors are nonnegative [3]. The TN band matrices can be regarded as a generalization of the symmetric positive-definite tridiagonal matrices and the dqd (differential qd) algorithm, which is a shiftless version of the dqds, can be naturally extended to deal with this type of matrix. We study the convergence properties and numerical properties of the dqd algorithm applied to the TN band matrix. In particular, we prove its global convergence and show that it shares the properties (i) and (ii) of the dqds algorithm. Using these facts, we show that the algorithm can compute all eigenvalues of a TN band matrix to high relative accuracy.

Recently, Fukuda et al. formulated a new algorithm for eigenvalue computation based on an integrable system called the discrete hungry Lotka-Volterra (dLV) system [4]. We also point out that there is a close relationship between this algorithm and the dqd algorithm for the TN band matrix.

2. The differential qd algorithm for a totally nonnegative band matrix

Let \( L_i \ (i = 1, \ldots, m_L) \) and \( R_i \ (i = 1, \ldots, m_R) \) be \( m \times m \) lower and upper bidiagonal matrices defined by

\[
L_i = \begin{pmatrix}
q_{i1} & 1 & & \\
& q_{i2} & 1 & \\
& & \ddots & \\
& & & q_{im}
\end{pmatrix}
\quad \text{and} \quad
R_i = \begin{pmatrix}
1 & e_{i1} & & \\
& 1 & e_{i2} & \\
& & \ddots & \ddots \\
& & & 1 & e_{i,m-1}
\end{pmatrix},
\]

respectively, where \( q_{ik} \ (i = 1, \ldots, m_L, \ k = 1, \ldots, m) \) and \( e_{ik} \ (i = 1, \ldots, m_R, \ k = 1, \ldots, m - 1) \) are some positive numbers. In this paper, we consider the problem of computing the eigenvalues of a matrix defined as the product of these bidiagonal factors:

\[
A = L_1 L_2 \cdots L_{m_L} R_1 R_2 \cdots R_{m_R}.
\]

Clearly, \( A \) is a nonsingular band matrix with lower bandwidth \( m_L \) and upper bandwidth \( m_R \). Moreover, \( A \) is totally nonnegative, since it is a product of positive bidiagonal matrices [5]. From this fact, it can also be concluded that all the eigenvalues of \( A \) are simple, real and
positive. When \( m_L = m_R = 1 \), \( A \) is similar to some symmetric positive-definite tridiagonal matrix. In this sense, the matrix in (2) can be regarded as a generalization of the symmetric positive-definite tridiagonal matrix.

Now, considering the eigenvalues of \( A \) with the LR algorithm [6]. In the LR algorithm, we first decompose the input matrix \( A \) into the product of lower and upper triangular matrices as \( A = L(0)R(0) \). Then, we reverse the order of the triangular factors to obtain the next iterate \( A(1) = R(0)L(0) \). This iterate is again decomposed as \( A(1) = L(1)R(1) \) and this process is continued until convergence.

In our case, because the original \( A \) is already defined as a product of multiple lower and upper triangular matrices, we can omit the first decomposition and write \( L(0) \) and \( R(0) \) as

\[
L(0) = L_1 \cdots L_{m_L}, \quad R(0) = R_1 \cdots R_{m_R}.
\]  

Furthermore, the decomposition \( A(1) = L(1)R(1) \) can be performed stepwise as the following example shows:

\[
A(1) = R(0)_{l_2}R(1)_{l_1}L(0)_{l_2}L(1)_{l_1} = R(0)_{l_2}R(1)_{l_1}L(0)_{l_2}L(1)_{l_1} = L(0)_{l_2}R(1)_{l_1}L(0)_{l_2}L(1)_{l_1} = L(0)_{l_2}R(0)_{l_1}L(0)_{l_2}L(1)_{l_1} = L(0)_{l_2}R(0)_{l_1}L(0)_{l_2}L(1)_{l_1} = L(0)_{l_2}R(0)_{l_1}L(0)_{l_2}L(1)_{l_1}.
\]

In summary, when the input matrix \( A \) is defined as a product of bidiagonal factors as in (2), one step of the LR algorithm can be performed as a sequence of LR transformations \( R_iL_j = L_jR_i \) for a pair of bidiagonal matrices without forming \( A(1) \) explicitly. Since each LR transformation can be done without subtractions by using the differential qd transformation [1], one step of the LR algorithm can be carried out without subtractions. We call this the multiple qd algorithm. The outline of this algorithm is shown below.

---

**Algorithm 1: The multiple qd algorithm**

**Initialization:**
- \( q_{j,k}^{(0,0)} = q_{j,k} \) (1 ≤ \( j \) ≤ \( m_L \), 1 ≤ \( k \) ≤ \( m \))
- \( e_{i,k}^{(0)} = e_{i,k} \) (1 ≤ \( i \) ≤ \( m_R \), 1 ≤ \( m ) ≤ m_L - 1)

1: for \( n = 0, 1, \ldots \) do
2: for \( j = 1, \ldots, m_L \) do
3: for \( i = m_P, \ldots, 1 \) do
4: \( d_{j,k} = q_{j,k}^{(n,m_R-1)} \)
5: for \( k = 1, \ldots, m - 1 \) do
6: \( q_{j,k}^{(n,m_R-1+i)} = d_{j,k} + e_{i,k}^{(n,j-1)} \)
7: \( e_{i,k}^{(n,j)} = e_{i,k}^{(n,j-1)}q_{j,k}^{(n,m_R-1+i)} / q_{j,k}^{(n,m_R-1+i)} \)
8: \( d_{j,k+1} = d_{j,k}q_{j,k+1}^{(n,m_R-1+i)} / q_{j,k}^{(n,m_R-1+i)} \)
9: end for
10: \( q_{j,m_R}^{(n,m_R-1+i)} = d_{j,m} \)
11: end for
12: end for
13: \( q_{j,k}^{(n+1,0)} = q_{j,k}^{(n,m_R)} \) (1 ≤ \( j \) ≤ \( m_L \), 1 ≤ \( k \) ≤ \( m )
14: \( e_{i,k}^{(n+1,0)} = e_{i,k}^{(n,m_L)} \) (1 ≤ \( i \) ≤ \( m_R \), 1 ≤ \( m ) ≤ m_L - 1)
15: end for

---

**3. Global convergence properties**

In this section, we establish a theorem that guarantees global convergence of the multiple qd algorithm. This is proved by extending a technique used in the proof of global convergence of the dqds algorithm [7].

**Theorem 1** Let the eigenvalues of a TN matrix defined by (2) be \( \lambda_1 > \lambda_2 > \cdots > \lambda_m > 0 \). When the multiple qd algorithm is applied to this matrix, the following equalities hold:

\[
\lim_{n \to \infty} \prod_{j=1}^{m_L} q_{j,k}^{(n,0)} = \lambda_k \quad (1 \leq k \leq m),
\]

\[
\lim_{n \to \infty} e_{i,k}^{(n,0)} = 0 \quad (1 \leq i \leq m_R, 1 \leq k \leq m - 1),
\]

\[
\lim_{n \to \infty} e_{i,k}^{(n+1,0)} = \frac{\lambda_{k+1}}{\lambda_k} \quad (1 \leq k \leq m - 1).
\]

That is, each subdiagonal element \( e_{i,k}^{(n,0)} \) of \( R(0) \) converges to zero linearly at a rate that depends only on \( k \) and not on \( i \). The product of the \( k \)th diagonal elements of \( L_1^{(n)}, \ldots, L_{m_L}^{(n)} \) converges to the eigenvalue \( \lambda_k \).

**Proof** Algorithm 1 uses the differential qd transformation (lines 4 to 10) to implement the LR transformation

\[
R_i^{(n,j-1)}L_j^{(n,m_R-n-1)} = L_j^{(n,m_R-n-1+i)}R_i^{(n,j)}.
\]

However, for the purpose of proof, it is more convenient to go back to the original equation (7). By comparing the diagonal elements of (7), we have

\[
q_{j,k}^{(n,m_R-n-1+i)} = q_{j,k}^{(n,m_R-i-1)} - e_{i,k}^{(n,j)} + e_{i,k}^{(n,j-1)}
\]

for \( 1 \leq k \leq m \) with the boundary condition \( e_{i,0}^{(n,j)} = 0 \). By summing up (8) for \( 1 \leq i \leq m_R \) and \( 1 \leq l \leq n \) and noting that \( q_{j,k}^{(n+1,0)} > 0 \), we have

\[
0 < \sum_{l=0}^{m_R} \sum_{i=1}^{m_L} e_{i,k}^{(l,j-1)} < q_{j,k}^{(0,0)} + \sum_{l=0}^{m_R} \sum_{i=1}^{m_L} e_{i,k}^{(l,j-1)}.
\]

Using (9) repeatedly, noting that \( e_{i,m}^{(l,j)} = 0 \) and taking the limit of \( n \to \infty \) [7], we have

\[
\sum_{n=0}^{\infty} q_{j,k}^{(n,m_R-n-1+i)} < \infty
\]

for \( 1 \leq k \leq m - 1 \). Hence \( e_{i,j}^{(n)} \to 0 \) and (5) is proved. Thus \( R_1^{(n)}, \ldots, R_{m_R}^{(n)} \) tend to identity matrices. From (8) and (9), we know that each \( q_{i,j}^{(n)} \) converges to a constant that does not depend neither on \( n \) nor \( i \). Consequently, \( A^{(n)} \) converges to a lower triangular matrix whose \( k \)th diagonal element is \( \prod_{l=1}^{m_L} q_{l,k}^{(n,0)} \), which in turn are the eigenvalues of \( A \). Using the same argument as in [7], we know that these eigenvalues appear in the descending order of magnitude, as claimed by (4). Eq. (6) follows by multiplying line 7 of Algorithm 1 side by side from \( j = 1 \) to \( j = m_L \) and using (4).

(QED)

**4. Accuracy of the computed eigenvalues**

Our next objective is to study the accuracy of the multiple qd algorithm in finite precision arithmetic. To this end, we need to combine rounding error analysis...
with perturbation theory, as in the case of the dqds algorithm [1]. First, we quote a lemma concerning rounding
errors resulting from a single LR transformation [1].

Lemma 2 Assume that \( \{q_k\}_{k=1}^m \) and \( \{\hat{e}_k\}_{k=1}^{m-1} \) are computed from \( \{q_k\}_{k=1}^m \) and \( \{e_k\}_{k=1}^{m-1} \) by the LR transformation \( RL = LR \) of differential qd type using finite precision arithmetic. Then there exist \( \{\hat{q}_k\}_{k=1}^m \), \( \{\hat{e}_k\}_{k=1}^{m-1} \), and \( \{\hat{e}_k\}_{k=1}^{m-1} \) such that

- each \( \hat{q}_k \) and \( \hat{e}_k \) differs from \( q_k \) and \( e_k \) by at most 3 ulps (units in the last place) and 1 ulp, respectively,
- each \( \hat{q}_k \) and \( \hat{e}_k \) differs from \( q_k \) and \( e_k \) by at most 2 ulps, respectively, and
- \( \{\hat{q}_k\}_{k=1}^m \) and \( \{\hat{e}_k\}_{k=1}^{m-1} \) are computed from \( \{q_k\}_{k=1}^m \) and \( \{e_k\}_{k=1}^{m-1} \) by an LR transformation in exact arithmetic.

As for the relative perturbation of the eigenvalues, Koepf proves the following lemma [5].

Lemma 3 Let \( A \) be a matrix obtained by multiplying an arbitrary element of one of the bidiagonal factors of the right-hand-side of (2) by \( 1 + e \), where \( |e| \ll 1 \). Denote the eigenvalue of \( A \) corresponding to \( \lambda_k \) by \( \tilde{\lambda}_k \). Then the following bound holds.

\[
|\tilde{\lambda}_k - \lambda_k| \leq \frac{2|e|}{1 - 2|e|} \lambda_k \quad (1 \leq k \leq m). \tag{10}
\]

By combining Lemmas 2 and 3, we can prove the following theorem.

Theorem 4 Assume that the multiple dqd algorithm is executed in finite precision arithmetic. Denote the matrix (defined implicitly as a product of bidiagonal factors) at the \( n \)th step by \( A^{(n)} \) and its eigenvalues by \( \lambda_1^{(n)}, \ldots, \lambda_m^{(n)} \). Then for \( 1 \leq k \leq m \),

\[
|\lambda_k^{(n+1)} - \lambda_k^{(n)}| \leq \frac{16m_1m_2m_3u}{1 - 16m_1m_2m_3u} \lambda_k^{(n)}, \tag{11}
\]

where \( u \) denotes the machine epsilon.

Proof From Lemma 2, we can decompose each LR transformation (7) in one step of the multiple dqd algorithm into three (virtual) steps:

(a) Multiply each diagonal element of matrix \( L_j^{(n,m,n-1)} \)
by \( 1 + e_k \) and each subdiagonal element of \( R_k^{(n,j-1)} \)
by \( 1 + \bar{e}_k \), where \( |e_k| \leq 3u \) and \( |\bar{e}_k| \leq u \), to obtain \( \bar{L}_j \) and \( \bar{R}_k \).

(b) Perform exact LR transformation to get \( \bar{L}_j \) and \( \bar{R}_k \) from \( \bar{L}_j \) and \( \bar{R}_k \).

(c) Multiply each diagonal element of \( \bar{L}_j \) by \( 1 + \bar{e}_k \)
and each subdiagonal element of \( \bar{R}_k \) by \( 1 + \bar{e}_k \),
where \( |\bar{e}_k| \leq 2u \) and \( |\bar{e}_k| \leq 2u \), to obtain \( L_j^{(n,m,n-1)} \) and \( R_k^{(n,j)} \).

We also recall a lemma from [8] that for positive integers \( m_1, m_2 \) and a sufficiently small positive number \( \delta \), if \( |\delta| \leq m_1\delta/(1-m_1\delta) \) and \( |\delta| \leq m_2\delta/(1-m_2\delta) \), then

\[
|(1 + \delta)(1 + \delta) - 1| \leq \frac{(m_1 + m_2)\delta}{1 - (m_1 + m_2)\delta}. \tag{12}
\]

Using Lemma 3 and (12) repeatedly, we know that the eigenvalues before and after step (a), which we denote by \( \lambda_k \) and \( \tilde{\lambda}_k \), respectively, are related as follows:

\[
|\tilde{\lambda}_k - \lambda_k| \leq \frac{8m_1u}{1 - 8m_1u} \lambda_k. \tag{13}
\]

Clearly, step (b) does not change the eigenvalues. The eigenvalues after step (c), which we denote by \( \bar{\lambda}_k \), is related with \( \tilde{\lambda}_k \) as follows:

\[
|\bar{\lambda}_k - \tilde{\lambda}_k| \leq \frac{8m_2u}{1 - 8m_2u} \lambda_k. \tag{14}
\]

Combining (13) and (14) using (12) again and noting that \( m_Lm_R \) LR transformations are needed to complete one multiple dqd step, we get at (11).

(QED)

Eq. (11) means that only small relative error is introduced into each eigenvalue by one multiple dqd step. Consequently, by iterating the step until \( A^{(n)} \) becomes sufficiently close to a lower triangular matrix, all eigenvalues can be computed to high relative accuracy.

5. Numerical results

To confirm our analysis in the preceding sections, we performed preliminary numerical experiments. We set \( m = 10 \), \( m_L = 1 \) and \( m_R = 3 \) and set the values of \( e_{ik} \) and \( q_{ik} \) using random numbers in \([0, 1]\). The computation was done using Fortran in double-precision arithmetic.

To check the accuracy of the computed eigenvalues, we also used Mathematica with 100 decimal digits, formed \( A = L_1R_1L_2R_2L_3 \) explicitly, and computed its eigenvalues.

Fig. 1 shows \( e_{i,k}^{(n,0)} \) as a function of \( n \). Clearly, all of them converge to zero linearly. Though there are 27 of them, only 9 lines can be seen in Fig. 1. This is because the convergence rate of \( e_{i,k}^{(n,0)} \) depends only on \( k \) and therefore the lines for \( e_{1,1}^{(n,0)}, e_{2,1}^{(n,0)} \) and \( e_{3,1}^{(n,0)} \), for example, overlap. This is in accordance with Theorem 1. The eigenvalues computed by the multiple dqd algorithm, as well as those computed by Mathematica, are shown in Table 1. It is clear that all the eigenvalues are computed to high relative accuracy. This confirms the analysis in the previous section.
6. Relationship with the dhLV algorithm

Let’s consider the case of $m_L = 1$ and $m_R = M$, where $M$ is some positive integer. In this case it is well known [9] that the eigenvalue problem of $A$ is equivalent to the eigenvalue problem of an $(M+1)m \times (M+1)m$ matrix $C$ defined by

$$C = \begin{pmatrix} R_M & L_1 \\ \vdots & \ddots \\ R_2 & \end{pmatrix}.$$

(15)

More precisely, if $\lambda_k$ is an eigenvalue of $A$, then $(\lambda_k)^{1/(M+1)}\exp(2\pi i p/(M+1))$ $(0 \leq p \leq M)$ are eigenvalues of $C$, and vice versa. Furthermore, express the row index $i$ of $C$ as $i = (l-1)m + b$ $(1 \leq l \leq M+1$, $1 \leq b \leq m)$ and permute the $i$-th row to the $i'$-th row, where $i' = (b-1)(M+1) + l$. Apply the same permutation also to the columns. This amounts to replacing $C$ with $F = PCP^T$, where $P$ is a permutation matrix, and is called shuffling [9]. It is easy to see that $F$ is a matrix with only two nonzero diagonals as follows:

$$F_{i+1,i} = 1 \quad (1 \leq i \leq (M+1)m - 1),$$

$$F_{i,i+M} = F_{i-(M+1)+1,i-M} = \begin{cases} q_{l,b} & (l = 1, 1 \leq b \leq m), \\ e_{M+2-l,b} & (2 \leq l \leq M+1, 1 \leq b \leq m-1). \end{cases}$$

By rewriting $F_{i,i+M}$ as $U_l$ $(1 \leq i \leq (m-1)m + m)$, it can be seen that $F$ is exactly the type of matrix for which the dhLV algorithm [4] has been designed. Thus we can say that the class of matrices whose eigenvalues can be computed by the dhLV algorithm is a subset of the class of matrices whose eigenvalues can be computed accurately by the multiple dqd algorithm.

7. Related work

It is widely recognized that by representing a TN matrix as a product of positive bidiagonal factors, various linear algebra operations can be performed without subtractions [10] [11]. Using this fact, several highly accurate algorithms have been proposed for linear simultaneous equations [11], eigenvalue problems [5] and singular value problems [12] with TN coefficient matrices.

Among them, Koev's algorithm [5] can compute all the eigenvalues of a general TN matrix to high relative accuracy. It first reduces a TN matrix in factored form to a product of a lower bidiagonal matrix an upper bidiagonal matrix using subtraction-free operations and then compute the eigenvalues of the resulting matrix with the dqds algorithm. In this approach, the reduction phase requires $O(m^3)$ flops. In contrast, the multiple dqd algorithm applies the LR transformation directly to a matrix represented by (2). The computational work of one LR transformation is $O(mn_1m_2)$. Hence the latter approach may be advantageous when $m_L, m_R \ll m$ and only a small number of eigenvalues of the smallest magnitude are required.

8. Conclusion

In this paper, we studied convergence properties and numerical properties of the differential qd algorithm for totally nonnegative band matrices. Our analysis shows that the algorithm is globally convergent and can compute all eigenvalues to high relative accuracy. These properties were confirmed by numerical experiments. Our future work includes introducing origin shifts into this algorithm to speed up the convergence. It is also the subject of our future research to further investigate the relationship between the multiple dqd algorithm and the dhLV algorithm.

Acknowledgements

The authors would like to thank Prof. Masashi Iwasaki, Prof. Satoshi Tsujimoto, Prof. Yoshimasa Nakamura, Ms. Akiko Fukuda and Mr. Kensuke Aishima for valuable discussion. This work is partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-aid for Scientific Research.

References


