The high order local discontinuous Galerkin schemes with variable-coefficient explicit-implicit-null time-marching for diffusion and dispersion equations

Meiqi Tan[1], Juan Cheng[2], and Chi-Wang Shu[3]

Abstract

In this paper, we discuss a third order variable-coefficient explicit-implicit-null (VC-EIN) time-marching method coupled with the local discontinuous Galerkin (LDG) methods for the diffusion and dispersion equations, respectively. The basic idea of the VC-EIN method is to add and subtract an appropriately large linear term with variable coefficient at one side of the considered equation, and then apply the standard implicit-explicit time-marching method to the equivalent equation. The numerical experiments show that the proposed schemes are stable and can achieve optimal orders of accuracy for both one-dimensional and two-dimensional quasi-linear and nonlinear equations. In addition, the performance of the VC-EIN method is compared with the constant-coefficient explicit-implicit-null (CC-EIN) method [16] of the same order. When the diffusion coefficient or the dispersion coefficient has a few high and narrow bumps and the bumps only account for a small part of the whole computational domain, the numerical results show that the VC-EIN method is vastly more accurate than the CC-EIN method, if proper stabilization parameters are chosen.

Keywords: diffusion equation, dispersion equation, stability, explicit-implicit-null time discretization, local discontinuous Galerkin method.

[1]Graduate School, China Academy of Engineering Physics, Beijing 100088, China. E-mail: tanmeiqi20@gscaep.ac.cn.

[2]Corresponding author. Laboratory of Computational Physics, Institute of Applied Physics and Computational Mathematics, Beijing 100088, China and HEDPS, Center for Applied Physics and Technology, and College of Engineering, Peking University, Beijing 100871, China. E-mail: cheng_juan@iapcm.ac.cn. Research is supported in part by NSFC grant 12031001, and National Key R&D Program of China No. 2022YFA1004500.

[3]Division of Applied Mathematics, Brown University, Providence, RI 02912. E-mail: chi-wang_shu@brown.edu. Research is supported in part by NSF grant DMS-2010107.

1 Introduction

In this paper, we exploit a third order variable-coefficient explicit-implicit-null (VC-EIN) time-marching method coupled with the local discontinuous Galerkin (LDG) methods for the diffusion and dispersion equations, respectively. For the simplification of notations, the equations described below are only one-dimensional. Note that the conclusions given in this paper can also be extended to equations in two dimensions, as we shall present in the numerical experiment section.

The diffusion equation

$$U_t = (d(U)U_x)_x, \tag{1.1}$$

where the diffusion coefficient $d(U) \geq 0$ is smooth and bounded, has been widely used to model various processes in engineering and industry, such as the thermo-chemical diffusion process of carburizing and nitriding [4], the miscible displacement in porous media [23] and so on. In this paper, we use the capital letter U to denote the exact solution to the considered equation.

The dispersion equation

$$U_t + g(U_x)_{xx} = 0 \tag{1.2}$$

where $g(U_x)$ is an arbitrary (smooth) function, is a special KdV-type equation typified by the Korteweg-de Vries (KdV) equation [13] and its generalizations. The KdV-type equations have been widely used to describe the propagation of waves in a variety of nonlinear dispersive media and appear often in applications. Applications include the study of waves in plasma physics, internal waves in coastal waters, flow in blood vessels and so on. For more details, we refer the readers to [3] and the references therein.

For the diffusion and dispersion equations, the LDG method, which was first introduced by Cockburn and Shu [5] for the convection-diffusion equations, has been popular. The main idea of the LDG method is to rewrite the equation with higher order derivatives into an equivalent first order system, and then apply the discontinuous Galerkin method [6] to the

system. A key ingredient for the success of the LDG method is the design of the numerical fluxes, which must be determined carefully to guarantee stability and local solvability of all the auxiliary variables introduced to approximate the derivatives of the solution. The LDG method can easily handle meshes with hanging nodes, elements of general shapes and local spaces of different types, thus it is flexible for *hp*-adaptivity. So far, the LDG techniques have already been developed for various high order equations [19–22].

The explicit-implicit-null (EIN) method, which was first introduced by Douglas and Dupont [7], has excellent stability property and great flexibility in dealing with stiff nonlinear problems. The basic idea of the EIN method is to add and subtract an appropriately large linear term at one side of the equation, and then apply the standard implicit-explicit (IMEX) method to the equivalent equation. The crucial step of the method consists in adding and subtracting the right term, of which the choice is quite flexible, but needs to have the same short-wavelength scaling as the stiffest part of the original equation. So far, the EIN method has been analyzed and implemented by a number of authors on a case-by-case basis. For further information, we refer the readers to [7–11, 15, 16, 18] (the list is far from being exhaustive), which encompass analyses and applications of the method in, for example, the problem of flow by mean curvature and surface diffusion, the Boltzmann kinetic equations and related problems with nonlinear stiff sources, etc.

The general practice of the EIN method is to add and subtract an appropriately large linear stiffest term with constant coefficient at one side of the equation, and then apply the IMEX time-marching method to the equivalent equation. We call such method the constant-coefficient explicit-implicit-null method in this paper and hereafter denote it by "CC-EIN". In the following, we take the diffusion equation (1.1) as an example to introduce the method in detail. Since the equation contains a stiff part corresponding to the second derivative, in order to stabilize the equation, a second derivative term with constant coefficient $b_1 U_{xx}$ is

added to and subtracted from the left-hand side of the considered equation

$$U_t + \underbrace{b_1 U_{xx} - (d(U)U_x)_x}_{T_1} - \underbrace{b_1 U_{xx}}_{T_2} = 0, \qquad b_1 = a_0 \times \max d(u^n), \tag{1.3}$$

where u^n represents the numerical solution at the n-th time level. Here, a_0 is a large enough constant such that T_1 is either not stiff, or less stiff and less dissipative compared to T_2, thus it can be treated explicitly, and T_2 is stiff and dissipative, thus will be discretized implicitly. The CC-EIN method so designed gives rise to a linear system, for which very efficient solution methods exist. In addition, the severe time step restriction imposed by the explicit treatment of the nonlinear stiff term $(d(U)U_x)_x$ can be removed. Since the auxiliary term $b_1 U_{xx}$ added to and subtracted from the equation are treated in different ways, i.e., one is treated explicitly and the other is treated implicitly, for a p-th order CC-EIN method, stabilization is achieved essentially at the cost of a somewhat additional temporal truncation error of order $O(\tau^p)$, where τ is the time step. In addition, when the dominant error is due to the time-stepping or the spatial accuracy is of order at least as high as the time-marching method, the numerical results in [16] show that the overall error will increase significantly with the increase of b_1. When the diffusion coefficient $d(U)$ has a few high and narrow bumps and the bumps only account for a small part of the whole computational domain, the CC-EIN method does not seem to be cost-effective. To resolve this issue, the VC-EIN method, which has not been studied before to our knowledge, is proposed and studied in this paper.

The VC-EIN method is obtained by adding and subtracting a more accurate approximation of the stiffest term at one side of the equation, and then treat them separately. In order to apply the VC-EIN idea, we still take the diffusion equation (1.1) as an example to introduce the approach in detail. From the stability and conservation points of view, we add and subtract a second derivative term with variable coefficient $(a_1(x)U_x)_x$ at the left-hand side of the considered equation

$$U_t - \underbrace{(d(U)U_x - a_1(x)U_x)_x}_{T_1} - \underbrace{(a_1(x)U_x)_x}_{T_2} = 0, \tag{1.4}$$

where

$$a_1(x) = a_0 \times \max_{t^n \le t \le t^{n+1}} d(u(x,t)),$$

and then treat the damping term T_2 implicitly and the remaining term T_1 explicitly. Note that this approach is not equivalent to a linearization of the problem [2, 12], and to some extent, it is much simpler to apply. If the diffusion coefficient varies in space (quasi-linear case), it is easy to obtain $a_1(x)$. If the diffusion coefficient depends on the solution (nonlinear case), as an alternative, the approach adopted in this paper is to obtain its approximation: $\tilde{a}_1(x)$ through the convolution technique. Given a unit-mass kernel

$$\Phi(x) = \frac{1}{\int_{-1}^{1} e^{\frac{1}{|x|^2-1}} \mathrm{d}x} \begin{cases} e^{\frac{1}{|x|^2-1}}, & |x| < 1, \\ 0, & |x| \ge 1, \end{cases}$$

we form a dilated mollifier

$$\Phi_{C_0,\delta}(x) := \frac{C_0}{\delta} \Phi\left(\frac{x}{\delta}\right) \tag{1.5}$$

with C_0, δ being two free dilation parameters at our disposal. By tuning δ we can adjust the support of $\Phi(x)$ over the symmetric interval $(-\delta, \delta)$. By adjusting C_0, we can always make

$$\tilde{a}_1(x) = a_0 \times \tilde{d}(x) \ge a_1(x) = a_0 \times \max_{t^n \le t \le t^{n+1}} d(u(x,t)), \tag{1.6}$$

where

$$\tilde{d}(x) = \int d(u(y, t^n)) \Phi_{C_0,\delta}(x - y) \, \mathrm{d}y.$$

Numerical experiments show that the VC-EIN method with variable coefficient $\tilde{a}_1(x)$ perform well in nonlinear tests if proper dilation parameters, δ and C_0, are chosen. Compared with the CC-EIN method, the additional error introduced by the VC-EIN method is smaller outside the bumps of $d(U)$. When the high and narrow bumps only account for a small part of the whole computational domain, the VC-EIN method will eventually lead to a smaller overall error at least for the L^1 and L^2 norms.

An outline of this paper is as follows. In Section 2, we present the semi-discrete LDG schemes and the time discretization method for the equations mentioned above. Section 3

shows a series of numerical tests to assess the stability and order accuracy of the proposed schemes for both one-dimensional and two-dimensional quasi-linear and nonlinear problems. In addition, we also carry out a comparative study of the performance of the VC-EIN-LDG scheme and the CC-EIN method with LDG spatial discretization (CC-EIN-LDG). When proper parameters: δ, C_0, a_0 are chosen, the numerical results show that the VC-EIN-LDG scheme is significantly more accurate than the CC-EIN-LDG scheme of the same order, if the diffusion coefficient $d(U)$ of the diffusion equation (1.1) or the dispersion coefficient $g'(U_x)$ of the dispersion equation (1.2) has a few high and narrow bumps and the bumps only account for a small part of the whole computational domain. Finally, the concluding remarks are given in Section 4.

2 The numerical schemes

In this section, we will present the semi-discrete LDG schemes and the time discretization method used in this paper. For simplicity of presentation, some preliminary notations are presented here which will be used throughout this section. Let $\mathcal{T}_h = \{I_j = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]\}_{j=1}^{N}$ be a uniform partition of the computational domain $\Omega = [x_L, x_R]$, where $x_{\frac{1}{2}} = x_L$ and $x_{N+\frac{1}{2}} = x_R$ are the two boundary endpoints. Let $x_j = \frac{1}{2}(x_{j-\frac{1}{2}} + x_{j+\frac{1}{2}})$ denote the midpoint of the cell I_j. The spatial mesh size is $h = (x_R - x_L)/N$.

2.1 The LDG method for the diffusion equation

Let us describe in detail the implementation of the LDG method for the diffusion equation (1.4) subject to periodic boundary condition and the initial condition

$$U(x, 0) = U_0(x), \qquad x \in \Omega. \tag{2.1}$$

The assumption of the periodic boundary condition is for simplicity only and is not essential. The method can be easily designed for non-periodic boundary conditions [17]. Note that the LDG method described in this paper is a little different from the classical one [6], which

involves the square root of the diffusion coefficient $d(U)$. By introducing the new variables

$$P = (d(U) - a_1(x))R, \qquad Q = a_1(x)R, \qquad R = U_x,$$

we can reformulate (1.4) as the following first order system:

$$U_t - P_x - Q_x = 0, \qquad P - (d(U) - a_1(x))R = 0,$$
$$Q - a_1(x)R = 0, \qquad R - U_x = 0.$$

The LDG approximation to the above periodic initial value problem can be defined as follows: seek piecewise polynomial solutions u, p, q, r from V_h such that for all the test functions ϕ_1, ϕ_2, ϕ_3, $\phi_4 \in V_h$ and $1 \le j \le N$, we have

$$\int_{I_j} u_t\phi_1 \,\mathrm{d}x + \int_{I_j} (p+q)(\phi_1)_x \,\mathrm{d}x - (\hat{p}+\hat{q})_{j+\frac{1}{2}}(\phi_1)^-_{j+\frac{1}{2}} + (\hat{p}+\hat{q})_{j-\frac{1}{2}}(\phi_1)^+_{j-\frac{1}{2}} = 0, \tag{2.2a}$$

$$\int_{I_j} p\phi_2 \,\mathrm{d}x = \int_{I_j} (d(u) - a_1(x))r\phi_2 \,\mathrm{d}x, \tag{2.2b}$$

$$\int_{I_j} q\phi_3 \,\mathrm{d}x = \int_{I_j} a_1(x)r\phi_3 \,\mathrm{d}x, \tag{2.2c}$$

$$\int_{I_j} r\phi_4 \,\mathrm{d}x + \int_{I_j} u(\phi_4)_x \,\mathrm{d}x - \hat{u}_{j+\frac{1}{2}}(\phi_4)^-_{j+\frac{1}{2}} + \hat{u}_{j-\frac{1}{2}}(\phi_4)^+_{j-\frac{1}{2}} = 0 \tag{2.2d}$$

and

$$u(x,0) = \mathbb{P}_h U_0(x), \tag{2.3}$$

where

$$V_h = \{v \in L^2(\Omega) : v|_{I_j} \in \mathcal{P}_m(I_j),\ \forall j = 1, ..., N\} \tag{2.4}$$

and $\mathbb{P}_h U_0(x)$ is the local L^2-projection of the initial condition $U_0(x)$ satisfying

$$\int_{I_j} \mathbb{P}_h U_0(x)\phi_1(x) \,\mathrm{d}x = \int_{I_j} U_0(x)\phi_1(x) \,\mathrm{d}x, \qquad \forall \phi_1(x) \in V_h. \tag{2.5}$$

Here, $\mathcal{P}_m(I_j)$ is the space of polynomials in I_j of degree no more than m. The functions in V_h are allowed to have discontinuities across element interfaces. For any piecewise function u in V_h, we denote by $u^-_{j+\frac{1}{2}}$ and $u^+_{j+\frac{1}{2}}$ the left and right limits of the discontinuous solution u at the interface $x_{j+\frac{1}{2}}$, respectively. The terms with the "hat" in the cell boundary terms

appearing from integration by parts are the so-called "numerical fluxes", which are single-valued functions defined on the edges and play important roles in ensuring stability of the LDG method. As shown in the appendix, we can prove a strong L^2-stability result if we adopt the following numerical fluxes:

$$\hat{p} = p^+, \qquad \hat{q} = q^+, \qquad \hat{u} = u^-, \tag{2.6}$$

where we have omitted the subscripts $j \pm \frac{1}{2}$ in the definition of the fluxes, as all quantities are evaluated at the interfaces $x_{j\pm\frac{1}{2}}$. We remark that the choice of the fluxes (2.6) is not unique. In fact the crucial part is taking $\hat{p}$ and $\hat{q}$ from the same side and taking $\hat{u}$ and $\hat{p}$ from opposite sides (alternating fluxes). We only need to replace the above-mentioned $a_1(x)$ with b_1 to obtain the LDG spatial discretization of the equation (1.3).

2.2 The LDG method for the dispersion equation

Since the dispersion equation (1.2) contains a stiff part corresponding to the third derivative, in order to stabilize the equation without sacrificing the conservation of the numerical solution, we add the same term $(a_1(x)U_x)_{xx}$, $a_1(x) \geq 0$ to both sides of the equation. It should be noted that the sign of the auxiliary term $(a_1(x)U_x)_{xx}$ we add to both sides of the equation needs to be adjusted according to the sign of $g'(U_x)$. If $g'(U_x) \geq 0$ within the whole computational domain Ω, then we should add two equal term with negative variable coefficient $-(a_1(x)U_x)_{xx}$ to both sides of the considered equation. Otherwise, the sign of the auxiliary term $(a_1(x)U_x)_{xx}$ needs to be positive. We only consider the case where the sign of $g'(U_x)$ is fixed. The discussion of the dispersion equation with the sign of $g'(U_x)$ varying in space and time goes beyond the scope of the present paper and is by itself an interesting topic for future investigation. Assuming that $g'(U_x) \geq 0$, we add two equal dispersion term with variable coefficient $-(a_1(x)U_x)_{xx}$ to both sides of the equation and get

$$U_t + \underbrace{(g(U_x) - a_1(x)U_x)_{xx}}_{T_1} = -\underbrace{(a_1(x)U_x)_{xx}}_{T_2}, \tag{2.7}$$

where

$$a_1(x) = a_0 \times \max_{t^n \le t \le t^{n+1}} g'(u_x(x,t))$$

and $a_0 > 0$ is a large enough constant yet to be determined. We begin with the equation (2.7) to describe the LDG method. For a detailed introduction of the method, we refer the readers to [21].

By introducing the new variables

$$V = (g(Z) - a_1(x)Z)_x, \qquad W = (a_1(x)Z)_x, \qquad Z = U_x,$$

we can rewrite (2.7) into the following first order system:

$$\begin{aligned} U_t + V_x + W_x &= 0, & V - (g(Z) - a_1(x)Z)_x &= 0, \\ W - (a_1(x)Z)_x &= 0, & Z - U_x &= 0. \end{aligned}$$

The semi-discrete LDG approximation to the dispersion equation (2.7) with the initial condition (2.1) and periodic boundary condition can be defined as follows: seek piecewise polynomial solutions u, v, w, z from V_h such that for all the test functions ϕ_1, ϕ_2, ϕ_3, $\phi_4 \in V_h$ and $1 \le j \le N$, we have

$$\int_{I_j} u_t \phi_1 \,\mathrm{d}x - \int_{I_j} (v+w)(\phi_1)_x \,\mathrm{d}x + (\hat{v}+\hat{w})_{j+\frac{1}{2}}(\phi_1)^-_{j+\frac{1}{2}} - (\hat{v}+\hat{w})_{j-\frac{1}{2}}(\phi_1)^+_{j-\frac{1}{2}} = 0, \tag{2.8a}$$

$$\begin{aligned} &\int_{I_j} v\phi_2 \,\mathrm{d}x + \int_{I_j} (g(z) - a_1(x)z)(\phi_2)_x \,\mathrm{d}x - \\ &(\hat{g} - \widehat{a_1 z})_{j+\frac{1}{2}}(\phi_2)^-_{j+\frac{1}{2}} + (\hat{g} - \widehat{a_1 z})_{j-\frac{1}{2}}(\phi_2)^+_{j-\frac{1}{2}} = 0, \end{aligned} \tag{2.8b}$$

$$\int_{I_j} w\phi_3 \,\mathrm{d}x + \int_{I_j} a_1(x) z(\phi_3)_x \,\mathrm{d}x - (\widehat{a_1 z})_{j+\frac{1}{2}}(\phi_3)^-_{j+\frac{1}{2}} + (\widehat{a_1 z})_{j-\frac{1}{2}}(\phi_3)^+_{j-\frac{1}{2}} = 0, \tag{2.8c}$$

$$\int_{I_j} z\phi_4 \,\mathrm{d}x + \int_{I_j} u(\phi_4)_x \,\mathrm{d}x - \hat{u}_{j+\frac{1}{2}}(\phi_4)^-_{j+\frac{1}{2}} + \hat{u}_{j-\frac{1}{2}}(\phi_4)^+_{j-\frac{1}{2}} = 0 \tag{2.8d}$$

and the equality (2.3) established. Here, V_h is defined by (2.4). As shown in [19, 21], we can prove an L^2-stability result, a cell entropy inequality for a more general case and obtain optimal error estimates for the linear case, if we adopt the following numerical fluxes:

$$\hat{v} = v^+, \qquad \hat{w} = w^+, \qquad \hat{u} = u^-, \qquad \hat{g} = g(z^+), \qquad \widehat{a_1 z} = (a_1 z)^+. \tag{2.9}$$

Notice that the choice of the numerical fluxes $\hat{g}$ and $\widehat{a_1 z}$ is based on the assumption that $g'(U_x) \geq 0$. If $g'(U_x) \leq 0$, we should take $\hat{g} = g(z^-)$, $\widehat{a_1 z} = (a_1 z)^-$. In addition, the assumption of the periodic boundary condition is for simplicity only. For other boundary conditions, we refer the readers to [14] and the references therein for the setting of the numerical fluxes. Similarly, we only need to replace the above $a_1(x)$ with b_1 to obtain the LDG spatial discretization of the equation

$$U_t + \underbrace{(g(U_x) - b_1 U_x)_{xx}}_{T_1} = \underbrace{-b_1 U_{xxx}}_{T_2},$$

where $b_1 = a_0 \times \max g'(u_x(x, t^n))$ and $g'(U_x) \geq 0$.

2.3 The explicit-implicit time discretization

The semi-discrete LDG schemes can be rewritten into the first-order ODE system

$$\frac{\mathrm{d}u}{\mathrm{d}t} = \mathcal{L}(t, u) + \mathcal{N}(t, u),$$

where $\mathcal{L}(t, u)$ arises from the spatial discretization of T_2 and will be treated implicitly, and $\mathcal{N}(t, u)$ is derived from the spatial discretization of T_1 and will be dealt with an explicit way. A third order IMEX Runge-Kutta (IMEX-RK) method will be considered in this paper. Given the numerical solution at time t^n, the IMEX-RK method forms 5 intermediate values $u^{n,s}$, $1 \leq s \leq 5$ according to

$$u^{n,s} = u^n + \tau \sum_{l=1}^{s} a_{sl} \mathcal{L}(t_l^n, u^{n,l}) + \tau \sum_{l=1}^{s-1} \hat{a}_{sl} \mathcal{N}(t_l^n, u^{n,l}), \tag{2.10a}$$

from which the approximation at time level t^{n+1} is assembled by

$$u^{n+1} = u^n + \tau \sum_{l=1}^{5} b_l \mathcal{L}(t_l^n, u^{n,l}) + \tau \sum_{l=1}^{5} \hat{b}_l \mathcal{N}(t_l^n, u^{n,l}), \tag{2.10b}$$

where the intermediate values $u^{n,s}$ are approximations to $u(x, t_l^n)$ and

$$t_l^n = t^n + \hat{c}_l \tau, \qquad \hat{c}_s = \sum_{l=1}^{s} a_{sl} = \sum_{l=1}^{s} \hat{a}_{sl}.$$

The IMEX-RK method can be represented by the following Butcher tableau:

$$\begin{array}{c|ccccc|ccccc|c}
 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \\
 & 0 & \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & \\
a_{sl} & 0 & \frac{1}{6} & \frac{1}{2} & 0 & 0 & \frac{11}{18} & \frac{1}{18} & 0 & 0 & 0 & \hat{a}_{sl} \\
 & 0 & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 0 & \frac{5}{6} & -\frac{5}{6} & \frac{1}{2} & 0 & 0 & \\
 & 0 & \frac{3}{2} & -\frac{3}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{4} & \frac{7}{4} & \frac{3}{4} & -\frac{7}{4} & 0 & \\
\hline
b_l & 0 & \frac{3}{2} & -\frac{3}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{4} & \frac{7}{4} & \frac{3}{4} & -\frac{7}{4} & 0 & \hat{b}_l
\end{array} \tag{2.11}$$

of which the left half lists a_{sl} and b_l, with the five rows from top to bottom corresponding to $s = 1, ..., 5$, and the columns from left to right corresponding to $l = 1, ..., 5$. Similarly, the right half lists $\hat{a}_{sl}$ and $\hat{b}_l$. With the above Butcher coefficients, we then arrive at a third order IMEX-RK method. The IMEX-RK method we consider is a combination of a four-stage, third order, L-stable, stiffly-accurate, singly diagonally implicit Runge-Kutta method and a four-stage, third order explicit Runge-Kutta method. For more details on the method, we refer to [1]. We have also considered other IMEX methods, but we will not state them here to save space.

3 Numerical experiments

Note that for the following simplified linear equations with periodic boundary conditions:

- The linear diffusion equation

$$U_t = dU_{xx}, \tag{3.1}$$

- The linear dispersion equation

$$U_t + dU_{xxx} = 0, \tag{3.2}$$

where $d > 0$ are two constants, the EIN-LDG schemes are shown to be unconditionally stable [16] provided $a_0 \geq 0.54$ regardless of the order accuracy of the LDG spatial discretizations. Here, we use the notation EIN-LDG to refer to both the VC-EIN-LDG and the CC-EIN-LDG schemes. After all, for linear equations with constant coefficients, these two schemes

are equivalent. Even though the analysis is only performed on the simplified linear equations with constant coefficients, numerical experiments [16] show that the CC-EIN-LDG scheme is unconditionally stable for the diffusion equation (1.1) if

$$b_1 \geq 0.54 \times d(u^n)$$

and for the dispersion equation (1.2) if

$$b_1 \geq 0.54 \times g'(u_x^n).$$

Based on the stability results of the CC-EIN-LDG schemes, we propose a guidance for the choice of $a_1(x)$ for the VC-EIN-LDG scheme for the diffusion equation (1.1), i.e.

$$a_1(x) = a_0 \times \max_{t^n \leq t \leq t^{n+1}} d(u(x,t)), \qquad a_0 \geq 0.54, \tag{3.3}$$

and for the dispersion equation (1.2), i.e.

$$a_1(x) = a_0 \times \max_{t^n \leq t \leq t^{n+1}} g'(u_x(x,t)), \qquad a_0 \geq 0.54. \tag{3.4}$$

Below, we will present a series of numerical tests aiming to show the order of accuracy and stability of the VC-EIN-LDG schemes for the diffusion and dispersion equations, respectively. In addition, the performance of the VC-EIN-LDG scheme is compared with the CC-EIN-LDG scheme of the same order. The equations with periodic boundary conditions will be considered, unless otherwise stated. Since time discretization is limited to only third order accuracy, we only concentrate on the piecewise quadratic polynomial ($m = 2$) case for the LDG spatial discretization in the numerical experiments. In addition, we take $\tau = h$ in the tests, such that the orders accuracy of errors in space and time match.

3.1 The diffusion equations

In this subsection, we would like to test the performance and stability of the proposed schemes for the diffusion equations in one and two space dimensions. Following the lines in [5], it will be straightforward to generalize the LDG scheme (2.2) for Cartesian meshes in the two-dimensional case.

3.1.1 The quasi-linear numerical test in one dimension

We consider the quasi-linear diffusion equation

$$U_t - \underbrace{(d(x,t)U_x)_x - (a_1(x)U_x)_x - s(x,t)}_{T_1} - \underbrace{(a_1(x)U_x)_x}_{T_2} = 0, \qquad x \in \left(-\frac{3}{2}\pi, \frac{1}{2}\pi\right) \tag{3.5}$$

augmented with the diffusion coefficient

$$d(x,t) = \alpha + \beta \tanh(\sigma \cos(\eta(x+t)))$$

the initial condition $U(x,0) = \sin(x+t)$ and the source term

$$\begin{aligned} s(x,t) =& \cos(x+t) + \beta\eta\sigma \cos(x+t)\,\mathrm{sech}^2(\sigma\cos(\eta(x+t)))\sin(\eta(x+t)) + \\ & \sin(x+t)(\alpha + \beta\tanh(\sigma\cos(\eta(x+t)))). \end{aligned}$$

The problem has an exact solution

$$U(x,t) = \sin(x+t). \tag{3.6}$$

Indeed, the standard IMEX methods can be directly adopted to solve the problem. To illustrate the necessity of the stability condition (3.3), in the test, we take $a_1(x)$ as $0.54 \times d(x,t^n)$, $0.53 \times \max_{t^n \le t \le t^{n+1}} d(x,t)$, $0.54 \times \max_{t^n \le t \le t^{n+1}} d(x,t)$ and $\max_{t^n \le t \le t^{n+1}} d(x,t)$, respectively. It is worth pointing out that it is not necessary to get the specific expression of $\max_{t^n \le t \le t^{n+1}} d(x,t)$. In the LDG scheme (2.2), these two formulas, (2.2b) and (2.2c), involve $a_1(x) = a_0 \times \max_{t^n \le t \le t^{n+1}} d(x,t)$ and are integral expressions, which will be solved by a high order numerical integration method in practical computing. Thus we only need to consider the maximum value of $d(x,t)$ at a series of discrete time points from t^n to t^{n+1} for each numerical integration point and this is easy to achieve. We compute to $T = \pi$ with the parameters $\alpha = 11$, $\beta = 10$, $\sigma = 3\pi$, $\eta = 1$. Note that with those parameters, the diffusion coefficient $d(x,t)$ is not a monotonic function of time t for any fixed x. In other words, $d(x,t^n) \neq \max_{t^n \le t \le t^{n+1}} d(x,t)$. The numerical errors and orders of accuracy are listed in Table 3.1. Intuitively speaking, since u^{n+1} involves multiple intermediate values $u^{n,s}, 2 \le s \le 5$,

as shown in (2.10), $a_1(x) < 0.54 \times \max_{t^n \le t \le t^{n+1}} d(x,t)$ is not large enough to remove the stiffness in all $\mathcal{N}(t_s^n, u^{n,s}), 2 \le s \le 5$ when $d(x,t)$ changes sharply with respect to time t and the mesh division is coarse, unless we use a small enough time step to temporally resolve the rapid transient of $d(x,t)$. Thus, from Table 3.1 we can see that the VC-EIN-LDG scheme is unstable or a sufficiently dense mesh grid is required to maintain the stability of the scheme if $a_1(x) < 0.54 \times \max_{t^n \le t \le t^{n+1}} d(x,t)$. When $a_1(x) \ge 0.54 \times \max_{t^n \le t \le t^{n+1}} d(x,t)$, the scheme is stable and the numerical orders of the scheme settle down towards the asymptotic value slowly. In fact, it is reasonable for this to happen, because the auxiliary term $(a_1(x)U_x)_x$ we add to and subtract from the equation are treated in different ways, i.e., one is treated explicitly and the other is treated implicitly. The two different time-stepping methods bring a certain error to the scheme, which increases with the increase of $a_1(x)$ and slows down the convergence of the scheme to the optimal order to some extent. The above explanation can also be used to interpret the similar convergence behavior of the VC-EIN-LDG and CC-EIN-LDG schemes for nonlinear equations.

Next, to demonstrate the optimal order of accuracy of the scheme, we consider again the quasi-linear numerical test proposed before. The simulation is run from $t = 0$ to $T = \pi$ with $a_1(x) = 0.54 \times \max_{t^n \le t \le t^{n+1}} d(x,t)$ and the parameters $\alpha = 1$, $\beta = \frac{1}{2}$, $\sigma = 1$, $\eta = 1$. We list the errors and the experimental orders of the VC-EIN-LDG scheme in Table 3.2, from which we can observe a rate of convergence about 3 for L^1, L^∞ and L^2 norms.

3.1.2 The nonlinear numerical test in one dimension

We consider the nonlinear diffusion equation

$$U_t = (d(U)U_x)_x + s(x,t) \tag{3.7}$$

augmented with the diffusion coefficient

$$d(U) = \alpha + \beta U^2 \tag{3.8}$$

Table 3.1: The errors and orders of the VC-EIN-LDG scheme for Example (3.5) with $\alpha = 11$, $\beta = 10$, $\sigma = 3\pi$, $\eta = 1$.

$a_1(x)$	N	L^1 error	order	L^∞ error	order	L^2 error	order
$0.54 \times d(x, t^n)$	64	2.12E+18		5.65E+19		7.22E+18	
	128	4.04E+21	-10.90	9.98E+22	-10.79	1.50E+22	-11.02
	256	2.02E+19	7.64	9.10E+20	6.78	1.00E+20	7.22
	512	2.54E+07	39.53	1.34E+09	39.31	1.37E+08	39.42
	1024	4.19E-06	42.46	2.83E-05	45.43	7.50E-06	44.05
$0.53 \times \max_{t^n \le t \le t^{n+1}} d(x,t)$	64	2.40E-03		1.78E-02		3.61E-03	
	128	1.22E-03	0.97	1.92E-02	-0.11	2.77E-03	0.38
	256	6.88E-03	-2.49	2.84E-01	-3.89	3.21E-02	-3.53
	512	9.49E+02	-17.07	7.64E+04	-18.04	6.25E+03	-17.57
	1024	1.90E+16	-44.19	3.20E+18	-45.25	1.32E+17	-44.26
$0.54 \times \max_{t^n \le t \le t^{n+1}} d(x,t)$	64	1.69E-03		7.42E-03		2.10E-03	
	128	4.62E-04	1.87	2.23E-03	1.73	6.24E-04	1.75
	256	1.04E-04	2.16	5.84E-04	1.93	1.49E-04	2.06
	512	2.12E-05	2.29	1.38E-04	2.08	3.39E-05	2.14
	1024	4.32E-06	2.29	3.27E-05	2.07	7.76E-06	2.13
$\max_{t^n \le t \le t^{n+1}} d(x,t)$	64	2.44E-03		6.98E-03		3.21E-03	
	128	6.42E-04	1.92	2.27E-03	1.62	9.06E-04	1.83
	256	1.55E-04	2.05	6.96E-04	1.71	2.28E-04	1.99
	512	3.43E-05	2.17	1.96E-04	1.83	5.46E-05	2.06
	1024	7.28E-06	2.24	5.22E-05	1.91	1.28E-05	2.09

Table 3.2: The errors and orders of the VC-EIN-LDG scheme for Example (3.5) with $\alpha = 1$, $\beta = \frac{1}{2}$, $\sigma = 1$, $\eta = 1$.

$a_1(x)$	N	L^1 error	order	L^∞ error	order	L^2 error	order
$0.54 \times \max_{t^n \le t \le t^{n+1}} d(x,t)$	64	1.70E-05		4.07E-05		2.10E-05	
	128	2.33E-06	2.86	5.77E-06	2.82	2.92E-06	2.85
	256	3.10E-07	2.91	7.92E-07	2.86	3.90E-07	2.90
	512	4.01E-08	2.95	1.05E-07	2.92	5.06E-08	2.95
	1024	5.10E-09	2.97	1.36E-08	2.95	6.46E-09	2.97

and the exact solution

$$U(x,t) = (\gamma + \lambda e^{-\sigma^2(1-\cos(x+t))})(\cos(\eta(x+t)) + 1).$$

The initial solution is extracted from the exact solution and the source term $s(x,t)$ is chosen properly such that the exact solution satisfies the given equation.

For such a nonlinear problem, since the stability condition (3.3) involves the unknown solutions above the n-th time level, the VC-EIN-LDG scheme is adjusted for use with the help of the convolution technique. In short, we add and subtract a second derivative term with variable coefficient $(\tilde{a}_1(x)U_x)_x$ at the left-hand side of the equation (3.7)

$$U_t + \underbrace{(\tilde{a}_1(x)U_x)_x - (d(U)U_x)_x - s(x,t)}_{T_1} - \underbrace{(\tilde{a}_1(x)U_x)_x}_{T_2} = 0,$$

where $\tilde{a}_1(x) = a_0 \times \tilde{d}(x)$ and $\tilde{d}(x)$ is the convolution of $d(u^n)$ and the dilated mollifier $\Phi_{C_0,\delta}(x)$ defined by (1.5). By adjusting the dilation parameters δ and C_0, we can always make $\tilde{a}_1(x)$ satisfy (1.6) and then ensure the stability of the scheme.

With the numerical solution at time t^n in hand, one might be tempted to increase the value of a_0 to make the inequality

$$a_0 \times d(u(x,t^n)) \geq 0.54 \times \max_{t^n \leq t \leq t^{n+1}} d(u(x,t))$$

tenable and ensure the stability of the scheme. However, this approach has two shortcomings that cannot be ignored. One is that much larger a_0 might be needed, which could bring larger errors. The other is that for degenerate parabolic equations, of which the diffusion coefficient $d(U)$ has a compact support, even if we increase the value of a_0, it still leads to $a_0 \times d(U) = 0$ outside the support. When the interface of the support is sharp and propagates with a high speed, it may still fail to remove the stiffness in all $\mathcal{N}(t_s^n, u^{n,s}), 2 \leq s \leq 5$ defined by (2.10), and ensure the stability of the scheme on a relatively coarse grid. With the convolution of $d(u^n)$ and the dilated mollifier $\Phi_{C_0,\delta}(x)$, we can expand the support to avoid such a situation.

However, now we encounter the difficulty on how to adjust the dilation parameters C_0 and δ to make $\tilde{d}(x) \geq \max_{t^n \leq t \leq t^{n+1}} d(u(x,t))$. At the beginning of the calculation, we preset the

parameters C_0 and δ according to $d(u^0)$. In the computing, we scan the sign of $\tilde{d}(x) - d(u^{n,s})$ at every stage, where $u^{n,s}$, $2 \le s \le 5$ are the intermediate stage values defined by (2.10a), at some preselected points, for example,

$$x_j^s = x_j + \frac{s-1}{3}h, \qquad s = 0, 1, 2, \;\; j = 1, 2, ..., N.$$

For all the stages, if no negativity is detected at these points, then it is acceptable to judge that $\tilde{d}(x) - \max_{t^n \le t \le t^{n+1}} d(u(x,t))$ is nonnegative, and we make the parameters C_0 and δ stay the same as before. If $\tilde{d}(x) - d(u^{n,s})$ is negative at some points at a certain stage, then we return to the n-th time level and modify the parameters C_0 and δ according to $d(u^n)$. When the time step is relatively small, we can always obtain satisfactory parameters through a finite number of adjustments to ensure the stability of the scheme.

First, we numerically validate the stability of the VC-EIN-LDG scheme for this nonlinear problem. In the test, we take the parameters $\alpha = 0$, $\beta = 1$, $\gamma = 0$, $\lambda = 0.5$, $\sigma = 6$, $\eta = 4$. The computational domain is set to be $(-\pi, \pi)$ and the final computing time is $T = 10$. With these parameters, the diffusion coefficient $d(U)$, for any time $t \ge 0$, forms a steep bump with the value of $d(U)$ outside the bump decaying exponentially. To understand this clearly, we have plotted in Figure 3.1 the pictures for the diffusion coefficient $d(U(x,t))$ at time $t = 0$ and $t = 1$, respectively. The numerical results of the VC-EIN-LDG scheme with $a_1(x) = 10 \times d(u^n)$, $a_1(x) = 100 \times d(u^n)$, $\tilde{a}_1(x) = 0.54 \times \tilde{d}(x)$ and $\tilde{a}_1(x) = \tilde{d}(x)$ are presented in Table 3.3. Note that we preset the dilation parameters as $C_0 = 2.8$ and $\delta = 0.6$ in the test. These two preset values are sufficient to ensure the stability of the scheme without adjustment, due to the fact that the bump of the diffusion coefficient does not deform except for moving. As we expected, the scheme is stable for $\tilde{a}_1(x) \ge 0.54 \times \max_{t^n \le t \le t^{n+1}} d(u(x,t))$. In addition, even though we greatly increase the value of a_0 in the test, $a_1(x) = a_0 \times d(u^n)$ still has no way to ensure the stability of the scheme on a relatively coarse grid. When $N = 2048$, the VC-EIN-LDG scheme is stable for $a_1(x) = 100 \times d(u^n)$, however, the errors of the scheme are several orders of magnitude larger than those of the scheme with $\tilde{a}_1(x) = 0.54 \times \tilde{d}(x)$ and $\tilde{a}_1(x) = \tilde{d}(x)$.

Second, we perform an optimal accuracy check for the VC-EIN-LDG scheme. In the test, we take the parameters $\alpha = 0$, $\beta = 1$, $\gamma = 0.01$, $\lambda = 0.1$, $\sigma = 3$, $\eta = 4$. Compared with the previous case, we reduce the values of these two parameters, λ and σ, and increase the value of γ. With these parameters, the diffusion coefficient $d(U)$, for any time $t \geq 0$, forms a gentler bump and the value of $d(U)$ outside the bump no longer decays exponentially. To understand this clearly, we have plotted in Figure 3.2 the pictures for the diffusion coefficient $d(U(x,t))$ at time $t = 0$ and $t = 1$, respectively. On the basis of keeping the final computing time and the computational domain unchanged, the simulation is run with $\tilde{a}_1(x) = 0.54 \times \tilde{d}(x)$. Note that we preset the dilation parameters as $C_0 = 2$ and $\delta = 0.6$ in the test. The discrete L^1, L^2 and L^∞ norms of the errors of the scheme are presented in Table 3.4. A clear third order accuracy is observed in all the norms. In addition, the CC-EIN-LDG scheme is also used to solve the nonlinear problem. In the test, we take $b_1 = 0.54 \times \max d(u^n)$. The numerical results of the scheme are also listed in Table 3.4. From the experiment we can see that the CC-EIN-LDG scheme is stable and can achieve optimal order of accuracy. Under the same mesh grid, the results of the VC-EIN-LDG scheme are compared against those of the CC-EIN-LDG scheme. Obviously, the VC-EIN-LDG scheme gives much more accurate results especially for the L^1 norm.

Third, we increase the deviation of the bump and compare the performance of the VC-EIN-LDG scheme with the CC-EIN-LDG scheme. On the basis of keeping other parameters of the previous case unchanged, the value of λ is increased to 0.5. Note that for the VC-EIN-LDG scheme, we still preset the dilation parameters as $C_0 = 2$ and $\delta = 0.6$. The errors and orders of accuracy table (Table 3.5) is given for these two schemes. Obviously, the superior errors give the VC-EIN-LDG scheme an advantage over the CC-EIN-LDG scheme.

3.1.3 The nonlinear numerical test in two dimension

We consider the two-dimensional nonlinear diffusion equation

$$U_t = \nabla \cdot (d(U)\nabla U) + s(x, y, t) \tag{3.9}$$

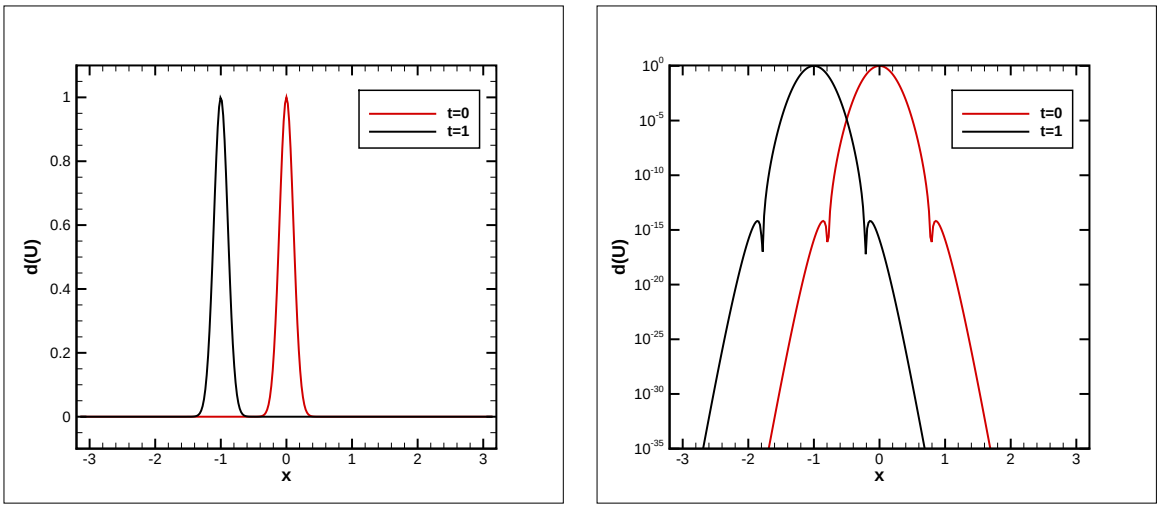

Figure 3.1: *Snapshots of the diffusion coefficient* (3.8) *with parameters* $\alpha = 0$, $\beta = 1$, $\gamma = 0$, $\lambda = 0.5$, $\sigma = 6$, $\eta = 4$ *at the indicated times.*

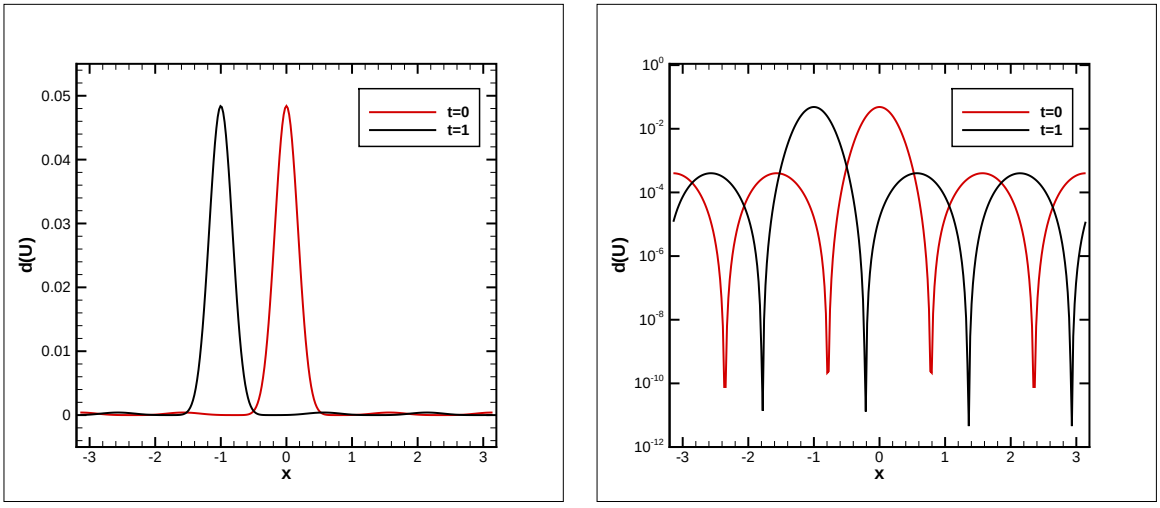

Figure 3.2: *Snapshots of the diffusion coefficient* (3.8) *with parameters* $\alpha = 0$, $\beta = 1$, $\gamma = 0.01$, $\lambda = 0.1$, $\sigma = 3$, $\eta = 4$ *at the indicated times.*

Table 3.3: The errors and orders of the VC-EIN-LDG scheme for Example (3.7) with $\alpha = 0$, $\beta = 1$, $\gamma = 0$, $\lambda = 0.5$, $\sigma = 6$, $\eta = 4$.

	N	L^1 error	order	L^∞ error	order	L^2 error	order
$\tilde{a}_1(x) = 0.54 \times \tilde{d}(x)$	128	3.32E-03		7.25E-02		8.73E-03	
	256	6.58E-04	2.34	1.02E-02	2.83	1.83E-03	2.25
	512	1.11E-04	2.57	2.20E-03	2.22	3.28E-04	2.48
	1024	1.71E-05	2.70	3.25E-04	2.76	5.15E-05	2.67
	2048	2.73E-06	2.65	4.91E-05	2.73	8.15E-06	2.66
$\tilde{a}_1(x) = \tilde{d}(x)$	128	NaN		NaN		NaN	
	256	2.21E-03	NaN	5.66E-02	NaN	6.38E-03	NaN
	512	5.40E-04	2.03	1.16E-02	2.29	1.49E-03	2.09
	1024	1.15E-04	2.23	2.37E-03	2.29	3.14E-04	2.25
	2048	2.16E-05	2.41	4.40E-04	2.43	5.84E-05	2.42
$a_1(x) = 10 \times d(u^n)$	128	NaN		NaN		NaN	
	256	NaN	NaN	NaN	NaN	NaN	NaN
	512	NaN	NaN	NaN	NaN	NaN	NaN
	1024	NaN	NaN	NaN	NaN	NaN	NaN
	2048	NaN	NaN	NaN	NaN	NaN	NaN
$a_1(x) = 100 \times d(u^n)$	128	NaN		NaN		NaN	
	256	NaN	NaN	NaN	NaN	NaN	NaN
	512	NaN	NaN	NaN	NaN	NaN	NaN
	1024	NaN	NaN	NaN	NaN	NaN	NaN
	2048	3.47E-03	NaN	8.32E-02	NaN	9.73E-03	NaN

Table 3.4: The errors and orders of the VC-EIN-LDG and CC-EIN-LDG schemes for Example (3.7) with $\alpha = 0$, $\beta = 1$, $\gamma = 0.01$, $\lambda = 0.1$, $\sigma = 3$, $\eta = 4$.

VC-EIN-LDG							
	N	L^1 error	order	L^∞ error	order	L^2 error	order
$\tilde{a}_1(x) = 0.54 \times \tilde{d}(x)$	128	3.62E-06		4.33E-05		7.66E-06	
	256	4.19E-07	3.11	5.45E-06	2.99	9.01E-07	3.09
	512	5.12E-08	3.03	6.84E-07	2.99	1.10E-07	3.03
	1024	6.36E-09	3.01	8.60E-08	2.99	1.36E-08	3.01
	2048	7.94E-10	3.00	1.08E-08	3.00	1.70E-09	3.00
CC-EIN-LDG							
	N	L^1 error	order	L^∞ error	order	L^2 error	order
$b_1 = 0.54 \times \max d(u^n)$	128	4.59E-06		5.11E-05		8.54E-06	
	256	5.70E-07	3.01	6.41E-06	3.00	1.07E-06	3.00
	512	7.21E-08	2.98	8.08E-07	2.99	1.35E-07	2.98
	1024	9.05E-09	2.99	1.02E-07	2.99	1.70E-08	2.99
	2048	1.14E-09	2.99	1.28E-08	2.99	2.14E-09	2.99

Table 3.5: The errors and orders of the VC-EIN-LDG and CC-EIN-LDG schemes for Example (3.7) with $\alpha = 0$, $\beta = 1$, $\gamma = 0.01$, $\lambda = 0.5$, $\sigma = 3$, $\eta = 4$.

VC-EIN-LDG							
	N	L^1 error	order	L^∞ error	order	L^2 error	order
$\tilde{a}_1(x) = 0.54 \times \tilde{d}(x)$	128	5.88E-04		9.46E-03		1.62E-03	
	256	9.88E-05	2.57	2.04E-03	2.21	3.01E-04	2.43
	512	1.57E-05	2.65	2.85E-04	2.84	4.72E-05	2.67
	1024	2.45E-06	2.68	4.09E-05	2.80	7.25E-06	2.70
	2048	3.69E-07	2.73	5.97E-06	2.78	1.08E-06	2.74
CC-EIN-LDG							
	N	L^1 error	order	L^∞ error	order	L^2 error	order
$b_1 = 0.54 \times \max d(u^n)$	128	1.40E-03		1.00E-02		2.41E-03	
	256	3.62E-04	1.95	2.19E-03	2.20	6.32E-04	1.93
	512	7.58E-05	2.25	5.72E-04	1.94	1.37E-04	2.21
	1024	1.33E-05	2.51	1.17E-04	2.29	2.47E-05	2.47
	2048	2.02E-06	2.71	1.96E-05	2.58	3.83E-06	2.69

augmented with the diffusion coefficient (3.8) and the exact solution

$$U(x,y,t) = (\gamma + \lambda e^{-\sigma^2(1-\cos(x+y+t))})(1+\cos(\eta(x+y+t))).$$

The initial solution is extracted from the exact solution and the source term $s(x,y,t)$ is chosen properly such that the exact solution satisfies the given equation.

First, we numerically validate the stability and order of accuracy of the VC-EIN-LDG scheme. In the test, we take the parameters $\alpha = 0$, $\beta = 1$, $\gamma = 0.01$, $\lambda = 0.1$, $\sigma = 3$, $\eta = 4$. The computational domain is set to be $(-\pi, \pi)^2$ and the final computing time is $T = 10$. The numerical results of the scheme with $a_1(x,y) = 0.54 \times d(u^n)$ and $\tilde{a}_1(x,y) = 0.54 \times \tilde{d}(x,y)$ are presented in Table 3.6. Note that for the VC-EIN-LDG scheme, the dilation parameters are preset as $C_0 = 2$ and $\delta = 0.6$. It is observed that the scheme is stable for $\tilde{a}_1(x,y) = 0.54 \times \tilde{d}(x,y)$ and can achieve very nice third order convergence rates for L^1, L^2 and L^∞ norms. Second, the CC-EIN-LDG scheme is also used to solve the nonlinear problem. In the test, we take $b_1 = 0.54 \times \max d(u^n)$. The numerical results of the scheme are also listed in Table 3.6, from which we can see that the CC-EIN-LDG scheme is stable as always and can achieve optimal order of accuracy. Under the same mesh grid, the results of the VC-EIN-LDG scheme are compared against those of the CC-EIN-LDG scheme. Obviously, the VC-EIN-LDG scheme gives more accurate results.

3.2 The dispersion equations

In this subsection, we would like to test the performance and stability of the proposed schemes for the dispersion equations in one and two space dimensions. The generalization of the LDG scheme (2.8) to the two-dimensional dispersion equation is straightforward; we refer the readers to [21] for the details.

3.2.1 The quasi-linear numerical test in one dimension

Indeed, the standard IMEX methods can be directly adopted to solve the quasi-linear equations. In order to illustrate the necessity of the stability condition (3.4), we consider the

Table 3.6: The errors and orders of the VC-EIN-LDG and CC-EIN-LDG schemes for Example (3.9) with $\alpha = 0$, $\beta = 1$, $\gamma = 0.01$, $\lambda = 0.1$, $\sigma = 3$, $\eta = 4$.

VC-EIN-LDG

	N	L^1 error	order	L^∞ error	order	L^2 error	order
$a_1(x,y) = 0.54 \times d(u^n)$	20	NaN		NaN		NaN	
	40	NaN	NaN	NaN	NaN	NaN	NaN
	60	NaN	NaN	NaN	NaN	NaN	NaN
	80	NaN	NaN	NaN	NaN	NaN	NaN
	100	NaN	NaN	NaN	NaN	NaN	NaN
$\tilde{a}_1(x,y) = 0.54 \times \tilde{d}(x,y)$	20	2.27E-03		4.33E-02		4.35E-03	
	40	6.14E-04	1.88	2.16E-02	1.01	1.49E-03	1.54
	60	1.36E-04	3.71	5.67E-03	3.30	3.45E-04	3.61
	80	5.41E-05	3.21	2.11E-03	3.43	1.34E-04	3.28
	100	2.79E-05	2.96	9.95E-04	3.37	6.77E-05	3.07

CC-EIN-LDG

	N	L^1 error	order	L^∞ error	order	L^2 error	order
$b_1 = 0.54 \times \max d(u^n)$	20	9.69E-03		2.39E-01		2.52E-02	
	40	5.68E-04	4.09	2.34E-02	3.35	1.48E-03	4.09
	60	1.38E-04	3.48	5.56E-03	3.55	3.66E-04	3.44
	80	6.17E-05	2.81	2.18E-03	3.25	1.63E-04	2.81
	100	3.11E-05	3.07	9.93E-04	3.53	8.36E-05	3.00

quasi-linear dispersion equation

$$U_t + \underbrace{[(g(x,t) - a_1(x))U_x]_{xx} - s(x,t)}_{T_1} + \underbrace{(a_1(x)U_x)_{xx}}_{T_2} = 0, \qquad x \in (0, 2\pi) \tag{3.10}$$

augmented with the dispersion coefficient

$$g(x,t) = \alpha - \beta \tanh^2(\eta \cos(x+t)),$$

the initial condition $U(x,0) = \sin(x)$ and the source term

$$\begin{aligned} s(x,t) = & -2\beta\eta^2 \cos(x+t)\,\mathrm{sech}^4(\eta\cos(x+t))\sin^2(x+t) + \\ & 2\beta\eta\,\mathrm{sech}^2(\eta\cos(x+t))\tanh(\eta\cos(x+t))(\cos^2(x+t) - \\ & 2\sin^2(x+t) + 2\eta\cos(x+t)\sin^2(x+t)\tanh(\eta\cos(x+t))) + \\ & \cos(x+t)(1-\alpha+\beta\tanh^2(\eta\cos(x+t))). \end{aligned}$$

The exact solution to the problem is defined by (3.6). In order to test the stability of the VC-EIN-LDG scheme in terms of $a_1(x)$, in the test, we take $a_1(x)$ as $0.54 \times g(x,t^n)$, $0.53 \times \max_{t^n \le t \le t^{n+1}} g(x,t)$, $0.54 \times \max_{t^n \le t \le t^{n+1}} g(x,t)$ and $\max_{t^n \le t \le t^{n+1}} g(x,t)$, respectively. We compute to $T = 10$ with the parameters $\alpha = 1$, $\beta = 1$, $\eta = 3$. With these parameters, the dispersion coefficient $g(x,t)$ is not a monotonic function of time t for any fixed x. The errors and orders of accuracy in different norms are presented in Table 3.7, from which we can see that the scheme is unstable or a sufficiently dense mesh grid is required to maintain the stability of the scheme if $a_1(x) < 0.54 \times \max_{t^n \le t \le t^{n+1}} g(x,t)$. When $a_1(x) = 0.54 \times \max_{t^n \le t \le t^{n+1}} g(x,t)$, the VC-EIN-LDG scheme is stable and can achieve a very nice third order convergence rate. When $a_1(x) = \max_{t^n \le t \le t^{n+1}} g(x,t)$, the scheme remains stable as always, but the numerical orders of the scheme settle down towards the asymptotic value slowly.

3.2.2 The nonlinear numerical test in one dimension

We consider the nonlinear dispersion equation

$$U_t + g(U_x)_{xx} = s(x,t) \tag{3.11}$$

Table 3.7: The errors and orders of the VC-EIN-LDG scheme for Example (3.10) with $\alpha = 1$, $\beta = 1$, $\eta = 3$.

$a_1(x)$	N	L^1 error	order	L^∞ error	order	L^2 error	order
$0.54 \times g(x, t^n)$	64	5.47E+50		5.00E+51		1.25E+51	
	128	1.06E+53	-7.60	1.18E+54	-7.88	2.65E+53	-7.72
	256	3.51E+46	21.52	4.63E+47	21.28	9.52E+46	21.41
	512	3.89E+25	69.61	5.72E+26	69.46	1.13E+26	69.51
	1024	6.30E-06	102.28	2.02E-05	104.48	7.93E-06	103.50
$0.53 \times \max_{t^n \le t \le t^{n+1}} g(x, t)$	64	4.62E-02		1.15E-01		5.53E-02	
	128	5.34E-03	3.11	1.28E-02	3.17	6.32E-03	3.13
	256	4.90E-04	3.44	1.28E-03	3.32	5.76E-04	3.46
	512	5.54E+20	-79.90	7.51E+21	-82.27	1.46E+21	-81.07
	1024	4.01E+70	-165.63	5.73E+71	-165.71	1.08E+71	-165.66
$0.54 \times \max_{t^n \le t \le t^{n+1}} g(x, t)$	64	4.64E-02		1.16E-01		5.55E-02	
	128	5.34E-03	3.12	1.28E-02	3.17	6.32E-03	3.13
	256	4.92E-04	3.44	1.16E-03	3.47	5.73E-04	3.46
	512	4.69E-05	3.39	1.09E-04	3.41	5.61E-05	3.35
	1024	5.86E-06	3.00	1.36E-05	3.00	7.06E-06	2.99
$\max_{t^n \le t \le t^{n+1}} g(x, t)$	64	1.17E-02		3.10E-02		1.43E-02	
	128	2.97E-04	5.30	1.09E-03	4.83	4.00E-04	5.16
	256	9.24E-05	1.69	2.92E-04	1.90	1.18E-04	1.77
	512	2.21E-05	2.06	8.52E-05	1.78	2.86E-05	2.04
	1024	5.41E-06	2.03	2.09E-05	2.02	6.88E-06	2.06

augmented with the dispersion coefficient

$$g(U_x) = \alpha U_x + \beta U_x^3$$

and the exact solution

$$U(x,t) = \gamma \sin(\eta(x+t)) - \lambda \tanh(\sigma \cos(x+t)).$$

The initial solution is extracted from the exact solution and the source term $s(x,t)$ is chosen properly such that the exact solution satisfies the given equation.

Similarly, for such a nonlinear problem, since the stability condition (3.4) involves the unknown solutions above the n-th time level, the VC-EIN-LDG scheme is adjusted for use with the help of the convolution technique. In short, we add and subtract a third derivative term with variable coefficient $(\tilde{a}_1(x)U_x)_{xx}$ at the left-hand side of the equation

$$U_t + \underbrace{g(U_x)_{xx} - (\tilde{a}_1(x)U_x)_{xx} - s(x,t)}_{T_1} + \underbrace{(\tilde{a}_1(x)U_x)_{xx}}_{T_2} = 0,$$

where $\tilde{a}_1(x) = a_0 \times \tilde{g}(x)$ and $\tilde{g}(x)$ is the convolution of $g'(u_x^n)$ and the dilated mollifier $\Phi_{C_0,\delta}(x)$ defined by (1.5). By adjusting the dilation parameters δ and C_0, we can always make $\tilde{a}_1(x)$ satisfy the following inequality

$$\tilde{a}_1(x) = a_0 \times \tilde{g}(x) \geq a_1(x) = a_0 \times \max_{t^n \leq t \leq t^{n+1}} g'(u_x(x,t)),$$

and ensure the stability of the VC-EIN-LDG scheme. The adjustment strategy of the dilation parameters δ and C_0 is similar to that described in Section 3.1.2.

First, we numerically validate the stability of the VC-EIN-LDG scheme for this nonlinear problem. In the test, we take the parameters $\alpha = 0.001$, $\beta = 0.1$, $\gamma = -0.2$, $\lambda = 0.2$, $\sigma = 2$, $\eta = 2$. The computational domain is set to be $(-\pi, \pi)$ and the final computing time is $T = 10$. The numerical results of the VC-EIN-LDG scheme with $a_1(x) = 0.54 \times g'(u_x^n)$, $a_1(x) = g'(u_x^n)$, $\tilde{a}_1(x) = 0.54 \times \tilde{g}(x)$ and $\tilde{a}_1(x) = \tilde{g}(x)$ are presented in Table 3.8. Note that the dilation parameters are preset as $C_0 = 2$ and $\delta = 1$. As we expected, the scheme is stable for $\tilde{a}_1(x) \geq 0.54 \times \tilde{g}(x)$. Although we slightly increase the value of a_0, $a_0 \times g'(u_x^n)$ still has

no way to ensure the stability of the scheme on a relatively coarse grid. In addition, even though the VC-EIN-LDG scheme is stable for $a_1(x) = g'(u_x^n)$ when $N \geq 256$, the errors of the scheme are larger than those of the scheme with $\tilde{a}_1(x) = 0.54 \times \tilde{g}(x)$ and $\tilde{a}_1(x) = \tilde{g}(x)$.

Second, we numerically validate the optimal order of accuracy of the VC-EIN-LDG scheme. In the test, we take the parameters $\alpha = 0.001$, $\beta = 0.025$, $\gamma = -0.05$, $\lambda = 0.05$, $\sigma = 4$, $\eta = 2$. On the basis of keeping the final computing time and the computational domain unchanged, the simulation is run with $\tilde{a}_1(x) = 0.54 \times \tilde{g}(x)$. Note that we preset the dilation parameters as $C_0 = 2$ and $\delta = 0.7$. In Table 3.9 we present the numerical errors and orders for the discrete L^1, L^2 and L^∞ norms and verify the optimal order of accuracy of the scheme. In addition, the CC-EIN-LDG scheme is also used to solve the nonlinear problem. In the test, we take $b_1 = 0.54 \times \max g'(u_x^n)$. The numerical results of the scheme are also listed in Table 3.9. From the experiment we can see that the CC-EIN-LDG scheme is stable as always and can achieve optimal order of accuracy. Compared with the CC-EIN-LDG scheme, the VC-EIN-LDG scheme is more accurate for this test.

Third, we take $\alpha = 0.001$, $\beta = 0.1$, $\gamma = -0.1$, $\lambda = 0.1$, $\sigma = 4$, $\eta = 2$ and compare the performance of the VC-EIN-LDG scheme with the CC-EIN-LDG scheme. Compared with the previous case, we have reduced the value of γ and increased the values of β and λ. With these parameters, $g'(U_x)$ has a larger deviation and the bump of $g'(U_x)$ tends to be steeper. To understand this clearly, we have plotted in Figure 3.3 the pictures for $g'(U_x)$ at time $t = 0$ and $t = 1$, respectively. Note that for the VC-EIN-LDG scheme, we preset the dilation parameters as $C_0 = 3$ and $\delta = 0.7$ in the test. The errors and orders of accuracy of these two schemes are listed in Table 3.10. Clearly, under the same mesh grid, the VC-EIN-LDG scheme gives much more accurate results than the CC-EIN-LDG scheme with several orders of magnitude.

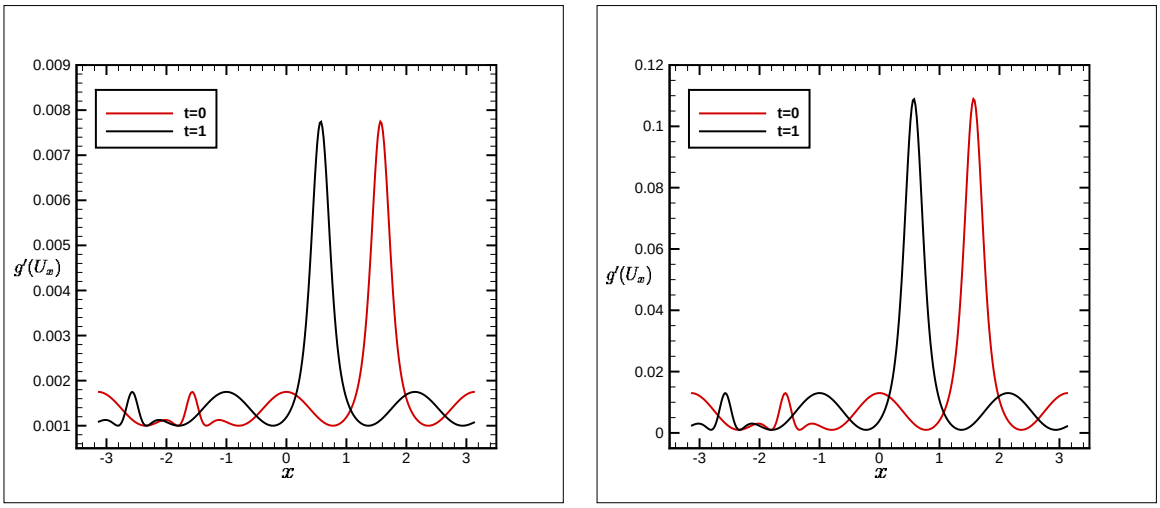

Figure 3.3: *Snapshots of* $g'(U_x)$ *at the indicated times. Left:* $\alpha = 0.001$, $\beta = 0.025$, $\gamma = -0.05$, $\lambda = 0.05$, $\sigma = 4$, $\eta = 2$. *Right:* $\alpha = 0.001$, $\beta = 0.1$, $\gamma = -0.1$, $\lambda = 0.1$, $\sigma = 4$, $\eta = 2$.

Table 3.8: The errors and orders of the VC-EIN-LDG scheme for Example (3.11) with $\alpha = 0.001$, $\beta = 0.1$, $\gamma = -0.2$, $\lambda = 0.2$, $\sigma = 2$, $\eta = 2$.

	N	L^1 error	order	L^∞ error	order	L^2 error	order
$a_1(x) = 0.54 \times g'(u_x^n)$	128	NaN		NaN		NaN	
	256	NaN	NaN	NaN	NaN	NaN	NaN
	512	NaN	NaN	NaN	NaN	NaN	NaN
	1024	NaN	NaN	NaN	NaN	NaN	NaN
	2048	NaN	NaN	NaN	NaN	NaN	NaN
$a_1(x) = g'(u_x^n)$	128	NaN		NaN		NaN	
	256	2.83E-05	NaN	1.58E-04	NaN	4.27E-05	NaN
	512	4.48E-06	2.66	3.28E-05	2.27	7.53E-06	2.50
	1024	8.92E-07	2.33	5.91E-06	2.48	1.39E-06	2.43
	2048	1.58E-07	2.50	9.61E-07	2.62	2.32E-07	2.59
$\tilde{a}_1(x) = 0.54 \times \tilde{g}(x)$	128	1.15E-04		7.43E-04		1.71E-04	
	256	1.78E-05	2.69	1.39E-04	2.42	2.88E-05	2.57
	512	2.89E-06	2.62	2.53E-05	2.45	5.18E-06	2.48
	1024	5.45E-07	2.41	4.07E-06	2.64	8.88E-07	2.55
	2048	9.22E-08	2.56	5.75E-07	2.82	1.38E-07	2.69
$\tilde{a}_1(x) = \tilde{g}(x)$	128	1.12E-04		6.01E-04		1.77E-04	
	256	2.52E-05	2.16	1.43E-04	2.07	3.80E-05	2.22
	512	4.66E-06	2.44	3.11E-05	2.21	7.39E-06	2.36
	1024	8.72E-07	2.42	5.77E-06	2.43	1.36E-06	2.44
	2048	1.56E-07	2.48	9.48E-07	2.61	2.28E-07	2.57

Table 3.9: The errors and orders of the VC-EIN-LDG and CC-EIN-LDG schemes for Example (3.11) with $\alpha = 0.001$, $\beta = 0.025$, $\gamma = -0.05$, $\lambda = 0.05$, $\sigma = 4$, $\eta = 2$.

VC-EIN-LDG							
	N	L^1 error	order	L^∞ error	order	L^2 error	order
$\tilde{a}_1(x) = 0.54 \times \tilde{g}(x)$	256	2.67E-06		1.17E-05		3.63E-06	
	512	3.67E-07	2.86	1.70E-06	2.78	4.81E-07	2.92
	1024	4.90E-08	2.91	2.10E-07	3.02	6.24E-08	2.95
	2048	6.43E-09	2.93	2.60E-08	3.01	8.23E-09	2.92
	4096	8.12E-10	2.99	3.22E-09	3.01	1.03E-09	3.00
CC-EIN-LDG							
	N	L^1 error	order	L^∞ error	order	L^2 error	order
$b_1 = 0.54 \times \max g'(u_x^n)$	256	1.27E-06		8.09E-06		1.95E-06	
	512	5.09E-07	1.32	1.99E-06	2.03	6.93E-07	1.49
	1024	9.76E-08	2.38	4.13E-07	2.27	1.36E-07	2.35
	2048	1.24E-08	2.97	5.57E-08	2.89	1.75E-08	2.96
	4096	1.53E-09	3.02	7.05E-09	2.98	2.16E-09	3.01

Table 3.10: The errors and orders of the VC-EIN-LDG and CC-EIN-LDG schemes for Example (3.11) with $\alpha = 0.001$, $\beta = 0.1$, $\gamma = -0.1$, $\lambda = 0.1$, $\sigma = 4$, $\eta = 2$.

VC-EIN-LDG							
	N	L^1 error	order	L^∞ error	order	L^2 error	order
$\tilde{a}_1(x) = 0.54 \times \tilde{g}(x)$	128	2.30E-04		1.29E-03		3.34E-04	
	256	3.99E-05	2.53	3.96E-04	1.71	7.49E-05	2.15
	512	7.71E-06	2.37	1.09E-04	1.86	1.81E-05	2.05
	1024	1.83E-06	2.08	2.64E-05	2.05	4.22E-06	2.10
	2048	3.94E-07	2.21	5.78E-06	2.19	8.96E-07	2.24
CC-EIN-LDG							
	N	L^1 error	order	L^∞ error	order	L^2 error	order
$b_1 = 0.54 \times \max g'(u_x^n)$	128	6.72E-04		3.05E-03		9.07E-04	
	256	3.30E-04	1.02	1.82E-03	0.74	4.67E-04	0.96
	512	1.40E-04	1.24	8.87E-04	1.04	2.12E-04	1.14
	1024	5.46E-05	1.36	3.75E-04	1.24	8.52E-05	1.31
	2048	2.00E-05	1.45	1.30E-04	1.52	3.13E-05	1.44

3.2.3 The nonlinear numerical test in two dimension

We consider the two-dimensional nonlinear dispersion equation

$$U_t + g(U_y)_{xy} = s(x, y, t) \tag{3.12}$$

augmented with the dispersion coefficient

$$g(U_y) = \alpha U_y + \beta U_y^3$$

and the exact solution

$$U(x, y, t) = \gamma \sin(\eta(x + y + t)) - \lambda \tanh(\sigma \cos(x + y + t)).$$

The initial solution is extracted from the exact solution and the source term $s(x, y, t)$ is chosen properly such that the exact solution satisfies the given equation.

First, we numerically validate the stability and order of accuracy of the VC-EIN-LDG scheme. In the test, we take the parameters $\alpha = 0.001$, $\beta = 0.02$, $\gamma = -0.05$, $\lambda = 0.05$, $\sigma = 4$, $\eta = 1$. The computational domain is set to be $(-\pi, \pi)^2$ and the final computing time is $T = 10$. The numerical results of the VC-EIN-LDG scheme with $a_1(x, y) = 0.54 \times g'(u_y^n)$ and $\tilde{a}_1(x, y) = 0.54 \times \tilde{g}(x, y)$ are presented in Table 3.11. Note that we preset the dilation parameters as $C_0 = 2$ and $\delta = 0.7$. It is observed that the scheme is stable for $\tilde{a}_1(x, y) = 0.54 \times \tilde{g}(x, y)$ and can achieve very nice third order convergence rates for L^1, L^2 and L^∞ norms. Second, the CC-EIN-LDG scheme is also used to solve the nonlinear problem. In the test, we take $b_1 = 0.54 \times \max g'(u_y^n)$. The numerical results of the scheme are also listed in Table 3.11, from which we can see that the CC-EIN-LDG scheme is stable as always and can achieve optimal order of accuracy. Under the same mesh grid, the results of the VC-EIN-LDG scheme are compared against those of the CC-EIN-LDG scheme. Obviously, the VC-EIN-LDG scheme gives more accurate results.

Table 3.11: The errors and orders of the VC-EIN-LDG and CC-EIN-LDG schemes for Example (3.12) with $\alpha = 0.001$, $\beta = 0.02$, $\gamma = -0.05$, $\lambda = 0.05$, $\sigma = 4$, $\eta = 1$.

VC-EIN-LDG

	N	L^1 error	order	L^∞ error	order	L^2 error	order
$a_1(x,y) = 0.54 \times g'(u_y^n)$	20	1.65E-03		2.06E-02		2.91E-03	
	40	NaN	NaN	NaN	NaN	NaN	NaN
	60	NaN	NaN	NaN	NaN	NaN	NaN
	80	NaN	NaN	NaN	NaN	NaN	NaN
	100	NaN	NaN	NaN	NaN	NaN	NaN
$\tilde{a}_1(x,y) = 0.54 \times \tilde{g}(x,y)$	20	5.88E-04		1.27E-02		1.26E-03	
	40	1.06E-04	2.47	3.18E-03	2.00	2.61E-04	2.27
	60	3.81E-05	2.53	1.08E-03	2.68	1.01E-04	2.34
	80	1.62E-05	2.97	4.60E-04	2.96	4.31E-05	2.96
	100	8.17E-06	3.07	2.28E-04	3.14	2.16E-05	3.10

CC-EIN-LDG

	N	L^1 error	order	L^∞ error	order	L^2 error	order
$b_1 = 0.54 \times \max g'(u_y^n)$	20	7.28E-04		1.47E-02		1.50E-03	
	40	1.07E-04	2.77	2.91E-03	2.34	2.55E-04	2.56
	60	4.83E-05	1.96	1.24E-03	2.10	1.24E-04	1.78
	80	1.95E-05	3.14	5.01E-04	3.15	4.74E-05	3.34
	100	9.83E-06	3.08	2.49E-04	3.13	2.39E-05	3.06

4 Concluding remarks

The present study investigates the stability and performance of a third order VC-EIN method in conjunction with the LDG methods for the diffusion and dispersion equations, respectively. Unlike the CC-EIN method, the one we add to and subtract from the original equation is a spatially varying linear term. Based on the stability results of the schemes for simplified linear equations [16], we provide a guidance for the choice of the variable coefficient $a_1(x)$ to ensure the stability of the VC-EIN-LDG schemes for the quasi-linear and nonlinear equations. Numerical experiments show that the schemes can be stable under a relatively coarse mesh grid and achieve optimal orders of accuracy when the stability constraints (3.3) and (3.4) are satisfied. As a comparative study we also revisit the CC-EIN-LDG scheme shown in [16]. When proper parameters: δ, C_0, a_0 are chosen, the numerical results show that the VC-EIN-LDG scheme is more accurate than the CC-EIN-LDG scheme of the same order, if the diffusion coefficient or the dispersion coefficient has a few high and narrow bumps and the bumps only account for a small part of the whole computational domain. We believe that the VC-EIN method can be successfully applied in many other contexts well beyond the ones presented in the paper.

Conflict of interest statement. On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

[1] U. M. Ascher, S. J. Ruuth and R. J. Spiteri, *Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations*, Applied Numerical Mathematics, 25, 1997, 151-167.

[2] F. Bassi, L. Botti, A. Colombo, A. Ghidoni and F. Massa, *Linearly implicit Rosenbrock-type Runge-Kutta schemes applied to the Discontinuous Galerkin solution of compress-*

ible and incompressible unsteady flows, Computers and Fluids, 118, 2015, 305-320.

[3] T. B. Benjamin, J. L. Bona and J. J. Mahony, *Model equations for long waves in nonlinear dispersive systems*, Philosophical Transactions of the Royal Society A: Mathematical Physical and Engineering Sciences, 272, 1972, 47-78.

[4] P. Cavaliere, G. Zavarise and M. Perillo, *Modeling of the carburizing and nitriding processes*, Computational Materials Science, 46, 2009, 26-35.

[5] B. Cockburn and C.-W. Shu, *The local discontinuous Galerkin method for time-dependent convection-diffusion systems*, SIAM Journal on Numerical Analysis, 35, 1998, 2440-2463.

[6] B. Cockburn and C.-W. Shu, *Runge-Kutta discontinuous Galerkin methods for convection-dominated problems*, Journal of Scientific Computing, 16, 2001, 173-261.

[7] J. Douglas Jr and T. Dupont, *Alternating-direction Galerkin methods on rectangles*, Numerical Solution of Partial Differential Equations-II, New York: Academic Press, 1971, 133-214.

[8] L. Duchemin and J. Eggers, *The explicit-implicit-null method: Removing the numerical instability of PDEs*, Journal of Computational Physics, 263, 2014, 37-52.

[9] J. Eggers, J. R. Lister and H. A. Stone, *Coalescence of liquid drops*, Journal of Fluid Mechanics, 401, 1999, 293-310.

[10] F. Filbet and S. Jin, *A class of asymptotic-preserving schemes for kinetic equations and related problems with stiff sources*, Journal of Computational Physics, 229, 2010, 7625-7648.

[11] F. X. Giraldo, J. F. Kelly and E. M. Constantinescu, *Implicit-Explicit formulations of a three-dimensional nonhydrostatic unified model of the atmosphere (NUMA)*, SIAM Journal on Scientific Computing, 35, 2013, B1162-B1194.

[12] V. John, G. Matthies and J. Rang, *A comparison of time-discretization/linearization approaches for the incompressible Navier-Stokes equations*, Computer Methods in Applied Mechanics and Engineering, 195, 2006, 5995-6010.

[13] D. J. Korteweg and G. De Vries, *On the change of form of long waves advancing in a rectangular canal, and on a new type of long stationary waves*, Philosophical Magazine, 91, 2011, 1007-1028.

[14] H. Liu and J. Yan, *A local discontinuous Galerkin method for the Korteweg-de Vries equation with boundary effect*, Journal of Computational Physics, 215, 2006, 197-218.

[15] P. Smereka, *Semi-Implicit level set methods for curvature and surface diffusion motion*, Journal of Scientific Computing, 19, 2003, 439-456.

[16] M. Tan, J. Cheng and C.-W. Shu, *Stability of high order finite difference and local discontinuous Galerkin schemes with explicit-implicit-null time-marching for high order dissipative and dispersive equations*, Journal of Computational Physics, 464, 2022, 111314.

[17] H. Wang, Q. Zhang and C.-W. Shu, *Third order implicit-explicit Runge-Kutta local discontinuous Galerkin methods with suitable boundary treatment for convection-diffusion problems with Dirichlet boundary conditions*, Journal of Computational and Applied Mathematics, 342, 2018, 164-179.

[18] H. Wang, Q. Zhang, S. Wang and C.-W. Shu, *Local discontinuous Galerkin methods with explicit-implicit-null time discretizations for solving nonlinear diffusion problems*, Science China Mathematics, 63, 2020, 183-204.

[19] Y. Xu and C.-W. Shu, *Optimal error estimates of the semidiscrete local discontinuous Galerkin methods for high order wave equations*, SIAM Journal on Numerical Analysis, 50, 2012, 79-104.

[20] Y. Xu and C.-W. Shu, *Local discontinuous Galerkin methods for three classes of nonlinear wave equations*, Journal of Computational Mathematics, 22, 2004, 250-274.

[21] J. Yan and C.-W. Shu, *A local discontinuous Galerkin method for KdV type equations*, SIAM Journal on Numerical Analysis, 40, 2002, 769-791.

[22] J. Yan and C.-W. Shu, *A Local discontinuous Galerkin methods for partial differential equations with higher order derivatives*, Journal of Scientific Computing, 17, 2002, 27-47.

[23] Q. Zhang and Z. Wu, *Numerical simulation for porous medium equation by local discontinuous Galerkin finite element method*, Journal of Scientific Computing, 38, 2009, 127-148.

Appendix

Stability analysis of the LDG method (2.2)

Theorem 1: *The numerical scheme* (2.2) *with the choice of fluxes* (2.6) *is* L^2 *stable, i.e.*

$$\frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega u^2\,\mathrm{d}x + \int_\Omega d(u)r^2\,\mathrm{d}x \le 0. \tag{A.1}$$

Proof. We sum up the four equalities in (2.2) and introduce the notation

$$\begin{aligned} B_j(u,p,q,r;\,\phi_1,\phi_2,\phi_3,\phi_4) = &\int_{I_j} u_t\phi_1\,\mathrm{d}x + \int_{I_j}(p+q)(\phi_1)_x\,\mathrm{d}x - (\hat{p}+\hat{q})_{j+\frac{1}{2}}(\phi_1)^-_{j+\frac{1}{2}} \\ &+ (\hat{p}+\hat{q})_{j-\frac{1}{2}}(\phi_1)^+_{j-\frac{1}{2}} + \int_{I_j} p\phi_2\,\mathrm{d}x - \int_{I_j}(d(u)-a_1(x))r\phi_2\,\mathrm{d}x \\ &+ \int_{I_j} q\phi_3\,\mathrm{d}x - \int_{I_j} a_1(x)r\phi_2\,\mathrm{d}x + \\ &\int_{I_j} r\phi_4\,\mathrm{d}x + \int_{I_j} u(\phi_4)_x\,\mathrm{d}x - \hat{u}_{j+\frac{1}{2}}(\phi_4)^-_{j+\frac{1}{2}} + \hat{u}_{j-\frac{1}{2}}(\phi_4)^+_{j-\frac{1}{2}}. \end{aligned}$$

Obviously, the solutions u, p, q, r of the scheme satisfy

$$B_j(u,p,q,r;\,\phi_1,\phi_2,\phi_3,\phi_4) = 0$$

for all ϕ_1, ϕ_2, ϕ_3, $\phi_4 \in V_h$. We then take

$$\phi_1 = u, \qquad \phi_2 = -r, \qquad \phi_3 = -r, \qquad \phi_4 = p+q$$

to obtain, after some algebraic manipulations,

$$0 = B_j(u,p,q,r;\,u,-r,-r,p+q) = \frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\int_{I_j} u^2\,\mathrm{d}x + \int_{I_j} d(u)r^2\,\mathrm{d}x + (\hat{H}_{j+\frac{1}{2}} - \hat{H}_{j-\frac{1}{2}}) + \Theta_{j-\frac{1}{2}},$$

where

$$\begin{aligned} \hat{H} &= \big(u(p+q)\big)^- - (\hat{p}+\hat{q})u^- - \hat{u}(p+q)^-, \\ \Theta &= -[u(p+q)] + (\hat{p}+\hat{q})[u] + \hat{u}[(p+q)]. \end{aligned}$$

Here, $[u]$ denotes $u^+ - u^-$. To this end, we notice that, with the definition of the numerical fluxes (2.6) and periodic boundary condition, we can easily obtain $\Theta_{j-\frac{1}{2}} = 0$. Then we sum over j to obtain (A.1). □