

## 離散戸田方程式のある拡張に基づく Totally Nonnegative 行列の固有値計算アルゴリズム

隅蔵, 亮  
東京理科大学

福田, 亜希子  
芝浦工業大学

石渡, 恵美子  
東京理科大学

山本, 有作  
電気通信大学

他

<https://doi.org/10.15017/1807771>

---

出版情報 : 応用力学研究所研究集会報告. 26A0-S2 (20), pp.115-120, 2015-03. 九州大学応用力学研究所  
バージョン :  
権利関係 :

応用力学研究所研究集会報告 No.26AO-S2  
「非線形波動研究の現状 — 課題と展望を探る—」 (研究代表者 増田 哲)

Reports of RIAM Symposium No.26AO-S2

*State of arts and perspectives of nonlinear wave science*

Proceedings of a symposium held at Chikushi Campus, Kyushu University,  
Kasuga, Fukuoka, Japan, October 30 - November 1, 2014

Article No. 20 (pp. 115 - 120)

# 離散戸田方程式のある拡張に基づく Totally Nonnegative 行列の固有値計 算アルゴリズム

隅蔵 亮 (SUMIKURA Ryo), 福田 亜希子 (FUKUDA Akiko), 石渡 恵美子 (ISHIWATA Emiko), 山本 有作 (YAMAMOTO Yusaku), 岩崎 雅史 (IWASAKI Masashi), 中村 佳正 (NAKAMURA Yoshimasa)

(Received 15 January 2015; accepted 24 February 2015)



Research Institute for Applied Mechanics  
Kyushu University  
March, 2015

# 離散戸田方程式のある拡張に基づく Totally Nonnegative 行列の固有値計算アルゴリズム

東京理科大学	隅歳 亮	(SUMIKURA Ryo)
芝浦工業大学	福田 亜希子	(FUKUDA Akiko)
東京理科大学	石渡 恵美子	(ISHIWATA Emiko)
電気通信大学	山本 有作	(YAMAMOTO Yusaku)
京都府立大学	岩崎 雅史	(IWASAKI Masashi)
京都大学	中村 佳正	(NAKAMURA Yoshimasa)

## 概要

まず離散戸田方程式の拡張版に基づいて上 Hessenberg 型をした totally nonnegative 行列に対する固有値計算アルゴリズムを定式化する。続いて、収束を加速させるための原点シフトを導入し、そのシフト付きアルゴリズムの収束性について述べる。さらに、数値実験を通じてシフト付きアルゴリズムの有効性を確認する。

## 1 はじめに

実対称 3 重対角行列の固有値を求められる qd アルゴリズムはその核となる漸化式が離散戸田方程式と一致する。箱玉系の視点から離散戸田方程式の拡張版とみなせる離散方程式として任意の自然数  $M$  を含む

$$\begin{cases} Q_j^{(n+M)} = Q_j^{(n)} + E_j^{(n)} - E_{j-1}^{(n+1)}, & j = 1, 2, \dots, m, \\ E_j^{(n+1)} = \frac{Q_{j+1}^{(n)}}{Q_j^{(n+M)}} E_j^{(n)}, & j = 1, 2, \dots, m-1, \\ E_0^{(n)} \equiv 0, E_m^{(n)} \equiv 0, & n = 0, 1, \dots, \end{cases} \quad (1.1)$$

$$\begin{cases} Q_j^{(n+1)} = Q_j^{(n)} + E_j^{(n)} - E_{j-1}^{(n+M)}, & j = 1, 2, \dots, m, \\ E_j^{(n+M)} = \frac{Q_{j+1}^{(n)}}{Q_j^{(n+1)}} E_j^{(n)}, & j = 1, 2, \dots, m-1, \\ E_0^{(n)} \equiv 0, E_m^{(n)} \equiv 0, & n = 0, 1, \dots \end{cases} \quad (1.2)$$

の 2 つが挙げられる [1, 2]。ここで、 $j, n$  はそれぞれ空間変数, 時間変数を表す。(1.1), (1.2) はともに離散ハングリー戸田方程式 (dhToda: discrete hungry Toda equation) と呼ばれる。本稿では、区別のために (1.1), (1.2) をそれぞれ  $\text{dhToda}_I$ ,  $\text{dhToda}_{II}$  と名付ける。

$\text{dhToda}_I$  (1.1) の時間発展を利用して、下 Hessenberg 型をした totally nonnegative (TN) 行列の固有値を求めるアルゴリズムが定式化されている [3]。ここで、TN 行列とは全ての小行列が非負の正方行列であり、組み合わせ論などの分野に現れる。アルゴリズムの収束を加速させるために原点シフトの導入も提案されている [4]。一方、 $\text{dhToda}_{II}$  (1.2) から導かれる固有値計算アルゴリズムは報告されていない。本稿では、 $\text{dhToda}_{II}$  (1.2) の時間発展によって上 Hessenberg 型をした TN 行列の固有値が求まることを明らかにし、収束加速のための原点シフトを導入したシフト付きアルゴリズムを定式化する。

本稿の構成は以下の通りである。2 節では  $\text{dhToda}_{II}$  (1.2) の時間発展によって上 Hessenberg 型の TN 行列の固有値が求まる仕組みについて概説する。3 節では、収束加速のための原点シフトを加えたシフト付きアルゴリズムを提案し、その大域的収束性についても示す。4 節では、原点シフトによる収束加速を数値例により確認する。

## 2 dhToda<sub>II</sub> に基づく固有値保存変形

本節では, dhToda<sub>II</sub> (1.2) のラックス表示と dhToda<sub>II</sub> 変数の  $n \rightarrow \infty$  における漸近挙動を踏まえて, dhToda<sub>II</sub> (1.2) の時間発展によって上 Hessenberg 型をした TN 行列の固有値が求まることを明らかにする.

dhToda<sub>II</sub> 変数  $Q_j^{(n)}, E_j^{(n)}$  を含む 2 重対角行列

$$L^{(n)} := \begin{pmatrix} Q_1^{(n)} & & & & & \\ & 1 & & & & \\ & & Q_2^{(n)} & & & \\ & & & 1 & & \ddots \\ & & & & \ddots & \ddots \\ & & & & & & 1 & Q_m^{(n)} \end{pmatrix}, R^{(n)} := \begin{pmatrix} 1 & E_1^{(n)} & & & & \\ & 1 & E_2^{(n)} & & & \\ & & \ddots & \ddots & & \\ & & & \ddots & \ddots & \\ & & & & \ddots & E_{m-1}^{(n)} \\ & & & & & 1 \end{pmatrix} \quad (2.1)$$

を導入すると, dhToda<sub>II</sub> (1.2) のラックス表示は

$$L^{(n+1)}R^{(n+M)} = R^{(n)}L^{(n)} \quad (2.2)$$

で与えられる. ここで, 上 Hessenberg 型行列として

$$A^{(n)} := L^{(n)}R^{(n+M-1)}R^{(n+M-2)} \dots R^{(n+1)}R^{(n)} \quad (2.3)$$

を導入すると, (2.2) において  $R^{(n)}$  が正則なので,  $A^{(n+1)} = R^{(n)}A^{(n)}(R^{(n)})^{-1}$  が成り立つ. これは, dhToda<sub>II</sub> (1.2) の  $n$  から  $n+1$  への時間発展によって  $A^{(n)}$  から  $A^{(n+1)}$  への固有値保存変形が与えられることを意味する.  $Q_j^{(0)} > 0, E_j^{(0)} > 0, E_j^{(1)} > 0, \dots, E_j^{(M-1)} > 0$  のとき  $L^{(0)}, R^{(0)}, R^{(1)}, \dots, R^{(M-1)}$  は TN 行列であり,  $A^{(0)}$  も TN 行列である. また,  $n \rightarrow \infty$  における dhToda<sub>II</sub> 変数  $Q_j^{(n)}, E_j^{(n)}$  の漸近挙動は  $c_1, c_2, \dots, c_m$  を  $c_1 > c_2 > \dots > c_m$  を満たす正の定数とすると,

$$\lim_{n \rightarrow \infty} Q_j^{(n)} = c_j, \quad j = 1, 2, \dots, m, \quad (2.4)$$

$$\lim_{n \rightarrow \infty} E_j^{(n)} = 0, \quad j = 1, 2, \dots, m-1 \quad (2.5)$$

なので,  $\lim_{n \rightarrow \infty} A^{(n)}$  は  $c_1, c_2, \dots, c_m$  が対角成分に並ぶ下 2 重対角行列となることが分かる. よって,  $\lim_{n \rightarrow \infty} A^{(n)}$  と相似である TN 行列  $A^{(0)}$  の固有値は  $c_1, c_2, \dots, c_m$  であると結論付けられる. ゆえに, TN 行列  $A^{(0)}$  の成分から  $Q_j^{(0)}, E_j^{(0)}, E_j^{(1)}, \dots, E_j^{(M-1)}$  を定め, dhToda<sub>II</sub> (1.2) の時間発展によって十分大きな  $N$  に対する  $Q_j^{(N)}$  を求めれば,  $Q_j^{(N)}$  が  $A^{(0)}$  の固有値の近似値となる. これを dhToda<sub>II</sub> アルゴリズムと呼ぶことにする.

## 3 原点シフトの導入

本節では, dhToda<sub>II</sub> アルゴリズムの収束加速のために, multiple dqd アルゴリズムに関する研究 [5] に倣って  $LR$  変換の視点から原点シフトを導入し, シフト付き dhToda<sub>II</sub> アルゴリズムの大域的収束性を明らかにする.

上 Hessenberg 型の TN 行列  $A^{(n)}$  に対するシフト付き  $LR$  変換は, シフト量を  $s^{(n)}$ ,  $m$  次単位行列を  $I$  とすると,

$$A^{(n)} - s^{(n)}I = L^{(n,0)}\mathcal{R}^{(n)}, \quad (3.1)$$

$$A^{(n+M)} = \mathcal{R}^{(n)} L^{(n,0)} + s^{(n)} I \quad (3.2)$$

で与えられる. (3.1) では  $A^{(n)} - s^{(n)} I$  の  $LU$  分解が  $L^{(n,0)} \mathcal{R}^{(n)}$  で表され,  $L^{(n,0)}$  は  $L^{(n)}$  と同様に下副対角成分が全て 1 である下 2 重対角行列,  $\mathcal{R}^{(n)}$  は上帯幅  $M+1$  の上三角行列である. シフト付き  $LR$  変換 (3.1), (3.2) は

$$A^{(n+M)} = (L^{(n,0)})^{-1} A^{(n)} L^{(n,0)} \quad (3.3)$$

のように  $A^{(n)}$  から  $A^{(n+M)}$  への固有値保存変形とみなせる. 新たに  $L^{(n)}$  と同じように下副対角成分が全て 1 である下 2 重対角行列  $L^{(n,1)}, L^{(n,2)}, \dots, L^{(n,M)}$  を導入し,

$$R^{(n+k)} L^{(n,k)} = L^{(n,k+1)} R^{(n+M+k)}, \quad k = 0, 1, \dots, M-1 \quad (3.4)$$

を満たすとする. (3.4) は  $R^{(n+k)} L^{(n,k)}$  の  $LU$  分解が  $L^{(n,k+1)} R^{(n+M+k)}$  であることを意味する. このとき, 下 2 重対角行列  $L^{(n+M)}$  を

$$L^{(n+M)} := (L^{(n,0)})^{-1} L^{(n)} L^{(n,M)} \quad (3.5)$$

で定めると, (3.3) の右辺は (3.4) の繰り返しによって以下のように変形される.

$$\begin{aligned} (L^{(n,0)})^{-1} A^{(n)} L^{(n,0)} &= (L^{(n,0)})^{-1} L^{(n)} R^{(n+M-1)} R^{(n+M-2)} \dots R^{(n+1)} \underline{R^{(n)} L^{(n,0)}} \\ &= (L^{(n,0)})^{-1} L^{(n)} R^{(n+M-1)} R^{(n+M-2)} \dots \underline{R^{(n+1)} L^{(n,1)}} R^{(n+M)} \\ &= (L^{(n,0)})^{-1} L^{(n)} R^{(n+M-1)} R^{(n+M-2)} \dots L^{(n,2)} R^{(n+M+1)} R^{(n+M)} \\ &\quad \vdots \\ &= \underline{(L^{(n,0)})^{-1} L^{(n)} L^{(n,M)} R^{(n+2M-1)}} \dots R^{(n+M+1)} R^{(n+M)} \\ &= L^{(n+M)} R^{(n+2M-1)} \dots R^{(n+M+1)} R^{(n+M)}. \end{aligned} \quad (3.6)$$

もし  $L^{(n,0)}$  が求まれば,  $A^{(n)}$  から  $A^{(n+M)}$  が得られる変換は (3.4) の繰り返しと (3.5) で実現できることが分かる. (3.4) の対角成分と上副対角成分について成り立つ等式を書き下すと,

$$\begin{cases} Q_j^{(n,k+1)} = Q_j^{(n,k)} + E_j^{(n+k)} - E_{j-1}^{(n+M+k)}, & k = 0, 1, \dots, M-1, \quad j = 1, 2, \dots, m, \\ E_j^{(n+M+k)} = \frac{Q_{j+1}^{(n,k)}}{Q_j^{(n,k+1)}} E_j^{(n+k)}, & k = 0, 1, \dots, M-1, \quad j = 1, 2, \dots, m-1, \\ E_0^{(n+M+k)} := 0, \quad E_m^{(n+k)} := 0, & k = 0, 1, \dots, M-1 \end{cases} \quad (3.7)$$

が得られる. ここで,  $Q_j^{(n,k)}$  は下 2 重対角行列  $L^{(n,k)}$  の  $(j, j)$  成分を表す. (3.5) を変形した  $L^{(n,0)} L^{(n+M)} = L^{(n)} L^{(n,M)}$  の対角成分と下副対角成分に関する等式として,

$$\begin{cases} Q_j^{(n+M)} = \frac{Q_j^{(n)} Q_j^{(n,M)}}{Q_j^{(n,0)}}, & j = 1, 2, \dots, m, \\ Q_{j+1}^{(n,0)} = Q_j^{(n,M)} + Q_{j+1}^{(n)} - Q_j^{(n+M)}, & j = 1, 2, \dots, m-1. \end{cases} \quad (3.8)$$

が導かれる. よって,  $L^{(n,0)}$  が求まれば, (3.7), (3.8) によって (3.6) の変換が実現できる.

(3.1) における  $A^{(n)} - s^{(n)} I$  の  $LU$  分解を陽的に求めてもよいが, (3.7), (3.8) を実行するには,  $L^{(n,0)}$  の  $(1, 1)$  成分  $Q_1^{(n,0)}$  が分かれば十分であることに注意したい. (3.1) の  $(1, 1)$  成分に関する等式は  $Q_1^{(n)} - s^{(n)} = Q_1^{(n,0)}$  であり,  $A^{(n)}$  に含まれる  $Q_1^{(n)}$  とシフト量  $s^{(n)}$  の差で  $L^{(n,0)}$  の  $(1, 1)$  成分が与えられる. (3.7) の第 1 式において  $j = 1$  とすると,  $k = 0, 1, \dots, m-1$  の順で計算が進み,  $Q_1^{(n,1)}, Q_1^{(n,2)}, \dots, Q_1^{(n,M)}$  が得

---

**Algorithm 1** シフト付き dhToda<sub>II</sub> アルゴリズム
 

---

1: Input: $\{Q_j^{(0)}\}_{j=1,2,\dots,m}$ , $\{E_j^{(0)}, E_j^{(1)}, \dots, E_j^{(M-1)}\}_{j=1,2,\dots,m-1}$ 2: <b>for</b> $n = 0, M, 2M, \dots, n_{\max}$ <b>do</b> 3:   Choose $s^{(n)}$ s.t. $s^{(n)} < \lambda_m$ 4: $Q_1^{(n,0)} = Q_1^{(n)} - s^{(n)}$ 5: <b>for</b> $k = 0, 1, \dots, M-1$ <b>do</b> 6: $D_1^{(n,k)} = Q_1^{(n,k)}$ 7: $Q_1^{(n,k+1)} = D_1^{(n,k)} + E_1^{(n+k)}$ 8: <b>end for</b> 9: $Q_1^{(n+M)} = (Q_1^{(n,M)} / Q_1^{(n,0)}) Q_1^{(n)}$ 10: $F_1^{(n)} = Q_1^{(n,M)} - Q_1^{(n+M)}$ 11: $Q_2^{(n,0)} = F_1^{(n)} + Q_2^{(n)}$	12: <b>for</b> $j = 1, 2, \dots, m-1$ <b>do</b> 13: <b>for</b> $k = 0, 1, \dots, M-1$ <b>do</b> 14: $D_{j+1}^{(n,k)} = (Q_{j+1}^{(n,k)} / Q_j^{(n,k+1)}) D_j^{(n,k)}$ 15: $E_j^{(n+M+k)} = (Q_{j+1}^{(n,k)} / Q_j^{(n,k+1)}) E_j^{(n+k)}$ 16: $Q_{j+1}^{(n,k+1)} = D_{j+1}^{(n,k)} + E_{j+1}^{(n+k)}$ 17: <b>end for</b> 18: $Q_{j+1}^{(n+M)} = (Q_{j+1}^{(n,M)} / Q_{j+1}^{(n,0)}) Q_{j+1}^{(n)}$ 19: <b>if</b> $j \neq m-1$ <b>then</b> 20: $F_{j+1}^{(n)} = (Q_{j+1}^{(n,M)} / Q_{j+1}^{(n,0)}) F_j^{(n)}$ 21: $Q_{j+2}^{(n,0)} = F_{j+1}^{(n)} + Q_{j+2}^{(n)}$ 22: <b>end if</b> 23: <b>end for</b> 24: <b>end for</b>
---	---

---

られる。また, (3.8) において  $j = 1$  とすれば  $Q_1^{(n+M)}, Q_2^{(n,0)}$  が求まる。続いて, (3.7) の第 2 式において  $j = 1, k = 0$ , 第 1 式において  $j = 2$  とすれば  $E_1^{(n+M)}, Q_2^{(n,1)}, E_1^{(n+M+1)}, Q_2^{(n,2)}, \dots, E_1^{(n+2M-1)}, Q_2^{(n,M)}$  の順に, その後 (3.8) において  $j = 2$  とすれば,  $Q_2^{(n+M)}, Q_3^{(n,0)}$  が求まる。同様の手順を全ての  $j$  に対して行えば,  $A^{(n+M)}$  を構成する  $L^{(n+M)}, R^{(n+M)}, R^{(n+M+1)}, \dots, R^{(n+2M-1)}$  の成分が全て得られる。

新たな変数として  $D_j^{(n,k)}, F_j^{(n)}$  を

$$\begin{aligned} D_1^{(n,k)} &:= Q_1^{(n,k)}, \\ D_j^{(n,k)} &:= Q_j^{(n,k)} - E_{j-1}^{(n+M+k)}, \quad j = 2, 3, \dots, m, \\ F_j^{(n)} &:= Q_j^{(n,M)} - Q_j^{(n+M)}, \quad j = 1, 2, \dots, m \end{aligned}$$

と定めれば, 減算が現れないように (3.7) と (3.8) の第 2 式を変形できる。

以上より, シフトを導入した dhToda<sub>II</sub> アルゴリズムは Algorithm 1 で与えられる。なお, Algorithm 1 に含まれる  $n_{\max}$  は  $n$  の上限値である。

TN 行列  $A^{(0)}$  の最小固有値を  $\lambda_{\min}$  とすると, シフト付き dhToda<sub>II</sub> アルゴリズムの収束性について以下の定理が得られる。

**定理 3.1** (シフト付き dhToda<sub>II</sub> アルゴリズムの収束性)

$\ell = 0, 1, \dots$  に対して  $s^{(\ell M)} < \lambda_{\min}$  ならば,

$$\lim_{\ell \rightarrow \infty} Q_j^{(\ell M)} = \lambda_j, \quad j = 1, 2, \dots, m, \quad (3.9)$$

$$\lim_{n \rightarrow \infty} E_j^{(n)} = 0, \quad j = 1, 2, \dots, m-1. \quad (3.10)$$

シフト量  $s^{(n)}$  を  $\lambda_{\min}$  未満となるように定めれば, シフト付き dhToda<sub>II</sub> アルゴリズムは破綻なく実行でき, シフトなし dhToda<sub>II</sub> アルゴリズムよりも速い収束が期待できる。

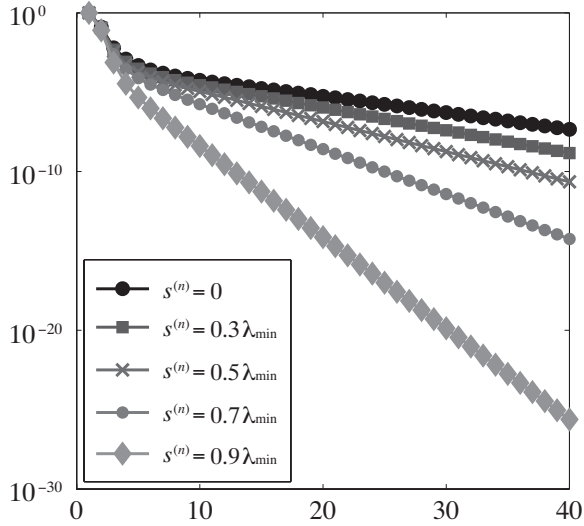


図1  $n = 3, 4, \dots, 40$  に対する  $E_{49}^{(n)}$  の値 (横軸:  $n$  の値, 縦軸:  $E_{49}^{(n)}$  の値)

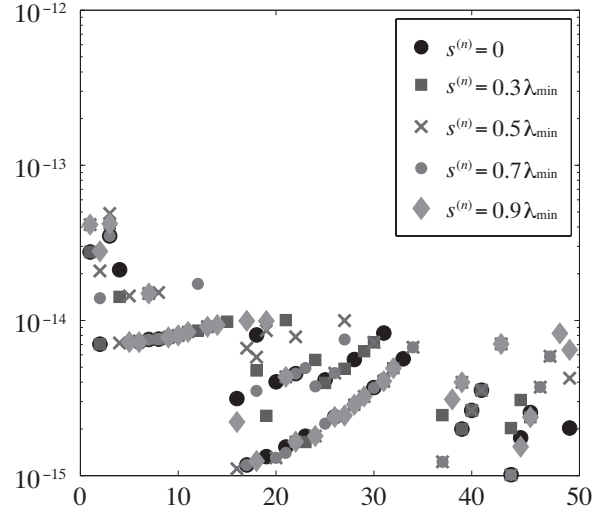


図2 求められた固有値に含まれる相対誤差 (横軸: 固有値を降順に並べたときの固有値番号, 縦軸: 相対誤差)

## 4 数値実験

本節では、シフト付き dhToda<sub>II</sub> アルゴリズムの挙動と求められた固有値の精度に関する数値例を示す。計算機環境は CPU: Intel Core-i5 2.60GHz, RAM: 8GB であり、数値計算ソフトウェア Matlab R2012a を使用した。テスト行列は以下の 50 次の TN 行列とした。

$$A^{(0)} = L^{(0)} R^{(3)} R^{(2)} R^{(1)} R^{(0)}, \quad (4.1)$$

$$L^{(0)} = \begin{pmatrix} 2 & & & \\ 1 & 2 & & \\ & \ddots & \ddots & \\ & & 1 & 2 \end{pmatrix}, \quad R^{(i)} = \begin{pmatrix} 1 & 1 & & \\ & 1 & \ddots & \\ & & \ddots & 1 \\ & & & 1 \end{pmatrix}, \quad i = 0, 1, 2, 3. \quad (4.2)$$

シフト付き dhToda<sub>II</sub> アルゴリズムの停止条件は  $\max_j \{E_j^{(n)}, E_j^{(n+1)}, E_j^{(n+2)}, E_j^{(n+3)}\} < 10^{-15}$  とした。TN 行列  $A^{(0)}$  の真の固有値として、数式処理ソフトウェア Mathematica 9.0 の `Eigenvalues[]` を用いて 50 桁精度計算で得られた値を採用した。シフト量  $s^{(n)}$  がシフト付き dhToda<sub>II</sub> アルゴリズムの収束性に与える影響を観察するために、予め計算された真の最小固有値  $\lambda_{\min}$  に基づくシフト量として  $s^{(n)} = 0, 0.3\lambda_{\min}, 0.5\lambda_{\min}, 0.7\lambda_{\min}, 0.9\lambda_{\min}$  の 5 種類について実験した。図 1 は  $n$  が増えるにつれて  $E_{49}^{(n)}$  の値が 0 に近づく様子を示したグラフである。図 1 より、 $s^{(n)}$  を  $\lambda_{\min}$  に近い値に定めれば、シフト付き dhToda<sub>II</sub> の収束が加速されることが確認できる。図 2 はシフト付き dhToda<sub>II</sub> で計算された固有値と真の固有値の相対誤差をプロットしたグラフである。シフト量  $s^{(n)}$  の定め方による大きな差は見られず、シフト付き dhToda<sub>II</sub> アルゴリズムで求められた固有値は高精度であることが分かる。

## 5 まとめ

本稿では、まず  $\text{dhToda}_{\text{II}}$  に基づいて上 Hessenberg 型をした TN 行列  $A^{(n)}$  の固有値が求まる仕組みについて概説した。  $\text{dhToda}_{\text{II}}$  の時間発展を利用すると  $A^{(n)}$  から  $A^{(n+M)}$  への固有値保存変形を与えられ、  $n \rightarrow \infty$  において  $\text{dhToda}_{\text{II}}$  変数  $Q_j^{(n)}$  が  $A^{(n)}$  の固有値に収束することを示した。この  $n \rightarrow \infty$  における漸近挙動を利用した  $\text{dhToda}_{\text{II}}$  アルゴリズムは、  $\text{dhToda}_{\text{I}}$  に基づいて定式化された  $\text{dhToda}$  アルゴリズム [3] と同様、非対称な TN 行列の固有値を求めることができる。ただし、  $\text{dhToda}$  アルゴリズムでは下 Hessenberg 型の TN 行列が下 3 角化されるのに対して、  $\text{dhToda}_{\text{II}}$  アルゴリズムでは上 Hessenberg 型の TN 行列が下 2 重対角化される。続いて、  $\text{dhToda}_{\text{II}}$  アルゴリズムに対して収束加速のための原点シフトを導入し、そのシフト付き  $\text{dhToda}_{\text{II}}$  アルゴリズムの収束性も示した。シフト付き  $\text{dhToda}_{\text{II}}$  アルゴリズムにおける  $A^{(n)}$  から  $A^{(n+M)}$  への固有値保存変形では、上 Hessenberg 型の  $A^{(n)}$  に対する直接的な  $LU$  分解は不要であり、3 重対角行列に対する  $LR$  変換が中心となる。数値実験においては、最小固有値に近い値をシフト量に選ぶとシフト付き  $\text{dhToda}_{\text{II}}$  アルゴリズムの収束が加速されること、シフト量にほとんど関係なくシフト付き  $\text{dhToda}_{\text{II}}$  アルゴリズムは十分な精度で固有値が求められることを確認した。シフト量の適切な選び方については今後の課題である。

## 参考文献

- [1] T. Tokihiro, A. Nagai, J. Satsuma: Proof of solitonical nature of box and ball systems by means of inverse ultra-discretization, *Inverse Probl.*, 15 (1999), 1639–1662.
- [2] 柿崎 苑美, 福田 亜希子, 石渡 恵美子, 山本 有作, 岩崎 雅史, 中村 佳正: 箱に番号が付いた新しい箱玉系について, 九州大学応用力学研究所研究集会報告, 25A-S2 (2014), 121–126.
- [3] A. Fukuda, E. Ishiwata, Y. Yamamoto, M. Iwasaki, Y. Nakamura: Integrable discrete hungry systems and their related matrix eigenvalues, *Annal. Mat. Pura Appl.*, 192 (2013), 423–445.
- [4] A. Fukuda, Y. Yamamoto, M. Iwasaki, E. Ishiwata, Y. Nakamura: On a shifted  $LR$  transformation derived from the discrete hungry Toda equation, *Monat. Math.*, 170 (2013), 11–26.
- [5] Y. Yamamoto, T. Fukaya: Differential qd algorithm for totally nonnegative Hessenberg matrices: introduction of origin shifts and relationship with the discrete hungry Lotka-Volterra system, *JSIAM Letters*, 2 (2010), 69–72.