Appl. Numer. Math. 229 (2026) 156–175

Contents lists available at ScienceDirect

Applied Numerical Mathematics

journal homepage: www.elsevier.com/locate/apnum

Research Paper

Energy-conserving successive secant-difference method for nonlinear wave equations

Jaemin Shin [a], June-Yub Lee [b,*]

[a] Department of Mathematics, Chungbuk National University, Cheongju, 28644, South Korea
[b] Department of Mathematics, Ewha Womans University, Seoul, 03760, South Korea

A R T I C L E I N F O

Keywords:
Nonlinear wave equation
Secant-difference
Energy conservation
Multi-stage method
Sine-Gordon and boussinesq equations

A B S T R A C T

We present a high-order time-discretization method that conserves energy in nonlinear wave equations. The method builds on the secant-difference approach, which inherently ensures energy conservation. To achieve higher temporal accuracy while preserving the structural properties necessary for energy preservation, we propose a novel high-order multi-stage method based on a successive strategy. While the successive secant-difference (SSD) method deviates from the standard Runge–Kutta framework, we derive the order conditions for high-order accuracy using classical Taylor expansion techniques. We further derive the full set of order conditions up to sixth-order and determine the minimal independent constraints by examining their linear dependencies. This derivation provides detailed insights and introduces a novel perspective distinct from traditional composition methods. We validate the temporal accuracy and unconditional energy stability of our proposed method through numerical examples, including the sine-Gordon and Boussinesq equations. We also provide numerical comparisons with fourth-order averaged vector field methods.

1. Introduction

Hyperbolic partial differential equations (PDEs) are essential for modeling wave propagation phenomena and appear in a wide range of scientific disciplines, including acoustics, elasticity, medical imaging, fluid mechanics, quantum mechanics, and electromagnetics. For more detailed information, refer to references [1–3]. While linear wave equations facilitate various modeling and applications, more complex challenges often arise in the nonlinear case. Regardless of whether the wave equations are linear or nonlinear, conserving energy is crucial from a modeling standpoint. Numerically, structure-preserving methods are vital for ensuring long-term stable simulations.

Geometric numerical integrators have become increasingly prominent, where geometric refers to preserving one or more physical or geometric properties of the system, accurate up to round-off error. Achieving both accuracy and stability in conservative methods is essential for long-term simulations, particularly for wave equations. Numerous energy-conserving methods have been developed even for the linear wave equation, a linear Hamiltonian PDE, starting from the theta method [4] and the leapfrog method [5] to more recent approaches [6–8]. The numerical treatment of Hamiltonian PDEs, including nonlinear cases, has been an intensive research area in recent years. The symplectic Runge–Kutta (RK) method [9–11] has been studied to preserve the symplectic form.

Other notable structure-preserving integrators include as follows. Hamiltonian boundary value methods [12–14] provide high-order, energy-preserving approximations based on discrete line-integral formulations. The invariant energy quadratization

* Corresponding author.
E-mail address: jyllee@ewha.ac.kr (J.-Y. Lee).

https://doi.org/10.1016/j.apnum.2026.05.016
Received 17 July 2025; Received in revised form 30 March 2026; Accepted 29 May 2026
Available online 6 June 2026
0168-9274/

approach [15,16] reformulates nonlinear energies into quadratic forms enabling linear energy-stable schemes and locally conservative schemes [17,18] maintain local conservation laws at the discrete level. Within this broad class of structure-preserving techniques, those most closely aligned with the present work can be summarized as follows. The average vector field methods [19–22] preserve energy by integrating the vector field along straight-line segments. Discrete gradient methods [23] construct modified gradients that guarantee exact discrete energy conservation. Discrete variational derivative methods [24,25] retain dissipation or conservation structures by employing discrete analogs of variational derivatives.

We consider wave equations in the domain $\Omega \subset \mathbb{R}^d$ $(d = 1, 2, 3)$ which can be represented as

$$\frac{\partial^2 u}{\partial t^2} = -\mathrm{grad}_V \mathcal{E}(u) = -\mathcal{G}_V \frac{\delta \mathcal{E}(u)}{\delta u}, \tag{1}$$

where $\mathcal{E}(u)$ is an energy functional defined on a specific inner product space V. Here, $\mathrm{grad}_V \mathcal{E}(u)$ is a gradient of $\mathcal{E}$ and $\mathcal{G}_V$ is a symmetric positive definite linear operator associated with V,

$$\langle \mathrm{grad}_V \mathcal{E}(u), v\rangle_V := \frac{d}{d\theta}\mathcal{E}(u+\theta v)\Big|_{\theta=0} = \left\langle \frac{\delta \mathcal{E}}{\delta u}, v\right\rangle_{L^2} = \left\langle \mathcal{G}_V \frac{\delta \mathcal{E}}{\delta u}, v\right\rangle_V.$$

We usually consider L^2- and H^{-1}-inner product spaces for the candidates of V, then $\mathcal{G}_{L^2} = I$ and $\mathcal{G}_{H^{-1}} = -\Delta$ since $\langle v_1, v_2\rangle_{L^2} = \langle -\Delta v_1, v_2\rangle_{H^{-1}}$. The system is completed with initial and boundary conditions, and for simplicity, we adopt periodic or zero-Neumann boundary conditions in this study.

The Hamiltonian functional $\mathcal{H}$ corresponding to (1) is defined as

$$\mathcal{H}(u) = \frac{1}{2}\left\| \frac{\partial u}{\partial t}\right\|_V^2 + \mathcal{E}(u) \tag{2}$$

and the solution satisfies the energy conservation property,

$$\frac{d\mathcal{H}}{dt} = \left\langle \frac{\partial^2 u}{\partial t^2}, \frac{\partial u}{\partial t}\right\rangle_V + \left\langle \frac{\delta \mathcal{E}}{\delta u}, \frac{\partial u}{\partial t}\right\rangle_{L^2} = \left\langle \frac{\partial^2 u}{\partial t^2} + \mathcal{G}_V \frac{\delta \mathcal{E}}{\delta u}, \frac{\partial u}{\partial t}\right\rangle_V = 0. \tag{3}$$

For example, with a given positive definite symmetric matrix $\mathbf{M}(\mathbf{x})$,

$$\mathcal{H}_{SG}(u) = \frac{1}{2}\left\| \frac{\partial u}{\partial t}\right\|_{L^2}^2 + \int_\Omega \left(\frac{1}{2}\left|\sqrt{\mathbf{M}(\mathbf{x})}\nabla u\right|^2 + \Phi(u)\right) d\mathbf{x}$$

is a Hamiltonian functional for a nonlinear wave equation in an inhomogeneous medium, $u_{tt} - \nabla\cdot(\mathbf{M}(\mathbf{x})\nabla u) + \Phi'(u) = 0$. A special case is the sine-Gordon equation with $\mathbf{M}(\mathbf{x}) = \mathrm{I}$ and $\Phi(u) = 1 - \cos u$. The Boussinesq equation $u_{tt} - \mathcal{G}_{H^{-1}}\left(\Delta u - \Phi'(u)\right) = 0$ is another example, acting as the gradient flow under the H^{-1}-inner product space corresponding to the following Hamiltonian functional

$$\mathcal{H}_{BS}(u) = \frac{1}{2}\left\| \frac{\partial u}{\partial t}\right\|_{H^{-1}}^2 + \int_\Omega \left(\frac{1}{2}|\nabla u|^2 + \Phi(u)\right) d\mathbf{x},$$

where $\Phi(u) = \frac{1}{2}u^2 - u^3$. These examples illustrate the variety of nonlinear wave equations and their corresponding Hamiltonian functionals, demonstrating the relevance and applicability of energy-conserving methods in modeling such systems.

Our proposed method builds on the second-order secant-difference method, aiming to achieve high-order accuracy through a successive procedure while maintaining energy conservation properties. The successive multistage (SMS) method, as proposed by the authors in [26], employs a strategy similar to the Crank–Nicolson method for the linear wave equation. The linear term's additivity allows the SMS method to be expressed as a classical Runge–Kutta (RK) method, which simplifies its accuracy analysis. However, it is important to note that the SMS method for semi-linear wave equations does not guarantee energy conservation [27, Section 5]. In contrast, the secant-difference method does not permit such an additive separation. Although the successive secant-difference (SSD) method does not conform to a standard RK method, the application of Taylor's theorem in multi-stage methods is similar, allowing us to determine the order conditions necessary for achieving high-order accuracy with the SSD methods.

It is worth to note that the classical composition method [28] provides a strategy for achieving high-order accuracy by recursively applying the symmetric second-order method. However, this approach demands a rapidly increasing number of stages and is inherently restricted to even-order methods. While the fourth-order method derived in this paper coincides with that obtained via the classical composition method [29,30], our approach extends beyond this. We systematically construct the algebraic structure of Taylor expansions to derive the full set of order conditions up to sixth order. This enables us to propose integration schemes of both even and odd orders. In particular, certain structural aspects of our order conditions closely resemble results presented in [31]. Interestingly, the fifth- and sixth-order coefficient tables we propose are adapted from symplectic RK methods [11,32]. Unlike our approach, however, these methods do not inherently conserve energy. As a result, our method allows for energy-preserving schemes with fewer stages compared to classical composition approaches.

In Section 2, we introduce the secant-difference method, which inherently conserves energy unconditionally. To attain higher-order temporal accuracy, we employ a successive strategy, resulting in what we term the successive secant-difference (SSD) method. The structural characteristics of the secant-difference approach simplify the proof of energy conservation. Section 3 details the derivation of the order conditions for the proposed methods, leveraging the midpoint properties of the secant-difference method. In Section 4, we present numerical examples to demonstrate the convergence order and energy conservation of the SSD method. In Section 5, we examine the complete set of order conditions up to sixth-order and identify minimal independent constraints based on their linear dependencies. Section 6 presents numerical comparisons with fourth-order averaged vector field methods applied to the sine-Gordon equation, clarifying the relationship between the SSD and AVF methods at higher orders. Finally, Section 7 concludes with a brief discussion.

2. Successive secant-difference methods

By defining an auxiliary variable $v = u_t$, we can represent the wave Eq. (1) in a canonical form:

$$\begin{aligned} \frac{\partial u}{\partial t} &= v, \\ \frac{\partial v}{\partial t} &= -\mathcal{G}_V \frac{\delta \mathcal{E}(u)}{\delta u}, \end{aligned} \tag{4}$$

where the corresponding Hamiltonian functional can be represented as

$$\mathcal{H}(u, v) = \frac{1}{2}\|v\|_V^2 + \mathcal{E}(u). \tag{5}$$

To express the problem in a canonical first-order system form suitable for applying one-step numerical methods such as the Runge–Kutta scheme, we introduce the auxiliary variable $v = u_t$. For example, v corresponds to the velocity when u denotes the displacement.

With the auxiliary variable v, the energy conservation is inherited as

$$\begin{aligned} \frac{d\mathcal{H}}{dt} &= \left\langle \frac{\partial v}{\partial t}, v \right\rangle_V + \left\langle \frac{\delta \mathcal{E}}{\delta u}, \frac{\partial u}{\partial t} \right\rangle_{L^2} \\ &= \left\langle \frac{\partial v}{\partial t}, \frac{\partial u}{\partial t} \right\rangle_V + \left\langle \mathcal{G}_V \frac{\delta \mathcal{E}}{\delta u}, \frac{\partial u}{\partial t} \right\rangle_V = \left\langle \frac{\partial v}{\partial t} + \mathcal{G}_V \frac{\delta \mathcal{E}}{\delta u}, \frac{\partial u}{\partial t} \right\rangle_V = 0. \end{aligned}$$

After discretization, we require the conservation of the discrete energy to be consistent with the physical energy.

We denote the approximation of (u, v) to (u^n, v^n) at time $t^n = n\Delta t$. Here, the vector $(u, v) \in V^2$ is expressed in a row-wise notation. Moreover, in the semi-discretized setting, (u^n, v^n) is understood as an element of V^2. We develop a time marching scheme for (4) to compute (u^{n+1}, v^{n+1}) using (u^n, v^n). A secant-difference method is given as follows:

$$\begin{aligned} \frac{u^{n+1} - u^n}{\Delta t} &= \frac{v^{n+1} + v^n}{2}, \\ \frac{v^{n+1} - v^n}{\Delta t} &= -\nabla_V \mathcal{E}\left(u^{n+1}, u^n\right), \end{aligned} \tag{6}$$

where $\nabla_V \mathcal{E}\left(u^{n+1}, u^n\right)$ is an approximation of $\mathrm{grad}_V \mathcal{E}(u)$ using u^n and u^{n+1}. We would like to define a secant-difference operator $\nabla_V \mathcal{E}\left(u^{n+1}, u^n\right)$ properly, so that the method preserves the discrete energy

$$\mathcal{H}(u^n, v^n) = \frac{1}{2}\|v^n\|_V^2 + \mathcal{E}(u^n),$$

while the time accuracy is at least second order. The following theorem provides a criterion to obtain the energy conserving property.

Theorem 1. *Suppose that a secant-difference operator $\nabla_V \mathcal{E}\left(u_2, u_1\right)$ for u_1, u_2 in the solution space of (4) satisfies the following equality,*

$$\left\langle \nabla_V \mathcal{E}\left(u_2, u_1\right), u_2 - u_1 \right\rangle_V = \mathcal{E}\left(u_2\right) - \mathcal{E}\left(u_1\right). \tag{7}$$

Then, the secant-difference method (6) is unconditionally energy conserving, meaning that, for any time step size Δt,

$$\mathcal{H}\left(u^{n+1}, v^{n+1}\right) = \mathcal{H}(u^n, v^n).$$

Proof. We first consider the difference of energy functionals:

$$\mathcal{H}\left(u^{n+1}, v^{n+1}\right) - \mathcal{H}(u^n, v^n) = \frac{1}{2}\left\|v^{n+1}\right\|_V^2 - \frac{1}{2}\|v^n\|_V^2 + \mathcal{E}\left(u^{n+1}\right) - \mathcal{E}(u^n). \tag{8}$$

Starting from the first two terms in (8), we can rearrange as follows:

$$\begin{aligned} \frac{1}{2}\left\|v^{n+1}\right\|_V^2 - \frac{1}{2}\|v^n\|_V^2 &= \frac{1}{2}\left\langle v^{n+1} - v^n, v^{n+1} + v^n \right\rangle_V \\ &= \left\langle -\nabla_V \mathcal{E}\left(u^{n+1}, u^n\right), u^{n+1} - u^n \right\rangle_V = \mathcal{E}(u^n) - \mathcal{E}\left(u^{n+1}\right) \end{aligned}$$

and then we obviously have $\mathcal{H}\left(u^{n+1}, v^{n+1}\right) - \mathcal{H}(u^n, v^n) = 0$. Therefore, we can conclude that the energy remains a physical constant $\mathcal{H}(t)$,

$$\mathcal{H}(t^n) := \mathcal{H}(u^n, v^n) = \mathcal{H}\left(u^0, v^0\right) = \mathcal{H}\left(t^0\right),$$

for any time step Δt. □

We remark that the condition (7) resembles the discrete gradient condition

$$\left\langle \bar{\nabla} H(y_2, y_1), y_2 - y_1 \right\rangle = H(y_2) - H(y_1),$$

where $H(y)$ denotes a Hamiltonian defined on the underlying phase space. This condition is known to guarantee exact energy conservation in discrete gradient methods [23,29]. For Hamiltonians of the form (5), it corresponds to a time-marching scheme of the type

$$y_2 = y_1 + hJ\bar{\nabla} H(y_2, y_1),$$

where J is the canonical symplectic matrix. In particular, if one defines

$$\bar{\nabla} H(y_2, y_1) = \left(\frac{v_2 + v_1}{2}, \nabla_V \mathcal{E}(u_2, u_1) \right),$$

where $y_i = (u_i, v_i)$, then (7) is equivalent to the discrete gradient formulation. Consequently, the proposed secant-difference formulation can be interpreted as a particular case within the class of discrete gradient methods, differing in the definition of the operator $\nabla_V \mathcal{E}(u_2, u_1)$.

We now define a secant-difference operator $\nabla_V \mathcal{E}(u_2, u_1)$. For simplicity of description, we start with the energy functional in the form of

$$\mathcal{E}(u) = \int_\Omega \left(\frac{1}{2} u \mathcal{L} u + \Phi(u) \right) d\mathbf{x}, \tag{9}$$

where $\mathcal{L}$ is a symmetric positive definite operator. And then the gradient under the L^2-inner product space is

$$\mathrm{grad}_{L^2} \mathcal{E}(u) = \mathcal{L} u + \Phi'(u).$$

Let us define the secant-difference for (9) under the L^2-inner product space as follows:

$$\nabla_{L^2} \mathcal{E}(u_2, u_1) = \mathcal{L} \frac{u_2 + u_1}{2} + \frac{\Phi(u_2) - \Phi(u_1)}{u_2 - u_1}, \tag{10}$$

where the pointwise value of $\frac{\Phi(u_2(x)) - \Phi(u_1(x))}{u_2(x) - u_1(x)}$ is replaced by $\Phi'(u_1(x))$ or $\Phi'(u_2(x))$ when $u_1(x) = u_2(x)$. We note that the use of the secant-difference term $\frac{\Phi(u_2) - \Phi(u_1)}{u_2 - u_1}$ is closely related to the discrete variational derivative approach [24,25,30], where such approximations play a fundamental role in achieving unconditional energy conservation. Then it can be shown that the secant-difference defined in (10) satisfies the desired criterion (7) from the simple algebraic manipulation as follows:

$$\begin{aligned}
\mathcal{E}(u_2) - \mathcal{E}(u_1) &= \int_\Omega \left(\frac{1}{2} u_2 \mathcal{L} u_2 - \frac{1}{2} u_1 \mathcal{L} u_1 + \Phi(u_2) - \Phi(u_1) \right) d\mathbf{x} \\
&= \int_\Omega \left(\frac{1}{2} (u_2 - u_1) \mathcal{L} (u_2 + u_1) + (u_2 - u_1) \frac{\Phi(u_2) - \Phi(u_1)}{u_2 - u_1} \right) d\mathbf{x} \\
&= \left\langle \mathcal{L} \frac{u_2 + u_1}{2} + \frac{\Phi(u_2) - \Phi(u_1)}{u_2 - u_1},\ u_2 - u_1 \right\rangle_{L^2}.
\end{aligned}$$

Since the gradient of (9) can be written as

$$\mathrm{grad}_V \mathcal{E}(u) = \mathcal{G}_V \left(\mathcal{L} u + \Phi'(u) \right),$$

we define the secant-difference for the V-inner product space as

$$\nabla_V \mathcal{E}(u_2, u_1) = \mathcal{G}_V \left(\mathcal{L} \frac{u_2 + u_1}{2} + \frac{\Phi(u_2) - \Phi(u_1)}{u_2 - u_1} \right). \tag{11}$$

It is easy to show that the secant-differences (11) also satisfies the desired criterion (7). To define a secant-difference $\nabla_V \mathcal{E}(u_2, u_1)$ using the gradient operator $\frac{\delta \mathcal{E}}{\delta u}$ for an arbitrary energy functional $\mathcal{E}(u)$, roughly speaking, we first take the average of two functions for the linear parts of the gradient operator and the pointwise secant-difference for the nonlinear parts, then apply the linear operator $\mathcal{G}_V$ to the computed secant-difference under the L^2 space. We do not explicitly write down form of the secant-difference operator for more complex cases, but the procedure above provides a basic idea for a recipe.

We extend the secant-difference method into a high-order, energy-conserving numerical scheme, termed the successive secant-difference (SSD) method. This approach shares similarities with the successive multistage (SMS) method [22], though the latter can apply only to the linear wave equation. Moreover, deriving the order conditions for this method proves more intricate than in the classical Runge–Kutta (RK) methods, as detailed in Section 3.

Algorithm (SSD). For a given coefficient vector

$$R_s = [r_1, r_2, \cdots, r_s],$$

the SSD(R_s) method computes (u^{n+1}, v^{n+1}) at $t^{n+1} = t^n + \Delta t$. Starting with

$$u_0 = u^n,\ v_0 = v^n$$

for the initial stage, we evaluate (u_i, v_i) for $i = 1, 2, \cdots, s$ by solving

$$\begin{aligned}
\frac{u_i - u_{i-1}}{r_i \Delta t} &= \frac{v_i + v_{i-1}}{2}, \\
\frac{v_i - v_{i-1}}{r_i \Delta t} &= -\nabla_V \mathcal{E}(u_i, u_{i-1}).
\end{aligned} \tag{12}$$

Then, the next time step approximations are given by

$$u^{n+1} = u_s,\ v^{n+1} = v_s.$$

Theorem 2. *The discrete energy between adjacent stages of the SSD method (12) satisfying the gradient condition (7) is conserved for any time step $r_i\Delta t$,*

$$\mathcal{H}(u_i, v_i) = \mathcal{H}(u_{i-1}, v_{i-1}).$$

Finally, the proposed method (12) is unconditionally energy conserving, meaning that $\mathcal{H}(u^{n+1}, v^{n+1}) = \mathcal{H}(u^n, v^n)$ for any time step Δt.

Proof. Similar to the proof of Theorem 1, we initially arrange the difference of adjacent discrete energy functionals.

$$\mathcal{H}(u_i, v_i) - \mathcal{H}(u_{i-1}, v_{i-1}) = \frac{1}{2}\|v_i\|_V^2 - \frac{1}{2}\|v_{i-1}\|_V^2 + \mathcal{E}(u_i) - \mathcal{E}(u_{i-1})$$

and then expand the first two terms as

$$\begin{aligned}\frac{1}{2}\|v_i\|_V^2 - \frac{1}{2}\|v_{i-1}\|_V^2 &= \frac{1}{2}\langle v_i - v_{i-1}, v_i + v_{i-1}\rangle_V \\ &= \left\langle -\nabla_V \mathcal{E}(u_i, u_{i-1}), u_i - u_{i-1}\right\rangle_V = \mathcal{E}(u_{i-1}) - \mathcal{E}(u_i).\end{aligned}$$

Now, we have $\mathcal{H}(u_i, v_i) = \mathcal{H}(u_{i-1}, v_{i-1})$ for any $r_i\Delta t$, $i = 1, 2, \cdots, s$. Therefore, by mathematical induction,

$$\mathcal{H}(u^{n+1}, v^{n+1}) = \mathcal{H}(u_s, v_s) = \mathcal{H}(u_0, v_0) = \mathcal{H}(u^n, v^n),$$

which proves the conservation of the discrete energy functional $\mathcal{H}(u^n, v^n)$. □

3. Order conditions for the SSD method up to fourth order

The SSD method (12) with the secant-difference (11) is motivated by the proof structure in Theorem 2 focused on energy conservation. To derive the order conditions for SSD methods, we need to reformulate them as expansions at the midpoints and follow the sequential Taylor expansions.

Lemma 3. *Let us define $\mathbf{w}_i = (u_i, v_i)$ and $\overline{\mathbf{w}}_i = (\overline{u}_i, \overline{v}_i) = \frac{1}{2}(\mathbf{w}_i + \mathbf{w}_{i-1})$. Then the SSD method (12) for the energy functional (11) can be written in the following form:*

$$\frac{\mathbf{w}_i - \mathbf{w}_{i-1}}{r_i\Delta t} = \mathbf{F}_i(\overline{\mathbf{w}}_i), \text{ for } i = 1, 2, \cdots, s. \tag{13}$$

Moreover, $\mathbf{F}_i(\overline{\mathbf{w}}_i)$ can be explicitly computed as follows:

$$\mathbf{F}_i(\overline{\mathbf{w}}_i) = \mathbf{f}_0(\overline{\mathbf{w}}_i) + \sum_{p=1}^{\infty} r_i^{2p}\Delta t^{2p}\, \mathbf{f}_{2p}(\overline{\mathbf{w}}_i), \tag{14}$$

where $\mathbf{f}_0(\overline{\mathbf{w}}_i) = (\overline{v}_i, -\mathcal{G}_V(\mathcal{L}\overline{u}_i + \Phi'(\overline{u}_i)))$, $\mathbf{f}_{2p}(\overline{\mathbf{w}}_i) = (0, \frac{-1}{2^{2p}(2p+1)!}\overline{v}_i^{2p}\,\Phi^{(2p+1)}(\overline{u}_i))$, for $p \geq 1$.

Proof. The SSD method (12) can be written as

$$\frac{\mathbf{w}_i - \mathbf{w}_{i-1}}{r_i\Delta t} = \left(\overline{v}_i, -\mathcal{G}_V\left(\mathcal{L}\overline{u}_i + D\Phi(u_i, u_{i-1})\right)\right),$$

where the secant-difference term is defined as

$$D\Phi(u_i, u_{i-1}) = \frac{\Phi(u_i) - \Phi(u_{i-1})}{u_i - u_{i-1}}.$$

Consider the two Taylor series for $\Phi(u_i)$ and $\Phi(u_{i-1})$ expanded at $\overline{u}_i$,

$$\begin{aligned}\Phi(u_i) &= \Phi(\overline{u}_i) + \delta u_i\Phi'(\overline{u}_i) + \frac{1}{2!}\delta u_i^2\Phi''(\overline{u}_i) + \frac{1}{3!}\delta u_i^3\Phi'''(\overline{u}_i) + \cdots, \\ \Phi(u_{i-1}) &= \Phi(\overline{u}_i) - \delta u_i\Phi'(\overline{u}_i) + \frac{1}{2!}\delta u_i^2\Phi''(\overline{u}_i) - \frac{1}{3!}\delta u_i^3\Phi'''(\overline{u}_i) + \cdots,\end{aligned}$$

where $\delta u_i = \frac{1}{2}(u_i - u_{i-1})$. Computing the difference, we have

$$\begin{aligned}D\Phi(u_i, u_{i-1}) &= \frac{\Phi(u_i) - \Phi(u_{i-1})}{2\delta u_i} = \Phi'(\overline{u}_i) + \frac{1}{3!}\delta u_i^2\Phi'''(\overline{u}_i) + \cdots \\ &= \Phi'(\overline{u}_i) + \sum_{p=1}^{\infty}\frac{\delta u_i^{2p}}{(2p+1)!}\,\Phi^{(2p+1)}(\overline{u}_i).\end{aligned}$$

Since $\delta u_i = \frac{1}{2}r_i\Delta t\overline{v}_i$, we can modify the secant-difference as

$$D\Phi(u_i, u_{i-1}) = \Phi'(\overline{u}_i) + \sum_{p=1}^{\infty}\frac{r_i^{2p}\Delta t^{2p}}{2^{2p}(2p+1)!}\overline{v}_i^{2p}\,\Phi^{(2p+1)}(\overline{u}_i)$$

and it means that $D\Phi(u_i, u_{i-1})$ depends on $\overline{\mathbf{w}}_i = (\overline{u}_i, \overline{v}_i)$. Thus, we can rewrite it in the forms of (13) and (14). □

Summing (13) up to the i-th stage, the difference between i-th stage and initial-stage approximations can be written as a linear combination of the previous stages:

$$\mathbf{w}_i - \mathbf{w}_0 = \Delta t \sum_{j=1}^{i} r_j \mathbf{F}_j(\overline{\mathbf{w}}_j), \tag{15}$$

and the approximation at midpoints $\overline{\mathbf{w}}_i$ can be written as follows,

$$\overline{\mathbf{w}}_i = \mathbf{w}_0 + r_1 \Delta t \mathbf{F}_1(\overline{\mathbf{w}}_1) + r_2 \Delta t \mathbf{F}_2(\overline{\mathbf{w}}_2) + \cdots + \frac{r_i}{2} \Delta t \mathbf{F}_i(\overline{\mathbf{w}}_i). \tag{16}$$

We write this procedure (16) in a matrix-wise form

$$\begin{bmatrix} \overline{\mathbf{w}}_1 \\ \overline{\mathbf{w}}_2 \\ \vdots \\ \overline{\mathbf{w}}_s \end{bmatrix} = \mathbf{1}\mathbf{w}_0 + \Delta t \begin{bmatrix} \frac{r_1}{2} & 0 & \cdots & 0 \\ r_1 & \frac{r_2}{2} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ r_1 & r_2 & \cdots & \frac{r_s}{2} \end{bmatrix} \begin{bmatrix} \mathbf{F}_1(\overline{\mathbf{w}}_1) \\ \mathbf{F}_2(\overline{\mathbf{w}}_2) \\ \vdots \\ \mathbf{F}_s(\overline{\mathbf{w}}_s) \end{bmatrix}, \tag{17}$$

where $\mathbf{1}$ is the s-by-1 constant vector having 1 and the matrix multiplication applies to each of components in the (u, v)-space. Since $\mathbf{w}^{n+1} = \mathbf{w}_s$ and $\mathbf{w}^n = \mathbf{w}_0$, (15) implies that

$$\mathbf{w}^{n+1} = \mathbf{w}^n + \Delta t \sum_{j=1}^{s} r_j \mathbf{F}_j(\overline{\mathbf{w}}_j). \tag{18}$$

For notational convenience, we collect the stage vectors into a block vector in $(V^2)^s$ and define as

$$\mathcal{W} = \left[\overline{\mathbf{w}}_j\right]_{j=1}^{s} \text{ and } \mathcal{F}(\mathcal{W}) = \left[\mathbf{F}_j(\overline{\mathbf{w}}_j)\right]_{j=1}^{s}.$$

Then, (17) and (18) can be conveniently summarized by the following tableau:

$$\begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{b}^T \end{array} = \begin{array}{c|cccccc} c_1 & \frac{r_1}{2} & 0 & 0 & \cdots & 0 \\ c_2 & r_1 & \frac{r_2}{2} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ c_s & r_1 & r_2 & r_3 & \cdots & \frac{r_s}{2} \\ \hline & r_1 & r_2 & r_3 & \cdots & r_s \end{array}, \tag{19}$$

where $\mathbf{A} \in \mathbb{R}^{s\times s}$, $\mathbf{b} \in \mathbb{R}^s$, and $\mathbf{c} = \mathbf{A}\mathbf{1}$ with $\mathbf{1} = (1, 1, \dots, 1)^T \in \mathbb{R}^s$. We compute the intermediate stage approximations

$$\mathcal{W} = \mathbf{1}\mathbf{w}^n + \Delta t \mathbf{A} \mathcal{F}(\mathcal{W}), \tag{20}$$

and evaluate the next time approximation $\mathbf{w}^{n+1} = \mathbf{w}^n + \Delta t \mathbf{b}^T \mathcal{F}(\mathcal{W})$. We interpret matrix-vector operations between real matrices and block vectors in the natural blockwise manner: $(A\mathbf{Z})_i = \sum_{j=1}^{s} a_{ij} Z_j$, for $A = (a_{ij}) \in \mathbb{R}^{s\times s}$ and $\mathbf{Z} = [Z_j]_{j=1}^{s} \in (V^2)^s$. The tableau in (19) has the same structure as the classical Butcher tableau for RK methods. However, the present SSD scheme is not an RK method since the function $\mathbf{F}_j$ differs across the j-stages.

To develop order conditions of $R_s = \left[r_1, r_2, \cdots, r_s\right]$ for a SSD method up to fourth order, we write $\mathbf{F}_i(\overline{\mathbf{w}}_i)$ of (14) in the vector form

$$\mathcal{F}(\mathcal{W}) = \mathbf{f}_0(\mathcal{W}) + \Delta t^2 \mathbf{b}^2 \mathbf{f}_2(\mathcal{W}) + \Delta t^4 \mathbf{b}^4 \mathbf{f}_4(\mathcal{W}) + \cdots, \tag{21}$$

where vectors listed together denotes component-wise multiplication in the Hadamard product sense: $(\mathbf{b}\mathbf{Z})_i = b_i Z_i$, for $\mathbf{b} = (b_i)_{i=1}^{s} \in \mathbb{R}^s$ and $\mathbf{Z} = [Z_i]_{i=1}^{s} \in (V^2)^s$.

To establish the order conditions for SSD methods, we compare the Taylor expansion of the exact solution with the Taylor expansion of the approximation after one step. This procedure is quite similar to derivation of the order conditions for a classical RK method although extra terms such as $\mathbf{f}_2(\mathcal{W})$, $\mathbf{f}_4(\mathcal{W})$ make the computation far more complex. Instead of detailing the derivation process for each order of accuracy, which can be tedious and cumbersome, we will summarize only the final derivation process for the fourth-order accuracy for simplicity.

Since the system can be represented as $\partial_t \mathbf{w} = \mathbf{f}_0(\mathbf{w}) = (v, -\mathcal{G}_V(\mathcal{L}u + \Phi'(u)))$ for $\mathbf{w} = (u, v)$, the Taylor expansion of the exact solution is

$$\mathbf{w}^{n+1} = \mathbf{w}^n + \Delta t \partial_t \mathbf{w}(t^n) + \frac{1}{2}\Delta t^2 \partial_t^2 \mathbf{w}(t^n) + \frac{1}{6}\Delta t^3 \partial_t^3 \mathbf{w}(t^n) + \frac{1}{24}\Delta t^4 \partial_t^4 \mathbf{w}(t^n) + O\left(\Delta t^5\right), \tag{22}$$

where

$$\partial_t \mathbf{w}(t^n) = \mathbf{f}_0,\ \partial_t^2 \mathbf{w}(t^n) = \mathbf{f}_0 \mathbf{f}_0',\ \partial_t^3 \mathbf{w}(t^n) = \mathbf{f}_0 \mathbf{f}_0 \mathbf{f}_0'' + \mathbf{f}_0 \mathbf{f}_0' \mathbf{f}_0',$$
$$\partial_t^4 \mathbf{w}(t^n) = \mathbf{f}_0 \mathbf{f}_0 \mathbf{f}_0 \mathbf{f}_0''' + 3\mathbf{f}_0(\mathbf{f}_0 \mathbf{f}_0')\mathbf{f}_0'' + \mathbf{f}_0(\mathbf{f}_0 \mathbf{f}_0'')\mathbf{f}_0' + \mathbf{f}_0 \mathbf{f}_0' \mathbf{f}_0' \mathbf{f}_0'.$$

Here, the prime denotes the derivative with respect to $\mathbf{w}$ and we use the abbreviation, $\mathbf{f}_0 = \mathbf{f}_0(\mathbf{w}^n)$, $\mathbf{f}_0' = \mathbf{f}_0'(\mathbf{w}^n)$, $\mathbf{f}_0'' = \mathbf{f}_0''(\mathbf{w}^n)$, and $\mathbf{f}_0''' = \mathbf{f}_0'''(\mathbf{w}^n)$.

We continue to use the row-wise notation, and interpret the product of two vectors as the Hadamard product. All operations involving the vector field $\mathbf{f}_0$ and its derivatives are evaluated in a right-associative manner, where $\mathbf{f}_0'$ is a second-order tensor and

Table 1
Order conditions of SSD methods up to the fourth-order accuracy.

order	1	2	3	4
RK-like conditions	$\mathbf{b}^T\mathbf{1} = 1$	$\mathbf{b}^T\mathbf{c} = \frac{1}{2}$	$\mathbf{b}^T\mathbf{c}^2 = \frac{1}{3}$ $\mathbf{b}^T\mathbf{A}\mathbf{c} = \frac{1}{6}$	$\mathbf{b}^T\mathbf{c}^3 = \frac{1}{4}$ $\mathbf{b}^T(\mathbf{c}\mathbf{A}\mathbf{c}) = \frac{1}{8}$ $\mathbf{b}^T\mathbf{A}\mathbf{c}^2 = \frac{1}{12}$ $\mathbf{b}^T\mathbf{A}^2\mathbf{c} = \frac{1}{24}$
Null conditions			$\mathbf{b}^T\mathbf{b}^2 = 0$	$\mathbf{b}^T\mathbf{A}\mathbf{b}^2 = 0$ $\mathbf{b}^T(\mathbf{b}^2\mathbf{c}) = 0$

higher-order derivatives produce tensors of increasing order. For completeness, we specify a consistent contraction rule for tensors of successive orders: the last index of the left tensor and the first index of the right tensor are summed over. For examples,

$$(a_i)(b_{ijk}) = \left(\sum_i a_i b_{ijk}\right),\ (a_i)(b_{ijkl}) = \left(\sum_i a_i b_{ijkl}\right),\ (a_{ij})(b_{jk}) = \left(\sum_j a_{ij} b_{jk}\right).$$

This rule extends the conventional vector–matrix multiplication to higher-order tensors. The resulting tensor retains all non-contracted indices of the operands, providing a natural hierarchical generalization of the matrix product to higher-order tensors while preserving linearity and dimensional consistency.

Assume that the coefficients satisfy the condition $\mathbf{c} = \mathbf{A}\mathbf{1}$. And we start considering the Taylor expansion of $\mathcal{F}(\mathcal{W})$ at $\mathbf{w}^n$ up to fourth order. Introducing a new variable, $\mathbf{g} := \frac{1}{\Delta t}(\mathcal{W} - \mathbf{1}\mathbf{w}^n) = \mathbf{A}\mathcal{F}(\mathcal{W})$ from (20), $\mathcal{W}$ can be written as $\mathcal{W} = \mathbf{1}\mathbf{w}^n + \Delta t\mathbf{g}$. Substituting this relation into (21) and applying the Taylor expansion to each term yields

$$\mathcal{F}(\mathcal{W}) = \mathbf{1}\mathbf{f}_0 + \Delta t\mathbf{g}\mathbf{f}_0' + \frac{1}{2}\Delta t^2\mathbf{g}\mathbf{g}\mathbf{f}_0'' + \frac{1}{6}\Delta t^3\mathbf{g}\mathbf{g}\mathbf{g}\mathbf{f}_0''' + \Delta t^2\mathbf{b}^2\mathbf{f}_2 + \Delta t^3\mathbf{b}^2\mathbf{g}\mathbf{f}_2' + O(\Delta t^4),$$

where we use the additional abbreviations $\mathbf{f}_2 = \mathbf{f}_2(\mathbf{w}^n)$ and $\mathbf{f}_2' = \mathbf{f}_2'(\mathbf{w}^n)$. As previously mentioned, when vectors are written in sequence, it implies component-wise multiplication. By recursively applying $\mathcal{F}(\mathcal{W})$ up to the fourth order, until only the term involving $\mathbf{w}^n$ remains, we obtain

$$\begin{aligned}\mathcal{F}(\mathcal{W}) &= \mathbf{1}\mathbf{f}_0 + \Delta t\mathbf{c}\mathbf{f}_0\mathbf{f}_0' + \Delta t^2\left(\mathbf{A}\mathbf{c}\mathbf{f}_0\mathbf{f}_0'\mathbf{f}_0' + \frac{1}{2}\mathbf{c}^2\mathbf{f}_0\mathbf{f}_0\mathbf{f}_0''\right) + \Delta t^2\mathbf{b}^2\mathbf{f}_2\\ &+ \Delta t^3\left(\mathbf{A}^2\mathbf{c}\mathbf{f}_0\mathbf{f}_0'\mathbf{f}_0'\mathbf{f}_0' + \frac{1}{2}\mathbf{A}\mathbf{c}^2\mathbf{f}_0(\mathbf{f}_0\mathbf{f}_0'')\mathbf{f}_0' + \mathbf{c}\mathbf{A}\mathbf{c}\mathbf{f}_0(\mathbf{f}_0\mathbf{f}_0')\mathbf{f}_0'' + \frac{1}{6}\mathbf{c}^3\mathbf{f}_0\mathbf{f}_0\mathbf{f}_0\mathbf{f}_0'''\right)\\ &+ \Delta t^3\left(\mathbf{A}\mathbf{b}^2\mathbf{f}_2\mathbf{f}_0' + \mathbf{b}^2\mathbf{c}\mathbf{f}_0\mathbf{f}_2'\right) + O(\Delta t^4).\end{aligned}$$

In expressions involving real-valued coefficient matrices and vectors, matrix multiplication is evaluated prior to vector-level products, in order to avoid ambiguity in expressions such as $\mathbf{c}\mathbf{A}\mathbf{c}$. By evaluating $\mathbf{w}^{n+1} = \mathbf{w}^n + \Delta t\mathbf{b}^T\mathcal{F}(\mathcal{W})$ and comparing into (22), we can derive the fourth-order conditions as

$$\begin{aligned}&\mathbf{b}^T\mathbf{1} = 1, \quad \mathbf{b}^T\mathbf{c} = \frac{1}{2}, \quad \mathbf{b}^T\mathbf{A}\mathbf{c} = \frac{1}{6},\ \mathbf{b}^T\mathbf{c}^2 = \frac{1}{3},\ \mathbf{b}^T\mathbf{b}^2 = 0,\\ &\mathbf{b}^T\mathbf{A}^2\mathbf{c} = \frac{1}{24},\ \mathbf{b}^T\mathbf{A}\mathbf{c}^2 = \frac{1}{12},\ \mathbf{b}^T(\mathbf{c}\mathbf{A}\mathbf{c}) = \frac{1}{8},\ \mathbf{b}^T\mathbf{c}^3 = \frac{1}{4},\ \mathbf{b}^T\mathbf{A}\mathbf{b}^2 = 0,\ \mathbf{b}^T(\mathbf{b}^2\mathbf{c}) = 0.\end{aligned}$$

Table 1 systematically organizes the conditions for the SSD method up to fourth-order accuracy. Except for the condition where the value is zero, all other conditions are the same as the accuracy conditions of the RK method. Thus, the categories are labeled as RK-like and null conditions. For comparison, please refer to [33,34].

There are total 11 (8 RK-like and 3 Null) conditions for the SSD methods up to fourth order, however, the following Lemmas can be used to reduce the number of the order conditions significantly.

Lemma 4. *The tables* $\mathbf{A}, \mathbf{b}$*, and* $\mathbf{c}$ *in* (19) *satisfy the following identity,*

$$\mathbf{b}^T\mathbf{A} = (\mathbf{b}^T\mathbf{1})\mathbf{b}^T - (\mathbf{b}\mathbf{c})^T.$$

Therefore, the consistency condition $\mathbf{b}^T\mathbf{1} = 1$ *implies* $\mathbf{b}^T\mathbf{A} = \mathbf{b}^T - (\mathbf{b}\mathbf{c})^T$.

Proof. Defining $l_{ij} = 1$ for $i > j$, $\delta_{ij} = 1$ for $i = j$, and 0 otherwise, we can represent as $\mathbf{A} = [a_{ij}]$ where $a_{ij} = l_{ij}r_j + \frac{1}{2}\delta_{ij}r_j$. Then we get the following,

$$\begin{aligned}(\mathbf{b}^T\mathbf{A})_j &= \sum_{i=1}^{s} r_i\left(l_{ij}r_j + \frac{1}{2}\delta_{ij}r_j\right)\\ &= \left(\sum_{i>j} r_i + \frac{1}{2}r_j\right)r_j = \left(\sum_{i=1}^{s} r_i - c_j\right)r_j.\end{aligned}$$

□

Corollary 5. *The following equations hold for the tables* $\mathbf{A}, \mathbf{b}$*, and* $\mathbf{c}$ *in* (19) *satisfying the lower-order conditions. So we can eliminate 4 RK-like and 1 null conditions out of 11 fourth-order SSD conditions.*

$$\begin{aligned}
\mathbf{b}^T\mathbf{c} = \mathbf{b}^T\mathbf{A}\mathbf{1} &\equiv (\mathbf{b}^T\mathbf{1})^2 - (\mathbf{bc})^T\mathbf{1} & \therefore\ \mathbf{b}^T\mathbf{1} = \pm 1 \ &\leftrightarrow\ \mathbf{b}^T\mathbf{c} = 1/2\\
\mathbf{b}^T\mathbf{A}\mathbf{c} &\equiv 1/2 - (\mathbf{bc})^T\mathbf{c} & \therefore\ \mathbf{b}^T\mathbf{A}\mathbf{c} = 1/6 \ &\leftrightarrow\ \mathbf{b}^T\mathbf{c}^2 = 1/3\\
\mathbf{b}^T\mathbf{A}\mathbf{c}^2 &\equiv 1/3 - (\mathbf{bc})^T\mathbf{c}^2 & \therefore\ \mathbf{b}^T\mathbf{A}\mathbf{c}^2 = 1/12 \ &\leftrightarrow\ \mathbf{b}^T\mathbf{c}^3 = 1/4\\
\mathbf{b}^T\mathbf{A}^2\mathbf{c} &\equiv 1/6 - (\mathbf{bc})^T\mathbf{A}\mathbf{c} & \therefore\ \mathbf{b}^T\mathbf{A}^2\mathbf{c} = 1/24 \ &\leftrightarrow\ \mathbf{b}^T(\mathbf{cAc}) = 1/8\\
\mathbf{b}^T\mathbf{A}\mathbf{b}^2 &\equiv \mathbf{b}^T\mathbf{b}^2 - (\mathbf{bc})^T\mathbf{b}^2 & \therefore\ \mathbf{b}^T\mathbf{A}\mathbf{b}^2 = 0 \ &\leftrightarrow\ \mathbf{b}^T\left(\mathbf{b}^2\mathbf{c}\right) = 0 \quad \textit{if} \quad \mathbf{b}^T\mathbf{b}^2 = 0
\end{aligned}$$

Lemma 6. *The tables* $\mathbf{A}, \mathbf{b}$*, and* $\mathbf{c}$ *in* (19) *satisfy the following equality,*

$$2\mathbf{A}\mathbf{c} = \mathbf{c}^2 + \frac{1}{4}\mathbf{b}^2.$$

Proof. The i-th components of $\mathbf{b}$, $\mathbf{c}$, and $\mathbf{Ac}$ can be expressed as follows:

$$b_i = r_i, \quad c_i = \sum_{j=1}^{i-1} r_j + \frac{1}{2}r_i, \quad (\mathbf{Ac})_i = \sum_{j=1}^{i-1} r_j c_j + \frac{1}{2}r_i c_i.$$

Clearly, $2(\mathbf{Ac})_1 = r_1 c_1 = c_1^2 + \frac{1}{4}r_1^2$. Let us use a mathematical induction starting with

$$2(\mathbf{Ac})_i = 2\sum_{j=1}^{i-1} r_j c_j + r_i c_i = c_i^2 + \frac{1}{4}r_i^2.$$

Because $c_i + \frac{1}{2}r_i = c_{i+1} - \frac{1}{2}r_{i+1}$, we have

$$\begin{aligned}
2(\mathbf{Ac})_{i+1} &= 2\sum_{j=1}^{i} r_j c_j + r_{i+1}c_{i+1} = 2(\mathbf{Ac})_i + r_i c_i + r_{i+1}c_{i+1}\\
&= \left(c_i + \frac{1}{2}r_i\right)^2 + r_{i+1}c_{i+1}\\
&= \left(c_{i+1} - \frac{1}{2}r_{i+1}\right)^2 + r_{i+1}c_{i+1} = c_{i+1}^2 + \frac{1}{4}r_{i+1}^2.
\end{aligned}$$

□

Corollary 7. *For the tables* $\mathbf{A}, \mathbf{b}$*, and* $\mathbf{c}$ *in* (19)*, the third- and fourth-order RK-like conditions imply the third- and fourth-order null conditions, respectively.*

$$\begin{aligned}
\mathbf{b}^T\mathbf{A}\mathbf{c} = 1/6,\ \mathbf{b}^T\mathbf{c}^2 = 1/3 \quad &\rightarrow \quad \mathbf{b}^T\mathbf{b}^2 = 8\mathbf{b}^T\mathbf{A}\mathbf{c} - 4\mathbf{b}^T\mathbf{c}^2 = 0\\
\mathbf{b}^T\mathbf{A}^2\mathbf{c} = 1/24,\ \mathbf{b}^T\mathbf{A}\mathbf{c}^2 = 1/12 \quad &\rightarrow \quad \mathbf{b}^T\mathbf{A}\mathbf{b}^2 = 8\mathbf{b}^T\mathbf{A}^2\mathbf{c} - 4\mathbf{b}^T\mathbf{A}\mathbf{c}^2 = 0
\end{aligned}$$

Lemma 8. *The tables* $\mathbf{A}, \mathbf{b}$*, and* $\mathbf{c}$ *in* (19) *satisfy the following equality,*

$$\mathbf{b}^T(\mathbf{xAy}) + \mathbf{b}^T(\mathbf{yAx}) = (\mathbf{b}^T\mathbf{x})(\mathbf{b}^T\mathbf{y}),$$

for any column vector $\mathbf{x}$ *and* $\mathbf{y}$*.*

Proof. Using $M_{ij} := l_{ij} + \frac{1}{2}\delta_{ij}$ defined 1 for $i > j$, $1/2$ for $i = j$, and 0 for $i < j$, $\mathbf{Ax}$ can be written as $\mathbf{M}(\mathbf{bx})$. Then for any vector $\mathbf{x}$ and $\mathbf{y}$,

$$\begin{aligned}
\mathbf{b}^T(\mathbf{xAy}) + \mathbf{b}^T(\mathbf{yAx}) &= (\mathbf{bx})^T\mathbf{M}(\mathbf{by}) + (\mathbf{by})^T\mathbf{M}(\mathbf{bx})\\
&= (\mathbf{bx})^T(\mathbf{M} + \mathbf{M}^T)(\mathbf{by}) = (\mathbf{b}^T\mathbf{x})(\mathbf{b}^T\mathbf{y}),
\end{aligned}$$

where $\mathbf{M} + \mathbf{M}^T = \mathbf{1}\mathbf{1}^T$. □

Lemma 9. *A note worthy special case of* Lemma 8 *is* $2\mathbf{b}^T(\mathbf{xAx}) = (\mathbf{b}^T\mathbf{x})^2$*.*

It is also possible to derive the lemma independently using

$$2\mathbf{b}^T(\mathbf{xAx}) - (\mathbf{b}^T\mathbf{x})^2 = \sum_i b_i x_i\left(\sum_j 2a_{ij}x_j - b_j x_j\right) = 2\sum_i b_i x_i\left(\sum_j k_{ij} b_j x_j\right) = 0,$$

where $k_{ij} = 1/2$ for $i > j$, $-1/2$ for $i < j$, and 0 for $i = j$. As a result, we have $\mathbf{b}^T(\mathbf{cAc}) = \frac{1}{2}(\mathbf{b}^T\mathbf{c})^2$.

With the aid of Lemma 4 to Lemma 9, Table 1 can be significantly simplified to Table 2. We only need three order conditions for the fourth order of accuracy.

The specific coefficient tables are discribed as follows. The one stage second-order SSD method can be given with $R_1 = [1]$. There is no two stage third-order method but any three stage SSD method with $R_3(\gamma) = [\gamma_1,\ \gamma,\ \gamma_3]$ satisfying

$$\gamma_1 + \gamma_3 = 1 - \gamma, \quad \gamma_1\gamma_3 = \frac{1 - 3\gamma + 3\gamma^2}{3(1-\gamma)}$$

Table 2
Reduced order conditions of SSD methods up to the fourth-order accuracy.

order	1/2	3	4
RK-like	$\mathbf{b}^T\mathbf{1} = 1$	$\mathbf{b}^T\mathbf{c}^2 = \frac{1}{3}$	$\mathbf{b}^T\mathbf{c}^3 = \frac{1}{4}$

becomes third order when $\gamma < \gamma^* := -2(\sqrt[3]{2} + 1/\sqrt[3]{2} + 1/2)/3$ or $\gamma > 1$. The three stage SSD method with $R_3(\gamma^*)$ satisfies all of the order conditions up to fourth order. For the numerical simulations in the following section, we use $R_1 = [1]$, $R_3(8/5) \approx [1.1985, 1.6000, -1.7985]$, and $R_3(\gamma^*) \approx [1.3512, -1.7024, 1.3512]$ for the second, the third, and the fourth-order SSD methods, respectively.

4. Numerical results up to fourth-order accuracy

For the numerical test, we consider semi-linear wave equations in the form of

$$u_{tt} + \mathcal{G}_V\left(\mathcal{L}u + \Phi'(u)\right) = 0, \tag{23}$$

corresponding to the following Hamiltonian functional

$$\mathcal{H}(u, v) = \frac{1}{2}\|v\|_V^2 + \int_\Omega \left(\frac{1}{2}u\mathcal{L}u + \Phi(u)\right) d\mathbf{x}.$$

The first example is the sine-Gordon equation in inhomogeneous medium,

$$u_{tt} - \nabla \cdot (\mathbf{M}(\mathbf{x})\nabla u) + \sin u = 0$$

with $\mathcal{G}_V = I$, $\mathcal{L}u = -\nabla \cdot (\mathbf{M}(\mathbf{x})\nabla u)$ for a given positive definite matrix $\mathbf{M}(\mathbf{x})$, and $\Phi(u) = 1 - \cos u$. The second example is the Boussinesq equation,

$$u_{tt} + \Delta\left(\Delta u - u + 3u^2\right) = 0 \tag{24}$$

with $\mathcal{G}_V = -\Delta$, $\mathcal{L}u = -\Delta u$, and $\Phi(u) = \frac{1}{2}u^2 - u^3$.

The proposed SSD method can be coupled with any type of space discretization methods. The spatial discretization of initial-boundary differential equations presents various challenges, whether using finite difference, finite element, or finite volume methods. These challenges include considerations for high-order spatial accuracy, staggered grids, local discontinuities, and more. To focus only on the temporal order of convergence and energy conservation, we use the Fourier (including Fourier-cosine or Fourier-sine) spectral method for spatial discretization in this paper, which provides fully resolved spatial accuracy for the problems with periodic, zero Neumann, or zero Dirichlet boundary conditions.

We first represent the scheme (12) as follows:

$$\begin{aligned}
\frac{u_i - u_{i-1}}{r_i\Delta t} &= \frac{v_i + v_{i-1}}{2},\\
\frac{v_i - v_{i-1}}{r_i\Delta t} &= -\mathcal{G}_V\left(\mathcal{L}\frac{u_i + u_{i-1}}{2} + \Psi\left(u_i, u_{i-1}\right)\right),
\end{aligned} \tag{25}$$

where

$$\Psi\left(u_i, u_{i-1}\right) = \frac{\Phi\left(u_i\right) - \Phi\left(u_{i-1}\right)}{u_i - u_{i-1}}.$$

Now, we rearrange (25) as

$$u_i + \frac{1}{4}r_i^2\Delta t^2\mathcal{G}_V\left(\mathcal{L}u_i + 2\Psi\left(u_i, u_{i-1}\right)\right) = \varphi_i, \tag{26}$$

where $\varphi_i = u_{i-1} + r_i\Delta t v_{i-1} - \frac{1}{4}r_i^2\Delta t^2\mathcal{G}_V\mathcal{L}u_{i-1}$. Employing a Newton-type iteration method [26,27], we have a nonlinear iterative solver for each i-th stage starting with the previous solution $u_i^{(0)} = u_{i-1}$:

$$\begin{aligned}
&u_i^{(m+1)} + \frac{1}{4}r_i^2\Delta t^2\mathcal{G}_V\left(\mathcal{L}u_i^{(m+1)} + 2\Psi'\left(u_i^{(m)}, u_{i-1}\right)u_i^{(m+1)}\right)\\
&\quad = \varphi_i - \frac{1}{2}r_i^2\Delta t^2\mathcal{G}_V\left(\Psi\left(u_i^{(m)}, u_{i-1}\right) - u_i^{(m)}\Psi'\left(u_i^{(m)}, u_{i-1}\right)\right),
\end{aligned}$$

where $\Psi'\left(u, u_{i-1}\right)$ is a derivative with respect to u while keeping u_{i-1} fixed.

It is worth to note that the solvability of the system depends on the invertability of the operator $I + \frac{1}{4}r_i^2\Delta t^2\mathcal{G}_V(\mathcal{L} + 2\Phi'')$ since $\Psi'\left(u_i^{(m)}, u_{i-1}\right)$ approaches to $\Phi''(u_i)$. Therefore, we can say that the Newton-type system can be inverted if a CFL-like condition holds.

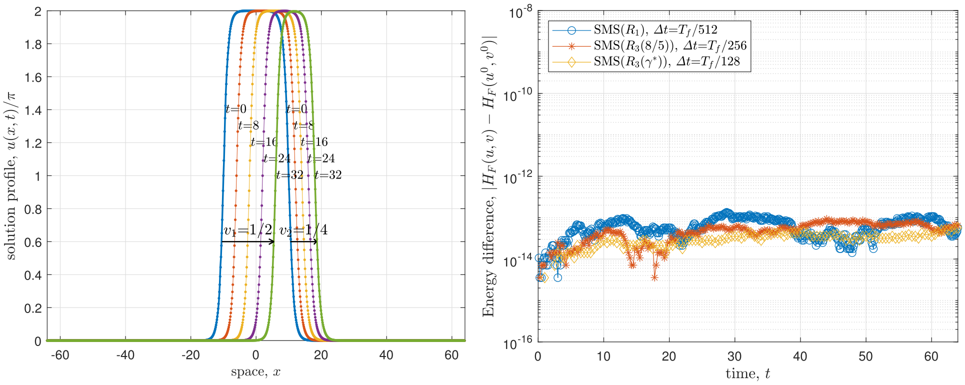

Fig. 1. Evolution of the solution before the collision of two wave forms at $t = 0, 8, 16, 24, 32$ and computational errors of $\mathcal{H}_F(u, v)$ for $t \le 64$.

We use a bi-conjugate gradient (BICG) method to solve the system and we employ the following pre-conditioner P to accelerate the convergence speed of the BICG iteration:

$$P = I + \frac{1}{4} r_i^2 \Delta t^2 \mathcal{G}_V \mathcal{L}.$$

We terminate the m-iteration when the relative consecutive error of $u_i^{(m+1)}$ is less than $tol = 10^{-10}$ and then define $u_i = u_i^{(m+1)}$. Finally, we have

$$v_i = 2\frac{u_i - u_{i-1}}{r_i \Delta t} - v_{i-1}.$$

After recursively obtaining all intermediates u_i and v_i for $i = 1, 2, \cdots, s$, we can determine the next time step approximations as $u^{n+1} = u_s$ and $v^{n+1} = v_s$.

4.1. Sine-Gordon equation in 1D homogeneous medium

We consider the sine-Gordon equation $u_{tt} = \Delta u - \sin u$ in the homogenous medium $\mathbf{M}(\mathbf{x}) = \mathbf{I}$, for the corresponding energy $\mathcal{E}(u) = \int_\Omega \left(\frac{1}{2}|\nabla u|^2 + \Phi(u)\right) d\mathbf{x}$ with $\Phi(u) = 1 - \cos u$. For the numerical test, we consider a superposition of two soliton-like traveling solutions: one rising and the other falling, which initially travel to the same direction but with different speeds,

$$u(x,t) = 4\tan^{-1}\left(e^{(x-x_1-v_1 t)/\sqrt{1-v_1^2}}\right) - 4\tan^{-1}\left(e^{(x-x_2-v_2 t)/\sqrt{1-v_2^2}}\right),$$

where $(x_1, v_1) = (-10, 0.5)$ and $(x_2, v_2) = (10, 0.25)$. We set a computational domain $\Omega = [-64, 64]$ with zero Neumann boundary condition.

Fig. 1 shows the time evolution of the solution by the fourth-order method SSD($R_3(\gamma^*)$) and difference of the Hamiltonian functional from the initial value up to $T_f = 64$. We choose the number of spatial discretization $N_x = 512$ and the relatively small time step $\Delta t = T_f/2^{12}$ to provide about 7 digits of spatial and temporal accuracy for the reference solution shown in the figure. The method, even for relatively large time steps $\Delta t = T_f/2^7, \cdots, T_f/2^9$, preserves the energy close to the machine precision, which is independent to the solution accuracy.

Fig. 2 shows that all SSD methods with $\Delta t = T_f/2^{11}, \cdots, T_f/2^6$ provide the desired temporal order of accuracy for the computed solution at $t = 16$ and $t = 48$. We remark that the third-order SMS($R_3(8/5)$) method shows super-convergence as two wave fronts emerge, which is a common phenomenon for a third-order numerical method to achieve slightly better order of accuracy for a solution with even symmetry.

We continue to evolve the system up to $T_f = 256$ to observe the waveform interactions with the other waveform and the Neumann boundary at $x = 64$. Fig. 3 shows some of snapshots of the solution at $t = 32, 64, \cdots, 256$. The raising soliton of velocity $v_1 = 0.5$ follows the falling soliton of velocity $v_2 = 0.25$ when $t < 64$ and the raising soliton starts to lead the falling when $t > 64$. The raising and the falling solitons reach the Neumann wall and move to the left after reflections during $128 < t < 224$.

Fig. 4 illustrates the time evolution shown in Fig. 3 represented on the $x - t$ plane, where the moving velocities can be easily visualized. The left side displays the 3D perspective of the solution evolution over time, while the right side provides a 2D heat map highlighting the speed variation of the waves. We recognize the speed of the raising and the falling solitons are about constant $v_1 = 0.5$, $v_2 = 0.25$ and reflected by the Neumann wall at $t \approx 137$, $t \approx 215$, respectively. It is worth to note that the raising soliton gains the speed and the falling solition loose the speed during the collision period around $t \approx 64$, while the speeds are restored back to the original values after the interaction.

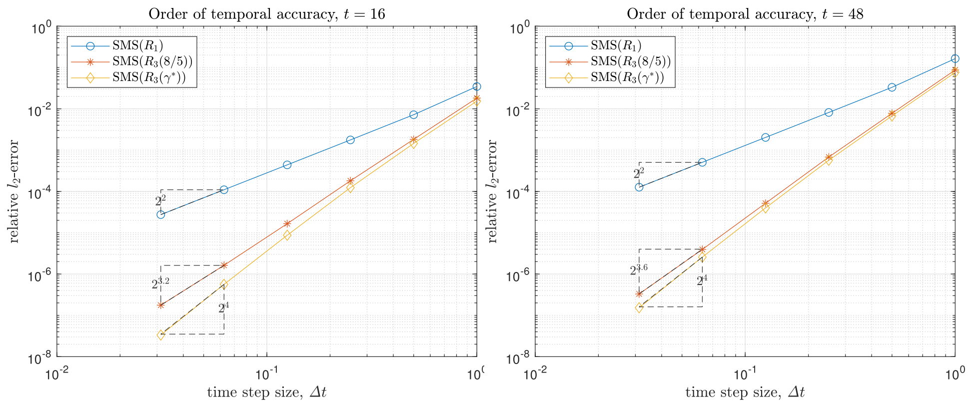

Fig. 2. Convergence of the solution at $t = 16, 48$ by various SSD methods.

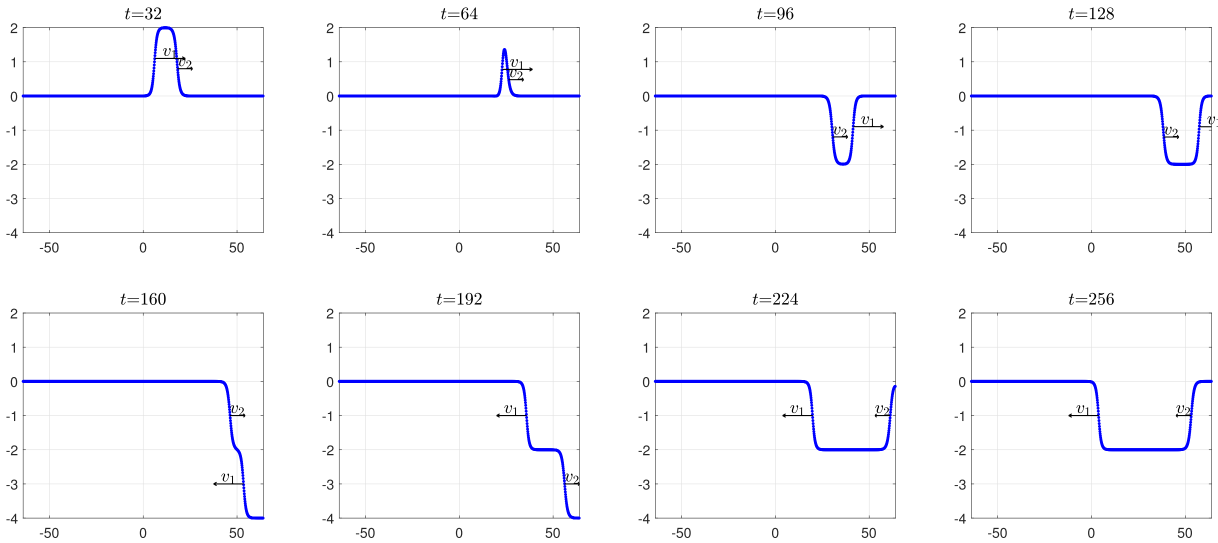

Fig. 3. Interaction of two wave fronts ($32 < t \le 128$) and reflection at the Neumann boundary ($128 < t \le 256$).

4.2. Sine-Gordon equation in 1D heterogeneous medium

We now consider a sine-Gordon equation with $\mathbf{M}(\mathbf{x}) = 1 + \frac{\alpha}{1+e^{-5\mathbf{x}}}$,

$$u_{tt} - \nabla \cdot (\mathbf{M}(\mathbf{x})\nabla u) + \sin u = 0,$$

for which the corresponding energy functional $\mathcal{E}(u) = \int_\Omega \left(\frac{1}{2}|\mathbf{M}(\mathbf{x})\nabla u|^2 + \Phi(u)\right) d\mathbf{x}$ with $\Phi(u) = 1 - \cos u$. For the numerical test, we consider a superposition of two soliton-like traveling solutions initially moving in the same velocity,

$$u(x,t) = 4\tan^{-1}\left(e^{(x-x_1-v_1 t)/\sqrt{1-v_1^2}}\right) - 4\tan^{-1}\left(e^{(x-x_2-v_2 t)/\sqrt{1-v_2^2}}\right),$$

where $(x_1, v_1) = (-30, 0.5)$ and $(x_2, v_2) = (-10, 0.5)$. We set a computational domain $\Omega = [-64, 64]$ with the periodic boundary condition.

Fig. 5 shows the evolution of the solution for $\alpha = 0.2$ (or $\mathbf{M}(64) \approx 1.2$) computed by the fourth-order method with $N_x = 768$, $\Delta t = 0.1$ up to $t \le 100$. We can easily observe how the velocities of the two soliton-like waves change as they pass the interface $x \approx 0$. It is worth to note that travelling velocities of two wave fronts after passing the green line become about 0.33 for $t > 25$ and $t > 65$, respectively.

Fig. 6 demonstrates the spatial and temporal accuracy with relative l_2-error. The left plot shows the high spectral accuracy of the fourth-order SSD($R_3(\gamma^*)$) plotted against the number of spatial points N_x. The right plot illustrates the temporal order of accuracy plotted against the time step size Δt. This figure clearly shows that the SSD methods achieve the expected order of time convergence, or even slightly better for third-order SSD($R_3(8/5)$).

Fig. 6 demonstrates the spatial and temporal accuracy with relative l_2-error. The left plot shows the high spatial accuracy of the fourth-order SSD($R_3(\gamma^*)$) plotted against the number of spatial points N_x. The right plot illustrates the temporal order of accuracy plotted against the time step size Δt. This figure clearly shows that the SSD methods achieve the expected order of time convergence, or even slightly better for third-order SSD($R_3(8/5)$).

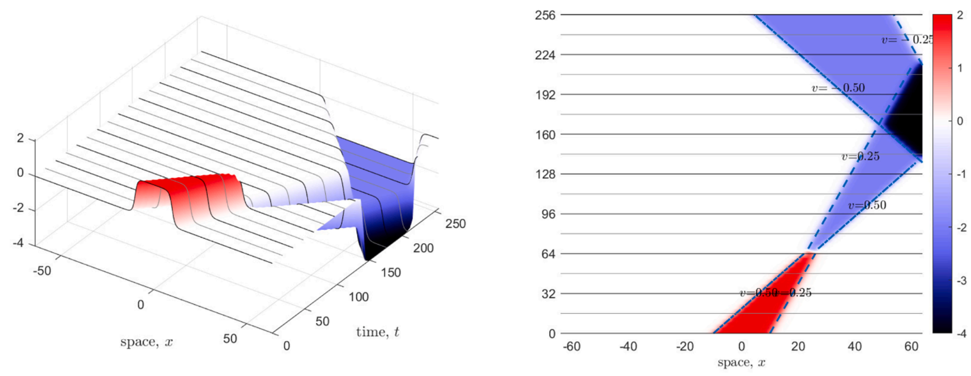

Fig. 4. Evolution of the solution in homogeneous medium, $v_1 = 1/2$, $v_2 = 1/4$.

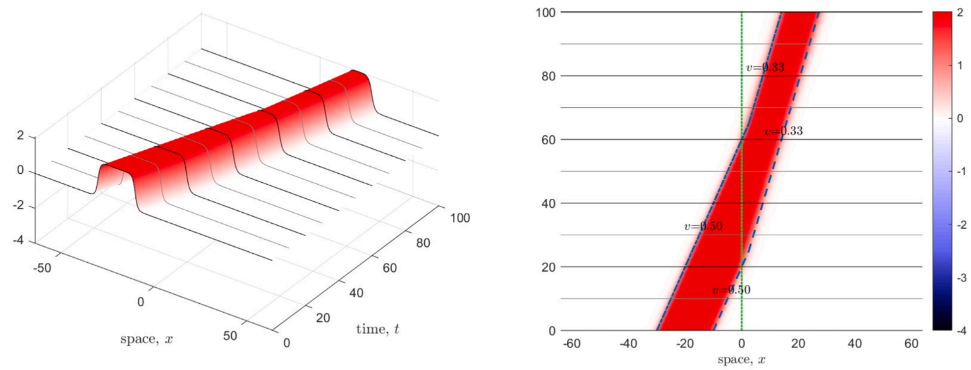

Fig. 5. Evolution of the solution in heterogeneous medium with $\alpha = 0.2$.

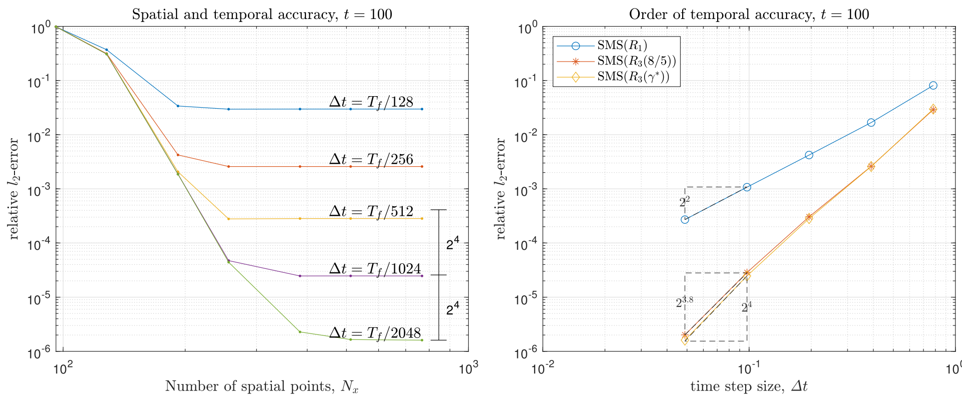

Fig. 6. Convergence of the solution at $t = 100$ in heterogeneous medium with $\alpha = 0.2$.

As increasing the variation of the values of $\mathbf{M}(\mathbf{x})$, say $\alpha = 0.4$, the falling and raising wave fronts are reflected near the interface at $t \approx 25$ and $t \approx 65$, respectively. Fig. 7 clearly shows the reflected waveforms moving in the opposite direction with the same velocity. This figure illustrates the reference solution obtained using the fourth-order method while keeping all other conditions the same with Fig. 5.

4.3. A Boussinesq-Type equation in 1D

The Boussinesq-type equation in 1D, $u_{tt} + \left(u_{xx} + 3u^2 - u\right)_{xx} = 0$ is a spatially fourth-order partial differential equation [35] in the form of (23), which corresponds to the Hamiltonian functional

$$\mathcal{H}_{BS}(u, v) = \frac{1}{2}\|v\|^2_{H^{-1}} + \int_\Omega \left(\frac{1}{2}\|\nabla u\|^2 + \Phi(u)\right) d\mathbf{x} \tag{27}$$

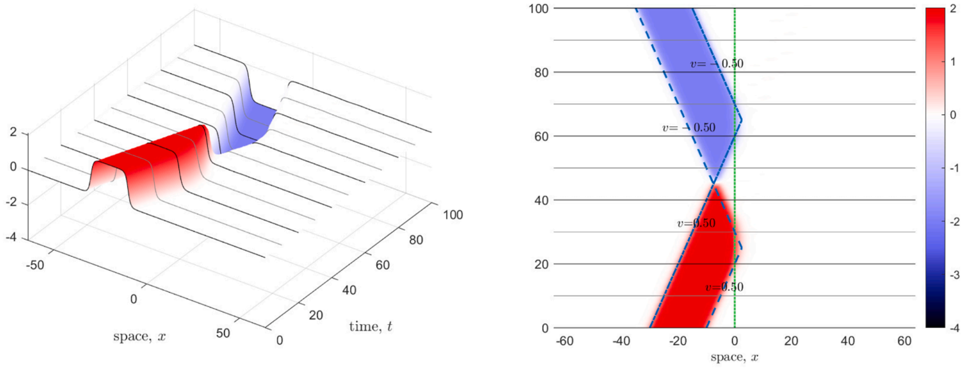

Fig. 7. Evolution of the solution in heterogeneous medium with $\alpha = 0.4$.

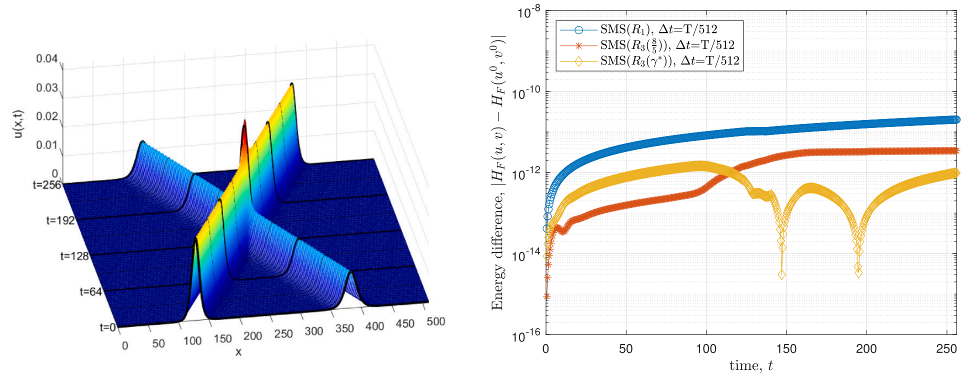

Fig. 8. Evolution of the solution and computational errors of $\mathcal{H}_F(u, v)$.

with $\Phi(u) = \frac{1}{2}u^2 - u^3$. In this subsection, we consider two 1D solitary wave packets in the form of $2b_i^2 \operatorname{sech}^2(b_i(x - x_i - c_i t))$ moving in opposite directions with slightly different speed, $(b_1, x_1) = (0.12, 128)$, $(b_2, x_2) = (0.08, 384)$, $c_i = (-1)^{i-1}\sqrt{1 - 4b_i^2}$. A true analytic solution is given [17] as follows,

$$u(x,t) = 2\frac{d^2}{dx^2}\log\left(1 + e^{\eta_1(x,t)} + e^{\eta_2(x,t)} + Ae^{\eta_1(x,t)+\eta_2(x,t)}\right)$$

where $\eta_i(x,t) = 2b_i(x - x_i - c_i t)$ and $A = \frac{(c_1-c_2)^2 - 12(b_1-b_2)^2}{(c_1-c_2)^2 - 12(b_1+b_2)^2}$.

For the numerical test, we set a computational domain $\Omega = [0, 512]$ with the periodic boundary condition and choose a number of spatial discretization $N_x = 512$ or $\Delta x = 1$ that provides sufficient spatial accuracy. Fig. 8 shows the computational solution with $\Delta t = 0.5$ up to $T_f = 256$ using the fourth-order SSD($R_3(\gamma^*)$) method with the initial conditions $u^0 = u(\cdot, 0)$ and $v^0 = u_t(\cdot, 0)$. The SSD method is an energy conserving method, thus energy difference is conserved independent to the solution accuracy and the difference in the right plot is merely a result of accumulation of the round off errors.

Fig. 9 shows spatial and temporal discretization errors by the various methods compared with the analytic solution at $t = 128$ when two solitons collide. The left plot shows that the computational solution by the SSD($R_3(\gamma^*)$) provides more than single precision spatial accuracy for $N_x \geq 256$ and converges in the fourth order of temporal accuracy. The right plot shows that the SSD methods provide the desired order of accuracy at $t = 128$ and we remark that the rate of the SSD($R_3(8/5)$) at $t = 256$ is about 3.2 which is slightly better than expected value due to the small leading order error coefficient after separation of the solitons.

4.4. A Boussinesq-Type equation in 2D

We consider the Boussinesq-type equation,

$$u_{tt} + \Delta\left(\Delta u + 3u^2 - u\right) = 0 \tag{28}$$

in a square domain $[-64, 64]^2$. Since there is no known closed form of soliton-like waveform in 2D, we choose a thin rod of length $2l_0$ at center (x_0, y_0) moving θ-direction. Using rotated coordinate

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x - x_0 \\ y - y_0 \end{bmatrix}$$

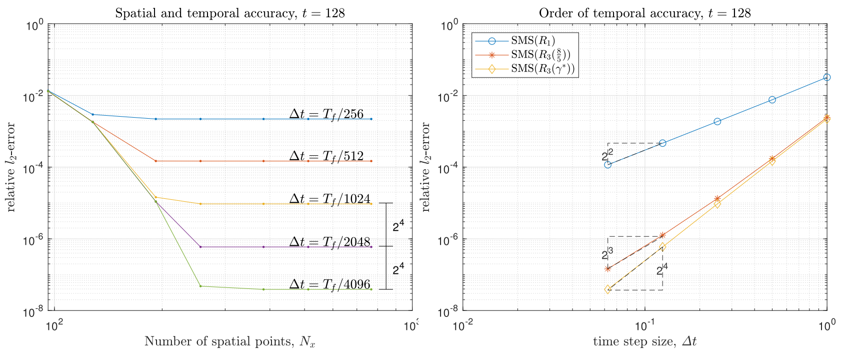

Fig. 9. Spatial and temporal error by the SSD($R_3(\gamma^*)$) and order of temporal convergence by the various SSD methods of the solution at $t = 128$.

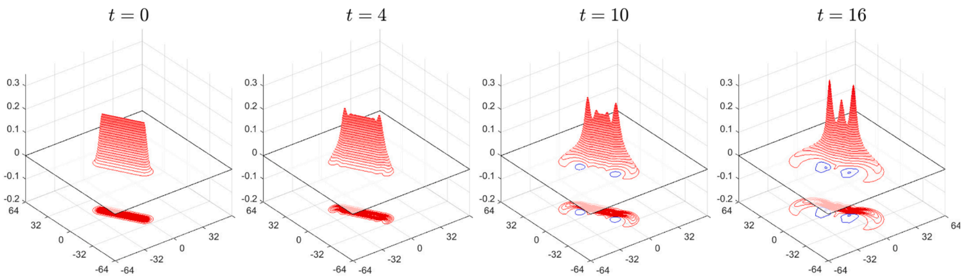

Fig. 10. Evolution of the pressure field.

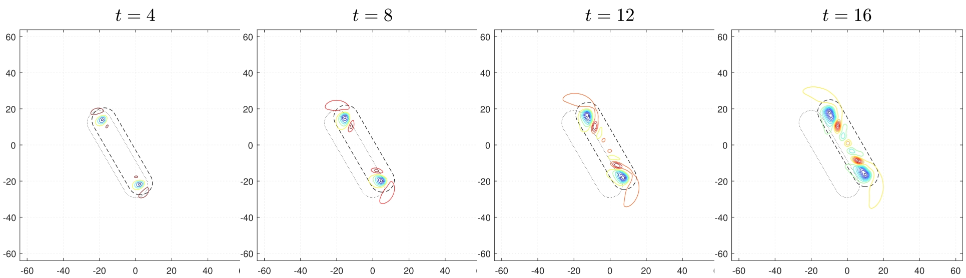

Fig. 11. Contour lines for the difference between the pressure field u and u_{shift}. Dotted line indicates the initial position and dashed line shows u_{shift}.

with $(x_0, y_0) = (-10, -5)$, $\theta = \pi/6$, we define a rod function with rounded tips as follows:

$$u_0(x, y) = \begin{cases} 2b_1^2 \operatorname{sech}^2(bx'), & |y'| \le l_0 \\ 2b_1^2 \operatorname{sech}^2(br'), & |y'| > l_0 \end{cases}$$

where $l_0 = 20$, $b = 1/3$, and $r' = \sqrt{(x')^2 + (|y'| - l_0)^2}$ is distance from the tips of the rod. The time derivative of $u_0(x, y)$ can be computed explicitly by moving $u_0(x, y)$ in θ-direction with velocity $c = \sqrt{1 - 4b^2}$. Since $u_0(x, y)$ is only a C^1-continuous function, we take smothered initial conditions, $u(x, y, t{=}0)$ and $u_t(x, y, t{=}0)$ by convolving them with a Gaussian function of width 2.0. The numerical convolution for the initial data has been taken by multiplying the corresponding Fourier coefficients of $e^{-(x^2+y^2)/4.0}$ with $\Delta x{=}\Delta y{=}1/4$.

Fig. 10 shows the evolution of the solution. At the early stage, the rod moves toward θ-direction as desired but soon the soliton shape becomes broken from near the tips. Fig. 11 plots the difference between the computed solution and the shifted solution u_{shift} (marked with dash lines) toward θ-direction of speed $c = \sqrt{5}/3$ which clearly shows the disturbance emerges from the tip moves in all directions.

In order to see the disturbance effect clearly, we extend the rod length $l_0 = 40, 80, 160$ while keeping the thickness same as the case for $l_0 = 20$. We set $\Delta x = \Delta y = 1$ to get single precision accuracy but make the computational domains larger, $[-128, 128]^2$, $[-192, 192]^2$, and $[-256, 256]^2$, respectively. Fig. 12 shows the difference between the computed and the shifted initial solutions near the lower right tips. The time evolution patterns of the differences look very similar and the relative differences $\frac{\|u_{\text{comp}} - u_{\text{shift}}\|_{L^2(\Omega)}}{\|u_0\|_{L^2(\Omega)}}$ are proportional to $1/\sqrt{l_0}$ since the denominator is proportional to the stick length $\sqrt{l_0}$.

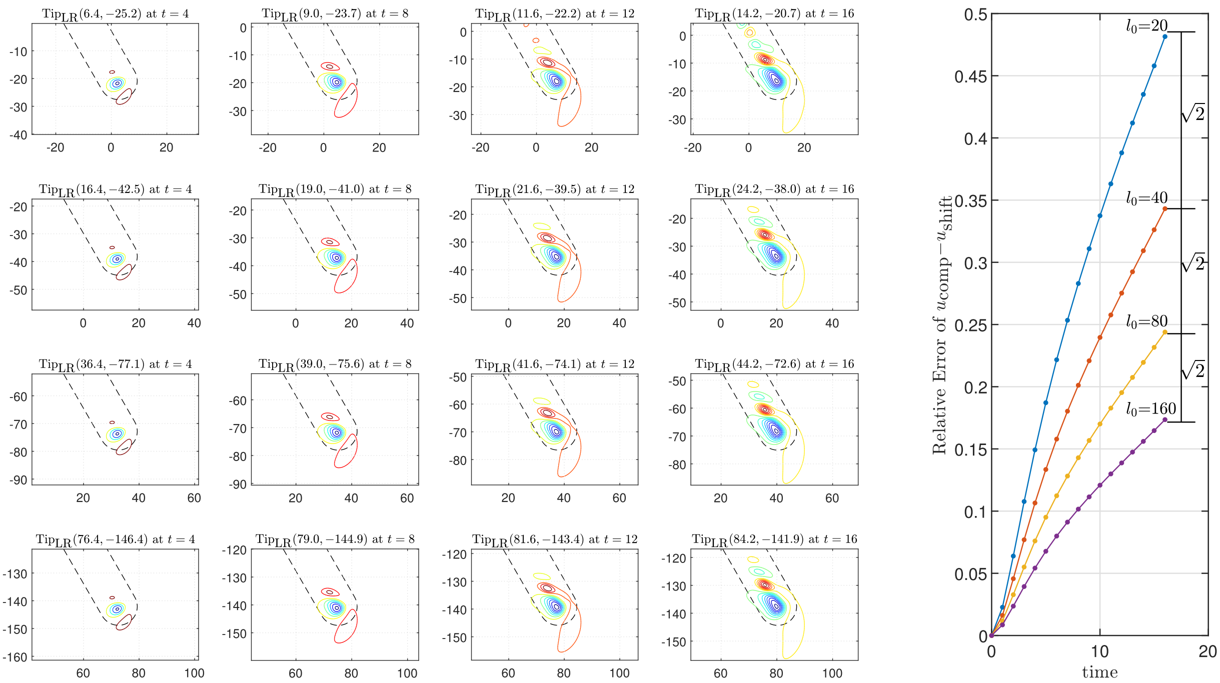

Fig. 12. Error at the lower right tip depending on length $l_0 = 20, 40, 80, 160$.

5. Higher-order SSD methods and numerical examples

In this section, we explore the construction and performance of higher-order energy-preserving methods based on the SSD framework. We first present the complete set of order conditions up to sixth-order accuracy and analyze their linear dependencies to identify the minimal number of independent constraints required at each order. Then, we demonstrate accuracy order of the resulting schemes through numerical experiments on the sine-Gordon equation. These examples highlight the practical effectiveness of our approach in preserving both qualitative and quantitative features of the solution over long simulation times.

5.1. Order conditions up to sixth-order accuracy

Table 3 summarizes all 37 RK conditions up to sixth-order SSD methods. These include 2, 2, 4, 9, and 20 conditions for second through sixth-order accuracy, respectively and equivalent conditions are grouped together. Although the corollaries in Section 4 are not fully derived, the equivalences among conditions can be readily proven: $\overset{\mathrm{L4}}{=}$, $\overset{\mathrm{L8}}{=}$, and $\overset{\mathrm{L9}}{=}$ indicate that each equality is provable with following lemmas,

$$\begin{aligned}
&\text{Lemma 4}: \quad \mathbf{b}^{\mathrm{T}}\mathbf{A} = \mathbf{b}^{\mathrm{T}} - (\mathbf{bc})^{\mathrm{T}},\\
&\text{Lemma 8}: \quad \mathbf{b}^{\mathrm{T}}(\mathbf{xAy}) + \mathbf{b}^{\mathrm{T}}(\mathbf{yAx}) = (\mathbf{b}^{\mathrm{T}}\mathbf{x})(\mathbf{b}^{\mathrm{T}}\mathbf{y}),\\
&\text{Lemma 9}: \quad 2\mathbf{b}^{\mathrm{T}}(\mathbf{xAx}) = (\mathbf{b}^{\mathrm{T}}\mathbf{x})^2.
\end{aligned}$$

For the dependent conditions, 0_{Rp} denotes zero and $\overset{\mathrm{R}p}{=}$ indicates that the equivalency can be established assuming p-th order RK conditions. For an example,

$$[\mathbf{b}^T(\mathbf{cAc}) - \frac{1}{8}] \overset{\mathrm{L9}}{=} 0_{R2}$$

can be shown using Lemma 9 when the second-order conditions hold. One bracket in each independent condition groups must vanish and only three conditions are required to achieve fourth-order accuracy, as shown in Section 3. For fifth- and sixth-order methods, five and seven conditions are required, respectively.

Table 4 summarizes all 25 Null conditions required for SSD methods up to sixth-order accuracy. As with the RK conditions, equivalent Null conditions are grouped together. The equivalency $\overset{\mathrm{L6}}{=}$ can be proved by

$$\text{Lemma 6}: \quad 2\mathbf{Ac} = \mathbf{c}^2 + \frac{1}{4}\mathbf{b}^2.$$

For the dependent conditions, 0_{Np} denotes zero and $\overset{\mathrm{N}p}{=}$ means that the equivalency holds when p-th order Null conditions are satisfied.

Table 3
Runge–Kutta conditions up to sixth-order accuracy.

Rp	Equivalent Independent Order Conditions	Dependent RK Conditions
2	$[(\mathbf{b}^T\mathbf{1})^2 - 1] \overset{\text{L4}}{=} 2[\mathbf{b}^T\mathbf{c} - \frac{1}{2}]$	
3	$[\mathbf{b}^T\mathbf{c}^2 - \frac{1}{3}] \overset{\text{L4}}{=} -[\mathbf{b}^T\mathbf{A}\mathbf{c} - \frac{1}{6}]$	
4	$[\mathbf{b}^T\mathbf{c}^3 - \frac{1}{4}] \overset{\text{L4}}{=} -[\mathbf{b}^T\mathbf{A}\mathbf{c}^2 - \frac{1}{12}]$	$[\mathbf{b}^T\mathbf{A}^2\mathbf{c} - \frac{1}{24}] \overset{\text{L4}}{=} [\mathbf{b}^T(\mathbf{c}\mathbf{A}\mathbf{c}) - \frac{1}{8}] \overset{\text{L9}}{=} 0_{R2}$
5	$-[\mathbf{b}^T\mathbf{A}(\mathbf{c}\mathbf{A}\mathbf{c}) - \frac{1}{40}] \overset{\text{L4}}{=} [\mathbf{b}^T(\mathbf{c}^2\mathbf{A}\mathbf{c}) - \frac{1}{10}]$ $\overset{\text{L8}}{=} -[\mathbf{b}^T(\mathbf{c}\mathbf{A}\mathbf{c}^2) - \frac{1}{15}] \overset{\text{L4}}{=} [\mathbf{b}^T\mathbf{A}^2\mathbf{c}^2 - \frac{1}{60}]$	$[\mathbf{b}^T\mathbf{A}\mathbf{c}^3 - \frac{1}{20}] \overset{\text{L4}}{=} -[\mathbf{b}^T\mathbf{c}^4 - \frac{1}{5}]$ $\overset{\text{R2}}{=} 4[\mathbf{b}^T\mathbf{A}(\mathbf{c}\mathbf{A}\mathbf{c}) - \frac{1}{40}] - [\mathbf{b}^T(\mathbf{A}\mathbf{c})^2 - \frac{1}{20}]$
	$[\mathbf{b}^T(\mathbf{A}\mathbf{c})^2 - \frac{1}{20}]$ $\overset{\text{L8}}{=} -[\mathbf{b}^T(\mathbf{c}\mathbf{A}^2\mathbf{c}) - \frac{1}{30}] \overset{\text{L4}}{=} [\mathbf{b}^T\mathbf{A}^3\mathbf{c} - \frac{1}{120}]$	
6	$-[\mathbf{b}^T\mathbf{A}(\mathbf{A}\mathbf{c})^2 - \frac{1}{120}] \overset{\text{L4}}{=} [\mathbf{b}^T(\mathbf{c}(\mathbf{A}\mathbf{c})^2) - \frac{1}{24}]$ $\overset{\text{L8}}{=} -[\mathbf{b}^T(\mathbf{c}\mathbf{A}(\mathbf{c}\mathbf{A}\mathbf{c})) - \frac{1}{48}]$ $\overset{\text{L4}}{=} [\mathbf{b}^T\mathbf{A}^2(\mathbf{c}\mathbf{A}\mathbf{c}) - \frac{1}{240}]$	$[\mathbf{b}^T\mathbf{A}(\mathbf{c}\mathbf{A}\mathbf{c}^2) - \frac{1}{90}] \overset{\text{L4}}{=} [\mathbf{b}^T(\mathbf{c}^2\mathbf{A}\mathbf{c}^2) - \frac{1}{18}] \overset{\text{L9}}{=} 0_{R3}$ $[\mathbf{b}^T((\mathbf{A}\mathbf{c})\mathbf{A}^2\mathbf{c}) - \frac{1}{72}] \overset{\text{L9}}{=} 0_{R3}$ $[\mathbf{b}^T\mathbf{A}^4\mathbf{c} - \frac{1}{720}] \overset{\text{L4}}{=} [\mathbf{b}^T(\mathbf{c}\mathbf{A}^3\mathbf{c}) - \frac{1}{144}] \overset{\text{L}c}{=} 0_{R3}$
	$-[\mathbf{b}^T\mathbf{A}(\mathbf{c}\mathbf{A}^2\mathbf{c}) - \frac{1}{180}] \overset{\text{L4}}{=} [\mathbf{b}^T(\mathbf{c}^2\mathbf{A}^2\mathbf{c}) - \frac{1}{36}]$ $\overset{\text{L8}}{=} -[\mathbf{b}^T((\mathbf{A}\mathbf{c})\mathbf{A}\mathbf{c}^2) - \frac{1}{36}]$ $\overset{\text{L8}}{=} [\mathbf{b}^T(\mathbf{c}\mathbf{A}^2\mathbf{c}^2) - \frac{1}{72}] \overset{\text{L4}}{=} -[\mathbf{b}^T\mathbf{A}^3\mathbf{c}^2 - \frac{1}{360}]$	$-[\mathbf{b}^T\mathbf{A}\mathbf{c}^4 - \frac{1}{30}] \overset{\text{L4}}{=} [\mathbf{b}^T\mathbf{c}^5 - \frac{1}{6}]$ $\overset{\text{R3}}{=} 9[\mathbf{b}^T(\mathbf{c}(\mathbf{A}\mathbf{c})^2) - \frac{1}{24}] + 2[\mathbf{b}^T(\mathbf{c}^2\mathbf{A}^2\mathbf{c}) - \frac{1}{36}]$ $[\mathbf{b}^T\mathbf{A}^2\mathbf{c}^3 - \frac{1}{120}] \overset{\text{L4}}{=} -[\mathbf{b}^T(\mathbf{c}\mathbf{A}\mathbf{c}^3) - \frac{1}{24}]$ $\overset{\text{L8}}{=} [\mathbf{b}^T(\mathbf{c}^3\mathbf{A}\mathbf{c}) - \frac{1}{12}]\{\overset{\text{L4}}{=} -[\mathbf{b}^T\mathbf{A}(\mathbf{c}^2\mathbf{A}\mathbf{c}) - \frac{1}{60}]\}$ $\overset{\text{R3}}{=} 3[\mathbf{b}^T(\mathbf{c}(\mathbf{A}\mathbf{c})^2) - \frac{1}{24}] + [\mathbf{b}^T(\mathbf{c}^2\mathbf{A}^2\mathbf{c}) - \frac{1}{36}]$

Table 4
Null conditions up to sixth-order accuracy.

Np	Equivalent Independent Zero Terms	Dependent Null Conditions
3	$\mathbf{b}^T\mathbf{b}^2 \overset{\text{L6}}{=} 0_{R3}$	
4	$-\mathbf{b}^T(\mathbf{b}^2\mathbf{c}) \overset{\text{L4}}{=} \mathbf{b}^T\mathbf{A}\mathbf{b}^2 \overset{\text{L6}}{=} 0_{R4}$	
5	$\mathbf{b}^T\mathbf{A}^2\mathbf{b}^2 \overset{\text{L4}}{=} -\mathbf{b}^T((\mathbf{A}\mathbf{b}^2)\mathbf{c}) \overset{\text{L8}}{=} \mathbf{b}^T((\mathbf{A}\mathbf{c})\mathbf{b}^2) \overset{\text{L6}}{=} 0_{R5}$	$8\mathbf{b}^T((\mathbf{A}\mathbf{c})\mathbf{b}^2) - 4\mathbf{b}^T(\mathbf{b}^2\mathbf{c}^2)$ $\overset{\text{L6}}{=} \mathbf{b}^T\mathbf{b}^4 \overset{\text{L6}}{=} 0_{R5}$
	$\mathbf{b}^T\mathbf{A}(\mathbf{b}^2\mathbf{c}) \overset{\text{L4}}{=} -\mathbf{b}^T(\mathbf{b}^2\mathbf{c}^2) \overset{\text{L6}}{=} 0_{R5}$	
6	$2\mathbf{b}^T\mathbf{A}^3\mathbf{b}^2 \overset{\text{L4}}{=} -2(\mathbf{b}\mathbf{c})^T\mathbf{A}^2\mathbf{b}^2$ $\overset{\text{L8}}{=} 2\mathbf{b}^T((\mathbf{A}\mathbf{b}^2)\mathbf{A}\mathbf{c}) \overset{\text{L8}}{=} -2\mathbf{b}^T((\mathbf{A}^2\mathbf{c})\mathbf{b}^2)$ $\overset{\text{L6}}{=} -\mathbf{b}^T((\mathbf{A}\mathbf{c}^2)\mathbf{b}^2) \overset{\text{L8}}{=} \mathbf{b}^T((\mathbf{A}\mathbf{b}^2)\mathbf{c}^2)$ $\overset{\text{L4}}{=} -\mathbf{b}^T\mathbf{A}(\mathbf{c}\mathbf{A}\mathbf{b}^2) \overset{\text{L6}}{=} 0_{R6}$	$\mathbf{b}^T((\mathbf{A}\mathbf{b}^2)\mathbf{b}^2) \overset{\text{L9}}{=} 0_{N3}$ $3(\mathbf{b}\mathbf{c})^T\mathbf{A}^2\mathbf{b}^2 + 3(\mathbf{b}\mathbf{c})^T\mathbf{A}(\mathbf{b}^2\mathbf{c})$ $\overset{\text{N3}}{=} -\mathbf{b}^T(\mathbf{b}^2\mathbf{c}^3) \overset{\text{L4}}{=} \mathbf{b}^T\mathbf{A}(\mathbf{b}^2\mathbf{c}^2) \overset{\text{L6}}{=} 0_{R6}$
	$-\mathbf{b}^T\mathbf{A}(\mathbf{b}^2\mathbf{A}\mathbf{c}) \overset{\text{L4}}{=} (\mathbf{b}\mathbf{c})^T((\mathbf{A}\mathbf{c})\mathbf{b}^2)$ $\overset{\text{L8}}{=} -(\mathbf{b}\mathbf{c})^T\mathbf{A}(\mathbf{b}^2\mathbf{c}) \overset{\text{L4}}{=} \mathbf{b}^T\mathbf{A}^2(\mathbf{b}^2\mathbf{c}) \overset{\text{L6}}{=} 0_{R6}$	$12(\mathbf{b}\mathbf{c})^T\mathbf{A}^2\mathbf{b}^2 + 4(\mathbf{b}\mathbf{c})^T\mathbf{A}(\mathbf{b}^2\mathbf{c})$ $\overset{\text{N4}}{=} \mathbf{b}^T(\mathbf{b}^4\mathbf{c}) \overset{\text{L4}}{=} -\mathbf{b}^T\mathbf{A}\mathbf{b}^4 \overset{\text{L6}}{=} 0_{R6}$

Table 5
Number of RK and Null conditions up to the sixth-order accuracy.

Approximation order, p	2	3	4	5	6
Number of RK conditions	2	2	4	9	20
Rank (# of dependent) of RK conditions	2	2	2 (2)	4, 3 (2)	4, 5 (11)
Number of Null conditions	-	1	2	6	16
Rank (# of dependent) of Null conditions	-	1	2	3, 2 (1)	7, 4 (5)

Table 5 lists the total number of RK and Null conditions. In the table, the term rank refers to the number of conditions grouped together by equivalence. For example, the entry “4, 3 for 5th-order RK” indicates two independent groups of equivalent RK conditions, one with four conditions and another with three. Values in parentheses denote the number of linearly dependent conditions. The sum of all rank entries matches the total number of conditions shown in the row above.

We observe that the group numbers of independent RK conditions are 1, 1, 1, 2, 2 (or 1, 2, 3, 5, 7 in total) for $p = 2, 3, 4, 5, 6$, respectively. These values exactly match the number of groups of independent Null conditions with the second-order consistency condition. This correspondence suggests a fundamental structural relationship between the RK and Null conditions. Specifically, all Null conditions vanish when the corresponding RK conditions are satisfied, and conversely, the RK conditions hold when the corresponding Null constraints with the consistency condition are met. Further insight into the derivation and reduction of such

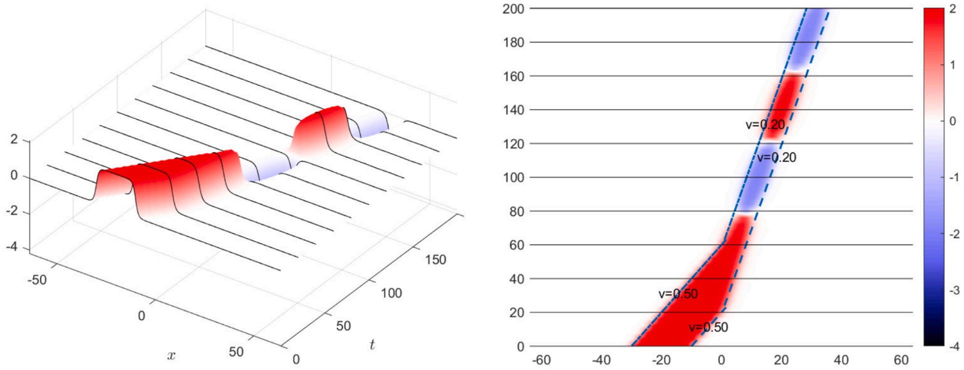

Fig. 13. Evolution of the solution in heterogeneous medium with $\alpha = 0.3$.

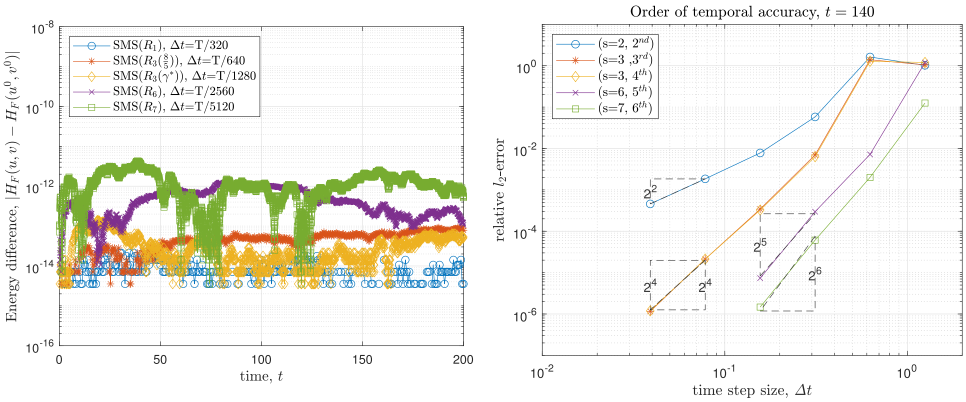

Fig. 14. Computational errors of $\mathcal{H}_F(u,v)$ and convergence of the solution at $t = 140$.

order conditions can be gained through the tree-theoretic framework described in [29]. In particular, the results in [31] closely correspond to the concept of Null conditions introduced here.

5.2. Numerical convergence order and energy conservation

We revisit the sine-Gordon equation tested in Section 4.2,

$$u_{tt} - \nabla \cdot (\mathbf{M}(\mathbf{x})\nabla u) + \sin u = 0,$$

on $\Omega = [-64, 64]$ with the periodic boundary conditions. An interesting case arises when we choose $\mathbf{M}(\mathbf{x}) = 1 + \frac{\alpha}{1+e^{-5\mathbf{x}}}$ with $\alpha = 0.3$. The leftmost plot in Fig. 13 shows the evolution of the solution where the wave fronts reach the interface around $t = 25, 65$ and that their speed decreases to about 0.2 after passing the interface. We observe that the wave fronts alternate in their forward and backward movement. While a detailed study of nonlinear wave interaction lies beyond the scope of this paper, our numerical methods are well-suited for such investigations, especially when high accuracy is required for long time evolution.

To evaluate the performance of the fifth- and sixth-order schemes discussed in the previous subsection, we employ specific coefficient vectors: $R_6 \approx$ [0.254002, 0.680054, 1.009647, 0.284283, -0.729926, -0.998060] and $R_7 \approx$ [0.392257, 0.117787, -0.588840, 0.657593, -0.588840, 0.117787, 0.392257]. These vectors were originally introduced in [32] and further elaborated in Section V.3 of [29]. It is worth noting that, although these RK tables are derived to preserve symplecticity, they do not necessarily retain symplecticity in our context. We adapt them here for constructing energy-preserving methods since they also satisfy the order conditions listed in the previous subsection for our target accuracy.

We observe interesting wavefronts interactions after $t \approx 80$ in Fig. 13 where the solution are computed by the sixth-order method with $N_x = 768$, $\Delta t = 0.1$ up to $t \leq 200$. The leftmost plot in Fig. 14 demonstrates that the methods are energy conserving and the rightmost plot presents the corresponding convergence results at $t = 140$. Since achieving an error close to single precision is sufficient for most practical purposes, we report only the time step sizes required to reach that threshold. The results clearly demonstrate a consistent decrease in error with time step refinement, and notably, the observed third-order convergence exceeds the expected rate.

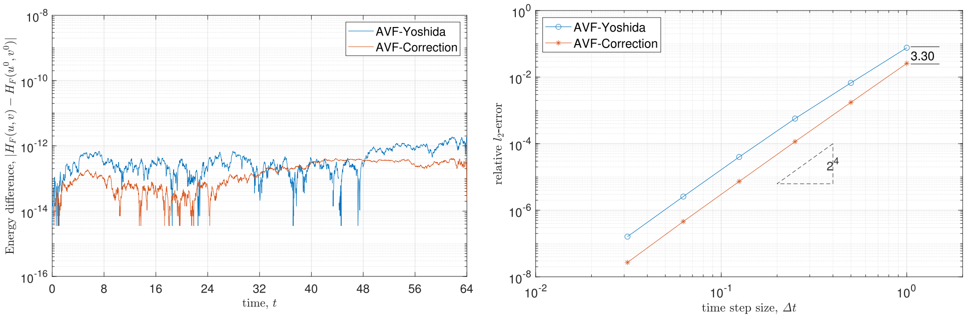

Fig. 15. Computational errors of $H(u^n, v^n)$, $0 \le t \le 64$ and convergence test at $t = 48$ for the fourth-order Yoshida and correction methods.

6. Comparison with higher-order averaged vector field methods

The one-stage, second-order SSD method can be interpreted within the framework of discrete gradient (DG) methods. Consequently, higher-order SSD schemes are naturally related to composition-based DG approaches. Another closely related class of numerical methods is the averaged vector field (AVF) method [20], which can be extended to higher-order accuracy in two different ways. The first approach applies a composition strategy [36] to the second-order AVF method; we refer to this as the AVF-Yoshida method. The second approach, introduced in [20], employs B-series corrections to the second-order AVF method; we refer to this as the AVF-correction method. In this section, we clarify the relationship between our SSD($R_3(\gamma^*)$) method and fourth-order AVF methods through direct numerical experiments with the same spatial discretization on the sine-Gordon equation, one of the few problems for which the implementation details of higher-order AVF methods are well established.

Denoting $z = (u, v) \in V^2$, we can reformulate the sine-Gordon equation as $z_t = f(z)$, where $f(z) = (v, \Delta u - \sin u)$. The second-order AVF method applied to the sine-Gordon equation takes the form

$$z^{n+1} = z^n + \Delta t F(z^n, z^{n+1}), \tag{29}$$

where

$$F(z^n, z^{n+1}) = \int_0^1 f((1-\xi)z^n + \xi z^{n+1})\, d\xi = \left(\frac{v^n + v^{n+1}}{2}, \Delta \frac{u^n + u^{n+1}}{2} + \frac{\Phi(u^{n+1}) - \Phi(u^n)}{u^{n+1} - u^n} \right),$$

which coincides with the second-order SSD method (25). For higher-order extensions, the AVF-Yoshida method is constructed by applying the composition strategy [36] to the second-order AVF method (29) with the coefficients $w_1 = w_3 = \frac{1}{2-2^{1/3}}$ and $w_2 = -\frac{2^{1/3}}{2-2^{1/3}}$. Starting with $z_0 = z^n$, we evaluate z_i for $i = 1, 2, 3$ by solving

$$z_i = z_{i-1} + w_i \Delta t F(z_{i-1}, z_i), \tag{30}$$

and set $z^{n+1} = z_3$. Note that these coefficients coincide with $R_3(\gamma^*)$ used in the fourth-order SSD method, and consequently the AVF-Yoshida method is equivalent to the fourth-order SSD method for the sine-Gordon equation. The AVF method can be also generalized using B-series corrections as proposed in [20]. The fourth-order AVF-Correction method is given by

$$z^{n+1} = z^n + F(z^n, z^{n+1}) - \frac{1}{12}\Delta t^2 \left[J(\hat{z}^n)\right]^2 F(z^n, z^{n+1}),$$

where J is a Jacobian of f and $\hat{z}^n = (z^n + z^{n+1})/2$. Here, for simplicity, we denote $(F_1, F_2) = F(z^n, z^{n+1})$ and write

$$\left[J(\hat{z}^n)\right]^2 F(z^n, z^{n+1}) = \left(\Delta F_1 - F_1 \cos(\hat{u}^n),\ \Delta F_2 - F_2 \cos(\hat{u}^n)\right).$$

The leftmost plot in Fig. 15 shows the discrete energy difference $H(u^n, v^n)$ for the AVF-Yoshida and AVF-Correction methods with $\Delta t = T_f/2^{11}$, applied to the sine-Gordon equation up to T_f=64 using the same initial state and parameters as in Fig. 1. The result demonstrates that both methods preserve the discrete energy up to machine precision. The rightmost plot presents the convergence result at t=48, confirming that both methods achieve fourth-order temporal accuracy with $\Delta t = T_f/2^{11}, \cdots, T_f/2^6$. The AVF-Correction method yields a smaller error than the AVF-Yoshida method. At the largest time step, the ratio of the two errors is approximately 3.30, a point that will be revisited in the discussion of Fig. 16.

The leftmost plot in Fig. 16 shows the Fourier spectral differences of the AVF-Yoshida and the AVF-Correction solution at $t = 48$ compared to the SSD($R_3(\gamma^*)$) with the largest time step $\Delta t = T_f/2^6$. The difference (Yos - SSD) between the Yoshida and SSD methods remains below prescribed single precision tolerance for the implicit non-linear Eq. (30), confirming that they are numerically equivalent. In contrast, the difference between the correction and SSD methods (Cor - SSD) is significantly larger than (Yos - SSD), indicating these methods produce distinct solutions. The rightmost plot shows the difference between each method and the reference solution (Yos - Ref and Cor - Ref). The ratio of the peak values of Cor - Ref and SSD - Ref is approximately 3.30, which is consistent with the error ratio observed in Fig. 15. This agreement further confirms that the two fourth-order methods produce distinct solutions even for the sine-Gordon equation.

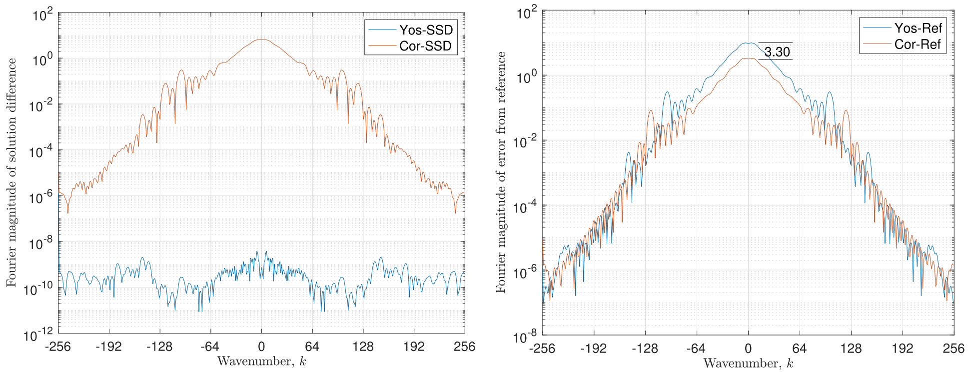

Fig. 16. Fourier spectral difference of Yoshida and correction solution compared to the SSD($R_3(\gamma^*)$) (leftmost plot) and the reference solution (rightmost plot).

7. Conclusions

We have introduced a high-order time-discretization method that conserves energy for nonlinear wave equations. Beginning with the energy-conserving secant-difference method, we developed the successive secant-difference (SSD) method to achieve higher temporal accuracy. The nonlinearity of the equations prevents the secant-difference method from being simplified into an additive scheme, and the successive strategy does not produce a Runge–Kutta method. However, by employing Taylor's theorem at the midpoints, our approach achieves high-order accuracy. This method shares similarities with the order conditions of Runge–Kutta methods, albeit with additional null conditions. Numerical examples, including the sine-Gordon and Boussinesq equations, demonstrate the accuracy and stability of the proposed methods. We established the full set of order conditions up to sixth order, identified their linear dependencies to determine the minimal independent constraints, and demonstrated that the resulting SSD-based methods achieve high-order accuracy while preserving energy effectively over long simulation times. We have also compared the proposed SSD methods with higher-order averaged vector field methods for the sine-Gordon equation, showing that the composition-based extension is numerically equivalent to the fourth-order SSD($R_3(\gamma^*)$) method, whereas the B-series correction approach in [20] leads to a distinct scheme with different solution structures.

CRediT authorship contribution statement

Jaemin Shin: Writing – original draft, Formal analysis, Conceptualization; **June-Yub Lee:** Writing – review & editing, Visualization, Formal analysis.

Acknowledgment

We thank the reviewer for the careful review of the manuscript and for the constructive comments that improved this article. In particular, the suggestion on the final comparative experiments was valuable in guiding our future research. This research was supported by the Basic Science Research Program (RS-2023-00214185, RS-2023-00273584) through the National Research Foundation of Korea (NRF), funded by the Ministry of Science and ICT (MSIT).

References

[1] R.J. LeVeque, Finite volume methods for hyperbolic problems, 31, Cambridge university press, 2002.
[2] S. Alinhac, Hyperbolic partial differential equations, Springer Science & Business Media, 2009.
[3] A. Meister, J. Struckmeier, Hyperbolic partial differential equations: theory, numerics and applications, Springer Science & Business Media, 2012.
[4] J. Chabassier, S. Imperiale, Introduction and study of fourth order theta schemes for linear wave equations, J. Comput. Appl. Math. 245 (2013) 194–212.
[5] J. Diaz, M.J. Grote, Energy conserving explicit local time stepping for second-order wave equations, SIAM J. Sci. Comput. 31 (3) (2009) 1985–2014.
[6] P. Joly, J. Rodríguez, Optimized higher order time discretization of second order hyperbolic problems: construction and numerical study, J. Comput. Appl. Math. 234 (6) (2010) 1953–1961.
[7] S. Britt, E. Turkel, S. Tsynkov, A high order compact time/space finite difference scheme for the wave equation with variable speed of sound, J. Sci. Comput. 76 (2) (2018) 777–811.
[8] E. Burman, O. Duran, A. Ern, Hybrid high-order methods for the acoustic wave equation in the time domain, Commun. Appl. Math. Comput. 4 (2) (2022) 597–633.
[9] T.J. Bridges, S. Reich, Numerical methods for hamiltonian PDEs, J. Phys. A: Math. Gen. 39 (19) (2006) 5287.
[10] L. Brugnano, G. Frasca-Caccia, F. Iavernaro, Energy conservation issues in the numerical solution of the semilinear wave equation, Appl. Math. Comput. 270 (2015) 842–870.
[11] M.A. Sánchez, C. Ciuca, N.C. Nguyen, J. Peraire, B. Cockburn, Symplectic hamiltonian HDG methods for wave propagation phenomena, J. Comput. Phys. 350 (2017) 951–973.
[12] L. Brugnano, F. Iavernaro, D. Trigiante, Hamiltonian boundary value methods (energy preserving discrete line integral methods), J. Numer. Anal. Ind. Appl. Math. 5 (1–2) (2010) 17–37.
[13] L. Brugnano, F. Iavernaro, Line integral methods for conservative problems, 13, CRC Press, 2016.

[14] L. Brugnano, F. Iavernaro, J.I. Montijano, L. Rández, Spectrally accurate space-time solution of hamiltonian PDEs, Numer. Algorithms 81 (4) (2019) 1183–1202.
[15] D. Li, W. Sun, Linearly implicit and high-order energy-conserving schemes for nonlinear wave equations, J. Sci. Comput. 83 (2020) 1–17.
[16] C. Jiang, Y. Wang, Y. Gong, Explicit high-order energy-preserving methods for general hamiltonian partial differential equations, J. Comput. Appl. Math. 388 (2021) 113298.
[17] G. Frasca-Caccia, P.E. Hydon, A new technique for preserving conservation laws, Found. Comput. Math. 22 (2) (2022) 477–506.
[18] G. Frasca-Caccia, Finite difference schemes with non polynomial local conservation laws, J. Comput. Appl. Math. 458 (2025) 116330.
[19] E. Celledoni, V. Grimm, R.I. McLachlan, D.I. McLaren, D. O'Neale, B. Owren, G. Quispel, Preserving energy resp. dissipation in numerical PDEs using the "average vector field" method, J. Comput. Phys. 231 (20) (2012) 6770–6789.
[20] G. Quispel, D.I. McLaren, A new class of energy-preserving numerical integration methods, J. Phys. a: Math. Theor. 41 (4) (2008) 045206.
[21] H. Li, Y. Wang, M. Qin, A sixth order averaged vector field method, J. Comput. Math. (2016) 479–498.
[22] J. Bai, H. Ullah, C. Li, Energy-preserving methods for non-smooth nonlinear schrödinger equations, Appl. Numer. Math. 185 (2023) 188–202.
[23] R.I. McLachlan, G. Quispel, Discrete gradient methods have an energy conservation law, Discrete Contin. Dyn. Syst. 34 (3) (2013) 1099–1104.
[24] D. Furihata, Finite-difference schemes for nonlinear wave equation that inherit energy conservation property, J. Comput. Appl. Math. 134 (1–2) (2001) 37–57.
[25] T. Matsuo, D. Furihata, Dissipative or conservative finite-difference schemes for complex-valued nonlinear partial differential equations, J. Comput. Phys. 171 (2) (2001) 425–447.
[26] J. Shin, J.-Y. Lee, Energy conserving successive multi-stage method for the linear wave equation, J. Comput. Phys. 458 (2022) 111098.
[27] J. Shin, J.-Y. Lee, Energy-conserving successive multi-stage method for the linear wave equation with forcing terms, J. Comput. Phys. (2023) 112255.
[28] Q. Meng-Zhao, Z. Wen-Jie, Construction of higher order symplectic schemes by composition, Computing 47 (3–4) (1991) 309–321.
[29] E. Hairer, C. Lubich, G. Wanner, Geometric numerical integration, 2nd ed., Springer-Verlag, Berlin, Heidelberg, Berlin, Heidelberg, 2006.
[30] D. Furihata, T. Matsuo, Discrete variational derivative method: a structure-preserving numerical method for partial differential equations, CRC Press, 2010.
[31] A. Murua, J.M. Sanz-Serna, Order conditions for numerical integrators obtained by composing simpler integrators, Philos. Trans. R. Soc. Lond. Ser. A: Math., Phys. Eng. Sci. 357 (1754) (1999) 1079–1100.
[32] Z. Kalogiratou, T. Monovasilis, Diagonally implicit symplectic runge–Kutta methods with special properties, Appl. Math. Inf. Sci. 9 (1L) (2015) 11–17.
[33] E. Hairer, G. Wanner, Solving ordinary differential equations. I, 8, Springer-Verlag, Berlin, 1993.
[34] E. Hairer, G. Wanner, Solving ordinary differential equations. II, 14, Springer-Verlag, Berlin, 1996.
[35] H.N. Hassan, Numerical solution of a boussinesq type equation using fourier spectral methods, Z. f"ur Naturforsch. a 65 (4) (2010) 305–314.
[36] H. Yoshida, Construction of higher order symplectic integrators, Phys. Lett. A 150 (5–7) (1990) 262–268.