Journal of Computational Physics 401 (2020) 109002

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Energy conserving local discontinuous Galerkin methods for the improved Boussinesq equation

Xiaole Li [a,b,1,4], Weizhou Sun [b,4], Yulong Xing [b,2], Ching-Shan Chou [b,3]

[a] College of Automation, Harbin Engineering University, Harbin 150001, China
[b] Department of Mathematics, The Ohio State University, Columbus, OH 43210, United States of America

ARTICLE INFO

Article history:
Received 4 October 2018
Received in revised form 12 August 2019
Accepted 30 September 2019
Available online 3 October 2019

Keywords:
Improved Boussinesq equation
Local discontinuous Galerkin methods
Energy conserving methods
Error estimate
Solitary waves

ABSTRACT

The Boussinesq-type equations describe the propagation of weakly non-linear long waves in shallow waters and are widely applied to model water waves in shallow seas and harbors. In this paper, we propose a high-order local discontinuous Galerkin method to solve the improved Boussinesq equation, coupled with both explicit leap-frog and implicit midpoint energy-conserving time discretization. The proposed full-discrete method can be shown to conserve the discrete versions of both mass and energy of the continuous solution. The error estimate with optimal order of convergence is provided for the semi-discrete method. Our numerical experiments confirm optimal rates of convergence as well as the mass and energy conserving property, and show that the errors of the numerical solutions do not grow significantly in time due to the energy conserving property. A series of numerical experiments are provided to show that the proposed method has the capability to simulate the interaction between two solitary waves, single wave break-up and blow-up behavior well.

1. Introduction

Minimizing the wave disturbance is essential to the design of harbors. In coastal engineering, conducting experiments in water tanks and numerically analyzing appropriate mathematical models are the two most effective methods. The advantage of mathematical models is that various physical conditions can be simulated with ease while the assumptions of the models can be verified. The Boussinesq-type equation, introduced in 1872 by Joseph Boussinesq [5], describes the propagation of weakly non-linear fairly long waves in shallow waters and since then have found wide applications in modeling water waves in shallow seas and harbors. The original equation proposed by Boussinesq is:

$$u_{tt} = u_{xx} + u_{xxxx} + (u^2)_{xx}. \tag{1.1}$$

It is known as the "bad" Boussinesq equation since the solution exhibits unrealistic instability for short wavelengths [3]. The presence of exponentially growing Fourier components renders linear instability [28]. When the coefficient of the fourth

E-mail addresses: leeggwp@163.com (X. Li), sun.435@osu.edu (W. Sun), xing.205@osu.edu (Y. Xing), chou@math.ohio-state.edu (C.-S. Chou).

[1] XL is supported by the College of Automation, Harbin Engineering University, China.
[2] YX is partially supported by the NSF grant DMS-1753581 and ONR grant N00014-16-1-2714.
[3] CSC is supported by NSF grant DMS-1253481 and DMS-1813071.
[4] Both authors contributed equally to this paper.

https://doi.org/10.1016/j.jcp.2019.109002
0021-9991/

order derivative term is changed to -1, the equation is called the "good" Boussinesq equation as it is linearly stable. Manoranjan et al. [28] showed the existence of soliton solutions for the "good" Boussinesq equation.

In this paper, we focus on the study of the so-called improved Boussinesq equation proposed by Bogolyubsky [2]:

$$u_{tt} = u_{xx} + u_{xxtt} + (u^2)_{xx}, \tag{1.2}$$

which has some proven desired properties, such as mass conserving and energy conserving. No instabilities have been observed by researchers for the improved Boussinesq equation. Iskandar et al. [20] first studied the improved Boussinesq equation numerically. They proposed a three-level iterative scheme using finite difference approximations. In [18], El-Zoheiry designed a three-level iterative scheme based on the compact implicit methods for the improved Boussinesq equation. Lin et al. [27] developed a B-spline finite element method for the improved Boussinesq equation. In particular, by partitioning the space into a set of elements and expressing the solution in terms of linear B-spline basis functions, they obtained a system for which many standard numerical integration algorithms are applicable. Their method has the merit that it is simple to implement and the nonlinearity can be taken care of easily. All the above methods, however, did not address the conservation properties that the theoretical analysis for improved Boussinesq equation has shown. Wang et al. [32] developed an energy-conserving finite volume element method for the improved Boussinesq equation. They constructed the second order accurate scheme using discrete variational derivative method. More recently, a class of high-order energy-conserving schemes based on Fourier pseudospectral methods and Hamiltonian boundary value methods, is proposed for the improved Boussinesq equation in [36].

Discontinuous Galerkin (DG) methods will be studied in this paper to solve the improved Boussinesq equation. DG methods were introduced in 1973 by Reed and Hill [29] to solve steady state linear hyperbolic equations. A major development of DG methods was carried out by Cockburn and Shu in a series of papers [13–16]. DG methods have gained extensive attentions and their applications include wave propagation problems, optimal control, compressible flows, incompressible flows, semiconductor device simulation, Hamilton-Jacobi equations, Maxwell and Magnetohydrodynamics equations, elasticity problem, KdV and other nonlinear dispersive equations, among many others. DG methods can be easily constructed to be of high order accuracy, and can be used on arbitrary triangulation, thus very suitable for $h-p$ refinement. Their local nature enables parallel computing and therefore enhances the efficiency.

DG methods have been applied to solve partial differential equations (PDEs) with high order spatial derivatives. Such DG methods include the interior penalty DG methods [1], local discontinuous Galerkin (LDG) methods [17], ultra-weak DG methods [9], hybridizable DG methods [12], among many others. One successful group of such methods is the LDG methods, proposed by Cockburn and Shu. The basic idea is to rewrite the original equation into a first order system and discretize it in space with DG methods. They showed that the stability of the proposed methods can be achieved with a careful choice of numerical fluxes. The LDG methods have been applied to an extensive list of PDEs with high order spatial derivatives and we refer any interested reader to the review paper [35]. Recently, LDG method has been used to simulate problems that require structure preserving property, such as mass, energy or Hamiltonian conservation. Energy conserving LDG methods have been designed for the generalized KdV equation [4,21], the second order wave equation [33,11], Camassa-Holm equation [26], the nonlinear Schrödinger equation [19,25], and the two-way wave equation [10], for which the schemes were shown to preserve energy exactly in the discrete level, leading to small phase and shape errors in long time simulations.

In this work, we develop novel structure preserving LDG methods to solve the improved Boussinesq equation, which can conserve both the mass and energy of the model in the discrete level. The equation is first decoupled into several first order differential equations, and the LDG methods are designed based on the first order system. With the choice of alternating numerical fluxes, we can show that the proposed methods conserve both mass and energy exactly. We have also provided a family of energy conserving numerical fluxes and energy dissipative numerical fluxes. Optimal error estimate is derived for the semi-discrete LDG methods. For the temporal discretization, both explicit leap-frog and implicit midpoint time discretization, with special treatment of the nonlinear term, are proposed to maintain the mass and energy conserving property of the full discrete schemes. Our numerical experiments demonstrate optimal rates of convergence as well as the mass and energy conserving property, and show that the errors of the numerical solutions do not grow significantly in time due to the energy conserving property. A series of numerical experiments are provided to show that the proposed method has the capability to simulate the interaction between two solitary waves, single wave break-up and blow-up behavior well.

The organization of the paper is as follows. In Section 2, we introduce the LDG numerical method and prove the mass and energy preserving property for the semi-discrete case. In Section 3, the error estimate of the proposed methods is presented. Two different energy-conserving temporal discretizations are presented in Section 4. In Section 5, we perform numerical tests to show the order of accuracy, the mass and energy conservation and long time simulation of our scheme, and also analyze its performance in various solitary wave examples. Conclusion remarks are presented in Section 6.

2. Local discontinuous Galerkin discretization

2.1. Model problem

We consider the improved Boussinesq equation (1.2) on the domain $I = [x_l, x_r]$ in this paper. Since we are interested in the energy conservation property of this model, it is more convenient to convert the equation into

$$\begin{cases} u_t = v_x, \\ v_t - v_{xxt} = (u + u^2)_x, \end{cases} \tag{2.1}$$

by introducing the variable v. This also appears in [32,36] to study the conservation property of this model. The system (2.1) has the following conserved mass and energy defined by

$$M = \int_I u\, dx, \qquad E = \int_I \left(\frac{v^2}{2} + \frac{(u_t)^2}{2} + \frac{u^2}{2} + \frac{u^3}{3} \right) dx. \tag{2.2}$$

The initial conditions are given by

$$u(x,0) = u_0(x), \quad u_t(x,0) = g(x), \quad \text{for all } x \in [x_l, x_r], \tag{2.3}$$

with periodic boundary conditions

$$\frac{\partial^i}{\partial^i x} u(x_l, t) = \frac{\partial^i}{\partial^i x} u(x_r, t), \quad \text{for all } t \in (0, \infty),\ i = 0, 1, 2. \tag{2.4}$$

In addition, we also require

$$\int_I g(x)\, dx = 0, \tag{2.5}$$

and this initial condition, along with Eq. (1.2), implies mass conservation $\int_I u\ dx = constant$ for $t \geq 0$.

2.2. Notations

We partition $I = [x_l, x_r]$ into J subintervals: $x_l = x_{\frac{1}{2}} < x_{\frac{3}{2}} < \ldots < x_{J+\frac{1}{2}} = x_r$. For each interval $I_i = (x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}})$, $1 \leq i \leq J$, we define the midpoint $x_i = (x_{i-\frac{1}{2}} + x_{i+\frac{1}{2}})/2$ and $h_i = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$ with $h = \max_i h_i$. The piecewise polynomial space is defined as $V^k = \{v : v|_{I_i} \in P^k(I_i), i = 1, \cdots, J\}$, where $P^k(I_i)$ denotes the space of polynomials of degree up to k on I_i. For any function $u_h \in V^k$, $u_h^+(x_{i+\frac{1}{2}})$ and $u_h^-(x_{i+\frac{1}{2}})$ denote the limit values of u_h at $x_{i+\frac{1}{2}}$ from the right interval I_{i+1} and the left interval I_i, respectively. We use the usual notations $[u_h]_{i+\frac{1}{2}} = u_h^+(x_{i+\frac{1}{2}}) - u_h^-(x_{i+\frac{1}{2}})$ and $\{u_h\}_{i+\frac{1}{2}} = \frac{1}{2}(u_h^+(x_{i+\frac{1}{2}}) + u_h^-(x_{i+\frac{1}{2}}))$ to represent the jump and the average of the function u_h at the cell interfaces, respectively. For shorthand notation, the inner product is denoted by $(w, v)_{I_i} = \int_{I_i} wv dx$ for the scalar variables w, v. The L^2 norm of v over the element I_i is denoted by $\|v\|_{I_i} = \sqrt{(v, v)_{I_i}}$. The inner product over the domain I is denoted by $(w, v) = (w, v)_I$, the L^2 norm $\|v\| = \sqrt{(v, v)}$ and the L^∞ norm $\|v\|_\infty = \max_{x \in I} |v(x)|$.

Three types of projection operators considered in the paper are defined as follows. We use P to denote the standard L^2 projection of a function ω into space V^k satisfying

$$(P\omega, \phi)_{I_i} = (\omega, \phi)_{I_i}, \quad \forall \phi \in P^k(I_i).$$

We use P^- to denote the projection of ω into space V^k with

$$\begin{aligned} &(P^-\omega, \phi)_{I_i} = (\omega, \phi)_{I_i}, && \forall \phi \in P^{k-1}(I_i), \\ &(P^-\omega)^- = \omega^-, && \text{at } x_{i+\frac{1}{2}}. \end{aligned}$$

Similarly, the projection P^+ of ω is defined as:

$$\begin{aligned} &(P^+\omega, \phi)_{I_i} = (\omega, \phi)_{I_i}, && \forall \phi \in P^{k-1}(I_i), \\ &(P^+\omega)^+ = \omega^+, && \text{at } x_{i-\frac{1}{2}}. \end{aligned}$$

For these projections, we have the following approximation property [8]:

$$\left\| P^* f - f \right\|^2 + h \sum_i \left((P^* f - f)^{\pm}_{i+\frac{1}{2}} \right)^2 \leq C h^{2k+2}, \tag{2.6}$$

where $P^* = P$ or $P^\pm$, and the constant C depends on f but is independent of h.

2.3. LDG method for the improved Boussinesq equation

The LDG method will be presented in this section. The main idea of LDG methods is to suitably rewrite a higher order PDE into a first order system, and then apply the DG method to the system. By introducing auxiliary variables w, r, s, the model (2.1) becomes a first order system taking the form of

$$\begin{cases} u_t = w, \\ w = v_x, \\ r = v - w_x, \\ s = u + u^2, \\ r_t = s_x. \end{cases} \tag{2.7}$$

The LDG scheme for (2.7) can be formulated as follows: we look for u_h, v_h, w_h, s_h, $r_h \in V^k$, such that

$$\int_{I_i} (u_h)_t \phi \, dx = \int_{I_i} w_h \phi \, dx, \tag{2.8}$$

$$\int_{I_i} w_h \psi \, dx = \widehat{v_h}\psi^-|_{x_{i+\frac{1}{2}}} - \widehat{v_h}\psi^+|_{x_{i-\frac{1}{2}}} - \int_{I_i} v_h \psi_x \, dx, \tag{2.9}$$

$$\int_{I_i} r_h \varphi \, dx = \int_{I_i} v_h \varphi \, dx - \left(\widehat{w_h}\varphi^-|_{x_{i+\frac{1}{2}}} - \widehat{w_h}\varphi^+|_{x_{i-\frac{1}{2}}} - \int_{I_i} w_h \varphi_x \, dx \right), \tag{2.10}$$

$$\int_{I_i} s_h \xi \, dx = \int_{I_i} (u_h + u_h^2) \xi \, dx, \tag{2.11}$$

$$\int_{I_i} (r_h)_t \zeta \, dx = \widehat{s_h}\zeta^-|_{x_{i+\frac{1}{2}}} - \widehat{s_h}\zeta^+|_{x_{i-\frac{1}{2}}} - \int_{I_i} s_h \zeta_x \, dx, \tag{2.12}$$

hold for all test functions ϕ, ψ, φ, ξ, $\zeta \in V^k$.

The terms $\widehat{v_h}$, $\widehat{w_h}$, $\widehat{s_h}$ are numerical fluxes resulting from integration by parts. As shown later in the proof of energy conservation, as long as we choose alternating fluxes for the pair $\{\widehat{v_h}, \widehat{w_h}\}$, and the pair $\{\widehat{v_h}, \widehat{s_h}\}$, the scheme will have the desired mass and energy conserving properties. For example we can choose:

$$\widehat{v_h} = v_h^-, \quad \widehat{w_h} = w_h^+, \quad \widehat{s_h} = s_h^+, \tag{2.13}$$

or

$$\widehat{v_h} = v_h^+, \quad \widehat{w_h} = w_h^-, \quad \widehat{s_h} = s_h^-. \tag{2.14}$$

In the remainder of the paper we use the flux (2.13). All the results presented in this paper can be proved for both fluxes (2.13) and (2.14) with similar proofs.

Remark 2.1. One can also define a family of numerical fluxes as:

$$\widehat{v_h} = \{v_h\} + \alpha[v_h], \quad \widehat{w_h} = \{w_h\} - \alpha[w_h], \quad \widehat{s_h} = \{s_h\} - \alpha[s_h], \quad \alpha \in \left[-\frac{1}{2}, \frac{1}{2}\right], \tag{2.15}$$

which are the generalization of the alternating fluxes (2.13) (when $\alpha = -\frac{1}{2}$) and (2.14) (when $\alpha = \frac{1}{2}$). This family of numerical fluxes can also be shown to produce energy conserving LDG methods.

If one is interested in an energy dissipative method, it can be obtained with the following choice of numerical fluxes:

$$\widehat{v_h} = \{v_h\} + \alpha[v_h], \quad \widehat{w_h} = \{w_h\} - \alpha[w_h], \quad \widehat{s_h} = \{s_h\} - \alpha[s_h] + \beta[v_h], \quad \alpha \in \left[-\frac{1}{2}, \frac{1}{2}\right], \beta > 0.$$

Remark 2.2. In the conventional LDG method, there is no need to introduce the extra auxiliary variable s_h in Eq. (2.11), which could be absorbed in Eq. (2.12). It was introduced here for the purpose of energy conservation.

2.4. Conservation of mass and energy

As is well known, the important physical quantities, the mass and the energy (2.2), are conserved in Eq. (1.2). Here we show that our numerical solution also preserves these quantities at the discrete level.

Proposition 2.1. *Let u_h and v_h be the solutions of the LDG scheme (2.8) – (2.12) with flux (2.13). Then the mass*

$$M_h(t) = \int_I u_h \, dx \tag{2.16}$$

is invariant in time t if periodic boundary condition is imposed for v_h.

Proof. To begin with, we show that the periodic boundary condition (2.4) and the requirement on the initial condition (2.5) leads to $v(x_l, t) = v(x_r, t)$. Recall that by the definition of v in (2.7), (1.2) can be written into:

$$u_t = v_x, \tag{2.17}$$

$$v_t = u_x + u_{xtt} + (u^2)_x. \tag{2.18}$$

Integrating (2.18) from 0 to t, we have

$$v(x,t) = v(x,0) + \int_0^t u_x + u_{xtt} + (u^2)_x \, dt.$$

By the periodic boundary conditions (2.4),

$$\int_0^t u_x(x_l,t) + u_{xtt}(x_l,t) + (u^2)_x(x_l,t)\, dt = \int_0^t u_x(x_r,t) + u_{xtt}(x_r,t) + (u^2)_x(x_r,t)\, dt,$$

and from (2.5) we have

$$v(x_r,0) = v(x_l,0) + \int_{x_l}^{x_r} v_x dx = v(x_l,0) + \int_{x_l}^{x_r} g(x) dx = v(x_l,0).$$

Therefore,

$$v(x_l,t) = v(x_r,t).$$

This implies that in the numerical scheme, we also need to impose periodic boundary conditions for v_h.

Setting $\phi = \psi = 1$ in Eqs. (2.8)-(2.9), we obtain:

$$\frac{d}{dt} M_h(t) = \frac{d}{dt} \int_{x_l}^{x_r} u_h \, dx = \sum_{i=1}^{n} \left(\widehat{v_h}(x_{i+\frac{1}{2}}) - \widehat{v_h}(x_{i-\frac{1}{2}}) \right) = \widehat{v_h}(x_{n+\frac{1}{2}}) - \widehat{v_h}(x_{\frac{1}{2}}) = 0,$$

by applying the periodic boundary condition for v_h, and this concludes the proof. □

Proposition 2.2. *Let u_h and v_h be the solutions of the scheme (2.8) – (2.12) with flux (2.13). The energy $E_h(t)$ defined as*

$$E_h(t) = \int_I \left(\frac{v_h^2}{2} + \frac{((u_h)_t)^2}{2} + \frac{u_h^2}{2} + \frac{u_h^3}{3} \right) dx \tag{2.19}$$

is invariant in time t.

Proof. Summing up the scheme (2.8) – (2.12) over all cells $i = 1, 2, \ldots, J$, we get

$$((u_h)_t, \phi)_I - (w_h, \phi)_I = 0, \tag{2.20}$$

$$(w_h, \psi)_I + (v_h, \psi_x)_I + \sum_{i=1}^{J} \widehat{v_h}[\psi]_{i+\frac{1}{2}} = 0, \tag{2.21}$$

$$(r_h,\varphi)_I-(v_h,\varphi)_I-(w_h,\varphi_x)_I-\sum_{i=1}^{J}\widehat{w_h}[\varphi]_{i+\frac{1}{2}}=0, \tag{2.22}$$

$$(s_h,\xi)_I-(u_h+u_h^2,\xi)_I=0, \tag{2.23}$$

$$((r_h)_t,\zeta)_I+(s_h,\zeta_x)_I+\sum_{i=1}^{J}\widehat{s_h}[\zeta]_{i+\frac{1}{2}}=0, \tag{2.24}$$

by using the periodic boundary conditions. Taking the time derivative of Eq. (2.22), adding the resulting equation to Eq. (2.21) and choosing the test functions $\psi=(w_h)_t$, $\varphi=-v_h$ yield

$$\begin{aligned}0&=(w_h,(w_h)_t)_I+(v_h,(w_h)_{xt})_I+\sum_{i=1}^{J}\widehat{v_h}[(w_h)_t]_{i+\frac{1}{2}}\\&\quad-((r_h)_t,v_h)_I+((v_h)_t,v_h)_I+((w_h)_t,(v_h)_x)_I+\sum_{i=1}^{J}\widehat{(w_h)_t}[v_h]_{i+\frac{1}{2}}\\&=(w_h,(w_h)_t)_I-((r_h)_t,v_h)_I+((v_h)_t,v_h)_I,\end{aligned} \tag{2.25}$$

where the last equality comes from the choice of the alternating flux (2.13). Next, setting $\zeta=v_h$ in Eq. (2.24), $\psi=s_h$ in Eq. (2.21), $\phi=s_h$ in Eq. (2.20), and summing them up, we have

$$\begin{aligned}0&=((u_h)_t,s_h)_I+(v_h,(s_h)_x)_I+\sum_{i=1}^{J}\widehat{v_h}[s_h]_{i+\frac{1}{2}}+((r_h)_t,v_h)_I+(s_h,(v_h)_x)_I+\sum_{i=1}^{J}\widehat{s_h}[v_h]_{i+\frac{1}{2}}\\&=((u_h)_t,s_h)_I+((r_h)_t,v_h)_I,\end{aligned} \tag{2.26}$$

where the last equality again follows from the choice of the alternating flux (2.13).

Summing up (2.25) and (2.26), we get

$$((u_h)_t,s_h)_I+((w_h)_t,w_h)_I+((v_h)_t,v_h)_I=0,$$

which leads to

$$\begin{aligned}\frac{d}{dt}E_h(t)&=\frac{d}{dt}\int_I\left(\frac{v_h^2}{2}+\frac{((u_h)_t)^2}{2}+\frac{u_h^2}{2}+\frac{u_h^3}{3}\right)dx\\&=((v_h)_t,v_h)_I+((u_h)_{tt},(u_h)_t)_I+((u_h)_t,u_h+u_h^2)_I\\&=((v_h)_t,v_h)_I+((w_h)_t,w_h)_I+((u_h)_t,s_h)_I\\&=0.\end{aligned}$$

This concludes the proof. □

3. Error estimate

In this section, we derive an optimal error estimate for the energy conserving LDG method proposed in Section 2. We define the following errors associated with a function f by

$$e^f=f-f_h=\eta^f+\zeta^f,\qquad \eta^f=f-P^*f,\qquad \zeta^f=P^*f-f_h,$$

which from left to right, respectively denote the error between the exact solution f and the numerical solution f_h, the projection error between f and a particular projection P^* of f, and the error between the numerical solution and the projection of f. f could be each of these functions u, v, w, r, s. The standard L^2 projection, i.e., $P^*=P$, is used for functions u, w, r, s, and the Radau projection $P^*=P^-$ is used for the function v. Below we list the main error estimate result.

Proposition 3.1. *Let u, v, w, s, r be the exact solutions of the Eqs. (2.7), and u_h, v_h, w_h, s_h, r_h be the numerical solutions of the semi-discrete LDG method (2.8) – (2.12), with the numerical fluxes defined in (2.13) and the initial conditions defined as*

$$u_h(x,0)=Pu(x,0),\quad v_h(x,0)=P^-v(x,0). \tag{3.1}$$

When the exact solution is sufficiently smooth, there holds the following error estimates:

$$(\|e^u\|^2 + \|e^v\|^2 + \|e^w\|^2)^{\frac{1}{2}} \le Ch^{k+1}, \tag{3.2}$$

where the constant C depends on k, u and the final time T, but is independent of h.

Before we present the detailed proof of this proposition, some useful lemmas and remarks are provided below.

Remark 3.1. For the initial projection defined in (3.1), one can follow the proof in [33] to show the following estimate:

$$\|\zeta^u(0)\| = 0, \quad \|\zeta^v(0)\| = 0, \quad \|\zeta^w(0)\| \le Ch^{k+1}. \tag{3.3}$$

Remark 3.2. To deal with the nonlinear term, we would like to make an a priori error estimate assumption that

$$\|u - u_h\| \le h,$$

following the setup in [34,23], where the same technique was used to treat the nonlinearity in the KdV and Keller-Segel models. This assumption can be easily verified, and we refer to [34,23] for the explanation and proof. This assumption implies that

$$\|u - u_h\|_\infty \le C, \qquad \|\zeta^u\|_\infty \le C.$$

The following lemma is one of the key ideas to derive the optimal error estimate.

Lemma 3.1. *Suppose $(v_h, w_h) \in V_h^k \times V_h^k$ satisfies the equation (2.9) with the flux $\widehat{v_h} = v_h^-$, the L^2 projections P is used for the variable w and the Radau projection P^- is used for v, then there exists a positive constant C, which is independent of h, such that*

$$\|\zeta_x^v\| + h^{-1/2}|[\zeta^v]| \le C\|\zeta^w\|. \tag{3.4}$$

Remark 3.3. This Lemma provides an important relationship between the error of the auxiliary variable and the primary variable, and was proven in [24, Lemma 2.4]. Similar result on the variable (instead of the error)

$$\|v_{h,x}\| + h^{-1/2}|[v_h]| \le C\|w_h\|,$$

appeared first in [31]. This provides a direct link between the DG polynomial w_h and the derivative, jump of v_h.

With these, we can now present the proof of our main result.

Proof. By subtracting the LDG methods (2.20)-(2.24) from the corresponding weak formulation satisfied by the exact solutions, we can derive the error equations

$$(e_t^u, \phi) - (e^w, \phi) = 0, \tag{3.5}$$

$$(e^w, \psi) + (e^v, \psi_x) + \sum_{i=1}^{J} e^{v,-}[\psi]_{i+\frac{1}{2}} = 0, \tag{3.6}$$

$$(e^r, \varphi) - (e^v, \varphi) - (e^w, \varphi_x) - \sum_{i=1}^{J} e^{w,+}[\varphi]_{i+\frac{1}{2}} = 0, \tag{3.7}$$

$$(e^s, \xi) - (u + u^2 - u_h - u_h^2, \xi) = 0, \tag{3.8}$$

$$(e_t^r, \zeta) + (e^s, \zeta_x) + \sum_{i=1}^{J} e^{s,+}[\zeta]_{i+\frac{1}{2}} = 0. \tag{3.9}$$

In Eqs. (3.5), (3.6) and (3.9), taking the test functions $\phi = \psi = \zeta^s$ and $\zeta = \zeta^v$, and summing them up leads to

$$\begin{aligned}(e_t^u, \zeta^s) + (e_t^r, \zeta^v) = &-(\eta^v, \zeta_x^s) - \sum_{i=1}^{J} \eta^{v,-}[\zeta^s]_{i+\frac{1}{2}} - (\zeta^v, \zeta_x^s) - \sum_{i=1}^{J} \zeta^{v,-}[\zeta^s]_{i+\frac{1}{2}} \\ &- (\eta^s, \zeta^v)_x - \sum_{i=1}^{J} \eta^{s,+}[\zeta^v]_{i+\frac{1}{2}} - (\zeta^s, \zeta_x^v) - \sum_{i=1}^{J} \zeta^{s,+}[\zeta^v]_{i+\frac{1}{2}}\end{aligned}$$

$$= -(\eta^v, \zeta_x^s) - \sum_{i=1}^{J} \eta^{v,-}[\zeta^s]_{i+\frac{1}{2}} - (\eta^s, \zeta_x^v) - \sum_{i=1}^{J} \eta^{s,+}[\zeta^v]_{i+\frac{1}{2}}. \tag{3.10}$$

Next, by taking the time derivative of Eq. (3.7), adding the resulting equation to Eq. (3.6) and choosing the test functions $\varphi = \zeta^v$, $\phi = -\zeta_t^w$, one obtains

$$\begin{aligned}(e_t^r, \zeta^v) - (e^w, \zeta_t^w) &= (e_t^v, \zeta^v) + (\eta_t^w, \zeta_x^v) + \sum_{i=1}^{J} \eta_t^{w,+}[\zeta^v]_{i+\frac{1}{2}} + (\zeta_t^w, \zeta_x^v) + \sum_{i=1}^{J} \zeta_t^{w,+}[\zeta^v]_{i+\frac{1}{2}} \\ &\quad + (\eta^v, \zeta_{xt}^w) + \sum_{i=1}^{J} \eta^{v,-}[\zeta_t^w]_{i+\frac{1}{2}} + (\zeta^v, \zeta_{xt}^w) + \sum_{i=1}^{J} \zeta^{v,-}[\zeta^v]_{i+\frac{1}{2}} \\ &= (e_t^v, \zeta^v) + (\eta_t^w, \zeta_x^v) + \sum_{i=1}^{J} \eta_t^{w,+}[\zeta^v]_{i+\frac{1}{2}} + (\eta^v, \zeta_{xt}^w) + \sum_{i=1}^{J} \zeta^{v,-}[\zeta_t^w]_{i+\frac{1}{2}}. \end{aligned} \tag{3.11}$$

Combine (3.10) and (3.11), we have

$$\begin{aligned}(e_t^u, \zeta^s) + (e^w, \zeta_t^w) + (e_t^v, \zeta^v) &= -(\eta^v, \zeta_x^s) - \sum_{i=1}^{J} \eta^{v,-}[\zeta^s]_{i+\frac{1}{2}} - (\eta^s, \zeta_x^v) - \sum_{i=1}^{J} \eta^{s,+}[\zeta^v]_{i+\frac{1}{2}} \\ &\quad - (\eta_t^w, \zeta_x^v) - \sum_{i=1}^{J} \eta_t^{w,+}[\zeta^v]_{i+\frac{1}{2}} - (\eta^v, \zeta_{xt}^w) - \sum_{i=1}^{J} \eta^{v,-}[\zeta_t^w]_{i+\frac{1}{2}}. \end{aligned} \tag{3.12}$$

By the choice of the projections of u, v, w, s, r, we have

$$(e_t^u, \zeta^s) + (e^w, \zeta_t^w) + (e_t^v, \zeta^v) = -\sum_{i=1}^{J} \eta^{s,+}[\zeta^v]_{i+\frac{1}{2}} - \sum_{i=1}^{J} \eta_t^{w,+}[\zeta^v]_{i+\frac{1}{2}},$$

which leads to

$$\begin{aligned}(\zeta_t^u, \zeta^s) + \frac{1}{2}\frac{d}{dt}(\|\zeta^w\|^2 + \|\zeta^v\|^2) &= -(\eta_t^u, \zeta^s) - (\eta^w, \zeta_t^w) - (\eta_t^v, \zeta^v) - \sum_{i=1}^{J} \eta^{s,+}[\zeta^v]_{i+\frac{1}{2}} - \sum_{i=1}^{J} \eta_t^{w,+}[\zeta^v]_{i+\frac{1}{2}} \\ &= -(\eta_t^v, \zeta^v) - \sum_{i=1}^{J} \eta^{s,+}[\zeta^v]_{i+\frac{1}{2}} - \sum_{i=1}^{J} \eta_t^{w,+}[\zeta^v]_{i+\frac{1}{2}}, \end{aligned} \tag{3.13}$$

where the last equality comes from the L^2 projection property: $(\eta^u, f) = (\eta^w, f) = 0$ for any $f \in V^k$. Note that

$$\begin{aligned}(\zeta_t^u, \zeta^s) &= (e^s - \eta^s, \zeta_t^u) = (e^s, \zeta_t^u) = (u + u^2 - u_h - u_h^2, \zeta_t^u) \\ &= (\zeta^u + \eta^u, \zeta_t^u) + (u^2 - u_h^2, \zeta_t^u) = (\zeta^u, \zeta_t^u) + (u^2 - u_h^2, \zeta_t^u), \end{aligned} \tag{3.14}$$

the error equation (3.13) becomes

$$\frac{1}{2}\frac{d}{dt}(\|\zeta^u\|^2 + \|\zeta^w\|^2 + \|\zeta^v\|^2) = -(u^2 - u_h^2, \zeta_t^u) - (\eta_t^v, \zeta^v) - \sum_{i=1}^{J} \eta^{s,+}[\zeta^v]_{i+\frac{1}{2}} - \sum_{i=1}^{J} \eta_t^{w,+}[\zeta^v]_{i+\frac{1}{2}}. \tag{3.15}$$

We now focus on the terms on the right hand side. One can show that

$$-\sum_{i=1}^{J} \eta^{s,+}[\zeta^v]_{i+\frac{1}{2}} \le Ch \sum_{i=1}^{J} (\eta_{i+\frac{1}{2}}^{s,+})^2 + Ch^{-1} \sum_{i=1}^{J} ([\zeta^v]_{i+\frac{1}{2}})^2 \le Ch^{2k+2} + C\|\zeta^w\|^2, \tag{3.16}$$

from the projection error (2.6) and the approximation (3.4) in Lemma 3.1. Similarly, we have

$$-\sum_{i=1}^{J} \eta_t^{w,+}[\zeta^v]_{i+\frac{1}{2}} \le Ch^{2k+2} + C\|\zeta^w\|^2. \tag{3.17}$$

Using the fact that

$$\zeta_t^u = Pu_t - (u_h)_t = Pw - w_h = \zeta^w,$$

the first term $-(u^2 - u_h^2, \zeta_t^u)$ can be estimated by:

$$\begin{aligned}
-(u^2 - u_h^2, \zeta_t^u) &= -(u^2 - u_h^2, \zeta^w) = -(2u(u-u_h) - (u-u_h)^2, \zeta^w) \\
&= -(2u\zeta^u + 2u\eta^u - (\zeta^u)^2 - 2\zeta^u\eta^u - (\eta^u)^2, \zeta^w) \\
&= -(2u\zeta^u, \zeta^w) - (2u\eta^u, \zeta^w) + ((\zeta^u)^2, \zeta^w) + 2(\zeta^u\eta^u, \zeta^w) + ((\eta^u)^2, \zeta^w) \\
&\le C|u|_\infty(\|\zeta^u\| + \|\eta^u\|)\|\zeta^w\| + C\|\zeta^u\|_\infty(\|\zeta^u\|^2 + \|\zeta^w\|^2) + C\|\eta^u\|_\infty\|\zeta^w\|(\|\zeta^u\| + \|\eta^u\|) \\
&\le C\|\zeta^u\|^2 + C\|\zeta^w\|^2 + Ch^{2k+2},
\end{aligned} \tag{3.18}$$

where the last equality comes from the assumption in Remark 3.2. Combining the results in Eqs. (3.13) – (3.18), we have

$$\frac{d}{dt}(\|\zeta^u\|^2 + \|\zeta^v\|^2 + \|\zeta^w\|^2) \le C(\|\zeta^u\|^2 + \|\zeta^v\|^2 + \|\zeta^w\|^2 + h^{2k+2}).$$

By using Gronwall's inequality and recalling the property of initial conditions in Remark 3.1, we obtain that

$$(\|\zeta^u\|^2 + \|\zeta^v\|^2 + \|\zeta^w\|^2)^{\frac{1}{2}} \le Ch^{k+1}.$$

The optimal error estimate (3.2) follows from this and the optimal projection error shown in (2.6). This concludes the proof. □

4. Temporal discretization

Energy conserving spatial discretization was presented in Section 2. In this section, we develop fully discrete methods which maintain the mass and energy conserving property, by presenting two different energy-conserving temporal discretizations. Both the explicit leap-frog and the implicit midpoint methods, with special attention paid to the discretization of the nonlinear term, will be considered.

Let $0 = t_0 < t_1 < \ldots < t_N = T$ be a partition of the interval $[0, T]$ with time step $\Delta t_n = t_{n+1} - t_n$. A uniform time step $\Delta t_n = \tau$ is considered in this paper, and for the implicit temporal discretization in section 4.2, nonuniform time step can also be used. The following notations

$$u_h^n = u_h(\cdot, t_n), \qquad u_h^{n+\frac{1}{2}} = \frac{1}{2}(u_h^{n+1} + u_h^n), \qquad \delta_t u_h^n = \frac{u_h^{n+1} - u_h^{n-1}}{2\tau}, \qquad \delta_t^+ u_h^n = \frac{u_h^{n+1} - u_h^n}{\tau},$$

are introduced to ease the presentation.

4.1. Explicit leap-frog method

The second order explicit leap-frog method is known to conserve the discrete energy. When coupled with the LDG method (2.8)–(2.11), the fully discrete explicit leap-frog local discontinuous Galerkin (ELF-LDG) scheme takes the following form: we are looking for the solutions u_h^{n+1}, v_h^{n+1}, w_h^{n+1}, s_h^{n+1}, and $r_h^{n+1} \in V^k$, for $n = 1, 2, \ldots, N-1$, such that

$$\int_{I_i} \delta_t u_h^n \phi \, dx = \int_{I_i} \frac{u_h^{n+1} - u_h^{n-1}}{2\tau}\phi \, dx = \int_{I_i} w_h^n \phi \, dx, \tag{4.1}$$

$$\int_{I_i} w_h^n \psi \, dx = \widehat{v_h^n}\psi^-|_{x_{i+\frac{1}{2}}} - \widehat{v_h^n}\psi^+|_{x_{i-\frac{1}{2}}} - \int_{I_i} v_h^n \psi_x \, dx, \tag{4.2}$$

$$\int_{I_i} r_h^n \varphi \, dx = \int_{I_i} v_h^n \varphi \, dx - \left(\widehat{w_h^n}\varphi^-|_{x_{i+\frac{1}{2}}} - \widehat{w_h^n}\varphi^+|_{x_{i-\frac{1}{2}}} - \int_{I_i} w_h^n \varphi_x \, dx \right), \tag{4.3}$$

$$\int_{I_i} s_h^n \xi \, dx = \int_{I_i} \left(u_h^n + \frac{1}{3}\left((u_h^{n+1})^2 + u_h^{n+1}u_h^{n-1} + (u_h^{n-1})^2 \right) \right) \xi \, dx, \tag{4.4}$$

$$\int_{I_i} \delta_t r_h^n \zeta \, dx = \int_{I_i} \frac{r_h^{n+1} - r_h^{n-1}}{2\tau}\zeta \, dx = \widehat{s_h^n}\zeta^-|_{x_{i+\frac{1}{2}}} - \widehat{s_h^n}\zeta^+|_{x_{i-\frac{1}{2}}} - \int_{I_i} s_h^n \zeta_x \, dx, \tag{4.5}$$

hold for all test functions $\phi, \psi, \varphi, \xi, \zeta \in V^k$. It is easy to verify that the explicit leap-frog method with this nonlinear term treatment is second order accurate in time.

The leap-frog method is a multi-step method, which requires numerical initial conditions for the first two time steps. Below we briefly mention how these two numerical initial conditions are provided in our numerical experiments. At the first time step $t_0 = 0$, the exact initial conditions of $u(x,0)$ and $u_t(x,0)$ are given in (2.3). The equation $u_t = v_x$ leads to $v(x,0) = v(x_l,0) + \int_{x_l}^{x} u_t(s,0)ds = v(x_l,0) + \int_{x_l}^{x} g(s)ds$. Since the constant term $v(x_l,0)$ won't affect the computation, we assume

$$v(x,0) = \int_{x_l}^{x} g(s)ds,$$

and define our numerical initial conditions of u_h and v_h as

$$u_h^0 = Pu(x,0), \qquad v_h^0 = P^* v(x,0), \tag{4.6}$$

where P is the standard L^2 projection and P^* could be either the L^2 projection P or the Radau projection $P^{\pm}$. In the numerical section, we will analyze the effect of different choice of P^* towards the accuracy. At the second time step $t_1 = \tau$, we consider the Taylor expansion of $u(x,\tau)$ and $v(x,\tau)$ at $t = 0$ via

$$u(x,\tau) = u(x,0) + \tau u_t(x,0) + \frac{\tau^2}{2} u_{tt}(x,0) + O(\tau^3),$$
$$v(x,\tau) = v(x,0) + \tau v_t(x,0) + \frac{\tau^2}{2} v_{tt}(x,0) + O(\tau^3),$$

which leads to the following choice of numerical initial conditions

$$u_h^1 = u_h^0 + \tau u_{h,t}^0 + \frac{\tau^2}{2} u_{h,tt}^0, \qquad v_h^1 = v_h^0 + \tau v_{h,t}^0 + \frac{\tau^2}{2} v_{h,tt}^0, \tag{4.7}$$

where $u_{h,t}^0$, $u_{h,tt}^0$, $v_{h,t}^0$, and $v_{h,tt}^0$ are all computed via the numerical methods (2.8)-(2.12) with the initial condition (4.6).

Next, we provide some details related to the implementation of the ELF-LDG method (4.1)-(4.5). Note that u_h^{n+1} appears on the right hand side of Eq. (4.4), but this won't affect the explicit property of the ELF-LDG scheme, as illustrated below.

Let U_h be the vectors containing the degree of freedom for the piecewise polynomial solution u_h, and denote $U_h^n = U_h(t_n)$. Similarly, we can define V_h^n, W_h^n, R_h^n and S_h^n. The ELF-LDG method (4.1)-(4.5) can be rewritten in the matrix form as

$$\begin{aligned}
&U_h^{n+1} = 2\tau W_h^n + U_h^{n-1},\\
&W_h^n = M_v V_h^n,\\
&R_h^n = V_h^n - M_w W_h^n,\\
&S_h^n = f_s(U_h^{n-1}, U_h^n, U_h^{n+1}),\\
&R_h^{n+1} = 2\tau M_s S_h^n + R_h^{n-1},
\end{aligned} \tag{4.8}$$

where M_v, M_w and M_s are matrices depending on the polynomial basis functions and the choices of the numerical fluxes $\widehat{v_h}$, $\widehat{w_h}$, $\widehat{s_h}$, respectively. The nonlinear function f_s comes from the discretization of the nonlinear term. The combination of the second and third equations leads to the following relation between R_h^n and V_h^n:

$$R_h^n = K_{vr} V_h^n, \qquad V_h^n = K_{vr}^{-1} R_h^n, \tag{4.9}$$

where $K_{vr} = I - M_w M_v$. Therefore, the system (4.8) can be reduced to

$$\begin{aligned}
&U_h^{n+1} = 2\tau M_v V_h^n + U_h^{n-1},\\
&V_h^{n+1} = 2\tau K_{vr}^{-1} M_s f_s(U_h^{n-1}, U_h^n, U_h^{n+1}) + V_h^{n-1}.
\end{aligned} \tag{4.10}$$

Note that one can evaluate U_h^{n+1} first, and then compute V_h^{n+1}, in an explicit way. The matrices K_{vr} and M_v are both sparse block matrices, hence their multiplication with coefficient vectors can be implemented efficiently. The evaluation of V_h^{n+1} involves a linear solver with the matrix K_{vr}, and one can perform an LU decomposition of K_{vr} at the initial time to save computational cost. The existence of the solution sequences $\{U_h^n\}_{n=0}^N$ and $\{V_h^n\}_{n=0}^N$ can be easily observed.

The conservation of continuous mass and energy of the semi-discrete LDG methods was shown in Section 2.4. Below, we prove that the fully discrete ELF-LDG methods can conserve the discrete mass and energy exactly.

Proposition 4.1. *Let u_h^n, v_h^n and w_h^n be the solutions of the fully discrete ELF-LDG methods (4.1)-(4.5). The discrete mass and energy defined as*

$$M_h^n = \int_I u_h^n dx, \qquad E_h^n = \int_I \left(\frac{u_h^n u_h^{n+1}}{2} + \frac{v_h^n v_h^{n+1}}{2} + \frac{w_h^n w_h^{n+1}}{2} + \frac{(u_h^{n+1})^3 + (u_h^n)^3}{6} \right) dx, \tag{4.11}$$

are invariant for all n.

Proof. We start with the conservation of the discrete mass M_h^n. Choose $\phi = 1$ and $\psi = 1$ in (4.1) and (4.2), and sum up over all cells. With periodic boundary condition, this leads to $M_h^{n+1} = M_h^{n-1}$. From the evaluation of u_h^1 and v_h^1 in (4.7), we can easily conclude that $M_h^0 = M_h^1$, therefore, the conservation of the discrete mass

$$M_h^n = M_h^0,$$

can be observed.

Next, we consider the conservation of the discrete energy E_h^n. Apply the operator δ_t on the equation (4.3), choose the test function $\varphi = v_h^n$, and sum up over all cells to obtain

$$(\delta_t r_h^n, v_h^n)_I = (\delta_t v_h^n, v_h^n)_I + (\delta_t w_h^n, v_{h,x}^n)_I + \sum_{i=1}^{J} \widehat{\delta_t w_h^n} [v_h^n]_{i+\frac{1}{2}}. \tag{4.12}$$

Choosing the test function $\psi = \delta_t w_h^n$ in the equation (4.2) yields

$$(w_h^n, \delta_t w_h^n)_I + (v_h^n, \delta_t w_{x,h}^n)_I + \sum_{i=1}^{J} \widehat{v_h^n} [\delta_t w_h^n]_{i+\frac{1}{2}} = 0. \tag{4.13}$$

By subtracting (4.13) from (4.12), we obtain

$$(\delta_t r_h^n, v_h^n)_I = (\delta_t v_h^n, v_h^n)_I + (w_h^n, \delta_t w_h^n)_I - \sum_{i=1}^{J} [v_h^n \delta_t w_h^n] + \sum_{i=1}^{J} \widehat{v_h^n} [\delta_t w_h^n]_{i+\frac{1}{2}} + \sum_{i=1}^{J} \widehat{\delta_t w_h^n} [v_h^n]_{i+\frac{1}{2}}. \tag{4.14}$$

Setting $\psi = \phi = s_h^n$ in the equations (4.2) and (4.1) leads to

$$(\delta_t u_h^n, s_h^n)_I + (v_h^n, s_{x,h}^n)_I + \sum_{i=1}^{J} \widehat{v_h^n} [s_h^n]_{i+\frac{1}{2}} = (\delta_t u_h^n, s_h^n)_I - (s_h^n, v_{h,x}^n)_I - \sum_{i=1}^{J} [s_h^n v_h^n]_{i+\frac{1}{2}} + \sum_{i=1}^{J} \widehat{v_h^n} [s_h^n]_{i+\frac{1}{2}} = 0, \tag{4.15}$$

and choosing $\zeta = v_h^n$ in the equation (4.5) gives

$$(\delta_t r_h^n, v_h^n)_I + (s_h^n, v_{h,x}^n)_I + \sum_{i=1}^{J} \widehat{s_h^n} [v_h^n]_{i+\frac{1}{2}} = 0. \tag{4.16}$$

Sum up (4.15) and (4.16) to get

$$(\delta_t u_h^n, s_h^n)_I + (\delta_t r_h^n, v_h^n)_I - \sum_{i=1}^{J} [s_h^n v_h^n]_{i+\frac{1}{2}} + \sum_{i=1}^{J} \widehat{v_h^n} [s_h^n]_{i+\frac{1}{2}} + \sum_{i=1}^{J} \widehat{s_h^n} [v_h^n]_{i+\frac{1}{2}} = 0. \tag{4.17}$$

Combining the equations (4.14) and (4.17), we have

$$\begin{aligned} &(\delta_t v_h^n, v_h^n)_I + (w_h^n, \delta_t w_h^n)_I + (\delta_t u_h^n, s_h^n)_I + \sum_{i=1}^{J} \left(-[v_h^n \delta_t w_h^n] + \widehat{v_h^n} [\delta_t w_h^n] + \widehat{\delta_t w_h^n} [v_h^n]_{i+\frac{1}{2}} \right)_{i+\frac{1}{2}} \\ &\quad + \sum_{i=1}^{J} \left(-[s_h^n v_h^n] + \widehat{v_h^n} [s_h^n]_{i+\frac{1}{2}} + \widehat{s_h^n} [v_h^n] \right)_{i+\frac{1}{2}} = 0, \end{aligned}$$

and by the choice of numerical fluxes (2.13) or (2.14), this becomes

$$(\delta_t v_h^n, v_h^n)_I + (w_h^n, \delta_t w_h^n)_I + (\delta_t u_h^n, s_h^n)_I = 0. \tag{4.18}$$

It is easy to observe that

$$(\delta_t v_h^n, v_h^n)_I = \frac{1}{2\tau}\int_I (v_h^{n+1}v_h^n - v_h^n v_h^{n-1})dx,$$

$$(\delta_t w_h^n, w_h^n)_I = \frac{1}{2\tau}\int_I (v_h^{n+1}v_h^n - v_h^n v_h^{n-1})dx,$$

$$\begin{aligned}(\delta_t u_h^n, s_h^n)_I &= (\delta_t u_h^n, u_h^n)_I + \left(\delta_t u_h^n, \frac{1}{3}\left((u_h^{n+1})^2 + (u_h^{n+1})(u_h^{n-1}) + (u_h^{n-1})^2\right)\right)_I \\ &= \frac{1}{2\tau}\int_I (u_h^{n+1}u_h^n - u_h^n u_h^{n-1})dx + \int_I \left(\frac{(u_h^{n+1})^3 + (u_h^n)^3}{6\tau} - \frac{(u_h^n)^3 + (u_h^{n-1})^3}{6\tau}\right)dx,\end{aligned}$$

therefore, the equation (4.18) reduces to

$$E_h^{n+1} = E_h^n,$$

which completes the proof. □

Remark 4.1. The nonlinear term u^2 is approximated by $\left((u_h^{n+1})^2 + (u_h^{n+1})(u_h^{n-1}) + (u_h^{n-1})^2\right)/3$ in the equation (4.4), which is crucial in the proof of energy conservation. One may also approximate it by $u_h^n(u_h^{n+1} + u_h^n + u_h^{n-1})/3$, which will leads to an energy conservation method with the newly defined energy

$$\hat{E}_h^n = \int_I \left(\frac{u_h^n u_h^{n+1}}{2} + \frac{v_h^n v_h^{n+1}}{2} + \frac{w_h^n w_h^{n+1}}{2} + \frac{u_h^n u_h^{n+1}(u_h^{n+1} + u_h^n)}{6}\right)dx.$$

We have tried both approaches numerically, and no obvious difference has been observed in their numerical performance.

4.2. Implicit midpoint method

In this subsection, we present an energy conserving implicit midpoint rule temporal discretization. The fully discrete scheme implicit midpoint rule local discontinuous Galerkin (IMR-LDG) scheme takes the following form: we are looking for the solutions u_h^{n+1}, v_h^{n+1}, w_h^{n+1}, s_h^{n+1}, and $r_h^{n+1} \in V^k$, for $n = 1, 2, ..., N-1$, such that

$$\int_{I_i} \delta_t^+ u_h^n \phi\, dx = \int_{I_i} \frac{u_h^{n+1} - u_h^n}{\tau}\phi\, dx = \int_{I_i} w_h^{n+\frac{1}{2}}\phi\, dx, \tag{4.19}$$

$$\int_{I_i} w_h^{n+\frac{1}{2}}\psi\, dx = \widehat{v_h^{n+\frac{1}{2}}}\psi^-|_{x_{i+\frac{1}{2}}} - \widehat{v_h^{n+\frac{1}{2}}}\psi^+|_{x_{i-\frac{1}{2}}} - \int_{I_i} v_h^{n+\frac{1}{2}}\psi_x\, dx, \tag{4.20}$$

$$\int_{I_i} r_h^{n+\frac{1}{2}}\varphi\, dx = \int_{I_i} v_h^{n+\frac{1}{2}}\varphi\, dx - \widehat{w_h^{n+\frac{1}{2}}}\varphi^-|_{x_{i+\frac{1}{2}}} + \widehat{w_h^{n+\frac{1}{2}}}\varphi^+|_{x_{i-\frac{1}{2}}} + \int_{I_i} w_h^{n+\frac{1}{2}}\varphi_x\, dx, \tag{4.21}$$

$$\int_{I_i} s_h^{n+\frac{1}{2}}\xi\, dx = \int_{I_i} \left(u_h^{n+\frac{1}{2}} + \frac{1}{3}\left((u_h^{n+1})^2 + (u_h^{n+1})(u_h^n) + (u_h^n)^2\right)\right)\xi\, dx, \tag{4.22}$$

$$\int_{I_i} \delta_t^+ r_h^n \zeta\, dx = \int_{I_i} \frac{r_h^{n+1} - r_h^n}{\tau}\zeta\, dx = \widehat{s_h^{n+\frac{1}{2}}}\zeta^-|_{x_{i+\frac{1}{2}}} - \widehat{s_h^{n+\frac{1}{2}}}\zeta^+|_{x_{i-\frac{1}{2}}} - \int_{I_i} s_h^{n+\frac{1}{2}}\zeta_x\, dx, \tag{4.23}$$

hold for all test functions ϕ, ψ, φ, ξ, $\zeta \in V^k$. Recall that $\omega^{n+\frac{1}{2}} = (\omega^n + \omega^{n+1})/2$, where $\omega = u_h, v_h, \cdots$. Easy to verify that the implicit midpoint rule method with this nonlinear term treatment is second order accurate in time.

Next, we provide details related to the implementation of the IMR-LDG method (4.19)-(4.23). One can define U_h^n, V_h^n, W_h^n, R_h^n and S_h^n as before. Denote $\omega_h^{n+\frac{1}{2}} = \frac{1}{2}(\omega_h^{n+1} + \omega_h^n)$ for $\omega = U$, V, W, R or S. The IMR-LDG method (4.19)-(4.19) can be rewritten in the matrix form as

$$U_h^{n+\frac{1}{2}} = \frac{\tau}{2}W_h^{n+\frac{1}{2}} + U_h^n,$$

$$W_h^{n+\frac{1}{2}} = M_v V_h^{n+\frac{1}{2}},$$

$$R_h^{n+\frac{1}{2}} = V_h^{n+\frac{1}{2}} - M_w W_h^{n+\frac{1}{2}}, \tag{4.24}$$
$$S_h^{n+\frac{1}{2}} = f_{s'}(U_h^n, U_h^{n+\frac{1}{2}}),$$
$$R_h^{n+\frac{1}{2}} = \frac{\tau}{2} M_s S_h^{n+\frac{1}{2}} + R_h^n,$$

where the nonlinear function $f_{s'}$ comes from the discretization of the nonlinear term, and the matrices M_v, M_w and M_s were discussed in Section 4.1. Again, the combination of the second and third equations leads to

$$R_h^{n+\frac{1}{2}} = K_{vr} V_h^{n+\frac{1}{2}}, \qquad V_h^{n+\frac{1}{2}} = K_{vr}^{-1} R_h^{n+\frac{1}{2}}. \tag{4.25}$$

Therefore, the system (4.24) can be reduced to

$$U_h^{n+\frac{1}{2}} = \frac{\tau}{2} M_v V_h^{n+\frac{1}{2}} + U_h^n, \tag{4.26}$$
$$V_h^{n+\frac{1}{2}} = \frac{\tau}{2} K_{vr}^{-1} M_s f_{s'}(U_h^n, U_h^{n+\frac{1}{2}}) + V_h^n. \tag{4.27}$$

We can plug the first equation into the second one to derive one nonlinear equation involving $V_h^{n+\frac{1}{2}}$. To solve this nonlinear equation, one could use Newton's method or fixed point iteration. Since iterative method is used, a stopping criterion of $\|(V_h^{n+\frac{1}{2}})^{(k)} - (V_h^{n+\frac{1}{2}})^{(k-1)}\| \le \varepsilon$ is used in the numerical implementation, where ε is the control error and is taken as 10^{-15} in our tests. The existence of the solution sequences $\{U_h^n\}_{n=0}^N$ and $\{V_h^n\}_{n=0}^N$ can be established via the fixed point theorem.

The next proposition show that the fully discrete IMR-LDG methods can conserve the discrete mass and energy exactly.

Proposition 4.2. *Let u_h^n, v_h^n and w_h^n be the solutions of the fully discrete IMR-LDG methods (4.19)-(4.23). The discrete mass and energy defined as*

$$M_h^n = \int_I u_h^n dx, \qquad \tilde{E}_h^n = \int_I \left(\frac{(u_h^n)^2}{2} + \frac{(v_h)^2}{2} + \frac{(w_h^n)^2}{2} + \frac{(u_h^n)^3}{3} \right) dx, \tag{4.28}$$

are invariant for all n.

The proof of this proposition is similar to that of Proposition 4.1, and is omitted here to save space.

4.3. Fourth order temporal discretization via extrapolation

Both temporal discretizations presented in the previous subsections are second order accurate. The Richardson extrapolation technique [39] is a practical method to achieve higher order numerical accuracy using lower order methods. The main idea is to combine the numerical solutions with various time step sizes in a particular way to obtain higher order accurate resolution. One advantage of this extrapolation technique is the preservation of the numerical stability of underlying lower-order methods.

In this paper, we use the extrapolation technique to derive fourth order accurate temporal discretization using the formula

$$u_E^n = \frac{4}{3} u_h^{2n}\left(\frac{\tau}{2}\right) - \frac{1}{3} u_h^n(\tau),$$

where $u_h^n(\tau)$ stands for the solution $u_h(x, t_n) = u_h(x, n * \tau)$ evaluated with the time step size τ, and $u_h^{2n}(\tau/2)$ stands for the solution $u_h(x, t_{2n}) = u_h(x, n * \tau)$ evaluated with the time step size $\tau/2$. Both the explicit and implicit fourth order temporal discretizations can be derived based on the explicit leap-frog and implicit midpoint rule methods.

5. Numerical experiments

In this section, we provide some numerical results of the proposed LDG methods with both implicit and explicit temporal discretizations. We first perform the accuracy tests on the methods with different initial projections and observe how that affects the order of accuracy. Errors of mass and energy are calculated to verify the mass and energy conservation property of our scheme, and we also studied the long time behavior of these methods. Examples of two solitary waves moving on a collision course, a solitary wave breaking up and solution blow-up are simulated to demonstrate the performance of our methods.

5.1. Accuracy test

The single solitary wave can be described by a special solution [3] of the improved Boussinesq equation (1.2):

$$u(x,t) = \alpha \operatorname{sech}^2\left(\sqrt{\frac{\alpha}{6}}\frac{x-\beta t - x_0}{\beta}\right), \tag{5.1}$$

where α is the amplitude of the solitary wave, β is the velocity, x_0 is the initial wave center and $\beta = \sqrt{1+\frac{2\alpha}{3}}$. To test the accuracy of our algorithm, we calculate the initial conditions by setting $t = 0$ in Eq. (5.1) and its derivative with respect to t:

$$u(x,0) = \alpha \operatorname{sech}^2\left(\sqrt{\frac{\alpha}{6}}\frac{x-x_0}{\beta}\right),$$

$$u_t(x,0) = 2\alpha\sqrt{\frac{\alpha}{6}} \operatorname{sech}^2\left(\sqrt{\frac{\alpha}{6}}\frac{x-x_0}{\beta}\right)\tanh\left(\sqrt{\frac{\alpha}{6}}\frac{x-x_0}{\beta}\right).$$

We perform two sets of simulations using the ELF-LDG and IMR-LDG (both with 4th order temporal discretization via extrapolation) schemes with the choice of numerical fluxes (2.13). In the first simulation we use L^2 initial projection for both u and v, and in the second simulation, we use L^2 initial projection for u and P^- initial projection for v (to match the fluxes (2.13), see (4.6) for details). In both simulations, we set:

$$\alpha = 0.5, \quad x_0 = 0, \quad -100 < x < 100, \quad t \in [0,1].$$

One could use the exact solution as the boundary conditions; however, in practice, periodic boundary conditions turn out to be a good approximation and therefore is used in the simulations because the solution almost vanishes at the boundaries.

Errors in L^2 and L^∞ norms are calculated. Tables 5.1 and 5.2 present the numerical order of accuracy for the ELF-LDG scheme with P^0, P^1, P^2 and P^3 basis for u and v with different choices of projections in evaluating the numerical initial conditions. Numerical results of the IMR-LDG scheme are given in Tables 5.3 and 5.4. As seen in these tables, when L^2 projections are applied to obtain the initial conditions of both u_h and v_h, only k-th order accuracy in u_h is obtained with basis functions in P^k for $k = 1, 2, 3$. In contrast, optimal order of accuracy can be obtained for all four cases when the initial projection of v is changed to P^-.

Based on these tables and other numerical tests we have done, we find that the effects of initial projection on the order of convergence are closely related to the choice of fluxes. When the fluxes (2.13) are chosen, if we use L^2 initial projection for u and P^+ initial projection for v, only suboptimal order of u can be obtained for P^1, P^2 and P^3 basis. When the fluxes (2.14) are chosen, the optimal order of accuracy can be obtained with P^+ initial projection for v while suboptimal order of accuracy is obtained with P^- initial projection for v. The same phenomena was also observed by us in [33,11], where energy conserving LDG methods were developed and studied for the second order linear wave equations. The improved Boussinesq equation studied here can be viewed as a generalization of the second order wave equation with the additional nonlinear term and fourth order derivative term, therefore our proposed methods are the extension of the energy conserving LDG methods studied in [33,11], with additional techniques introduced to numerically approximate the nonlinear term and fourth order derivative term, while keeping the energy conservation property.

5.2. Mass, energy conservation and long time behavior

In this test, we consider the single solitary wave problem, with the same setup as in section 5.1. The parameters $\alpha = 0.5$, $x_0 = 0$, $-40 < x < 40$, $h = 0.5$, $\tau = 0.05$ are used in this example. The test is run until the final stopping time $T = 250$.

First, we plot the time history of the error of mass and energy (i.e., $M_h^N - M_h^0$ and $E_h^N - E_h^0$) of our methods in Fig. 5.1, where we can observe that the mass and energy are both exactly preserved by our methods up to the machine error at the level of 10^{-14}.

We also show the time history of the numerical error in L^2 and L^∞ norms to check the long time behavior of our methods. Fig. 5.2 provides the time history of the error e_u of both ELF-LDG and IMR-LDG methods, with P^0, P^1 and P^2 polynomial basis. Fig. 5.3 presents the time history of the errors e_u, e_v and e_w when P^3 polynomial basis is used. From these figures, we can observe that the errors of ELF-LDG and IMR-LDG methods are similar when the same time step is used, although we would like to comment that, by design, IMR-LDG method allows larger time step. The error of ELF-LDG method tends to be more oscillating than that of IMR-LDG method. For all cases except the P^0 one, numerical errors do not grow significantly in time, which is consistent with our observations of energy conserving methods for other wave equations.

Table 5.1
Numerical error and convergence orders of u and w of **ELF-LDG** scheme in the single solitary wave example. L^2 projection for both u^0 and v^0.

	J	N	u				v			
			$\|e^u\|_2$	C_2	$\|e^u\|_\infty$	C_∞	$\|e^v\|_2$	C_2	$\|e^v\|_\infty$	C_∞
P^0	200	100	9.3230E−02	*	6.3510E−02	*	1.0936E−01	*	8.3849E−02	*
	400	200	4.7135E−02	0.9840	3.4033E−02	0.9001	5.4858E−02	0.9953	4.1718E−02	1.0071
	800	400	2.3658E−02	0.9945	1.7407E−02	0.9673	2.7428E−02	1.0001	2.0590E−02	1.0187
	1600	800	1.1846E−02	0.9979	8.8129E−03	0.9820	1.3708E−02	1.0006	1.0235E−02	1.0084
P^1	200	100	2.3293E−02	*	2.3803E−02	*	7.2707E−03	*	1.0541E−02	*
	400	200	1.2475E−02	0.9008	1.2039E−02	0.9835	1.7859E−03	2.0255	2.6515E−03	1.9911
	800	400	6.4414E−03	0.9537	6.0270E−03	0.9982	4.4008E−04	2.0208	6.5169E−04	2.0246
	1600	800	3.2697E−03	0.9782	3.0116E−03	1.0009	1.0909E−04	2.0123	1.6170E−04	2.0109
P^2	200	100	1.6790E−03	*	1.4164E−03	*	3.4423E−04	*	4.8748E−04	*
	400	200	4.0062E−04	2.0673	3.2527E−04	2.1225	4.2580E−05	3.0151	5.9007E−05	3.0464
	800	400	9.8736E−05	2.0206	8.0494E−05	2.0147	5.2808E−06	3.0113	7.3468E−06	3.0057
	1600	800	2.4593E−05	2.0053	2.0076E−05	2.0034	6.5712E−07	3.0065	9.0978E−07	3.0135
P^3	200	100	1.4156E−04	*	2.0905E−04	*	1.6208E−05	*	2.6062E−05	*
	400	200	1.8608E−05	2.9274	2.8671E−05	2.8662	1.0092E−06	4.0054	1.6892E−06	3.9475
	800	400	2.3676E−06	2.9745	3.6732E−06	2.9645	6.2806E−08	4.0062	1.0402E−07	4.0214
	1600	800	2.9792E−07	2.9904	4.6234E−07	2.9900	3.9149E−09	4.0039	6.4758E−09	4.0057

Table 5.2
Numerical error and convergence orders of u and w of **ELF-LDG** scheme in the single solitary wave example. L^2 projection for u^0 and P^- projection for v^0.

	J	N	u				v			
			$\|e^u\|_2$	C_2	$\|e^u\|_\infty$	C_∞	$\|e^v\|_2$	C_2	$\|e^v\|_\infty$	C_∞
P^0	200	100	7.4327E−02	*	4.5375E−02	*	1.7425E−01	*	1.0903E−01	*
	400	200	3.7242E−02	0.9970	2.2867E−02	0.9886	8.6718E−02	1.0068	5.4373E−02	1.0038
	800	400	1.8631E−02	0.9992	1.1442E−02	0.9989	4.3211E−02	1.0049	2.7141E−02	1.0024
	1600	800	9.3165E−03	0.9998	5.7186E−03	1.0007	2.1563E−02	1.0028	1.3566E−02	1.0005
P^1	200	100	4.9784E−03	*	6.6124E−03	*	8.2028E−03	*	1.0899E−02	*
	400	200	1.2552E−03	1.9877	1.7168E−03	1.9455	1.9907E−03	2.0429	2.7786E−03	1.9717
	800	400	3.1448E−04	1.9970	4.3265E−04	1.9885	4.8881E−04	2.0259	6.8834E−04	2.0132
	1600	800	7.8660E−05	1.9992	1.0837E−04	1.9972	1.2102E−04	2.0140	1.7078E−04	2.0110
P^2	200	100	2.4070E−04	*	2.9805E−04	*	3.7750E−04	*	5.1371E−04	*
	400	200	3.0407E−05	2.9847	3.7439E−05	2.9929	4.6416E−05	3.0238	6.2733E−05	3.0336
	800	400	3.8111E−06	2.9961	4.7477E−06	2.9792	5.7447E−06	3.0143	7.7004E−06	3.0262
	1600	800	4.7671E−07	2.9990	5.9413E−07	2.9984	7.1428E−07	3.0077	9.5320E−07	3.0141
P^3	200	100	1.1264E−05	*	1.5191E−05	*	1.7196E−05	*	2.6429E−05	*
	400	200	7.1039E−07	3.9870	1.0075E−06	3.9144	1.0666E−06	4.0110	1.7418E−06	3.9235
	800	400	4.4502E−08	3.9967	6.3418E−08	3.9897	6.6283E−08	4.0082	1.0824E−07	4.0082
	1600	800	2.7833E−09	3.9990	3.9577E−09	4.0021	4.1292E−09	4.0047	6.7114E−09	4.0115

5.3. Two solitary waves

In this example, we consider the case when two solitary waves move towards each other and collide. The initial conditions are given by

$$u(x,0) = \alpha_1 \operatorname{sech}^2\left(\sqrt{\frac{\alpha_1}{6}}\frac{x-x_1}{\beta_1}\right) + \alpha_2 \operatorname{sech}^2\left(\sqrt{\frac{\alpha_2}{6}}\frac{x-x_2}{\beta_2}\right),$$

and

$$\begin{aligned} u_t(x,0) =& 2\alpha_1\sqrt{\frac{\alpha_1}{6}} \operatorname{sech}^2\left(\sqrt{\frac{\alpha_1}{6}}\frac{x-x_1}{\beta_1}\right)\tanh\left(\sqrt{\frac{\alpha_1}{6}}\frac{x-x_1}{\beta_1}\right) \\ &+ 2\alpha_2\sqrt{\frac{\alpha_2}{6}} \operatorname{sech}^2\left(\sqrt{\frac{\alpha_2}{6}}\frac{x-x_2}{\beta_2}\right)\tanh\left(\sqrt{\frac{\alpha_2}{6}}\frac{x-x_2}{\beta_2}\right), \end{aligned}$$

where $\beta_1 = \sqrt{1+\frac{2\alpha_1}{3}}$, $\beta_2 = -\sqrt{1+\frac{2\alpha_2}{3}}$. The first solitary wave is initially located at $x = x_1$ with the amplitude α_1, and moves to the right with speed β_1. The other one is initially located at $x = x_2$ with the amplitude α_2, and moves to the left with speed $|\beta_2|$.

Table 5.3
Numerical error and convergence orders of u and w of **IMR-LDG** scheme in the single solitary wave example. L^2 projection for both u^0 and v^0.

	J	N	u				v			
			$\|e^u\|_2$	C_2	$\|e^u\|_\infty$	C_∞	$\|e^v\|_2$	C_2	$\|e^v\|_\infty$	C_∞
P^0	200	100	9.3230E−02	*	6.3510E−02	*	1.0936E−01	*	8.3849E−02	*
	400	200	4.7135E−02	0.9840	3.4033E−02	0.9001	5.4858E−02	0.9953	4.1718E−02	1.0071
	800	400	2.3658E−02	0.9945	1.7407E−02	0.9673	2.7428E−02	1.0001	2.0590E−02	1.0187
	1600	800	1.1846E−02	0.9979	8.8129E−03	0.9820	1.3708E−02	1.0006	1.0235E−02	1.0084
P^1	200	100	2.3293E−02	*	2.3803E−02	*	7.2707E−03	*	1.0541E−02	*
	400	200	1.2475E−02	0.9008	1.2039E−02	0.9835	1.7859E−03	2.0255	2.6515E−03	1.9911
	800	400	6.4414E−03	0.9537	6.0270E−03	0.9982	4.4008E−04	2.0208	6.5169E−04	2.0246
	1600	800	3.2697E−03	0.9782	3.0116E−03	1.0009	1.0909E−04	2.0123	1.6170E−04	2.0109
P^2	200	100	1.6790E−03	*	1.4164E−03	*	3.4423E−04	*	4.8748E−04	*
	400	200	4.0062E−04	2.0673	3.2527E−04	2.1225	4.2580E−05	3.0151	5.9007E−05	3.0464
	800	400	9.8736E−05	2.0206	8.0494E−05	2.0147	5.2808E−06	3.0113	7.3467E−06	3.0057
	1600	800	2.4593E−05	2.0053	2.0076E−05	2.0034	6.5712E−07	3.0065	9.0977E−07	3.0135
P^3	200	100	1.4156E−04	*	2.0907E−04	*	1.6208E−05	*	2.6065E−05	*
	400	200	1.8608E−05	2.9274	2.8674E−05	2.8662	1.0092E−06	4.0054	1.6896E−06	3.9474
	800	400	2.3676E−06	2.9745	3.6735E−06	2.9645	6.2806E−08	4.0062	1.0407E−07	4.0211
	1600	800	2.9792E−07	2.9904	4.6237E−07	2.9900	3.9149E−09	4.0039	6.4817E−09	4.0050

Table 5.4
Numerical error and convergence orders of u and w of **IMR-LDG** scheme in the single solitary wave example. L^2 projection for u^0 and P^- projection for v^0.

	J	N	u				v			
			$\|e^u\|_2$	C_2	$\|e^u\|_\infty$	C_∞	$\|e^v\|_2$	C_2	$\|e^v\|_\infty$	C_∞
P^0	200	100	7.4327E−02	*	4.5375E−02	*	1.7425E−01	*	1.0903E−01	*
	400	200	3.7242E−02	0.9970	2.2867E−02	0.9886	8.6718E−02	1.0068	5.4373E−02	1.0038
	800	400	1.8631E−02	0.9992	1.1442E−02	0.9989	4.3211E−02	1.0049	2.7141E−02	1.0024
	1600	800	9.3165E−03	0.9998	5.7186E−03	1.0007	2.1563E−02	1.0028	1.3566E−02	1.0005
P^1	200	100	4.9784E−03	*	6.6124E−03	*	8.2028E−03	*	1.0899E−02	*
	400	200	1.2552E−03	1.9877	1.7168E−03	1.9455	1.9907E−03	2.0429	2.7786E−03	1.9717
	800	400	3.1448E−04	1.9970	4.3265E−04	1.9885	4.8881E−04	2.0259	6.8834E−04	2.0132
	1600	800	7.8660E−05	1.9992	1.0837E−04	1.9972	1.2102E−04	2.0140	1.7078E−04	2.0110
P^2	200	100	2.4070E−04	*	2.9804E−04	*	3.7750E−04	*	5.1370E−04	*
	400	200	3.0407E−05	2.9847	3.7438E−05	2.9929	4.6416E−05	3.0238	6.2733E−05	3.0336
	800	400	3.8111E−06	2.9961	4.7476E−06	2.9792	5.7447E−06	3.0143	7.7003E−06	3.0262
	1600	800	4.7671E−07	2.9990	5.9412E−07	2.9984	7.1428E−07	3.0077	9.5320E−07	3.0141
P^3	200	100	1.1264E−05	*	1.5200E−05	*	1.7196E−05	*	2.6432E−05	*
	400	200	7.1038E−07	3.9870	1.0086E−06	3.9137	1.0666E−06	4.0110	1.7421E−06	3.9233
	800	400	4.4500E−08	3.9967	6.3554E−08	3.9882	6.6283E−08	4.0082	1.0829E−07	4.0079
	1600	800	2.7828E−09	3.9992	3.9779E−09	3.9979	4.1292E−09	4.0047	6.7173E−09	4.0108

In this example, we have chosen P^3 as the space of basis functions and set

$$t \in [0, 80], \quad x \in [-60, 100], \quad h = 1, \quad \tau = 0.1, \quad x_1 = -20, \quad x_2 = 60.$$

The numerical results of both ELF-LDG and IMR-LDG methods with the same set of parameters are similar for this test, and hence we only present the results of ELF-LDG methods with P^3 basis to save space.

We are interested in the interaction between two waves of various parameters. As shown in [18,20,32,7,6,22], secondary small solitary waves may appear numerically after the collision of two waves for a certain range of wave amplitudes, and when such phenomenon appears, the interaction is named inelastic. Experience from these literatures suggested that, when $\max(\alpha_1, \alpha_2) \leq 0.4$, there are no visible secondary soliton waves in the interior of the region between the principal solitons, and the collision is elastic. And when $\max(\alpha_1, \alpha_2) \geq 0.4$, the interaction is inelastic and noticeable secondary soliton waves will appear after the collision.

We first study the collision of two solitary waves with the same amplitude. Fig. 5.4 presents the propagation of two solitary waves with $\alpha_1 = \alpha_2 = 0.2$. We can observe that both solitary waves preserve their original amplitudes and no displacement is observed. There is no visible secondary solitary waves and this collision is elastic. Fig. 5.5 presents the propagation of two solitary waves with $\alpha_1 = \alpha_2 = 1.2$, where some secondary solitary waves can be observed after the collision. Both results are in good agreement with those reported in the literatures.

Next, we present some results when solitary waves of different amplitudes interact. Fig. 5.6 includes the interaction of two solitary waves with $\alpha_1 = 0.1, \alpha_2 = 0.3$, and we can observe that the collision is elastic with no secondary solitary waves. Figs. 5.7, 5.8 and 5.9 present the interaction of two solitary waves with $\alpha_1 = 0.1, \alpha_2 = 0.5$, with $\alpha_1 = 0.1, \alpha_2 = 1.2$

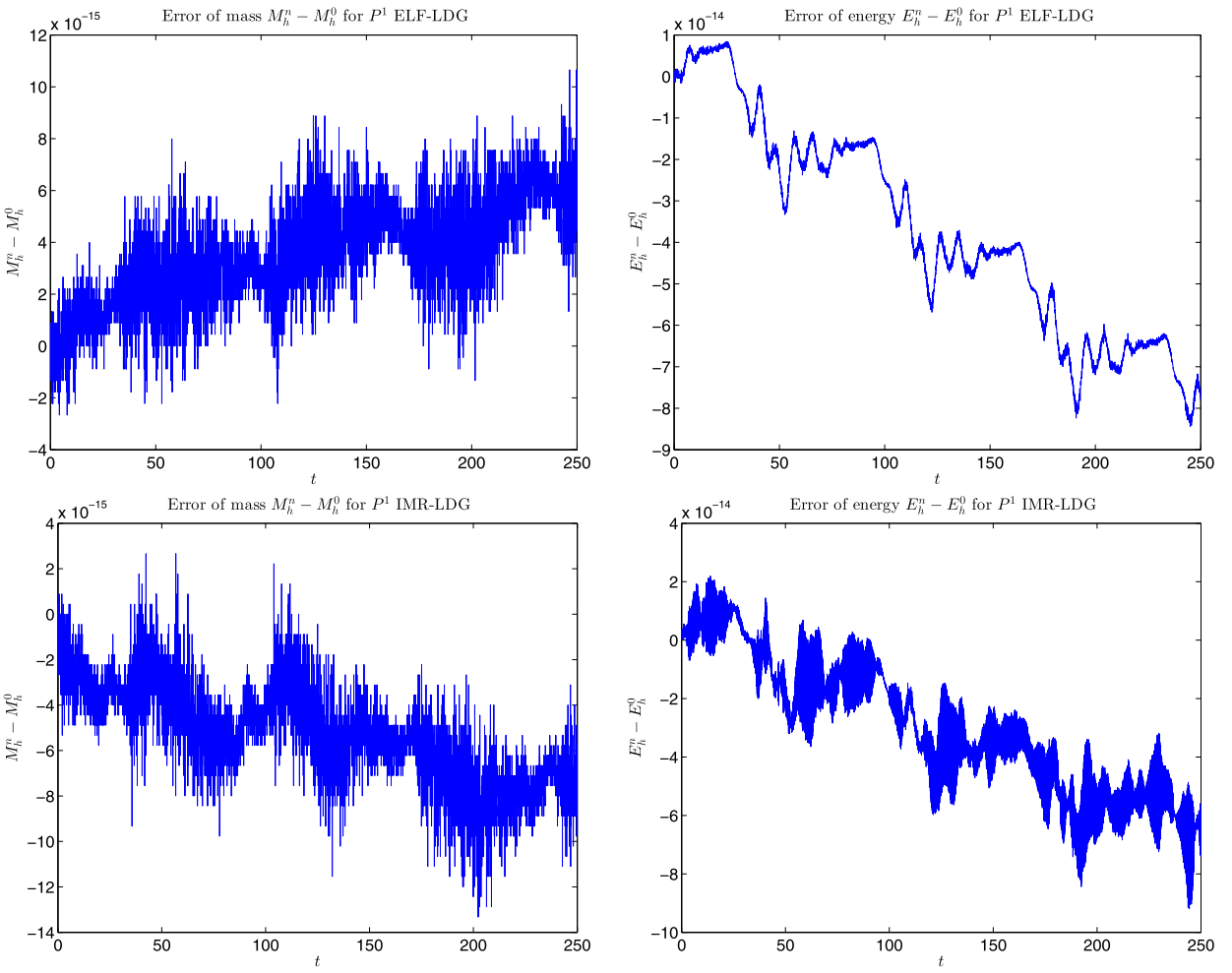

Fig. 5.1. Errors of mass (left) and energy (right) in the single solitary wave example. Top: results of the ELF-LDG scheme with P^1 basis; Bottom: results of the IMR-LDG scheme with P^1 basis.

and with $\alpha_1 = 1.2, \alpha_2 = 1.5$, respectively. Secondary solitary waves appear after the interaction (although some of them are relatively small) and the collisions are inelastic.

Following the definition in [32,6], the inelasticity coefficient K_α is defined as follows

$$K_\alpha = \frac{\alpha_m}{\max(\alpha_1, \alpha_2)},$$

where α_m is the maximum joint amplitude at the collision time. Table 5.5 shows the maximum joint amplitudes of our methods at the collision time, compared with those in [30] and [7]. For comparison, the parameters to generate these data in Table 5.5 are set as

$$x_1 = -20, \quad x_r = 30, \quad x_l = -80, \quad x_r = 120,$$

which is the same as those in [30,7]. We notice that α_m is always smaller than the average of two wave amplitudes $(\alpha_1 + \alpha_2)/2$ and the coefficient K_α will decrease when α_2 increase while keeping α_1 invariant. The numerical results of our methods are similar as those in the literatures, but with a much larger spatial and temporal step size.

5.4. Wave break-up

In this example, we consider the wave break-up phenomenon. The initial conditions are given in the following:

$$u(x,0) = 2\alpha \operatorname{sech}^2\left(\sqrt{\frac{\alpha}{6}}\frac{x - x_0}{\beta}\right), \quad u_t(x,0) = 0,$$

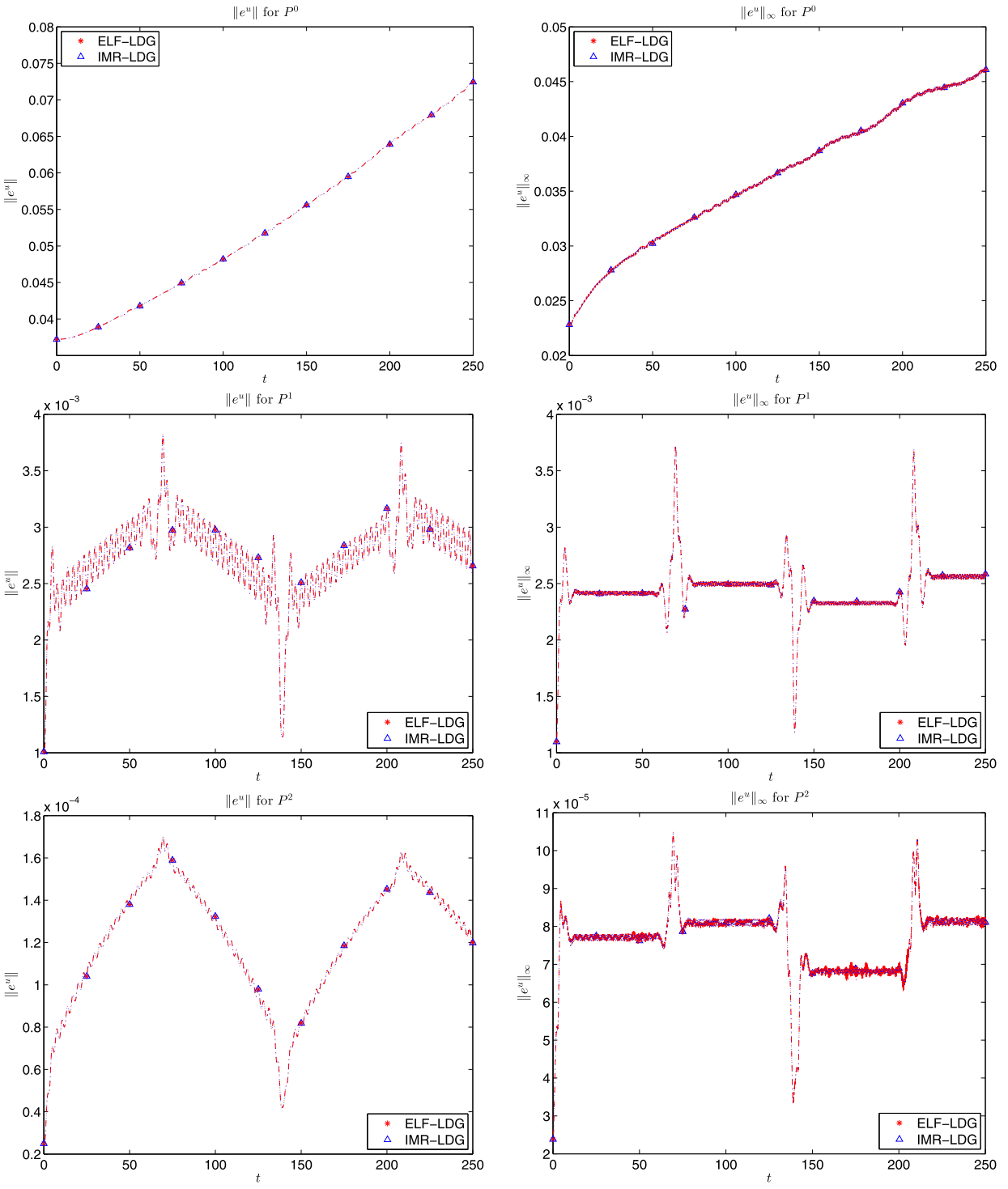

Fig. 5.2. Time history of the numerical errors of u_h in L^2 norm (left) and L^∞ norm (right) of ELF-LDG and IMR-LDG methods in the single solitary wave example with different polynomial basis P^0 (top), P^1 (middle) and P^2 (bottom). (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

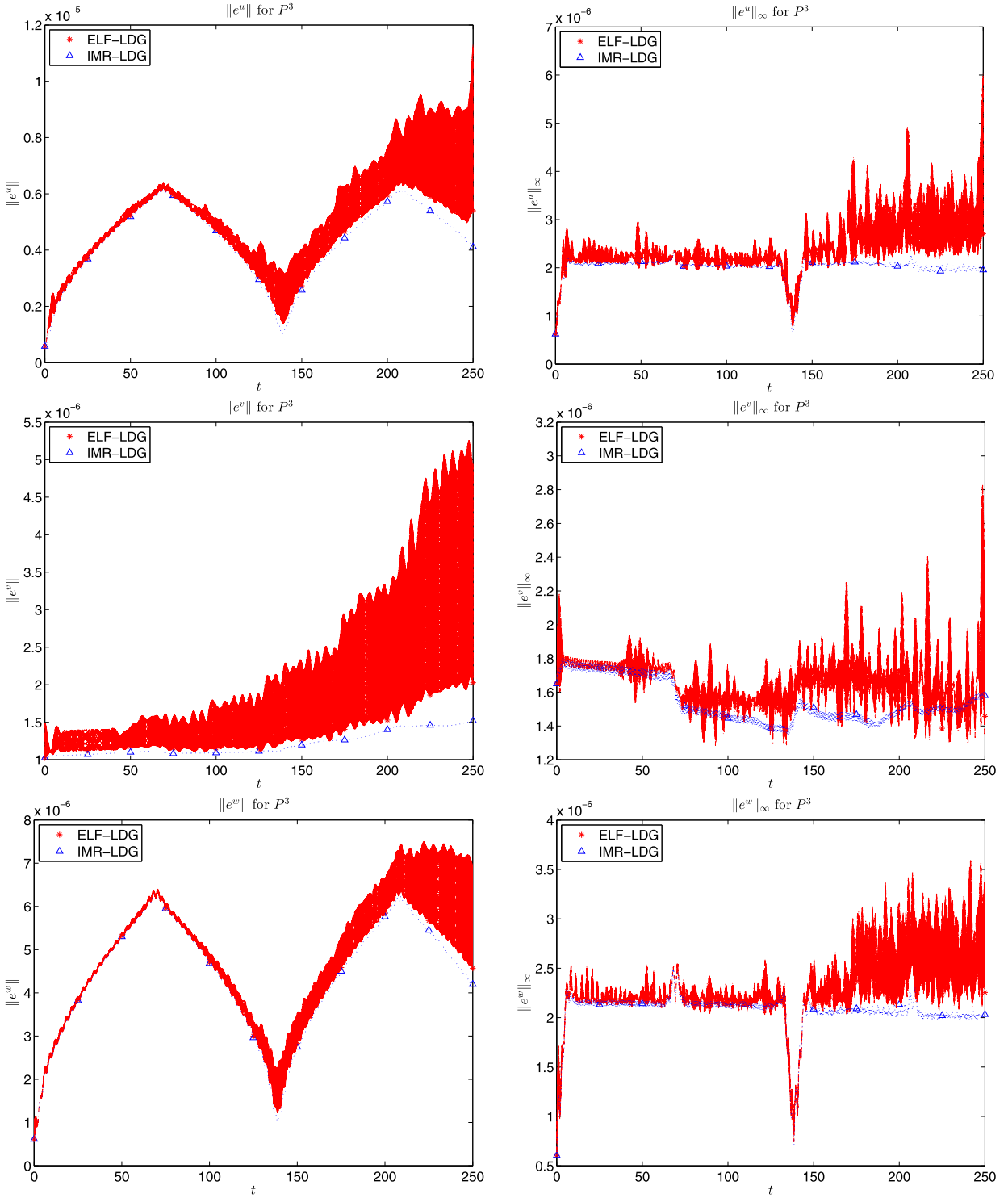

Fig. 5.3. Time history of the numerical errors in L^2 norm (left) and L^∞ norm (right) of ELF-LDG and IMR-LDG methods in the single solitary wave example with P^3 polynomial basis. Top: numerical error of u_h; Middle: numerical error of v_h; Bottom: numerical error of w_h.

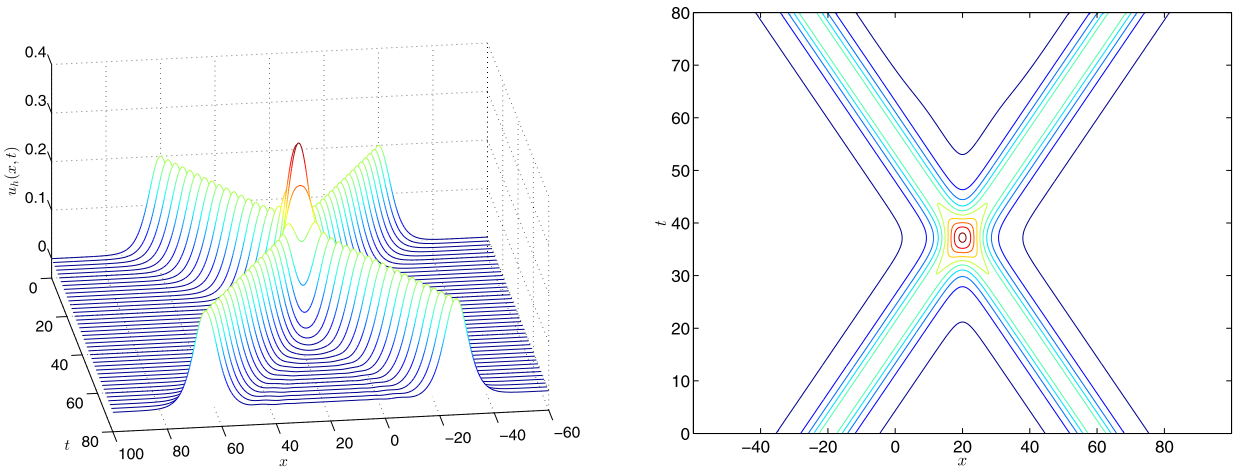

Fig. 5.4. The propagation of two solitary waves (left) and the contour line (right) with $\alpha_1 = 0.2$, $\alpha_2 = 0.2$.

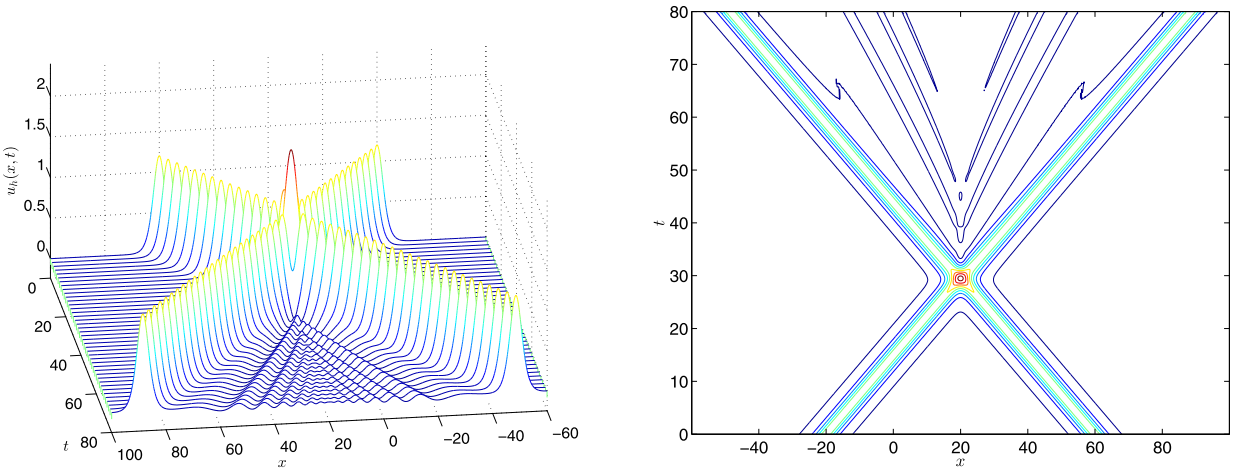

Fig. 5.5. The propagation of two solitary waves (left) and the contour line (right) with $\alpha_1 = 1.2$, $\alpha_2 = 1.2$.

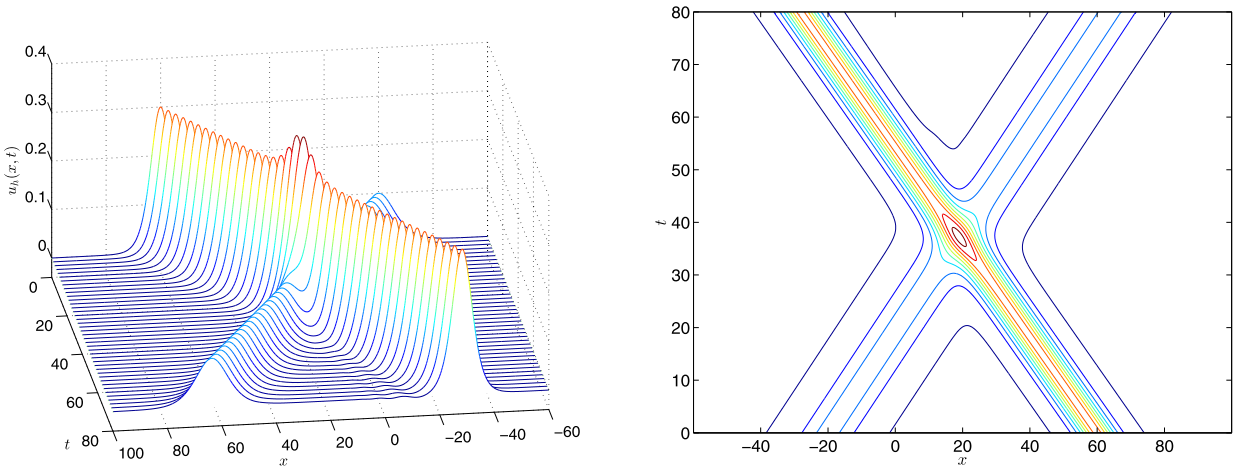

Fig. 5.6. The propagation of two solitary waves (left) and the contour line (right) with $\alpha_1 = 0.1$, $\alpha_2 = 0.3$.

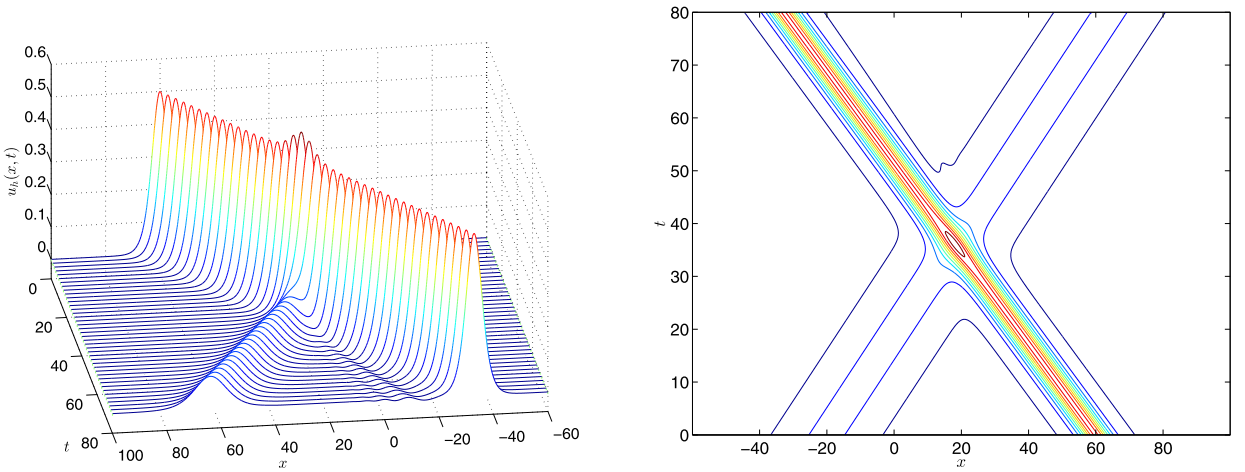

Fig. 5.7. The propagation of two solitary waves (left) and the contour line (right) with $\alpha_1 = 0.1$, $\alpha_2 = 0.5$.

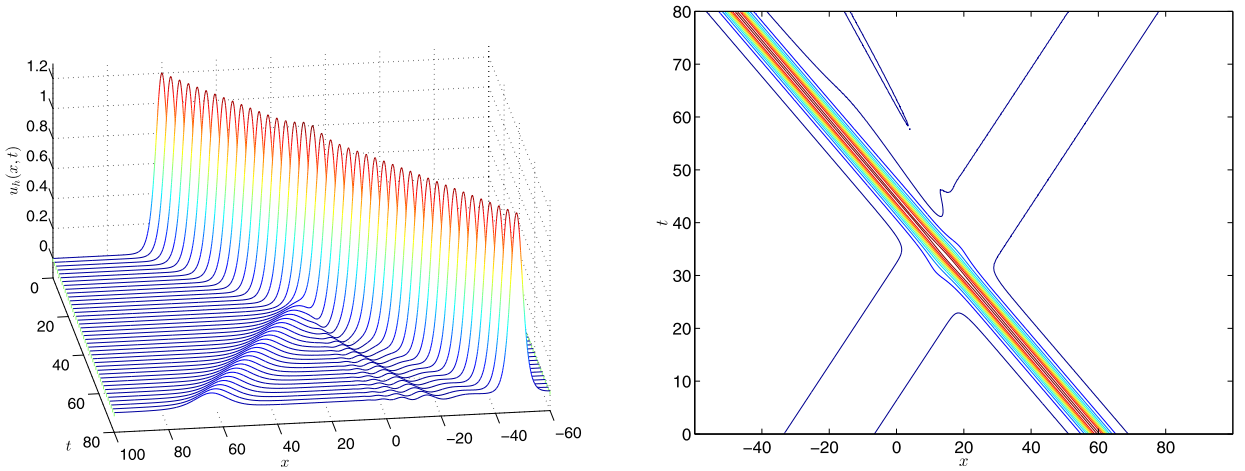

Fig. 5.8. The propagation of two solitary waves (left) and the contour line (right) with $\alpha_1 = 0.1$, $\alpha_2 = 1.2$.

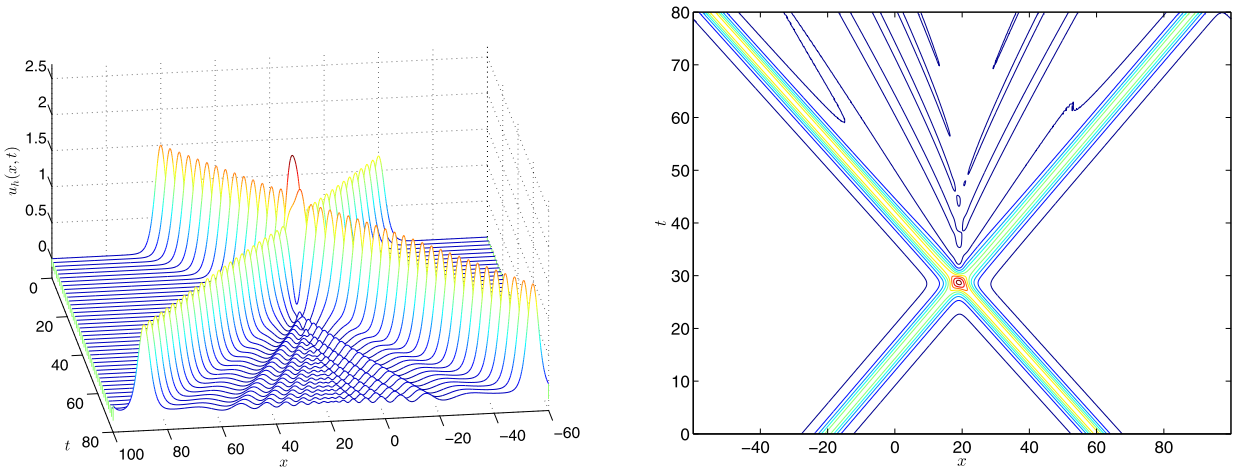

Fig. 5.9. The propagation of two solitary waves (left) and the contour line (right) with $\alpha_1 = 1.2$, $\alpha_2 = 1.5$.

Table 5.5
The maximum joint amplitudes at the collision of ELF-LDG methods with P^3 basis and various α_1, α_2. Parameters: $h = 0.5$, $\tau = 0.01$, $x_1 = -20$, $x_2 = 30$, $x_l = -80$, $x_r = 120$ and $T = 30$.

α_1	α_2	Our α_m	α_m in [30]	α_m in [7]	Our K_α
		$h = 0.5$, $\tau = 0.01$	$h = 0.1$, $\tau = 0.001$	$h = 0.1$, $\tau = 0.001$	
0.4	0.4	0.72243	0.72243	0.71642	1.80607
2	2	3.31617	3.31620	3.25880	1.65809
0.4	0.5	0.80846	0.80843	0.79614	1.61692
0.4	1.5	1.71761	1.71760	1.62530	1.14507
0.4	2.5	2.66881	2.66880	2.56680	1.06752

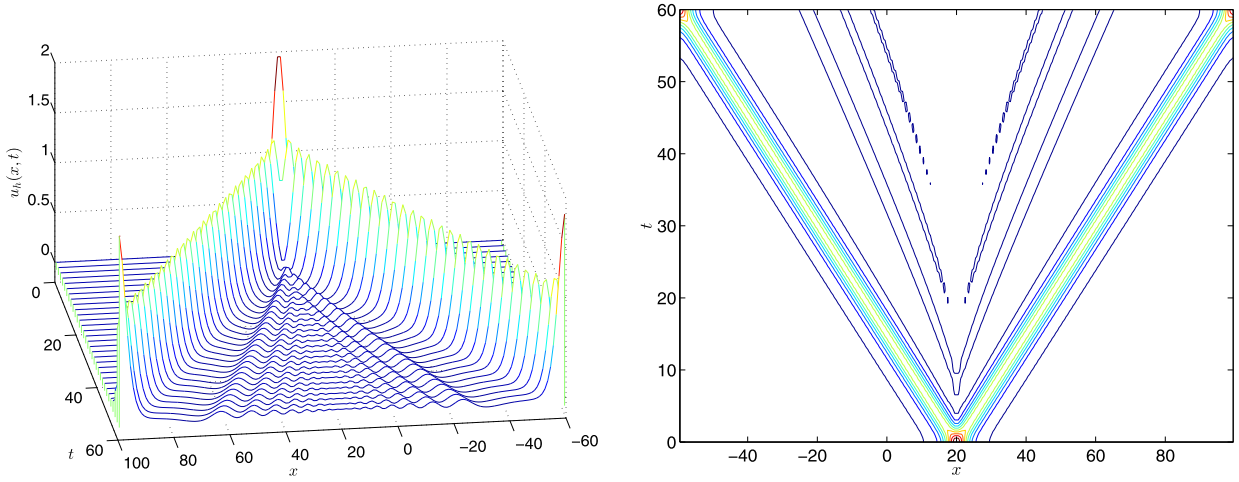

Fig. 5.10. A solitary wave breaks up (left) and the contour line (right) test with $\alpha = 1$.

where $\beta = \sqrt{1 + \frac{2\alpha}{3}}$. In the numerical experiment, we have chosen P^3 as the space of basis functions, and set

$$x_0 = 20, \quad x \in [-60, 100], \quad t \in [0, 60], \quad h = 1, \quad \tau = 0.1.$$

Again, the numerical results are based on the ELF-LDG methods.

Fig. 5.10 shows an initial solitary wave with amplitude $\alpha = 1$ breaking into two smaller diverging solitary waves. Note that the two solitary waves are symmetric and move in opposite directions with oscillating tails between them. In Fig. 5.11, we observe an initial wave with amplitude $\alpha = 0.1$ breaks into two smaller solitary waves, symmetric and moving in opposite directions, while no visible oscillating tails between them. The wave break-up can be viewed as two same amplitude waves collision by setting the initial time as the time of two waves collision. Therefore the observation of with or without oscillating tails in Figs. 5.10 and 5.11 is consistent with the conclusions in Section 5.3.

5.5. Finite time blow-up

In this example, we simulate the finite time blow-up of the numerical solution, which was discussed in [27,37,38]. The equation is considered on $x \in [0, 1]$ with the initial conditions

$$u(x, 0) = -3\sin(\pi x), \quad u_t(x, 0) = -\sin(\pi x).$$

Under this setup, it is known from the discussion in [37] that there exists a finite time T^0 such that a unique local solution $u \in C^2([0, T^0])$; $H^2(0, 1) \bigcap H_0^1(0, 1)$ exists and satisfies

$$\|u(\cdot, t)\|_{L_2(1,2)} \to \infty, \quad \text{as} \quad t \to T^0,$$

and

$$I(t) = \int_0^1 u(x, t)\sin(\pi x)dx \to -\infty, \quad \text{as} \quad t \to T^0.$$

We use ELF-LDG scheme with P^3 basis and $h = 0.005$, $\tau = 0.001$ to simulate this example. The numerical solution u_h at various times (before the blow-up) is shown in Fig. 5.12, where we can observe that the solution tends to blow up at

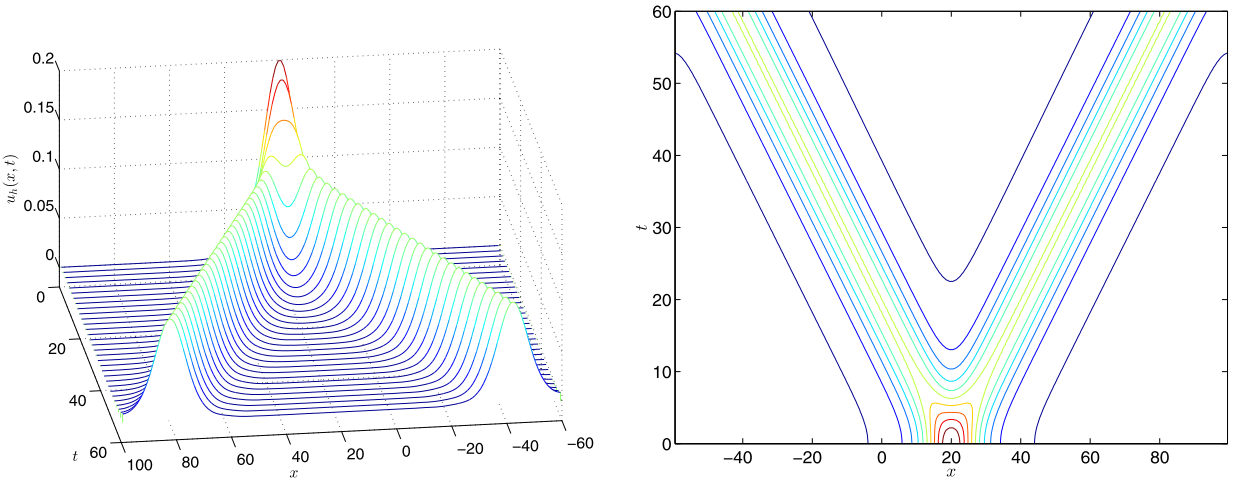

Fig. 5.11. A solitary wave breaks up (left) and the contour line (right) test with $\alpha = 0.1$.

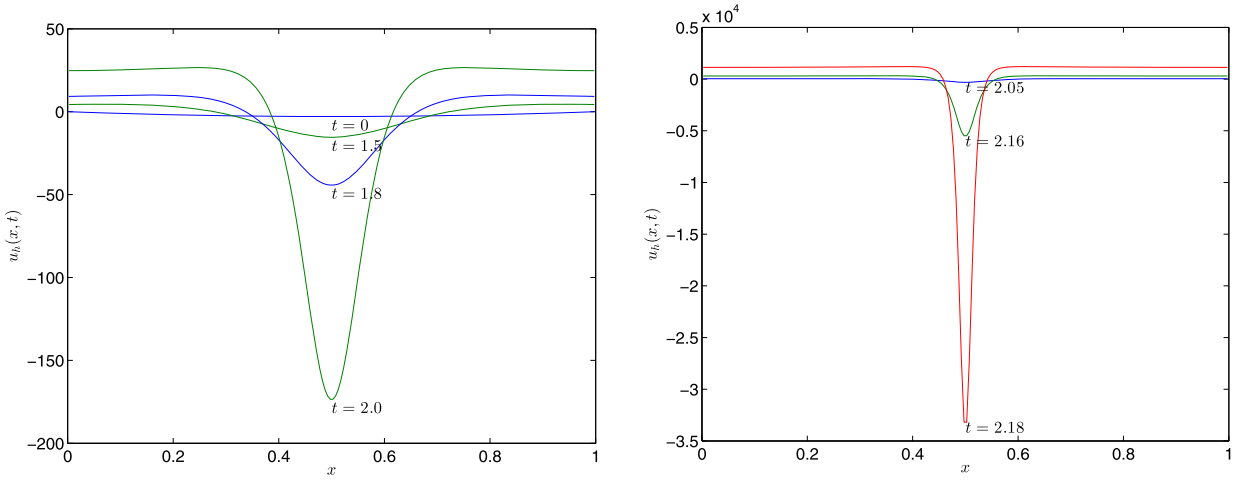

Fig. 5.12. The numerical solution u_h of the blow-up test at different times.

$x = 0.5$ and reaches the value of -3×10^4 when $t = 2.18$. The time history of the quantity I(t) and the 3D plot of the numerical solution $u_h(x,t)$ are shown in Fig. 5.13.

6. Concluding remarks

In this paper, we have developed and analyzed energy conserving LDG methods for solving improved Boussinesq equation. We proved that the proposed semi-discrete scheme has the desired property of preserving mass and energy exactly. An optimal error estimate is provided for the semi-discrete methods if the flux and initial projection are chosen carefully, and numerical tests confirm that optimal convergence rate can be obtained. Both explicit and implicit temporal discretizations were presented to obtain two kinds of fully discrete methods, which are shown to conserve the discrete mass and energy exactly. The proposed LDG methods have the advantage of high order accuracy and easily extended to arbitrary order, mass and energy conservation, and optimal error estimate. They also inherit other advantages of the DG methods, including the adaptivity and excellent parallel efficiency. Numerous examples of wave propagation showed the proposed LDG scheme has the capacity to simulate one traveling solitary wave, two solitary waves interaction, single wave break-up and blow-up phenomenon well. The proposed methods and the analysis to investigate the energy conservation and error estimate would be very useful for other types of wave equations involving nonlinear high order and mixed derivatives terms. Study on the error estimate of the fully discrete methods, and generalization to other wave equations will be the subject of our future investigation.

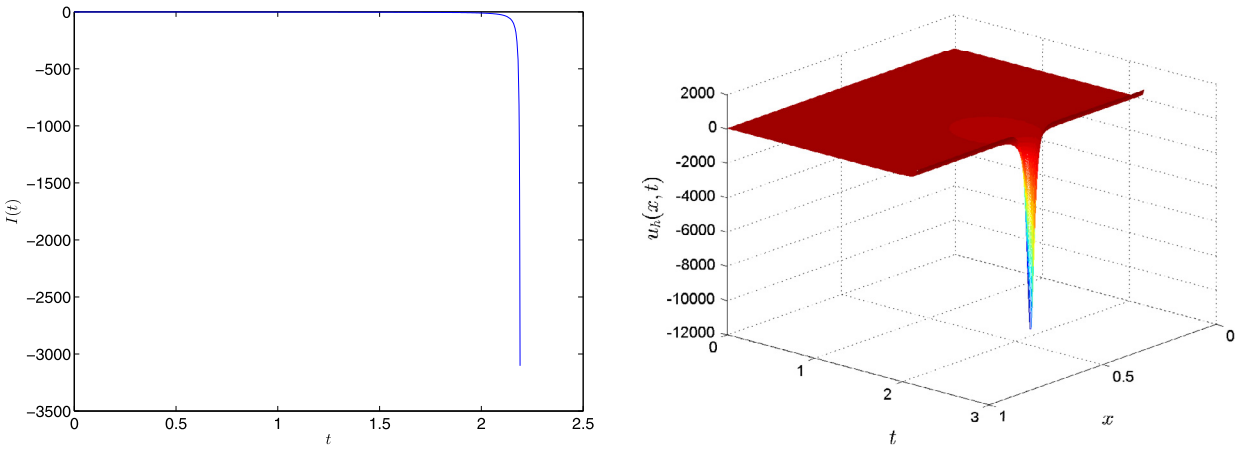

Fig. 5.13. Numerical results of the blow-up test. Left: time history of the quantity $I(t)$; Right: 3D plot of the time history of the numerical solution u_h.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] D.N. Arnold, An interior penalty finite element method with discontinuous elements, SIAM J. Numer. Anal. 19 (1982) 742–760.
[2] I.L. Bogolubsky, Modified equation of a nonlinear string and inelastic interaction of solitons, J. Exp. Theor. Phys. 24 (1976) 184–186.
[3] I.L. Bogolubsky, Some examples of inelastic soliton interaction, Comput. Phys. Commun. 13 (1977) 149–155.
[4] J.L. Bona, H. Chen, O.A. Karakashian, Y. Xing, Conservative discontinuous Galerkin methods for the Generalized Korteweg-de Vries equation, Math. Comput. 82 (2013) 1401–1432.
[5] J. Boussinesq, Théorie des ondes et des remous qui se propagent le long d'un canal rectangulaire horizontal, en communiquant au liquide contenu dans ce canal des vitesses sensiblement pareilles de la surface au fond, J. Math. Pures Appl., Deux. Sér. 17 (1872) 55–108.
[6] A.G. Bratsos, A second order numerical scheme for the improved Boussinesq equation, Phys. Lett. A 370 (2) (2007) 145–147.
[7] A.G. Bratsos, A predictor-corrector scheme for the improved Boussinesq equation, Chaos Solitons Fractals 40 (2009) 2083–2094.
[8] P. Ciarlet, The Finite Element Method for Elliptic Problem, North-Holland, 1975.
[9] Y. Cheng, C.-W. Shu, A discontinuous Galerkin finite element method for time dependent partial differential equations with higher order derivatives, Math. Comput. 77 (2008) 699–730.
[10] Y. Cheng, C.-S. Chou, F. Li, Y. Xing, L2 stable discontinuous Galerkin methods for one-dimensional two-way wave equations, Math. Comput. 86 (2017) 121–155.
[11] C.-S. Chou, C.-W. Shu, Y. Xing, Optimal energy conserving local discontinuous Galerkin methods for second-order wave equation in heterogeneous media, J. Comput. Phys. 272 (2014) 88–107.
[12] B. Cockburn, J. Gopalakrishnan, R. Lazarov, Unified hybridization of discontinuous Galerkin, mixed, and continuous Galerkin methods for second order elliptic problems, SIAM J. Numer. Anal. 42 (2009) 1319–1365.
[13] B. Cockburn, S.-Y. Hou, C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: The multidimensional case, J. Comput. Phys. 141 (1998) 199–224.
[14] B. Cockburn, S.-Y. Lin, C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one dimensional systems, J. Comput. Phys. 84 (1989) 90–113.
[15] B. Cockburn, C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for scalar conservation laws II: general framework, Math. Comput. 52 (1989) 411–435.
[16] B. Cockburn, C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for scalar conservation laws V: multidimensional systems, Math. Comput. 52 (1989) 411–435.
[17] B. Cockburn, C.-W. Shu, The local discontinuous Galerkin method for time-dependent convection-diffusion systems, SIAM J. Numer. Anal. 35 (1998) 2440–2463.
[18] H. El-Zoheiry, Numerical study of the improved Boussinesq equation, Chaos Solitons Fractals 14 (2002) 377–384.
[19] L. Guo, Y. Xu, Energy conserving local discontinuous Galerkin methods for the nonlinear Schrodinger equation with wave operator, J. Sci. Comput. 65 (2015) 622–647.
[20] L. Iskandar, P.C. Jain, Numerical solutions of the improved Boussinesq equation, Proc. Indian Acad. Sci. Math. Sci. 89 (1980) 171–181.
[21] O. Karakashian, Y. Xing, A posteriori error estimates for conservative local discontinuous Galerkin methods for the Generalized Korteweg-de Vries equation, Commun. Comput. Phys. 20 (2016) 250–278.
[22] Q. Lin, Y. Wu, R. Loxton, S. Lai, Linear B-spline finite element method for the improved Boussinesq equation, J. Comput. Appl. Math. 224 (2) (2009) 658–667.
[23] X.H. Li, C.-W. Shu, Y. Yang, Local discontinuous Galerkin method for the Keller-Segel chemotaxis model, J. Sci. Comput. 73 (2017) 943–967.
[24] X. Li, Y. Xing, C.-S. Chou, Optimal energy conserving and energy dissipative local discontinuous Galerkin methods for the Benjamin-Bona-Mahony equation, submitted.
[25] X. Liang, A.Q.M. Khaliq, Y. Xing, Fourth order exponential time differencing method with local discontinuous Galerkin approximation for coupled nonlinear Schrodinger equations, Commun. Comput. Phys. 17 (2015) 510–541.
[26] H. Liu, Y. Xing, An invariant preserving discontinuous Galerkin method for the Camassa-Holm equation, SIAM J. Sci. Comput. 38 (2016) A1919–A1934.

[27] Q. Lin, Y.H. Wu, R. Loxton, S. Lai, Linear B-spline finite element method for the improved Boussinesq equation, J. Comput. Appl. Math. 224 (2009) 658–667.
[28] V.S. Manoranjan, A.R. Mitchell, J. Li Morris, Numerical solutions of the good Boussinesq equation, SIAM J. Sci. Stat. Comput. 5 (1984) 946–957.
[29] W.H. Reed, T.R. Hill, Triangular Mesh Methods for the Neutron Transport Equation, Tech. Report LA-UR-73-479, Los Alamos Scientific Laboratory, Los Alamos, NM, 1973.
[30] A. Shokri, M. Dehghan, A not-a-knot meshless method using radial basis functions and predictor-corrector scheme to the numerical solution of improved Boussinesq equation, Comput. Phys. Commun. 181 (2010) 1990–2000.
[31] H.J. Wang, C.-W. Shu, Q. Zhang, Stability and error estimates of local discontinuous Galerkin method with implicit-explicit time-marching for advection-diffusion problems, SIAM J. Numer. Anal. 53 (2015) 206–227.
[32] Q. Wang, Z. Zhang, X. Zhang, Q. Zhu, Energy-preserving finite volume element method for the improved Boussinesq equation, J. Comput. Phys. 270 (2014) 58–69.
[33] Y. Xing, C.-S. Chou, C.-W. Shu, Energy conserving local discontinuous Galerkin methods for wave propagation problems, Inverse Probl. Imaging 7 (2013) 967–986.
[34] Y. Xu, C.-W. Shu, Error estimates of the semi-discrete local discontinuous Galerkin method for nonlinear convection diffusion and KdV equations, Comput. Methods Appl. Mech. Eng. 196 (2007) 3805–3822.
[35] Y. Xu, C.-W. Shu, Local discontinuous Galerkin methods for high-order time-dependent partial differential equations, Commun. Comput. Phys. 7 (2010) 1–46.
[36] J. Yan, Z. Zhang, T. Zhao, D. Liang, High-order energy-preserving schemes for the improved Boussinesq equation, Numer. Methods Partial Differ. Equ. 34 (2018) 1145–1165.
[37] Z. Yang, Existence and non-existence of global solutions to a generalized modification of the improved Boussinesq equation, Math. Methods Appl. Sci. 21 (1998) 1467–1477.
[38] Z. Yang, X. Wang, Blow up of solutions for improved Boussinesq-type equation, J. Math. Anal. Appl. 278 (2003) 335–353.
[39] Z. Zlatev, I. Dimov, I. Farago, A. Havasi, Richardson Extrapolation: Practical Aspects and Applications, De Gruyter Series in Applied and Numerical Mathematics, vol. 2, De Gruyter, 2017.