

ERROR SELF-CANCELING OF A DIFFERENCE SCHEME MAINTAINING TWO CONSERVATION LAWS FOR LINEAR ADVECTION EQUATION

CUI YANFEN AND MAO DE-KANG

ABSTRACT. In recent years, Mao and his co-workers developed a new type of difference schemes for evolution partial differential equations. The core of the new schemes is to simulate, in addition to the original unknowns of the equations, some quantities that are nonlinear functions of the unknowns; therefore, they maintain additional nonlinear discrete structures of the equations. The schemes show a super-convergence property, and their numerical solutions are far better than that of traditional difference schemes at both accuracy and long-time behavior.

In this paper, to understand the super-convergence properties of the schemes, we carry out a truncation error investigation on the scheme maintaining two conservation laws for the linear advection equation. This scheme is the simplest one of this type. Our investigation reveals that the numerical errors of the scheme produced in different time steps are accumulated in a nonlinear fashion, in which they cancel each other. As to our knowledge, such an error self-canceling feature has not been seen in other numerical methods, and it is this feature that brings the super-convergence property of the scheme.

1. INTRODUCTION

In recent years, Mao and his co-workers developed a new type of difference schemes for numerical simulations of evolutionary partial differential equations; see [13], [14], [15], [20] and [21]. The core of the new schemes can be conceptually described for a scalar PDE as follows:

For a scalar evolutionary PDE

$$(1.1) \quad u_t = \mathcal{L}(u),$$

where u is the unknown and \mathcal{L} is the spatial differential operator, we may introduce mathematical quantities $U_1(u), \dots, U_k(u)$, which are nonlinear functions of u . These redundant quantities satisfy related PDEs,

$$(1.2) \quad (U_l(u))_t = \mathcal{L}_l(u, U), \quad l = 1, \dots, k,$$

Received by the editor March 14, 2009 and, in revised form, January 22, 2011.

2010 *Mathematics Subject Classification.* Primary 65M06, 65M15.

Key words and phrases. Scheme maintaining two conservation laws, truncation error, error self-canceling.

This research was supported by China National Science Foundation Grant No.10971132 and Shanghai Pu Jiang Program [2006] 118, and also by Shanghai Leading Academic Discipline project (J50101).

©2011 American Mathematical Society
Reverts to public domain 28 years from publication

which are derived from (1.1), where \mathcal{L}_l 's are the corresponding spatial differential operators. In our scheme for (1.1), the numerical solution in each grid cell at time t_n has several components, $u_j^n, U_{j,1}^n, \dots, U_{j,k}^n$, the first one of which is the approximation to the original unknown u and the following ones of which are the approximations to the quantities $U_l(u)$. The approximations can be in any sense, and at present we take them as the cell-average approximations.

The numerical solution in each grid cell can then be reconstructed as a polynomial,

$$(1.3) \quad \begin{aligned} R(x; u^n, U_1^n, \dots, U_k^n) = & s_{j,0}^n + s_{j,1}^n(x - x_j) + \dots \\ & + s_{j,k}^n(x - x_j)^k, \quad x \in (x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}), \end{aligned}$$

and the coefficients of the polynomials $s_{j,l}^n$ can be determined by the requirements

$$(1.4) \quad \frac{1}{h} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} R(x; u^n, U_1^n, \dots, U_k^n) dx = u_j^n$$

and

$$(1.5) \quad \frac{1}{h} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} U_l(R(x; u^n, U_1^n, \dots, U_k^n)) dx = U_{j,l}^n, \quad j = 1, \dots, k.$$

The evolution of the numerical solution is carried by a proper temporal discretization of (1.1), in which all u^n and all U_l^n are computed. The schemes computing all U_l^n are discretizations of the PDEs in (1.2); however, they are consistent with the scheme computing u^n in a certain sense.

We should note the similarity of our methods with the discontinuous Galerkin (DG) methods [4]. Both methods consider numerical solutions of the form (1.3) with several components in a grid cell. However, in DG methods, the coefficients of the orthogonal polynomial expansion of the numerical solution are taken as the components, which are linearly related to the numerical solution (1.3); while in our methods, the approximations to $U_l(u)$, $U_{j,l}^n$, are taken as the components, which are nonlinearly related to (1.3). Moreover, as in ordinary Galerkin methods, equation (1.1) in DG methods is changed into weak form in each cell by multiplying it with test functions and then integrating it over the cell. Such operations can actually be viewed as linear functionals acting on (1.1). While in our methods, equation (1.1) is multiplied with $\frac{\partial U_l}{\partial u}$ and integrated over the cell to be changed into the integral forms of the redundant equations in (1.2); see (2.14) and (2.16) in the following section. Such operations can actually be viewed as nonlinear functionals acting on (1.1). In this sense, the methods can be viewed as *nonlinear* extensions of the DG methods. In practical applications, the quantities $U_l(u)$ are usually chosen to have physical meanings, such as energy or entropy.

This type of scheme shows an extremely outstanding quality in both accuracy and long-time behavior of numerical solutions. In the development of the methods, Mao and his co-workers started with the linear advection equation (2.1). This equation possesses infinitely many conservation laws, and they chose u^2 and u^3 as the redundant quantities to construct schemes maintaining two and three conservation laws; see the references cited at the beginning of this section. The numerical experiments showed that the scheme maintaining three conservation laws is superior to the scheme maintaining two conservation laws. Cui and Mao constructed a scheme for the KdV equation that maintained the conservation laws of both u and u^2

(see [5]) and the numerical experiments showed that the scheme had a very good quality, comparable to that of symplectic schemes (see [1] and [2]) in long-time numerical integration. Li, Wang and Mao [16], and Chen and Mao [3], chose the entropy as the redundant quantity to construct a modified Godunov scheme for the 1D Euler system. They used the so-called Ultra-bee technique (see [19] and [6]) to suppress the numerical oscillations. The modified Godunov scheme greatly improved the resolution of the second characteristic waves in numerical solutions.

Despite the success of the methods in improving accuracy and long-time behavior of numerical solutions, a theoretical understanding of them is still lacking. The purpose of this paper is to give a theoretical explanation of the good quality of the methods in accuracy and long-time behavior. For the simplicity of discussion, we will study the scheme maintaining u and u^2 conserved for the linear advection equation. This scheme is the simplest one of this type. Since the scheme is nonlinear as mentioned before, linear analysis methods, such as the phase error analysis (see [17]) do not apply. We thus give a truncation error study of the scheme to investigate the accumulation of the numerical errors.

The investigation reveals that the numerical errors in different time steps of the scheme are accumulated in a nonlinear fashion, in which they cancel each other. It is this error self-canceling feature that makes the scheme a super-convergence property: the global error of the scheme is of an order higher than its truncation error; see §5. We have not seen this error self-canceling feature in other numerical methods for PDE.

The paper is organized as follows: §1 is the introduction. In §2 we describe the scheme maintaining two conservation laws for the linear advection equation. In §3 we present two numerical experiments to illustrate the super-convergence property of the scheme and its good quality in long-time simulation. In §4 we prove two theorems concerning the evolution of the global error. In §5 we investigate the error self-canceling feature of the scheme and show that this feature brings the super-convergence property to the scheme. Finally, §6 is the conclusion.

2. SCHEME MAINTAINING TWO CONSERVATION LAWS FOR LINEAR ADVECTION EQUATION

In this section, we describe the scheme maintaining two conservation laws for the linear advection equation,

$$(2.1) \quad u_t + u_x = 0.$$

We know that (2.1) possesses infinitively many conservation laws; as a matter of fact, for any smooth function $U(u)$ of u the equation

$$(2.2) \quad U(u)_t + U(u)_x = 0$$

also holds. We call u the momentum. We then choose $U(u) = u^2$ as the redundant quantity and call it the energy.

For numerical discretization, we consider only uniform grids and use the notation $x_j = jh$, $x_{j \pm \frac{1}{2}} = (j \pm \frac{1}{2})h$ and $t_n = n\tau$, with h and τ being the spatial and temporal increments, respectively. The scheme is of the Godunov-type; however, different from the ordinary Godunov-type schemes as described in [11] and [12], it involves

two numerical components, one is a cell-average approximation to the momentum,

$$(2.3) \quad u_j^n \simeq \bar{u}_j^n = \frac{1}{h} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} u(x, t_n) dx,$$

and the other is a cell-average approximation to the energy,

$$(2.4) \quad U_j^n \simeq \bar{U}_j^n = \frac{1}{h} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} U(u(x, t_n)) dx = \frac{1}{h} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} u^2(x, t_n) dx.$$

Like all Godunov-type schemes, the scheme proceeds in the *reconstruction*, *evolution* and *cell-averaging* steps.

Reconstruction. We use the numerical solution $\{(u_j^n, U_j^n)\}$ to reconstruct in each cell a piecewise linear function $R(x; u^n, U^n)$, which is of the form

$$(2.5) \quad R(x; u^n, U^n) = u_j^n + s_j^n(x - x_j), \quad x \in (x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}).$$

Obviously, $R(x; u^n, U^n)$ satisfies

$$(2.6) \quad \frac{1}{h} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} R(x; u^n, U^n) dx = u_j^n.$$

We require that

$$(2.7) \quad \frac{1}{h} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} U(R(x; u^n, U^n)) dx = U_j^n,$$

i.e., the energy cell-average of the reconstructed solution must be equal to the numerical energy. (2.7) is an equation of slope s_j^n , from which we solve s_j^n for the reconstruction (2.5), i.e.,

$$(2.8) \quad (s_j^n)^2 = \frac{12(U_j^n - (u_j^n)^2)}{h^2}.$$

Since s_j^n is an approximation to $u_x(x_j, t_n)$, it should have the same sign as that of $(u_{j+1}^n - u_{j-1}^n)$; therefore, it is computed as

$$(2.9) \quad s_j^n = \text{sgn}(u_{j+1}^n - u_{j-1}^n) \sqrt{\frac{12(U_j^n - (u_j^n)^2)}{h^2}}.$$

Evolution. Evolve (2.1) with the reconstructed solution $R(x; u^n, U^n)$ as the initial value at time t_n ,

$$(2.10) \quad \begin{cases} v_t + v_x = 0, & -\infty < x < \infty, \quad t_n < t \leq t_{n+1}, \\ v(x, t_n) = R(x; u^n, U^n), & -\infty < x < \infty, \end{cases}$$

and we obtain the solution $v(x, t)$ over the time interval $[t_n, t_{n+1}]$.

Cell-Averaging. At t_{n+1} , the numerical momentum u_j^{n+1} and numerical energy U_j^{n+1} are computed as

$$(2.11) \quad u_j^{n+1} = \frac{1}{h} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} v(x, t_{n+1}) dx$$

and

$$(2.12) \quad U_j^{n+1} = \frac{1}{h} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} (v(x, t_{n+1}))^2 dx.$$

In practice, we use the integral form of (2.1) and (2.2) over the cell, instead, to compute u_j^{n+1} and U_j^{n+1} , i.e.,

$$(2.13) \quad u_j^{n+1} = u_j^n - \lambda(\hat{f}_{j+\frac{1}{2}}^n - \hat{f}_{j-\frac{1}{2}}^n)$$

and

$$(2.14) \quad U_j^{n+1} = U_j^n - \lambda(\hat{F}_{j+\frac{1}{2}}^n - \hat{F}_{j-\frac{1}{2}}^n),$$

where $\lambda = \frac{\tau}{h}$ is the mesh ratio and the numerical momentum flux $\hat{f}_{j\pm\frac{1}{2}}^n$ and numerical energy flux $\hat{F}_{j\pm\frac{1}{2}}^n$ are computed as

$$(2.15) \quad \hat{f}_{j\pm\frac{1}{2}}^n = \frac{1}{\tau} \int_{t_n}^{t_{n+1}} v(x_{j\pm\frac{1}{2}}, t) dt$$

and

$$(2.16) \quad \hat{F}_{j\pm\frac{1}{2}}^n = \frac{1}{\tau} \int_{t_n}^{t_{n+1}} v^2(x_{j\pm\frac{1}{2}}, t) dt.$$

Thus, we complete a step of computation. For the stability of the scheme the mesh ratio is restricted by the CFL condition

$$(2.17) \quad \lambda < 1,$$

so that waves emanating from each $x_{j+\frac{1}{2}}$ at t_n will not affect the neighboring cell-edges.

Remark 2.1. The scheme maintains the discrete conservations of the momentum and energy.

Remark 2.2. Because the solution $v(x, t)$ to (2.10) is simply $R(x - t; u^n, U^n)$, the integrals (2.15) and (2.16) can be exactly computed. However, we may numerically compute the integrals, say, by a trapezoid formula, and our experiments show that the scheme using numerical integrations is almost of the same quality as that using exact integrations in computing smooth solutions. This is because when computing smooth solutions the discontinuities of the reconstructed piecewise solution at cell boundaries $x_{j+1/2}$ are of type $O(h^2)$; therefore, there is no strong nonsmooth waves generated from there. The following discussion in this paper will consider only the scheme using exact integrations for the two numerical fluxes.

We present the following theorems concerning the feasibility and stability of the scheme without proof. The proofs of these theorems can be found in [5], [13] or [20].

Theorem 2.1. *The scheme maintains $U_j^n \geq (u_j^n)^2$ provided $U_j^0 \geq (u_j^0)^2$ holds at the initial time.*

Remark 2.3. Theorem 2.1 guarantees that the square-root operation in (2.9) is always valid. According to the Jensen inequality and the convexity of $U(u)$ the true solution $u(x, t_n)$ and its energy $U(u(x, t_n))$ satisfy

$$(2.18) \quad \frac{1}{h} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} U(u(x, t_n)) dx \geq U\left(\frac{1}{h} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} u(x, t_n) dx\right);$$

therefore, we called the condition

$$(2.19) \quad U_j^n \geq (u_j^n)^2$$

the *Jensen condition*.

Theorem 2.2. *For the scheme, (1) $\|u^n\|_2 < C$; (2) $U^n \geq 0$; (3) $\|U^n\|_1 \leq C$, provided the initial energy satisfies $\|U^0\|_1 \leq C$ and the Jensen condition holds at the initial time.*

Remark 2.4. Theorem 2.2 shows that the numerical momentum is L_2 controlled by the numerical energy, and the numerical energy itself is nonnegative and L_1 bounded, which indicates that the scheme is stable.

As will be seen in §5, the truncation error of the scheme is of second-order. However, the numerical experiments in the following section will show that the scheme has a third-order convergence rate away from extremes of the solution. We will give an explanation of the super-convergence property of the scheme in §4 and §5.

3. NUMERICAL EXPERIMENTS

To give a numerical view of the scheme, we recondut in this section two numerical experiments in [20] and [21]. However, we shall more carefully investigate the accuracy and long-time behavior of the numerical solutions. We will only look at the momentum u ; therefore, in the following discussion of this section, when talk about the true or numerical solutions, we mean the true or numerical momentums.

As a convention of numerical practice (see §8.1 of [11]) we measure the accuracy of a numerical solution by its global error, the error between the true and numerical solutions, at the final time T of the experiment,

$$(3.1) \quad e_j^n = \bar{u}_j^n - u_j^n,$$

where \bar{u}_j^n is the cell-average of the true solution as defined in (2.3). The time step n is determined by $T = n\tau$ and will go to infinity as the time step τ tends to zero. When the global error is periodic in x direction with L the period, which happens when both the true and numerical solutions are periodic, the l_1 and l_∞ norms of it over the domain $(0, L)$ are defined as

$$(3.2) \quad \|e^n\|_1 = \sum_{1 \leq j \leq J} h |e_j^n|$$

and

$$(3.3) \quad \|e^n\|_\infty = \max_{1 \leq j \leq J} |e_j^n|,$$

respectively, with $Jh = L$. The number of cells, J , also goes to infinity as h tends to zero. We will use these two norms to measure the magnitude of the numerical error. We will also define sub-domain l_1 and l_∞ norms of e^n over a sub-domain $\Omega \subset (0, L)$ as

$$(3.4) \quad \|e^n\|_{\Omega,1} = \sum_{x_j \in \Omega} h |e_j^n|$$

and

$$(3.5) \quad \|e^n\|_{\Omega,\infty} = \max_{x_j \in \Omega} |e_j^n|,$$

respectively, with $x_j = jh$, and sometimes use them to measure the magnitude of the numerical error in the sub-domain.

Example 3.1. Consider the following initial value for the linear advection equation (2.1),

$$(3.6) \quad u(x, 0) = \sin(2\pi x), \quad 0 \leq x \leq 1,$$

with periodic boundary conditions at the two ends. The true solution of this problem is simply a periodic function $u(x, t) = \sin(2\pi(x - t))$ with the period $L = 1$. We use this example to study the accuracy and convergence rate of the scheme.

The mesh ration $\lambda = \frac{\tau}{h}$ is taken to be 0.8. We perform the computation on grids of 20, 40, 80, 160, 320, 640 and 1280 cells, respectively, up to final time $T = 1$ (one period), and the l_1 and l_∞ norms of the global error and the convergence rate of the numerical solution are presented in Table 1.

TABLE 1. Global error and convergence rates over domain $(0, L)$.

N	l_∞	rate	l_1	rate
20	1.1660E-002	-	2.5230E-003	-
40	5.0662E-003	1.2030	5.5097E-004	2.1952
80	2.0434E-003	1.3100	1.1655E-004	2.2412
160	7.9422E-004	1.3635	2.4419E-005	2.2550
320	3.0190E-004	1.3956	5.2994E-006	2.2040
640	1.0828E-004	1.4794	1.1327E-006	2.2262
1280	4.0400E-005	1.4224	2.3414E-007	2.2745

The table shows that the l_1 norm of the numerical error of the numerical solution is higher than second-order, which is contrary to what is expected since the numerical error of a second-order scheme is expected to be also of second-order; see the discussion in §5. The l_∞ norm of the numerical error of the scheme is however lower than second-order, but still higher than first-order. We checked the numerical error cell by cell and found that the accuracy degeneration happened in the cells near the two extremes, $x = \frac{1}{4}$ and $x = \frac{3}{4}$, of the solution.

TABLE 2. Numerical error and convergence rates over the sub-domain Ω .

N	L^∞	rate	L^1	rate
40	2.3945E-004	-	6.7777E-005	-
80	3.6268E-005	2.7231	9.5596E-006	2.8260
160	4.5369E-006	2.9991	1.1395E-006	3.0687
320	5.9680E-007	2.9266	1.4615E-007	2.9631
640	7.4664E-008	2.9990	1.8050E-008	3.0176
1280	9.4589E-009	2.9808	2.2708E-009	2.9909
2560	1.1924E-009	2.9880	2.8293E-010	3.0049

To more carefully investigate the accumulation of numerical error away from the two extremes, we take $\Omega = [0, \frac{1}{6}] \cup [\frac{1}{3}, \frac{2}{3}] \cup [\frac{5}{6}, 1]$ as a sub-domain, which excludes the vicinities of the two extremes, and measure the global error over Ω . Table 2 shows the sub-domain l_1 and l_∞ norms of the global error and the corresponding convergence rates. It is clearly seen from Table 2 that the global error of the

numerical solution in both the norms is of third-order away from the two extremes of the solution. This indicates that the scheme is super-convergent away from the extremes of solutions, and the accuracy degeneration occurs only near the extremes.

Example 3.2. Consider the linear advection equation (2.1) with the following initial value

$$(3.7) \quad u(x, 0) = \exp\left\{-100 \times \left(x - \frac{1}{2}\right)^2\right\} \sin(80x), \quad 0 \leq x < 1,$$

with periodic boundary conditions at the two ends. This example is the Wave-packet problem; see [11]. The true solution of the problem is still $u(x, t) = u(x-t, 0)$. Because of the highly oscillatory structure of the solution, this example is more difficult than the previous one, and is always used to test the long-time behaviors of different numerical methods.

We conduct the numerical experiment on a grid of 200 cells ($h = 0.005$), with the mesh ratio $\lambda = 0.8$, up to $t = 10, 100$ and 200 , and the plots at the times are displayed in Figure 3.1 with solid lines to represent the true solution and circles to represent the numerical. It is seen that the numerical solution of the scheme has a good long-time behavior with the oscillatory structure well preserved even at very late times. There are over- and under-shootings near the extremes of the solution, which also indicate that there is accuracy degeneration there.

To make comparisons, we use a second-order DG method (without TVD limiter) and the Lax-Wendroff scheme (second-order) to compute the same problem. To make the comparisons fair, the experiment of the DG method is implemented on a 200-cell grid and the experiment of the Lax-Wendroff scheme is implemented on a 400-cell grid, so that all the numerical methods in competition are of the same order and have the same degrees of freedom. The plots of the two solutions at $t = 10$ are displayed in Figure 3.2, and still the solid lines are the true solution and the circles are the numerical solutions. It is clearly seen that both the numerical solutions at the time lose their integrities due to the numerical diffusions and phase errors and have no way to compete with the solution computed by our scheme.

Remark 3.1. In [20] and [21], Wang used the scheme maintaining three conservation laws for (2.1) to compute this Wave-packet problem up to the time $t = 20,000$, and the numerical result at that time was even better than the result at $t = 100$ presented in Figure 3.1; see also [5] and [16] for the numerical results.

4. TWO THEOREMS CONCERNING THE EVOLUTION OF GLOBAL ERROR

In this section, we shall prove two theorems that describe the evolution of global error of our scheme. In the following discussion we assume that the true solution to (2.1), $u(x, t)$, is smooth enough, which means that $u(x, t)$ and all its derivatives appear in the following discussion are continuous functions of x and t . Also when we say a point (x_j, t_n) is away from extremes of the solution, we mean

$$(4.1) \quad |u_x(x_j, t_n)| \geq \alpha > 0,$$

with α an arbitrary constant. The global error of the momentum has been stated in (3.1), which is equivalent to

$$(4.2) \quad \bar{u}_j^n = u_j^n + e_j^n.$$

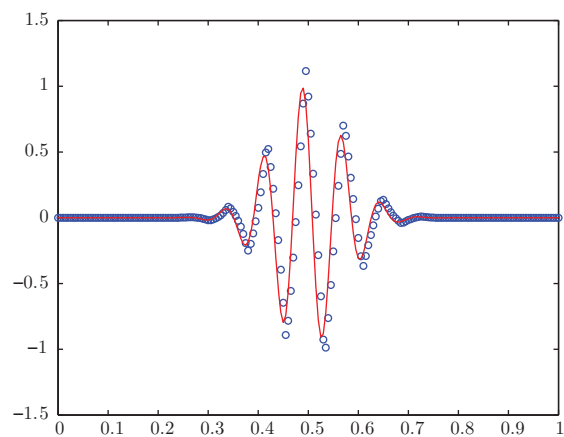
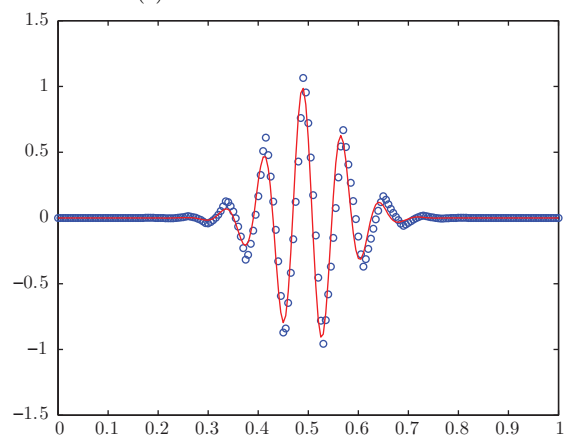
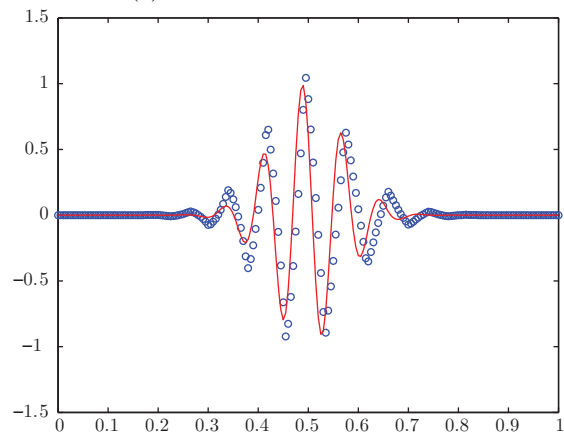
(a) Numerical solution at $t = 10$.(b) Numerical solution at $t = 100$.(c) Global solution at $t = 200$.

FIGURE 3.1. Numerical solutions computed by the scheme

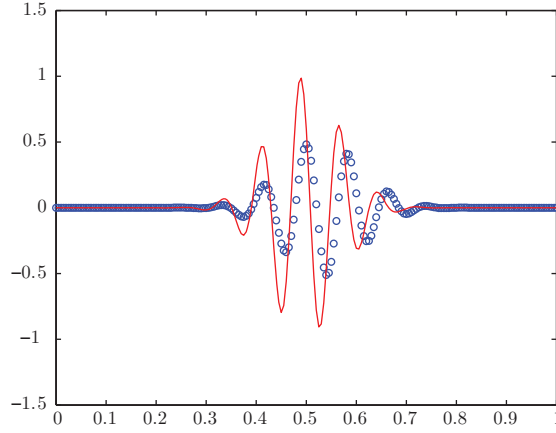
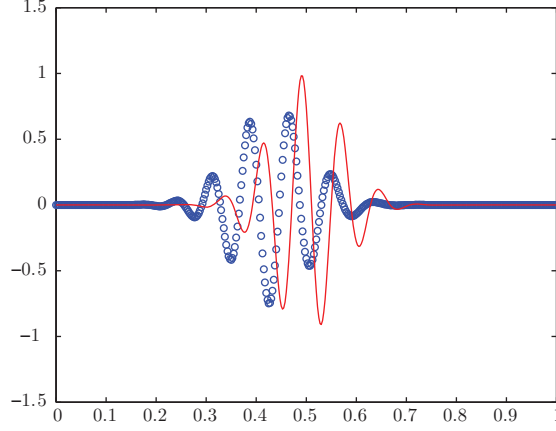
(a) Numerical solution at $t = 10$ computed by the second-order DG method.(b) Numerical solution at $t = 10$ computed by the Lax-Wendroff scheme (second-order).

FIGURE 3.2. Numerical solutions computed by the DG method and Lax-Wendroff scheme.

For the investigation, we need also to introduce the global error of the energy, E_j^n , at t_n as

$$(4.3) \quad \bar{U}_j^n = U_j^n + E_j^n,$$

where \bar{U}_j^n is the cell-average of the true energy defined as in (2.4). In the following discussion we shall use the notation

$$(4.4) \quad \Delta w_j = w_{j+1} - w_j.$$

For simplicity of discussion, we take $\lambda = \tau/h$ to be a constant smaller than 1 in the following discussion; therefore, we can use only the spatial increment h to measure all the errors in the discussion. We say that a grid function $w = O(h^r)$ is continuous in space if

$$(4.5) \quad \Delta w_j = O(h^{r+1}).$$

We assume that the global errors e^n and E^n are all continuous in space. This assumption comes from the fact that the scheme, as described in §2, performs in

the same way in all grid cells and involves no discontinuous operations; therefore, the global errors should be continuous in space when a smooth solution is simulated. We should note that this is not always true for schemes simulating (2.1). Nonlinear schemes such as TVD or ENO types (see [7] and [8]) involve sudden changes of stencils, and they may thus perform in different ways in neighboring cells. However, the WENO schemes using weighted ENO-stencils (see [10]) do not involve sudden change of stencils, and the assumption can be made for them.

We shall first prove the following theorem concerning the error evolution of the scheme.

Theorem 4.1. *If both e^n and E^n are of $O(h^3)$, then away from extremes of the solutions we have*

$$(4.6) \quad e_j^{n+1} = e_j^n + \lambda C(\lambda) u_{xxx}(x_j, t_n) h^3 + \lambda D(\lambda) \Delta d_j^n h^2 + O(h^4),$$

$$(4.7) \quad \begin{aligned} E_j^{n+1} = E_j^n &+ 2\lambda C(\lambda) (u(x_j, t_n) u_{xx}(x_j, t_n))_x h^3 \\ &+ 2\lambda D(\lambda) \Delta(u(x_j, t_n) d_j^n) h^2 + O(h^4), 1 \end{aligned}$$

where

$$(4.8) \quad d_j^n = \frac{6}{u_x(x_j, t_n) h^3} (2u(x_j, t_n) e_j^n - E_j^n)$$

and

$$(4.9) \quad C(\lambda) = -(2\lambda^2 - 3\lambda + 1)/12, \quad D(\lambda) = (1 - \lambda)/2,$$

and near extremes of the solution e_j^n and E_j^n are at least of $O(h^2)$.

We should note that according to the assumptions on e_j^n and E_j^n and (4.8), the term d_j^n is of $O(1)$, and both the terms Δd_j^n and $\Delta(u(x_j, t_n) d_j^n)$ in (4.6) and (4.7) are of $O(h)$. Therefore, the global errors of the momentum and energy at t_{n+1} are still of $O(h^3)$ as at t_n away from extremes of the true solution.

The proof of Theorem 4.1 is somehow lengthy, and it is helpful to sketch a road-map of the proof at first. We see that the linear advection equation (2.1) can be written in an integral form,

$$(4.10) \quad \bar{u}_j^{n+1} = \bar{u}_j^n - \lambda(\tilde{f}_{j+\frac{1}{2}}^n - \tilde{f}_{j-\frac{1}{2}}^n),$$

with $\tilde{f}_{j\pm\frac{1}{2}}^n$ the cell-boundary-average of the true momentum flux at $x = x_{j\pm\frac{1}{2}}$,

$$(4.11) \quad \tilde{f}_{j\pm\frac{1}{2}}^n = \frac{1}{\tau} \int_{t_n}^{t_{n+1}} u(x_{j\pm\frac{1}{2}}, t) dt.$$

We also have for the energy of the solution,

$$(4.12) \quad \bar{U}_j^{n+1} = \bar{U}_j^n - \lambda(\tilde{F}_{j+\frac{1}{2}}^n - \tilde{F}_{j-\frac{1}{2}}^n),$$

with $\tilde{F}_{j\pm\frac{1}{2}}^n$ the cell-boundary-average of the true energy flux at $x_{j\pm\frac{1}{2}}$,

$$(4.13) \quad \tilde{F}_{j\pm\frac{1}{2}}^n = \frac{1}{\tau} \int_{t_n}^{t_{n+1}} u^2(x_{j\pm\frac{1}{2}}, t) dt.$$

By subtracting (2.13) from (4.10) and (2.14) from (4.12) and denoting the errors between the numerical and true fluxes by

$$(4.14) \quad R_{j+\frac{1}{2}}^{n,f} = \tilde{f}_{j+\frac{1}{2}}^n - \hat{f}_{j+\frac{1}{2}}^n \text{ and } R_{j+\frac{1}{2}}^{n,F} = \tilde{F}_{j+\frac{1}{2}}^n - \hat{F}_{j+\frac{1}{2}}^n,$$

we obtain

$$(4.15) \quad e_j^{n+1} = e_j^n - \lambda(R_{j+\frac{1}{2}}^{n,f} - R_{j-\frac{1}{2}}^{n,f}),$$

and

$$(4.16) \quad E_j^{n+1} = E_j^n - \lambda(R_{j+\frac{1}{2}}^{n,F} - R_{j-\frac{1}{2}}^{n,F}).$$

It is clearly seen from (4.15) and (4.16) that the estimations of e_j^{n+1} and E_j^{n+1} turn out to be the estimations of the two flux errors defined in (4.14). According to the description of the scheme in §2, to estimate the flux errors one needs first to estimate the error between the reconstruction slope s_j^n computed by (2.9) and the $u_x(x_j, t_n)$ of the true solution. Thus, the roadmap of the proof is: 1) estimate the error of s_j^n ; 2) estimate the errors of the two numerical fluxes; and then 3) estimate the errors of the numerical momentum and energy.

Lemma 4.1. *If both e^n and E^n are of $O(h^3)$, we then have for the slope computed by (2.9),*

$$(4.17) \quad s_j^n = u_x(x_j, t_n) + d_j^n h + O(h^2)$$

away from extremes of the true solution with the $O(h^2)$ remainder continuous in space in the sense of (4.5), and we still have

$$(4.18) \quad s_j^n = u_x(x_j, t_n) + \beta_j^n$$

near extremes of the solution with β_j^n a grid function of $O(1)$ and $\Delta\beta_j^n = O(h)$.

Lemma 4.2. *If both e^n and E^n are of $O(h^3)$, we then have for the error of the momentum flux $R^{n,f}$,*

$$(4.19) \quad R_{j+\frac{1}{2}}^{n,f} = -C(\lambda)u_{xx}(x_j, t_n)h^2 - D(\lambda)d_j^n h^2 + O(h^3)$$

away from extremes of the true solution with the $O(h^3)$ remainder continuous in space, and we still have

$$(4.20) \quad R_{j+\frac{1}{2}}^{n,f} = D(\lambda)\beta_j^n h + O(h^2)$$

near extremes of the solution.

Lemma 4.3. *If e^n and E^n are of $O(h^3)$, we then have for the error of the energy flux $R^{n,F}$,*

$$(4.21) \quad R_{j+\frac{1}{2}}^{n,F} = -2C(\lambda)u(x_j, t_n)u_{xx}(x_j, t_n)h^2 - 2D(\lambda)u(x_j, t_n)d_j^n h^2 + O(h^3)$$

away from extremes of the true solution with the $O(h^3)$ remainder continuous in space, and still have

$$(4.22) \quad R_{j+\frac{1}{2}}^{n,F} = 2D(\lambda)u(x_j, t_n)\beta_j^n h + O(h^2)$$

near extremes of the solution.

The proofs of all of the lemmas are left to Appendix A.

Proof of Theorem 4.1. The first conclusion of the theorem, (4.6) and (4.7) in the regions away from extremes of the solution, follows easily by substituting (4.19) into (4.15) and (4.21) into (4.16) and noting that the $O(h^3)$ remainders are continuous in space. The second conclusion of the theorem, i.e., the errors of the numerical

momentum and energy near extremes of the solution are still at least of second-order, and follows easily by substituting (4.20) into (4.15) and (4.22) into (4.16) and noting $\Delta\beta_j^n = O(h)$. \spadesuit

In the following discussion, we will focus only on the solution regions away from extremes. We need to prove the following theorem concerning the evolution of d^n , Δd^n and $\Delta(u(x_j, t_n)d_j^n)$ for the further study of error accumulation.

Theorem 4.2. *If both e^n and E^n are of $O(h^3)$, we then have for d^n away from extremes of the solution,*

$$(4.23) \quad d_j^{n+1} = (1 - 12\lambda D(\lambda))d_j^n - 12\lambda C(\lambda)u_{xx}(x_j, t_n) + O(h)$$

and

$$(4.24) \quad \Delta d_j^{n+1} = (1 - 12\lambda D(\lambda))\Delta d_j^n - 12\lambda C(\lambda)u_{xxx}(x_j, t_n)h + O(h^2).$$

If we denote $D_j^n = u(x_j, t_n)d_j^n$, we then have for ΔD_j^n away from extremes of the solution

$$(4.25) \quad \Delta D_j^{n+1} = (1 - 12\lambda D(\lambda))\Delta D_j^n - 12\lambda C(\lambda)(u(x_j, t_n)u_{xx}(x_j, t_n))_x h + O(h^2).$$

The proof of the theorem will be left to Appendix B.

5. ERROR SELF-CANCELING OF THE SCHEME

In this section, we shall investigate the accumulation of numerical errors of the scheme to give an explanation of its super-convergence property away from extremes of the solutions. To this end, we need first to observe how the numerical error is accumulated in a linear scheme modeling (2.1). We consider a linear finite volume scheme in the form,

$$(5.1) \quad u_j^{n+1} = u_j^n - \tau L_h(u^n; j) = u_j^n - \lambda L_h(u^n; j)h,$$

where u_j^n is a cell-average approximation to the true solution as in (2.3) and L_h is a linear difference operator approximating $\frac{\partial}{\partial x}$. It is obvious that L_h must be a linear combination of Δ 's as defined in (4.4) acting in the nearby cells. The most important linear schemes for (2.1), such as the Lax-Wendroff scheme, Beam-Warming scheme, MacCormack scheme, etc., can all be written in this form.

We know that the truncation error of scheme (5.1), T_j^n , is defined by replacing the cell-average approximation u_j^n by the cell-average of the true solution \bar{u}_j^n ,

$$(5.2) \quad T_j^n = \frac{\bar{u}_j^{n+1} - \bar{u}_j^n}{\tau} + L_h(\bar{u}^n; j);$$

see [12]. The truncation error measures how well the scheme models (2.1) locally. A scheme is said to be of r th-order if $T_j^n = O(h^r)$. The quantity $\tau T_j^n = \lambda T_j^n h$, called residual, is the numerical error produced in one step by the scheme starting with the true solution.

It is a well-known fact that for a linear scheme the numerical errors produced in different time steps are accumulated in a linear fashion; therefore, the truncation and global errors are of the same order. The fact can be theoretically interpreted in many different ways, say, by phase error analysis (see [17] and [18]), or by energy analysis; see [9]. Here we would like to give an intuitive interpretation of it in

truncation error. Let us assume that (5.1) is of second-order with its truncation error in the form

$$(5.3) \quad T_j^n = A(x_j, t_n, \lambda; u)h^2 + O(h^3),$$

where $A(x_j, t_n, \lambda; u)$ is the coefficient of the leading part of the error, which can be evaluated by Taylor expansions and is thus related to the third derivatives of $u(x, t)$.

The global error e^n is still defined as in (4.2), which can also be assumed to be continuous in space in the sense of (4.5) because of the linear form of L_h . Now we are going to observe the accumulation of numerical error of the scheme, and to this end we assume $e^n = O(h^3)$. Then by multiplying τ onto (5.2), subtracting (5.1) from it, substituting (5.3), and noting the linearity of L_h , we have

$$(5.4) \quad e_j^{n+1} = e_j^n + \lambda A(x_j, t_n, \lambda; u)h^3 + O(h^4).$$

Note that $L_h(e^n; j) = O(h^4)$ because of the linear form of L_h and the continuity of e^n in space. Estimation (5.4) indicates that the numerical error of the scheme is accumulated in a linear fashion. To see this, let us start from the initial time level at which $e^n \equiv 0$. We then repeatedly use (5.4) to obtain

$$(5.5) \quad \begin{aligned} e_j^1 &= \lambda A(x_j, t_1, \lambda; u)h^3 + O(h^4), \\ e_j^2 &= \lambda A(x_j, t_1, \lambda; u)h^3 + \tau A(x_j, t_2, \lambda; u)h^2 + O(h^4), \dots \end{aligned}$$

Estimations (5.5) indicate that the leading parts of the residuals in different time steps are simply summed up in the global error as the scheme evolves in time. If we ignore the $O(h^4)$ remainder in (5.4) and sum it up from $n = 0$ to $n \leq T/\tau$ with T the final time, we can easily obtain an estimation of the global error

$$(5.6) \quad |e_j^n| \leq K_A T h^2, \text{ for } 0 < n \leq \frac{T}{\tau},$$

with K_A an upper bound of $A(x_j, t_n, \lambda; u)$. Estimation (5.6) indicates that the global error is of second-order. Moreover, the coefficient of the global error, $K_A T$, is proportional to the final time T . In the real situation, the ignored $O(h^4)$ remainder also contributes to the global error; therefore, the coefficient of the global error is expected to be of $\exp(CT)$ with C a constant; see [9] and [18].

Estimation (5.6) is obtained assuming that the truncation errors in different time steps do not cancel each other, and one may expect that there is self-canceling of the truncation error in reality, which may increase the order of global error. Unfortunately, all numerical practices show that the order of global error of a linear scheme is the same as its truncation error and the global error grows exponentially along with the time, which indicates that there is essentially no self-canceling of truncation error. Moreover, this situation is true even for most modern nonlinear scheme modeling (2.1), such as that of the TVD, ENO or WENO types; numerical errors in those nonlinear schemes are also accumulated in linear fashions.

Now we are going to investigate the accumulation of numerical error of our scheme. We can also define the truncation error of the scheme as by replacing the cell-average approximations by the true cell-averages. However, since our scheme involves two numerical entities, the truncation error will also have two components. For clarity, we denote by L_h^u and L_h^U the difference operators in (2.13) and (2.14),

i.e.,

$$(5.7) \quad L_h^u(u^n, U^n; j) = \frac{\hat{f}_{j+\frac{1}{2}}^n - \hat{f}_{j-\frac{1}{2}}^n}{h}$$

and

$$(5.8) \quad L_h^U(u^n, U^n; j) = \frac{\hat{F}_{j+\frac{1}{2}}^n - \hat{F}_{j-\frac{1}{2}}^n}{h}.$$

Note that L_h^u and L_h^U are nonlinearly related to u^n and U^n because both the numerical fluxes $\hat{f}_{j\pm\frac{1}{2}}^n$ and $\hat{F}_{j\pm\frac{1}{2}}^n$ are evaluated in nonlinear fashions as described in §2.

Definition 5.1. The two components of the truncation error of the scheme described in §2 are

$$(5.9) \quad T_j^{n,u} = \frac{\bar{u}_j^{n+1} - \bar{u}_j^n}{\tau} + L_h^u(\bar{u}^n, \bar{U}^n; j)$$

and

$$(5.10) \quad T_j^{n,U} = \frac{\bar{U}_j^{n+1} - \bar{U}_j^n}{\tau} + L_h^U(\bar{u}^n, \bar{U}^n; j),$$

respectively.

To evaluate the truncation error of our scheme, we take $e^n \equiv 0$ and $E^n \equiv 0$ in Theorem 4.1, which implies also $d^n \equiv 0$, and note $T_j^{n,u} = e_j^{n+1}/\tau$ and $T_j^{n,U} = E_j^{n+1}/\tau$ at the time. Then from Theorem 4.1, the two components of the truncation error are

$$(5.11) \quad T_j^{n,u} = C(\lambda)u_{xxx}(x_j, t_n)h^2 + O(h^3)$$

and

$$(5.12) \quad T_j^{n,U} = 2C(\lambda)(u(x_j, t_n)u_{xx}(x_j, t_n))_x h^2 + O(h^3),$$

away from extremes of the solution, and they are at least of $O(h)$ near extremes of the solution. This indicates that the scheme is of second-order away from extremes of the solution and is still at least of first-order near extremes.

In the following discussion we focus only on the region away from extremes of the solution. As for the linear scheme (5.1), we use Theorem 4.1 and Theorem 4.2 to estimate the global error of our scheme starting from the initial time level. It is easy to obtain by taking $n = 0$ in (5.11) and (5.12),

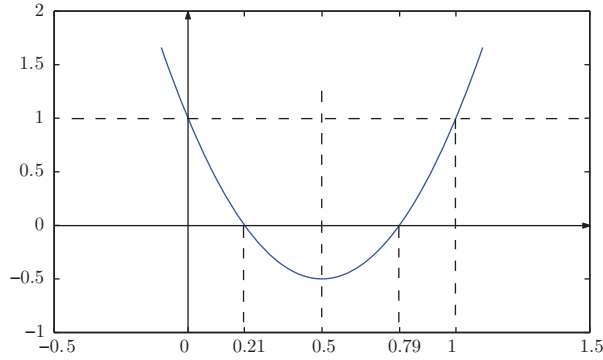
$$(5.13) \quad e_j^{1,u} = \lambda C(\lambda)u_{xxx}(x_j, t_0)h^3 + O(h^4)$$

and

$$(5.14) \quad E_j^{1,U} = 2\lambda C(\lambda)(u(x_j, t_0)u_{xx}(x_j, t_0))_x h^3 + O(h^4).$$

Note that (5.13) and (5.14) indicate that both e^1 and E^1 are of $O(h^3)$ and are continuous in space because of the smoothness of the true solution; therefore, we can use Theorem 4.1 and Theorem 4.2 again to obtain the estimation of the global error at t_2 . By noting $d^0 \equiv 0$, we can obtain from Theorem 4.2

$$(5.15) \quad \Delta d_j^1 = -12\lambda C(\lambda)u_{xxx}(x_j, t_0)h + O(h^2)$$

FIGURE 5.1. The plot of the function $P(\lambda)$

and

$$(5.16) \quad \Delta(u(x_j, t_1)d_j^1) = -12\lambda C(\lambda)(u(x_j, t_0)u_{xx}(x_j, t_0))_x h + O(h^2),$$

away from extremes of the solution. Then from Theorem 4.1 we can obtain

$$(5.17) \quad e_j^2 = (1 - 12\lambda D(\lambda))\lambda C(\lambda)u_{xxx}(x_j, t_0)h^3 + \lambda C(\lambda)u_{xxx}(x_j, t_1)h^3 + O(h^4)$$

and

$$(5.18) \quad \begin{aligned} E_j^2 = & (1 - 12\lambda D(\lambda))2\lambda C(\lambda)(u(x_j, t_0)u_{xx}(x_j, t_0))_x h^3 \\ & + 2\lambda C(\lambda)(u(x_j, t_1)u_{xx}(x_j, t_1))_x h^3 + O(h^4), \end{aligned}$$

away from extremes of the solution. We note in (5.17) and (5.18) that the leading parts of the residuals in the two steps are not simply summed up in e^2 and E^2 as in linear scheme (5.1), the residual in the first step is reduced by a factor $(1 - 12\lambda D(\lambda))$. This is because the terms Δd_j^1 and $\Delta(u(x_j, t_1)d_j^1)$ produce quantities that cancel part of them.

We need to study the factor $(1 - 12\lambda D(\lambda))$, and we denote it by $P(\lambda)$. It is not difficult to obtain from (4.9) that

$$(5.19) \quad P(\lambda) = 6\lambda^2 - 6\lambda + 1.$$

The function $P(\lambda)$ is quadratic of the mesh ratio $\lambda = \frac{\tau}{h}$ and its plot is presented in Figure 5.1. The plot tells us the following things:

(1) When $0 < \lambda < 1$, which is required by the CFL condition for the scheme, we have $|P(\lambda)| < 1$. This indicates that $P(\lambda)$ does reduce the residual in the first step in (5.17) and (5.18).

(2) Function $P(\lambda)$ has two roots $\lambda_1 = (3 - \sqrt{3})/6$ and $\lambda_2 = (3 + \sqrt{3})/6$, and $P(\lambda) < 0$ when $\lambda_1 < \lambda < \lambda_2$. This indicates that in this region of λ the residuals in the first and second steps in (5.17) and (5.18) continue to cancel each other.

(3) Function $P(\lambda)$ reaches its minimum -0.5 at $\lambda = 0.5$. This indicates that at $\lambda = 0.5$ the canceling-each-other of the residuals in the two successive steps reaches its maximum.

We would like to have a view of the global error at late times. To simplify the discussion, we ignore all the higher-order remainders in (4.6), (4.7), (4.24) and (4.25). We then have the following proposition.

Proposition 5.1. *If the $O(h^4)$ remainders in (4.6) and (4.7) and the $O(h^2)$ remainders in (4.24) and (4.25) are ignored, then away from extremes of the solution we have the following estimations of the two components of the global error at t_n ,*

$$(5.20) \quad e_j^n = \sum_{i=0}^{n-1} \lambda C(\lambda) P^i(\lambda) u_{xxx}(x_j, t_{n-1-i}) h^3$$

and

$$(5.21) \quad E_j^n = \sum_{i=0}^{n-1} 2\lambda C(\lambda) P^i(\lambda) (u(x_j, t_{n-1-i}) u_{xx}(x_j, t_{n-1-i}))_x h^3.$$

The proof of Proposition 5.1 is left to Appendix C.

Now if we assume $|u_{xxx}| \leq K_1$ and $|(uu_{xx})_x| \leq K_2$ we can obtain the following estimations of the global error

$$(5.22) \quad |e_j^n| \leq \sum_{i=0}^{n-1} \lambda C(\lambda) P^i(\lambda) K_1 h^3 < \frac{2\lambda C(\lambda)}{1 - P(\lambda)} K_1 h^3$$

and

$$(5.23) \quad |E_j^n| \leq \sum_{i=0}^{n-1} 2\lambda C(\lambda) P^i(\lambda) K_2 h^3 < \frac{4\lambda C(\lambda)}{1 - P(\lambda)} K_2 h^3.$$

The estimations (5.22) and (5.23) indicate that the two components of the global error of the numerical solution are both of the third-order, an order higher than the truncation error. Moreover, the coefficients of the global error, $\frac{2\lambda C(\lambda)}{1 - P(\lambda)} K_1$ and $\frac{4\lambda C(\lambda)}{1 - P(\lambda)} K_2$, are all uniformly bounded with respect to the final time $T > n\tau$. The ignored higher-order remainders in (4.6), (4.7), (4.24) and (4.25) will surely contribute to the global errors; however, their contribution will not affect the order but change the coefficients in (5.22) and (5.23). We guess that there are also self-canceling in the higher-order remainders, and thus the real situation is that the increase of the coefficients with respect to the time is slower than the linear fashion. However, this should be investigated with further analysis. We thus finally explain the super-convergence of the scheme away from extremes of the solution.

From the above discussion, we see that d_j^n defined in (4.8) plays an important role in the self-canceling of numerical errors; therefore, we would like to see its mathematical meaning before ending this section. From (A.3) in the proof of the Lemma 4.1 in Appendix A, we see

$$(5.24) \quad \bar{u}_j^n = u(x_j, t_n) + \frac{1}{24} h^2 u_{xx}(x_j, t_n) + O(h^4).$$

If $e_j^n = \bar{u}_j^n - u_j^n = O(h^3)$, we have from (5.24),

$$(5.25) \quad 2u(x_j, t_n) = 2\bar{u}_j^n + O(h^2) = \bar{u}_j^n + u_j^n + O(h^2).$$

Substituting $e_j^n = \bar{u}_j^n - u_j^n$ and $E_j^n = \bar{U}_j^n - U_j^n$ into (4.8) and noting (5.25), we have

$$(5.26) \quad d_j^n = \frac{6}{u_x(x_j, t_n) h^3} [(U_j^n - (u_j^n)^2) - (\bar{U}_j^n - (\bar{u}_j^n)^2)] + O(h^2).$$

We call the term

$$(5.27) \quad \bar{U}_j^n - (\bar{u}_j^n)^2$$

the Jensen difference of the true solution and the term

$$(5.28) \quad U_j^n - (u_j^n)^2$$

the Jensen difference of the numerical solution. According to Remark 2.3 and Theorem 2.1, both the terms are nonnegative. Now (5.26) shows that d_j^n is essentially the discrepancy of the numerical and true Jensen differences, which is obviously closely related to the multi-conservation feature of the scheme.

6. CONCLUSION

We have carried out a truncation error investigation on the scheme maintaining two conservation laws for the linear advection equation (2.1). Our investigation reveals that the numerical errors produced in different steps of the scheme are accumulated in a nonlinear fashion, in which they cancel each other. This error self-canceling feature brings the super-convergence property to the scheme, which makes it superior to other numerical schemes modeling (2.1) in both accuracy and long-time simulation.

APPENDIX A

Proof of Lemma 4.1. We first investigate the true solution in the grid cell $[x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$, and we start with the momentum $u(x, t)$. By the Taylor expansion of $u(x, t_n)$ at (x_j, t_n) we obtain

$$(A.1) \quad \begin{aligned} u(x, t_n) = & u(x_j, t_n) + u_x(x_j, t_n)(x - x_j) + \frac{1}{2}u_{xx}(x_j, t_n)(x - x_j)^2 \\ & + \frac{1}{6}u_{xxx}(x_j, t_n)(x - x_j)^3 + O(h^4). \end{aligned}$$

Cell-averaging the two sides of (A.1), we obtain

$$(A.2) \quad u(x_j, t_n) = \bar{u}_j^n - \frac{1}{24}h^2u_{xx}(x_j, t_n) + O(h^4).$$

Now we substitute (A.2) into (A.1) and obtain

$$(A.3) \quad u(x, t_n) = \bar{u}_j^n - \frac{1}{24}h^2u_{xx}(x_j, t_n) + k_j(x, t_n),$$

with

$$(A.4) \quad \begin{aligned} k_j(x, t_n) = & u_x(x_j, t_n)(x - x_j) + \frac{1}{2}u_{xx}(x_j, t_n)(x - x_j)^2 \\ & + \frac{1}{6}u_{xxx}(x_j, t_n)(x - x_j)^3 + O(h^4). \end{aligned}$$

Now we investigate the cell-average of the true energy \bar{U}_j^n and note

$$(A.5) \quad \bar{U}_j^n = \frac{1}{h} \int_{x_{j-1/2}}^{x_{j+1/2}} U(u(x, t_n)) dx = \frac{1}{h} \int_{x_{j-1/2}}^{x_{j+1/2}} u^2(x, t_n) dx.$$

Substituting (A.3) into (A.5), we obtain

$$(A.6) \quad \bar{U}_j^n = (\bar{u}_j^n)^2 + \frac{h^2}{12}(u_x(x_j, t_n))^2 + O(h^4).$$

We then investigate the numerical solution. From (4.2) and (4.3) we have

$$(A.7) \quad u_j^n = \bar{u}_j^n - e_j^n$$

and

$$(A.8) \quad U_j^n = \bar{U}_j^n - E_j^n,$$

with both e_j^n and E_j^n are $O(h^3)$. Substituting (A.7) and (A.8) into (2.8), we obtain

$$(A.9) \quad (\bar{u}_j^n)^2 - 2\bar{u}_j^n e_j^n + \frac{h^2}{12}(s_j^n)^2 + O(h^6) = \bar{U}_j^n - E_j^n.$$

Substituting (A.6) and (A.2) into (A.9), we have

$$(A.10) \quad (s_j^n)^2 = (u_x(x_j, t_n))^2 + 12(2u(x_j, t_n)e_j^n - E_j^n)/h^2 + O(h^2).$$

In the following we first consider the case that the grid cell under concern is away from extremes of the solution. Because $u_x(x_j, t_n)$ has the same sign as that of $(u_{j+1}^n - u_{j-1}^n)$ away from extremes of the solution, we have from (2.9)

$$(A.11) \quad s_j^n = \text{sgn}(u_x(x_j, t_n)) \sqrt{(u_x(x_j, t_n))^2 + 12(2u(x_j, t_n)e_j^n - E_j^n)/h^2 + O(h^2)}.$$

By the Taylor expansion of the square-root function on the RHS of (A.11) at $(u_x(x_j, t_n))^2$ we have

$$(A.12) \quad s_j^n = u_x(x_j, t_n) + d_j^n h + O(h^2),$$

where

$$(A.13) \quad d_j^n = \frac{6}{u_x(x_j, t_n)h^3}(2u(x_j, t_n)e_j^n - E_j^n).$$

The $O(h^2)$ remainder in (A.12) is continuous in space because both e^n and E^n are continuous in space and the true solution $u(x, t)$ is smooth.

Now we consider the case that the grid cell under concern is near extremes of the solution. In this case, from (A.10) and (2.9) we can have

$$(A.14) \quad s_j^n = u_x(x_j, t_n) + \beta_j^n,$$

with

$$(A.15) \quad \beta_j^n = \text{sgn}(u_{j+1}^n - u_{j-1}^n) \sqrt{(u_x(x_j, t_n))^2 + 12(2u(x_j, t_n)e_j^n - E_j^n)/h^2 + O(h^2)} - u_x(x_j, t_n),$$

Obviously, β_j^n is bounded and satisfies $\Delta(\beta_j^n) = O(h)$ because of the smoothness of the true solution. Thus, the proof is complete. \spadesuit

Proof of Lemma 4.2. We first consider the true momentum flux

$$(A.16) \quad \tilde{f}_{j+\frac{1}{2}}^n = \frac{1}{\tau} \int_{t_n}^{t_{n+1}} u(x_{j+\frac{1}{2}}, t) dt.$$

Because $u(x, t)$ is the true solution of the linear advection equation (2.1), the true flux can be written as

$$(A.17) \quad \tilde{f}_{j+\frac{1}{2}}^n = \frac{1}{\tau} \int_{t_n}^{t_{n+1}} u(x_{j+\frac{1}{2}} - (t - t_n), t_n) dt.$$

We substitute $x = x_{j+\frac{1}{2}} - (t - t_n)$ in (A.3) and obtain

$$(A.18) \quad u(x_{j+\frac{1}{2}} - (t - t_n), t_n) = \bar{u}_j^n - \frac{1}{24}h^2 u_{xx}(x_j, t_n) + k_j(x_{j+\frac{1}{2}} - (t - t_n), t_n),$$

with

$$(A.19) \quad \begin{aligned} k_j(x_{j+\frac{1}{2}} - (t - t_n), t_n) = & u_x(x_j, t_n) \left(\frac{h}{2} - (t - t_n) \right) \\ & + \frac{1}{2} u_{xx}(x_j, t_n) \left(\frac{h}{2} - (t - t_n) \right)^2 + O(h^3). \end{aligned}$$

In (A.19) we ignore the term $\frac{1}{6} u_{xxx}(x_{j+\frac{1}{2}} - (t - t_n), t_n)(x - x_j)^3$ in $k_j(x_{j+\frac{1}{2}} - (t - t_n), t_n)$ because, as will be seen in the following discussion, this term will have no contribution to the leading part of momentum flux error.

Now we substitute (A.18) into (A.17) and have for the true momentum flux

$$(A.20) \quad \tilde{f}_{j+\frac{1}{2}}^n = I_1 + I_2 + I_3 + O(h^3),$$

where the three integrations I_1 , I_2 and I_3 are

$$(A.21) \quad \begin{aligned} I_1 &= \frac{1}{\tau} \int_{t_n}^{t_{n+1}} [\bar{u}_j^n - \frac{1}{24} h^2 u_{xx}(x_j, t_n)] dt, \\ I_2 &= \frac{1}{\tau} \int_{t_n}^{t_{n+1}} u_x(x_j, t_n) \left(\frac{h}{2} - (t - t_n) \right) dt, \\ I_3 &= \frac{1}{\tau} \int_{t_n}^{t_{n+1}} \frac{1}{2} u_{xx}(x_j, t_n) \left(\frac{h}{2} - (t - t_n) \right)^2 dt. \end{aligned}$$

By noting

$$(A.22) \quad \frac{1}{\tau} \int_{t_n}^{t_{n+1}} \left(\frac{h}{2} - (t - t_n) \right) dt = \frac{1 - \lambda}{2} h$$

and

$$(A.23) \quad \frac{1}{\tau} \int_{t_n}^{t_{n+1}} \left(\frac{h}{2} - (t - t_n) \right)^2 dt = \frac{4\lambda^2 - 6\lambda + 3}{12} h^2,$$

we can easily compute

$$(A.24) \quad \begin{aligned} I_1 &= \bar{u}_j^n - \frac{1}{24} h^2 u_{xx}(x_j, t_n), \\ I_2 &= \frac{1 - \lambda}{2} u_x(x_j, t_n) h, \\ I_3 &= \frac{4\lambda^2 - 6\lambda + 3}{24} u_{xx}(x_j, t_n) h^2. \end{aligned}$$

We substitute (A.24) into (A.20), and then the true flux is

$$(A.25) \quad \tilde{f}_{j+\frac{1}{2}}^n = \bar{u}_j^n + \frac{1 - \lambda}{2} u_x(x_j, t_n) h + \frac{2\lambda^2 - 3\lambda + 1}{12} u_{xx}(x_j, t_n) h^2 + O(h^3).$$

Now we consider the numerical momentum flux $\hat{f}_{j+\frac{1}{2}}^n$. From (2.15) and by noting that $v(x, t)$ there is the solution of the IVP (2.10), we have

$$(A.26) \quad \hat{f}_{j+\frac{1}{2}}^n = \frac{1}{\tau} \int_{t_n}^{t_{n+1}} R(x_{j+\frac{1}{2}} - (t - t_n); u^n, U^n) dt.$$

Substituting the reconstruct function (2.5) into (A.26), we obtain for the numerical momentum flux

$$(A.27) \quad \hat{f}_{j+\frac{1}{2}}^n = \frac{1}{\tau} \int_{t_n}^{t_1} [u_j^n + s_j^n(x_{j+\frac{1}{2}} - x_j - (t - t_n))] dt = u_j^n + \frac{1 - \lambda}{2} s_j^n h.$$

When the grid cell under concern is away from extremes of the solution, we substitute (A.7) and (A.12) into (A.27) and obtain

$$(A.28) \quad \hat{f}_{j+\frac{1}{2}}^n = \bar{u}_j^n + \frac{1-\lambda}{2} u_x(x_j, t_n) h + \frac{1-\lambda}{2} d_j^n h^2 + O(h^3).$$

with d_j^n defined as in (A.13) or (4.8). By comparing the true flux in (A.25) and the numerical flux in (A.28) we obtain the error of the momentum flux

$$(A.29) \quad R_{j+\frac{1}{2}}^{n,f} = \frac{2\lambda^2 - 3\lambda + 1}{12} u_{xx}(x_j, t_n) h^2 - \frac{1-\lambda}{2} d_j^n h^2 + O(h^3).$$

The $O(h^3)$ remainder in (A.29) is continuous in space because the $O(h^2)$ remainder in (A.12) is continuous in space and the true solution is smooth. By noting $C(\lambda)$ and $D(\lambda)$ in (4.9), the first conclusion of the lemma then follows.

When the grid cell under concern is near extremes of the solution, by substituting (A.7) and (A.14) into (A.27) we obtain

$$(A.30) \quad \hat{f}_{j+\frac{1}{2}}^n = \bar{u}_j^n + \frac{1-\lambda}{2} u_x(x_j, t_n) h + \frac{1-\lambda}{2} \beta_j^n h + O(h^2),$$

with β_j^n as in (A.15). Again, by comparing the true flux in (A.25) and the numerical flux in (A.30) we obtain the error of the momentum flux in this case

$$(A.31) \quad R_{j+\frac{1}{2}}^{n,f} = \frac{1-\lambda}{2} \beta_j^n h + O(h^2),$$

which is the second conclusion of the lemma. The proof is thus complete. \spadesuit

Proof of Lemma 4.3. We first consider the true energy flux, and by the same argument as for deriving (A.17) for the true momentum flux we have

$$(A.32) \quad \tilde{F}_{j+\frac{1}{2}}^n = \frac{1}{\tau} \int_{t_n}^{t_{n+1}} u^2(x_{j+\frac{1}{2}} - (t - t_n), t_n) dt.$$

We also substitute (A.18), with $k_j(x_{j+\frac{1}{2}} - (t - t_n), t_n)$ as defined in (A.19), into (A.32) and have for the true energy flux

$$(A.33) \quad \tilde{F}_{j+\frac{1}{2}}^n = I_1 + I_2 + I_3 + I_4 + I_5 + O(h^3),$$

where the five integrations I_1, I_2, I_3, I_4 and I_5 are

$$(A.34) \quad \begin{aligned} I_1 &= \frac{1}{\tau} \int_{t_n}^{t_{n+1}} (\bar{u}_j^n)^2 dt, \\ I_2 &= \frac{1}{\tau} \int_{t_n}^{t_{n+1}} u_x^2(x_j, t_n) \left(\frac{h}{2} - (t - t_n)\right)^2 dt, \\ I_3 &= \frac{1}{\tau} \int_{t_n}^{t_{n+1}} -\frac{h^2}{12} \bar{u}_j^n u_{xx}(x_j, t_n) dt, \\ I_4 &= \frac{1}{\tau} \int_{t_n}^{t_{n+1}} 2\bar{u}_j^n u_x(x_j, t_n) \left(\frac{h}{2} - (t - t_n)\right) dt, \\ I_5 &= \frac{1}{\tau} \int_{t_n}^{t_{n+1}} \bar{u}_j^n u_{xx}(x_j, t_n) \left(\frac{h}{2} - (t - t_n)\right)^2 dt. \end{aligned}$$

By noting (A.22) and (A.23), the five integrations I_1 , I_2 , I_3 , I_4 and I_5 can be easily computed as

$$\begin{aligned}
 I_1 &= (\bar{u}_j^n)^2, \\
 I_2 &= \frac{4\lambda^2 - 6\lambda + 3}{12} u_x^2(x_j, t_n) h^2, \\
 I_3 &= -\frac{h^2}{12} \bar{u}_j^n u_{xx}(x_j, t_n), \\
 I_4 &= \bar{u}_j^n u_x(x_j, t_n) (1 - \lambda) h, \\
 I_5 &= \frac{4\lambda^2 - 6\lambda + 3}{12} \bar{u}_j^n u_{xx}(x_j, t_n) h^2.
 \end{aligned}
 \tag{A.35}$$

We substitute (A.35) into (A.33), and then the true energy flux is

$$\begin{aligned}
 \tilde{F}_{j+\frac{1}{2}}^n &= (\bar{u}_j^n)^2 + \frac{4\lambda^2 - 6\lambda + 3}{12} u_x^2(x_j, t_n) h^2 + \bar{u}_j^n u_x(x_j, t_n) (1 - \lambda) h \\
 &\quad + \frac{2\lambda^2 - 3\lambda + 1}{6} \bar{u}_j^n u_{xx}(x_j, t_n) h^2 + O(h^3).
 \end{aligned}
 \tag{A.36}$$

Now we consider the numerical energy flux $\hat{F}_{j+\frac{1}{2}}^n$, and in the same way as for the numerical momentum flux we have

$$\hat{F}_{j+\frac{1}{2}}^n = \frac{1}{\tau} \int_{t_n}^{t_{n+1}} R^2(x_{j+\frac{1}{2}} - (t - t_n); u^n, U^n) dt.
 \tag{A.37}$$

Substituting the reconstruct function (2.5) into (A.37), we obtain the numerical energy flux

$$\hat{F}_{j+\frac{1}{2}}^n = (u_j^n)^2 + u_j^n s_j^n (1 - \lambda) h + (s_j^n)^2 \frac{4\lambda^2 - 6\lambda + 3}{12} h^2,
 \tag{A.38}$$

When the grid cell under concern is away from extremes of the solution, we substitute (A.7) and (A.12) into (A.38) and obtain

$$\begin{aligned}
 \hat{F}_{j+\frac{1}{2}}^n &= (\bar{u}_j^n)^2 + [\bar{u}_j^n u_x(x_j, t_n) + \bar{u}_j^n d_j^n h] (1 - \lambda) h \\
 &\quad + \frac{4\lambda^2 - 6\lambda + 3}{12} u_x^2(x_j, t_n) h^2 + O(h^3),
 \end{aligned}
 \tag{A.39}$$

with d_j^n as in (A.13) or (4.8). By comparing the true flux in (A.36) and the numerical flux in (A.39) we obtain the error of the energy flux

$$R_{j+\frac{1}{2}}^{n,F} = \frac{2\lambda^2 - 3\lambda + 1}{6} \bar{u}_j^n u_{xx}(x_j, t_n) h^2 - (1 - \lambda) \bar{u}_j^n d_j^n h^2 + O(h^3).
 \tag{A.40}$$

Substituting (A.3) into (A.40), the error of the energy flux can be written as

$$R_{j+\frac{1}{2}}^{n,F} = \frac{2\lambda^2 - 3\lambda + 1}{6} u(x_j, t_n) u_{xx}(x_j, t_n) h^2 - (1 - \lambda) u(x_j, t_n) d_j^n h^2 + O(h^3).
 \tag{A.41}$$

The $O(h^3)$ remainder in (A.41) is continuous in space because the $O(h^2)$ remainder in (A.12) is continuous in space and the true solution is smooth. By noting the definition of $C(\lambda)$ and $D(\lambda)$ in (4.9), the first conclusion of the lemma then follows.

When the grid cell under concern is near extremes of the solution, we substitute (A.7) and (A.14) into (A.38) and obtain

$$\hat{F}_{j+\frac{1}{2}}^n = (\bar{u}_j^n)^2 + (1 - \lambda) \bar{u}_j^n u_x(x_j, t_n) h + (1 - \lambda) \bar{u}_j^n \beta_j^n h + O(h^2),
 \tag{A.42}$$

with β_j^n as in (A.15). Again, by comparing the true flux in (A.36) and the numerical flux in (A.42) we obtain the error of the energy flux in this case,

$$(A.43) \quad R_{j+\frac{1}{2}}^{n,F} = (1 - \lambda)\bar{u}_j^n \beta_j^n h + O(h^2).$$

Substituting (A.3) into (A.43), the error of the energy flux can be written as

$$(A.44) \quad R_{j+\frac{1}{2}}^{n,F} = (1 - \lambda)u(x_j, t_n)\beta_j^n h + O(h^2),$$

which is the second conclusion of the lemma. The proof is thus complete. \spadesuit

APPENDIX B

Proof of Theorem 4.2. Multiplying (4.6) by $2u(x_j, t_n)$ and then subtracting (4.7) from it, we obtain

$$(B.1) \quad L_1 = R_1 + R_2 + R_3 + O(h^4),$$

with

$$(B.2) \quad \begin{aligned} L_1 &= 2u(x_j, t_n)e_j^{n+1} - E_j^{n+1}, \\ R_1 &= 2u(x_j, t_n)e_j^n - E_j^n, \\ R_2 &= 2C(\lambda)[u(x_j, t_n)u_{xxx}(x_j, t_n) - (u(x_j, t_n)u_{xx}(x_j, t_n))_x]h^3, \\ R_3 &= 2D(\lambda)[u(x_j, t_n)\Delta d_j^n - \Delta(u(x_j, t_n)d_j^n)]h^2. \end{aligned}$$

Multiply (B.1) by $\frac{6}{u_x(x, t_n)h^2}$ and then noting the definition of d_j^n (4.8) and the smoothness of the solution, we obtain

$$(B.3) \quad \begin{aligned} \frac{6}{u_x(x, t_n)h^2}L_1 &= d_j^{n+1}h + O(h^2), \\ \frac{6}{u_x(x, t_n)h^2}R_1 &= d_j^n h, \\ \frac{6}{u_x(x, t_n)h^2}R_2 &= -12C(\lambda)u_{xx}(x_j, t_n)h, \end{aligned}$$

and

$$(B.4) \quad \frac{6}{u_x(x, t_n)h^2}R_3 = -12D(\lambda)d_{j-1}^n h + O(h^2) = -12D(\lambda)d_j^n h + O(h^2).$$

In the derivation of (B.4) we used the fact $\Delta d_j^n = O(h)$. Substitute all of these relations into (B.1) and the conclusion of (4.23) then follows.

We shall note that the $O(h)$ remainder in (4.23) is continuous in space because of the continuity of e^n and E^n in space and the smoothness of the true solution. Therefore, the conclusion of (4.24) follows by taking the difference operation Δ on the two sides of (4.23). Likewise, the conclusion of (4.25) follows by multiplying $u(x_j, t_{n+1})$ to (4.23) and then taking operation Δ on both sides of it. \spadesuit

APPENDIX C

Proof of Proposition 5.1. We are actually going to prove the following four conclusions:

$$(C.1) \quad e_j^n = \sum_{i=0}^{n-1} \lambda C(\lambda) P^i(\lambda) u_{xxx}(x_j, t_{n-1-i}) h^3,$$

$$(C.2) \quad E_j^n = \sum_{i=0}^{n-1} 2\lambda C(\lambda) P^i(\lambda) (u(x_j, t_{n-1-i}) u_{xx}(x_j, t_{n-1-i}))_x h^3,$$

$$(C.3) \quad \Delta d_j^n = - \sum_{i=0}^{n-1} 12\lambda C(\lambda) P^i(\lambda) u_{xxx}(x_j, t_{n-1-i}) h,$$

and

$$(C.4) \quad \Delta D_j^n = - \sum_{i=0}^{n-1} 12\lambda C(\lambda) P^i(\lambda) (u(x_j, t_{n-1-i}) u_{xx}(x_j, t_{n-1-i}))_x h,$$

with $D_j^n = u(x_j, t_n) d_j^n$ as defined in Theorem 4.2. The first two of the estimations are the conclusions of the proposition. We are going to use the induction arguments to prove them.

We first need to verify the conclusions at $n = 1$. Note that initially we have $e_j^0 = 0$ and $E_j^0 = 0$, and thus $d_j^0 = 0$ from (4.8); therefore, at $n = 1$ we have from (4.6) and (4.7) in Theorem 4.1 and (4.24) and (4.25) in Theorem 4.2 and the assumptions of the proposition

$$(C.5) \quad e_j^1 = \lambda C(\lambda) u_{xxx}(x_j, t_0) h^3,$$

$$(C.6) \quad E_j^1 = 2\lambda C(\lambda) (u(x_j, t_0) u_{xx}(x_j, t_0))_x h^3,$$

$$(C.7) \quad C(\lambda) u_{xxx}(x_j, t_0) h,$$

and

$$(C.8) \quad \Delta D_j^1 = -12\lambda C(\lambda) (u(x_j, t_0) u_{xx}(x_j, t_0))_x h.$$

This verifies the conclusions (C.1), (C.2), (C.3) and (C.4) at $n = 1$.

We assume that the conclusions (C.1), (C.2), (C.3) and (C.4) are true at $n = k$, i.e.,

$$(C.9) \quad e_j^k = \sum_{i=0}^{k-1} \lambda C(\lambda) P^i(\lambda) u_{xxx}(x_j, t_{k-1-i}) h^3,$$

$$(C.10) \quad E_j^k = \sum_{i=0}^{k-1} 2\lambda C(\lambda) P^i(\lambda) (u(x_j, t_{k-1-i}) u_{xx}(x_j, t_{k-1-i}))_x h^3,$$

$$(C.11) \quad \Delta d_j^k = - \sum_{i=0}^{k-1} 12\lambda C(\lambda) P^i(\lambda) u_{xxx}(x_j, t_{k-1-i}) h,$$

and

$$(C.12) \quad \Delta D_j^k = - \sum_{i=0}^{k-1} 12\lambda C(\lambda) P^i(\lambda) (u(x_j, t_{k-1-i}) u_{xx}(x_j, t_{k-1-i}))_x h.$$

Now we are going to prove that the conclusions are also true at $n = k + 1$ under the assumptions of the proposition. We first prove the conclusions (C.1) and (C.2) at $n = k + 1$. From (4.6) and (4.7) in Theorem 4.1, we have

$$(C.13) \quad e_j^{k+1} = e_j^k + \lambda C(\lambda) u_{xxx}(x_j, t_k) h^3 + \lambda D(\lambda) \Delta d_j^k h^2,$$

$$(C.14) \quad \begin{aligned} E_j^{k+1} &= E_j^k + 2\lambda C(\lambda) (u(x_j, t_k) u_{xx}(x_j, t_k))_x h^3 \\ &\quad + 2\lambda D(\lambda) \Delta (u(x_j, t_k) d_j^k) h^2. \end{aligned}$$

We substitute (C.9) and (C.11) into (C.13) and obtain

$$(C.15) \quad \begin{aligned} e_j^{k+1} &= \sum_{i=0}^{k-1} \lambda C(\lambda) P^i(\lambda) u_{xxx}(x_j, t_{k-1-i}) h^3 + \lambda C(\lambda) u_{xxx}(x_j, t_k) h^3 \\ &\quad - \sum_{i=0}^{k-1} 12\lambda D(\lambda) \lambda C(\lambda) P^i(\lambda) u_{xxx}(x_j, t_{k-1-i}) h^3 \\ &= \sum_{i=0}^{k-1} (1 - 12\lambda D(\lambda)) \lambda C(\lambda) P^i(\lambda) u_{xxx}(x_j, t_{k-1-i}) h^3 \\ &\quad + \lambda C(\lambda) u_{xxx}(x_j, t_k) h^3. \end{aligned}$$

Note in (C.15) that $(1 - 12\lambda D(\lambda)) = P(\lambda)$; therefore, we obtain from (C.15),

$$(C.16) \quad e_j^{k+1} = \sum_{i=0}^k \lambda C(\lambda) P^i(\lambda) u_{xxx}(x_j, t_{k-i}) h^3,$$

which is the conclusion (C.1) at $n = k + 1$. By substituting (C.10) and (C.12) into (C.14) the conclusion (C.2) at $n = k + 1$,

$$(C.17) \quad E_j^{k+1} = \sum_{i=0}^k 2\lambda C(\lambda) P^i(\lambda) (u(x_j, t_{k-i}) u_{xx}(x_j, t_{k-i}))_x h^3,$$

can be proved in the same way.

Now we prove the conclusions (C.3) and (C.4) at $n = k + 1$. From (4.24) and (4.25) in Theorem 4.2 and the assumptions of the proposition we have

$$(C.18) \quad \Delta d_j^{k+1} = P(\lambda) \Delta d_j^k - 12\lambda C(\lambda) u_{xxx}(x_j, t_k) h,$$

$$(C.19) \quad \Delta D_j^{k+1} = P(\lambda) \Delta D_j^k - 12\lambda C(\lambda) (u(x_j, t_k) u_{xx}(x_j, t_k))_x h.$$

We substitute (C.11) into (C.18) and obtain

$$(C.20) \quad \begin{aligned} \Delta d_j^{k+1} &= -P(\lambda) \sum_{i=0}^{k-1} 12\lambda C(\lambda) P^i(\lambda) u_{xxx}(x_j, t_{k-1-i}) h - 12\lambda C(\lambda) u_{xxx}(x_j, t_k) h \\ &= -\sum_{i=0}^k 12\lambda C(\lambda) P^i(\lambda) u_{xxx}(x_j, t_{k-i}) h. \end{aligned}$$

In the same way, we substitute (C.12) into (C.19) and obtain

$$(C.21) \quad \Delta D_j^{k+1} = - \sum_{i=0}^k 12\lambda C(\lambda) P^i(\lambda) (u(x_j, t_{k-i}) u_{xx}(x_j, t_{k-i}))_x h.$$

Thus, the conclusions (C.3) and (C.4) at $n = k + 1$ are obtained and the proof of the proposition is complete. ♠

ACKNOWLEDGEMENT

The authors want to thank the referees for their valuable comments and suggestions.

REFERENCES

- [1] U. Ascher and R. McLachlan, *Multisymplectic box schemes and the Korteweg-de Vries equation*, Appl. Numer. Math., **48**, (2004), pp. 255-269. MR2056917
- [2] U. Ascher and R. McLachlan, *On symplectic and multisymplectic schemes for the KdV equation*, J. Sci. Comput., **25**, (2005), pp. 83-104. MR2231944 (2007a:37107)
- [3] R. Chen, *Several numerical methods for hyperbolic conservation laws*, Doctoral thesis, No. 10280-06810055, Shanghai University (in Chinese).
- [4] B. Cockburn and C. Shu, *TVD Runge-Kutta local projection Galerkin finite element methods for conservation laws III: One-dimensional systems*, J. Comput. Phys., **84** (1989), pp. 90-113. MR1015355 (90k:65161)
- [5] Y. Cui and D. Mao, *Numerical method satisfying the first two conservation laws for the Korteweg-de Vries equation*, J. Comput. Phys., **227** (2007), pp. 376-399. MR2361527 (2008i:65172)
- [6] B. Després, and F. Lagoutière, *Contact discontinuity capturing schemes for linear advection and compressible gas dynamics*, J. Sci. Comput., **16**, (2001), pp. 479-524. MR1881855 (2002k:65121)
- [7] A. Harten, *High resolution schemes for hyperbolic conservation laws*, J. Comput. Phys., **49** (1983), pp. 357-393. MR701178 (84g:65115)
- [8] A. Harten, B. Engquist, S. Osher, and S. R. Chakravarthy, *Uniformly high order accurate essentially non-oscillatory schemes, III*, J. Comput. Phys., **71**, (1987), pp. 231-303. MR897244 (90a:65199)
- [9] Guo Benyu, *The finite difference methods for Partial Differential Equations*, Science Publication, 1988 (in Chinese).
- [10] G. Jiang and C. Shu, *Efficient Implementation of Weighted ENO Schemes*, J. Comput. Phys. **126** (1996), pp. 202-228. MR1391627 (97e:65081)
- [11] R. J. LeVeque, *Finite volume methods for hyperbolic problems*, Cambridge University Press, 2002. MR1925043 (2003h:65001)
- [12] R. J. LeVeque, *Numerical methods for conservation laws*, Birkhäuser Verlag, Basel, Boston, Berlin, 1990. MR1077828 (91j:65142)
- [13] H. Li, *Entropy dissipating scheme for hyperbolic system of conservation laws in one space dimension*, Doctoral thesis, No. 11903-02820022, Shanghai University (in Chinese).
- [14] H. Li, *Second-order entropy dissipation scheme for scalar conservation laws in one space dimension*, Master's thesis, No. 11903-99118086, Shanghai University (in Chinese).
- [15] H. Li and D. Mao, *The design of the entropy dissipator of the entropy dissipating scheme for scalar conservation law*, Chinese J. Comput. Phys., **21**, (2004), pp. 319-326 (in Chinese).
- [16] H. Li, Z. Wang and D. Mao, *Numerically neither dissipative nor compressive scheme for linear advection equation and its application to the Euler system*, J. Sci. Comput., **36** (2008), pp. 285-331. MR2434848 (2009h:65115)
- [17] K. W. Morton and D. F. Mayers, *Numerical solution of partial differential equations*, Cambridge University Press, Britain, 2005. MR2153063 (2006a:65003)
- [18] R.D. Richtmyer and K.W. Morton, *Difference Methods for Initial Value Problems*, second edition. New York, Wiley-Interscience Reprinted, New York, Kreiger, 1994. MR1275838 (95b:65003)

- [19] P.L. Roe, *Some contribution to the modelling of discontinuous flows*, Lectures in Appl. Math. **22**, (1985), pp. 163-193. MR818787 (87m:65149)
- [20] Z. Wang, *Finitie difference Scemes satisfying multiconservation laws for linear advection equations*, Master's thesis, No. 11903-99118086, Shanghai University (in Chinese).
- [21] Z. Wang and D. Mao, *Conservative difference scheme satisfying three conservation laws for linear advection equation*, J. SHU, **12** (2006), pp. 588-592, (in Chinese). MR2281178

SHANGHAI INSTITUTE OF APPLIED MATHEMATICS AND MECHANICS, SHANGHAI, 200072, PEOPLE'S REPUBLIC OF CHINA

Current address: Department of Mathematics, Shanghai University, Shanghai, 200444, People's Republic of China

DEPARTMENT OF MATHEMATICS, SHANGHAI UNIVERSITY, SHANGHAI, 200444, PEOPLE'S REPUBLIC OF CHINA