Annali di Matematica (2013) 192:423–445
DOI 10.1007/s10231-011-0231-0

Integrable discrete hungry systems and their related matrix eigenvalues

Akiko Fukuda · Emiko Ishiwata · Yusaku Yamamoto · Masashi Iwasaki · Yoshimasa Nakamura

Received: 22 February 2011 / Accepted: 1 November 2011 / Published online: 13 November 2011

Abstract Recently, some of the authors designed an algorithm, named the dhLV algorithm, for computing complex eigenvalues of a certain class of band matrix. The recursion formula of the dhLV algorithm is based on the discrete hungry Lotka–Volterra (dhLV) system, which is an integrable system. One of the authors has proposed an algorithm, named the multiple dqd algorithm, for computing eigenvalues of a totally nonnegative (TN) band matrix. In this paper, by introducing a theorem on matrix eigenvalues, we first show that the eigenvalues of a TN matrix are also computable by the dhLV algorithm. We next clarify the asymptotic behavior of the discrete hungry Toda (dhToda) equation, which is also an integrable system, and show that a similarity transformation for a TN matrix is given through the dhToda equation. Then, by combining these properties of the dhToda equation, we design a new algorithm, named the dhToda algorithm, for computing eigenvalues of a TN matrix. We also describe the close relationship among the above three algorithms and give numerical examples.

This was partially supported by Grants-in-Aid for Young Scientists (B) No. 20740064, Scientific Research (A) No. 20246027, and Scientific Research (C) No. 20540137 from the Japan Society for the Promotion of Science.

A. Fukuda · E. Ishiwata
Department of Mathematical Information Science, Tokyo University of Science,
1-3 Kagurazaka, Shinjuku-ku, Tokyo 162-8601, Japan
e-mail: afukuda@rs.tus.ac.jp

Y. Yamamoto
Graduate School of System Informatics, Kobe University, Kobe, Japan

M. Iwasaki (✉)
Department of Informatics and Environmental Science, Kyoto Prefectural University,
1-5 Nakaragi-cho, Shimogamo, Sakyo-ku, Kyoto 606-8522, Japan
e-mail: imasa@kpu.ac.jp

Y. Nakamura
Graduate School of Informatics, Kyoto University, Kyoto, Japan

Keywords Discrete hungry Lotka–Volterra system · Discrete hungry Toda equation · Matrix eigenvalue · Similarity transformation

Mathematics Subject Classification (2000) 37K10 · 37K40 · 65F15

1 Introduction

Several integrable systems have profound relationships with various algorithms. In [29], Symes finds that one step of the QR algorithm, an algorithm for computing matrix eigenvalues, corresponds to the time evolution of the continuous-time Toda equation. Hirota's discretization technique [9] leads to the discrete-time version of the Toda equation,

$$\begin{cases} q_k^{(n+1)} = q_k^{(n)} - e_{k-1}^{(n+1)} + e_k^{(n)}, \quad k = 1, 2, \dots, m, \\ e_k^{(n+1)} = e_k^{(n)} \dfrac{q_{k+1}^{(n)}}{q_k^{(n+1)}}, \quad k = 1, 2, \dots, m-1, \\ e_0^{(n)} \equiv 0, \quad e_m^{(n)} \equiv 0, \quad n = 0, 1, \dots, \end{cases} \tag{1}$$

where $q_k^{(n)}, e_k^{(n)}$ denote the values of q_k, e_k at the discrete time n, respectively. The discrete Toda equation (1) is exactly the recursion formula of Rutishauser's quotient difference (qd) algorithm [28] for the eigenvalues of a symmetric tridiagonal matrix. It is to be remarked that in [26], Rutishauser himself derives the continuous-time Toda equation from the recurrence formula of the qd algorithm to investigate its asymptotic behavior. The qd algorithm can be viewed as the special case of the general LR algorithm where the given matrix is tridiagonal [27]. The qd algorithm actually gives the eigenvalues of the tridiagonal matrix in the generic case. Therefore, the qd algorithm and the LR algorithm were the predecessors of the QR algorithm. Rutishauser's original qd algorithm is now called the progressive qd (pqd). Several variants of the qd algorithm such as dqd and oqd are discussed in [22] as methods for finding eigenvalues with high relative accuracy. The potential use of these variants for the accurate computation of the eigenvalues of totally nonnegative (TN) band matrices is evaluated by Koev in [16]. Other applications of the qd algorithm (1) have also been observed in various fields; these applications include the BCH-Goppa decoding [20] and the Laplace transformation [19]. In [18], it is shown that the qd algorithm (1) is related to the ϵ-algorithm for accelerating the convergence rate of a sequence.

Another integrable discrete system yielding numerical algorithms related to matrix eigenvalues and singular values is the discrete Lotka–Volterra (dLV) system, which is an integrable discrete-time system. In [12, 13], Iwasaki and Nakamura design an algorithm for computing singular values based on the dLV system, which is as follows [10]:

$$\begin{cases} u_k^{(n+1)}(1 + \delta^{(n+1)} u_{k-1}^{(n+1)}) = u_k^{(n)}(1 + \delta^{(n)} u_{k+1}^{(n)}), \quad k = 1, 2, \dots, 2m-1, \\ u_0^{(n)} \equiv 0, \quad u_{2m}^{(n)} \equiv 0, \quad n = 0, 1, \dots, \end{cases} \tag{2}$$

which is a time discretization of the continuous-time Lotka–Volterra (LV) system, where $\delta^{(n)}$ denotes the nth discrete step size and $u_k^{(n)}$ denotes the number of the kth species at the discrete time $\sum_{j=0}^{n-1} \delta^{(j)}$. The LV system was originally used to describe the struggle for survival of $2m-1$ species among which the kth species preys upon the $(k+1)$th species and is in turn preyed upon by the $(k-1)$th species [33]. The dLV system (2) also arises from the discrete Toda equation (1) through the Miura transformation:

$$\begin{cases} q_k^{(n)} = \dfrac{1}{\delta^{(n)}}\left(1+\delta^{(n)}u_{2k-2}^{(n)}\right)\left(1+\delta^{(n)}u_{2k-1}^{(n)}\right), & k=1,2,\ldots,m, \\ e_k^{(n)} = \delta^{(n)}u_{2k-1}^{(n)}u_{2k}^{(n)}, \quad k=1,2,\ldots,m-1. \end{cases} \tag{3}$$

A remarkable property of the dLV system (2) is that, for a suitable initial $u_k^{(0)}$, the dLV variable $u_{2k-1}^{(n)}$ converges to the square of a singular value of some bidiagonal matrix as $n \to \infty$. This asymptotic convergence immediately gives an algorithm, named the dLV algorithm, for computing singular values. In [4], the dLV algorithm is reviewed by Chu. In order to accelerate the convergence rate, Iwasaki and Nakamura in [14] introduce shift of origin into the dLV algorithm. The shifted version of the dLV algorithm is called the modified dLV with shift (mdLVs) algorithm.

The LV system is naturally extended to the continuous-time hungry LV (hLV) system by considering the case where the kth species preys not only on the $(k+1)$th species but also on the $(k+2)$th, $(k+3)$th, $\ldots$, $(k+M)$th ones [2,11]. The hLV system with $M=1$ coincides with the LV system. A time discretization [21] of the hLV (dhLV) system for $n=0,1,\ldots$ is given as

$$\begin{cases} u_k^{(n+1)}\displaystyle\prod_{j=1}^{M}\left(1+\delta^{(n+1)}u_{k-j}^{(n+1)}\right) = u_k^{(n)}\prod_{j=1}^{M}\left(1+\delta^{(n)}u_{k+j}^{(n)}\right), \\ \quad k=1,2,\ldots,M_m, \\ u_{1-M}^{(n)} \equiv 0,\ldots,u_0^{(n)} \equiv 0, \quad u_{M_m+1}^{(n)} \equiv 0,\ldots,u_{M_m+M}^{(n)} \equiv 0, \end{cases} \tag{4}$$

where $M_k := (M+1)k - M$ and the notation k, $\delta^{(n)}$, and $u_k^{(n)}$ in (4) is the same as that in the dLV system (2). From the dhLV system (4), Fukuda et al. in [5] derive an algorithm, named the dhLV algorithm, for computing complex eigenvalues of a band matrix. In [32], Yamamoto and Fukaya propose an algorithm, named the multiple dqd algorithm, for computing the eigenvalues of TN band matrices [1], for which all the minors are nonnegative. TN matrices appear in many branches of mathematics involving applications, including combinatorics, probability, stochastic processes, and inverse problems [3,6,7,15]. It is also shown in [32] that the multiple dqd variables correspond to those of the dhLV.

The box and ball system (BBS) was introduced by Takahashi and Matsukidaira in [24] by considering the viewpoint of integrable systems. The BBS represents the movement of a finite number of balls in an array of boxes. The rule of the BBS is that the leftmost ball moves to the nearest empty box to the right. The dynamics of the BBS is related to the discrete Toda equation (1). Tokihiro et al. in [25] propose a different BBS, named the numbered BBS, in which balls are numbered from 1 to M so that they can be distinguished by their index. The numbered BBS requires that first the leftmost ball with index 1 is moved to the nearest empty box to the right, and then, this procedure is repeated for balls with index $2, 3, \ldots, M$. The numbered BBS is also associated with the integrable discrete hungry Toda (dhToda) equation,

$$\begin{cases} Q_k^{(n+M)} = Q_k^{(n)} + E_k^{(n)} - E_{k-1}^{(n+1)}, \quad k=1,2,\ldots,m, \\ E_k^{(n+1)} = \dfrac{Q_{k+1}^{(n)}E_k^{(n)}}{Q_k^{(n+M)}}, \quad k=1,2,\ldots,m-1, \\ E_0^{(n)} := 0, \quad E_m^{(n)} := 0, \end{cases} \tag{5}$$

which is regarded as an extension of the discrete Toda equation (1). Tokihiro, Nagai, and Satsuma predicted that the dhToda equation (5) has some interesting relationship with matrix eigenvalues. However, to the best of our knowledge, the dhToda equation (5) has not been shown to be related to any particular type of matrix, much less been used to achieve a new algorithm for computing matrix eigenvalues.

The main purpose of the present paper is twofold. First, its purpose is to clarify that the eigenvalues of a banded Hessenberg TN matrix are computable by the dhLV algorithm. Second, a new algorithm for these types of matrix eigenvalue problems is designed using a matrix representation and an asymptotic analysis of the dhToda equation (5). The relationship among the three algorithms—the dhLV algorithm, the algorithm based on the dhToda equation (5), and the multiple dqd algorithm—is also shown, and some numerical examples are given.

There are several algorithms that can be used to compute the eigenvalues of a banded Hessenberg TN matrix. For example, the BR algorithm by Geist et al. [8] is an efficient iterative algorithm for computing the eigenvalues of a banded Hessenberg matrix. It is a variant of the LR iteration with pivoting, but is designed to roughly preserve the banded form to minimize the computational cost. However, as with many other algorithms, it can compute only large eigenvalues to high relative accuracy. In contrast, as will be shown by numerical experiments, our algorithm can compute the smallest eigenvalues to high relative accuracy by exploiting the TN structure. In addition, the TN structure enables us to prove the global convergence of our algorithm. Koev also proposes an algorithm for computing the eigenvalues of a TN matrix to high relative accuracy [16]. In his algorithm, the input TN matrix is first reduced to a tridiagonal matrix by a sequence of dqd transformations, and the eigenvalues of the resulting tridiagonal matrix are computed by the dqds (dqd with shift) algorithm. In contrast, our algorithm operates directly on the input TN matrix and transforms it to an upper triangular matrix without destroying the banded Hessenberg structure. We expect that the latter approach is advantageous when the bandwidth is small.

This paper is organized as follows. In Sect. 2, we briefly explain how to derive the dhLV algorithm proposed in [5] from the dhLV system (4). We expand the class of matrices to which the dhLV algorithm is applicable by considering a similarity transformation. And then, with the help of a theorem in [31] on matrix eigenvalues, we show as a special case that eigenvalues of a TN matrix are computable by the dhLV algorithm. We also clarify the relationship of the dhLV algorithm with the multiple dqd algorithm. In Sect. 3, we investigate a matrix representation, named the Lax form, for the dhToda equation (5) and the asymptotic behavior of the dhToda variables as time variable $n \to \infty$. Based on the dhToda equation (5), we design a new algorithm for eigenvalues of a TN matrix. In addition, we describe the relationship of the dhLV algorithm, the algorithm designed in Sect. 3, and the multiple dqd algorithm. In Sect. 4, we confirm the theoretical results in Sects. 2 and 3 through a number of numerical examples. Finally, we give concluding remarks in Sect. 5.

2 dhLV algorithm for band matrices

A matrix A is said to be TN if every minor of A is nonnegative [1]. The main purpose of this section is to show that the eigenvalues of certain TN matrices are computable by the dhLV algorithm proposed in [5]. The dhLV algorithm and its basic properties are briefly reviewed in Sect. 2.1. In Sect. 2.2, we describe a different aspect of the dhLV algorithm. Not only the band matrices $\mathscr{L}^{(n)} + dI$ appearing in Sect. 2.1 but also the TN matrices in Sect. 2.2 are shown to be the targets for the dhLV algorithm. A theorem on matrix eigenvalues [31] plays a key role in this proof. In Sect. 2.3, we explain the multiple dqd algorithm designed in [32] for computing the eigenvalues of TN matrices. We also clarify the relationship between the dhLV algorithm and the multiple dqd algorithm.

Here, to assist the reader's understanding, we describe the target matrices of the dhLV algorithm appearing in the following subsections for the simple case where $M = 2$ and $m = 3$. The dhLV algorithm was originally shown in [5] to be applicable for computing the eigenvalues of the band matrix

$$\mathscr{L} = \begin{pmatrix} 0 & 0 & l_1 & & & & & & \\ 1 & 0 & 0 & l_2 & & & & & \\ & 1 & 0 & 0 & l_3 & & & & \\ & & 1 & 0 & 0 & l_4 & & & \\ & & & 1 & 0 & 0 & l_5 & & \\ & & & & 1 & 0 & 0 & l_6 & \\ & & & & & 1 & 0 & 0 & l_7 \\ & & & & & & 1 & 0 & 0 \\ & & & & & & & 1 & 0 \end{pmatrix}, \tag{6}$$

where $l_1 > 0, l_2 > 0, \ldots, l_7 > 0$. The similarity transformation by $D = \mathrm{diag}(1, \alpha_1, \alpha_1\alpha_2, \ldots, (\alpha_1\alpha_2 \cdots \alpha_8))$ leads to $\hat{\mathscr{L}} := D\mathscr{L}D^{-1}$, whose $(i+1, i)$ and $(i, i+2)$ entries are α_i and $l_i/(\alpha_i\alpha_{i+1} \cdots \alpha_8)$, respectively. It is easily expected that we can get the eigenvalues of the extended $\hat{\mathscr{L}}$ through computing the eigenvalues of $\mathscr{L}$ by the dhLV algorithm. Moreover, by a suitable permutation matrix P, we can transform $\mathscr{L}$ into

$$\mathscr{B} := P\mathscr{L}P^{-1} = \begin{pmatrix} & & \mathscr{L}_1 \\ \mathscr{R}_2 & & \\ & \mathscr{R}_1 & \end{pmatrix}, \tag{7}$$

where

$$\mathscr{L}_1 := \begin{pmatrix} l_1 & & \\ 1 & l_4 & \\ & 1 & l_7 \end{pmatrix}, \quad \mathscr{R}_1 := \begin{pmatrix} 1 & l_3 & \\ & 1 & l_6 \\ & & 1 \end{pmatrix}, \quad \mathscr{R}_2 := \begin{pmatrix} 1 & l_2 & \\ & 1 & l_5 \\ & & 1 \end{pmatrix}.$$

According to [31], the eigenvalues of $\mathscr{B}$ are clearly related to those of $\mathscr{A} := \mathscr{L}_1\mathscr{R}_1\mathscr{R}_2$. Since $\mathscr{L}_1, \mathscr{R}_1$, and $\mathscr{R}_2$ are TN matrices, $\mathscr{A}$ is also. The above discussion implies that the eigenvalues of the TN matrix are computable by using the dhLV algorithm. Of course, the dhLV algorithm is related to the multiple dqd algorithm in terms of computing the eigenvalues of a TN matrix.

2.1 dhLV system and band matrix eigenvalues

We first survey some important properties of the dhLV system, which are the basis of the dhLV algorithm. One of the essential properties of integrable systems is a matrix representation called the Lax form. Although the Lax form is an idea that arises from the study of integrable systems, it is useful to reconsider it from the viewpoint of matrix analysis. The QR and qd algorithms are actually related to integrable systems, specifically, the Toda equation and the discrete Toda equation (1), respectively, through the Lax form. The dLV algorithm was designed with the help of a Lax form for the dLV system [12,13].

A Lax form for the dhLV system (4) is presented in [30] as follows.

$$\mathscr{R}^{(n)}\mathscr{L}^{(n+1)} = \mathscr{L}^{(n)}\mathscr{R}^{(n)}, \tag{8}$$

$$\mathscr{L}^{(n)} := \begin{pmatrix} \overbrace{0 & \dots & 0}^{M} & U_1^{(n)} & & & & \\ 1 & 0 & \dots & 0 & U_2^{(n)} & & & \\ & 1 & \ddots & & \ddots & \ddots & & \\ & & \ddots & \ddots & & \ddots & U_{M_m}^{(n)} \\ & & & \ddots & \ddots & & 0 \\ & & & & \ddots & \ddots & \vdots \\ & & & & & 1 & 0 \end{pmatrix}, \tag{9}$$

$$\mathscr{R}^{(n)} := \begin{pmatrix} V_1^{(n)} & & & & & & \\ 0 & V_2^{(n)} & & & & & \\ \vdots & 0 & \ddots & & & & \\ 0 & \vdots & \ddots & \ddots & & & \\ \delta^{(n)} & 0 & & \ddots & \ddots & & \\ & \ddots & \ddots & & \ddots & \ddots & \\ & & \delta^{(n)} & \underbrace{0 & \dots & 0}_{M} & V_{M_m+M}^{(n)} \end{pmatrix}, \tag{10}$$

$$U_k^{(n)} := u_k^{(n)} \prod_{j=1}^{M} \left(1 + \delta^{(n)} u_{k-j}^{(n)}\right), \tag{11}$$

$$V_k^{(n)} := \prod_{j=0}^{M} \left(1 + \delta^{(n)} u_{k-j}^{(n)}\right). \tag{12}$$

The equality in (8) is equivalent to the dhLV system (4).

Assume that

$$0 < u_k^{(0)} < K_0, \quad k = 1, 2, \dots, M_m, \tag{13}$$

where K_0 is some positive constant. Then, it is obvious from (12) that $V_k^{(n)} \geq 1$ in $\mathscr{R}^{(n)}$ for $k = 1, 2, \dots, M_m + M$. Hence, there exists an inverse matrix of $\mathscr{R}^{(n)}$, and so (8) can be transformed to give

$$\mathscr{L}^{(n+1)} = (\mathscr{R}^{(n)})^{-1}\mathscr{L}^{(n)}\mathscr{R}^{(n)}. \tag{14}$$

This is a similarity transformation from $\mathscr{L}^{(n)}$ to $\mathscr{L}^{(n+1)}$. Namely, the eigenvalues of $\mathscr{L}^{(n)}$ are invariant under the time evolution from n to $n + 1$. Therefore, the matrices $\mathscr{L}^{(0)}, \mathscr{L}^{(1)}, \mathscr{L}^{(2)}, \dots$ are similar to each other. For the identity matrix I and an arbitrary constant d, the matrices $\mathscr{L}^{(0)} + dI, \mathscr{L}^{(1)} + dI, \mathscr{L}^{(2)} + dI, \dots$ are also similar.

There exist other important invariants under the time evolution of dhLV systems. For example, the sums and the products of variables $U_k^{(n)}$,

$$\sum_{k=1}^{M_m} U_k^{(n)} = \sum_{k=1}^{M_m} U_k^{(n+1)}, \tag{15}$$

$$\prod_{k=1}^{m} U_{M_k}^{(n)} = \prod_{k=1}^{m} U_{M_k}^{(n+1)}, \tag{16}$$

are invariant. From the assumption (13), it follows that $0 < \sum_{k=1}^{M_m} U_k^{(0)} < K_1$ and $0 < \prod_{k=1}^{m} U_{M_k}^{(0)} < K_2$, where K_1 and K_2 are positive constants. Making use of (15) and (16), we derive $0 < u_k^{(n)} < K$ for a positive constant K. The asymptotic behavior of $u_k^{(n)}$ given (13) is

$$\lim_{n\to\infty} u_{M_k}^{(n)} = c_k, \quad k = 1, 2, \dots, m, \tag{17}$$

$$\lim_{n\to\infty} u_{M_k+p}^{(n)} = 0, \quad k = 1, 2, \dots, m-1, \quad p = 1, 2, \dots, M, \tag{18}$$

where $c_1, c_2, \dots, c_m$ are positive constants such that

$$c_1 \geq c_2 \geq \cdots \geq c_m. \tag{19}$$

See [5] for the proof of (15), (16) and (17), (18) with (19).

Next, we explain how to apply the dhLV system (4) to matrix eigenvalue computation. Obviously, from (11), (12) and (17), (18), the limits of $U_k^{(n)}$ and $V_k^{(n)}$ also exist as $n \to \infty$. The limit of the matrix $\mathscr{L}^{(n)} + dI$ is given by

$$\mathscr{L}(d) := \lim_{n\to\infty} (\mathscr{L}^{(n)} + dI) = \begin{pmatrix} \mathscr{L}_1(d) & & & \\ \mathscr{E}_M & \mathscr{L}_2(d) & & \\ & \ddots & \ddots & \\ & & \mathscr{E}_M & \mathscr{L}_m(d) \end{pmatrix}, \tag{20}$$

where $\mathscr{L}_k(d)$ and $\mathscr{E}_M$ are $(M+1) \times (M+1)$ block matrices

$$\mathscr{L}_k(d) := \begin{pmatrix} d & & & c_k \\ 1 & d & & \\ & \ddots & \ddots & \\ & & 1 & d \end{pmatrix}, \quad \mathscr{E}_M := \begin{pmatrix} 0 & \cdots & 0 & 1 \\ & \ddots & & 0 \\ & & \ddots & \vdots \\ & & & 0 \end{pmatrix}. \tag{21}$$

It is of significance that, by cofactor expansion,

$$\det(\lambda I - \mathscr{L}(d)) = \prod_{k=1}^{m} \det(\lambda I - \mathscr{L}_k(d)),$$

$$\det(\lambda I - \mathscr{L}_k(d)) = (\lambda - d)^{M+1} - c_k.$$

Thus, the characteristic polynomial of $\mathscr{L}(d)$ is given as

$$\det(\lambda I - \mathscr{L}(d)) = \prod_{k=1}^{m} \left[(\lambda - d)^{M+1} - c_k \right].$$

Consequently, we obtain the eigenvalues $\lambda_{k,\ell}$ of $\mathscr{L}^{(0)} + dI$ as follows.

$$\lambda_{k,\ell} = c_k^{\frac{1}{M+1}} \left[\exp\left(\frac{2\pi i}{M+1} \right) \right]^{\ell} + d, \quad k = 1, 2, \ldots, m, \quad \ell = 0, 1, \ldots, M. \tag{22}$$

Namely, the eigenvalues of $\mathscr{L}^{(0)} + dI$ are given by using the $(M+1)$th root of c_k derived from the time evolution of the dhLV system (4). Since, for a sufficiently large N, $u_{M_k}^{(N)}$ is a good approximation to c_k, the $(M+1)$th root of $u_{M_k}^{(n)}$ can be used to approximate the eigenvalues of $\mathscr{L}^{(0)} + dI$.

The above discussion is a brief review of [5]. Next, we will expand the applicable range of the dhLV algorithm. Let us introduce a diagonal matrix

$$D := \mathrm{diag}\left(1, \alpha_1, \alpha_1\alpha_2, \ldots, (\alpha_1\alpha_2 \cdots \alpha_{M_m+M-1})\right), \tag{23}$$

with arbitrary positive constants $\alpha_1, \alpha_2, \ldots, \alpha_{M_m+M-1}$. Then, the similarity transformation by D yields

$$\hat{\mathscr{L}}^{(n)} + dI := D(\mathscr{L}^{(n)} + dI)D^{-1} \tag{24}$$

$$= \begin{pmatrix} d & & \hat{U}_1^{(n)} & & & \\ \alpha_1 & d & & \ddots & & \\ & \alpha_2 & \ddots & & \hat{U}_{M_m}^{(n)} & \\ & & \ddots & \ddots & & \\ & & & \ddots & \ddots & \\ & & & & \alpha_{M_m+M-1} & d \end{pmatrix}, \tag{25}$$

where $\hat{U}_k^{(n)} = U_k^{(n)}/(\alpha_{k+1}\alpha_{k+2}\cdots\alpha_{k+M-1})$. Obviously, the eigenvalues of $\hat{\mathscr{L}}^{(n)} + dI$ coincide with those of $\mathscr{L}^{(n)} + dI$. Hence, the eigenvalues of $\hat{\mathscr{L}}^{(0)} + dI$ are given as (22) if the initial $U_1^{(0)}, U_2^{(0)}, \ldots, U_{M_m}^{(0)}$, as a function of $\hat{U}_1^{(0)}, \hat{U}_2^{(0)}, \ldots, \hat{U}_{M_m}^{(0)}$ and $\alpha_1, \alpha_2, \ldots, \alpha_{M_m+M-1}$, are set such that

$$U_k^{(0)} = \hat{U}_k^{(0)}(\alpha_k\alpha_{k+1}\cdots\alpha_{k+M-1}). \tag{26}$$

The dhLV algorithm for computing the eigenvalues of $\hat{\mathscr{L}}^{(0)} + dI$ is as follows.

dhLV algorithm for $\hat{\mathscr{L}}^{(0)} + dI$

for $k := 1, 2, \ldots, M_m$ **do**
$\quad U_k^{(0)} = \hat{U}_k^{(0)} \prod_{j=0}^{M-1} \alpha_{k+j}$
end for
for $k := 1, 2, \ldots, M_m$ **do**
$\quad u_k^{(0)} = U_k^{(0)} / \prod_{j=1}^{M} \left(1 + \delta^{(0)} u_{k-j}^{(0)}\right)$
end for
for $n := 1, 2, \ldots, n_{\max}$ **do**
$\quad$ **for** $k := 1, 2, \ldots, M_m$ **do**
$\quad\quad u_k^{(n+1)} := u_k^{(n)} \left[\prod_{j=1}^{M} \left(1 + \delta^{(n)} u_{k+j}^{(n)}\right) / \prod_{j=1}^{M} \left(1 + \delta^{(n+1)} u_{k-j}^{(n+1)}\right) \right]$
$\quad$ **end for**
end for
for $k := 1, 2, \ldots, m$ **do**

13: **for** $\ell := 1, 2, \ldots, M+1$ **do**
14: $\quad \lambda_{k,\ell} := \sqrt[M+1]{u_{M_k}^{(n)}}\{\cos[2\ell\pi/(M+1)] + i\sin[2\ell\pi/(M+1)]\} + d$
15: **end for**
16: **end for**

Lines 7 through 11 of the dhLV algorithm are repeated until $\max_{k \neq M_1, M_2, \ldots, M_m} u_k \leq eps$ or $n > n_{\max}$ is satisfied for a sufficiently small $eps > 0$.

2.2 dhLV algorithm for TN matrix

We show here that the eigenvalues of a TN matrix are computable by the dhLV algorithm. It has already been reported in [5] that, as is described in Sect. 2.1, the band matrix $\mathscr{L}^{(n)} + dI$ is a target for the dhLV algorithm. Of course, $\mathscr{L}^{(n)} + dI$ is not TN for general d. For simplicity, we hereinafter discuss the case where $d = 0$.

Let us introduce a technique for matrix permutation, which is a special case of [31]. Let P be the permutation matrix such that $P\mathscr{L}^{(n)}$ is the matrix given by interchanging the $[(k-1)(M+1)+j]$th and $[(j-1)m+k]$th rows of $\mathscr{L}^{(n)}$ for $j = 1, 2, \ldots, M+1$ and $k = 1, 2, \ldots, m$. Namely, P is the matrix whose $((j-1)m+k, (k-1)(M+1)+j)$ entries are 1 and the others are 0. Since $\mathscr{L}^{(n)}P^{-1}$ is the matrix given by interchanging the $[(k-1)(M+1)+j]$th and $[(j-1)m+k]$th columns of $\mathscr{L}^{(n)}$, it follows that

$$\begin{aligned} \mathscr{B}^{(n)} &:= P\mathscr{L}^{(n)}P^{-1} \\ &= \begin{pmatrix} & & & & \mathscr{L}_1^{(n)} \\ \mathscr{R}_M^{(n)} & & & & \\ & \ddots & & & \\ & & \mathscr{R}_2^{(n)} & & \\ & & & \mathscr{R}_1^{(n)} & \end{pmatrix}, \end{aligned} \tag{27}$$

where

$$\mathscr{L}_1^{(n)} = \begin{pmatrix} U_{M_1}^{(n)} & & & \\ 1 & U_{M_2}^{(n)} & & \\ & \ddots & \ddots & \\ & & 1 & U_{M_m}^{(n)} \end{pmatrix}, \tag{28}$$

$$\mathscr{R}_j^{(n)} = \begin{pmatrix} 1 & U_{M_2-j}^{(n)} & & \\ & 1 & \ddots & \\ & & \ddots & U_{M_m-j}^{(n)} \\ & & & 1 \end{pmatrix}. \tag{29}$$

To consider matrix eigenvalues, we use the following theorem.

Theorem 1 **(Watkins [31])** *The nonzero complex number* λ *is an eigenvalue of* X *if and only if its kth roots* $\lambda^{1/k}, \lambda^{1/k}\omega, \lambda^{1/k}\omega^2, \ldots, \lambda^{1/k}\omega^{k-1}$ *are all eigenvalues of* $\hat{X}$, *where* $\omega = \exp(2\pi i/k)$ *and*

$$X = X_k X_{k-1} \cdots X_1 \in \mathbf{C}^{m\times m}, \quad X_j \in \mathbf{C}^{m\times m}, \quad j = 1, 2, \ldots, k, \tag{30}$$

$$\hat{X} = \begin{pmatrix} & & & & X_k \\ X_1 & & & & \\ & X_2 & & & \\ & & \ddots & & \\ & & & X_{k-1} & \end{pmatrix}. \tag{31}$$

Let $k = M + 1$ in Theorem 1. Then, $\mathscr{B}^{(n)}$ in (27) has the same form as $\hat{X}$ in (31). The blocks $\mathscr{R}_M^{(n)}, \mathscr{R}_{M-1}^{(n)}, \ldots, \mathscr{R}_1^{(n)}$ and $\mathscr{L}_1^{(n)}$ correspond to $X_1, X_2, \ldots, X_M$ and X_{M+1}, respectively. Thus, $\mathscr{A}^{(n)} := \mathscr{L}_1^{(n)}\mathscr{R}_1^{(n)}\mathscr{R}_2^{(n)}\cdots\mathscr{R}_M^{(n)}$ has the same form as X in (30). Let us assume that $U_1^{(n)}, U_2^{(n)}, \ldots, U_{M_m}^{(n)}$, appearing in $\mathscr{R}_1^{(n)}, \mathscr{R}_2^{(n)}, \ldots, \mathscr{R}_M^{(n)}$ and $\mathscr{L}_1^{(n)}$, are positive. Obviously, $\mathscr{L}_1^{(n)}, \mathscr{R}_1^{(n)}, \mathscr{R}_2^{(n)}, \ldots, \mathscr{R}_M^{(n)}$ are the TN matrices, and so $\mathscr{A}^{(n)}$ is also. As is shown in Sect. 2.1, the eigenvalues of $\mathscr{B}^{(n)}$ are the $(M+1)$th roots of $c_1, c_2, \ldots, c_m$. Hence, the eigenvalues of $\mathscr{A}^{(n)}$ are $c_1, c_2, \ldots, c_m$. In the case where $\mathscr{A}^{(n)}$ is decomposed as $\mathscr{A}^{(n)} = \mathscr{L}_1^{(n)}\mathscr{R}_1^{(n)}\mathscr{R}_2^{(n)}\cdots\mathscr{R}_M^{(n)}$, the eigenvalues of $\mathscr{A}^{(n)}$ are accordingly computed by the dhLV algorithm.

In [1], it is shown that any strictly sign-regular matrix has real and distinct eigenvalues. TN matrices are strictly sign-regular. In other words, TN matrices do not have multiple eigenvalues. Since $\mathscr{A}^{(n)}$ is a TN matrix, it is concluded that $c_1, c_2 \ldots, c_m$ are distinct. From (19), we have the following theorem.

Theorem 2 *Let $u_k^{(0)} > 0$ for $k = 1, 2, \ldots, M_m$. As $n \to \infty$, the dhLV variable $u_{M_k}^{(n)}$ converges to c_k, where*

$$c_1 > c_2 > \cdots > c_m. \tag{32}$$

More precisely, c_k is the eigenvalue of the TN matrix $\mathscr{A}^{(0)}$.

It is to be noted that (32) also holds in the eigenvalue computation of $\hat{\mathscr{L}}^{(0)} + dI$. The discussion in this subsection implies the sorting property (32), which is stronger than $c_1 \geq c_2 \geq \cdots \geq c_m$, that of (19).

2.3 Relationship with multiple dqd algorithm

Recently, one of the authors proposed the multiple dqd algorithm for computing eigenvalues of a TN band matrix [32].

Let $L_1, L_2, \ldots, L_{m_L}$ and $R_1, R_2, \ldots, R_{m_R}$ be the $m \times m$ lower and upper bidiagonal matrices, respectively, defined by

$$L_j = \begin{pmatrix} q_{j,1} & & & \\ 1 & q_{j,2} & & \\ & \ddots & \ddots & \\ & & 1 & q_{j,m} \end{pmatrix}, \quad R_j = \begin{pmatrix} 1 & e_{j,1} & & \\ & 1 & \ddots & \\ & & \ddots & e_{j,m-1} \\ & & & 1 \end{pmatrix}, \tag{33}$$

where $q_{j,1}, q_{j,2}, \ldots, q_{j,m} > 0$ and $e_{j,1}, e_{j,2}, \ldots, e_{j,m-1} > 0$. Then, the target matrix of the multiple dqd algorithm is represented as

$$A_{\mathrm{TN}} = L_1 L_2 \cdots L_{m_L} R_1 R_2 \cdots R_{m_R}. \tag{34}$$

Since it is obvious that $L_1, L_2, \ldots, L_{m_L}$ and $R_1, R_2, \ldots, R_{m_R}$ are TN matrices, so is A_{TN} [23]. Besides being a TN matrix, A_{TN} has no multiple eigenvalues. The basic idea of the multiple dqd algorithm is to employ the dqd algorithm $m_L \times m_R$ times for one LR transformation of A_{TN}. See [32] for the details concerning the appropriate convergence theorem.

Let us consider the case where $m_L = 1, m_R = M$ in the multiple dqd algorithm. Note that L_j and R_j in (33) have the same form as $\mathscr{L}_j^{(n)}$ in (28) and $\mathscr{R}_j^{(n)}$ in (29), respectively. Then, the form of A_{TN} in (34) coincides with that of $\mathscr{A}^{(n)} = \mathscr{L}_1^{(n)}\mathscr{R}_1^{(n)}\mathscr{R}_2^{(n)}\cdots\mathscr{R}_M^{(n)}$.

The target TN matrix of the dhLV algorithm is accordingly equal to that of the multiple dqd algorithm with $m_L = 1$ and $m_R = M$.

3 dhToda equation and matrix eigenvalue

We here investigate some properties of the dhToda equation (5) and then design a new algorithm for computing matrix eigenvalues in terms of the dhToda equation (5). The dhToda equation (5) with $M = 1$ is the discrete Toda equation (1), which has a close relationship to the qd algorithm for tridiagonal matrix eigenvalues. It is known that the discrete Toda equation (1) is just the recursion formula of the qd algorithm. And so it is no surprise that the dhToda equation (5) is also related to matrix eigenvalue problems. The main purpose of this section is to design a matrix eigenvalue algorithm in terms of the dhToda equation (5).

The dqd algorithm is an improved version of the qd algorithm and is algebraically equivalent to the qd algorithm. The dqd algorithm differs from the qd algorithm in that its recursion formula, called the differential form, has no subtraction. In other words, the dqd algorithm employs the differential form of the discrete Toda equation (1). In Sect. 3.1, we first derive a differential form of the dhToda equation, and we next show the positivity and the asymptotic behavior of the dhToda variables. In Sect. 3.2, based on the differential form of the dhToda equation, we finally design a new algorithm for computing eigenvalues. In Sect. 3.3, we also give relationships of the dhToda algorithm with the dhLV and the multiple dqd algorithms.

3.1 Properties of dhToda equation

Let us begin our analysis by deriving a differential form without subtraction from the dhToda equation (5). Let us introduce a new variable $D_k^{(n)}$ defined by

$$\begin{aligned} D_1^{(n)} &:= Q_1^{(n)}, \\ D_k^{(n)} &:= Q_k^{(n)} - E_{k-1}^{(n+1)}, \quad k = 2, 3, \ldots, m. \end{aligned}$$

Then, by combining this definition with (5), we obtain the relationship between $D_k^{(n)}$ and $D_{k+1}^{(n)}$,

$$D_{k+1}^{(n)} = \frac{Q_{k+1}^{(n)}}{Q_k^{(n+M)}} D_k^{(n)}.$$

Note that the ratio $Q_{k+1}^{(n)}/Q_k^{(n+M)}$ also appears in the second equation of (5). Moreover, let

$$F_{k+1}^{(n)} := \frac{Q_{k+1}^{(n)}}{Q_k^{(n+M)}}.$$

Then, the differential form without subtraction of (5) is given by

$$\begin{cases} Q_k^{(n+M)} = E_k^{(n)} + D_k^{(n)}, \quad k = 1, 2, \dots, m, \\ E_k^{(n+1)} = F_{k+1}^{(n)} E_k^{(n)}, \quad k = 1, 2, \dots, m-1, \\ D_{k+1}^{(n)} = F_{k+1}^{(n)} D_k^{(n)}, \quad D_1^{(n)} = Q_1^{(n)}, \quad F_{k+1}^{(n)} = \dfrac{Q_{k+1}^{(n)}}{Q_k^{(n+M)}}. \end{cases} \tag{35}$$

Though the recursion formula employed in (35) is different from that in (5), the sequences of $Q_k^{(n)}$ and $E_k^{(n)}$ generated by (35) coincide with those by (5). The differential form (35) is useful for clarifying the positivity of the dhToda variables $Q_k^{(n)}$ and $E_k^{(n)}$. If $Q_k^{(0)}, Q_k^{(1)}, \dots, Q_k^{(M-1)}$ for $k = 1, 2, \dots, m$ and $E_k^{(0)}$ for $k = 1, 2, \dots, m-1$ are positive, then $Q_k^{(M)}$ and $E_k^{(1)}$ are also positive. For $n = 1, 2, \dots$, by induction, we obtain the following proposition on the positivity of the dhToda variables.

Proposition 1 *Let $Q_k^{(0)} > 0, Q_k^{(1)} > 0, \dots, Q_k^{(M-1)} > 0$ for $k = 1, 2, \dots, m$ and $E_k^{(0)} > 0$ for $k = 1, 2, \dots, m-1$. Then, the variables $Q_k^{(n)}$, $E_k^{(n)}$, and $D_k^{(n)}$ in the differential form* (35) *satisfy the positivity conditions*

$$Q_k^{(n)} > 0, \quad k = 1, 2, \dots, m, \quad n = M, M+1, \dots, \tag{36}$$

$$E_k^{(n)} > 0, \quad k = 1, 2, \dots, m-1, \quad n = 1, 2, \dots, \tag{37}$$

$$D_k^{(n)} > 0, \quad k = 1, 2, \dots, m, \quad n = 0, 1, \dots. \tag{38}$$

With the help of Proposition 1, we have a theorem on an asymptotic convergence of the dhToda variables $Q_k^{(n)}$ and $E_k^{(n)}$ as $n \to \infty$.

Theorem 3 *Let $Q_k^{(0)} > 0, Q_k^{(1)} > 0, \dots, Q_k^{(M-1)} > 0$ for $k = 1, 2, \dots, m$ and $E_k^{(0)} > 0$ for $k = 1, 2, \dots, m-1$. As $n \to \infty$, the limits of $Q_k^{(n)}$ and $E_k^{(n)}$ are given by*

$$\lim_{n \to \infty} \prod_{j=0}^{M-1} Q_k^{(n-j)} = C_k, \quad k = 1, 2, \dots, m, \tag{39}$$

$$\lim_{n \to \infty} E_k^{(n)} = 0, \quad k = 1, 2, \dots, m-1, \tag{40}$$

where C_k is a nonnegative constant and $C_1 \geq C_2 \geq \dots \geq C_m$.

Proof We first give a proof of (40). Let us sum both sides of the first equation of (5) over the superscripts, from 0 to n:

$$\sum_{j=0}^{n} Q_k^{(j+M)} = \sum_{j=0}^{n} Q_k^{(j)} + \sum_{j=0}^{n} E_k^{(j)} - \sum_{j=0}^{n} E_{k-1}^{(j+1)}. \tag{41}$$

In order to consider the limit $n \to \infty$, we may assume that $n > M$ without loss of generality. Noting that $Q_k^{(M)}, Q_k^{(M+1)}, \dots, Q_k^{(n)}$ appear on both sides of (41), we derive

$$\sum_{j=n-M+1}^{n} Q_k^{(j+M)} = \sum_{j=0}^{M-1} Q_k^{(j)} + \sum_{j=0}^{n} E_k^{(j)} - \sum_{j=0}^{n} E_{k-1}^{(j+1)}. \tag{42}$$

From Proposition 1, it is obvious that $\sum_{j=n-M+1}^{n} Q_k^{(j+M)} > 0$. This implies that the right-hand side of (42) is positive. Hence, it follows that

$$\sum_{j=0}^{n} E_{k-1}^{(j+1)} < \sum_{j=0}^{M-1} Q_k^{(j)} + \sum_{j=0}^{n} E_k^{(j)}. \tag{43}$$

The case where $k = m$ and $n \to \infty$ in (43) with $\sum_{j=0}^{\infty} E_m^{(j)} = 0$ leads to, for positive constants $\bar{K}_0$,

$$\sum_{j=1}^{\infty} E_{m-1}^{(j)} < \sum_{j=0}^{M-1} Q_m^{(j)} < \bar{K}_0. \tag{44}$$

Successively, by considering the cases where $k = m-1, m-2, \dots, 1$, we have, for positive constant $\bar{K}_{m-k}$,

$$\sum_{j=0}^{\infty} E_k^{(j)} < \bar{K}_{m-k}, \quad k = m-1, m-2, \dots, 1. \tag{45}$$

From $E_k^{(n)} > 0$, it is concluded that $E_k^{(n)} \to 0$ as $n \to \infty$.

We next prove (39) with the help of (40). Let $n = \ell \times M + j$ in the first equation of (5); then

$$Q_k^{((\ell+1)\times M+j)} = Q_k^{(\ell\times M+j)} + E_k^{(\ell\times M+j)} - E_{k-1}^{(\ell\times M+j+1)}. \tag{46}$$

Moreover, let us sum both sides of (46) over ℓ from ℓ_1 to $\ell_2 - 1$, where $\ell_2 \geq \ell_1$. Then, it follows that

$$Q_k^{(\ell_2\times M+j)} = Q_k^{(\ell_1\times M+j)} + \sum_{\ell=\ell_1}^{\ell_2-1} E_k^{(\ell\times M+j)} - \sum_{\ell=\ell_1}^{\ell_2-1} E_{k-1}^{(\ell\times M+j+1)}. \tag{47}$$

By combining this with $E_k^{(n)} > 0$, we derive

$$\begin{aligned} \left| Q_k^{(\ell_2\times M+j)} - Q_k^{(\ell_1\times M+j)} \right| &\leq \left| \sum_{\ell=\ell_1}^{\ell_2-1} E_k^{(\ell\times M+j)} \right| + \left| \sum_{\ell=\ell_1}^{\ell_2-1} E_{k-1}^{(\ell\times M+j+1)} \right| \\ &\leq \left| \sum_{\ell=\ell_1}^{\infty} E_k^{(\ell\times M+j)} \right| + \left| \sum_{\ell=\ell_1}^{\infty} E_{k-1}^{(\ell\times M+j+1)} \right|. \end{aligned} \tag{48}$$

Noting that the right-hand side of (48) converges to zero as $\ell_1 \to \infty$, we have

$$\lim_{\ell_1,\ell_2\to\infty} |Q_k^{(\ell_2\times M+j)} - Q_k^{(\ell_1\times M+j)}| = 0, \tag{49}$$

which implies that $\left\{ Q_k^{(0\times M+j)}, Q_k^{(1\times M+j)}, Q_k^{(2\times M+j)}, \dots \right\}$ is a Cauchy sequence. Since $\left\{ Q_k^{(0\times M+j)}, Q_k^{(1\times M+j)}, Q_k^{(2\times M+j)}, \dots \right\}$ is a real positive sequence, it follows that $Q_k^{(\ell\times M+j)}$, for each j, converges to some nonnegative constant $C_{k,j}$ as $\ell \to \infty$. It is concluded that $\prod_{j=0}^{M-1} Q_k^{(n-j)}$ converges to some nonnegative constant $C_k = \prod_{j=0}^{M-1} C_{k,j}$ as $n \to \infty$.

We finally show the inequality conditions of C_k for $k = 1, 2, \ldots, m$. From the second equation of the dhToda equation (5),

$$E_k^{(n)} = E_k^{(0)} \prod_{N=0}^{n-1} \frac{Q_{k+1}^{(N)}}{Q_k^{(N+M)}} = E_k^{(0)} \prod_{\ell=0}^{n'} \frac{\mathscr{Q}_{k+1}^{(\ell)}}{\mathscr{Q}_k^{(\ell+1)}}, \tag{50}$$

where $\mathscr{Q}_k^{(\ell)} = \prod_{j=0}^{M-1} Q_k^{(\ell \times M - j)}$ and $n > n' \in \mathbf{N}$. Note that since $E_k^{(0)}$ is bounded and $\lim_{n\to\infty} E_k^{(n)} = 0$ for $k = 1, 2, \ldots, m-1$, from (50) we have

$$\lim_{n\to\infty} \prod_{\ell=0}^{n} \frac{\mathscr{Q}_{k+1}^{(\ell)}}{\mathscr{Q}_k^{(\ell+1)}} = 0, \quad k = 1, 2, \ldots, m-1. \tag{51}$$

If $\lim_{\ell\to\infty} \mathscr{Q}_{k+1}^{(\ell)}/\mathscr{Q}_k^{(\ell+1)} > 1$, this would contradict (51). From (39), we obtain

$$C_k \geq C_{k+1}, \quad k = 1, 2, \ldots, m-1. \tag{52}$$

□

To summarize, setting initial values of the dhToda variables appropriately yields that as n grows larger, $\prod_{j=0}^{M-1} Q_k^{(n-j)}$ and $E_k^{(n)}$, assuming $Q_k^{(n)} > 0$ and $E_k^{(n)} > 0$, converge to some nonnegative constant and zero, respectively. It is emphasized here that, as $n \to \infty$, the limit of $Q_k^{(n)}$ does not exist. This asymptotic behavior does not appear in the discrete Toda, the dLV, and the dhLV variables.

3.2 dhToda algorithm for a TN matrix

In order to find the conserved quantities for the dhToda equation, we consider the Lax form,

$$L^{(n+1)} R^{(n+M)} = R^{(n)} L^{(n)}, \tag{53}$$

$$L^{(n)} := \begin{pmatrix} 1 & & & & \\ E_1^{(n)} & 1 & & & \\ & E_2^{(n)} & \ddots & & \\ & & \ddots & 1 & \\ & & & E_{m-1}^{(n)} & 1 \end{pmatrix}, \tag{54}$$

$$R^{(n)} := \begin{pmatrix} Q_1^{(n)} & 1 & & & \\ & Q_2^{(n)} & 1 & & \\ & & \ddots & \ddots & \\ & & & \ddots & 1 \\ & & & & Q_m^{(n)} \end{pmatrix}. \tag{55}$$

Let us introduce the matrix, given by the matrix products of $L^{(n)}$ and $R^{(n)}, R^{(n+1)}, \ldots, R^{(n+M-1)}$,

$$A^{(n)} := L^{(n)} R^{(n+M-1)} R^{(n+M-2)} \cdots R^{(n+1)} R^{(n)}. \tag{56}$$

Note here that the entries of $A^{(n)}$ consist of the dhToda variables. Then, from (53), we derive $A^{(n+1)} = R^{(n)} A^{(n)} (R^{(n)})^{-1}$, which implies that the eigenvalues of $A^{(n)}$ are invariant under the time evolution from n to $n+1$. So, it should be emphasized here that $A^{(n)}$, for any n, has the same eigenvalues as $A^{(0)}$.

In [25], the conserved quantities of so-called numbered box and ball systems are presented based on the Lax form (53) for the dhToda equation (5). Since the eigenvalues of $A^{(n)}$ are invariant, conserved quantities of the dhToda equation (5) are given by

$$\mathrm{Tr}\{(A^{(n)})^j\}, \quad j = 1, 2, \ldots, m. \tag{57}$$

The authors of [25] suggest that the dhToda equation (5) has an interesting relationship with matrix eigenvalues. However, to the best of our knowledge, no matrix eigenvalue algorithm has been derived from the dhToda equation (5).

Now, we design a new algorithm for computing eigenvalues of $m \times m$ band matrix $A^{(0)}$ given by the matrix products of lower bidiagonal $L^{(0)}$ and upper bidiagonal $R^{(M-1)}, R^{(M-2)}, \ldots, R^{(0)}$ such that $A^{(0)} = L^{(0)} R^{(M-1)} R^{(M-2)} \cdots R^{(0)}$. If $m > M$, then the form of $A^{(n)}$ is as follows.

$$A^{(n)} = \begin{pmatrix} \overbrace{* & \cdots & *}^{M} & 1 & & \\ * & * & \cdots & * & \ddots & \\ & * & \ddots & & \ddots & 1 \\ & & \ddots & \ddots & & * \\ & & & \ddots & \ddots & \vdots \\ & & & & * & * \end{pmatrix}, \tag{58}$$

where $*$ denotes a nonzero entry. The $(i, i+M)$ entry of $A^{(n)}$ is fixed as 1, and the other nonzero entries consist of the dhToda variables $Q_k^{(n)}$ and $E_k^{(n)}$. If $m \leq M$, $A^{(n)}$ is the upper Hessenberg form without the entries fixed as 1. In both cases, the eigenvalues of $A^{(0)}$ are computable with the dhToda equation (5), as is shown in the following theorem.

Theorem 4 *Let* $Q_k^{(0)} > 0, Q_k^{(1)} > 0, \ldots, Q_k^{(M-1)} > 0$ *for* $k = 1, 2, \ldots, m$ *and* $E_k^{(0)} > 0$ *for* $k = 1, 2, \ldots, m-1$. *Then,* $C_k = \lim_{n\to\infty} \prod_{j=0}^{M-1} Q_k^{(n-j)}$ *for* $k = 1, 2, \ldots, m$ *coincide with the eigenvalues of* $A^{(0)}$.

Proof As is shown in the proof of Theorem 3, the subsequence $\left\{Q_k^{(0\times M+j)}, Q_k^{(1\times M+j)}, Q_k^{(2\times M+j)}, \ldots\right\}$ is a Cauchy sequence for all $k = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, M$. Obviously, the Cauchy sequence $\left\{Q_k^{(0\times M+j)}, Q_k^{(1\times M+j)}, Q_k^{(2\times M+j)}, \ldots\right\}$ is bounded. For arbitrary k and n, $Q_k^{(n)}$ is also bounded.

By combining the above with the convergence of $E_k^{(n)}$ shown in Theorem 3, we find that the $(i+1, i)$ entry of $A^{(n)}$, written as $E_i^{(n)} \prod_{j=0}^{M-1} Q_i^{(n+j)}$, converges to zero as $n \to \infty$. Let $(A_k^{(n)})_{i,i}$ be the diagonal (i, i) entry of $A_k^{(n)} = L^{(n)} R^{(n+M-1)} R^{(n+M-2)} \cdots R^{(n+M-k)}$. Then, $(A_k^{(n)})_{i,i}$ is given by

$$(A_k^{(n)})_{i,i} = E_{i-1}^{(n)} \prod_{j=M+1-k}^{M-1} Q_{i-1}^{(n+j)} + Q_i^{(n+M-k)} (A_{k-1}^{(n)})_{i,i}, \quad k = 2, 3, \ldots, M,$$
$$(A_0^{(n)})_{i,i} = 1, \quad (A_1^{(n)})_{i,i} = E_{i-1}^{(n)} + Q_i^{(n+M-1)}.$$

Noting that $A_M^{(n)} = A^{(n)}$, we derive from the limit of $E_k^{(n)}$ in (40) and the boundedness of $Q_k^{(n)}$ that the limit of the diagonal entry is $C_k = \lim_{n\to\infty} \prod_{j=0}^{M-1} Q_k^{(n-j)}$. Consequently, as $n \to \infty$, the matrix $A^{(n)}$ converges to the upper triangular matrix

$$\lim_{n\to\infty} A^{(n)} = \begin{pmatrix} C_1 & * & \cdots & * & 1 & & \\ & C_2 & * & \cdots & * & \ddots & \\ & & \ddots & \ddots & & \ddots & 1 \\ & & & \ddots & \ddots & & * \\ & & & & \ddots & \ddots & \vdots \\ & & & & & C_{m-1} & * \\ & & & & & & C_m \end{pmatrix}, \tag{59}$$

and the diagonal entries C_k for $k = 1, 2, \ldots, m$ are the eigenvalues of $A^{(0)}$. □

Let us recall here that $C_k \geq 0$ for $k = 1, 2, \ldots, m$ in Theorem 3. Moreover, using that $A^{(0)}$ are nonsingular in Theorem 4, we see that $C_k > 0$ for $k = 1, 2, \ldots, m$.

Suppose that the positive sequences $\{E_1^{(0)}, E_2^{(0)}, \ldots, E_{m-1}^{(0)}\}$ and $\{Q_1^{(0)}, Q_2^{(0)}, \ldots, Q_m^{(0)}\}$, $\{Q_1^{(1)}, Q_2^{(1)}, \ldots, Q_m^{(1)}\}, \ldots, \{Q_1^{(M-1)}, Q_2^{(M-1)}, \ldots, Q_m^{(M-1)}\}$ are given. Then, from (54)–(56), we have the band matrix $A^{(0)}$. Theorem 4 claims that, for sufficiently large n, $\prod_{j=0}^{M-1} Q_k^{(n-j)}$ becomes an approximate eigenvalue of $A^{(0)}$ through the dhToda equation (5). The above procedure for matrix eigenvalues is called the dhToda algorithm and is shown below.

dhToda algorithm

for $n := 0, 1, 2, \ldots, n_{\max}$ **do**
  $D_1^{(n)} = Q_1^{(n)}$
  for $k := 1, 2, \ldots, m-1$ **do**
    $Q_k^{(n+M)} = E_k^{(n)} + D_k^{(n)}$
    $F_{k+1}^{(n)} = Q_{k+1}^{(n)} / Q_k^{(n+M)}$
    $E_k^{(n+1)} = F_{k+1}^{(n)} E_k^{(n)}$
    $D_{k+1}^{(n)} = F_{k+1}^{(n)} D_k^{(n)}$
  end for
  $Q_m^{(n+M)} = D_m^{(n)}$
end for
for $k := 0, 1, 2, \ldots, m$ **do**
  $C_k = \prod_{j=0}^{M-1} Q_k^{(n-j)}$
end for

The values M and m are given from the form of $A^{(0)}$, and the parameter $n_{\max}$ is set as the maximum iteration number. The inequality $\max_k E_k^{(n)} < eps$ is employed as the stopping criterion, where $eps > 0$ is sufficiently small.

In concluding this subsection, we discuss the possibility of incorporating shift of origin to accelerate convergence into the dhToda algorithm. Starting from the expression (56) for $A^{(n+M)}$ and using (53) repeatedly, we have

$$\begin{aligned} A^{(n+M)} &= L^{(n+M)} R^{(n+2M-1)} R^{(n+2M-2)} \cdots R^{(n+M+1)} R^{(n+M)} \\ &= R^{(n+M-1)} L^{(n+M-1)} R^{(n+2M-2)} \cdots R^{(n+M+1)} R^{(n+M)} \\ &\ \ \vdots \\ &= R^{(n+M-1)} R^{(n+M-2)} \cdots R^{(n+1)} R^{(n)} L^{(n)} \\ &= (L^{(n)})^{-1} A^{(n)} L^{(n)}. \end{aligned} \tag{60}$$

This shows that the transformation from $A^{(n)}$ to $A^{(n+M)}$ can be viewed as one step of the LR algorithm. Hence, we can introduce the shift $s^{(n)}$ as follows:

$$A^{(n)} - s^{(n)} I = L^{(n)} R^{(n+M-1)} R^{(n+M-2)} \cdots R^{(n+1)} R^{(n)} - s^{(n)} I = \mathscr{L}^{(n)} \mathscr{R}^{(n)}, \tag{61}$$

$$A^{(n+M)} = (\mathscr{L}^{(n)})^{-1} A^{(n)} \mathscr{L}^{(n)}. \tag{62}$$

Here, the rightmost-hand side of (61) is the LR decomposition of the left-hand side. In this modified transformation, if we compute $A^{(n)} - s^{(n)} I$ explicitly and then compute its LR factors, small eigenvalues can suffer from loss of accuracy. This is because small relative error in the matrix element of $A^{(n)} - s^{(n)} I$ generally causes large relative error in the smallest eigenvalues [16]. Fortunately, thanks to the implicit L theorem for the LR algorithm, we can compute the LR transformation (62) without forming $A^{(n)} - s^{(n)} I$ explicitly. To develop a shifted dhToda algorithm based on this idea, however, we still need to solve a few problems. For example, the condition imposed on $s^{(n)}$ to ensure positivity of the bidiagonal factors must be clarified. Also, an efficient strategy for determining the shift based on the bidiagonal factors must be developed. We are currently investigating these problems, and they will be the subjects of our next paper.

3.3 Relationships with dhLV and multiple dqd algorithms

We next clarify the relationship of the dhToda algorithm to the dhLV algorithm. Let us introduce the block matrix $B^{(n)} \in \mathbf{R}^{m(M+1) \times m(M+1)}$, composed of the matrices $L^{(n)}$ in (54) and $R^{(n+M-1)}, \ldots, R^{(n+1)}, R^{(n)}$ in (55), such that

$$B^{(n)} = \begin{pmatrix} & & & L^{(n)} \\ R^{(n+M-1)} & & & \\ & R^{(n+M-2)} & & \\ & & \ddots & \\ & & & R^{(n)} \end{pmatrix}. \tag{63}$$

Let us recall here the permutation technique shown in Sect. 2.2. We consider the inverse of the permutation. Since the permutation matrix P is such that $PB^{(n)}$ is the matrix given by interchanging the $[(j-1)m+k]$th and $[(k-1)(M+1)+j]$th rows of $B^{(n)}$, $P^\top$ is such that $P^\top B^{(n)}$ is the matrix given by interchanging the $[(k-1)(M+1)+j]$th and $[(j-1)m+k]$th rows of $B^{(n)}$. It follows that

$$P^{-1}B^{(n)}P = \begin{pmatrix} S_1^{(n)} & J & & \\ H_1^{(n)} & S_2^{(n)} & \ddots & \\ & \ddots & \ddots & J \\ & & H_{m-1}^{(n)} & S_m^{(n)} \end{pmatrix}, \tag{64}$$

$$S_k^{(n)} := \begin{pmatrix} 0 & & & 1 \\ Q_k^{(n+M-1)} & 0 & & \\ & \ddots & \ddots & \\ & & Q_k^{(n)} & 0 \end{pmatrix}, \tag{65}$$

$$H_k^{(n)} := \begin{pmatrix} 0 & \cdots & 0 & E_k^{(n)} \\ 0 & \cdots & 0 & 0 \\ \vdots & & \vdots & \vdots \\ 0 & \cdots & 0 & 0 \end{pmatrix}, \quad J := \begin{pmatrix} 0 & & & \\ 1 & 0 & & \\ & \ddots & \ddots & \\ & & 1 & 0 \end{pmatrix}. \tag{66}$$

The band matrix $P^{-1}B^{(n)}P$ in (64) has the same form as $\hat{\mathscr{L}}^{(n)}$ in (25), whose eigenvalues are computable by the dhLV algorithm. Just as in the discussion in Sect. 2.2, the band matrices $A^{(n)}$ in (58) and $B^{(n)}$ in (63) correspond to the matrices X and $\hat{X}$ in Theorem 1, respectively. So, it follows that all the eigenvalues of $B^{(n)}$ are given as the $(M+1)$th roots of those of $A^{(n)}$. On the other hand, $B^{(n)}$ and $P^{-1}B^{(n)}P$ in (64) are similar and have the same eigenvalues. Thus, it is concluded that all the eigenvalues of $P^{-1}B^{(n)}P$ are given as the $(M+1)$th roots of those of $A^{(n)}$. Namely, the target matrix of the dhLV algorithm becomes that of the dhToda algorithm by a suitable initial setting.

The dhToda algorithm is also related to the multiple dqd algorithm. By comparing $L^{(n)}$ in (54) and $R^{(n)}$ in (55) with L_j, R_j in (33), we see that the transpose of $L^{(n)}$ and $R^{(n)}$ have the same forms as R_j and L_j, respectively. So, let

$$(L^{(0)})^\top = R_1, \quad (R^{(0)})^\top = L_1, \quad (R^{(1)})^\top = L_2, \quad \ldots, \quad (R^{(M-1)})^\top = L_M.$$

Let us recall that $A^{(0)} = L^{(0)}R^{(M-1)}R^{(M-2)}\cdots R^{(0)}$ is the target matrix of the dhToda algorithm. Then, the transpose of $A^{(0)}$ is

$$\begin{aligned}(A^{(0)})^\top &= (R^{(0)})^\top (R^{(1)})^\top \cdots (R^{(M-1)})^\top (L^{(0)})^\top \\ &= L_1 L_2 \cdots L_M R_1.\end{aligned}$$

This implies that the target matrices of the dhToda algorithm and the multiple dqd algorithm with $m_L = M, m_R = 1$ are similar to each other.

4 Numerical experiments

In this section, we numerically confirm our results shown in the previous sections. Numerical experiments have been carried out on a computer with the following specifications: OS, Windows XP; CPU, Genuine Intel (R) CPU L2400 @ 1.66 GHz; RAM, 2 GB; compiler, Microsoft(R) C/C++ Optimizing Compiler Version 15.00.30729.01.

As an example matrix, we adopt the TN matrix $A_0 = L^{(0)} R^{(2)} R^{(1)} R^{(0)}$ with

$$L^{(0)} = \begin{pmatrix} 1 & & & \\ 2 & 1 & & \\ & 2 & 1 & \\ & & 2 & 1 \end{pmatrix}, \quad R^{(0)} = R^{(1)} = R^{(2)} = \begin{pmatrix} 5 & 1 & & \\ & 5 & 1 & \\ & & 5 & 1 \\ & & & 5 \end{pmatrix}.$$

Note here that $M = 3$ and $m = 4$ in our dhLV and the dhToda algorithms. We first discuss the behavior of the dhToda variables $Q_k^{(n)}$ and $E_k^{(n)}$. See also [5] for the behavior of the dhLV variables $u_k^{(n)}$. From Fig. 1, it is clear that, as is stated in Theorem 3, $E_k^{(n)}$ converges to zero. Figure 2 shows that the behavior of $Q_k^{(n)}$ gradually becomes periodic as n grows larger. From Fig. 3, it is obvious that the product $p_k^{(n)} := \prod_{j=0}^{M-1} Q_k^{(n+j)}$ converges to some positive constant. This numerical convergence also agrees with Theorem 3.

Next, we demonstrate that, for two kinds of matrices, the eigenvalues are computable by the dhToda and the dhLV algorithms. Let us set $eps = 1.0E - 16$ in both algorithms. We introduce the block matrix

$$B_0 = \begin{pmatrix} & & & L^{(0)} \\ R^{(2)} & & & \\ & R^{(1)} & & \\ & & R^{(0)} & \end{pmatrix}. \tag{67}$$

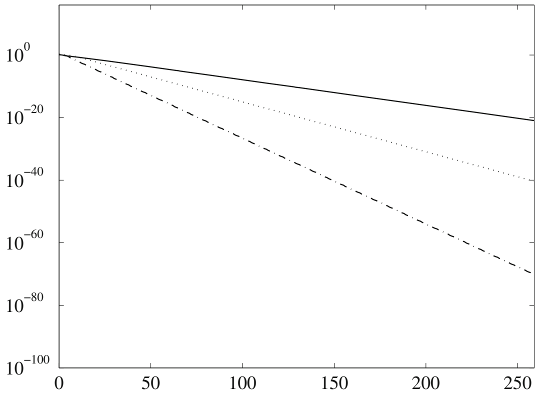

Fig. 1 Graph of the iteration number n (x-axis) and $E_1^{(n)}, E_2^{(n)}, E_3^{(n)}$ (y-axis) in the dhToda algorithm. *Solid line* $E_1^{(n)}$; *dotted line* $E_2^{(n)}$; and *dashed line* $E_3^{(n)}$

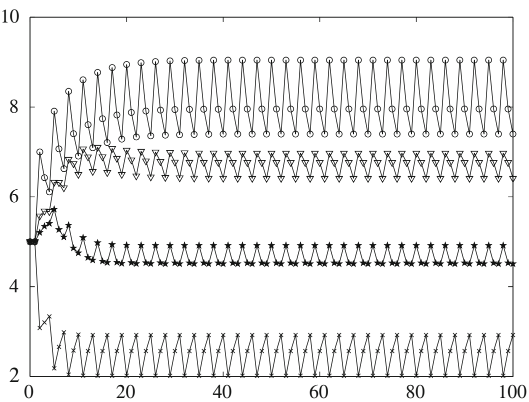

Fig. 2 Graph of the iteration number n (x-axis) and $Q_1^{(n)}, Q_2^{(n)}, Q_3^{(n)}, Q_4^{(n)}$ (y-axis) in the dhToda algorithm. *Circle* $Q_1^{(n)}$; *bigtriangledown* $Q_2^{(n)}$; *star* $Q_3^{(n)}$, and *cross* $Q_4^{(n)}$

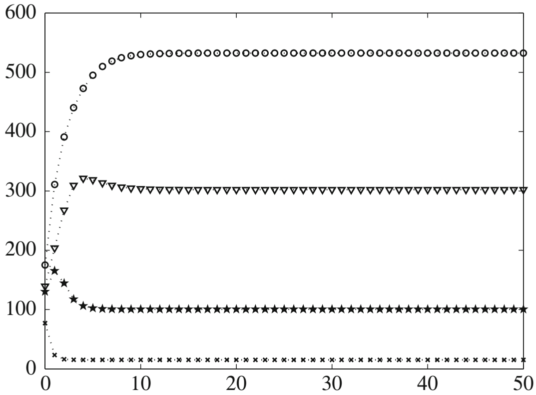

Fig. 3 Graph of the ratio of the iteration number n to the parameter $M = 3$ (x-axis) and $p_1^{(n)}, p_2^{(n)}, p_3^{(n)}, p_4^{(n)}$ (y-axis) in the dhToda algorithm. *Circle* $p_1^{(n)}$; *bigtriangledown* $p_2^{(n)}$; *star* $p_3^{(n)}$; and *cross* $p_4^{(n)}$

For the suitable permutation matrix P_0 shown in Sect. 3.3, the band matrix $L_0 := P_0^\top B_0 P_0$ has two diagonals, as follows:

$$L_0 = \left(\begin{array}{cccc|cccc|cccc|cccc}
 & & & 1 & & & & & & & & & & & & \\
5 & & & & 1 & & & & & & & & & & & \\
 & 5 & & & & 1 & & & & & & & & & & \\
 & & 5 & & & & 1 & & & & & & & & & \\
\hline
 & & & 2 & & & & 1 & & & & & & & & \\
 & & & & 5 & & & & 1 & & & & & & & \\
 & & & & & 5 & & & & 1 & & & & & & \\
 & & & & & & 5 & & & & 1 & & & & & \\
\hline
 & & & & & & & 2 & & & & 1 & & & & \\
 & & & & & & & & 5 & & & & 1 & & & \\
 & & & & & & & & & 5 & & & & 1 & & \\
 & & & & & & & & & & 5 & & & & 1 & \\
\hline
 & & & & & & & & & & & 2 & & & & 1 \\
 & & & & & & & & & & & & 5 & & & \\
 & & & & & & & & & & & & & 5 & & \\
 & & & & & & & & & & & & & & 5 &
\end{array}\right). \tag{68}$$

In order to get the eigenvalues of A_0 and L_0 with high relative accuracy, we employ the Mathematica function `eigenvalues[ ]` with 100-digit arithmetic. We also use our dhLV and dhToda algorithms in double-precision arithmetic.

Tables 1 and 2 show the eigenvalues of A_0 and L_0, respectively, computed by `eigen-values[ ]` and our algorithms. The first column of numbers in both tables display the results obtained by rounding `eigenvalues[`A_0`]` or `eigenvalues[`L_0`]` into double-precision numbers. The computed eigenvalues by the dhLV and the dhToda algorithms, respectively, are shown in the second and the third columns of numbers. By comparing the second and third columns with the first column for each table, we conclude that the eigenvalues of both matrices are computed by the dhLV and dhToda algorithms with high relative accuracy.

Finally, we give a comparison of the dhToda algorithm with the routine `dhseqr` in the famous LAPACK [17] with respect to the relative accuracy of computed eigenvalues. Let us introduce the 20-by-20 TN matrix $A_1 = L^{(0)} R^{(7)} R^{(6)} \cdots R^{(0)}$, where the nonzero entries of $L^{(0)}$, and $R^{(0)}, R^{(1)}, \ldots, R^{(7)}$ are all 1. The largest and the smallest eigenvalues of A_1 are $2.284895467291726E + 1$ and $1.448103061761318E - 7$, respectively. Figure 4 shows the

Table 1 Computed eigenvalues of A_0

	Mathematica	dhLV algorithm	dhToda algorithm
λ_1	532.35140651953578	532.35140651953509	532.35140651953520
λ_2	302.15799192937254	302.15799192937277	302.15799192937300
λ_3	100.36858294952133	100.36858294952130	100.36858294952131
λ_4	15.122018601570330	15.122018601570332	15.122018601570332

Table 2 Computed eigenvalues of L_0

	Mathematica	dhLV algorithm	dhToda algorithm
$\lambda_{1,1}$	4.803409376080853	4.803409376080851	4.803409376080851
$\lambda_{1,2}$	4.803409376080853i	4.803409376080851i	4.803409376080851i
$\lambda_{1,3}$	−4.803409376080853	−4.803409376080851	−4.803409376080851
$\lambda_{1,4}$	−4.803409376080853i	−4.803409376080851i	−4.803409376080851i
$\lambda_{2,1}$	4.169255606169454	4.169255606169454	4.169255606169455
$\lambda_{2,2}$	4.169255606169454i	4.169255606169454i	4.169255606169455i
$\lambda_{2,3}$	−4.169255606169454	−4.169255606169454	−4.169255606169455
$\lambda_{2,4}$	−4.169255606169454i	−4.169255606169454i	−4.169255606169455i
$\lambda_{3,1}$	3.165187545316407	3.165187545316406	3.165187545316406
$\lambda_{3,2}$	3.165187545316407i	3.165187545316406i	3.165187545316406i
$\lambda_{3,3}$	−3.165187545316407	−3.165187545316406	−3.165187545316406
$\lambda_{3,4}$	−3.165187545316407i	−3.165187545316406i	−3.165187545316406i
$\lambda_{4,1}$	1.971979709463506	1.971979709463506	1.971979709463506
$\lambda_{4,2}$	1.971979709463506i	1.971979709463506i	1.971979709463506i
$\lambda_{4,3}$	−1.971979709463506	−1.971979709463506	−1.971979709463506
$\lambda_{4,4}$	−1.971979709463506i	−1.971979709463506i	−1.971979709463506i

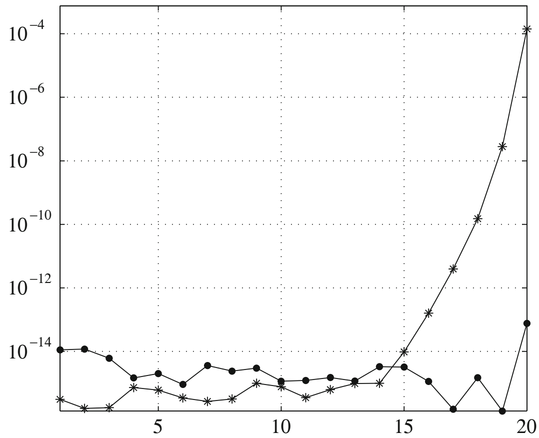

Fig. 4 Graph of the index k of the computed eigenvalues $\hat{\lambda}_1, \hat{\lambda}_2, \dots, \hat{\lambda}_{20}$ with $\hat{\lambda}_1 > \hat{\lambda}_2 > \cdots > \hat{\lambda}_{20}$ (x-axis) and the relative errors of $\hat{\lambda}_1, \hat{\lambda}_2, \dots, \hat{\lambda}_{20}$ (y-axis) by the dhToda algorithm and by the routine `dhseqr` in the case where the target matrix is A_1. *Filled circle* dhToda algorithm; and *asterisk*, routine `dhseqr`

relative errors of the eigenvalues of A_1 computed by the dhToda algorithm and by the routine `dhseqr`. From Fig. 4, we observe that the dhToda algorithm is preferable to the routine `dhseqr` for computing eigenvalues of A_1 with high relative accuracy.

5 Concluding remarks

In this paper, we first survey the dhLV algorithm in [5] derived from the integrable dhLV system and then expand the target matrix of the dhLV algorithm by considering a TN matrix. We next investigate properties of the integrable dhToda equation. It is found that the dhToda variables become periodic and their products converge to matrix eigenvalues as the time variable $n \to \infty$. Using this asymptotic convergence, we design a new algorithm, named the dhToda algorithm, for computing eigenvalues of a TN matrix. We describe the relationship of the dhLV algorithm to the dhToda algorithm, namely, that the classes of matrices whose eigenvalues are computable by both the dhLV algorithm and the dhToda algorithm are essentially the same. It should be remarked here that a transformation, such as a Miura transformation (3), from the dhLV variables to those of the dhToda or vice versa has not yet been reported, but the dhLV algorithm is nevertheless related to the dhToda algorithm from the viewpoint of matrix eigenvalues. It is also shown that our two algorithms are related to the multiple dqd algorithm, which is proposed for the computation of eigenvalues of a TN matrix in [32]. Through numerical experiments, we confirm that the dhToda variables have the asymptotic convergence predicted theoretically and that the eigenvalues computed by our two algorithms have high relative accuracy.

Acknowledgments The authors would like to thank Prof. S. Tsujimoto and Prof. A. Nagai for many fruitful discussions and helpful advice regarding this work. The authors also thank the reviewers for their careful reading and insightful suggestions.

References

1. Ando, T.: Totally positive matrices. Linear Algebra Appl. **90**, 165–219 (1987)
2. Bogoyavlensky, O.I.: Integrable discretizations of the KdV equation. Phys. Lett. A **134**, 34–38 (1988)
3. Brenti, F.: Combinatrics and total positivity. J. Combin. Theory Ser. A **71**, 175–218 (1995)
4. Chu, M.T.: Linear algebra algorithm as dynamical systems. Acta Numer. **17**, 1–86 (2008)
5. Fukuda, A., Ishiwata, E., Iwasaki, M., Nakamura, Y.: The discrete hungry Lotka-Volterra system and a new algorithm for computing matrix eigenvalues. Inverse Probl. **25**, 015007 (2009)
6. Gantmacher, F., Krein, M.: Oscillation Matrices and Kernels and Small Vibrations of Mechanical Systems, revised edition. AMS Chelsea, Providence (2002)
7. Gasca, M., Micchelli, C.A. (eds.): Total Positivity and Its Applications, Math. Appl. **359**. Kluwer, Dordrecht (1996)
8. Geist, G.A., Howell, G.W., Watkins, D.S.: The BR eigenvalue algorithm. SIAM J. Mat. Anal. Appl. **20**, 1083–1098 (1999)
9. Hirota, R.: Discrete analogue of a generalized Toda equation. J. Phys. Soc. Jpn. **50**, 3785–3791 (1981)
10. Hirota, R.: Conserved quantities of a random-time Toda equation. J. Phys. Soc. Jpn. **66**, 283–284 (1997)
11. Itoh, Y.: Integrals of a Lotka-Volterra system of odd number of variables. Prog. Theor. Phys. **78**, 507–510 (1987)
12. Iwasaki, M., Nakamura, Y.: On the convergence of a solution of the discrete Lotka-Volterra system. Inverse Probl. **18**, 1569–1578 (2002)
13. Iwasaki, M., Nakamura, Y.: An application of the discrete Lotka-Volterra system with variable step-size to singular value computation. Inverse Probl. **20**, 553–563 (2004)
14. Iwasaki, M., Nakamura, Y.: Accurate computation of singular values in terms of shifted integrable schemes. Jpn. J. Indust. Appl. Math. **23**, 239–259 (2006)
15. Karlin, S.: Total Positivity, vol. I. Stanford University Press, Stanford (1968)

16. Koev, P.: Accurate eigenvalue and SVDs of totally nonnegative matrices. SIAM J. Matrix Anal. Appl. **27**, 1–23 (2005)
17. LAPACK: http://www.netlib.org/lapack/
18. Nagai, A., Tokihiro, T., Satsuma, J.: The Toda molecule equation and the ϵ-algorithm. Math. Comp. **67–224**, 1565–1575 (1998)
19. Nakamura, Y.: Calculating Laplace transforms in terms of the Toda molecule. SIAM J. Sci. Comput. **20**, 306–317 (1999)
20. Nakamura, Y., Mukaihira, A.: Dynamics of the finite Toda molecule over finite fields and a decoding algorithm. Phys. Lett. A **249**, 295–302 (1998)
21. Nakamura, Y. (ed.): Applied Integrable Systems (in Japanese). Shokabo, Tokyo (2000)
22. Parlett, B.N.: The new qd algorithms. Acta Numer. **4**, 459–491 (1995)
23. Pinkus, A.: Totally Positive Matrices. Cambridge University Press, New York (2010)
24. Takahashi, D., Matsukidaira, J.: Box and ball system with a carrier and ultradiscrete modified KdV equation. J. Phys. A **30–21**, L733–L739 (1997)
25. Tokihiro, T., Nagai, A., Satsuma, J.: Proof of solitonical nature of box and ball systems by means of inverse ultra-discretization. Inverse Probl. **15**, 1639–1662 (1999)
26. Rutishauser, H.: Ein infinitesimales Analogon zum Quotienten-Differenzen-Algorithmus. Arch. Math. **5**, 132–137 (1954)
27. Rutishauser, H.: Solution of eigenvalue problems with the LR transformation. Nat. Bur. Stand. Appl. Math. Ser. **49**, 47–81 (1958)
28. Rutishauser, H.: Lectures on Numerical Mathematics, Birkhäuser, Boston (1990)
29. Symes, W.W.: The QR algorithm and scattering for the finite nonperiodic Toda lattice. Physica D **4**, 275–280 (1982)
30. Tsujimoto, S., Hirota, R., Oishi, S.: An extension and discretization of Volterra equation I (in Japanese). Tech. Rep. Proc. IEICE NLP **92–90**, 1–3 (1993)
31. Watkins, D.S.: Product eigenvalue problems. SIAM Rev. **47**, 3–40 (2005)
32. Yamamoto, Y., Fukaya, T.: Differential qd algorithm for totally nonnegative band matrices: convergence properties and error analysis. JSIAM Lett. **1**, 56–59 (2009)
33. Yamazaki, S.: On the system of non-linear differential equations $\dot{y}_k = y_k(y_{k+1} - y_{k-1})$. J. Phys. A: Math. Gen. **20**, 6237–6241 (1987)