

Benchmarking numerical methods for lattice equations with the Toda lattice

Deniz Bilman^{a,*}, Thomas Trogdon^b

^a Department of Mathematics, University of Michigan, 530 Church Street, Ann Arbor, MI 48109, United States of America

^b University of California, Irvine, Rowland Hall, Irvine, CA 92697, United States of America

ARTICLE INFO

Article history:

Available online 27 October 2018

Keywords:

FPUT lattice

Toda lattice

Numerical analysis

Benchmarking numerical methods

Ordinary differential equations

Runge–Kutta methods

Symplectic integrators

ABSTRACT

We compare the performance of well-known numerical time-stepping methods that are widely used to compute solutions of the doubly-infinite Fermi–Pasta–Ulam–Tsingou (FPUT) lattice equations. The methods are benchmarked according to (1) their accuracy in capturing the soliton peaks and (2) in capturing highly-oscillatory parts of the solutions of the Toda lattice resulting from a variety of initial data. The numerical inverse scattering transform method is used to compute a reference solution with high accuracy. We find that benchmarking a numerical method on pure-soliton initial data can lead one to overestimate the accuracy of the method.

© 2018 IMACS. Published by Elsevier B.V. All rights reserved.

1. Introduction

Consider the classical problem of one-dimensional infinite chain of particles on a line with nearest-neighbor interactions as depicted in Fig. 1. Assume that each particle has unit mass, and that there are no impurities, i.e. the potential energies of the springs between the particles are identical. We let $V: \mathbb{R} \rightarrow \mathbb{R}$ denote the interaction potential between the neighboring particles. With these assumptions, the equations of motion that govern this particle system are given by Newton's Second Law of Motion:

$$\frac{d^2}{dt^2} q_n = V'(q_{n+1} - q_n) - V'(q_n - q_{n-1}), \quad n \in \mathbb{Z}, \quad (1)$$

where q_n stands for the displacement of the n th particle from its equilibrium position. Denoting by p_n the momentum of the n th particle, (1) is equivalent to the system of first-order differential equations:

$$\frac{dp_n}{dt} = V(q_{n+1} - q_n) - V(q_n - q_{n-1}), \quad \frac{dq_n}{dt} = p_n, \quad n \in \mathbb{Z}. \quad (2)$$

With the assumptions that $q_{n+1} - q_n \rightarrow 0$ and $p_n \rightarrow 0$ sufficiently fast as $|n| \rightarrow \infty$ (i.e. no motion at infinity), (2) is a Hamiltonian system of equations

$$\frac{dp_n}{dt} = -\frac{\partial \mathcal{H}(p, q)}{\partial q_n}, \quad \frac{dq_n}{dt} = \frac{\partial \mathcal{H}(p, q)}{\partial p_n}, \quad n \in \mathbb{Z}, \quad (3)$$

* Corresponding author.

E-mail addresses: bilman@umich.edu (D. Bilman), ttrogdon@uci.edu (T. Trogdon).

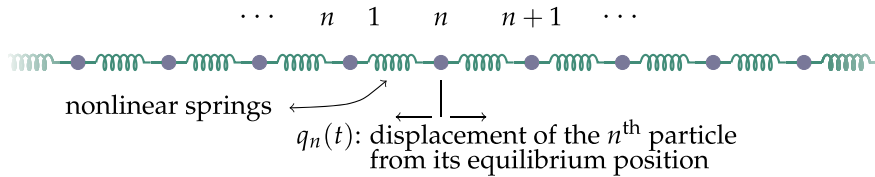


Fig. 1. One-dimensional chain of particles with nearest neighbor interactions.

with the Hamiltonian functional $\mathcal{H}(p, q)$:

$$\mathcal{H}(p, q) = \sum_{n \in \mathbb{Z}} \frac{1}{2} p_n^2 + V(q_{n+1} - q_n). \quad (4)$$

Such nearest-neighbor interacting particle systems with nonlinear interaction forces (anharmonic potentials) include the systems studied in the famous experiment at Los Alamos by Fermi, Pasta, Ulam, and Tsingou [9] in 1953, which a decade later led to discovery of solitons by Zabusky and Kruskal [34]. Such lattice equations model various physical phenomena with a multitude of applications [25]. From a purely mathematical perspective these lattices are used to investigate Poincaré recurrence, chaos, and nonlinear wave phenomena (interaction of solitary waves, solitary wave resolution, see [3] and the references therein). The FPUT-type systems (2) are still an active area of mathematical research [11,13,33].

The particular choice $V(r) = V_{\text{Toda}}(r) := e^{-r} + r - 1$ in (2) results in an infinite dimensional, continuous time, discrete space, completely integrable system: the celebrated Toda lattice [28]. The main purpose of this work is to compare the performance of various well known numerical time-stepping methods that are widely used to compute the solution of the Cauchy initial value problem for (1). The accuracy of a time-stepping method is most easily inspected when there are exact solutions available at hand, which is of course, rarely the case. Often, numerical analysts make use of nonlinear integrable wave models that possess classes of explicit solutions (e.g. the Toda lattice, the Korteweg–de Vries (KdV) equation or the nonlinear Schrödinger (NLS) equation). Then one has the luxury of being able to test their numerical scheme against exact formulae. Many of these infinite-dimensional integrable systems feature dispersive radiation and/or oscillatory tails. This is a critically important¹ oscillatory component of the solution that decays slowly to the background as $t \rightarrow \infty$. These exact solution formulae often give solitary waves such as breathers or solitons – coherent structures that are localized in space without oscillatory tails or dispersive radiation. In this paper, we set out to investigate questions such as: Which time-stepping methods capture the solitons with more accuracy? How do they perform when computing highly-oscillatory solutions? How does their performance depend on the solution itself? Our strategy is as follows. Using a numerical inverse scattering transform (IST) method [5], we can accurately construct solutions of the Toda lattice for each (n, t) without any time stepping for arbitrarily large values of t with high accuracy. Indeed, one can expect to maintain relative accuracy for large t [23]. This opens the door to benchmarking time-stepping methods on solutions with different characters (oscillatory, has solitons, no solitons, etc.), without restricting oneself to those with exact formulae. Furthermore, it is reasonable to expect behavior of a numerical method for the Toda lattice to extend to other FPUT-type systems.

From a convergence (as time-step size tends to 0) and computational complexity points of view, comparison of numerical time-stepping methods is a well-trodden path, see for example, [14] for a detailed study in this direction, or the more recent survey article [7] by Butcher and the references therein. Using the numerical IST method, in this work we aim to add a new dimension to such studies.

The list of applicable time-stepping methods is long. For example, there are modern lattice methods such as the Lattice-Boltzmann method that are used in computational fluid mechanics, specifically with applications to nanofluids. See, for example, [26,27]. Our focus is not to determine which modern method is best but rather to first benchmark classical higher-order methods and classical symplectic methods to demonstrate the utility and nuance of the solution captured by the numerical IST method.

The paper is organized as follows. In the remainder of this section we discuss specifics of the Toda lattice. In Section 2 we summarize the numerical IST procedure, give an overview of the numerical time-stepping methods used in this work and present the explicit formulae for the initial data used in the numerical experiments. In Section 4, we compare the performance of second order methods. In Section 5, we compare the performance of higher order methods. We then draw some conclusions.

1.1. Properties of the Toda lattice

Complete integrability of the Toda lattice was proved by H. Flaschka [10] and S. V. Manakov [20] in 1974, independently and simultaneously, by realizing that the system possesses a Lax pair and (2) with $V(r) = V_{\text{Toda}}(r)$ is in one-to-one correspondence with isospectral deformations of Jacobi matrices. Jacobi matrices are symmetric, tridiagonal matrices with positive off-diagonal elements. Indeed, through the bijection

¹ Generic initial data for the Toda lattice gives rise to dispersive radiation.

$$a_n := \frac{1}{2}e^{-(q_{n+1}-q_n)/2}, \quad b_n := -\frac{1}{2}p_n, \quad (5)$$

the equations of motion (1) Toda lattice take the form (see [10,20]):

$$\frac{d}{dt}a_n = a_n(b_{n+1} - b_n), \quad \frac{d}{dt}b_n = 2(a_n^2 - a_{n-1}^2), \quad n \in \mathbb{Z}. \quad (6)$$

Defining the following second-order linear difference operators $\mathbf{L}$ and $\mathbf{P}$ on the Hilbert space $\ell^2(\mathbb{Z})$ of square-summable sequences:

$$\begin{aligned} (\mathbf{L}\phi)_n &:= a_{n-1}\phi_{n-1} + b_n\phi_n + a_n\phi_{n+1}, \\ (\mathbf{P}\phi)_n &:= -a_{n-1}\phi_{n-1} + a_n\phi_{n+1}, \end{aligned} \quad (7)$$

it can be verified that (1) is equivalent to the Lax equation [18]

$$\frac{d}{dt}\mathbf{L} = [\mathbf{P}, \mathbf{L}] := \mathbf{P}\mathbf{L} - \mathbf{L}\mathbf{P}. \quad (8)$$

The operators $(\mathbf{P}, \mathbf{L})$ are called a Lax pair and in the standard basis, $\mathbf{L}$ is a doubly-infinite Jacobi matrix. The Lax pair (8) constitutes the foundation of the IST method to solve the Cauchy initial value problem for (6) for sufficiently decaying initial data (see, for example, [2] for a recent survey of the IST for the Toda lattice).

A numerical IST method was recently developed by the authors [5] for the doubly-infinite Toda lattice. Implementations for other integrable systems can be found in [23,29,30,32] and these are summarized in [31]. An implementation for the numerical IST method can be found at [4]. The method works, loosely speaking, by performing the following steps:

- (1) Compute the spectral data: This involves solving the eigenvalue problem $\mathbf{L}\varphi = \lambda\varphi$ for bounded eigenfunctions φ . Assuming $(a_n, b_n) \rightarrow (1/2, 0)$ at an appropriately rapid rate as $|n| \rightarrow \infty$, the spectrum consists of the interval $[-1, 1]$ and a finite number of simple eigenvalues in $\mathbb{R} \setminus [-1, 1]$. One also computes, a function $R(z)$, defined on the spectrum, which is directly related to the spectral measure for $\mathbf{L}$.
- (2) Solve the inverse problem: Once $R(z)$ is known, for each (n, t) there exists a contour $\Gamma = \Gamma(n, t) \subset \mathbb{C}$, a function $G(z; n, t)$, $G: \Gamma \rightarrow \mathbb{C}^{2 \times 2}$ and an integral equation

$$\left(I - \frac{G}{2}\right)U + \mathcal{H}_\Gamma(U)G = G - I, \quad U: \Gamma \rightarrow \mathbb{C}^{2 \times 2}.$$

Here $\mathcal{H}_\Gamma$ is the Hilbert transform over Γ and I is the identity matrix. This integral equation is solved for $U: \Gamma \rightarrow \mathbb{C}^{2 \times 2}$ using the framework of S. Olver [22] (see [21] for an implementation) and the solution of the Toda lattice can be obtained in terms of integrals of U . So, one can compute the map $(n, t) \mapsto U$ and hence $(n, t) \rightarrow (a_n(t), b_n(t))$ without time stepping.

2. Time-stepping methods

In this section we describe the time-stepping methods used to produce the results shown in this paper. The first three methods described below are classical second-order methods and the following methods are higher-order methods. All of the methods are used to numerically compute the solution $y(T) \in \mathbb{R}^N$ of the Cauchy initial value problem for an autonomous differential equation

$$\frac{dy}{dt} = f(y(t)), \quad (9)$$

with initial condition $y(0) = y^0 \in \mathbb{R}^N$ at a final time $t = T$. We focus on classical methods to illustrate how one can use the numerical IST method to benchmark them.

A time stepping method is given by an explicit function F , depending on the time step h , the function f and y^j for $k - \ell \leq j < k + 1$, $\ell \geq 0$, mapping into $\mathbb{R}^N$. The definition of F differs among different methods. The iteration is then given by

$$y^{k+1} = F(f, h, y^k, \dots, y^{k-\ell}),$$

and F is chosen so that y^k gives an approximation to $y(t)$ at time $t = kh$. We truncate the spatial domain (the lattice $\mathbb{Z}$) to $\mathbb{S} = \{-K, \dots, 0, \dots, K\}$, with the appropriate boundary conditions for the variables (a, b) in (6) or (p, q) in (2). Then $K > 0$ is chosen sufficiently large so that the $y(T)$ does not feel the effect of the boundary. Thus, $N = 2K + 1$.

2.1. Nomenclature and abbreviations

In the material that follows, $h > 0$ and $dt > 0$ both to denote the time-step size used in a numerical scheme. We use superscripts to denote the numerical iterates to avoid confusion with the indices of sequences, e.g., y^k denotes a numerically computed solution at the k -th time-step. Given a computed solution y^k at a time t , we denote the computed solution at time $t + h$ by y^{k+1} . Such superscript indices k are therefore nonnegative integers, except for when auxiliary half-step computations are involved in algorithms, in which case $y^{k+1/2}$ is used. Since we treat lattice equations, y^k will denote a finite sequence of real numbers, y_n^k is a scalar, the n -th element of the sequence $y^k \in \mathbb{R}^N$. We reserve the variables a_n , b_n and p_n , q_n to refer to the solution of the Toda lattice equations given in (2) and (6), respectively. We have the following abbreviations for the numerical methods used

- Midpoint method: `midpoint`.
Unless otherwise stated for this method, we integrate (6). When we integrate (2) instead we use the label `midpointqp`.
- Second-order Störmer–Verlet method: `sv2symp`.
- Fourth-order Adams–Bashforth method: `ab4`.
- Fourth-order Runge–Kutta method: `rk4`.
Unless otherwise stated for this method, we integrate (6). When we integrate (2) we use the label `rk4qp`.
- Four–Five-order Runge–Kutta–Fehlberg method: `rkf45`.

We remind the reader of these abbreviations when the corresponding methods are described below. The abbreviations `NoS`, `PureS`, `double`, `quad`, `dirac`, which we use for the initial data considered in this work are given in parentheses next to the corresponding titles in Section 3. The acronym “ID” is used to abbreviate “initial data”.

We proceed with brief descriptions of the numerical algorithms considered in this work.

2.2. Second-order methods

2.2.1. Midpoint (`midpoint` and `midpointqp`)

Given y^k , $k \in \mathbb{N}$, the algorithm to compute y^{k+1} is as follows:

$$y^{k+1} = y^k + hf \left(y^k + \frac{1}{2}hf(y^k) \right).$$

2.2.2. Second-order Störmer–Verlet (`sv2symp`)

The Störmer–Verlet method is symplectic: It preserves the Hamiltonian (4) under exact arithmetic. We use the equations of motions (1) and set

$$f_p(q) := \left(e^{-(q_n - q_{n-1})} - e^{-(q_{n+1} - q_n)} \right)_{n \in \mathbb{Z}}, \quad f_q(p) := (p_n)_{n \in \mathbb{Z}},$$

to denote the right hand sides. Since the Hamiltonian for the Toda lattice is separable, the method becomes explicit. Given (p^k, q^k) , the algorithm to compute (p^{k+1}, q^{k+1}) is as follows:

$$\begin{aligned} p^{k+1/2} &= p^k + \frac{1}{2}hf_p(q^k), \\ q^{k+1} &= q^k + hf_q(p^{k+1/2}), \\ p^{k+1} &= p^{k+1/2} + \frac{1}{2}hf_p(q^{k+1}). \end{aligned}$$

2.3. Fourth-order methods

2.3.1. Fourth-order Adams–Bashforth (`ab4`)

The Adams–Bashforth method is a linear explicit multi-step method. To compute the first three iterates we perform lower-step Adams–Bashforth methods successively:

$$\begin{aligned} y^0 &= y^0, \\ y^1 &= y^0 + hf(y^0), \\ y^2 &= y^1 + \frac{1}{2}h(-f(y^0) + 3f(y^1)), \\ y^3 &= y^2 + \frac{1}{12}h(5f(y^0) - 16f(y^1) + 23f(y^2)). \end{aligned}$$

Then, given computed solutions $y^k, y^{k-1}, y^{k-2}, y^{k-3}$, $k \geq 3$ the algorithm to compute y^{k+1} is as follows:

$$y^{k+1} = y^k + \frac{1}{24}h \left(-9f(y^{k-3}) + 37f(y^{k-2}) - 59f(y^{k-1}) + 55f(y^k) \right).$$

2.3.2. Fourth-order Runge–Kutta (rk4 and rk4qp)

We denote this method by rk4. The Runge–Kutta method is an explicit single-step method. Given the computed solution y^k , the algorithm [17,24] to compute y^{k+1} is as follows:

$$\begin{aligned} s_1 &= hf(y^k), \\ s_2 &= hf\left(y^0 + \frac{1}{2}s_1\right), \\ s_3 &= hf\left(y^0 + \frac{1}{2}s_2\right), \\ s_4 &= hf\left(y^0 + s_3\right), \\ y^{k+1} &= y^k + \frac{1}{6}h(s_1 + 2s_2 + 2s_3 + s_4). \end{aligned}$$

2.3.3. Four–five-order Runge–Kutta–Fehlberg (rkf45)

This method is fourth order method but with only a single extra computation at each step the local error can be controlled by a fifth order method. This feature [8] is very practical for implementing adaptive-step-size methods. We do not implement any adaptive methods. Given the computed solution y^k , the algorithm [8,12] to compute y^{k+1} is as follows:

$$\begin{aligned} s_1 &= hf(y), \\ s_2 &= f(y + h\alpha_{21}s_1), \\ s_3 &= f(y + h(\alpha_{31}s_1 + \alpha_{32}s_2)), \\ s_4 &= f(y + h(\alpha_{41}s_1 + \alpha_{42}s_2 + \alpha_{43}s_3)), \\ s_5 &= f(y + h(\alpha_{51}s_1 + \alpha_{52}s_2 + \alpha_{53}s_3 + \alpha_{54}s_4)), \\ s_6 &= f(y + h(\alpha_{61}s_1 + \alpha_{62}s_2 + \alpha_{63}s_3 + \alpha_{64}s_4 + \alpha_{65}s_5)), \\ y^{k+1} &= y^k + h(\beta_1s_1 + \beta_2s_2 + \beta_3s_3 + \beta_4s_4 + \beta_5s_5 + \beta_6s_6). \end{aligned}$$

For the constants α_{ij} and β_j , see the Butcher tableau provided in the Appendix A. See [6] for a survey article by Butcher on Runge–Kutta methods.

Remark 2.1. For a detailed study of these methods and more, the reader may consult to textbooks [12] or [19], for example.

3. Initial data considered in numerical experiments

We now present the types of initial data (ID) used in the numerical experiments underlying the results presented in this paper.

3.1. Purely dispersive (NoS)

We chose ID that gives rise to pure radiation, that is, the discrete spectrum of $\mathbf{L}$ is empty and consequently there are no solitons. The ID used is given by

$$a_n = \frac{1}{2} - \frac{1}{4}e^{-n^2}, \quad b_n = \frac{1}{10} \operatorname{sech}(n), \quad n \in \mathbb{Z}. \quad (10)$$

The solution at $t = 1000$ is shown in Fig. 2.

3.2. Pure 1-soliton (PureS)

We chose a pure 1-soliton solution of the Toda lattice at time $t = 0$. This ID is given by

$$\begin{aligned} a_n &= 1 - \frac{1}{2} \frac{\sqrt{(1 + e^{-2\kappa(n-1)})(1 + e^{-2\kappa(n+1)})}}{1 + e^{-2\kappa n}}, \\ b_n &= \frac{e^{-\kappa} - e^{\kappa}}{2} \left(\frac{e^{-2\kappa n}}{1 + e^{-2\kappa n}} - \frac{e^{-2\kappa(n-1)}}{1 + e^{-2\kappa(n-1)}} \right), \quad \kappa = 0.4, \quad n \in \mathbb{Z}. \end{aligned} \quad (11)$$

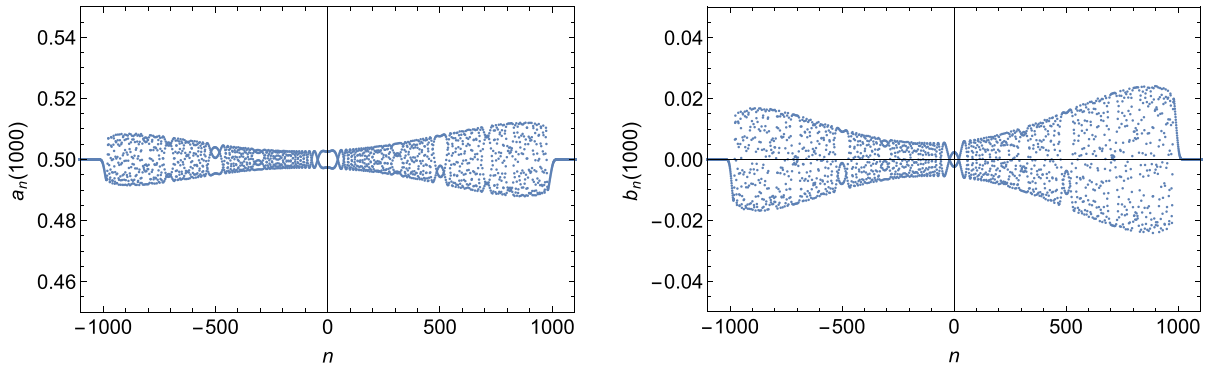


Fig. 2. The solution of the Toda lattice with NoS ID at $t = 1000$.

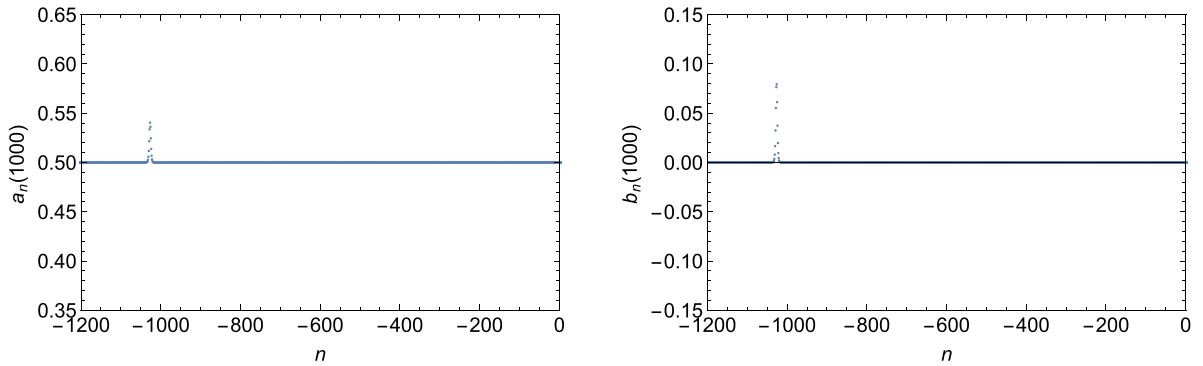


Fig. 3. The solution of the Toda lattice with PureS ID at $t = 1000$.

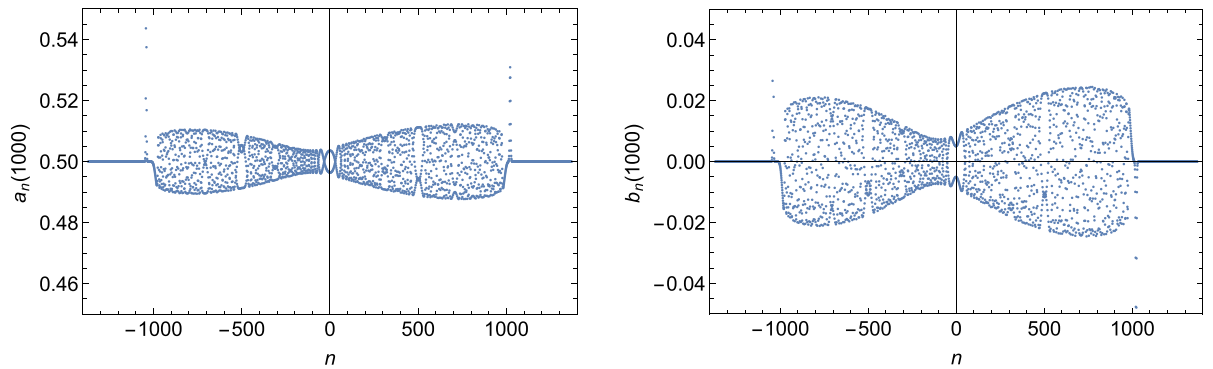


Fig. 4. The solution of the Toda lattice with double ID at $t = 1000$.

The solution at $t = 1000$ is shown in Fig. 3. Notice that this is the only solution we consider without dispersive radiation.

3.3. 2 solitons (double)

A choice of ID that gives rise to 2 solitons and radiation is

$$a_n = \frac{1}{2} + \frac{4}{5}ne^{-n^2}, \quad b_n = \frac{1}{10} \operatorname{sech}(n), \quad n \in \mathbb{Z}. \quad (12)$$

The solution at $t = 1000$ is shown in Fig. 4.

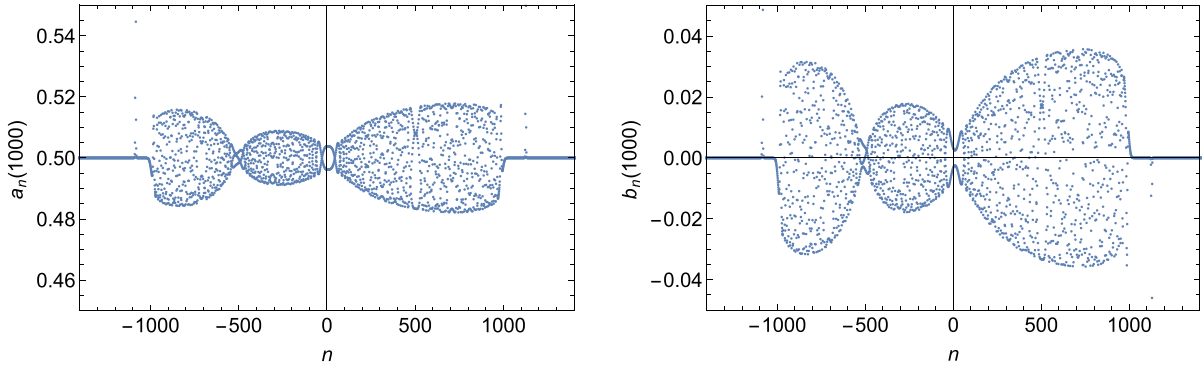


Fig. 5. The solution of the Toda lattice with `quad` ID at $t = 1000$.

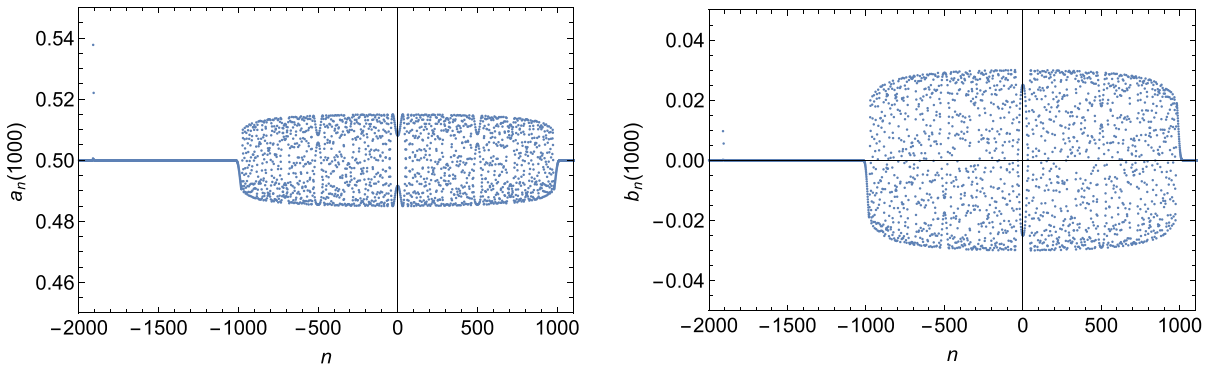


Fig. 6. The solution of the Toda lattice with `dirac` ID at $t = 1000$.

3.4. 4 solitons (`quad`)

ID that gives rise to 4 solitons and radiation is given by

$$a_n = \left| \frac{1}{2} - ne^{-n^2+n} \right|, \quad b_n = n \operatorname{sech}(n), \quad n \in \mathbb{Z}. \quad (13)$$

The solution at $t = 1000$ is shown in Fig. 5.

3.5. Dirac δ -type (`dirac`)

Dirac- δ -type ID that leads to 1 soliton and a radiating tail that is highly oscillatory is simply given by

$$a_n = \frac{1}{2}, \quad b_n = \begin{cases} 4, & n = 0 \\ 0, & \text{otherwise} \end{cases}, \quad n \in \mathbb{Z}. \quad (14)$$

The solution at $t = 1000$ is shown in Fig. 6.

4. Comparison of second-order methods

In order to compare our methods we take the following approach. We choose three time steps $dT = 0.01, 0.001, 0.0001$, three final times $T = 1000, 2000, 5000$, and two regions (dispersive and soliton regions, for only $n < 0$) and examine the relative errors made in approximating the solution of the Toda lattice at the final times for every choice of ID and every choice of time-stepping method. In accordance with the asymptotic analysis [5,15,16], the soliton region is effectively $\mathbb{Z} \setminus [-T, T]$. And so, the soliton region for this work is $[-(s+100)T, -T]$ where s is the speed of the fastest moving soliton. This can be computed from the spectrum of $\mathbf{L}$ [5]. The dispersive region is $[-cT, cT]$ for $0 < c < 1$. So, we fix our dispersive region as $[-T/2 - 50, -T/2 + 50]$. We always take the solution computed with the numerical IST method to be our “true” solution. A significant benefit of the numerical IST method is that the solution at each (n, t) can be

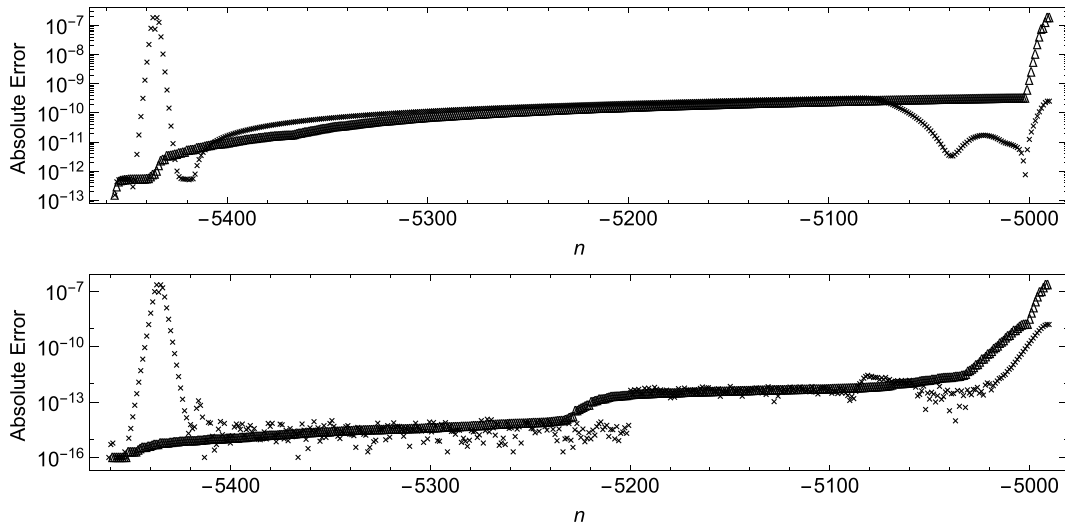


Fig. 7. Absolute errors in the soliton region, for quad ID, $dT = 0.0001$ with the ab4 (top) and sv2symp (bottom) methods. The $\times$ symbols represent the actual absolute errors as a function of n and the Δ symbols are the errors sorted in increasing order.

computed independently of all others. Thus to compute the “true” solution we only need to compute on the intervals $[-(s + 100)T, -T]$ and $[-T/2 - 50, -T/2 + 50]$ and we can refrain from computing the entire solution profile.

To define the relative error measure we use, consider the plot of errors in the soliton region, for quad ID, $dT = 0.0001$ with the ab4 and sv2symp methods as shown in Fig. 7. If one uses an ∞ -norm (or max norm) measurement (see the $\times$ symbols in Fig. 7), one might conclude that the errors are comparable. What this fails to account for is that the sv2symp method has fewer errors that are near the maximum error when compared with the ab4 method. To account for this we define the *sorted norm* on $\mathbb{R}^n$ for $0 < d < 1$ by

$$\|x\|_{\text{sort},d} = \|(y_1, y_2, \dots, y_{\lceil dn \rceil})^T\|_2, \quad y = \text{sort}(|x|).$$

Here the function $\text{sort}(\cdot)$ sorts the positive vector $|x|$ in decreasing order, and the sorted norm takes the ℓ^2 norm of the largest $\sim dn$ entries of the vector $|x|$. In all our computation we take $d = 0.1$, taking 10% of the entries. This is a hybrid of the ℓ^2 -norm and the ∞ -norm. Then the error of a vector x relative to a vector y with background c is given by

$$\text{rel}_{y,c}(x) = \frac{\|x - y\|_{\text{sort},d}}{\|c - y\|_{\text{sort},d}}. \quad (15)$$

We introduce the background c because for fixed n , $a_n(t) \rightarrow 1/2$ as $t \rightarrow \infty$ while $b_n(t) \rightarrow 0$ as $t \rightarrow \infty$. We want to approximate $a_n(t) - 1/2$ and $b_n(t)$.

Remark 4.1. For PureS ID in the dispersive region, the solution is, to machine precision, zero. A relative error metric here does not make sense and we use absolute error $\|x - y\|_{\text{sort},s}$.

4.1. Soliton region

We first consider the errors $\text{rel}_{y,1/2}(x)$ made in the approximation of the solution $a_n(T)$ of the Toda lattice in the soliton region $[-(s + 100)T, -T]$ at time T . Here y is chosen to be the reference solution obtained by the numerical IST method. In each panel of Fig. 8 and Fig. 9 we plot the relative error of the computed solution plotted versus dT , at $T = 1000$ and at $T = 5000$, respectively. In all panels sv2symp outperforms the other methods.

4.2. Dispersive region

We now consider the errors made in the approximation of the solution $a_n(T)$ of the Toda lattice in the dispersive region $[-T/2 - 50, -T/2 + 50]$ at time T . We can make an important point with Figs. 10 and 11. If one could only work with the pure soliton solution (right panel in the figures), that person might conclude that the midpointqp for small enough time step performs as well as sv2symp away from the soliton. This is true for the PureS ID, but not for the other ID. This illustrates why having accurate solutions with dispersive tails to compare against is important.

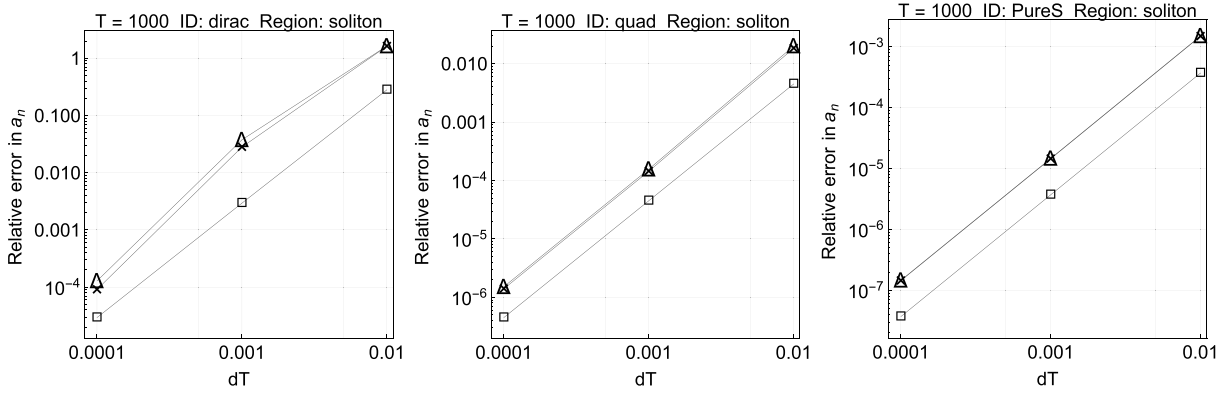


Fig. 8. Relative errors in the soliton region for the second-order time-stepping methods (midpoint ($\times$), midpointqp (Δ), sv2symp ($\square$)) at $T = 1000$ plotted versus dT for three choices of time step for three different choices of ID.

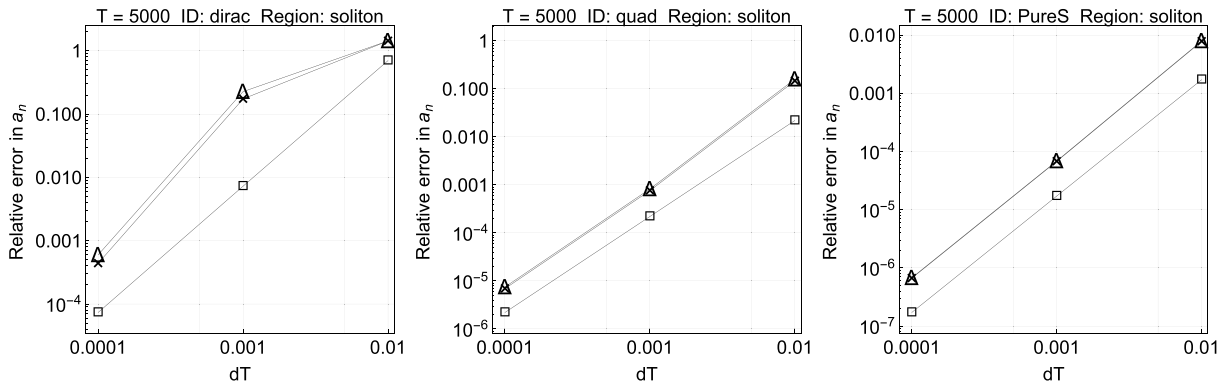


Fig. 9. Relative errors in the soliton region for the second-order time-stepping methods (midpoint ($\times$), midpointqp (Δ), sv2symp ($\square$)) at $T = 5000$ plotted versus dT for three choices of time step for three different choices of ID.

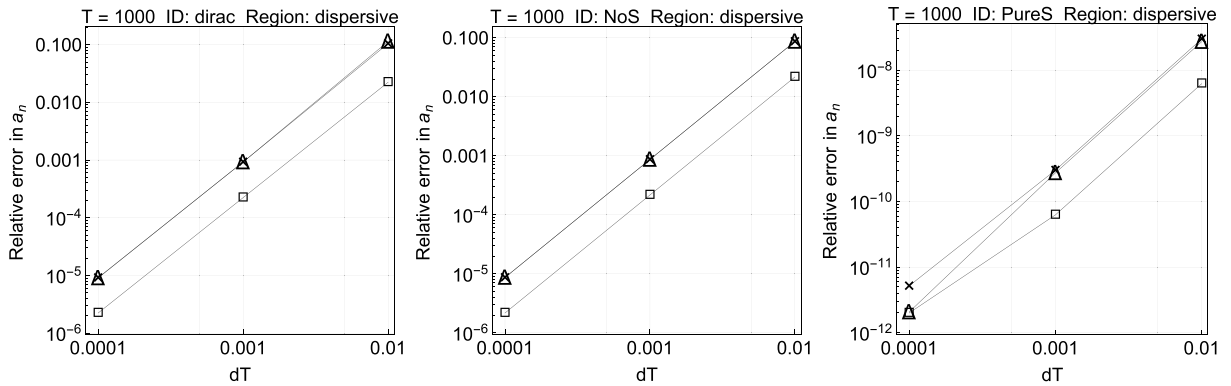


Fig. 10. Relative errors in the dispersive region for the second-order time-stepping methods (midpoint ($\times$), midpointqp (Δ), sv2symp ($\square$)) at $T = 1000$ plotted versus dT for three choices of time step for three different choices of ID.

5. Comparison of fourth-order methods

We now move to the comparison of the fourth-order methods listed above. We use the same relative error metric $\text{rel}_{y,1/2}(x)$ as described in (15), where the reference solution y is computed with the numerical IST method.

5.1. Soliton region

In each of the panels of Figs. 12 and 13 we plot the relative error of the computed approximation of $a_n(T)$ plotted versus dT for fourth-order methods in the soliton region. We can see that we are operating near the maximum accuracy of these

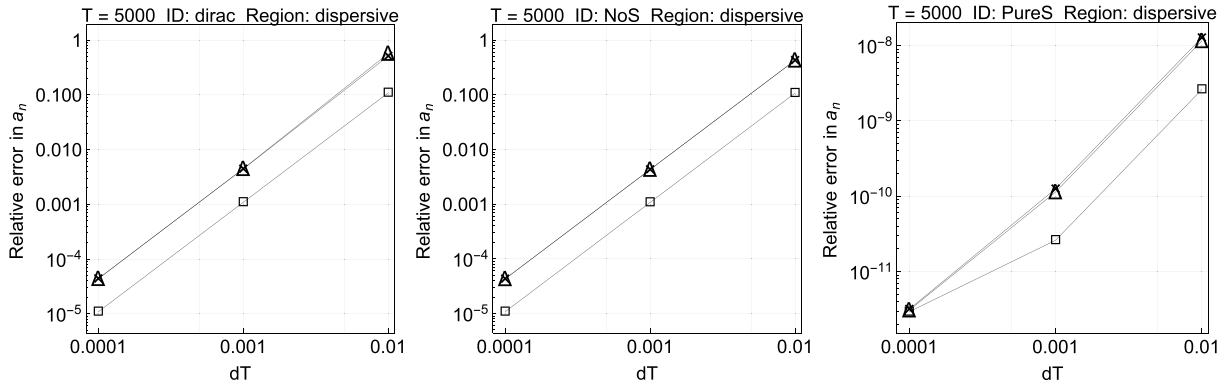


Fig. 11. Relative errors in the dispersive region for the second-order time-stepping methods (midpoint ($\times$), midpointqp (Δ), sv2symp ($\square$)) at $T = 5000$ plotted versus dT for three choices of time step for three different choices of ID.

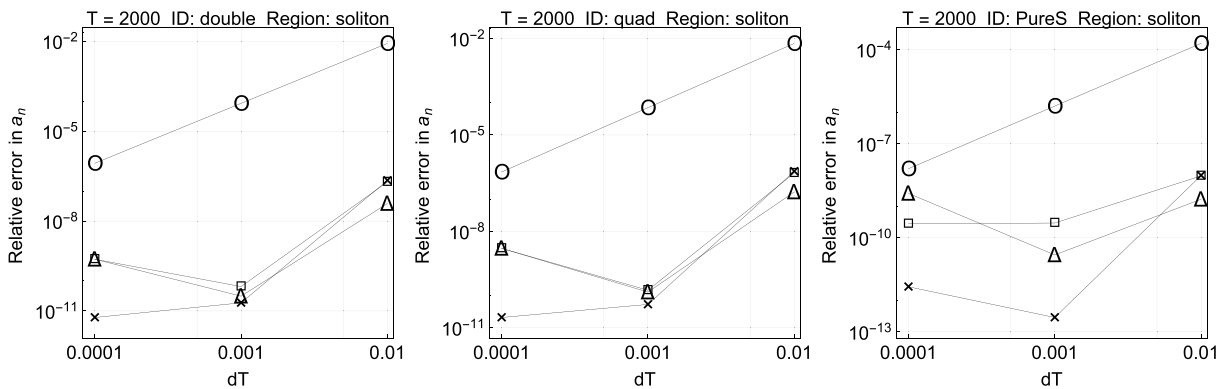


Fig. 12. Relative errors in the soliton region for the fourth-order time-stepping methods (rk4 ($\square$), rk4qp ($\times$), rkf45 (Δ), ab4 ($\circ$)) at $T = 2000$ plotted versus dT for three choices of time step for three different choices of ID.

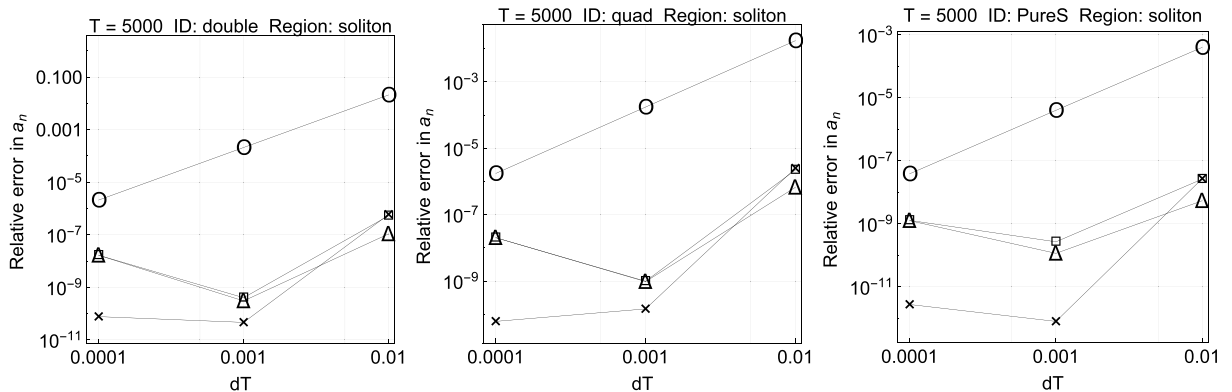


Fig. 13. Relative errors in the soliton region for the fourth-order time-stepping methods (rk4 ($\square$), rk4qp ($\times$), rkf45 (Δ), ab4 ($\circ$)) at $T = 5000$ plotted versus dT for three choices of time step for three different choices of ID.

methods as the relative error can increase as dT decreases. We see that ab4 under-performs and rk4qp is almost always the method of choice in this region. But it is important to note that the relative error encountered for the PureS ID is less than that encountered for the other ID. This points again to the need for a wide class of test solutions.

5.2. Dispersive region

In each of the panels of Figs. 14 and 15 we plot the relative error of the computed approximation of $a_n(T)$ plotted versus dT for fourth-order method in the dispersive region. We can again see that we are operating near the maximum accuracy. We see that ab4 under-performs but not as severely as in the soliton region and rk4qp is still almost always the method

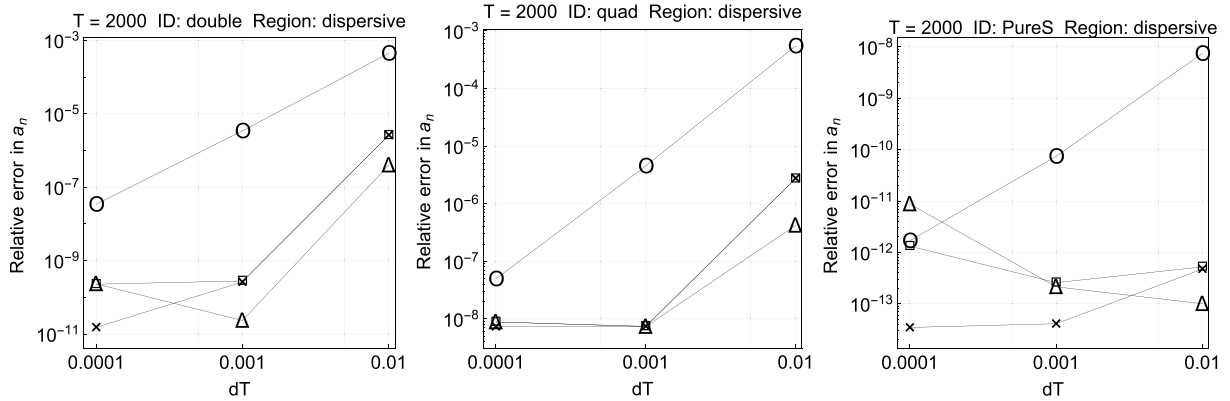


Fig. 14. Relative errors in the dispersive region for the fourth-order time-stepping methods ($rk4$ (□), $rk4qp$ (×), $rkf45$ (Δ), $ab4$ (○)) at $T = 2000$ plotted versus dT for three choices of time step for three different choices of ID.

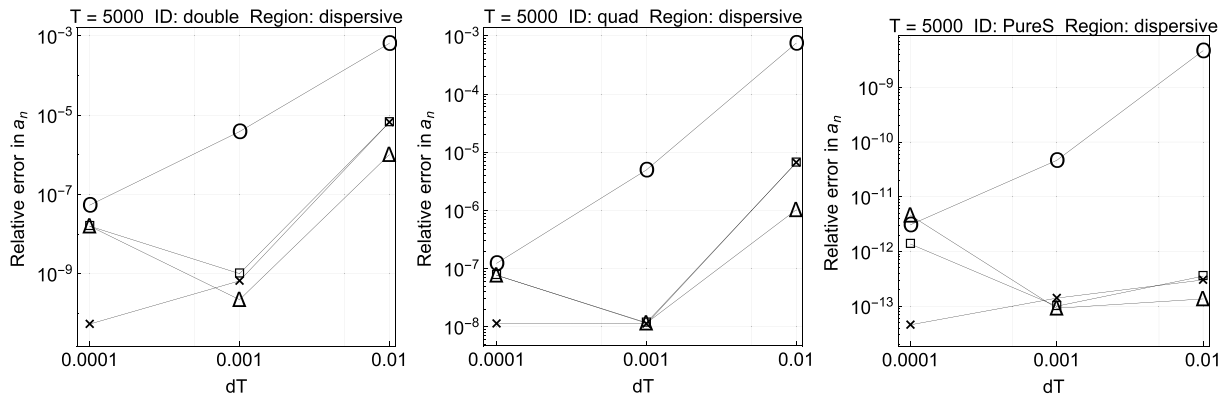


Fig. 15. Relative errors in the dispersive region for the fourth-order time-stepping methods ($rk4$ (□), $rk4qp$ (×), $rkf45$ (Δ), $ab4$ (○)) at $T = 5000$ plotted versus dT for three choices of time step for three different choices of ID.

of choice in this region, at least for small time steps. Again, the PureS ID gives smaller errors (recall we can only measure the absolute error for PureS ID in the dispersive region) than the other choices of ID, illustrating the importance of being able to compute these solutions accurately with the numerical IST method.

6. Conclusions

We have used the numerical IST method outlined in [5] to benchmark classical time-stepping routines on the Toda lattice. It is reasonable, especially following the perturbation work in [3], that such benchmarks will hold for other lattice equations – not just the Toda lattice and not just integrable lattices. The method $rk4qp$ appears to be the method of choice while the symplectic method $sv2symp$ performs surprisingly well given that it is only second-order accurate. We see that the preservation of the Hamiltonian is physically and numerically a good idea.

We have illustrated that having a wider class of methods allows one to detect deeper differences in methods, and analyze the accumulation of round-off error while getting a handle on the maximum accuracy of a method. We have also shown that benchmarking a method on a pure soliton initial condition can lead one to overestimate the maximum accuracy of the method. As we consider nonlinear lattices, both the lattice equation and the choice of ID matter in the performance of a given method. But this is not unexpected. Generic solutions have dispersive radiation with energy propagating rapidly in both directions, giving much more complicated dynamics, something the resolution of the method must take into account.

In Appendix B we give a complete listing of relative errors for both a_n and b_n . In [1] we have made available our data for the reference solutions computed with the numerical IST to allow others to benchmark their time-stepping routines.

Acknowledgement

D.B. is thankful for partial support from the National Science Foundation through the grant NSF-DMS-1513054.

Appendix A. Butcher tableau for Fehlberg's RKF45

We provide the Butcher tableau for the constants used in the `rkf45` method in this paper [8].

j	1	2	3	4	5	6
α_{2j}	$\frac{1}{4}$					
α_{3j}	$\frac{3}{32}$	$\frac{9}{32}$				
α_{4j}	$\frac{1932}{2197}$	$-\frac{7200}{2197}$	$\frac{7296}{2197}$			
α_{5j}	$\frac{439}{216}$	-8	$\frac{3680}{513}$	$-\frac{845}{4104}$		
α_{6j}	$-\frac{8}{27}$	2	$-\frac{3544}{2565}$	$\frac{1859}{4104}$	$-\frac{11}{40}$	
β_j	$\frac{25}{216}$	0	$\frac{1408}{2565}$	$\frac{2197}{4104}$	$-\frac{1}{5}$	0

Appendix B. Data for the methods

In the tables below, we display the data for the methods discussed in the body of the paper. In each table we display the performance of 3–4 methods, run until $T = 1000, 2000, 3000$ each with three different time steps. Each table is associated to a choice of initial data (`dirac`, `double`, `NoS`, `PureS`, `quad`) and a region (`soliton region = Sol.`, `dispersive region = Disp.`).

B.1. Second order – errors for $a_n(t)$

quad/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	4.499×10^{-3}	4.499×10^{-5}	4.499×10^{-7}	8.914×10^{-3}	8.912×10^{-5}	8.912×10^{-7}	2.195×10^{-2}	2.196×10^{-4}	2.197×10^{-6}
midpoint	1.834×10^{-2}	1.441×10^{-4}	1.402×10^{-6}	4.222×10^{-2}	2.913×10^{-4}	2.782×10^{-6}	1.463×10^{-1}	7.608×10^{-4}	6.898×10^{-6}
midpointqp	2.004×10^{-2}	1.561×10^{-4}	1.516×10^{-6}	4.608×10^{-2}	3.153×10^{-4}	3.008×10^{-6}	1.593×10^{-1}	8.234×10^{-4}	7.456×10^{-6}

quad/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	2.218×10^{-2}	2.217×10^{-4}	2.218×10^{-6}	4.43×10^{-2}	4.429×10^{-4}	4.428×10^{-6}	1.084×10^{-1}	1.084×10^{-3}	1.084×10^{-5}
midpoint	8.916×10^{-2}	8.891×10^{-4}	8.891×10^{-6}	1.781×10^{-1}	1.776×10^{-3}	1.775×10^{-5}	4.347×10^{-1}	4.341×10^{-3}	4.333×10^{-5}
midpointqp	8.954×10^{-2}	8.896×10^{-4}	8.892×10^{-6}	1.788×10^{-1}	1.777×10^{-3}	1.776×10^{-5}	4.364×10^{-1}	4.343×10^{-3}	4.34×10^{-5}

dirac/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	2.805×10^{-1}	2.937×10^{-3}	2.938×10^{-5}	3.859×10^{-1}	2.729×10^{-3}	2.715×10^{-5}	6.994×10^{-1}	7.258×10^{-3}	7.347×10^{-5}
midpoint	1.622	2.808×10^{-2}	9.098×10^{-5}	1.451	4.956×10^{-2}	1.036×10^{-4}	1.428	1.734×10^{-1}	4.393×10^{-4}
midpointqp	1.634	3.753×10^{-2}	1.285×10^{-4}	1.322	6.718×10^{-2}	1.443×10^{-4}	1.43	2.261×10^{-1}	5.969×10^{-4}

dirac/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	2.229×10^{-2}	2.23×10^{-4}	2.23×10^{-6}	4.381×10^{-2}	4.383×10^{-4}	4.383×10^{-6}	1.091×10^{-1}	1.089×10^{-3}	1.089×10^{-5}
midpoint	1.044×10^{-1}	9.254×10^{-4}	9.138×10^{-6}	2.04×10^{-1}	1.801×10^{-3}	1.778×10^{-5}	5.126×10^{-1}	4.444×10^{-3}	4.386×10^{-5}
midpointqp	1.137×10^{-1}	9.241×10^{-4}	9.033×10^{-6}	2.242×10^{-1}	1.809×10^{-3}	1.768×10^{-5}	5.709×10^{-1}	4.481×10^{-3}	4.378×10^{-5}

double/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	1.724×10^{-3}	1.723×10^{-5}	1.723×10^{-7}	3.388×10^{-3}	3.387×10^{-5}	3.387×10^{-7}	8.378×10^{-3}	8.377×10^{-5}	8.378×10^{-7}
midpoint	7.759×10^{-3}	7.098×10^{-5}	7.03×10^{-7}	1.588×10^{-2}	1.404×10^{-4}	1.385×10^{-6}	4.366×10^{-2}	3.52×10^{-4}	3.434×10^{-6}
midpointqp	8.85×10^{-3}	8.088×10^{-5}	8.013×10^{-7}	1.804×10^{-2}	1.598×10^{-4}	1.577×10^{-6}	4.919×10^{-2}	3.999×10^{-4}	3.907×10^{-6}

double/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	2.193×10^{-2}	2.193×10^{-4}	2.193×10^{-6}	4.328×10^{-2}	4.329×10^{-4}	4.329×10^{-6}	1.099×10^{-1}	1.099×10^{-3}	1.099×10^{-5}
midpoint	8.748×10^{-2}	8.763×10^{-4}	8.765×10^{-6}	1.725×10^{-1}	1.73×10^{-3}	1.731×10^{-5}	4.369×10^{-1}	4.393×10^{-3}	4.392×10^{-5}
midpointqp	8.768×10^{-2}	8.765×10^{-4}	8.765×10^{-6}	1.729×10^{-1}	1.731×10^{-3}	1.731×10^{-5}	4.378×10^{-1}	4.394×10^{-3}	4.394×10^{-5}

NoS/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	1.762×10^{-5}	1.762×10^{-7}	1.762×10^{-9}	2.428×10^{-5}	2.427×10^{-7}	2.427×10^{-9}	3.18×10^{-5}	3.18×10^{-7}	3.18×10^{-9}
midpoint	7.551×10^{-5}	7.512×10^{-7}	7.549×10^{-9}	1.031×10^{-4}	1.026×10^{-6}	1.034×10^{-8}	1.332×10^{-4}	1.327×10^{-6}	1.418×10^{-8}
midpointqp	7.82×10^{-5}	7.767×10^{-7}	7.762×10^{-9}	1.06×10^{-4}	1.054×10^{-6}	1.053×10^{-8}	1.358×10^{-4}	1.352×10^{-6}	1.352×10^{-8}

NoS/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	2.167×10^{-2}	2.167×10^{-4}	2.167×10^{-6}	4.304×10^{-2}	4.303×10^{-4}	4.303×10^{-6}	1.077×10^{-1}	1.077×10^{-3}	1.077×10^{-5}
midpoint	8.663×10^{-2}	8.666×10^{-4}	8.667×10^{-6}	1.722×10^{-1}	1.721×10^{-3}	1.721×10^{-5}	4.285×10^{-1}	4.306×10^{-3}	4.307×10^{-5}
midpointqp	8.673×10^{-2}	8.669×10^{-4}	8.669×10^{-6}	1.723×10^{-1}	1.721×10^{-3}	1.721×10^{-5}	4.289×10^{-1}	4.307×10^{-3}	4.307×10^{-5}

PureS/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	3.707×10^{-4}	3.706×10^{-6}	3.706×10^{-8}	7.05×10^{-4}	7.05×10^{-6}	7.05×10^{-8}	1.712×10^{-3}	1.712×10^{-5}	1.712×10^{-7}
midpoint	1.485×10^{-3}	1.447×10^{-5}	1.442×10^{-7}	2.917×10^{-3}	2.756×10^{-5}	2.742×10^{-7}	7.791×10^{-3}	6.756×10^{-5}	6.64×10^{-7}
midpointqp	1.507×10^{-3}	1.467×10^{-5}	1.463×10^{-7}	2.964×10^{-3}	2.798×10^{-5}	2.781×10^{-7}	7.932×10^{-3}	6.865×10^{-5}	6.758×10^{-7}

PureS/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	1.295×10^{-5}	1.295×10^{-5}	1.295×10^{-5}	6.467×10^{-6}	6.467×10^{-6}	6.467×10^{-6}	2.586×10^{-6}	2.586×10^{-6}	2.586×10^{-6}
midpoint	1.295×10^{-5}	1.295×10^{-5}	1.295×10^{-5}	6.469×10^{-6}	6.467×10^{-6}	6.467×10^{-6}	2.588×10^{-6}	2.586×10^{-6}	2.586×10^{-6}
midpointqp	1.295×10^{-5}	1.295×10^{-5}	1.295×10^{-5}	6.469×10^{-6}	6.467×10^{-6}	6.467×10^{-6}	2.588×10^{-6}	2.586×10^{-6}	2.586×10^{-6}

B.2. Second order – errors for $b_n(t)$

quad/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	4.471×10^{-3}	4.47×10^{-5}	4.47×10^{-7}	8.82×10^{-3}	8.819×10^{-5}	8.819×10^{-7}	2.197×10^{-2}	2.196×10^{-4}	2.196×10^{-6}
midpoint	1.823×10^{-2}	1.432×10^{-4}	1.393×10^{-6}	4.176×10^{-2}	2.882×10^{-4}	2.753×10^{-6}	1.471×10^{-1}	7.608×10^{-4}	6.898×10^{-6}
midpointqp	1.992×10^{-2}	1.551×10^{-4}	1.507×10^{-6}	4.558×10^{-2}	3.12×10^{-4}	2.976×10^{-6}	1.603×10^{-1}	8.234×10^{-4}	7.456×10^{-6}

quad/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	2.18×10^{-2}	2.18×10^{-4}	2.181×10^{-6}	4.394×10^{-2}	4.392×10^{-4}	4.392×10^{-6}	1.082×10^{-1}	1.082×10^{-3}	1.082×10^{-5}
midpoint	8.761×10^{-2}	8.745×10^{-4}	8.745×10^{-6}	1.768×10^{-1}	1.761×10^{-3}	1.761×10^{-5}	4.327×10^{-1}	4.337×10^{-3}	4.328×10^{-5}
midpointqp	8.798×10^{-2}	8.749×10^{-4}	8.746×10^{-6}	1.775×10^{-1}	1.762×10^{-3}	1.761×10^{-5}	4.345×10^{-1}	4.338×10^{-3}	4.335×10^{-5}

dirac/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	1.257×10^{-1}	1.046×10^{-3}	1.045×10^{-5}	3.659×10^{-1}	4.826×10^{-3}	4.835×10^{-5}	9.797×10^{-1}	1.175×10^{-2}	1.169×10^{-4}
midpoint	1.356	1.015×10^{-2}	3.231×10^{-5}	1.445	7.853×10^{-2}	1.846×10^{-4}	1.567	4.909×10^{-1}	6.994×10^{-4}
midpointqp	1.358	1.365×10^{-2}	4.567×10^{-5}	1.546	1.026×10^{-1}	2.569×10^{-4}	1.638	6.32×10^{-1}	9.508×10^{-4}

dirac/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	2.17×10^{-2}	2.17×10^{-4}	2.17×10^{-6}	4.377×10^{-2}	4.377×10^{-4}	4.377×10^{-6}	1.073×10^{-1}	1.074×10^{-3}	1.074×10^{-5}
midpoint	1.018×10^{-1}	9.005×10^{-4}	8.892×10^{-6}	2.041×10^{-1}	1.799×10^{-3}	1.776×10^{-5}	5.004×10^{-1}	4.383×10^{-3}	4.326×10^{-5}
midpointqp	1.109×10^{-1}	8.992×10^{-4}	8.79×10^{-6}	2.244×10^{-1}	1.806×10^{-3}	1.765×10^{-5}	5.568×10^{-1}	4.42×10^{-3}	4.318×10^{-5}

double/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	1.723×10^{-3}	1.723×10^{-5}	1.723×10^{-7}	3.387×10^{-3}	3.386×10^{-5}	3.386×10^{-7}	8.373×10^{-3}	8.372×10^{-5}	8.372×10^{-7}
midpoint	7.758×10^{-3}	7.096×10^{-5}	7.029×10^{-7}	1.588×10^{-2}	1.403×10^{-4}	1.385×10^{-6}	4.363×10^{-2}	3.518×10^{-4}	3.432×10^{-6}
midpointqp	8.848×10^{-3}	8.086×10^{-5}	8.011×10^{-7}	1.804×10^{-2}	1.597×10^{-4}	1.577×10^{-6}	4.916×10^{-2}	3.996×10^{-4}	3.905×10^{-6}

double/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	2.125×10^{-2}	2.125×10^{-4}	2.125×10^{-6}	4.355×10^{-2}	4.355×10^{-4}	4.355×10^{-6}	1.074×10^{-1}	1.074×10^{-3}	1.074×10^{-5}
midpoint	8.478×10^{-2}	8.49×10^{-4}	8.492×10^{-6}	1.737×10^{-1}	1.741×10^{-3}	1.741×10^{-5}	4.267×10^{-1}	4.294×10^{-3}	4.293×10^{-5}
midpointqp	8.497×10^{-2}	8.492×10^{-4}	8.492×10^{-6}	1.741×10^{-1}	1.741×10^{-3}	1.741×10^{-5}	4.277×10^{-1}	4.295×10^{-3}	4.295×10^{-5}

NoS/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	1.849×10^{-5}	1.849×10^{-7}	1.849×10^{-9}	2.528×10^{-5}	2.528×10^{-7}	2.529×10^{-9}	3.275×10^{-5}	3.275×10^{-7}	3.276×10^{-9}
midpoint	7.965×10^{-5}	7.925×10^{-7}	7.981×10^{-9}	1.076×10^{-4}	1.071×10^{-6}	1.079×10^{-8}	1.373×10^{-4}	1.369×10^{-6}	1.461×10^{-8}
midpointqp	8.245×10^{-5}	8.193×10^{-7}	8.188×10^{-9}	1.106×10^{-4}	1.1×10^{-6}	1.099×10^{-8}	1.4×10^{-4}	1.394×10^{-6}	1.394×10^{-8}

NoS/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	2.143×10^{-2}	2.143×10^{-4}	2.143×10^{-6}	4.282×10^{-2}	4.282×10^{-4}	4.282×10^{-6}	1.089×10^{-1}	1.089×10^{-3}	1.089×10^{-5}
midpoint	8.568×10^{-2}	8.569×10^{-4}	8.57×10^{-6}	1.711×10^{-1}	1.713×10^{-3}	1.713×10^{-5}	4.328×10^{-1}	4.354×10^{-3}	4.354×10^{-5}
midpointqp	8.578×10^{-2}	8.572×10^{-4}	8.572×10^{-6}	1.712×10^{-1}	1.713×10^{-3}	1.713×10^{-5}	4.332×10^{-1}	4.354×10^{-3}	4.354×10^{-5}

PureS/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	3.703×10^{-4}	3.703×10^{-6}	3.702×10^{-8}	7.047×10^{-4}	7.047×10^{-6}	7.047×10^{-8}	1.712×10^{-3}	1.712×10^{-5}	1.712×10^{-7}
midpoint	1.485×10^{-3}	1.447×10^{-5}	1.442×10^{-7}	2.916×10^{-3}	2.755×10^{-5}	2.742×10^{-7}	7.789×10^{-3}	6.755×10^{-5}	6.639×10^{-7}
midpointqp	1.506×10^{-3}	1.467×10^{-5}	1.463×10^{-7}	2.963×10^{-3}	2.797×10^{-5}	2.781×10^{-7}	7.93×10^{-3}	6.864×10^{-5}	6.757×10^{-7}

PureS/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
sv2symp	2.589×10^{-5}	2.589×10^{-5}	2.589×10^{-5}	1.293×10^{-5}	1.293×10^{-5}	1.293×10^{-5}	5.173×10^{-6}	5.172×10^{-6}	5.172×10^{-6}
midpoint	2.59×10^{-5}	2.589×10^{-5}	2.589×10^{-5}	1.294×10^{-5}	1.293×10^{-5}	1.293×10^{-5}	5.175×10^{-6}	5.172×10^{-6}	5.172×10^{-6}
midpointqp	2.59×10^{-5}	2.589×10^{-5}	2.589×10^{-5}	1.294×10^{-5}	1.293×10^{-5}	1.293×10^{-5}	5.175×10^{-6}	5.172×10^{-6}	5.172×10^{-6}

B.3. Fourth order – errors for $q_n(t)$

quad/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	2.865×10^{-7}	5.51×10^{-10}	5.898×10^{-10}	6.509×10^{-7}	1.476×10^{-10}	2.968×10^{-9}	2.228×10^{-6}	9.889×10^{-10}	2.031×10^{-8}
rk4qp	3.27×10^{-7}	5.507×10^{-10}	5.501×10^{-10}	7.39×10^{-7}	5.353×10^{-11}	2.099×10^{-11}	2.511×10^{-6}	1.453×10^{-10}	6.118×10^{-11}
rkf45	6.78×10^{-8}	5.502×10^{-10}	5.904×10^{-10}	1.685×10^{-7}	1.296×10^{-10}	2.972×10^{-9}	6.68×10^{-7}	9.647×10^{-10}	2.032×10^{-8}
ab4	3.533×10^{-3}	3.53×10^{-5}	3.527×10^{-7}	7.064×10^{-3}	7.063×10^{-5}	7.055×10^{-7}	1.745×10^{-2}	1.751×10^{-4}	1.749×10^{-6}

quad/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	1.371×10^{-6}	3.443×10^{-9}	3.466×10^{-9}	2.732×10^{-6}	7.455×10^{-9}	8.864×10^{-9}	6.599×10^{-6}	1.172×10^{-8}	7.7×10^{-8}
rk4qp	1.381×10^{-6}	3.444×10^{-9}	3.468×10^{-9}	2.75×10^{-6}	7.424×10^{-9}	7.356×10^{-9}	6.638×10^{-6}	1.12×10^{-8}	1.13×10^{-8}
rkf45	2.129×10^{-7}	3.471×10^{-9}	3.466×10^{-9}	4.216×10^{-7}	7.379×10^{-9}	8.864×10^{-9}	1.023×10^{-6}	1.152×10^{-8}	7.685×10^{-8}
ab4	4.195×10^{-4}	3.768×10^{-6}	3.747×10^{-8}	5.5×10^{-4}	4.546×10^{-6}	4.976×10^{-8}	7.508×10^{-4}	4.96×10^{-6}	1.228×10^{-7}

dirac/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	3.284×10^{-3}	4.11×10^{-8}	3.874×10^{-11}	5.916×10^{-3}	6.747×10^{-8}	1.03×10^{-10}	4.282×10^{-2}	4.228×10^{-7}	6.462×10^{-10}
rk4qp	4.241×10^{-3}	5.391×10^{-8}	2.022×10^{-11}	7.602×10^{-3}	8.783×10^{-8}	7.222×10^{-11}	5.61×10^{-2}	5.468×10^{-7}	6.104×10^{-11}
rkf45	1.621×10^{-3}	1.759×10^{-8}	7.666×10^{-11}	$3. \times 10^{-3}$	3.113×10^{-8}	6.779×10^{-11}	1.947×10^{-2}	2.051×10^{-7}	3.397×10^{-10}
ab4	2.952×10^{-1}	1.09×10^{-3}	1.095×10^{-5}	6.553×10^{-1}	1.01×10^{-3}	1.016×10^{-5}	1.28	2.692×10^{-3}	2.755×10^{-5}

dirac/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	3.676×10^{-6}	1.685×10^{-10}	2.575×10^{-10}	7.051×10^{-6}	3.221×10^{-10}	8.182×10^{-10}	1.722×10^{-5}	1.122×10^{-9}	6.378×10^{-9}
rk4qp	5.037×10^{-6}	1.86×10^{-10}	6.011×10^{-11}	9.665×10^{-6}	3.503×10^{-10}	8.565×10^{-11}	2.362×10^{-5}	8.024×10^{-10}	1.398×10^{-10}
rkf45	1.503×10^{-6}	3.749×10^{-11}	2.449×10^{-10}	2.94×10^{-6}	3.258×10^{-11}	9.732×10^{-10}	7.283×10^{-6}	3.389×10^{-10}	6.417×10^{-9}
ab4	5.565×10^{-3}	5.217×10^{-5}	5.223×10^{-7}	6.307×10^{-3}	5.578×10^{-5}	5.59×10^{-7}	8.02×10^{-3}	6.146×10^{-5}	6.204×10^{-7}

double/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	1.001×10^{-7}	2.442×10^{-11}	1.786×10^{-10}	2.006×10^{-7}	6.527×10^{-11}	5.316×10^{-10}	5.291×10^{-7}	4.041×10^{-10}	1.666×10^{-8}
rk4qp	1.129×10^{-7}	9.596×10^{-12}	2.287×10^{-12}	2.256×10^{-7}	1.868×10^{-11}	6.062×10^{-12}	5.923×10^{-7}	4.655×10^{-11}	7.707×10^{-11}
rkf45	1.867×10^{-8}	1.009×10^{-11}	1.788×10^{-10}	3.843×10^{-8}	3.121×10^{-11}	5.324×10^{-10}	1.083×10^{-7}	3.049×10^{-10}	1.666×10^{-8}
ab4	4.271×10^{-3}	4.284×10^{-5}	4.284×10^{-7}	8.436×10^{-3}	8.463×10^{-5}	8.461×10^{-7}	2.093×10^{-2}	2.1×10^{-4}	2.099×10^{-6}

double/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	1.317×10^{-6}	1.381×10^{-10}	7.958×10^{-11}	2.595×10^{-6}	2.79×10^{-10}	2.274×10^{-10}	6.58×10^{-6}	$1. \times 10^{-9}$	1.58×10^{-8}
rk4qp	1.32×10^{-6}	1.312×10^{-10}	1.007×10^{-11}	2.602×10^{-6}	2.608×10^{-10}	1.522×10^{-11}	6.598×10^{-6}	6.569×10^{-10}	5.339×10^{-11}
rkf45	2.02×10^{-7}	1.81×10^{-11}	8.081×10^{-11}	3.981×10^{-7}	2.337×10^{-11}	2.324×10^{-10}	1.009×10^{-6}	2.204×10^{-10}	1.581×10^{-8}
ab4	3.715×10^{-4}	3.203×10^{-6}	3.206×10^{-8}	4.441×10^{-4}	3.416×10^{-6}	3.429×10^{-8}	6.49×10^{-4}	3.844×10^{-6}	5.351×10^{-8}

NoS/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	2.405×10^{-10}	8.695×10^{-11}	8.286×10^{-10}	1.726×10^{-10}	1.418×10^{-10}	1.287×10^{-9}	1.282×10^{-10}	4.785×10^{-10}	4.865×10^{-9}
rk4qp	2.577×10^{-10}	3.684×10^{-11}	3.684×10^{-11}	1.815×10^{-10}	1.949×10^{-11}	1.949×10^{-11}	1.214×10^{-10}	9.323×10^{-12}	9.33×10^{-12}
rkf45	5.713×10^{-11}	8.667×10^{-11}	8.286×10^{-10}	4.106×10^{-11}	1.415×10^{-10}	1.287×10^{-9}	5.762×10^{-11}	4.79×10^{-10}	4.865×10^{-9}
ab4	1.217×10^{-5}	1.211×10^{-7}	1.46×10^{-9}	1.149×10^{-5}	1.143×10^{-7}	1.727×10^{-9}	$1. \times 10^{-5}$	9.939×10^{-8}	4.972×10^{-9}

NoS/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	1.299×10^{-6}	1.335×10^{-10}	4.173×10^{-11}	2.578×10^{-6}	2.697×10^{-10}	5.083×10^{-11}	6.454×10^{-6}	6.209×10^{-10}	2.603×10^{-10}
rk4qp	1.3×10^{-6}	1.294×10^{-10}	1.311×10^{-11}	2.582×10^{-6}	2.607×10^{-10}	1.954×10^{-11}	6.462×10^{-6}	6.337×10^{-10}	4.847×10^{-11}
rkf45	1.997×10^{-7}	2.053×10^{-11}	4.3×10^{-11}	3.963×10^{-7}	3.68×10^{-11}	5.154×10^{-11}	9.92×10^{-7}	1.46×10^{-10}	2.212×10^{-10}
ab4	9.795×10^{-5}	6.17×10^{-7}	6.166×10^{-9}	1.503×10^{-4}	6.476×10^{-7}	6.439×10^{-9}	3.12×10^{-4}	6.84×10^{-7}	6.546×10^{-9}

PureS/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	4.753×10^{-9}	1.185×10^{-10}	2.224×10^{-10}	9.438×10^{-9}	2.965×10^{-10}	2.793×10^{-10}	2.631×10^{-8}	2.683×10^{-10}	1.294×10^{-9}
rk4qp	4.881×10^{-9}	2.739×10^{-11}	2.744×10^{-11}	9.722×10^{-9}	1.063×10^{-11}	1.104×10^{-11}	2.714×10^{-8}	1.619×10^{-12}	3.488×10^{-12}
rkf45	7.944×10^{-10}	3.156×10^{-11}	8.664×10^{-10}	1.693×10^{-9}	3.015×10^{-11}	2.611×10^{-9}	5.422×10^{-9}	1.172×10^{-10}	1.265×10^{-9}
ab4	7.725×10^{-5}	7.743×10^{-7}	7.677×10^{-9}	1.579×10^{-4}	1.583×10^{-6}	1.592×10^{-8}	4.003×10^{-4}	4.016×10^{-6}	3.879×10^{-8}

PureS/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	1.295×10^{-5}	1.295×10^{-5}	1.295×10^{-5}	6.467×10^{-6}	6.467×10^{-6}	6.467×10^{-6}	2.586×10^{-6}	2.586×10^{-6}	2.586×10^{-6}
rk4qp	1.295×10^{-5}	1.295×10^{-5}	1.295×10^{-5}	6.467×10^{-6}	6.467×10^{-6}	6.467×10^{-6}	2.586×10^{-6}	2.586×10^{-6}	2.586×10^{-6}
rkf45	1.295×10^{-5}	1.295×10^{-5}	1.295×10^{-5}	6.467×10^{-6}	6.467×10^{-6}	6.467×10^{-6}	2.586×10^{-6}	2.586×10^{-6}	2.586×10^{-6}
ab4	1.295×10^{-5}	1.295×10^{-5}	1.295×10^{-5}	6.467×10^{-6}	6.467×10^{-6}	6.467×10^{-6}	2.586×10^{-6}	2.586×10^{-6}	2.586×10^{-6}

B.4. Fourth order — errors for $b_n(t)$

quad/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	2.847×10^{-7}	1.003×10^{-9}	1.022×10^{-9}	6.441×10^{-7}	1.373×10^{-10}	2.712×10^{-9}	2.228×10^{-6}	9.067×10^{-10}	1.857×10^{-8}
rk4qp	3.25×10^{-7}	1.003×10^{-9}	1.002×10^{-9}	7.313×10^{-7}	5.296×10^{-11}	2.053×10^{-11}	2.511×10^{-6}	1.454×10^{-10}	6.31×10^{-11}
rkf45	6.737×10^{-8}	1.002×10^{-9}	1.023×10^{-9}	1.667×10^{-7}	1.186×10^{-10}	2.718×10^{-9}	6.679×10^{-7}	8.814×10^{-10}	1.86×10^{-8}
ab4	3.512×10^{-3}	3.508×10^{-5}	3.505×10^{-7}	6.989×10^{-3}	6.989×10^{-5}	6.981×10^{-7}	1.747×10^{-2}	1.751×10^{-4}	1.749×10^{-6}

quad/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	1.348×10^{-6}	3.368×10^{-9}	3.399×10^{-9}	2.706×10^{-6}	7.313×10^{-9}	8.297×10^{-9}	6.593×10^{-6}	1.201×10^{-8}	7.663×10^{-8}
rk4qp	1.358×10^{-6}	3.369×10^{-9}	3.406×10^{-9}	2.724×10^{-6}	7.302×10^{-9}	7.283×10^{-9}	6.632×10^{-6}	1.119×10^{-8}	1.117×10^{-8}
rkf45	2.101×10^{-7}	3.413×10^{-9}	3.399×10^{-9}	4.185×10^{-7}	7.276×10^{-9}	8.305×10^{-9}	1.022×10^{-6}	1.175×10^{-8}	7.646×10^{-8}
ab4	4.187×10^{-4}	3.77×10^{-6}	3.745×10^{-8}	5.455×10^{-4}	4.51×10^{-6}	4.83×10^{-8}	7.542×10^{-4}	4.992×10^{-6}	1.231×10^{-7}

dirac/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	1.164×10^{-3}	1.46×10^{-8}	1.849×10^{-11}	1.071×10^{-2}	1.202×10^{-7}	1.683×10^{-10}	6.202×10^{-2}	6.725×10^{-7}	9.511×10^{-10}
rk4qp	1.503×10^{-3}	1.915×10^{-8}	1.413×10^{-11}	1.382×10^{-2}	1.565×10^{-7}	1.287×10^{-10}	7.915×10^{-2}	8.697×10^{-7}	9.272×10^{-11}
rkf45	5.764×10^{-4}	6.248×10^{-9}	3.149×10^{-11}	5.305×10^{-3}	5.547×10^{-8}	8.918×10^{-11}	3.248×10^{-2}	3.262×10^{-7}	3.574×10^{-10}
ab4	9.661×10^{-2}	3.876×10^{-4}	3.889×10^{-6}	9.443×10^{-1}	1.794×10^{-3}	1.809×10^{-5}	1.495	4.31×10^{-3}	4.382×10^{-5}

dirac/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	3.579×10^{-6}	1.693×10^{-10}	2.559×10^{-10}	7.044×10^{-6}	3.165×10^{-10}	8.138×10^{-10}	1.699×10^{-5}	1.126×10^{-9}	6.312×10^{-9}
rk4qp	4.905×10^{-6}	1.875×10^{-10}	6.095×10^{-11}	9.656×10^{-6}	3.43×10^{-10}	8.417×10^{-11}	2.33×10^{-5}	8.094×10^{-10}	1.479×10^{-10}
rkf45	1.464×10^{-6}	3.167×10^{-11}	2.43×10^{-10}	2.938×10^{-6}	3.728×10^{-11}	9.681×10^{-10}	7.185×10^{-6}	3.512×10^{-10}	6.348×10^{-9}
ab4	5.444×10^{-3}	5.106×10^{-5}	5.111×10^{-7}	6.308×10^{-3}	5.581×10^{-5}	5.592×10^{-7}	7.896×10^{-3}	6.046×10^{-5}	6.104×10^{-7}

double/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	1.001×10^{-7}	2.425×10^{-11}	1.776×10^{-10}	2.006×10^{-7}	6.516×10^{-11}	5.24×10^{-10}	5.287×10^{-7}	3.88×10^{-10}	1.59×10^{-8}
rk4qp	1.128×10^{-7}	9.553×10^{-12}	2.335×10^{-12}	2.255×10^{-7}	1.866×10^{-11}	6.009×10^{-12}	5.919×10^{-7}	4.644×10^{-11}	7.897×10^{-11}
rkf45	1.867×10^{-8}	9.875×10^{-12}	1.779×10^{-10}	3.842×10^{-8}	3.104×10^{-11}	5.248×10^{-10}	1.083×10^{-7}	2.96×10^{-10}	1.589×10^{-8}
ab4	4.27×10^{-3}	4.283×10^{-5}	4.283×10^{-7}	8.434×10^{-3}	8.461×10^{-5}	8.459×10^{-7}	2.091×10^{-2}	2.098×10^{-4}	2.098×10^{-6}

double/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	1.275×10^{-6}	1.357×10^{-10}	7.93×10^{-11}	2.61×10^{-6}	2.795×10^{-10}	2.291×10^{-10}	6.431×10^{-6}	9.859×10^{-10}	1.544×10^{-8}
rk4qp	1.278×10^{-6}	1.287×10^{-10}	8.07×10^{-12}	2.617×10^{-6}	2.611×10^{-10}	1.358×10^{-11}	6.448×10^{-6}	6.493×10^{-10}	4.824×10^{-11}
rkf45	1.956×10^{-7}	1.569×10^{-11}	8.033×10^{-11}	4.003×10^{-7}	2.324×10^{-11}	2.334×10^{-10}	9.861×10^{-7}	2.212×10^{-10}	1.546×10^{-8}
ab4	3.616×10^{-4}	3.123×10^{-6}	3.126×10^{-8}	4.477×10^{-4}	3.446×10^{-6}	3.459×10^{-8}	6.343×10^{-4}	3.762×10^{-6}	5.233×10^{-8}

NoS/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	2.315×10^{-10}	8.416×10^{-11}	8.496×10^{-10}	1.709×10^{-10}	1.501×10^{-10}	1.363×10^{-9}	1.287×10^{-10}	4.976×10^{-10}	5.047×10^{-9}
rk4qp	2.47×10^{-10}	2.2×10^{-11}	2.201×10^{-11}	1.786×10^{-10}	1.828×10^{-11}	1.828×10^{-11}	1.21×10^{-10}	1.335×10^{-11}	1.335×10^{-11}
rkf45	4.745×10^{-11}	8.395×10^{-11}	8.496×10^{-10}	4.034×10^{-11}	1.497×10^{-10}	1.363×10^{-9}	6.015×10^{-11}	4.979×10^{-10}	5.047×10^{-9}
ab4	1.246×10^{-5}	1.24×10^{-7}	1.511×10^{-9}	1.175×10^{-5}	1.168×10^{-7}	1.788×10^{-9}	1.017×10^{-5}	1.011×10^{-7}	5.151×10^{-9}

NoS/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	1.282×10^{-6}	1.325×10^{-10}	4.284×10^{-11}	2.566×10^{-6}	2.653×10^{-10}	4.116×10^{-11}	6.524×10^{-6}	6.385×10^{-10}	2.428×10^{-10}
rk4qp	1.284×10^{-6}	1.278×10^{-10}	9.143×10^{-12}	2.57×10^{-6}	2.546×10^{-10}	1.756×10^{-11}	6.532×10^{-6}	6.535×10^{-10}	3.917×10^{-11}
rkf45	1.971×10^{-7}	1.87×10^{-11}	4.349×10^{-11}	3.946×10^{-7}	3.901×10^{-11}	4.469×10^{-11}	1.003×10^{-6}	1.359×10^{-10}	2.087×10^{-10}
ab4	9.781×10^{-5}	6.202×10^{-7}	6.199×10^{-9}	1.487×10^{-4}	6.416×10^{-7}	6.377×10^{-9}	3.154×10^{-4}	6.868×10^{-7}	6.585×10^{-9}

PureS/Sol.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	4.752×10^{-9}	1.161×10^{-10}	2.201×10^{-10}	9.436×10^{-9}	2.965×10^{-10}	2.74×10^{-10}	2.63×10^{-8}	2.681×10^{-10}	1.292×10^{-9}
rk4qp	4.879×10^{-9}	1.034×10^{-11}	1.054×10^{-11}	9.72×10^{-9}	5.901×10^{-12}	7.241×10^{-12}	2.713×10^{-8}	2.156×10^{-12}	1.005×10^{-11}
rkf45	7.937×10^{-10}	1.935×10^{-11}	8.682×10^{-10}	1.692×10^{-9}	2.888×10^{-11}	2.609×10^{-9}	5.421×10^{-9}	1.172×10^{-10}	1.264×10^{-9}
ab4	7.724×10^{-5}	7.741×10^{-7}	7.675×10^{-9}	1.579×10^{-4}	1.583×10^{-6}	1.592×10^{-8}	4.002×10^{-4}	4.015×10^{-6}	3.878×10^{-8}

PureS/Disp.	T = 1000			T = 2000			T = 5000		
dT	0.01	0.001	0.0001	0.01	0.001	0.0001	0.01	0.001	0.0001
rk4	2.589×10^{-5}	2.589×10^{-5}	2.589×10^{-5}	1.293×10^{-5}	1.293×10^{-5}	1.293×10^{-5}	5.172×10^{-6}	5.172×10^{-6}	5.172×10^{-6}
rk4qp	2.589×10^{-5}	2.589×10^{-5}	2.589×10^{-5}	1.293×10^{-5}	1.293×10^{-5}	1.293×10^{-5}	5.172×10^{-6}	5.172×10^{-6}	5.172×10^{-6}
rkf45	2.589×10^{-5}	2.589×10^{-5}	2.589×10^{-5}	1.293×10^{-5}	1.293×10^{-5}	1.293×10^{-5}	5.172×10^{-6}	5.172×10^{-6}	5.172×10^{-6}
ab4	2.589×10^{-5}	2.589×10^{-5}	2.589×10^{-5}	1.294×10^{-5}	1.293×10^{-5}	1.293×10^{-5}	5.173×10^{-6}	5.172×10^{-6}	5.172×10^{-6}

References

- [1] TodaBenchmarking, <https://bitbucket.org/trogdon/todabenchmarking>. (Accessed 12 September 2017).
- [2] D. Bilman, S. Konstantinou-Rizos, Discrete integrable systems, Darboux transformations, and Yang–Baxter maps, in: D. Levi, R. Rebelo, P. Winternitz (Eds.), Symmetries Integr. Differ. Equations, Springer International Publishing, Cham, 2017, pp. 195–260.

- [3] D. Bilman, I. Nenciu, On the evolution of scattering data under perturbations of the Toda lattice, *Phys. D, Nonlinear Phenom.* 330 (2016) 1–16.
- [4] D. Bilman, T. Trogdon, ISTPackage, <https://bitbucket.org/trogdon/istpackage>, 2013.
- [5] D. Bilman, T. Trogdon, Numerical inverse scattering for the Toda lattice, *Commun. Math. Phys.* 352 (2017) 805–879.
- [6] J. Butcher, A history of Runge–Kutta methods, *Appl. Numer. Math.* 20 (1996) 247–260.
- [7] J. Butcher, Numerical methods for ordinary differential equations in the 20th century, *J. Comput. Appl. Math.* 125 (2000) 1–29.
- [8] E. Fehlberg, Klassische Runge–Kutta-Formeln vierter und niedrigerer Ordnung mit Schrittweiten-Kontrolle und ihre Anwendung auf Wärmeleitungsprobleme, *Computing* 6 (1970) 61–71.
- [9] I. Fermi, P. Pasta, S. Ulam, M. Tsingou, Studies of the Nonlinear Problems, Tech. rep., Los Alamos National Laboratory (LANL), Los Alamos, NM (United States), May 1955.
- [10] H. Flaschka, The Toda lattice. II. Existence of integrals, *Phys. Rev. B* 9 (1974) 1924–1925.
- [11] J. Gaison, S. Moskow, J.D. Wright, Q. Zhang, Approximation of polyatomic FPU lattices by KdV equations, *Multiscale Model. Simul.* 12 (2014) 953–995.
- [12] E. Hairer, S. Nørsett, G. Wanner, *Solving Ordinary Differential Equation I: Nonstiff Problems*, 2 ed., Springer-Verlag, Berlin, 1993.
- [13] A. Hoffman, J.D. Wright, Nanopteron solutions of diatomic Fermi–Pasta–Ulam–Tsingou lattices with small mass-ratio, *Phys. D, Nonlinear Phenom.* (2017), <https://doi.org/10.1016/j.physd.2017.07.004>.
- [14] T.E. Hull, W.H. Enright, B.M. Fellen, A.E. Sedgwick, Comparing numerical methods for ordinary differential equations, *SIAM J. Numer. Anal.* 9 (1972) 603–637.
- [15] S. Kamvissis, On the long time behavior of the doubly infinite Toda lattice under initial data decaying at infinity, *Commun. Math. Phys.* 153 (1993) 479–519.
- [16] H. Krüger, G. Teschl, Long-time asymptotics of the Toda lattice for decaying initial data revisited, *Rev. Math. Phys.* 21 (2009) 61–109.
- [17] W. Kutta, Beitrag zur näherungsweise Integration totaler Differentialgleichungen, *Z. Math. Phys.* 46 (1901) 435–453.
- [18] P.D. Lax, Integrals of nonlinear equations of evolution and solitary waves, *Commun. Pure Appl. Math.* 21 (1968) 467–490.
- [19] R.J. LeVeque, *Finite Difference Methods for Ordinary and Partial Differential Equations*, SIAM, Philadelphia, PA, 2007.
- [20] S. Manakov, Complete integrability and stochastization of discrete dynamical systems, *Sov. Phys. JETP* 40 (1975) 269–274.
- [21] S. Olver, RHPackage, <http://www.maths.usyd.edu.au/u/olver/projects/RHPackage.html>, 2010.
- [22] S. Olver, A general framework for solving Riemann–Hilbert problems numerically, *Numer. Math.* 122 (2012) 305–340.
- [23] S. Olver, T. Trogdon, Nonlinear steepest descent and numerical solution of Riemann–Hilbert problems, *Commun. Pure Appl. Math.* 67 (2014) 1353–1389.
- [24] C. Runge, Über die numerische Auflösung von Differentialgleichungen, *Math. Ann.* 46 (1895) 167–178.
- [25] M. Sato, B.E. Hubbard, A.J. Sievers, B. Ilic, D.A. Czaplewski, H.G. Craighead, Observation of locked intrinsic localized vibrational modes in a micromechanical oscillator array, *Phys. Rev. Lett.* 90 (2003) 044102.
- [26] M. Sheikholeslami, D.D. Ganji, Entropy generation of nanofluid in presence of magnetic field using lattice Boltzmann method, *Phys. A, Stat. Mech. Appl.* 417 (2015) 273–286.
- [27] M. Sheikholeslami, M. Gorji-Bandpy, D.D. Ganji, Lattice Boltzmann method for MHD natural convection heat transfer using nanofluid, *Powder Technol.* 254 (2014) 82–93.
- [28] M. Toda, Wave propagation in anharmonic lattices, *J. Phys. Soc. Jpn.* 23 (1967) 501–506.
- [29] T. Trogdon, B. Deconinck, A Riemann–Hilbert problem for the finite-genus solutions of the KdV equation and its numerical solution, *Physica D* 251 (2013) 1–18.
- [30] T. Trogdon, S. Olver, Numerical inverse scattering for the focusing and defocusing nonlinear Schrödinger equations, *Proc. R. Soc. A, Math. Phys. Eng. Sci.* 469 (2012) 20120330.
- [31] T. Trogdon, S. Olver, *Riemann–Hilbert Problems, Their Numerical Solution and the Computation of Nonlinear Special Functions*, SIAM, Philadelphia, PA, 2016.
- [32] T. Trogdon, S. Olver, B. Deconinck, Numerical inverse scattering for the Korteweg–de Vries and modified Korteweg–de Vries equations, *Physica D* 241 (2012) 1003–1025.
- [33] A. Vainchtein, Y. Starosvetsky, J.D. Wright, R. Perline, Solitary waves in diatomic chains, *Phys. Rev. E* 93 (2016) 042210.
- [34] N.J. Zabusky, M.D. Kruskal, Interaction of “solitons” in a collisionless plasma and the recurrence of initial states, *Phys. Rev. Lett.* 15 (1965) 240–243.