Inverse Lax–Wendroff boundary treatment of Discontinuous Galerkin method for 1D conservation laws

Lei Yang [1], Shun Li [2], Yan Jiang [3], Chi-Wang Shu [4] Mengping Zhang[5]

This paper is dedicated to the memory of Professor Zhong-Ci Shi.

Abstract: In this paper, we propose a new class of discontinuous Galerkin (DG) methods for solving one-dimensional conservation laws on unfitted meshes. The standard DG method is used on the interior cells. For the small cut elements around the boundaries, we directly design approximation polynomials based on inverse Lax-Wendroff principles for the inflow boundary conditions and introduce a post-processing to preserve the local conservation properties of the DG method. Theoretical analysis shows that our proposed methods have the same stability and numerical accuracy as the standard DG method in the inner region. An additional nonlinear limiter is designed to prevent spurious oscillations if a shock is near the boundary. Numerical results indicate that our methods achieve optimal numerical accuracy for smooth problems and do not introduce additional oscillations in discontinuous problems.

Key words: Discontinuous Galerkin method; Hyperbolic conservation laws; Numerical boundary conditions; Inverse Lax-Wendroff method; High order accuracy; Stability analysis.

1 Introduction

Many practical problems in computational fluid dynamics can be described by hyperbolic conservation laws on complex geometries. To avoid the difficulty in generating high-quality meshes in many problems, such as complex domains or moving interfaces problems, using unfitted meshes is an common approach. In this framework, the computational mesh is composed of the embedded Cartesian cells within the domain internally, and irregular "cut" cells are produced by the boundary interacting with each regular grid cell . Since the boundary of the computational area does not coincide with the physical boundary, boundary conditions on computational domain must be carefully treated to obtain optimal

[1]School of Mathematical Sciences, University of Science and Technology of China, Hefei, Anhui 230026, China. E-mail: ylcg@mail.ustc.edu.cn.

[2]School of Mathematical Sciences, University of Science and Technology of China, Hefei, Anhui 230026, China. E-mail: lishun@mail.ustc.edu.cn

[3]School of Mathematical Sciences, University of Science and Technology of China, Hefei, Anhui 230026, China. E-mail: jiangy@ustc.edu.cn. Research is supported in part by NSFC grant 12271499 and Cyrus Tang Foundation.

[4]Division of Applied Mathematics, Brown University, Providence, RI 02912, USA. E-mail: chi-wang_shu@brown.edu. Research is supported in part by NSF grant DMS-2309249.

[5]School of Mathematical Sciences, University of Science and Technology of China, Hefei, Anhui 230026, China. E-mail: mpzhang@ustc.edu.cn. Research is supported in part by NSFC grant 12126604.

convergence rate. On the other hand, the irregular cut cells on the domain boundary may be arbitrary small. It will bring the so called "small-cell" problem, leading to a severe time step restriction for time-dependent problems. In this paper, we develop a high order discontinuous Galerkin (DG) method for one-dimensional hyperbolic conservation laws on unfitted meshes. In particular, a novel boundary treatment would be carefully designed to overcome the "small-cell" problem.

The DG method is a class of finite element methods that employs discontinuous basis functions. The DG method was first introduced by Reed and Hill [26]. Significant breakthroughs were gained by the so called RKDG methods for solving hyperbolic conservation laws [5, 6, 7, 8, 9], in which the DG discretization is only used for the spatial variables, and the time discretization is achieved by the Runge-Kutta methods. DG methods offer numerous advantages compared to classical finite element methods and have found wide applications in various fields. For more details on DG, we refer to [28]. DG methods on unfitted meshes have been addressed in the literature, and the small-cell problem can be avoided in distinct ways, such as implicit time stepping [1], cell merging or cell agglomeration [25, 24, 27], stabilization with ghost penalty [2, 15, 30, 31, 14, 12, 11], state redistribution [13] and shifted boundary method [29]. In this paper, we are going to give a novel boundary treatment based on the inverse Lax-Wendroff (ILW) principle for DG method.

The ILW method was developed to give values of ghost points for high order finite difference methods on a Cartesian mesh. The first work for hyperbolic conservation law equations was introduced by Tan and Shu [32], and applied to inviscid compressible fluids. Given the inflow boundary conditions, they transformed the normal derivative into time derivative and tangential derivative via using the partial differential equation repeatedly. Then, values at the ghost points near the boundary were obtained by a Taylor expansion with these normal derivatives. To avoid the heavy algebra of the ILW procedure for nonlinear systems especially in the high-dimensional cases, the simplified ILW (SILW) method was proposed in [34], in which high order derivatives are constructed by extrapolation directly. In [23], Lu et al. proposed an ILW method to deal with problems with changing wind direction on the boundary. The numerical fluxes near the boundary were suitably modified so that strict conservation of the total mass is achieved [10] . In additional, the ILW method has been subsequently applied to various other time-dependent problems, including the advection equations and viscous compressible fluids [22], and moving boundary problems [33, 21, 4, 20]. Linear stability of (S)ILW methods was studied in [18, 19, 16, 17].

Owing to the advantages, including the ability to achieve arbitrary orders of accuracy and efficiently avoid the small-cell problem, extensive and in-depth research has been conducted on the ILW method within the context of finite difference schemes. In this paper, we want to incorporate the ILW boundary treatment with DG methods, which could be beneficial for DG on unfitted meshes and complex boundary problems. As an initial step and a proof of concept, in this paper we restrict our attention to 1D hyperbolic conservation laws on an unfitted mesh. The standard DG method would be used on the interior elements. Meanwhile, we treat these small boundary elements

as special virtual elements and apply the principles of the (S)ILW treatment to design the numerical boundary conditions. This approach completely mitigates the issue of small time steps, ensuring that our method maintains the same explicit time step as the standard DG method. Moreover, we find that the boundary treatment will break the local conservative property on those virtual elements, leading to a significant impact on the magnitude of errors. An additional post-processing technique is proposed to preserve the local conservation. In the presence of shocks propagating from the inflow boundary, we develop an extra nonlinear limiter to prevent oscillations and maintain accuracy.

The rest of the paper is organized as follows: In Section 2, we develop the high order boundary treatment based on (S)ILW principles for both scalar problems and linear systems. Theoretical analysis, including stability analysis and error estimation, is presented in Section 3. Numerical examples are presented in Section 4 to demonstrate the effectiveness of our approach. Concluding remarks are given in Section 5.

2 (S)ILW method for 1D conservation laws

We start our discussion with the one dimensional scalar hyperbolic conservation law on the physical domain $\Omega = [a, b]$,

$$\begin{cases} u_t + f(u)_x = 0, & x \in (a, b), \quad t > 0, \\ u(x, 0) = u_0(x), & x \in [a, b], \\ u(a, t) = g(t), & t > 0. \end{cases} \tag{2.1}$$

We assume that $f'(u(a, t)) \geq \sigma > 0$ and $f'(u(b, t)) \geq 0$ for $t > 0$. Under this assumption, the left boundary $x = a$ is an inflow boundary where a Dirichlet boundary condition is imposed, and the right boundary $x = b$ is an outflow boundary where no boundary condition is needed.

Here, we assume the domain is partitioned by uniform mesh (see Fig. 1)

$$a + \delta_1 = x_{\frac{1}{2}} < x_{\frac{3}{2}} < \cdots < x_{N+\frac{1}{2}} = b - \delta_2.$$

with uniform mesh size $h = (b - a - \delta_1 - \delta_2)/N$, $N \in \mathbb{N}$. Note that the physical boundary is allowed to be not coinciding with grid points, and the distance $\delta_{1,2}$ can be any non-negative number between 0 and h. Let $I_j = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$ denote an element with length of h, $j = 1, \ldots, N$, and $\widetilde{\Omega} = [x_{\frac{1}{2}}, x_{N+\frac{1}{2}}] = \cup_{i=1}^{N} I_j$ is called the *computational domain*, in which the numerical solutions are computed via standard DG method. The non-overlapping *cut cell* between Ω and $\widetilde{\Omega}$ is denoted by $\tilde{I}_0 = [a, x_{\frac{1}{2}}]$ and $\tilde{I}_{N+1} = [x_{N+\frac{1}{2}}, b]$.

Let $P^k(I_j)$ be the space of polynomials of degree at most $k \geq 0$ on I_j, and the DG finite element space is defined as

$$V_h^k = \{v : v|_{I_j} \in P^k(I_j),\ j = 1, \ldots, N\}. \tag{2.2}$$

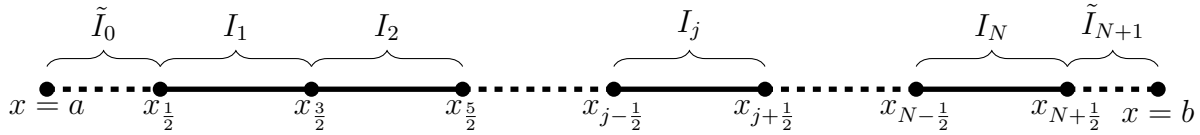


Figure 1: Domain decomposition.

The semi-discrete DG method for solving (2.1) is defined as follows: find the unique function $u_h(\cdot,t) \in V_h^k$ such that for all test functions $v_h \in V_h^k$ and all $1 \le j \le N$, we have

$$\int_{I_j} (u_h)_t v_h dx - \int_{I_j} f(u_h)(v_h)_x dx + \hat{f}_{j+\frac{1}{2}} v_h(x^-_{j+\frac{1}{2}}) - \hat{f}_{j-\frac{1}{2}} v_h(x^+_{j-\frac{1}{2}}) = 0. \tag{2.3}$$

Here, $w(x^-_{j+\frac{1}{2}},t)$ and $w(x^+_{j+\frac{1}{2}},t)$ are the left and right limits of the discontinuous solution $w(x,t)$ at the interface $x_{j+\frac{1}{2}}$, respectively. $\hat{f}_{j+\frac{1}{2}} = \hat{f}(u_h(x^-_{j+\frac{1}{2}},t), u_h(x^+_{j+\frac{1}{2}},t))$ $(j = 1,2,\cdots,N-1)$ is the monotone numerical flux as in the standard DG method, which satisfies the following conditions:

- It is consistent with the flux $f(u)$, i.e. $\hat{f}(u,u) = f(u)$.
- It is at least locally Lipschitz continuous with respect to both arguments.
- It is a nondecreasing function of its first argument, and a nonincreasing function of its second argument.

In particular, the Lax-Friedrichs flux is one of the most commonly used monotone fluxes

$$\hat{f}^{LF}(u^-,u^+) = \frac{1}{2}\left(f(u^-) + f(u^+) - \alpha(u^+ - u^-)\right), \quad \alpha = \max_u |f'(u)|. \tag{2.4}$$

The semi-discrete DG scheme (2.3) can be rewritten as the first-order ordinary differential equation (ODE) system $u_t = \mathcal{L}(u)$, where the operator $\mathcal{L}(u)$ arises from spatial discretization. Third order TVD Runge-Kutta method is used for time discretization.

$$\begin{aligned} u^{(1)} &= u^n + \Delta t \mathcal{L}(u^n), \\ u^{(2)} &= \frac{3}{4}u^n + \frac{1}{4}u^{(1)} + \frac{1}{4}\Delta t \mathcal{L}(u^{(1)}), \\ u^{n+1} &= \frac{1}{3}u^n + \frac{2}{3}u^{(1)} + \frac{2}{3}\Delta t \mathcal{L}(u^{(2)}). \end{aligned} \tag{2.5}$$

It has been demonstrated in [7] that to ensure the stability of the RKDG method with P^k elements for problems with periodic boundary conditions, the time step Δt must satisfy the condition $\Delta t \le \frac{1}{2k+1}\frac{h}{\alpha}$. For the hyperbolic problem (2.1) with Dirichlet boundary conditions, [3] indicates that the following match of time is necessary to maintain the

third-order accuracy of (2.5).

$$\begin{aligned} u^n &\sim g(t_n), \\ u^{(1)} &\sim g(t_n) + \Delta t g'(t_n), \\ u^{(2)} &\sim g(t_n) + \frac{\Delta t}{2} g'(t_n) + \frac{(\Delta t)^2}{4} g''(t_n). \end{aligned} \tag{2.6}$$

Note that the RKDG scheme can also be used to compute numerical solutions on two small cells $\tilde{I}_0$ and $\tilde{I}_{N+1}$. However, time step Δt would then be restricted to be proportional to $\min(\delta_1, \delta_2)$ to ensure stability, which is referred to as the *cut-cell* problem. In this paper, we will employ the idea of the ILW method to construct values on $\tilde{I}_0$ and $\tilde{I}_{N+1}$ through boundary treatment, and further define the numerical fluxes $\hat{f}_{1/2}$ and $\hat{f}_{N+1/2}$. Through this approach, the scheme can employ time step as the standard DG method, meaning that Δt is only proportional to h and independent of $\delta_{1,2}$.

2.1 (S)ILW boundary treatment at the inflow boundary

In this section, we focus on the inflow boundary $x = a$ and cell $\tilde{I}_0 = [a, a + \delta_1]$. A high-order ILW method is designed to construct a polynomial $p(x, t) \in P^k(\tilde{I}_0)$. After that, we can set the numerical flux $\hat{f}_{\frac{1}{2}} = \hat{f}(p(x_{\frac{1}{2}}, t), u_h(x^+_{\frac{1}{2}}, t))$.

Contrary to the idea of calculating the time derivatives with the space derivatives in the Lax-Wendroff scheme, the inverse Lax-Wendroff method uses the differential equation to calculate space derivatives with time derivatives. Thus, we can obtain the spatial derivatives on the boundary $x = a$ by employing the PDE and boundary conditions repeatedly. For example

$$\begin{aligned} \partial_x^{(0)} u|_{x=a} =& g(t), \\ \partial_x^{(1)} u|_{x=a} =& \frac{g_l'(t)}{-f'(g_l(t))}, \\ \partial_x^{(2)} u|_{x=a} =& \frac{f'(g_l(t)) g_l''(t) - 2f''(g_l(t)) g_l'(t)^2}{f'(g_l(t))^3}, \\ &\vdots \end{aligned} \tag{2.7}$$

Then, we can obtain the polynomial $p(x, t)$ with degree at most k by setting

$$\partial_x^{(m)} p|_{x=a} = \partial_x^{(m)} u|_{x=a}, \quad m = 0, 1, \ldots, k.$$

In practice, the polynomial $p(x, t)$ can be written in the form of Taylor expansion:

$$p(x, t) = \sum_{m=0}^{k} \frac{1}{m!} (x - a)^m \partial_x^{(m)} u|_{x=a}. \tag{2.8}$$

This method can achieve arbitrary high order accuracy. However, the formula may be very complicated for high order spatial derivatives. To avoid the very heavy algebra

of the above method, we employ the idea of simplified ILW (SILW) method in which information from the interior interval would be used.

Denote the cell average of u_h on I_j by

$$\bar{u}_j = \frac{1}{h}\int_{I_j} u_h(x)dx,$$

and the cell averages of derivatives by

$$(\bar{u}_x)_j = \frac{1}{h}\int_{I_j} (u_h)_x(x)dx, \quad (\bar{u}_{xx})_j = \frac{1}{h}\int_{I_j} (u_h)_{xx}(x)dx.$$

Then, we can construct the polynomial $p(x)$ of degree $k \geq 1$ with information on the boundary $x = a$ and on the first element I_1. Strategies for P^k, $k \leq 3$, are listed in Table 2.1. For example, SILW-1 means the unique polynomial $p(x) \in P^k(\tilde{I}_0)$ satisfies

$$\partial_x^{(m)} p|_{x=a} = \partial_x^{(m)} u|_{x=a}, \quad m = 0, 1, \ldots, k-1, \tag{2.9a}$$

$$\int_{I_1} p(x,t)dx = h\bar{u}_1. \tag{2.9b}$$

SILW-2 gives the constraint equations

$$\begin{aligned} &\partial_x^{(m)} p|_{x=a} = \partial_x^{(m)} u|_{x=a}, \quad m = 0, 1, \ldots, k-2, \\ &\int_{I_1} p(x,t)dx = h\bar{u}_1, \quad \int_{I_1} p_x(x,t)dx = h(\bar{u}_x)_1. \end{aligned} \tag{2.10}$$

Table 1: Strategies which can be used in (S)ILW.

	Strategy	info on the boundary	info from I_1
$k=0$	ILW	$u(a,t)$	
$k=1$	ILW	$u(a,t), u_x(a,t)$	
	SILW-1	$u(a,t)$	$\bar{u}_1$
$k=2$	ILW	$u(a,t), u_x(a,t), u_{xx}(a,t)$	
	SILW-1	$u(a,t), u_x(a,t)$	$\bar{u}_1$
	SILW-2	$u(a,t)$	$\bar{u}_1, (\bar{u}_x)_1$
$k=3$	ILW	$u(a,t), u_x(a,t), u_{xx}(a,t), u_{xxx}(a,t)$	
	SILW-1	$u(a,t), u_x(a,t), u_{xx}(a,t)$	$\bar{u}_1$
	SILW-2	$u(a,t), u_x(a,t)$	$\bar{u}_1, (\bar{u}_x)_1$

Remark 2.1. *To construct a polynomial $p(x,t) \in P^k$, we can select $k+1$ constraints from two sets of information (boundary information and information from the first element I_1) to form a system of constraint equations. It is crucial to note that the selection of constraints should not be arbitrary. In our experiment, constraints from each set are progressively added from lower order to higher order.*

Remark 2.2. *Obviously, the less boundary information $\partial_x^{(m)} u|_{x=a}$ is used, the more efficient the scheme is. However, it is essential to avoid utilizing too few constraints provided by the boundary information. For example, considering $k = 3$, if we employ three constraints from I_1 and only take $u(a,t)$ on boundary, i.e.,*

$$\begin{cases} p(a,t) = u|_{x=a}, \\ \int_{I_1} p(x,t)dx = h\bar{u}_1, \\ \int_{I_1} p_x(x,t)dx = h(\bar{u}_x)_1, \\ \int_{I_1} p_{xx}(x,t)dx = h(\bar{u}_{xx})_1, \end{cases}$$

the constraint equations will lead to singularity and no solution exists when δ approaches 0. Additionally, selecting too little boundary information will directly affect the stability of the scheme. Theoretical analysis of the SILW method for high order finite difference schemes was given in [18, 19, 16, 17]. We will discuss the stability and accuracy of the SILW boundary treatment for the DG method in this paper later.

Numerical results show that the schemes can always achieve $(k+1)$-th order accuracy. However, the magnitude of error is closely related to the parameter δ_1/h. For example, we use the RKDG method with P^k basis, $k = 0, 1, 2, 3$, to solve the following equation:

$$\begin{cases} u_t + u_x = 0, \quad 0 < x < 2\pi, t > 0, \\ u(x,0) = -\sin(x), \quad 0 < x < 2\pi, \\ u(0,t) = \sin t. \end{cases} \tag{2.11}$$

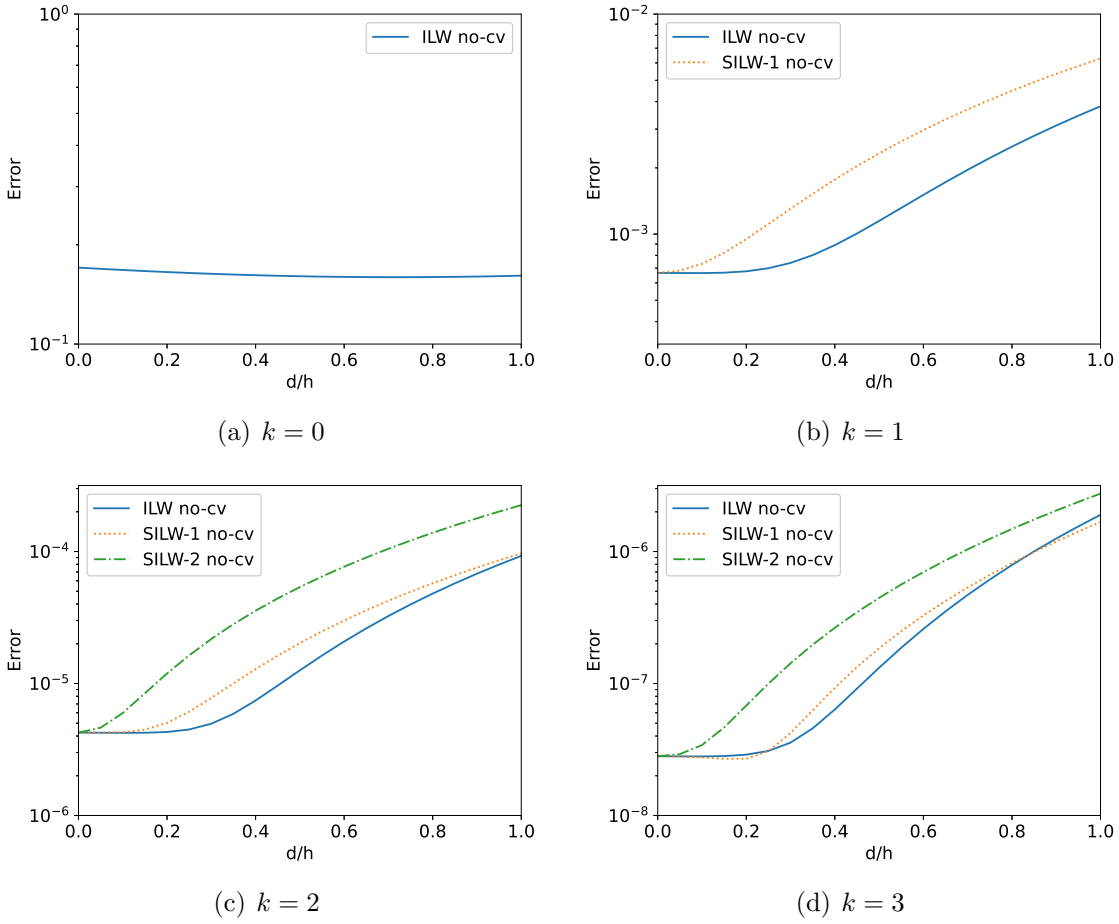
This example has the exact solution $u(x,t) = \sin(t-x)$. We divide the domain with $N = 80$, $\delta_2 = 0$ and varying $\delta = \delta_1 \in [0, h)$. Since the numerical flux $\hat{u}_{i+1/2} = u_h|^-_{i+1/2}$, $i = 1, \ldots, N$, no additional processing is required for the outflow boundary $x = b$. The proposed (S)ILW method is used for the inflow boundary condition. The L_2 errors on the computational domain $\tilde{\Omega}$ at the final time $T = 3$ are plotted in Fig. 2, showing that the error increases significantly when δ/h approaches 1 with $k \geq 1$.

We find that this phenomenon is caused by the loss of conservation of the reconstructed polynomials $p(x,t) \in P^k(\tilde{I}_0)$. To overcome this disadvantage, we propose a post-processing based on the local conservation in the following subsection.

2.2 Post-processing based on local conservation

One significant advantage of the DG method is that it can ensure the local conservation properties: taking test function $v_h = 1$ in classic DG scheme (2.3), we have that

$$\int_{I_j} (u_h)_t dx + \hat{f}_{j+\frac{1}{2}} - \hat{f}_{j-\frac{1}{2}} = 0, \quad j = 1, \ldots, N. \tag{2.12}$$



(a) $k = 0$ (b) $k = 1$

(c) $k = 2$ (d) $k = 3$

Figure 2: Errors of the test example (2.11) with different δ/h. "no-cv" represents that no conservation post-processing has been applied.

This tells us the local conservative of the DG scheme. Note that $p(x,t) \in P^k(\tilde{I}_0)$ is constructed via the (S)ILW method, and the local conservative property fails on this cut cell,

$$\frac{d}{dt}\int_a^{a+\delta_1} p(x,t)dx + \hat{f}_{\frac{1}{2}} - f(g(t)) \neq 0, \tag{2.13}$$

where $\hat{f}_{\frac{1}{2}} = \hat{f}(p(a+\delta_1,t), u_h(x_{\frac{1}{2}}^+))$.

In the following, we introduce the post-processing based on the local conservation for $k \geq 1$. We primarily focus on the treatment of inflow boundaries. The treatment of outflow boundaries will be addressed in the next subsection. For the inflow boundary, the numerical flux $\hat{f}_{\frac{1}{2}}$ would be modified via the conservation

$$\hat{f}^c_{\frac{1}{2}} \triangleq f(g(t)) - \frac{d}{dt}\int_a^{a+\delta_1} p(x,t)dx. \tag{2.14}$$

If $p(x,t) \in P^k$ is obtained by the ILW method (2.8), then

$$\begin{aligned}
\hat{f}^c_{\frac{1}{2}} =& f(g(t)) - \int_a^{a+\delta_1} \partial_t p(x,t) dx \\
=& f(g(t)) - \sum_{m=0}^{k} \int_a^{a+\delta_1} \frac{1}{m!}(x-a)^m \partial_t \partial_x^{(m)} u|_{x=a} dx \\
=& f(g(t)) - \sum_{m=0}^{k} \frac{\delta_1^{m+1}}{(m+1)!} \partial_t \partial_x^{(m)} u|_{x=a} \\
=& f(g(t)) + \sum_{m=0}^{k} \frac{{\delta_1}^{m+1}}{(m+1)!} \partial_x^{(m+1)} f(u)|_{x=a}.
\end{aligned} \tag{2.15}$$

Here we define the notation $\Pi_s^k[w](x)$ to represent the k-th order Taylor polynomial of $w(x)$

$$\Pi_s^k[w](x) = \sum_{m=0}^{k} \frac{1}{m!}(x-s)^m \partial_x^{(m)} w|_{x=s}. \tag{2.16}$$

Notice that the right-hand side of (2.15) is precisely the Taylor expansion of $f(u)$,

$$\hat{f}^c_{\frac{1}{2}} = \Pi_a^{k+1}[f(u)](a+\delta_1) \approx f(u(a+\delta_1, t)).$$

This indicates that we actually use the Taylor expansion of function $f(u)$ at the physical boundary to approximate the numerical flux at the computational boundary, which provides another perspective on the validity of the ILW method.

If $p(x,t)$ is obtained by the SILW method, the interior value would be used, whose time derivatives could be given via DG scheme on I_1. In particular, we can directly set $v_h = 1$ to get the time derivative of the cell average

$$\frac{d}{dt}\bar{u}_1 = \frac{1}{h}(\hat{f}^c_{\frac{1}{2}} - \hat{f}_{\frac{3}{2}}). \tag{2.17}$$

Higher order information can be obtained by taking special test function in the DG scheme. But specific expression is dependent on the choice of DG basis functions and the degree k. For instance, let ϕ_i denote the i-th orthogonal Legendre polynomials basis function in I_1,

$$\frac{d}{dt}(\bar{u}_x)_1 = \begin{cases} \frac{12}{h^3}\int_{I_1} f(u_h)dx - \frac{6}{h^2}(\hat{f}^c_{\frac{1}{2}} + \hat{f}_{\frac{3}{2}}), & (k=1,2) \\ \frac{12}{h^3}\int_{I_1} f(u_h)dx + \frac{14}{h^2}\int_{I_1} f(u_h(x,t))\phi_3'(x)\,dx - \frac{20}{h^2}(\hat{f}^c_{\frac{1}{2}} + \hat{f}_{\frac{3}{2}}), & (k=3) \end{cases} \tag{2.18}$$

Plugging them in the conservation form (2.14), we can get an algebraic equation of $\hat{f}^c_{1/2}$. The numerical flux can be obtained by solving this equation.

Here, we give the details of the SILW-1 method as an example. Utilizing (2.9a), we know that $p(x,t) \in P^k$ has the form

$$\begin{aligned} p(x,t) =& \sum_{m=0}^{k-1} \frac{1}{m!}(x-a)^m \partial_x^{(m)} u(a,t) + R(t)(x-a)^k \\ =& \Pi_a^{k-1}[u](x,t) + R(t)(x-a)^k. \end{aligned} \tag{2.19}$$

Plugging in (2.9b), we can have

$$R(t) = \frac{k+1}{(a+\delta_1+h)^{k+1} - (a+\delta_1)^{k+1}} \left(h\bar{u}_1 - \int_{I_1} \Pi_a^{k-1}[u](x,t)\, dx \right). \tag{2.20}$$

Utilizing the expression (2.14) and (2.19) - (2.20), we can obtain the algebraic equations regarding the numerical flux,

$$\begin{aligned} \hat{f}^c_{\frac{1}{2}} =& f(g(t)) - \frac{d}{dt} \int_a^{a+\delta_1} \left(\Pi_a^{k-1}[u](x,t) + R(t)(x-a)^k \right) dx \\ =& f(g(t)) - \int_a^{a+\delta_1} \partial_t \Pi_a^{k-1}[u](x,t)\, dx - \frac{\delta_1^{k+1} \left(\hat{f}^c_{\frac{1}{2}} - \hat{f}_{\frac{3}{2}} - \frac{d}{dt} \int_{I_1} \Pi_a^{k-1}[u](x,t)\, dx \right)}{(a+\delta_1+h)^{k+1} - (a+\delta_1)^{k+1}}. \end{aligned} \tag{2.21}$$

Solve the above equation,

$$\begin{aligned} \hat{f}^c_{\frac{1}{2}} =& \sum_{m=0}^{k} \left[\frac{\partial_x^{(m)} f(u(a,t))}{m!} \right] \delta_1^m + \frac{\delta_1^{k+1}}{(\delta_1+h)^{k+1}} \left\{ \hat{f}_{\frac{3}{2}} - \sum_{m=0}^{k} \left[\frac{\partial_x^{(m)} f(u(a,t))}{m!} \right] (\delta_1+h)^m \right\} \\ =& \Pi_a^k[f(u)](a+\delta_1) + \frac{\delta_1^{k+1}}{(\delta_1+h)^{k+1}} \left\{ \hat{f}_{\frac{3}{2}} - \Pi_a^k[f(u)](a+\delta_1+h) \right\}. \end{aligned} \tag{2.22}$$

The spatial derivatives at $x = a$ can be conversed to the mixed derivatives by using the PDE,

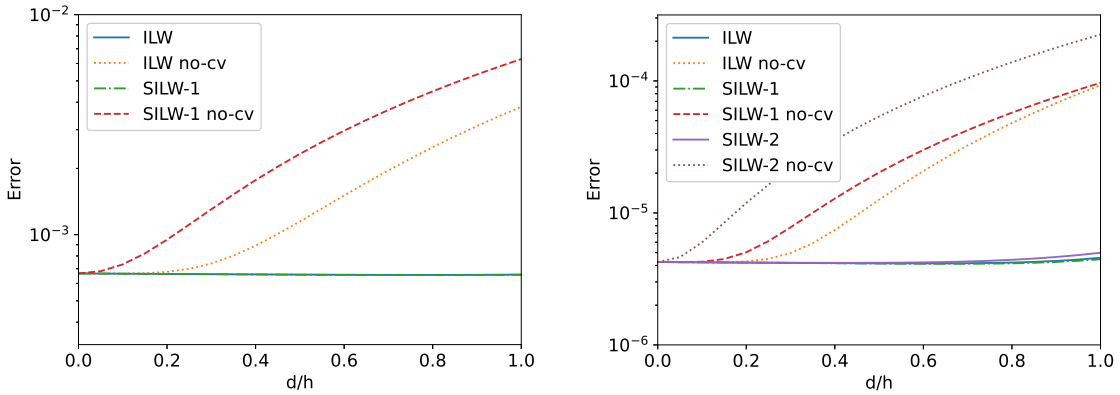
$$\partial_x^{(m)} f(u(a,t)) = -\partial_x^{(m-1)} \partial_t u(a,t).$$

After that, all terms including time derivatives can be obtained via the ILW procedure (2.7).

To illustrate the corrective impact of the post-processing on the numerical flux, we provide Fig. 3 as a comparison to Fig. 2. We can observe that after post-processing, the magnitude of the error no longer exhibits significant variations as δ/h approaches 1.

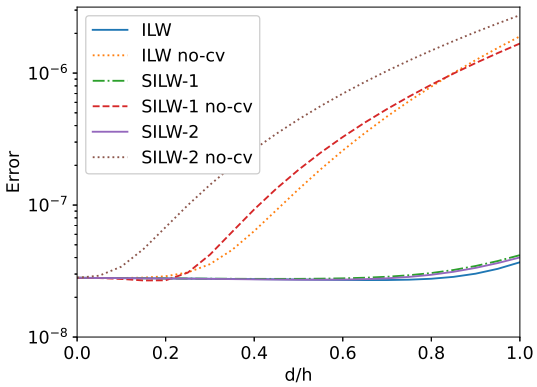
2.3 Nonlinear limiter for shock waves

Until this point, the numerical flux is designed with the assumption that $u(x)$ is smooth enough. Significantly, shock wave propagating from the inflow boundary will lead to numerical oscillations and disrupt the stability of the scheme. Therefore, a nonlinear limiter is required to prevent oscillations and ensure the stability in the presence of strong



(a) $k = 1$

(b) $k = 2$



(c) $k = 3$

Figure 3: Errors of test example (2.11) with different δ/h. "no-cv" represents that no conservation post-processing has been applied.

discontinuities in $\tilde{I}_0$. Numerical experiments demonstrate that the first order numerical flux $\hat{f}_{\frac{1}{2}} = f(g(t))$ (this is the numerical flux of $k = 0$ without conservative modification) always has a good performance in avoiding spurious oscillations. Thus, we design an extra limiting procedure as follows, to switch the high-order numerical flux $\hat{f}^c_{\frac{1}{2}}$ to a first-order flux $f(g(t))$ when encountering such situations, while retaining high-order accuracy in smooth regions.

We use the following notations to present the variation in $\tilde{I}_0$ and I_1,

$$\begin{aligned}\Delta_1 =& |f(u(a,t)) - \hat{f}^c_{\frac{1}{2}}|,\\ \Delta_2 =& |f(u_h(x^+_{\frac{1}{2}},t)) - f(u_h(x_{\frac{1}{2}}+\delta)|.\end{aligned}$$

Next, we define a discontinuous index $\theta \in [0,1]$ as follows,

$$\theta = \eta\left(\left(\frac{\Delta_1 - \Delta_2}{\Delta_1 + \varepsilon}\right)^4\right), \tag{2.23}$$

where $\eta(x) = \frac{1}{2}(1 + \tanh(20(x - 1/2))$ and ε is a small number used to avoid division by zero. We choose $\varepsilon = 10^{-16}$ in practice. Finally, $\hat{f}_{\frac{1}{2}}$ will be modified as a convex combination of $\hat{f}^c_{\frac{1}{2}}$ and $f(g(t))$,

$$\hat{f}^{mod}_{\frac{1}{2}} = (1 - \theta)\hat{f}^c_{\frac{1}{2}} + \theta\, f(g(t)). \tag{2.24}$$

Such modification will lead to the following effects:

1. If $f(u)$ is smooth, both Δ_1 and Δ_2 are about $|f_x(u(x_{\frac{1}{2}}, t))|\delta + \mathcal{O}(\delta^2)$. Thus, $\theta = \eta(\mathcal{O}(\delta^4)) = \mathcal{O}(\delta^4)$ approaches to 0 and $\hat{f}^{mod}_{\frac{1}{2}} = \hat{f}^c_{\frac{1}{2}} + \mathcal{O}(\delta^4)$. Furthermore, the modification (2.24) can maintain the original order of accuracy for $k \leq 3$.

2. If a shock comes from the boundary, $\Delta_1 = \mathcal{O}(1)$. Consequently, $\theta \approx \eta(1)$ and $\hat{f}^{mod}_{\frac{1}{2}}$ would degenerate to $f(g(t))$, which could avoid numerical oscillation around boundary.

It is remarkable that both $\hat{f}^c_{\frac{1}{2}}$ and $f(g(t))$ give the result that the magnitude of the error does not depend on δ/h significantly. This leads to the fact that the flux $\hat{f}^{mod}_{\frac{1}{2}}$ has the same property.

2.4 Boundary treatment at the outflow boundary

In this subsection, we are concerned about the treatment of outflow boundaries. Based on the characteristics of the conservation law, information on the downwind side should not affect the interior of the computational domain. Hence, for one-dimensional scalar equations, we can directly compute the numerical flux using values on the interior side of the boundary $\hat{f}_{N+1/2} = f(u_h(x^-_{N+1/2}), t)$, without considering the information at the physical boundary $x = b$.

However, when solving systems of equations, inflow and outflow boundary conditions are often mixed at the boundary. The numbers of inflow boundary conditions and outflow boundary conditions depend on how many eigenvalues of the Jacobian at the boundary are positive and negative respectively. Different components of the solution may interact at the physical boundary through boundary conditions. Therefore, it is necessary to obtain information about these outflow components at the physical boundary.

Let us consider the treatment of the outflow boundary $x = b$. First, we take $p(x, t)$ as the direct extension of the polynomial u_h on I_N to the virtual cell $\tilde{I}_{N+1} = [x_{N+\frac{1}{2}}, b]$.

$$p(x, t) = u_h|_{I_N}(x), \quad x \in \tilde{I}_{N+1}.$$

Meanwhile, we still have that $\hat{f}_{N+1/2} = f(u_h(x^-_{N+1/2}), t)$. However, we face the problem of violating conservation properties once again,

$$\frac{d}{dt}\int_{x_{N+\frac{1}{2}}}^{b} p(x, t)dx + f(p(b, t)) - \hat{f}_{N+\frac{1}{2}} \neq 0. \tag{2.25}$$

In order to preserve conservation on the cell $\tilde{I}_{N+1}$, we modify the point value at $x = b$ by adding a constant $p^*(b,t) = p(b,t) + C$,

$$\frac{d}{dt}\int_{x_{N+\frac{1}{2}}}^{b} p(x,t)dx + f(p^*(b,t)) - \hat{f}_{N+\frac{1}{2}} = 0. \tag{2.26}$$

Ultimately, we obtain information at the outflow physical boundary through the following expressions,

$$u_h|_{x=b} = p^*(c,t), \quad \partial_x^{(m)} u_h|_{x=b} = \partial_x^{(m)} p(x,t)|_{x=b},\ m = 1,\dots,k. \tag{2.27}$$

2.5 The (S)ILW scheme for linear systems

Consider the following system on the physical domain $\Omega = [a,b]$,

$$\begin{cases} \mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = 0, & x \in (a,b), \quad t > 0 \\ \mathbf{U}(x,0) = \mathbf{U}_0(x), & x \in [a,b], \end{cases} \tag{2.28}$$

where $\mathbf{F}(\mathbf{U}) = \mathbf{A}\mathbf{U}$ and $\mathbf{A} \in \mathbb{R}^{n\times n}$ is a constant matrix. This system is hyperbolic meaning the matrix $\mathbf{A}$ is diagonalizable with real eigenvalues. For problems involving systems of equations, the required boundary conditions at the boundaries are determined by the sign of eigenvalues. Assume the eigenvalues of $\mathbf{A}$ have the property

$$\lambda_1 \le \cdots \lambda_{n_1} \le 0 < \lambda_{n_1+1} \le \cdots \le \lambda_n.$$

Then,

$$\mathbf{A} = \mathbf{P}^{-1}\begin{pmatrix} \mathbf{\Lambda}^{(L)} & \\ & \mathbf{\Lambda}^{(R)} \end{pmatrix}\mathbf{P}, \quad \mathbf{P} \in \mathbb{R}^{n\times n}, \mathbf{\Lambda}^{(L)} \in \mathbb{R}^{n_1\times n_1}, \mathbf{\Lambda}^{(R)} \in \mathbb{R}^{n_2\times n_2},\ n = n_1 + n_2.$$

with $\mathbf{\Lambda}^{(L)} = diag(\lambda_1,\cdots,\lambda_{n_1})$ and $\mathbf{\Lambda}^{(R)} = diag(\lambda_{n_1+1},\cdots,\lambda_n)$. In this case, we require n_2 boundary conditions at the left boundary $x = a$ and n_1 conditions at the right boundary $x = b$.

As an example, we are concerned with the left boundary $x = a$ imposed with n_2 Dirichlet boundary conditions.

$$\mathbf{U} = \begin{pmatrix} \mathbf{U}^{(1)} \\ \mathbf{U}^{(2)} \end{pmatrix}, \quad \mathbf{U}^{(1)}(a,t) = \mathbf{G}(t) \in \mathbb{R}^{n_2}, \quad \forall\, t > 0.$$

Denoted $\mathbf{V}$ as the characteristic variables $\mathbf{V} = \mathbf{P}\mathbf{U} = (\mathbf{V}^{(L)}, \mathbf{V}^{(R)})^T$, where $\mathbf{V}^{(L)} \in \mathbb{R}^{n_1}$ and $\mathbf{V}^{(R)} \in \mathbb{R}^{n_2}$ are variables moving to the left and right respectively. The proposed (S)ILW boundary treatment for scalar problems will be applied on each component of $\mathbf{V}$ directly. Particularly, Dirichlet boundary conditions are imposed on conservative variables $\mathbf{U}$ rather than the characteristic variables $\mathbf{V}$. Hence, $\mathbf{V}^{(L)}$ and $\mathbf{V}^{(R)}$ will change information at $x = a$, and they should be handled simultaneously.

To be more specific, the (S)ILW boundary treatment for systems is given as follows. Here, we use the superscript $*$ to present values located at $x = a$ to ease the notation.

1. First, we consider the outgoing characteristic components $\mathbf{V}^{(L)}$. The algorithm introduced in subsection 2.4 should be applied on each component v_i, $i = 1, \ldots, n_1$. Then, we obtain the values and spatial derivatives at $x = a$,

$$\mathbf{V}^{(L),*}, \mathbf{V}_x^{(L),*}, \mathbf{V}_{xx}^{(L),*}, \ldots,$$

Moreover, by utilizing the Lax-Wendroff procedure on each v_i with the equation $(v_i)_t + \lambda_i (v_i)_x = 0$, we can also obtain time derivatives at the boundary

$$\mathbf{V}_t^{(L),*}, \mathbf{V}_{tt}^{(L),*}, \ldots,$$

2. Through the following system of equations

$$\mathbf{P}^{(1)} \begin{pmatrix} \mathbf{V}^{(L),*} \\ \mathbf{V}^{(R),*} \end{pmatrix} = \mathbf{U}^{(1),*} = \mathbf{G}(t) \tag{2.29}$$

where $\mathbf{P}^{(1)}$ is a matrix formed by the first n_2 rows of the matrix $\mathbf{P}$, we can get the values of the inflow variable by solving the above system. Similarly, we can obtain higher order time derivatives from the system as well.

$$\mathbf{P}^{(1)} \begin{pmatrix} \partial_t^m \mathbf{V}^{(L),*} \\ \partial_t^m \mathbf{V}^{(R),*} \end{pmatrix} = \partial_t^m \mathbf{U}^{(1),*} = \mathbf{G}^{(m)}(t).$$

3. We construct numerical fluxes for each inflow component, which is the same as those for the scalar equations.

We would like to remark that this method can provide the desired high order accuracy. However, numerical experiments indicate that the errors grow as δ/h increases. Based on the results of scalar problems, we suspect that this should be caused by the outflow boundary treatment, even though the errors may have been reduced to a certain extent by the conservative modification. Further improvement of the outflow boundary treatment will be considered in the future.

We would also want to point out that the proposed methods can also be extended to nonlinear systems. However, the algorithm becomes more complicated, especially for the SILW method with conservative modification. Because the numerical fluxes are generally not one-sided fluxes, outflow variables and inflow variables are coupled more tightly. To do the conservative modification on the outgoing variables, numerical flux $\hat{\mathbf{F}}_{1/2}$ is needed, which also depends on the inflow values on $\tilde{I}$. Hence, solving a complex nonlinear system appears necessary.

3 Theoretical analysis

In this section, theoretical analysis, including stability analysis and error estimates, will be provided for the one-dimensional linear initial boundary value problem

$$\begin{cases} u_t + u_x = 0, & x \in (a, b), \quad t > 0, \\ u(x, 0) = u_0(x), & x \in [a, b], \\ u(a, t) = g(t), & t \geq 0. \end{cases} \tag{3.1}$$

In this case, the left boundary $x = a$ is always an inflow boundary and the right boundary $x = b$ is an outflow boundary. Here, we only need to consider the effect of treatment of inflow boundary. Again the domain is divided by the following uniform mesh with mesh size $h = x_{j+1/2} - x_{j-1/2}$ for $j = 1, \cdots, N$, and $\delta \in [0, h]$,

$$a + \delta = x_{\frac{1}{2}} < x_{\frac{3}{2}} < \cdots < x_{N+\frac{1}{2}} = b.$$

For simplicity, we assume $a = 0$ and the prescribed condition $g(t) = 0$.

In the following, DG schemes with P^k bases are considered, $k = 1, 2, 3$. The simple upwind numerical flux is used for the interior elements, i.e., $\hat{u}_{j+1/2} = u_h|^-_{j+1/2}$, $j = 1, \ldots, N$. The flux $\hat{u}_{1/2}$ can be obtained by the proposed SILW strategies with or without conservative modification. Firstly, we will prove the linear stability of all these schemes for both semi-discrete and fully discrete cases. This can be achieved via *eigenvalue spectrum visualization [35]*. Specially, the energy stability analysis and error estimates for SILW-1 with conservative modification will be discussed in Sections 3.2 and 3.3.

3.1 Eigenvalue spectrum visualization

In this subsection, we present the stability analysis by visualizing the eigenspectrum of the discretized operators. For the k-th order schemes, we set $k + 1$ equally spaced points in I_j,

$$\hat{x}_j^{(i)} = x_j + \frac{2i - k}{2(k + 1)} h, \quad i = 0, \ldots, k.$$

After picking a local Lagrange basis, the standard DG semi-discrete scheme (2.3) for linear advection equation with upwind flux in inner cells can be written as

$$\frac{d\mathbf{U}_j}{dt} = \frac{1}{h}(\mathbf{A}\mathbf{U}_{j-1} + \mathbf{B}\mathbf{U}_j), \quad j = 2, \ldots, N, \tag{3.2}$$

where $\mathbf{U}_j$ is a vector of length $k + 1$ containing the coefficients of the solution u_h corresponding to the local basis inside I_j, which also represents the values of u_h at the nodes $\{\hat{x}_j^{(i)}\}_{i=0}^k$. $\mathbf{A}$ and $\mathbf{B}$ are $(k + 1) \times (k + 1)$ constant matrices.

On the other hand, the numerical flux $\hat{u}_{\frac{1}{2}}$ can be presented with u_h in I_1 and boundary information (which is always 0), DG semi-discrete scheme (2.3) on I_1 can be sorted as follows

$$\frac{d\mathbf{U}_1}{dt} = \frac{1}{h}\mathbf{C}\mathbf{U}_1.$$

Here, $\mathbf{C}$ is a $(k + 1) \times (k + 1)$ constant matrices.

Finally, the semi-discrete scheme yields a linear system expressed in a matrix-vector form as

$$\frac{d\mathbf{U}}{dt} = \frac{1}{h}\mathbf{Q}\mathbf{U}, \tag{3.3}$$

where $\mathbf{U}$ is the vector representing the values of the solution at all nodes $\{\hat{x}_j^{(i)}, j = 1, \ldots, N, i = 0, \ldots, k\}$,

$$\mathbf{U} = ((\mathbf{U}_1)^T, (\mathbf{U}_2)^T, \cdots, (\mathbf{U}_N)^T)^T,$$

and

$$\mathbf{Q} = \begin{pmatrix} \mathbf{C} & & & & \\ \mathbf{A} & \mathbf{B} & & & \\ & \mathbf{A} & \mathbf{B} & & \\ & & \ddots & \ddots & \\ & & & \mathbf{A} & \mathbf{B} \end{pmatrix}. \tag{3.4}$$

This system contains the chosen inner scheme as well as the inflow boundary treatments.

We apply *normal mode analysis* to (3.3) to get the eigenvalue problem. Assuming a solution of the form $u(x,t) = e^{st}u_0(x)$ and $\tilde{s} = hs$, the semi-discrete scheme (3.3) yields

$$\tilde{s}\,\mathbf{U} = \mathbf{Q}\mathbf{U},$$

with $\tilde{s}$ being the eigenvalue. The semi discretization provided with the considered boundary conditions is stable if the whole eigenvalue spectrum of the coefficient matrix $\mathbf{Q}$ lies in the left half-plane, i.e., $\mathrm{Re}\,(\tilde{s}) \leq 0$. Note that eigenvalues of $\mathbf{Q}$ (3.4) are composed of the eigenvalues of $\mathbf{B}$ and $\mathbf{C}$, denoted by κ^B and κ^C respectively. It is easy to verify that κ^B satisfies the condition. Therefore, we only need to focus on κ^C in the following.

For example, considering the semi-discrete DG scheme with P^1 basis and SILW-1 boundary treatment, $\hat{u}_{\frac{1}{2}}$ and the coefficient matrix $\mathbf{C}$ have the forms as follows

$$\hat{u}_{\frac{1}{2}} = \frac{2\delta}{2\delta + h}\bar{u}_1, \quad \mathbf{C} = \frac{1}{4}\begin{pmatrix} -7 & -3 \\ 11 & -9 \end{pmatrix} + \frac{2\delta}{2\delta + h}\begin{pmatrix} 5/4 & 5/4 \\ -1/4 & -1/4 \end{pmatrix}. \tag{3.5}$$

Denote $\mu = \frac{\delta}{h} \in [0, 1)$. We can analytically calculate the eigenvalues of $\mathbf{C}$,

$$\kappa^C_{1,2} = -\frac{3\mu + 2 \pm \sqrt{9\mu^2 - 2}}{2\mu + 1}.$$

It can be verified that $\mathrm{Re}\,(\kappa^C_{1,2}) < 0$ for all $\mu \in [0, 1)$, indicating that the semi-discrete scheme is stable. Moreover, if we construct $\hat{u}_{\frac{1}{2}}$ with conservative modification, i.e. $\hat{u}^c_{\frac{1}{2}}$,

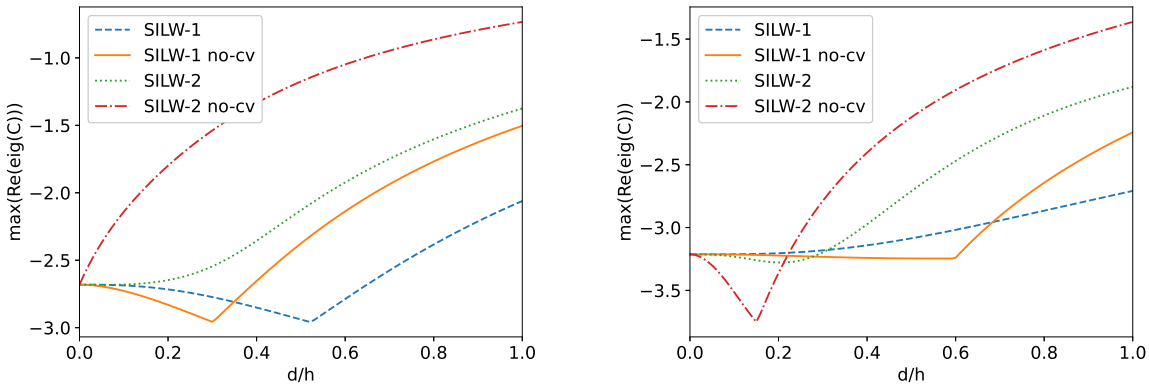
$$\hat{u}^c_{\frac{1}{2}} = \frac{\delta^2}{(\delta + h)^2}\hat{u}_{\frac{3}{2}} = \frac{\delta^2}{(\delta + h)^2}u^-_{\frac{3}{2}}, \quad \mathbf{C} = \frac{1}{4}\begin{pmatrix} -7 & -3 \\ 11 & -9 \end{pmatrix} + \frac{\delta^2}{(\delta + h)^2}\begin{pmatrix} -5/4 & 15/4 \\ 1/4 & -3/4 \end{pmatrix}. \tag{3.6}$$

In this case, we have the eigenvalues of $\mathbf{C}$ as

$$\kappa^C_{1,2} = \frac{-(3\mu^2 + 4\mu + 2 \pm \sqrt{9\mu^4 + 12\mu^3 - 2\mu^2 - 8\mu - 2})}{(\mu + 1)^2},$$

and $\mathrm{Re}\,(\kappa^C_{1,2}) < 0$ also holds for all $\mu \in [0, 1)$.

For the semi-discrete scheme with $k = 2, 3$, the results are displayed in Fig. 4 due to the complexity of analytic eigenvalue expressions. We plot the maximum of $Re(\kappa^c)$ for different δ/h. It is observed that these semi-discrete scheme are stable since real parts of the eigenvalues are all negative.



(a) $k = 2$. (b) $k = 3$.

Figure 4: Stability analysis of semi discrete DG schemes: the maximum of $Re(\kappa^C)$ for different $\mu = \delta/h \in [0, 1]$. no-cv here represents that no conservation post-processing has been applied.

We can also prove the stability of the full discrete scheme with third order RK time discretization (2.5). The full-discrete scheme yields a linear system expressed in a matrix-vector form as

$$\mathbf{U}^{n+1} = g(\mathbf{Q})\mathbf{U}^n, \tag{3.7}$$

where $g(\mathbf{X})$ is a matrix function

$$g(\mathbf{X}) = \mathbf{I} + \lambda\mathbf{X} + \frac{1}{2!}(\lambda\mathbf{X})^2 + \frac{1}{3!}(\lambda\mathbf{X})^3,$$

and $\lambda = \frac{\Delta t}{h} = \frac{1}{2k+1}$ is the CFL number. The full-discrete scheme is stable if the whole eigenvalues spectrum of $g(\mathbf{Q})$ are contained within the unit disk, i.e. $\rho(g(\mathbf{Q})) \leq 1$. It is obvious that the eigenvalues of $g(\mathbf{Q})$ are composed of the eigenvalues of $g(\mathbf{B})$ and $g(\mathbf{C})$. Similarly, we only need to focus on the part corresponding to the boundaries, i.e. the eigenvalues of $g(\mathbf{C})$.

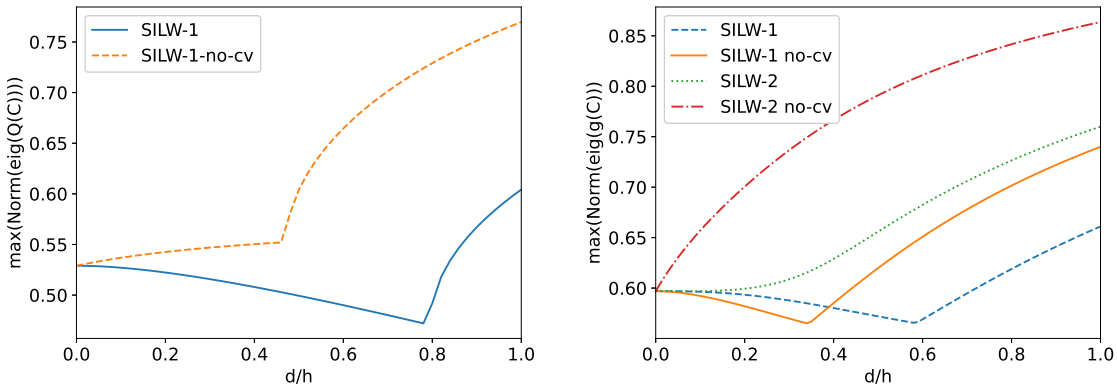
For $k = 1, 2, 3$, we plot the $\rho(g(\mathbf{C}))$ for different δ/h in Fig. 5. The results indicate that the full-discrete schemes are stable when an appropriate time discretization is employed.

3.2 Energy stability

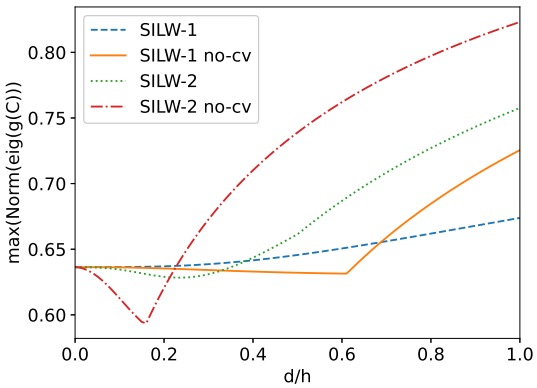
In this subsection, we establish energy stability of the semi-discrete DG scheme with the SILW-1 boundary treatment.

Proposition 3.1. *For the linear equation* (3.1) *with the homogeneous inflow boundary conditions, if we employ the upwind numerical flux internally and conservative SILW-1 flux near the inflow boundary, the corresponding semi-discrete DG scheme* (2.3) *with* $k \geq 1$ *is energy stable*

$$\frac{d}{dt}\mathcal{E}_h(t) \leq 0, \tag{3.8}$$



(a) $k = 1$ (b) $k = 2$



(c) $k = 3$

Figure 5: Stability analysis of fully discrete DG schemes: $\rho(g(\mathbf{C}))$ for different $\mu = \delta/h \in [0, 1]$. no-cv here represents that no conservation post-processing has been applied.

with the discrete energy defined as

$$\mathcal{E}_h(t) = \left[1 - \frac{1}{2^{2k+2}}\right]^{-1} \int_{I_1} (u_h)^2 dx + \int_{x_{3/2}}^{x_{N+1/2}} (u_h)^2 dx. \tag{3.9}$$

Proof. Let $v_h = u_h$ and use the upwind numerical flux in (2.3), we have that

$$\begin{aligned} 0 &= \int_{I_j} (u_h)_t u_h \, dx - \int_{I_j} u_h (u_h)_x + u^-_{j+\frac{1}{2}} u^-_{j+\frac{1}{2}} - u^-_{j-\frac{1}{2}} u^+_{j-\frac{1}{2}} \\ &= \int_{I_j} \frac{1}{2} \frac{d}{dt} (u_h)^2 \, dx - \frac{1}{2} (u^-_{j+\frac{1}{2}})^2 + \frac{1}{2} (u^+_{j-\frac{1}{2}})^2 + (u^-_{j+\frac{1}{2}})^2 - u^-_{j-\frac{1}{2}} u^+_{j-\frac{1}{2}}. \end{aligned}$$

Summing over $j \geq 2$, we can obtain that

$$\sum_{j=2}^{N} \int_{I_j} \frac{1}{2} \frac{d}{dt} (u_h)^2 \, dx \leq -\frac{1}{2} (u^-_{N+\frac{1}{2}})^2 + \frac{1}{2} (u^-_{\frac{3}{2}})^2 \leq \frac{1}{2} (u^-_{\frac{3}{2}})^2.$$

Combining the homogeneous boundary condition $g(t) = 0$, the conservative SILW-1 fluxes (2.22) for general $k \geq 1$ are in the same form,

$$\hat{u}^c_{\frac{1}{2}} = \frac{\delta^{k+1}}{(\delta+h)^{k+1}}\hat{u}_{\frac{3}{2}} = \frac{\delta^{k+1}}{(\delta+h)^{k+1}}u^-_{\frac{3}{2}}. \tag{3.10}$$

Taking $v_h = u_h$ in the first item I_1, we have

$$\begin{aligned}\int_{I_1} \frac{1}{2}\frac{d}{dt}(u_h)^2\,dx &= -\frac{1}{2}(u^-_{\frac{3}{2}})^2 - \frac{1}{2}(u^+_{\frac{1}{2}})^2 + \frac{\delta^{k+1}}{(\delta+h)^{k+1}}u^-_{\frac{3}{2}}u^+_{\frac{1}{2}} \\ &\leq -\frac{1}{2}(u^-_{\frac{3}{2}})^2 + \frac{1}{2}\left(\frac{\delta^{k+1}}{(\delta+h)^{k+1}}\right)^2(u^-_{\frac{3}{2}})^2.\end{aligned}$$

Using weighted summation over all cells with a constant $W = \left[1 - \frac{1}{2^{2k+2}}\right]^{-1}$, we can get

$$\begin{aligned}\frac{d}{dt}\mathcal{E}_h(t) =& W\int_{I_1}\frac{d}{dt}(u_h)^2\,dx + \sum_{j=2}^{N}\int_{I_j}\frac{d}{dt}(u_h)^2\,dx \\ \leq& W\left[\left(\frac{\delta^{k+1}}{(\delta+h)^{k+1}}\right)^2 - 1\right](u^-_{\frac{3}{2}})^2 + (u^-_{\frac{3}{2}})^2 \\ =& \left[W\left(\frac{\delta^{k+1}}{(\delta+h)^{k+1}}\right)^2 - W + 1\right](u^-_{\frac{3}{2}})^2 \\ \leq& 0.\end{aligned}$$

Hence, the energy stability is proved. □

When employing the ILW boundary treatment for linear function with homogeneous Dirichlet boundary condition, the numerical flux $\hat{u}_{1/2} = 0$. In this case, energy stability can be easily proven in the same way,

Proposition 3.2. *For the linear equation* (3.1) *with the homogeneous inflow boundary condition, if we use the upwind numerical flux internally and ILW flux at the inflow boundary, the corresponding semi-discrete DG scheme* (2.3) *with* $k \geq 0$ *is stable in the* L^2 *norm*

$$\frac{d}{dt}\|u_h(\cdot,t)\|^2_{2,\tilde{\Omega}} \leq 0. \tag{3.11}$$

3.3 Error estimate

In this section, we will establish the error estimates of the semi-discrete scheme for the linear advection equation with the conservative SILW-1 flux.

Suppose that $u(x)$ is sufficiently differentiable, we have Taylor formula with the Peano form of the remainder

$$u(x_{\frac{1}{2}},t) = \Pi^k_0[u](\delta) + \frac{\partial^{(k+1)}_x u(0,t)}{(k+1)!}\delta^{k+1} + O(h^{k+2}),$$

$$u(x_{\frac{3}{2}},t)=\Pi_0^k[u](\delta+h)+\frac{\partial_x^{(k+1)}u(0,t)}{(k+1)!}(\delta+h)^{k+1}+O(h^{k+2}).$$

Note that the explicit expression of the conservative SILW-1 flux (2.22) for the linear equation (3.1) degenerates to

$$\hat{u}^c_{\frac{1}{2}}=\Pi_0^k[u](\delta)+\frac{\delta^{k+1}}{(\delta+h)^{k+1}}\left(u^-_{\frac{3}{2}}-\Pi_0^k[u](\delta+h)\right). \tag{3.12}$$

Hence,

$$\begin{aligned}\hat{u}^c_{\frac{1}{2}}=&\left[u(x_{\frac{1}{2}},t)-\frac{\partial_x^{(k+1)}u(0,t)}{(k+1)!}\delta^{k+1}+O(h^{k+2})\right]\\&+\frac{\delta^{k+1}}{(\delta+h)^{k+1}}\left[u^-_{\frac{3}{2}}-u(x_{\frac{3}{2}},t)+\frac{\partial_x^{(k+1)}u(0,t)}{(k+1)!}(\delta+h)^{k+1}+O(h^{k+2})\right]\\=&u(x_{\frac{1}{2}},t)-\frac{\delta^{k+1}}{(\delta+h)^{k+1}}\left(u(x_{\frac{3}{2}},t)-u^-_{\frac{3}{2}}\right)+O(h^{k+2}).\end{aligned}$$

Next, we introduce the Gauss-Radau projection P_- and L^2 projection P into V_h^k. For a given function w, the projection $P_-w\in V_h$ satisfying

$$\begin{cases}\int_{I_j}(P_-w(x)-w(x))v_h(x)\,dx=0, \quad \forall\, v_h\in P^{k-1}(I_j),\\ P_-w(x^-_{j+\frac{1}{2}})=w(x^-_{j+\frac{1}{2}}).\end{cases} \tag{3.13}$$

And the projection $Pw\in V_h$ satisfies

$$\int_{I_j}(Pw(x)-w(x))v_h(x)\,dx=0, \quad \forall\, v_h\in P^k(I_j). \tag{3.14}$$

Moreover, the projection P_*, which can be P_- or P, has the properties

$$\|u-P_*u\|_{\infty,I_j}\le Ch^{k+1}|u|_{\infty,k+1,I_j}, \tag{3.15}$$

$$\|u-P_*u\|_{2,I_j}\le Ch^{k+\frac{3}{2}}|u|_{\infty,k+1,I_j}, \tag{3.16}$$

for any $u\in C^{k+1}(I_j)$. Here $C>0$ is some constant that is independent of $h=|I_j|$ and u.

Proposition 3.3. *Consider the linear equation* (3.1) *with homogeneous Dirichlet boundary condition. Suppose the exact solution u is smooth. We consider the semi-discrete DG scheme* (2.3) *with upwind numerical flux internally and conservative SILW-1 flux at the inflow boundary, then the numerical solution u_h satisfies the following error estimate*

$$||u-u_h||_{2,\tilde{\Omega}}\le Ch^{k+1}, \tag{3.17}$$

where $C>0$ depends on u and its derivatives but is independent of h and δ.

Proof. We decompose error into two parts

$$u - u_h = \eta + \xi, \quad \eta = u - P_- u, \quad \xi = P_- u - u_h.$$

Note that the scheme (2.3) in the cell I_j, $j \geq 2$, is the same as the standard DG method. Hence, we can get the following inequality from the error estimate of standard DG method [28],

$$\frac{1}{2}\sum_{j=2}^{N}\frac{d}{dt}\int_{I_j}\xi^2\,dx \leq -\sum_{j=2}^{N}\eta_t\xi\,dx - \frac{1}{2}(\xi^-_{N+\frac{1}{2}})^2 + \frac{1}{2}(\xi^-_{\frac{3}{2}})^2. \tag{3.18}$$

Next, we focus on the special treatment for inflow boundary cell I_1. For any $v_h \in V_h$,

$$\begin{aligned}
&\int_{I_1}(u_h)_t v_h\,dx - \int_{I_1}u_h(v_h)_x\,dx + (u_h)^-_{\frac{3}{2}}v_h(x^-_{\frac{3}{2}}) - \hat{u}^c_{\frac{1}{2}}v_h(x^+_{\frac{1}{2}}) = 0,\\
&\int_{I_1}u_t v_h\,dx - \int_{I_1}u(v_h)_x\,dx + u(x_{\frac{3}{2}},t)v_h(x^-_{\frac{3}{2}}) - u(x_{\frac{1}{2}},t)v_h(x^+_{\frac{1}{2}}) = 0.
\end{aligned}$$

Subtracting these two equations, we obtain the error equation

$$\begin{aligned}
&\int_{I_1}(u-u_h)_t v_h\,dx - \int_{I_1}(u-u_h)(v_h)_x\,dx\\
&\quad + (u(x_{\frac{3}{2}},t) - (u_h)^-_{\frac{3}{2}})v_h(x^-_{\frac{3}{2}}) - (u(x_{\frac{1}{2}},t) - \hat{u}^c_{\frac{1}{2}})v_h(x^+_{\frac{1}{2}}) = 0,\ \forall v_h \in V_h.
\end{aligned}$$

Specially, take $v_h = \xi \in V_h$,

$$\begin{aligned}
0 &= \int_{I_1}(\eta+\xi)_t\xi\,dx - \int_{I_1}(\eta+\xi)\xi_x\,dx + (\eta^-_{\frac{3}{2}} + \xi^-_{\frac{3}{2}})\xi^-_{\frac{3}{2}} - (u(x_{\frac{1}{2}},t) - \hat{u}^c_{\frac{1}{2}})\xi^+_{\frac{1}{2}}\\
&= \int_{I_1}\eta_t\xi\,dx + \int_{I_1}\xi_t\xi\,dx - \frac{1}{2}(\xi^-_{\frac{3}{2}})^2 + \frac{1}{2}(\xi^+_{\frac{1}{2}})^2 + (\xi^-_{\frac{3}{2}})^2 - (u(x_{\frac{1}{2}},t) - \hat{u}^c_{\frac{1}{2}})\xi^+_{\frac{1}{2}}.
\end{aligned}$$

Employing the equivalent formula of $\hat{u}^c_{\frac{1}{2}}$ (3.13), we have that

$$\begin{aligned}
u(x_{\frac{1}{2}},t) - \hat{u}^c_{\frac{1}{2}} &= \frac{\delta^{k+1}}{(\delta+h)^{k+1}}\left[u(x_{\frac{3}{2}},t) - u_h(x^-_{\frac{3}{2}})\right] + \beta(u,\delta,h)\\
&= \frac{\delta^{k+1}}{(\delta+h)^{k+1}}\xi^-_{\frac{3}{2}} + \beta(u,\delta,h),
\end{aligned}$$

where $|\beta(u,\delta,h)| \leq Ch^{k+2}$. Therefore,

$$\begin{aligned}
\int_{I_1}\xi_t\xi\,dx &= -\int_{I_1}\eta_t\xi\,dx - \frac{1}{2}(\xi^-_{\frac{3}{2}})^2 - \frac{1}{2}(\xi^+_{\frac{1}{2}})^2 + \left[\frac{\delta^{k+1}}{(\delta+h)^{k+1}}\xi^-_{\frac{3}{2}} + \beta(u,\delta,h)\right]\xi^+_{\frac{1}{2}}\\
&\leq -\int_{I_1}\eta_t\xi\,dx - \frac{1}{2}(\xi^-_{\frac{3}{2}})^2 + \frac{1}{2}\left[\frac{\delta^{k+1}}{(\delta+h)^{k+1}}\xi^-_{\frac{3}{2}} + \beta(u,\delta,h)\right]^2.
\end{aligned} \tag{3.19}$$

We now take $\alpha = \frac{\delta^{k+1}}{(\delta+h)^{k+1}} < \left(\frac{1}{2}\right)^{k+1} < 1$. Because

$$-\frac{1}{2}(\xi^-_{\frac{3}{2}})^2 + \frac{1}{2}\left[\alpha\xi^-_{\frac{3}{2}} + \beta\right]^2 \leq \frac{\alpha^2-1}{2}(\xi^-_{\frac{3}{2}})^2 + \frac{1-\alpha^2}{2}(\xi^-_{\frac{3}{2}})^2 + \frac{\alpha^2\beta^2}{2(1-\alpha^2)} + \frac{1}{2}\beta^2 \leq Ch^{2k+4},$$

we can obtain that

$$\begin{aligned}
\int_{I_1} \xi_t \xi \, dx &\leq -\int_{I_1} \eta_t \xi \, dx - \frac{1}{2}(\xi^-_{\frac{3}{2}})^2 + \frac{1}{2}\left[\alpha\xi^-_{\frac{3}{2}} + \beta(u,\delta,h)\right]^2 \\
&\leq -\int_{I_1} \eta_t \xi \, dx + Ch^{2k+4} \\
&\leq \frac{1}{2}\int_{I_1} (\eta_t)^2 \, dx + \frac{1}{2}\int_{I_1} \xi^2 \, dx + Ch^{2k+4} \\
&\leq \frac{1}{2}\int_{I_1} \xi^2 \, dx + Ch^{2k+3}.
\end{aligned}$$

The last inequality holds when u is smooth enough. If we take the initial value $u_h = Pu$ where P is standard L^2 projection, we can get the following estimate using Gronwall's inequality,

$$\int_{I_1} \xi(x,t)^2 \, dx \leq e^t \int_{I_1} \xi(x,0)^2 \, dx + e^t \int_0^\tau e^{-\tau} Ch^{2k+3} \, d\tau \leq C(1+e^t)h^{2k+3}. \tag{3.20}$$

According to the inverse inequality, there exists a constant C related to u and t such that

$$|\xi^-_{\frac{3}{2}}| \leq \|\xi\|_{\infty,I_1} \leq Ch^{-\frac{1}{2}}\|\xi\|_{2,I_1} \leq Ch^{k+1}. \tag{3.21}$$

Plugging (3.21) into (3.19), we cam obtain

$$\begin{aligned}
\int_{I_1} \xi_t \xi \, dx &\leq -\int_{I_1} \eta_t \xi \, dx - \frac{1}{2}(\xi^-_{\frac{3}{2}})^2 + \frac{1}{2}\left[\alpha(\delta,h)\xi^-_{\frac{3}{2}} + \beta(u,\delta,h)\right]^2 \\
&\leq -\int_{I_1} \eta_t \xi \, dx - \frac{1}{2}(\xi^-_{\frac{3}{2}})^2 + Ch^{2k+2}.
\end{aligned} \tag{3.22}$$

Now, we combine the estimates (3.18) and (3.22), and have

$$\begin{aligned}
\frac{1}{2}\sum_{j=1}^N \frac{d}{dt}\int_{I_j} \xi^2 \, dx &\leq -\sum_{j=1}^N \int_{I_j} \eta_t \xi \, dx - \frac{1}{2}(\xi^-_{N+\frac{1}{2}})^2 + Ch^{2k+2} \\
&\leq \frac{1}{2}\sum_{j=1}^N \int_{I_j} \xi^2 \, dx + Ch^{2k+2}.
\end{aligned} \tag{3.23}$$

Using Gronwall's inequality, we can obtain that

$$\sum_{j=1}^N \int_{I_j} \xi(x,t)^2 \, dx \leq e^t \sum_{j=1}^N \int_{I_j} \xi(x,0)^2 \, dx + \int_0^t e^{t-\tau} Ch^{2k+2} \, d\tau. \tag{3.24}$$

Note that for the initial value,

$$\sum_{j=1}^{N} \|\xi(\cdot,0)\|_{2,I_j}^2 = \sum_{j=1}^{N} \|P_-u - Pu\|_{2,I_j}^2 \le 2\sum_{j=1}^{N} \left(\|u - P_-u\|_{2,I_j}^2 + \|u - Pu\|_{2,I_j}^2 \right) \le Ch^{2k+2}. \tag{3.25}$$

we can get the L^2 estimate of ξ

$$\|\xi(x,t)\|_{2,\tilde{\Omega}}^2 = \sum_{j=1}^{N} \int_{I_j} \xi(x,t)^2 \, dx \le C(1+e^t)h^{2k+2}. \tag{3.26}$$

which gives us the final conclusion

$$\|u(x,t) - u_h(x,t)\|_{2,\tilde{\Omega}}^2 \le 2\left(\|\xi(x,t)\|_2^2 + \|\eta(x,t)\|_2^2 \right) \le C(1+e^t)h^{2k+2}. \tag{3.27}$$

□

4 Numerical examples

In this section, we present some numerical examples to demonstrate the performance of our proposed scheme. For the scalar case, both linear and nonlinear equations are considered. We also employ this method to solve linear systems.

In all our computations, we test DG method with V_h^k $(k = 1, 2, 3)$ coupling with the conservative SILW-k boundary treatment. Without special declaration, the domain is divided uniformly with a cut element sized δ at the left boundary. And we choose two extreme values $\delta/h = 0.01, 0.99$, to show their influence on the numerical performance. The third order Runge-Kutta method (2.5) is used for the time discretization, with time step $\Delta t = \lambda h$ for $k \le 2$ and $\Delta t = \lambda h^{\frac{k+1}{3}}$ for $k = 3$. The CFL number $\lambda = \frac{1}{(2k+1)\alpha}$ is the same as that used in standard RKDG method.

4.1 Accuracy test for the linear scalar equation

In this example, we consider the initial-boundary value problem of linear advection equation

$$\begin{cases} u_t + u_x = 0, & x \in (0, 2\pi), \quad t > 0 \\ u(x,0) = -\sin(x), & x \in [0, 2\pi], \\ u(0,t) = \sin(t), & t > 0. \end{cases} \tag{4.1}$$

The exact solution is $u(x,t) = \sin(t-x)$. In this case, the left boundary $x = 0$ is an inflow boundary all the time and the right boundary $x = 2\pi$ is an outflow boundary. Errors are calculated on the computational domain $\tilde{\Omega}$. The L^2 errors and L^∞ errors at time $t = 3$ are shown in Table 2. We can see that for all cases, the schemes are stable and can achieve the optimal $(k+1)$-th order. Moreover, with the help of conservative modification, errors with different δ have no significant difference.

Table 2: Example 4.1: numerical errors and orders of linear equation at $t = 3$.

P^1 SILW-1								
	$\delta = 0.01h$				$\delta = 0.99h$			
N	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
20	1.08e-02	-	1.41e-02	-	1.10e-02	-	1.29e-02	-
40	2.67e-03	2.01	3.61e-03	1.96	2.64e-03	2.06	3.46e-03	1.90
80	6.66e-04	2.00	9.14e-04	1.98	6.57e-04	2.01	8.93e-04	1.95
160	1.66e-04	2.00	2.30e-04	1.99	1.65e-04	1.99	2.27e-04	1.98
P^2 SILW-1								
	$\delta = 0.01h$				$\delta = 0.99h$			
N	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
20	2.71e-04	-	4.00e-04	-	4.96e-04	-	5.53e-04	-
40	3.39e-05	3.00	5.04e-05	2.99	4.28e-05	3.54	5.48e-05	3.33
80	4.24e-06	3.00	6.36e-06	2.98	4.43e-06	3.27	6.31e-06	3.12
160	5.31e-07	3.00	7.94e-07	3.00	5.26e-07	3.07	7.80e-07	3.02
P^2 SILW-2								
	$\delta = 0.01h$				$\delta = 0.99h$			
N	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
20	2.71e-04	-	4.00e-04	-	6.15e-04	-	5.62e-04	-
40	3.39e-05	3.00	5.04e-05	2.99	5.15e-05	3.58	5.59e-05	3.33
80	4.24e-06	3.00	6.36e-06	2.98	4.95e-06	3.38	6.41e-06	3.12
160	5.31e-07	3.00	7.94e-07	3.00	5.54e-07	3.16	7.90e-07	3.02
P^3 SILW-1								
	$\delta = 0.01h$				$\delta = 0.99h$			
N	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
20	7.16e-06	-	1.08e-05	-	2.49e-05	-	2.06e-05	-
40	4.49e-07	4.00	6.76e-07	4.00	9.74e-07	4.68	8.52e-07	4.60
80	2.81e-08	4.00	4.24e-08	4.00	4.08e-08	4.58	4.17e-08	4.35
160	1.76e-09	4.00	2.65e-09	4.00	2.02e-09	4.33	2.59e-09	4.01
P^3 SILW-2								
	$\delta = 0.01h$				$\delta = 0.99h$			
N	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
20	7.16e-06	-	1.08e-05	-	2.62e-05	-	2.38e-05	-
40	4.49e-07	4.00	6.76e-07	4.00	9.88e-07	4.73	9.44e-07	4.66
80	2.81e-08	4.00	4.24e-08	4.00	3.93e-08	4.65	4.11e-08	4.52
160	1.76e-09	4.00	2.65e-09	4.00	1.90e-09	4.37	2.59e-09	3.99

4.2 Linear scalar equation with non-smooth boundary

Next, we consider the linear problem with non-smooth inflow boundary data

$$\begin{cases} u_t + u_x = 0, & x \in (0,1), \quad t > 0 \\ u(x,0) = 0, & x \in [0,1], \\ u(0,t) = g(t), & t > 0. \end{cases} \tag{4.2}$$

where

$$g(t) = \begin{cases} 1, & t \le 0.5, \\ 0, & t > 0.5. \end{cases}$$

We can see that discontinuities enter the computational domain from the inflow boundary. To control oscillations, the TVD limiter is applied on the interior cells and the modification mentioned in section 2.3 is used on $\hat{f}_{\frac{1}{2}}$. Numerical results with SILW-1 and SILW-2 are shown in Fig. 6 and Fig. 7 respectively, demonstrating that our algorithm does not introduce additional numerical oscillations when handling strong discontinuities near cut cells.

4.3 Nonlinear Burgers' equation

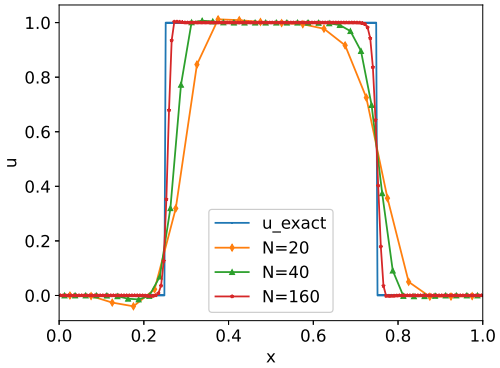
In this example, we apply the scheme to the Burgers' equation

$$\begin{cases} u_t + \left(\frac{u^2}{2}\right)_x = 0, & x \in (-\pi,\pi), \quad t > 0 \\ u(x,0) = 1 + 2\sin(x), & x \in [-\pi,\pi], \\ u(-\pi,t) = g(t), & t > 0. \end{cases} \tag{4.3}$$

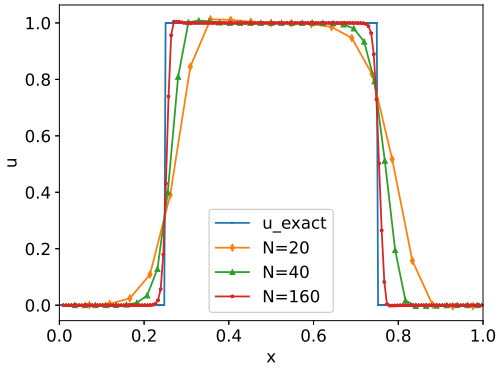
We take the boundary condition $g(t) = w(-\pi,t)$, where $w(x,t)$ is the exact solution of the initial value problem on $(-\pi,\pi)$ with periodic boundary conditions and can be obtained by Newton's method. For all time, the left boundary $x = -\pi$ is an inflow boundary and the right boundary $x = \pi$ is an outflow boundary. To verify the performance of nonlinear limiter on boundary treatment, we always use the modified numerical flux (2.24) in this example.

When $t = 0.3$, the exact solution is still smooth. Errors are shown in Table 3. It is observed that the scheme can always achieve the the optimal $(k+1)$-th order accuracy and the ratio δ/h does not effect the magnitude of errors significantly, demonstrating the limiters do not destroy accuracy of smooth solutions.

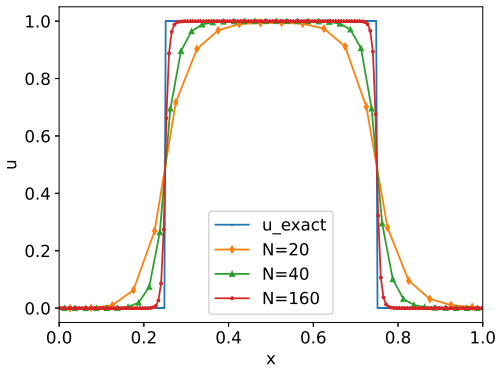
At $t = 0.5$, a shock is fully developed in the interior of the computational domain. Moreover, a shock enters the inflow boundary at t = 2π and moves to $x = 0$ at $t = 3\pi$. In order to capture the shock and avoid numerical oscillations, TVD limiter is applied on the interior cells. Fig. 8 and Fig. 9 indicate that the shock is well captured by our method.



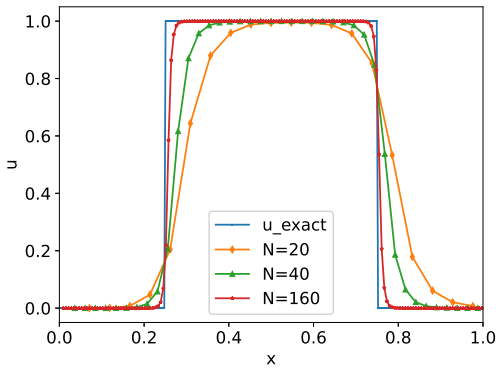
(a) $\delta = 0.01h$, $k = 1$, SILW-1



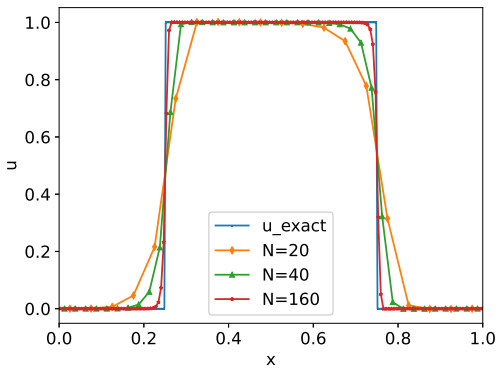
(b) $\delta = 0.99h$, $k = 1$, SILW-1



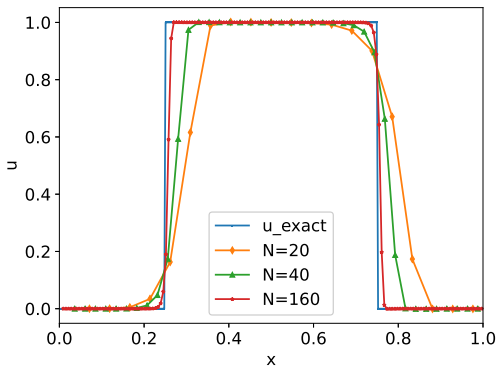
(c) $\delta = 0.01h$, $k = 2$, SILW-1



(d) $\delta = 0.99h$, $k = 2$, SILW-1

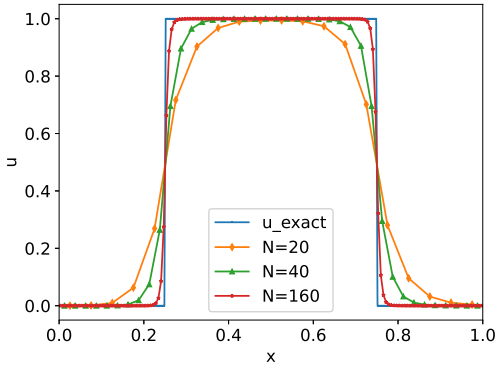


(e) $\delta = 0.01h$, $k = 3$, SILW-1

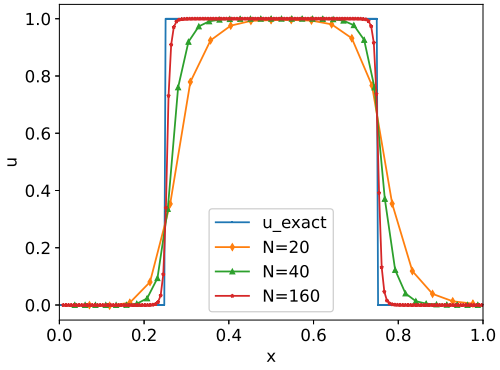


(f) $\delta = 0.99h$, $k = 3$, SILW-1

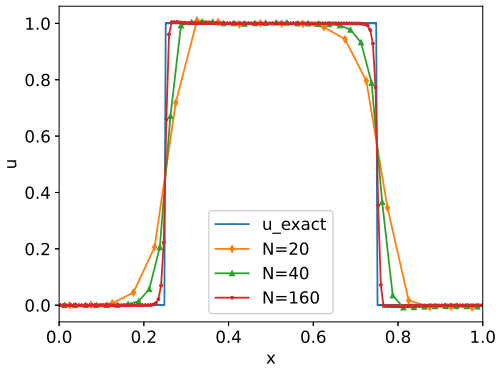
Figure 6: Example 4.2: numerical solutions for linear equation with non-smooth boundary condition at $t = 0.75$. SILW-1 boundary treatment is used. Left: $\delta = 0.01h$; right: $\delta = 0.9h$.



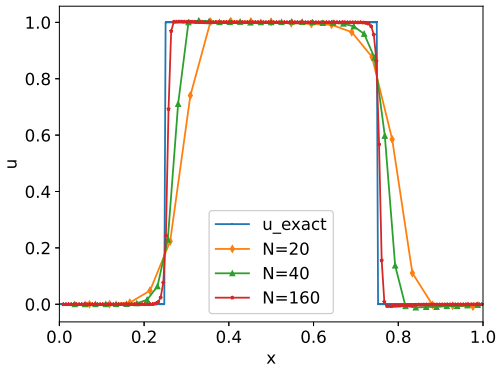
(a) $\delta = 0.01h$, $k = 2$, SILW-2



(b) $\delta = 0.99h$, $k = 2$, SILW-2



(c) $\delta = 0.01h$, $k = 3$, SILW-2

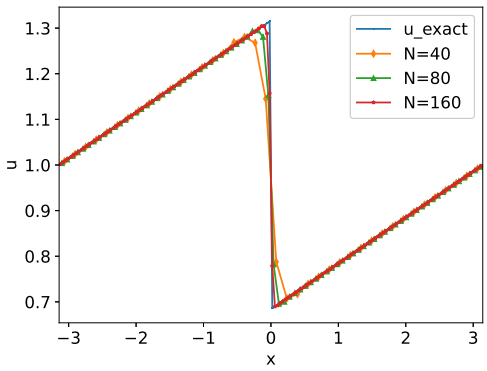


(d) $\delta = 0.99h$, $k = 3$, SILW-2

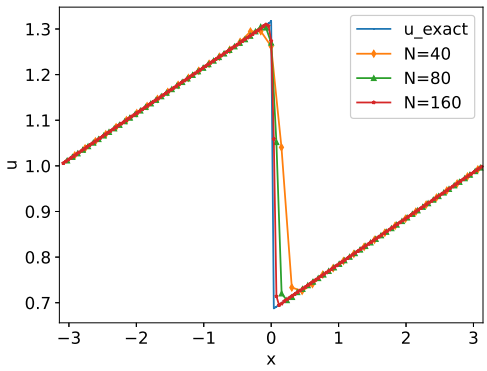
Figure 7: Example 4.2: numerical solutions for linear equation with non-smooth boundary condition at $t = 0.75$. SILW-2 boundary treatment is used Left: $\delta = 0.01h$; right: $\delta = 0.9h$.

Table 3: Example 4.3: numerical errors and orders of Burgers' equation at $t = 0.3$.

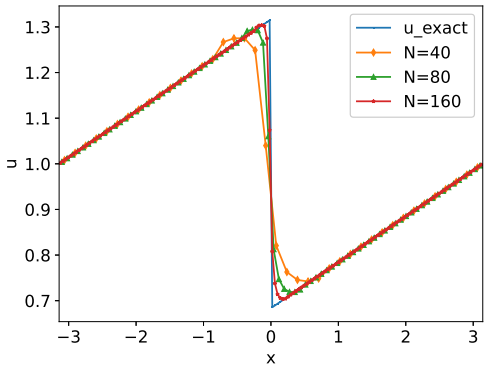
$k = 1$, SILW-1								
	$\delta = 0.01h$				$\delta = 0.99h$			
N	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
40	9.89e-03	-	2.54e-02	-	1.55e-02	-	6.04e-02	-
80	2.61e-03	1.92	7.27e-03	1.81	2.97e-03	2.39	1.12e-02	2.43
160	6.82e-04	1.93	1.94e-03	1.91	6.90e-04	2.10	2.43e-03	2.21
320	1.77e-04	1.95	5.03e-04	1.95	1.75e-04	1.98	5.63e-04	2.11
640	4.53e-05	1.96	1.28e-04	1.97	4.50e-05	1.96	1.36e-04	2.05
$k = 2$, SILW-1								
	$\delta = 0.01h$				$\delta = 0.99h$			
N	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
40	5.66e-04	-	2.59e-03	-	1.69e-03	-	1.06e-02	-
80	7.77e-05	2.87	4.13e-04	2.65	1.39e-04	3.61	5.99e-04	4.15
160	1.01e-05	2.95	5.63e-05	2.87	1.25e-05	3.47	4.78e-05	3.65
320	1.28e-06	2.98	7.34e-06	2.94	1.35e-06	3.21	7.00e-06	2.77
640	1.60e-07	2.99	9.41e-07	2.96	1.62e-07	3.06	9.23e-07	2.92
$k = 2$, SILW-2								
	$\delta = 0.01h$				$\delta = 0.99h$			
N	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
40	5.66e-04	-	2.59e-03	-	1.35e-03	-	4.63e-03	-
80	7.77e-05	2.87	4.13e-04	2.65	1.21e-04	3.48	6.13e-04	2.92
160	1.01e-05	2.95	5.63e-05	2.87	1.29e-05	3.23	6.43e-05	3.25
320	1.28e-06	2.98	7.34e-06	2.94	1.39e-06	3.22	7.77e-06	3.05
640	1.60e-07	2.99	9.41e-07	2.96	1.64e-07	3.08	9.62e-07	3.01
$k = 3$, SILW-1								
	$\delta = 0.01h$				$\delta = 0.99h$			
N	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
40	3.85e-05	-	1.11e-04	-	4.54e-04	-	1.65e-03	-
80	2.48e-06	3.95	1.12e-05	3.30	1.12e-05	5.34	4.46e-05	5.21
160	1.65e-07	3.91	7.85e-07	3.84	3.58e-07	4.96	1.75e-06	4.67
320	1.08e-08	3.93	5.23e-08	3.91	1.45e-08	4.63	7.74e-08	4.50
640	7.01e-10	3.95	3.37e-09	3.95	7.53e-10	4.27	3.94e-09	4.30
$k = 3$, SILW-2								
	$\delta = 0.01h$				$\delta = 0.99h$			
N	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
40	3.85e-05	-	1.11e-04	-	5.64e-04	-	2.20e-03	-
80	2.48e-06	3.95	1.12e-05	3.30	1.36e-05	5.38	4.91e-05	5.48
160	1.65e-07	3.91	7.85e-07	3.84	3.93e-07	5.11	1.42e-06	5.11
320	1.08e-08	3.93	5.23e-08	3.91	1.54e-08	4.67	5.44e-08	4.71
640	7.01e-10	3.95	3.37e-09	3.95	8.32e-10	4.21	3.14e-09	4.11



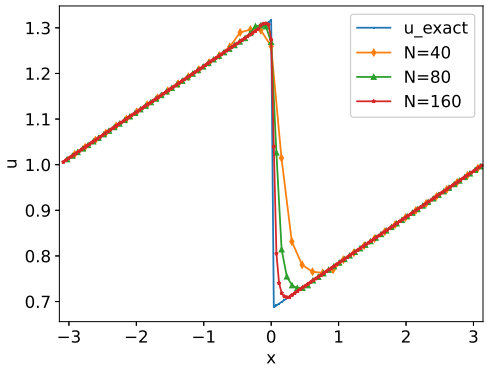
(a) $\delta = 0.01h$, $k = 1$, SILW-1



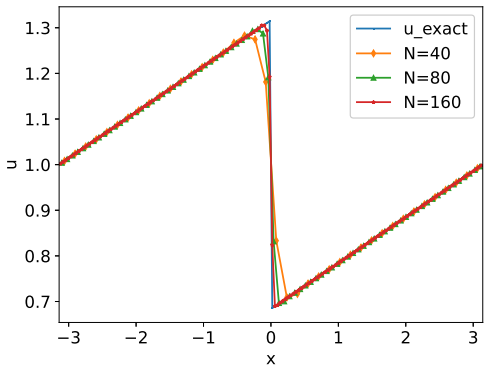
(b) $\delta = 0.99h$, $k = 1$, SILW-1



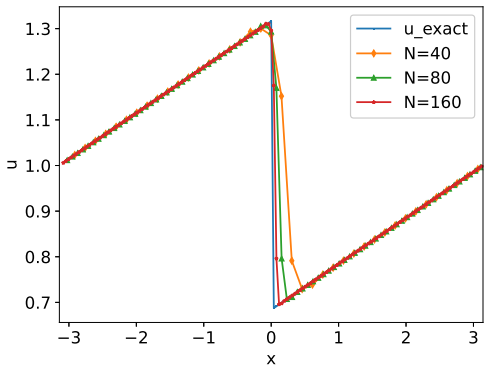
(c) $\delta = 0.01h$, $k = 2$, SILW-1



(d) $\delta = 0.99h$, $k = 2$, SILW-1

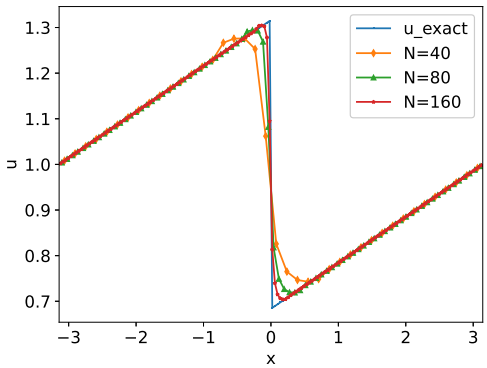


(e) $\delta = 0.01h$, $k = 3$, SILW-1

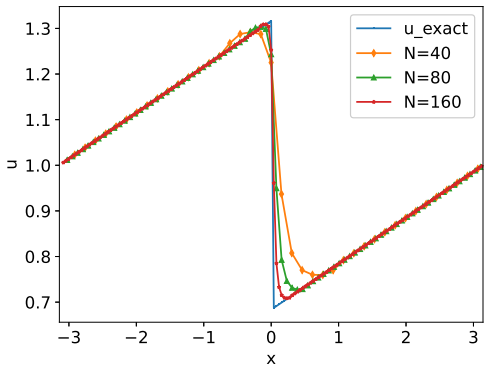


(f) $\delta = 0.99h$, $k = 3$, SILW-1

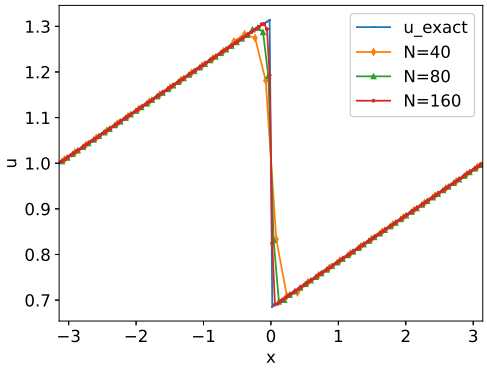
Figure 8: Example 4.2: numerical solutions for burgers' equation at $t = 3\pi$, with the SILW-1 boundary treatment. Left: $\delta = 0.01h$; right: $\delta = 0.9h$.



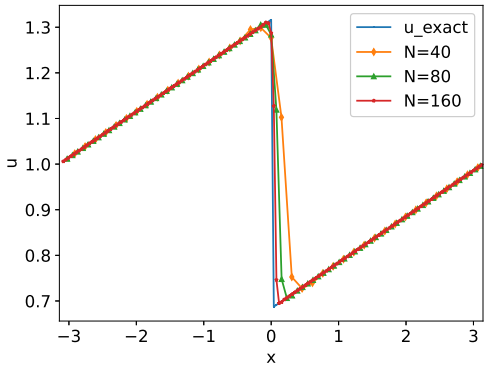
(a) $\delta = 0.01h$, P^2 SILW-2



(b) $\delta = 0.99h$, P^2 SILW-2



(c) $\delta = 0.01h$, P^3 SILW-2



(d) $\delta = 0.99h$, P^3 SILW-2

Figure 9: Example 4.2: numerical solutions for burgers' equation at $t = 3\pi$, with the SILW-2 boundary treatment. Left: $\delta = 0.01h$; right: $\delta = 0.9h$.

4.4 Linear system

Finally, we consider the linear system,

$$\begin{pmatrix} u \\ v \end{pmatrix}_t + \begin{pmatrix} & -1 \\ -c^2 & \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}_x = 0, \quad x \in (0, 2\pi),\, t > 0. \tag{4.4}$$

We fix the coefficient $c = 1.5$ and choose the corresponding initial value to match the exact solution

$$\begin{cases} u(x,t) = \cos(x + ct), \\ v(x,t) = c\cos(x + ct). \end{cases}$$

Because the eigenvalues of coefficient matrix are $\pm c$, one boundary condition is needed at each boundary. Here, we take the boundary condition

$$u(0,t) = \cos(ct), \quad u(2\pi,t) = \cos(ct).$$

The computational errors of all components, as shown in Table 4, indicate that our method still maintains the optimal convergence rate. However, in comparison to the scalar case, the impact of the ratio δ/h is shown, leading to different magnitudes of errors. This is due to the outflow boundary treatment for the system. Even though the conservative modification has reduced the influence of the ratio δ/h, errors of the outflow variable are transmitted into the inflow variable through the boundary treatment, and further affect the internal solutions.

5 Conclusion

In this paper, we design high order DG methods to solve hyperbolic conservation laws on unfitted meshes. The standard RKDG method is employed on the interior cells. On the small cut cells near physical boundaries, we apply the idea of the inverse Lax-Wendroff method to construct approximation polynomials. Moreover, a post-processing is given in order to preserve the local conservation properties of the DG solution. The eigenvalue spectrum visualization method is used to analyze the stability of the boundary treatment for both the semi-discrete and fully discrete cases. Energy stability and error analysis are given for a specific boundary treatment. Numerical results demonstrate that our methods can achieve the optimal convergence rate and completely avoid the typical issue of small time steps. Additionally, an extra nonlinear limiter is provided to prevent oscillations if a shock is close to the boundary. Numerical examples illustrate that our method has the capacity to treat shocks going through the boundaries.

We note that the treatment for outflow boundary can give high order accuracy. However, the ratio δ/h has a significant impact on the magnitude of errors, and the same is true for systems. We would like to improve this algorithm in the future. Moreover, boundary treatments for nonlinear systems and high dimensional problems are subject to future research.

Table 4: Example 4.4: numerical errors and orders of linear system at $t = 4$.

P^1 SILW-1								
	$\delta = 0.01h$				$\delta = 0.99h$			
N	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
40	4.95e-03	-	5.28e-03	-	1.12e-02	-	1.07e-02	-
80	1.21e-03	2.03	1.35e-03	1.97	1.79e-03	2.65	1.95e-03	2.45
160	3.00e-04	2.01	3.42e-04	1.98	3.46e-04	2.37	4.65e-04	2.07
320	7.50e-05	2.00	8.61e-05	1.99	7.82e-05	2.14	1.10e-04	2.08
P^2 SILW-1								
	$\delta = 0.01h$				$\delta = 0.99h$			
N	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
40	6.50e-05	-	7.44e-05	-	8.00e-04	-	4.15e-04	-
80	8.12e-06	3.00	9.33e-06	2.99	5.44e-05	3.88	2.92e-05	3.83
160	1.02e-06	3.00	1.17e-06	3.00	3.72e-06	3.87	2.13e-06	3.78
320	1.27e-07	3.00	1.46e-07	3.00	2.72e-07	3.77	1.70e-07	3.65
P^2 SILW-2								
	$\delta = 0.01h$				$\delta = 0.99h$			
N	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
40	6.50e-05	-	7.44e-05	-	5.30e-04	-	2.84e-04	-
80	8.12e-06	3.00	9.33e-06	2.99	3.69e-05	3.84	2.07e-05	3.78
160	1.02e-06	3.00	1.17e-06	3.00	2.64e-06	3.81	1.60e-06	3.69
320	1.27e-07	3.00	1.46e-07	3.00	2.10e-07	3.65	1.38e-07	3.54
P^3 SILW-1								
	$\delta = 0.01h$				$\delta = 0.99h$			
N	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
40	1.48e-06	-	1.90e-06	-	3.93e-04	-	9.72e-04	-
80	9.31e-08	3.99	1.20e-07	3.98	1.85e-05	4.41	5.92e-05	4.04
160	5.82e-09	4.00	7.55e-09	4.00	8.18e-07	4.50	3.48e-06	4.09
320	3.65e-10	4.00	4.73e-10	4.00	3.54e-08	4.53	2.04e-07	4.09
P^3 SILW-2								
	$\delta = 0.01h$				$\delta = 0.99h$			
N	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
40	1.48e-06	-	1.90e-06	-	5.46e-05	-	9.45e-05	-
80	9.31e-08	3.99	1.20e-07	3.98	2.20e-06	4.64	5.47e-06	4.11
160	5.82e-09	4.00	7.55e-09	4.00	8.98e-08	4.61	3.30e-07	4.05
320	3.65e-10	4.00	4.73e-10	4.00	3.74e-09	4.58	1.98e-08	4.06

Conflict of interest statement

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

References

[1] Peter Bastian, Christian Engwer, Jorrit Fahlke, and Olaf Ippisch. An Unfitted Discontinuous Galerkin method for pore-scale simulations of solute transport. *Mathematics and Computers in Simulation*, 81(10):2051–2061, 2011.

[2] Erik Burman and Peter Hansbo. Fictitious domain finite element methods using cut elements: II. A stabilized Nitsche method. *Applied Numerical Mathematics*, 62(4):328–341, 2012.

[3] Mark H Carpenter, David Gottlieb, Saul Abarbanel, and Wai-Sun Don. The theoretical accuracy of Runge–Kutta time discretizations for the initial boundary value problem: a study of the boundary error. *SIAM Journal on Scientific Computing*, 16(6):1241–1252, 1995.

[4] Ziqiang Cheng, Shihao Liu, Yan Jiang, Jianfang Lu, Mengping Zhang, and Shuhai Zhang. A high order boundary scheme to simulate complex moving rigid body under impingement of shock wave. *Applied Mathematics and Mechanics*, 42(6):841–854, 2021.

[5] Bernardo Cockburn, Suchung Hou, and Chi-Wang Shu. The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. IV. The multidimensional case. *Mathematics of Computation*, 54(190):545–581, 1990.

[6] Bernardo Cockburn, San-Yih Lin, and Chi-Wang Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one-dimensional systems. *Journal of Computational Physics*, 84(1):90–113, 1989.

[7] Bernardo Cockburn and Chi-Wang Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. II. General framework. *Mathematics of computation*, 52(186):411–435, 1989.

[8] Bernardo Cockburn and Chi-Wang Shu. The Runge-Kutta local projection-discontinuous-Galerkin finite element method for scalar conservation laws. *ESAIM: Mathematical Modelling and Numerical Analysis*, 25(3):337–361, 1991.

[9] Bernardo Cockburn and Chi-Wang Shu. The Runge–Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems. *Journal of Computational Physics*, 141(2):199–224, 1998.

[10] Shengrong Ding, Chi-Wang Shu, and Mengping Zhang. On the conservation of finite difference WENO schemes in non-rectangular domains using the inverse Lax-Wendroff boundary treatments. *Journal of Computational Physics*, 415:109516, 2020.

[11] Pei Fu, Thomas Frachon, Gunilla Kreiss, and Sara Zahedi. High order discontinuous cut finite element methods for linear hyperbolic conservation laws with an interface. *Journal of Scientific Computing*, 90(3):84, 2022.

[12] Pei Fu and Gunilla Kreiss. High order cut discontinuous Galerkin methods for hyperbolic conservation laws in one space dimension. *SIAM Journal on Scientific Computing*, 43(4):A2404–A2424, 2021.

[13] Andrew Giuliani. A two-dimensional stabilized discontinuous Galerkin method on curvilinear embedded boundary grids. *SIAM Journal on Scientific Computing*, 44(1):A389–A415, 2022.

[14] Ceren Gurkan, Simon Sticko, and André Massing. Stabilized cut discontinuous Galerkin methods for advection-reaction problems. *SIAM Journal on Scientific Computing*, 42(5):A2620–A2654, 2020.

[15] Peter Hansbo, Mats G Larson, and Sara Zahedi. A cut finite element method for a Stokes interface problem. *Applied Numerical Mathematics*, 85:90–114, 2014.

[16] Tingting Li, Jianfang Lu, and Chi-Wang Shu. Stability analysis of inverse Lax–Wendroff boundary treatment of high order compact difference schemes for parabolic equations. *Journal of Computational and Applied Mathematics*, 400:113711, 2022.

[17] Tingting Li, Jianfang Lu, and Pengde Wang. Stability Analysis of Inverse Lax-Wendroff Procedure for a High order Compact Finite Difference Schemes. *Communications on Applied Mathematics and Computation*, pages 1–48, 2023.

[18] Tingting Li, Chi-Wang Shu, and Mengping Zhang. Stability analysis of the inverse Lax–Wendroff boundary treatment for high order upwind-biased finite difference schemes. *Journal of Computational and Applied Mathematics*, 299:140–158, 2016.

[19] Tingting Li, Chi-Wang Shu, and Mengping Zhang. Stability analysis of the inverse Lax–Wendroff boundary treatment for high order central difference schemes for diffusion equations. *Journal of Scientific Computing*, 70:576–607, 2017.

[20] Shihao Liu, Ziqiang Cheng, Yan Jiang, Jianfang Lu, Mengping Zhang, and Shuhai Zhang. Numerical simulation of a complex moving rigid body under the impingement of a shock wave in 3d. *Advances in Aerodynamics*, 4(1):1–29, 2022.

[21] Shihao Liu, Yan Jiang, Chi-Wang Shu, Mengping Zhang, and Shuhai Zhang. A high order moving boundary treatment for convection-diffusion equations. *Journal of Computational Physics*, 473:111752, 2023.

[22] Jianfang Lu, Jinwei Fang, Sirui Tan, Chi-Wang Shu, and Mengping Zhang. Inverse Lax–Wendroff procedure for numerical boundary conditions of convection–diffusion equations. *Journal of Computational Physics*, 317:276–300, 2016.

[23] Jianfang Lu, Chi-Wang Shu, Sirui Tan, and Mengping Zhang. An inverse Lax-Wendroff procedure for hyperbolic conservation laws with changing wind direction on the boundary. *Journal of Computational Physics*, 426:109940, 2021.

[24] Björn Müller, Stephan Krämer-Eis, Florian Kummer, and Martin Oberlack. A high-order discontinuous Galerkin method for compressible flows with immersed boundaries. *International Journal for Numerical Methods in Engineering*, 110(1):3–30, 2017.

[25] Ruibin Qin and Lilia Krivodonova. A discontinuous Galerkin method for solutions of the Euler equations on Cartesian grids with embedded geometries. *Journal of Computational Science*, 4(1-2):24–35, 2013.

[26] William H Reed and Thomas R Hill. Triangular mesh methods for the neutron transport equation. Technical report, Los Alamos Scientific Lab., N. Mex.(USA), 1973.

[27] Svenja Schoeder, Simon Sticko, Gunilla Kreiss, and Martin Kronbichler. High-order cut discontinuous Galerkin methods with local time stepping for acoustics. *International Journal for Numerical Methods in Engineering*, 121(13):2979–3003, 2020.

[28] Chi-Wang Shu. Discontinuous Galerkin methods: general approach and stability. In *Numerical Solutions of Partial Differential Equations, S. Bertoluzza, S. Falletta, G. Russo and C.-W. Shu, Advanced Courses in Mathematics CRM Barcelona*, pages 149–201. Birkhäuser, Basel, 2009.

[29] Ting Song, Alex Main, Guglielmo Scovazzi, and Mario Ricchiuto. The shifted boundary method for hyperbolic systems: Embedded domain computations of linear waves and shallow water flows. *Journal of Computational Physics*, 369:45–79, 2018.

[30] Simon Sticko and Gunilla Kreiss. A stabilized Nitsche cut element method for the wave equation. *Computer methods in applied mechanics and engineering*, 309:364–387, 2016.

[31] Simon Sticko and Gunilla Kreiss. Higher order cut finite elements for the wave equation. *Journal of Scientific Computing*, 80:1867–1887, 2019.

[32] Sirui Tan and Chi-Wang Shu. Inverse Lax-Wendroff procedure for numerical boundary conditions of conservation laws. *Journal of Computational Physics*, 229(21):8144–8166, 2010.

[33] Sirui Tan and Chi-Wang Shu. A high order moving boundary treatment for compressible inviscid flows. *Journal of Computational Physics*, 230(15):6023–6036, 2011.

[34] Sirui Tan and Chi-Wang Shu. Inverse Lax–Wendroff procedure for numerical boundary conditions of hyperbolic equations: survey and new developments. In *Advances in Applied Mathematics, Modeling and Computational Science, R. Melnik and I. Kotsireas, Editors, Fields Institute Communications 66*, pages 41–63. Springer, New York, 2013.

[35] Franois Vilar and Chi-Wang Shu. Development and stability analysis of the inverse Lax-Wendroff boundary treatment for central compact schemes. *Esaim Mathematical Modelling and Numerical Analysis*, 49(1):c115, 2014.