A fourth-order conservative semi-Lagrangian finite volume WENO scheme without operator splitting for kinetic and fluid simulations

Nanyi Zheng[a], Xiaofeng Cai[b,c], Jing-Mei Qiu[d], Jianxian Qiu[e,*]

[a]*School of Mathematical Sciences, Xiamen University, Xiamen, Fujian 361005, China*
[b]*Research Center for Mathematics, Beijing Normal University at Zhuhai, Zhuhai 519087, China*
[c]*Division of Science and Technology, BNU-HKBU United international College, Zhuhai 519087, China*
[d]*Department of Mathematical Sciences, University of Delaware, Newark, DE, 19716, USA*
[e]*School of Mathematical Sciences and Fujian Provincial Key Laboratory of Mathematical Modeling and High-Performance Scientific Computing, Xiamen University, Xiamen, Fujian 361005, China*

Abstract

In this paper, we present a fourth-order conservative semi-Lagrangian (SL) finite volume (FV) weighted essentially non-oscillatory (WENO) scheme without operator splitting for two-dimensional linear transport equations with applications of kinetic models including the nonlinear Vlasov-Poisson system, the guiding center Vlasov model and the incompressible Euler equation in the vorticity-stream function formulation. To achieve fourth-order accuracy in space, two main ingredients are proposed in the SL FV formulation. Firstly, we introduce a so-called cubic-curved quadrilateral upstream cell and applying an efficient clipping method to evaluate integrals on upstream cells. Secondly, we construct a new WENO reconstruction operator, which recovers a P^3 polynomial from neighboring cell averages. Mass conservation is accomplished with the mass conservative nature of the reconstruction operator and the SL formulation. A positivity-preserving limiter is applied to maintain the positivity of the numerical solution wherever appropriate. For nonlinear

*Corresponding authors
Email addresses: nyzheng@stu.xmu.edu.cn (Nanyi Zheng), xfcai@bnu.edu.cn (Xiaofeng Cai), jingqiu@udel.edu (Jing-Mei Qiu), jxqiu@xmu.edu.cn (Jianxian Qiu)

Preprint submitted to Computer Methods in Applied Machanics and EngineeringMarch 26, 2022

kinetic models, the SL scheme is coupled with a fourth-order Runge-Kutta exponential integrator for high-order temporal accuracy. Extensive bench marks are tested to verify the designed properties.

Keywords: Vlasov systems, non-splitting scheme, semi-Lagrangian, WENO reconstruction, mass conservation, positivity preservation, high-order accuracy, cubic-curved cell.

1. Introduction

Semi-Lagrangian (SL) schemes are popular for solving transport equation which can be found in many areas of applications, such as climate modeling [1, 2] and kinetic description of plasma [3, 4, 5]. We are concerned with solving a transport equation in the form of

$$u_t + \nabla_{\mathbf{x}} \cdot (\mathbf{a}(u, \mathbf{x}, t)u) = 0, \tag{1}$$

where $u(\mathbf{x}, t)$ represents a density function and $\mathbf{a}(u, \mathbf{x}, t)$ is the velocity field. The SL approach can be both efficient and of high accuracy with its designed nature for solving (1). On one hand, as the Lagrangian approach, the information propagates along characteristics for the SL approach. Hence, it can escape the CFL time restriction and use large time steps. On the other hand, the solution space of the SL approach is built on a fixed mesh as the Eulerian approach. This feature ensures its capability to design high-order schemes.

The SL approach can be coupled with different spatial discretizations, such as finite element methods [6, 7, 8], spectral element methods [9, 10, 11], discontinous Galerkin (DG) methods [12, 13, 14], finite difference (FD) methods [15, 16, 17] and finite volume (FV) methods [18, 3, 1]. Many existing SL schemes are 1-D schemes coupled with operator-splitting method, such as the second-order Strang splitting [13] and the fourth-order Forest-Ruch splitting [5]. A main advantage of splitting-based SL schemes is its simplicity in terms of extending to multi-dimensional problems. However, splitting-based SL schemes have two unavoidable disadvantages. Firstly, the number of subproblems of splitting-based SL schemes for multi-dimensional problems proliferates, especially for the fourth-order splitting method. Secondly, without extra limitation on numerical time step and careful modifications to the splitting procedure [19], the accuracy of the splitting-based schemes decays to first order for some strongly nonlinear problems such as the guiding center

Vlasov model. For non-splitting SL schemes, a convenient idea is to construct an FD scheme, which traces the characteristics back and updates the point value information by an interpolation procedure. However, such SL FD scheme may lead to significant loss of total mass for some simulations [20]. In [21], a true non-splitting multi-dimensional mass conservative SL FD scheme is successfully built with flux-difference form for passive transport problems. However, extra time step restriction is required for numerical stability as analysed in [21]. In [13], the L^2 stability is proved for the proposed 1-D SL DG scheme. With an additional reconstruction procedure, the non-splitting SL FV scheme proposed in this paper is proved to be unconditionally stable under linearized settings by Fourier analysis. This result supports our confidence in using larger time steps.

We aim to design a high-order SL FV scheme for nonlinear dynamics. The high order accuracy comes from several careful designs. Firstly, the proposed SL FV scheme requires high-order solution remapping between a fixed Eulerian cell and a twisted upstream mesh. Such remapping procedure relies on a clipping procedure between two meshes. In [22], the authors introduced a clipping method on staggered meshes and perform numerical integration for triangles. Unfortunately, such integral strategy is difficult to extend for general curved polygons appeared in a high-order SL FV scheme. In [23], a similar clipping method is introduced on straight-sided quadrilateral meshes. The numerical integral in [23] is based on Green090005s theorem and the 2-D area integrals are convert to line integrals, which has the potential for high-order approximation for evaluating integrals on curved polygons. In our previous work in [14, 24], we followed such strategy. Quadratic-curved quadrilateral upstream cells and a clipping method are introduced in [14, 24] to achieve third-order accuracy in space. Following the same idea, we construct cubic-curved quadrilateral upstream cells, introduce a clipping method and evaluate integrals on curved polygons by the approach in [23].

Secondly, we construct a new truly 2-D cell-average-based weighted essentially non-oscillatory (WENO) reconstruction method, which follows the same idea as in [25]. The key purpose of applying WENO-type reconstruction is to provide high-order spatial approximation where the exact solution is smooth and to ensure that the numerical solution is essentially non-oscillatory where the solution is discontinuous. For hyperbolic conservation laws, the WENO schemes requires reconstructed point values at specific spatial position of each Eulerian cell. However, SL schemes may require reconstructed point values at any spatial position for general convection problem (1). This

feature precludes the use of traditional WENO reconstruction method, denoted by WENO-JS, in [26, 27, 28]. The WENO reconstruction method introduced in this paper, denoted by WENO-ZQ as in [25], uses artificial positive linear weights with their sum being one. It has the capacity to provide a P^3 reconstruction polynomial for each Eulerian cell. In the region where the solution is smooth, the WENO-ZQ reconstruction offers a fourth-order approximation to the exact solution. In the region where the solution is discontinuous, the WENO-ZQ reconstruction automatically decays to a lower order polynomial reconstruction. Hence, such reconstruction procedure shares a similar spirit with p-adaptive DG methods [29, 30, 31]. Positivity preserving (PP) is also an important consideration for numerical simulations with positive solutions. We apply a PP limiter introduced in [32] to ensure positivity for such systems. This PP limiter requires the local minimum value of each P^3 reconstruction polynomial.

Finally, for nonlinear problems with their velocity fields depends on the solutions, we couple the SL FV scheme with a fourth-order Runge-Kutta exponential integrator (RKEI) [33], denoted by CF4, for high-order temporal accuracy.

We analyze the proposed SL FV WENO scheme in terms of mass conservation, positivity preservation, error analysis, and L^2 stability. Major complexity comes from the error analysis. With regularity hypothesis on the velocity field, we quantitatively show the properties of the edges of the exact upstream cells. Such investigation helps us prove that the cubic-curved upstream cells offer fourth-order approximations to the exact upstream cells.

The rest of the paper is organized as follows. In Section 2, we introduce the implementation of the 2-D SL FV scheme. Theoretical properties and their proofs are organized in Section 3. Then, we present the numerical tests of the linear transport equation, the nonlinear Vlasov-Poisson system, the guiding center Vlasov model and the incompressible Euler equation in the vorticity-stream function formulation in Section 4. Finally, we give a conclusion in Section 5.

2. 2-D SL FV WENO scheme

Consider a 2-D linear transport equation

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}(a(x,y,t)u) + \frac{\partial}{\partial y}(b(x,y,t)u) = 0, \tag{2}$$

where $u(x,y,t)$ is a density function of a conserved quantity transported in a flow with velocity field $(a(x,y,t),b(x,y,t))$. We define the computational domain as $\Omega = [x_L,x_R]\times[y_B,y_T]$ and assume a discretization such that $x_L = x_{\frac{1}{2}} < x_{\frac{3}{2}} < \cdots < x_{i-\frac{1}{2}} < x_{i+\frac{1}{2}} < \cdots < x_{N_x-\frac{1}{2}} < x_{N_x+\frac{1}{2}} = x_R$, $y_B = y_{\frac{1}{2}} < y_{\frac{3}{2}} < \cdots < y_{j-\frac{1}{2}} < y_{j+\frac{1}{2}} < \cdots < y_{N_y-\frac{1}{2}} < y_{N_y+\frac{1}{2}} = y_T$, with $I_{i,j} := \left[x_{i-\frac{1}{2}},x_{i+\frac{1}{2}}\right]\times\left[y_{j-\frac{1}{2}},y_{j+\frac{1}{2}}\right]$, $x_i := \left(x_{i-\frac{1}{2}}+x_{i+\frac{1}{2}}\right)/2$, $y_j := \left(y_{j-\frac{1}{2}}+y_{j+\frac{1}{2}}\right)/2$, $\Delta x_i := x_{i+\frac{1}{2}}-x_{i-\frac{1}{2}}$ and $\Delta y_j := y_{j+\frac{1}{2}}-y_{j-\frac{1}{2}}$. Now, we consider an Eulerian cell $I_{i,j}$ at $t=t^{n+1}$ and define a dynamic characteristic region $I_{i,j}(t) := \{(x^\star,y^\star)\,|\,(x^\star,y^\star) = (X(x,y;t),Y(x,y;t)),\quad (x,y)\in I_{i,j}\}$, where $(X(x,y;t),Y(x,y;t))$ represents the characteristic curve emanating from (x,y,t^{n+1}), i.e., the solution of the ordinary differential equations (ODEs)

$$\begin{cases} dX(t)/dt = a(X(t),Y(t),t),\\ dY(t)/dt = b(X(t),Y(t),t),\\ X(t^{n+1}) = x,\\ Y(t^{n+1}) = y. \end{cases} \tag{3}$$

By Reynolds transport Theorem

$$\begin{aligned} \frac{d}{dt}\iint_{I_{i,j}(t)} u(x,y,t)dxdy &= \iint_{I_{i,j}(t)}\frac{\partial u}{\partial t}dxdy + \int_{\partial I_{i,j}(t)}((a,b)\cdot\vec{n})\,uds,\\ &= \iint_{I_{i,j}(t)}\frac{\partial u}{\partial t}dxdy + \iint_{I_{i,j}(t)}\nabla\cdot((a,b)u)\,dxdy = 0. \end{aligned} \tag{4}$$

Hence, an SL scheme can be naturally formulated from (4):

$$\frac{1}{\Delta x_i\Delta y_j}\iint_{I_{i,j}} u(x,y,t^{n+1})dxdy = \frac{1}{\Delta x_i\Delta y_j}\iint_{I^\star_{i,j}} u(x,y,t^n)dxdy, \tag{5}$$

where $I^\star_{i,j} = I_{i,j}(t^n)$ (see Figure 1).

The design of a fourth-order SL FV WENO scheme relies on an accurate evaluation of the right-hand side of (5). Firstly, we approximate upstream cells, $\{I^\star_{i,j}\}$, by cubic-curved quadrilaterals, denoted by $\{\widetilde{I}^\star_{i,j}\}$. We discuss the construction of $\{\widetilde{I}^\star_{i,j}\}$ in Section 2.1. Secondly, we introduce the WENO-ZQ reconstruction method to map the FV information, $\{\overline{u}^n_{i,j}\}$, to a piecewise P^3 polynomial, $\widetilde{u}(x,y)$, in Section 2.2. Finally, in Section 2.3, we describe the clipping procedures to integrate $\widetilde{u}(x,y)$ over $\{\widetilde{I}^\star_{i,j}\}$.

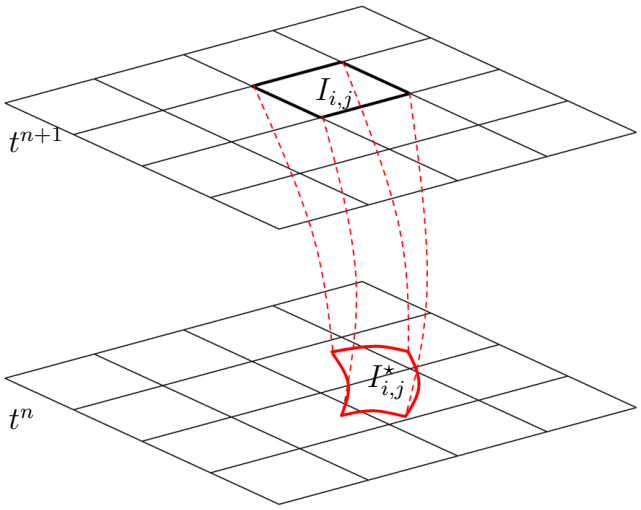


Figure 1: Schematic illustration for the characteristic upstream cell $I^{\star}_{i,j}$.

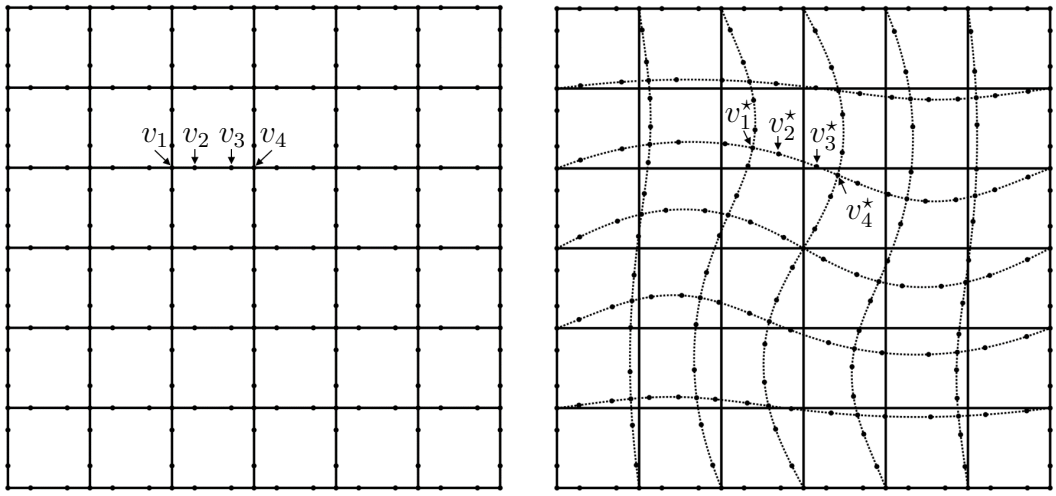
2.1. Constructing cubic-curved quadrilateral upstream cells

A cubic-curved quadrilateral upstream cell $\widetilde{I}^{\star}_{i,j}$ is a closed region enclosed by four cubic interpolated curves. We construct $\{\widetilde{I}^{\star}_{i,j}\}$ as follows.

1. Tracing characteristics back.
   We locate four Gauss-Legendre-Lobatto (GLL) points in each edge of the Eulerian cells, $\{I_{i,j}\}$, and determine their characteristic feet by solving (3) at $t = t^n$ (see Figure 2). In practice, the ODEs (3) is solved by high-order ODE solvers such as a fourth-order Runge-Kutta (RK) method.
2. Reconstructing edges of upstream cells.
   For given curved edge of a characteristic upstream cell, say $I^{\star}_{i,j}$, there are four characteristic feet, denoted as $\{v^{\star}_k\}$ (see Figure 2). By $\{v^{\star}_k\}$, we interpolate a cubic curve as the edge of $\widetilde{I}^{\star}_{i,j}$ in parametric form:

$$\begin{cases} x(\xi) = x_a\xi^3 + x_b\xi^2 + x_c\xi + x_d, \\ y(\xi) = y_a\xi^3 + y_b\xi^2 + y_c\xi + y_d, \quad \xi \in [-1, 1]. \end{cases} \tag{6}$$

   The constructing procedure of (6) is arranged in Appendix A for brevity.



(a) (b)

Figure 2: Left: the black solid lines represent the Eulerian mesh; the black dots are the GLL points located on each edge of the Eulerian cells. Right: the black solid lines represent the Eulerian mesh; the dashed lines represent the boundaries of $\{I^\star_{i,j}\}$; the black dots are the characteristic feet obtained by solving (3).

2.2. 2-D WENO reconstruction

For convenience, we require that $\Delta x_i \equiv \Delta x \quad \forall i$, and $\Delta y_j \equiv \Delta y \quad \forall j$. The WENO-ZQ reconstruction method maps the FV solution, $\{\overline{u}^n_{i,j}\}$, to a piecewise P^3 polynomial,

$$\widetilde{u}(x,y) = \widetilde{u}^{(i,j)}(x,y), \quad (x,y) \in I_{i,j}, \quad \forall (i,j), \tag{7}$$

where $\widetilde{u}^{(i,j)}(x,y) \in P^3(I_{i,j})$. We first define a set of local orthogonal basis of $P^3(I_{i,j})$ denoted as $\{v^l(x,y)\}$ with:

$$
\begin{aligned}
&v^1(x,y) = 1, \quad v^2(x,y) = \mu_i(x) := \frac{x - x_i}{\Delta x}, \quad v^3(x,y) = \nu_j(y) := \frac{y - y_j}{\Delta y}, \\
&v^4(x,y) = \mu_i^2 - \frac{1}{12}, v^5(x,y) = \mu_i \nu_j, \quad v^6(x,y) = \nu_j^2 - \frac{1}{12}, \\
&v^7(x,y) = \mu_i^3 - \frac{3}{20}\mu_i, \quad v^8(x,y) = \left(\mu_i^2 - \frac{1}{12}\right)\nu_j, \quad v^9(x,y) = \mu_i\left(\nu_j^2 - \frac{1}{12}\right), \\
&v^{10}(x,y) = \nu_j^3 - \frac{3}{20}\nu_j, \quad v^{11}(x,y) = \mu_i^4 - \frac{3}{14}\mu_i^2 + \frac{3}{560}, \\
&v^{12}(x,y) = \left(\mu_i^2 - \frac{1}{12}\right)\left(\nu_j^2 - \frac{1}{12}\right), \quad v^{13}(x,y) = \nu_j^4 - \frac{3}{14}\nu_j^2 + \frac{3}{560}.
\end{aligned}
\tag{8}
$$

We also define that $\overline{u}_5^n := \overline{u}_{i,j}^n$, $I_5 := I_{i,j}$ and other $\{\overline{u}_s^n\}$, $\{I_s\}$ represent corresponding cell averages and Eulerian cells based on the serial numbers in Figure 3. Then, the WENO-ZQ reconstruction over $I_{i,j}$ is performed as follows.

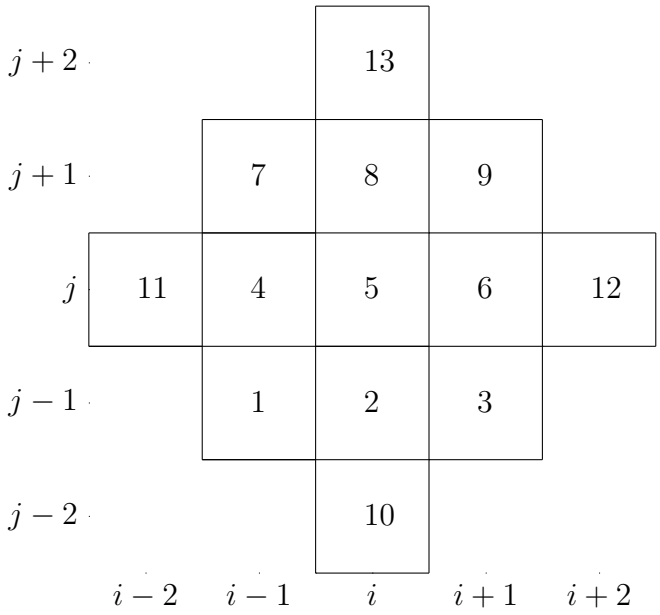


Figure 3: Stencil for the WENO-ZQ reconstruction on 2-D Cartesian mesh.

1. Construct a polynomial $\widetilde{q}_0(x,y) := \sum_{l=1}^{13} a_l^{q_0} v^l(x,y)$ and four P^1 polynomials $\{q_k(x,y)\}_{k=1}^4 := \{\sum_{l=1}^{3} a_l^{q_k} v^l(x,y)\}_{k=1}^4$ satisfying:

$$\frac{1}{\Delta x \Delta y} \iint_{I_s} \widetilde{q}_0(x,y) dxdy = \overline{u}_s^n, \tag{9}$$

where $s = 1, 2, \ldots, 13$;

$$\frac{1}{\Delta x \Delta y} \iint_{I_s} q_k(x,y) dxdy = \overline{u}_s^n, \tag{10}$$

where

$$\begin{array}{llll} s = 2,4,5, & \text{for} \quad k = 1; & s = 2,5,6, & \text{for} \quad k = 2; \\ s = 4,5,8, & \text{for} \quad k = 3; & s = 5,6,8, & \text{for} \quad k = 4. \end{array} \tag{11}$$

Let $q_0(x,y) := \sum_{l=1}^{10} a_l^{q_0} v^l(x,y)$, which is the orthogonal projection of $\widetilde{q}_0$ to $P^3(I_{i,j})$.

2. Compute the smoothness indicators $\{\beta_k\}$ of $\{q_k(x,y)\}$:

$$\begin{aligned} \beta_0 =& \frac{1}{\Delta x \Delta y} \sum_{l_1+l_2<=3} \iint_{I_{i,j}} \left(\Delta x^{l_1} \Delta y^{l_2} \frac{\partial^{|l_1+l_2|}}{\partial x^{l_1} \partial y^{l_2}} q_0(x,y) \right)^2 dxdy; \\ \beta_k =& \frac{1}{\Delta x \Delta y} \sum_{l_1+l_2<=1} \iint_{I_{i,j}} \left(\Delta x^{l_1} \Delta y^{l_2} \frac{\partial^{|l_1+l_2|}}{\partial x^{l_1} \partial y^{l_2}} q_k(x,y) \right)^2 dxdy, \quad k = 1,2,3,4. \end{aligned} \tag{12}$$

3. Calculate a new parameter:

$$\tau := \left(\frac{|\beta_0 - \beta_1| + |\beta_0 - \beta_2| + |\beta_0 - \beta_3| + |\beta_0 - \beta_4|}{4} \right)^2. \tag{13}$$

4. Compute the nonlinear weights $\{\omega_k\}$ (see [25]):

$$\omega_k = \frac{\overline{\omega}_k}{\sum_{l=0}^{4} \overline{\omega}_l}, \quad \text{with} \quad \overline{\omega}_k = \gamma_k \left(1 + \frac{\tau}{\beta_k + \epsilon} \right), \quad k = 0, 1, \ldots, 4. \tag{14}$$

where ϵ, which is 10^{-14} in our program, is used to avoid the denominator being zero and $\{\gamma_k\}$ is a set of positive linear weights satisfying that the sum of which is 1. In numerical tests, we take $\gamma_0 = 0.8$, $\gamma_1 = \ldots = \gamma_4 = 0.05$.

5. The WENO-ZQ reconstruction polynomial $\widetilde{u}^{(i,j)}(x,y) := \sum_{l=1}^{10} a_l^{(i,j)} v^l(x,y)$ is given by

$$\widetilde{u}^{(i,j)}(x,y) = \frac{\omega_0}{\gamma_0}\left(q_0(x) - \sum_{k=1}^{4}\gamma_k q_k(x,y)\right) + \sum_{k=1}^{4}\omega_k q_k(x,y), \tag{15}$$

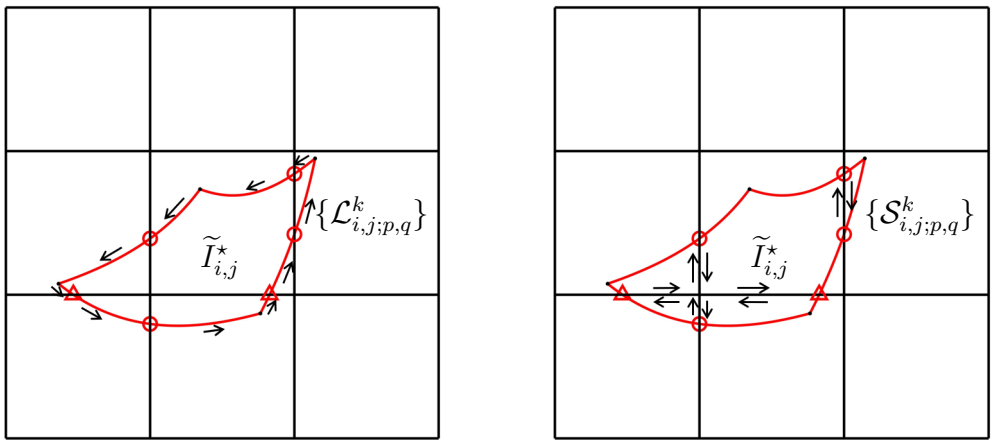
where $\{a_l^{(i,j)}\}$ can be explicitly provided by

$$\begin{aligned} &a_1^{(i,j)} = \overline{u}_{i,j}^n; \quad a_l^{(i,j)} = \frac{\omega_0}{\gamma_0} a_l^{q_0} + \sum_{k=1}^{4}\left(\omega_k - \frac{\omega_0}{\gamma_0}\gamma_k\right) a_l^{q_k} \quad \text{for} \quad l = 2,3; \\ &a_l^{(i,j)} = \frac{\omega_0}{\gamma_0} a_l^{q_0} \quad \text{for} \quad l = 4,5,\ldots,10. \end{aligned} \tag{16}$$

In particular, if $\widetilde{u}^{(i,j)}(x,y) = \sum_{l=1}^{10} a_l^{q_0} v^l(x,y) \ \ \forall i,j$, we call such a reconstruction linear reconstruction.

2.3. Calculating integrals over cubic curved quadrilateral upstream cells

2.3.1. Clipping



(a) (b)

Figure 4: Schematic illustration for the definitions of outer integral segments (a) and inner integral segments (b). The red circles and triangles represent the intersection points of $\widetilde{I}_{i,j}^{\star}$ and the Eulerian mesh.

As introduced in Section 2.2, we will evaluate the integration of a reconstructed piecewise polynomial over each $\widetilde{I}_{i,j}^{\star}$, which may cross different

Eulerian cells. Hence, we clip $\{\widetilde{I}^{\star}_{i,j}\}$ into curved polygons such that $\widetilde{I}^{\star}_{i,j} = \cup_{(p,q)}(\widetilde{I}^{\star}_{i,j} \cap I_{p,q})$ and integrate piecewisely. We denote the curved polygons $\left\{\widetilde{I}^{\star}_{i,j} \cap I_{p,q}\right\}$ by $\left\{\widetilde{I}^{\star}_{i,j;p,q}\right\}$. The clipping algorithm introduced here focuses on determining edges of the curved polygons. In particular, we define the *outer integral segments* of a upstream cell $\widetilde{I}^{\star}_{i,j}$ as the edges, overlapping $\partial\widetilde{I}^{\star}_{i,j}$, of $\left\{\widetilde{I}^{\star}_{i,j;p,q}\right\}$ with counterclockwise direction with respect to $\widetilde{I}^{\star}_{i,j}$, denoted by $\{\mathcal{L}^{k}_{i,j;p,q}\}$ (see Figure 4 (a)). The *inner integral segments* are defined as the edges, overlapping mesh lines, of $\left\{\widetilde{I}^{\star}_{i,j;p,q}\right\}$ with counterclockwise direction with respect to the curved polygons, denoted by $\{\mathcal{S}^{k}_{i,j;p,q}\}$ (see Figure 4 (b)).

For the implementation of clipping, we introduce the basic procedure in Appendix B for conciseness.

2.3.2. Numerical integral

For numerical integral, we define a reference integral region $[-\frac{1}{2},\frac{1}{2}] \times [-\frac{1}{2},\frac{1}{2}]$ and corresponding projections $\mathbf{J}^{(i,j)} : \mathbb{R}^2 \to \mathbb{R}^2,\ (x,y) \mapsto (\mu_i,\nu_j) := ((x-x_i)/\Delta x, (y-y_j)/\Delta y)$ for all i,j. With the clipped outer integral segments, $\{\mathcal{L}^{k}_{i,j;p,q}\}$, as well as the inner integral segments, $\{\mathcal{S}^{k}_{i,j;p,q}\}$, and the reconstruction piecewise P^3 polynomial, $\widetilde{u}(x,y)$, we can numerically approximate (5) by

$$
\begin{aligned}
\overline{u}^{n+1}_{i,j} =& \frac{1}{\Delta x\Delta y}\iint_{\widetilde{I}^{\star}_{i,j}} \widetilde{u}(x,y)dxdy \\
=& \frac{1}{\Delta x\Delta y}\sum_{(p,q)}\iint_{\widetilde{I}^{\star}_{i,j;p,q}} \widetilde{u}^{(p,q)}(x,y)dxdy \\
=& \sum_{(p,q)}\iint_{\mathbf{J}^{(p,q)}(\widetilde{I}^{\star}_{i,j;p,q})} \widetilde{u}^{(p,q)}(\mu_p,\nu_q)d\mu_p d\nu_q \\
=& \sum_{(p,q)}\int_{\mathbf{J}^{(p,q)}(\partial(\widetilde{I}^{\star}_{i,j;p,q}))} \left[\widetilde{P}^{(p,q)}d\mu_p + \widetilde{Q}^{(p,q)}d\nu_q\right] \\
=& \sum_{(p,q)}\Big\{\sum_{k}\int_{\mathbf{J}^{(p,q)}(\mathcal{L}^{k}_{i,j;p,q})} \left[\widetilde{P}^{(p,q)}d\mu_p + \widetilde{Q}^{(p,q)}d\nu_q\right] \\
&+ \sum_{k}\int_{\mathbf{J}^{(p,q)}(\mathcal{S}^{k}_{i,j;p,q})} \left[\widetilde{P}^{(p,q)}d\mu_p + \widetilde{Q}^{(p,q)}d\nu_q\right]\Big\},
\end{aligned}
\tag{17}
$$

where $\widetilde{P}^{(p,q)}(\mu_p,\nu_q)$ and $\widetilde{Q}^{(p,q)}(\mu_p,\nu_q)$ are piecewise smooth auxiliary functions such that

$$-\frac{\partial \widetilde{P}^{(p,q)}}{\partial \nu_q}+\frac{\partial \widetilde{Q}^{(p,q)}}{\partial \mu_p}=\widetilde{u}^{(p,q)}(\mu_p,\nu_q). \tag{18}$$

In our program, we choose:

$$\widetilde{P}^{(p,q)}(\mu_p,\nu_q)=-\left[a_2^{(p,q)}\mu_p+a_4^{(p,q)}\left(\mu_p^2-\frac{1}{12}\right)+a_7^{(p,q)}\left(\mu_p^3-\frac{3}{20}\mu_p\right)\right]\nu_q, \tag{19}$$

and

$$\begin{aligned}\widetilde{Q}^{(p,q)}(\mu_p,\nu_q)=&\left[a_1^{(p,q)}+\left(a_3^{(p,q)}-\frac{a_8^{(p,q)}}{18}\right)\nu_q+\frac{1}{2}a_5^{(p,q)}\mu_p\nu_q+a_6^{(p,q)}\left(\nu_q^2-\frac{1}{12}\right)\right.\\&\left.+\frac{1}{3}a_8^{(p,q)}\left(\mu_p^2-\frac{1}{12}\right)\nu_q+\frac{1}{2}a_9^{(p,q)}\mu_p\left(\nu_q^2-\frac{1}{12}\right)+a_{10}^{(p,q)}\left(\nu_q^3-\frac{3}{20}\nu_q\right)\right]\mu_p.\end{aligned} \tag{20}$$

For the line integral over a given outer integral segment, say $\mathcal{L}_{i,j;p,q}^k$,

$$\begin{aligned}&\int_{\mathbf{J}^{(p,q)}(\mathcal{L}_{i,j;p,q}^k)}[\widetilde{P}^{(p,q)}d\mu_p+\widetilde{Q}^{(p,q)}d\nu_q]\\=&\int_{\xi_k}^{\xi_{k+1}}\left[\widetilde{P}^{(p,q)}\left(\mu_p(\xi),\nu_q(\xi)\right)\mu_p'(\xi)+\widetilde{Q}^{(p,q)}\left(\mu_p(\xi),\nu_q(\xi)\right)\nu_q'(\xi)\right]d\xi,\end{aligned} \tag{21}$$

where ξ_k and ξ_{k+1} represent the ξ value of the start point and end point of $\mathcal{L}_{i,j;p,q}^k$ with respect to corresponding cubic curve (6), and $(\mu_p(\xi),\nu_q(\xi)):=\left(\frac{x(\xi)-x_p}{\Delta x},\frac{y(\xi)-y_q}{\Delta y}\right)$.

In practical programming, we use 3-point Gauss-Legendre quadrature for line integrals. It can be checked that the 3-point Gauss-Legendre quadrature is exact for integration on inner integral segments and is of sixth-order accuracy for outer integral segments.

2.3.3. PP limiter

When the analytical solution of (2) enjoys the PP property, we introduce the PP limiter in [32] to ensure the PP property of the numerical solution. The PP limiter applied here replaces $\widetilde{u}^{(i,j)}(x,y)$ by

$$\widetilde{\widetilde{u}}^{(i,j)}(x,y):=\theta_{i,j}\left(\widetilde{u}^{(i,j)}(x,y)-\overline{u}_{ij}^n\right)+\overline{u}_{ij}^n,\quad (x,y)\in I_{i,j} \tag{22}$$

where

$$\theta_{i,j} := \min\left\{\left|\frac{\overline{u}_{ij}^n}{\overline{u}_{ij}^n - m_{ij}}\right|\right\}, \quad m_{ij} := \min_{(x,y)\in I_{i,j}}\left\{\widetilde{u}^{(i,j)}(x,y)\right\}. \tag{23}$$

To determine m_{ij}, one has to find the extreme points of $R^{(i,j)}(x,y)$ by solving a system of quadratic equations with two variables:

$$\begin{cases} \dfrac{\partial \widetilde{u}^{(i,j)}(x,y)}{\partial x} = 0, \\ \dfrac{\partial \widetilde{u}^{(i,j)}(x,y)}{\partial y} = 0. \end{cases} \tag{24}$$

Through careful classification, solving (24) is equivalent to finding the intersection points of conic sections.

It can be proved as in [32] that $\widetilde{\widetilde{u}}^{(i,j)}(x,y)$ stays non-negative and maintains the original accuracy of $\widetilde{u}^{(i,j)}(x,y)$ approximating $u(x,y,t^n)$.

Remark 2.1. *By our numerical test, we find that we can not only compute the minimum value of $\widetilde{u}(x,y)$ at the corresponding Gauss-Legendre points as in [34]. This is probably because that the Green's formulation is involved in (17).*

3. Theoretical properties

In this section, we demonstrate four basic properties of the SL FV WENO scheme.

Proposition 3.1. *(Mass conservation). The SL FV WENO scheme is mass conservative if periodic boundary condition is imposed.*

Proof.

$$\begin{aligned} \Delta x\Delta y\sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\overline{u}_{i,j}^{n+1} &= \Delta x\Delta y\sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\int_{\mathbf{J}^{(p,q)}(\partial I_{i,j})}\left[\widetilde{P}^{(i,j)}d\mu_i + \widetilde{Q}^{(i,j)}d\nu_j\right] \\ = \sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\iint_{I_{i,j}}\widetilde{u}^{(i,j)}(x,y)dxdy &= \Delta x\Delta y\sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\overline{u}_{i,j}^n, \end{aligned} \tag{25}$$

where the first equality comes from the periodic boundary condition, the second equality is based on the fact that integrals over inner integral segments are exact and the third equality comes from the mass conservation property of the WENO reconstruction.

□

Proposition 3.2. *(Positivity preservation). The SL FV WENO scheme coupled with the PP limiter in Section 2.3.3 is PP if the line integrals in* (17) *are exactly evaluated.*

Proof. As stated in Section 2.3.3, $\widetilde{\widetilde{u}}^{(i,j)}(x,y)$ is non-negative for all i,j. The conclusion is clearly valid given the line integrals in (17) are exactly evaluated.

□

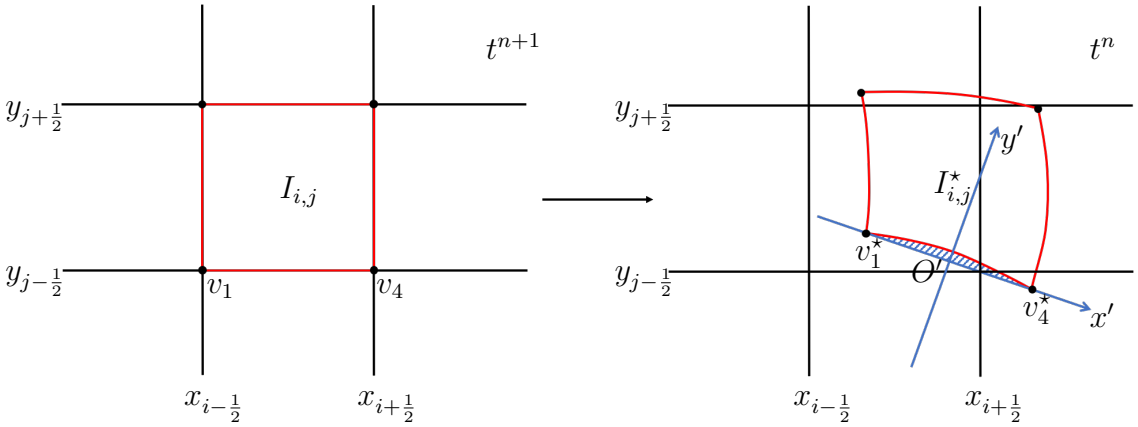


Figure 5: Left: the black solid lines represent the Eulerian mesh; the black dots are the vertexes of $I_{i,j}$; the red solid lines are the edges of $I_{i,j}$. Right: the black solid lines represent the Eulerian mesh; the black dots are the vertexes of $I^\star_{i,j}$; the red curves represent the boundaries of $\{I^\star_{i,j}\}$.

Some preparations are required before providing the error analysis. Firstly, we emphasize that the characteristics will not intersect at the PDE level for the problems we consider. Secondly, we assume that $\Delta t \sim \Delta x \sim \Delta y$, which guarantees that the exact upstream cells are only mildly distorted, as will be proved below. Now, Consider an edge of $I_{i,j}$, $\overline{v_1 v_4}$, at $t = t^{n+1}$ and its characteristic upstream curve $\widehat{v_1^\star v_4^\star}$ (see Figure 5). Here, we use the same notation of $\{v_k^\star\}_{k=1}^4$ as in Section 2.1 and assume that $\{v_k^\star\}$ are obtained

by exactly solving (3). For simplicity, we define $\mathcal{X}(\alpha) := X(\alpha, y_{j-\frac{1}{2}}; t^n)$, and $\mathcal{Y}(\alpha) := Y(\alpha, y_{j-\frac{1}{2}}; t^n)$ with $(\alpha, y_{j-\frac{1}{2}})$ $\alpha \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ representing $\overline{v_1 v_4}$. Then, a parametric equation of $\widehat{v_1^\star v_4^\star}$ is represented by

$$\begin{bmatrix} \mathcal{X}(\alpha) \\ \mathcal{Y}(\alpha) \end{bmatrix} = \begin{bmatrix} \alpha \quad + \int_{t^{n+1}}^{t^n} a(X(\alpha, y_{j-\frac{1}{2}}; t), Y(\alpha, y_{j-\frac{1}{2}}; t), t) dt \\ y_{j-\frac{1}{2}} + \int_{t^{n+1}}^{t^n} b(X(\alpha, y_{j-\frac{1}{2}}; t), Y(\alpha, y_{j-\frac{1}{2}}; t), t) dt \end{bmatrix} \quad \alpha \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]. \tag{26}$$

Lemma 3.3. *The derivatives of $\mathcal{X}(\alpha)$, $\mathcal{Y}(\alpha)$ satisfies*

$$\begin{bmatrix} \frac{d\mathcal{X}(\alpha)}{d\alpha} \\ \frac{d\mathcal{Y}(\alpha)}{d\alpha} \end{bmatrix} = \begin{bmatrix} 1 + O(\Delta t) \\ O(\Delta t) \end{bmatrix} \quad \alpha \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}], \tag{27}$$

and

$$\begin{bmatrix} \frac{d^k\mathcal{X}(\alpha)}{d\alpha^k} \\ \frac{d^k\mathcal{Y}(\alpha)}{d\alpha^k} \end{bmatrix} = \begin{bmatrix} O(\Delta t) \\ O(\Delta t) \end{bmatrix} \quad \alpha \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \quad k = 2, 3, 4. \tag{28}$$

if $a(x, y, t), b(x, y, t) \in C^4$.

Proof. Taking the derivative of both sides of (26) with respect to α and expand the right-hand side integrands at $(\alpha, y_{j-\frac{1}{2}}; t^n)$, we have

$$\begin{bmatrix} \frac{d\mathcal{X}(\alpha)}{d\alpha} \\ \frac{d\mathcal{Y}(\alpha)}{d\alpha} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} - \Delta t \begin{bmatrix} a_1' & a_2' \\ b_1' & b_2' \end{bmatrix} \Bigg|_{(\alpha, y_{j-\frac{1}{2}}; t^n)} \begin{bmatrix} \frac{d\mathcal{X}(\alpha)}{d\alpha} \\ \frac{d\mathcal{Y}(\alpha)}{d\alpha} \end{bmatrix} + \delta A, \tag{29}$$

where $\delta A \in \mathbb{R}^2$ and $\|\delta A\| = O(\Delta t^2)$. Dropping the high-order term δA, we have

$$\begin{bmatrix} 1 + \Delta t a_1' & \Delta t a_2' \\ \Delta t b_1' & 1 + \Delta t b_2' \end{bmatrix} \begin{bmatrix} \frac{d\mathcal{X}(\alpha)}{d\alpha} \\ \frac{d\mathcal{Y}(\alpha)}{d\alpha} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}. \tag{30}$$

Denote the matrix in (30) by B. We find that $\det(B) = 1 + O(\Delta t)$. It is obvious that $\det(B) \neq 0$ for sufficiently small Δt. Hence, we obtain

$$\begin{bmatrix} \frac{d\mathcal{X}(\alpha)}{d\alpha} \\ \frac{d\mathcal{Y}(\alpha)}{d\alpha} \end{bmatrix} = \begin{bmatrix} (1 + \Delta t b_2')/\det(B) \\ -\Delta t b_1'/\det(B) \end{bmatrix} = \begin{bmatrix} 1 + O(\Delta t) \\ O(\Delta t) \end{bmatrix} \quad \alpha \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]. \tag{31}$$

Similarly, by taking the second, third, and fourth derivative of both sides of (26) with respect to α, we can prove that

$$\begin{bmatrix} \frac{d^{(k)}\mathcal{X}(\alpha)}{d\alpha^k} \\ \frac{d^{(k)}\mathcal{Y}(\alpha)}{d\alpha^k} \end{bmatrix} = \begin{bmatrix} O(\Delta t) \\ O(\Delta t) \end{bmatrix} \quad \alpha \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \quad \text{k=2, 3, 4.} \tag{32}$$

□

Lemma 3.3 naturally leads to the following lemma.

Lemma 3.4. *Assuming that* $\{v_k^\star\}_{k=1}^4 := \{(x_k, y_k)\}_{k=1}^4$ *in* $x-y$ *space, we have*

$$x_4 - x_1 = \Delta x + O(h^2), \quad y_4 - y_1 = O(h^2), \quad d(v_1^\star, v_4^\star) = \Delta x + O(h^2) \tag{33}$$

if $a(x, y, t),\ b(x, y, t) \in C^4$, *where* $h \sim \Delta x \sim \Delta y \sim \Delta t$.

Proof. By Lemma 3.3, the result follows by a simple calculation. □

We construct a new coordinate space $x'-y'$ such that $x'-y'$ is transformed by $x - y$ through a rotation and translation transformation (see Figure 5). $x' - y'$ is set up so that the direction of x'-axis is the same with $\overrightarrow{v_1^\star v_4^\star}$. Let O' be the midpoint of $v_1^\star$ and $v_4^\star$. Assume that the coordinates of $\{v_k^\star\}$ in $x'-y'$ space are $\{(x'_k, y'_k)\}$, where $y'_1 = y'_4 = 0$. Then, we define a parametric equation of $\widehat{v_1^\star v_4^\star}$ as $(x', f(x'))$ with $x' \in [x'_1, x'_4]$.

Lemma 3.5. *The derivatives of* $f(x')$ *satisfies*

$$f^{(k)}(x') = O(h) \quad x' \in [x'_1, x'_4] \ \ k = 1, 2, 3, 4 \tag{34}$$

if $a(x, y, t),\ b(x, y, t) \in C^4$, *where* $h \sim \Delta x \sim \Delta y \sim \Delta t$.

Proof. With the definition of $x' - y'$, we have

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \frac{x_4-x_1}{d} & \frac{y_4-y_1}{d} \\ -\frac{y_4-y_1}{d} & \frac{x_4-x_1}{d} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} := T \begin{bmatrix} x \\ y \end{bmatrix} + \vec{c}, \tag{35}$$

where $d = d(v_1^\star, v_4^\star)$. Let $T = (t_{ij})_{2\times 2}$. By Lemma 3.4, we immediately obtain

$$t_{11} = 1 + O(h), \quad t_{12} = O(h), \quad t_{21} = O(h), \quad t_{22} = 1 + O(h). \tag{36}$$

Then, for $\widehat{v_1^\star v_4^\star}$, we have

$$\begin{bmatrix} x' \\ f(x') \end{bmatrix} = T \begin{bmatrix} \mathcal{X}(\alpha) \\ \mathcal{Y}(\alpha) \end{bmatrix} + \vec{c} \quad x' \in [x_1', x_4']. \tag{37}$$

Taking the derivative of both sides of (37) with respect to x', we obtain

$$\begin{bmatrix} 1 \\ f'(x') \end{bmatrix} = T \begin{bmatrix} \frac{d\mathcal{X}(\alpha)}{d\alpha} \cdot \frac{d\alpha}{dx'} \\ \frac{d\mathcal{Y}(\alpha)}{d\alpha} \cdot \frac{d\alpha}{dx'} \end{bmatrix} \quad x' \in [x_1', x_4']. \tag{38}$$

By Lemma 3.3, we obtain

$$\begin{bmatrix} \frac{d\alpha}{dx'} \\ f'(x') \end{bmatrix} = \begin{bmatrix} 1/\left(t_{11}\frac{d\mathcal{X}}{d\alpha} + t_{12}\frac{d\mathcal{Y}}{d\alpha}\right) \\ \left(t_{21}\frac{d\mathcal{X}}{d\alpha} + t_{22}\frac{d\mathcal{Y}}{d\alpha}\right)\frac{d\alpha}{dx'} \end{bmatrix} = \begin{bmatrix} 1 + O(h) \\ O(h) \end{bmatrix} \quad x' \in [x_1', x_4']. \tag{39}$$

Similarly, taking the second, third, and fourth derivative of both sides of (37) with respect to x', we can prove that

$$f^{(k)}(x') = O(h) \quad x' \in [x_1', x_4'] \quad k = 2, 3, 4. \tag{40}$$

□

Proposition 3.6. *(Error analysis). The numerical update given by* (17) *satisfies:*

$$\left| \frac{1}{\Delta x \Delta y} \iint_{I_{i,j}} u\left(x, y, t^{n+1}\right) dxdy - \overline{u}_{i,j}^{n+1} \right| = O\left(h^4\right) \tag{41}$$

with $h \sim \Delta x \sim \Delta y \sim \Delta t \quad (h \to 0)$ *if corresponding* $\{\overline{u}_{p,q}^n\}$ *are exact and* $u(x, y, t), a(x, y, t), b(x, y, t) \in C^4$.

Proof.

$$
\begin{aligned}
&\left|\frac{1}{\Delta x\Delta y}\iint_{I_{i,j}} u\left(x,y,t^{n+1}\right)dxdy-\overline{u}_{i,j}^{n+1}\right| \\
=&\frac{1}{\Delta x\Delta y}\left|\iint_{I_{i,j}^{\star}} u\left(x,y,t^{n}\right)dxdy-\left(\iint_{\widetilde{I}_{i,j}^{\star}}\widetilde{u}\left(x,y\right)dxdy+e_I\right)\right| \\
\leq&\frac{1}{\Delta x\Delta y}\left[\iint_{I_{i,j}^{\star}}\left|u\left(x,y,t^{n}\right)-\widetilde{u}(x,y)\right|dxdy\right. \\
&\left.+\left|\iint_{I_{i,j}^{\star}}\widetilde{u}(x,y)dxdy-\iint_{\widetilde{I}_{i,j}^{\star}}\widetilde{u}(x,y)dxdy\right|+|e_I|\right] \\
\leq&\frac{1}{\Delta x\Delta y}\left[\max_{(x,y)\in I_{i,j}^{\star}}\left\{\left|u\left(x,y,t^{n}\right)-\widetilde{u}(x,y)\right|\right\}\left|I_{i,j}^{\star}\right|+M\left(|I_{i,j}^{\star}\backslash\widetilde{I}_{i,j}^{\star}|+|\widetilde{I}_{i,j}^{\star}\backslash I_{i,j}^{\star}|\right)+|e_I|\right],
\end{aligned}
\tag{42}
$$

where $M=\max\left\{|\widetilde{u}\left(x,y\right)|\right\}$ and e_I is the error introduced by numerical integration. For $\left|u\left(x,y,t^{n}\right)-\widetilde{u}(x,y)\right|$, we refer to [25] for similar analysis and conclude that

$$
\max_{(x,y)\in I_{i,j}^{\star}}\left\{\left|u\left(x,y,t^{n}\right)-\widetilde{u}(x,y)\right|\right\}=O\left(h^4\right). \tag{43}
$$

For $|e_I|$, notice that we use the 3-point Gauss-Legendre quadrature for numerical integral, which offers a six-order approximation to exact integral. Hence, we have $|e_I|=O(h^6)$.

To prove (41), it is sufficient to prove that

$$
\left|I_{i,j}^{\star}\right|=O\left(h^2\right) \tag{44}
$$

and

$$
\left(|I_{i,j}^{\star}\backslash\widetilde{I}_{i,j}^{\star}|+|\widetilde{I}_{i,j}^{\star}\backslash I_{i,j}^{\star}|\right)=O\left(h^6\right). \tag{45}
$$

Define that $I_{i,j}^{\star\star}$ represents the straight-sided quadrilateral determined by the four vertexes of $I_{i,j}^{\star}$. Then, it is obvious that $|I_{i,j}^{\star\star}|=O(h^2)$ by Lemma 3.4. Notice that $|I_{i,j}^{\star}|=|I_{i,j}^{\star\star}|+|I_{i,j}^{\star}\backslash I_{i,j}^{\star\star}|-|I_{i,j}^{\star\star}\backslash I_{i,j}^{\star}|$. We denote the blue shaded area in Figure 5 by Ω_b. To prove $|I_{i,j}^{\star}|=O(h^2)$, it is sufficient to investigate $|\Omega_b|$. By Lemma 3.4 and Lemma 3.5, we have

$$
\begin{aligned}
|\Omega_b| &= \left| \int_{x_1'}^{x_4'} f(x')dx' \right| \\
&= \left| \int_{x_1'}^{x_4'} \left[f(x') - 0 \right] dx' \right| \\
&= \left| \int_{x_1'}^{x_4'} \frac{f''(\beta(x'))}{2!}(x' - x_1')(x' - x_4')dx' \right| \\
&= O(h^4),
\end{aligned} \tag{46}
$$

where $\frac{f''(\beta(x'))}{2!}(x' - x_1')(x' - x_4')$ is the Lagrangian interpolation remainder of the linear interpolation polynomial with information $\{(x_1', 0), (x_4', 0)\}$ and $\beta(x') \in [x_1', x_4']$ for all x'. Hence, $|I_{i,j}^{\star}| = O(h^2)$.

For $\left(|I_{i,j}^{\star} \backslash \widetilde{I}_{i,j}^{\star}| + |\widetilde{I}_{i,j}^{\star} \backslash I_{i,j}^{\star}| \right) = O(h^6)$, similar to (46), one can find that it is sufficient to prove

$$
\left| \int_{x_1'}^{x_4'} \frac{f^{(4)}(\widetilde{\beta}(x'))}{4!}(x' - x_1')(x' - x_2')(x' - x_3')(x' - x_4')dx' \right| = O(h^6), \tag{47}
$$

where $\frac{f^{(4)}(\widetilde{\beta}(x'))}{4!}(x' - x_1')(x' - x_2')(x' - x_3')(x' - x_4')$ is the Lagrangian interpolation remainder of the cubic interpolation polynomial with information $\{(x_k', f(x_k')\}_{k=1}^4$. The approximation (47) holds naturally by Lemma 3.4 and Lemma 3.5.

□

Remark 3.7. *According to Lemma 3.3 and Lemma 3.5, we prove quantitatively that the upstream curved edges are actually smooth curves close to vertical or horizontal straight edges.*

Remark 3.8. *Notice that the conclusion is proved based on the assumption that* $\{v_k^{\star}\}_{k=1}^4$ *is obtained by exactly solving* (3)*. The numerical error occurred by the fourth-order RK integrator can be found to be not dominant in this error analysis. For conciseness, we skip this discussion.*

Proposition 3.9. *(L^2 stability) The numerical update given by* (17) *is unconditionally stable for transport equations with constant coefficients and periodic boundary condition, if* $\widetilde{u}(x, y)$ *is constructed by the fourth-order linear reconstruction.*

Proof. The proposition is proved by standard von Neumann analysis. The proof is arranged in Appendix C for conciseness. □

4. Numerical tests

4.1. Linear transport equations

In this subsection, two linear transport equations are chosen to test the non-splitting SL FV WENO scheme. We compare the non-splitting SL FV scheme with a fourth-order splitting-based SL FV WENO scheme [35] in terms of accuracy and efficiency. We adopt the 1-D WENO-ZQ [25] for the splitting scheme.

Unless specified, we set $\Delta t = \frac{\text{CFL}}{\frac{\max\{|a(x,y,t)|\}}{\Delta x} + \frac{\max\{|b(x,y,t)|\}}{\Delta y}}$ and CFL = 10.2. The PP limiter is applied for the problems with non-negative initial conditions for both schemes.

Example 4.1. *(Transport equation with constant coefficients). Consider*

$$u_t + u_x + u_y = 0, \quad x \in [-\pi, \pi], \quad y \in [-\pi, \pi]. \tag{48}$$

with a smooth initial condition $u(x, y, 0) = sin(x + y)$ and periodic boundary condition. The exact solution for this problem is $u(x, y, t) = sin(x + y - 2t)$.

Table 1: (Transport equation with constant coefficients). L^2 errors and corresponding orders of accuracy of the non-splitting and splitting-based schemes for (48) with $u(x, y, 0) = \sin(x + y)$ at $T = 2$.

	Non-splitting		**Splitting**	
mesh	L^2 error	order	L^2 error	order
40× 40	1.58E-04	—	2.23E-05	—
80× 80	8.40E-07	7.55	1.76E-07	6.99
160× 160	5.16E-08	4.03	1.96E-09	6.49
320× 320	3.22E-09	4.00	5.43E-11	5.17

In Table 1, we show the L^2 errors, corresponding orders of accuracy of the non-splitting and splitting-based SL FV WENO schemes at $T = 2$. For this problem, there is no temporal error for both schemes. Hence, the orders in Table 1 are spatial orders. The spatial order of the non-splitting SL FV WENO scheme is fourth as expected. The spatial order of the splitting-based SL FV WENO scheme is fifth. In Figure 6, we present the log-log plot of the

CPU times vs. the L^2 errors of both schemes. For this problem, the splitting-based SL FV WENO scheme is observed to be much more efficient than the non-splitting SL FV WENO scheme.

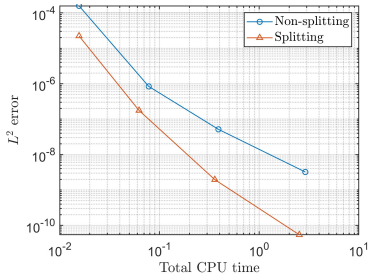


Figure 6: (Transport equation with constant coefficients). Log-log plot of the CPU times vs. the L^2 errors of the non-splitting and splitting-based schemes for (48) with $u(x, y, 0) = \sin(x + y)$ at $T = 2$.

Example 4.2. *(Swirling deformation flow). Consider*

$$u_t - (2\pi cos^2(\frac{x}{2})sin(y)g(t)u)_x + (2\pi sin(x)cos^2(\frac{y}{2})g(t)u)_y = 0, \quad x \in [-\pi, \pi], \quad y \in [-\pi, \pi], \tag{49}$$

where $g(t) = cos(\pi t/T)$ and $T = 1.5$. We consider (49) with zero boundary condition and a smooth initial condition

$$u(x, y, 0) = \begin{cases} r_0^b cos(\frac{r^b(\mathbf{x})\pi}{2r_0^b})^6 & \text{if } r^b(\mathbf{x}) < r_0^b, \\ 0, & \text{otherwise}, \end{cases} \tag{50}$$

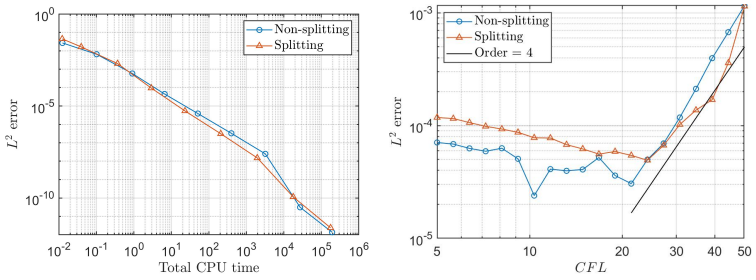
where $r_0^b = 0.3\pi$, $r^b(\mathbf{x}) = \sqrt{(x - x_0^b)^2 + (y - y_0^b)^2}$ and the center of the cosine bell $(x_0^b, y_0^b) = (0.3\pi, 0)$.

We give the L^2 errors and corresponding orders of accuracy of the non-splitting and spitting-based SL FV WENO schemes at $t = 1.5$ in Table 2. Fourth order accuracy is observed for the non-splitting SL FV scheme. Under the same mesh, we observe that the non-splitting SL FV scheme is more accurate than the splitting-based SL FV scheme. In Figure 7a, we show the log-log plot of the CPU times vs. the L^2 errors of both schemes with the same settings in Table 2. We observe that the CPU efficiency of the two schemes are very close. In Figure 7b, with a fixed mesh of 160×160, we present the

Table 2: (Swirling deformation flow). L^2 errors and corresponding orders of accuracy of the non-splitting and splitting-based schemes for (49) with initial condition (50) at $t = 1.5$.

	Non-splitting		**Splitting**	
mesh	L^2 error	order	L^2 error	order
40× 40	6.47E-03	—	1.63E-02	—
80× 80	5.82E-04	3.47	2.01E-03	3.02
160× 160	4.47E-05	3.70	9.42E-05	4.41
320× 320	3.90E-06	3.52	5.39E-06	4.13
640× 640	3.28E-07	3.57	3.11E-07	4.12
1280× 1280	2.51E-08	3.71	1.50E-08	4.37
2560× 2560	3.11E-11	9.66	1.15E-10	7.03
5120× 5120	1.37E-12	4.50	2.39E-12	5.59

log-log plot of the CFL numbers vs. the L^2 errors. When Δt is small enough, the spatial error dominates. Accumulated error increases with reducing CFL, or Δt, because more time steps are taken. Hence, the L^2 error slightly goes up as CFL is reduced. We can observe that the temporal orders of the both schemes are fourth when the temporal error starts to dominate for larger CFL.



(a) (b)

Figure 7: (Swirling deformation flow). Left: log-log plot of the CPU times vs. the L^2 errors with the same settings in Table 2. Right: log-log plot of the CFL numbers vs. the L^2 errors with a fixed mesh of 160×160 for (49) with initial condition (50) at $t = 1.5$.

Then, we test (49) *with zero boundary condition and a discontinuous*

initial condition shown in Figure 8. In Figure 9, we present the mesh plot and the contour plot of the numerical solution of the non-splitting SL FV scheme at $t = 0.75$. In Figure 10, we show the mesh plot, the contour plot, and two cross-sections of the numerical solution of the non-splitting SL FV scheme at $t = 1.5$. We observe that the non-splitting SL FV WENO scheme captures the complex structure of the solution. The PP property is also observed.

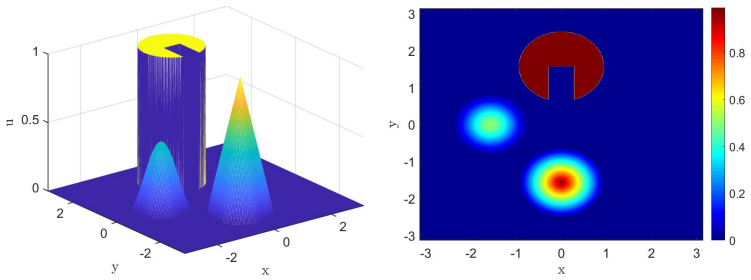


Figure 8: (Rigid body rotation). The mesh plot (left) and the contour plot (right) of the discontinuous initial data for (49).

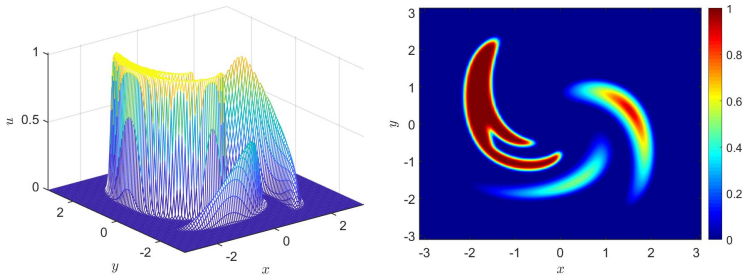


Figure 9: (Swirling deformation flow). The mesh plot (left) and the contour plot (right) of the numerical solution of the non-splitting SL FV WENO scheme for (49) with initial condition Figure 8 at $t = 0.75$.

4.2. Nonlinear Vlasov-Poisson system

The nonlinear VP system describes collisionless plasma with a negligible magnetic field. In this subsection, we consider the 1-D physical space and 1-D velocity space (1D1V) nonlinear VP system on $(x, v, t) \in \Omega_x \times \mathbb{R} \times \mathbb{R}^+$:

$$f_t + v f_x + E(x,t) f_v = 0, \tag{51}$$

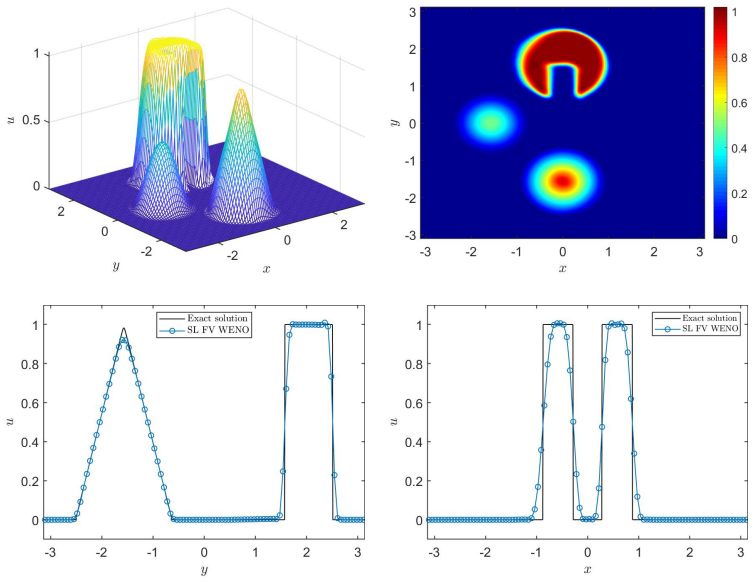


Figure 10: (Swirling deformation flow). Top left and top right panels are the mesh plot and contour plot of the numerical solution of the non-splitting SL FV WENO scheme for (49) with initial condition Figure 8 at $t = 1.5$. Bottom left and bottom right panels present the cross-sections at $x = 0$ and $y = 1.2$.

$$E(x,t) = -\phi_x, \quad -\phi_{xx}(x,t) = \rho(x,t), \tag{52}$$

where x and v are spatial position and velocity respectively, f(x,v,t) describes the probability of a particle arises at position x with velocity v at time t, E is the electric field, ϕ is the self-consistent electrostatic potential, and $\rho = \int_{\mathbb{R}} f(x,v,t)dv - \rho_0$ is the charge density with $\rho_0 = \frac{1}{|\Omega_x|}\int_{\Omega_x}\int_{\mathbb{R}} f(x,v,0)dvdx$. We assume periodic boundary condition on the x-dimension and zero boundary condition on the v-dimension.

For the nonlinear 1D1V VP system, we couple the non-splitting SL FV WENO scheme with a fourth-order RKEI in the same framework as in [36]. The fourth-order RKEI is a commutator-free Lie group method introduced in [33] with its Butcher tableau shown in Table 3.

Table 3: CF4.

0				
$\frac{1}{2}$	$\frac{1}{2}$			
$\frac{1}{2}$	0	$\frac{1}{2}$		
1	$\frac{1}{2}$	0	0	
	$-\frac{1}{2}$	0	1	
	$\frac{1}{4}$	$\frac{1}{6}$	$\frac{1}{6}$	$-\frac{1}{12}$
	$-\frac{1}{12}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{4}$

We briefly summarize the SL FV WENO scheme coupled with CF4 for the 1D1V VP system as follows:

$$\begin{aligned}
\overline{f}^{(1)} &= \overline{f}^n \\
\overline{f}^{(2)} &= SLWENO\left(\mathbf{V}\left(\overline{f}^{(1)}\right), \frac{1}{2}\Delta t\right)\overline{f}^n \\
\overline{f}^{(3)} &= SLWENO\left(\mathbf{V}\left(\overline{f}^{(2)}\right), \frac{1}{2}\Delta t\right)\overline{f}^n \\
\overline{f}^{(4)} &= SLWENO\left(-\frac{1}{2}\mathbf{V}\left(\overline{f}^{(1)}\right) + \mathbf{V}\left(\overline{f}^{(3)}\right), \frac{1}{2}\Delta t\right)\overline{f}^{(2)} \\
\overline{f}^{n+1} &= SLWENO\left(-\frac{1}{12}\mathbf{V}\left(\overline{f}^{(1)}\right) + \frac{1}{6}\mathbf{V}\left(\overline{f}^{(2)}\right) + \frac{1}{6}\mathbf{V}\left(\overline{f}^{(3)}\right) + \frac{1}{4}\mathbf{V}\left(\overline{f}^{(4)}\right), \frac{1}{2}\Delta t\right) \\
&\quad SLWENO\left(\frac{1}{4}\mathbf{V}\left(\overline{f}^{(1)}\right) + \frac{1}{6}\mathbf{V}\left(\overline{f}^{(2)}\right) + \frac{1}{6}\mathbf{V}\left(\overline{f}^{(3)}\right) - \frac{1}{12}\mathbf{V}\left(\overline{f}^{(4)}\right), \frac{1}{2}\Delta t\right)\overline{f}^n,
\end{aligned} \tag{53}$$

where $\mathbf{V}\left(\overline{f}^{(k)}\right)$ represents the numerical velocity field obtained by a given FV solution $\overline{f}^{(k)}$, $SLWENO\left(\mathbf{V}\left(\overline{f}^{(k)}\right), \frac{1}{2}\Delta t\right)\overline{f}^{(l)}$ represents the solution evolved from $\overline{f}^{(l)}$ with time step $\frac{1}{2}\Delta t$ and velocity field $\mathbf{V}\left(\overline{f}^{(k)}\right)$ by the non-splitting SL FV WENO scheme. For $\mathbf{V}\left(\overline{f}^{(k)}\right) := \left(v, \widetilde{E}(x)\right)$, we approximate the electric field as follows.

1. Compute the cell averages of the charge density $\{\overline{\rho}_i\} := \{\Delta v \sum_j \overline{f}_{i,j}^{(k)} - \rho_0\}$.
2. Compute the nodal values of the charge density $\{\rho_i\}$ at $\{x_i\}$ by $\{\overline{\rho}_i\}$ with fifth-order accuracy:
$$\rho_i = \frac{3}{640}\overline{\rho}_{i-2} - \frac{29}{480}\overline{\rho}_{i-1} + \frac{1067}{960}\overline{\rho}_i - \frac{29}{480}\overline{\rho}_{i+1} + \frac{3}{640}\overline{\rho}_{i+2} \quad \forall i. \tag{54}$$
3. Solve the Poisson090005s equation (52) and compute the nodal values of the electric field $\{E_i\}$ at $\{x_i\}$ by a Fast Fourier transform (FFT) solver.
4. Reconstruct a piecewise P^4 polynomial $\widetilde{E}(x)$ satisfying
$$\widetilde{E}(x) = \widetilde{E}^i(x) \quad \forall x \in I_i, \quad \forall i, \tag{55}$$
with $\widetilde{E}^i(x) \in P^4(I_i)$ being the interpolation polynomial interpolated by $\{(x_{i+l}, E_{i+l})\}_{l=-2}^{2}$ for all i.

In this subsection, standard tests such as Landau damping, two stream instability, and bump-on-tail instability are tested. We use an FFT solver for the Poisson090005s equation (52) for both non-splitting and splitting SL FV WENO schemes. Unless otherwise specified, we set the computational domain as $[0, 4\pi] \times [-v_{\max}, v_{\max}]$ with $v_{\max} = 2\pi$ and set $N_x = 128$, $N_v = 256$, CFL $= 10.2$, $\Delta t = \text{CFL}/\left(v_{\max}/\Delta x + \max\{|E|\}/\Delta v\right)$. We adopt the PP limiter for all the tests below.

Example 4.3. *(Landau damping). Consider the VP system with the initial condition*
$$f(x, v, t = 0) = \frac{1}{\sqrt{2\pi}}\left(1 + \alpha cos(kx)\right) exp\left(-\frac{v^2}{2}\right), \tag{56}$$
where $k = 0.5$, $\alpha = 0.01$ for the weak Landau damping and $k = 0.5$, $\alpha = 0.5$ for the strong Landau damping.

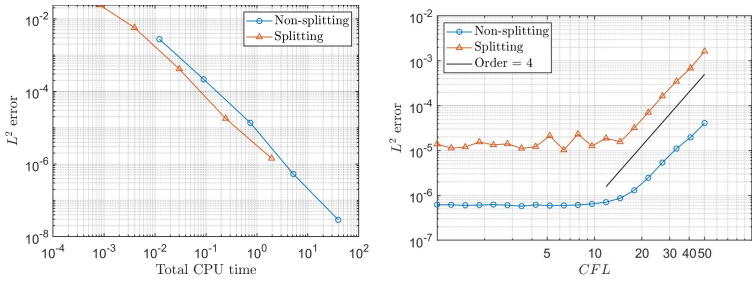
In Table 4, we present the L^2 errors and corresponding orders of accuracy of the non-splitting and splitting-based SL FV WENO schemes at $T = 2$ with $CFL = 10.2$. The errors are computed by comparing the solution to a reference solution with mesh refinement. As shown, the order of accuracy for the non-splitting SL FV WENO scheme is 4th as expected. For the same mesh, the non-splitting scheme is observed to be more accurate than the splitting-based scheme. In Figure 11a, we show the log-log plot of the CPU times vs. the L^2 errors of the non-splitting and splitting-based SL FV WENO schemes with the same settings in Table 4. As shown, the splitting-based scheme is more efficient. In Figure 11b, the log-log plot of the CFL numbers vs. the L^2 errors of the non-splitting and splitting-based SL FV schemes with a fixed mesh of 128×128 at $T = 2$ is shown. The fourth-order temporal orders of both schemes are observed.

Table 4: (Strong Landau damping). L^2 errors and corresponding orders of accuracy of the non-splitting and splitting-based SL FV WENO schemes for strong Landau damping at $T = 2$.

	Non-splitting		**Splitting**	
mesh	L^2 error	order	L^2 error	order
16× 16	2.74E-03	—	2.40E-02	—
32× 32	2.15E-04	3.67	5.68E-03	2.08
64× 64	1.36E-05	3.98	4.15E-04	3.77
128× 128	5.30E-07	4.68	1.79E-05	4.53
256× 256	2.90E-08	4.19	1.43E-06	3.65

In Figure 12, we show the mesh plot and the contour plot of the numerical solution of the non-splitting SL FV WENO scheme at $T = 40$. We observe that the filamentation structure of strong Landau damping problem is well captured and the numerical solution is non-negative.

For 1D1V VP system, there are several conservative quantities including mass, L^p norms, energy and entropy [5]. In Figure 13, we present the relative deviation of mass and the L^1 norm for the non-splitting SL FV WENO scheme for weak and strong Landau damping which are observed to be around $O(10^{-13})$. Hence, the proposed non-splitting SL FV WENO scheme enjoys the mass conservation and PP property. For other conservative quantities, the proposed scheme is not designed to exactly preserve them. The performance of preserving those quantities is similar to the results in [5, 24]. We



(a) (b)

Figure 11: (Strong Landau damping). Left: log-log plot of the CPU times vs. the L^2 errors with the same settings in Table 4. Right: log-log plot of the CFL numbers vs. the L^2 errors with a fixed mesh of 128×128 for strong Landau damping at $T = 2$.

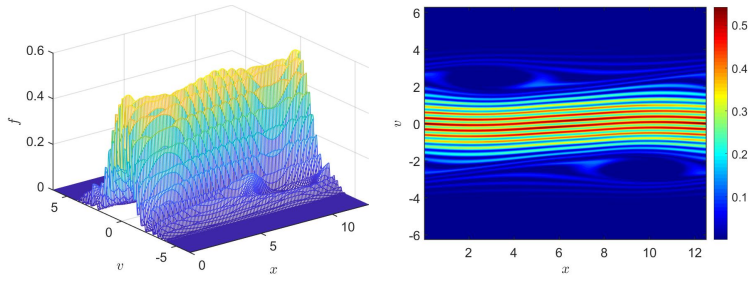


Figure 12: (Strong Landau damping). The mesh plot (left) and the contour plot (right) of the numerical solution of the non-splitting SL FV WENO scheme at $T = 40$.

skip them for saving space. For the same reason, we skip presenting the time history of the electric field to save space.

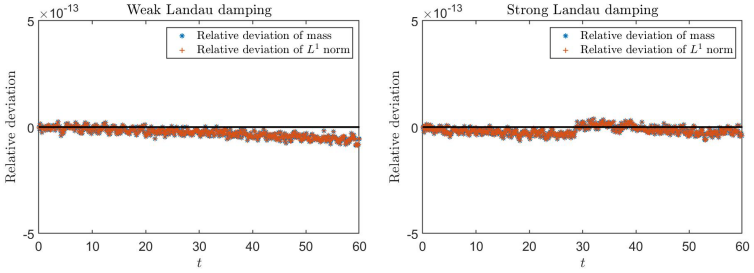


Figure 13: (Landau damping). Performance of mass conservation and PP properties of the non-splitting SL FV WENO scheme for the weak (left) and strong (right) Landau damping with $v_{\max} = 10$.

Example 4.4. *(Two stream instability [37]). Consider the symmetric warm two stream instability, i.e. the VP system with the initial condition*

$$\begin{aligned} &f(x,v,t=0) \\ =&\frac{2}{7\sqrt{2\pi}}(1+5v^2)\left(1+\alpha\left((cos(2kx)+cos(3kx))/1.2+cos(kx)\right)\right)exp\left(-\frac{v^2}{2}\right), \end{aligned} \tag{57}$$

where $\alpha = 0.01$ and $k = 0.5$. On the left of Figure 14, we show the contour plot of the numerical solution of the non-splitting SL FV WENO scheme at $T = 53$. The result is comparable to the results in [13, 21].

Example 4.5. *(Bump-on-tail instability [38, 24].) Consider the bump-on-tail instability with the initial condition*

$$f(x,v,t=0) = \left(n_p exp\left(-\frac{v^2}{2}\right) + n_b exp\left(-\frac{(v-u)^2}{2v_t^2}\right)\right)(1+0.04cos(kx)), \tag{58}$$

where $n_p = \frac{9}{10\sqrt{2\pi}}$, $n_b = \frac{2}{10\sqrt{2\pi}}$, $u = 4.5$, $v_t = 0.5$ and $k = 0.3$. The computational domain for this test is $[0, \frac{20}{3}\pi] \times [-13, 13]$. On the right of Figure 14, we show the contour plot of the numerical solution of the non-splitting SL FV WENO scheme at $T = 40$. As shown, the numerical result is consistent with the ones in [24, 35].

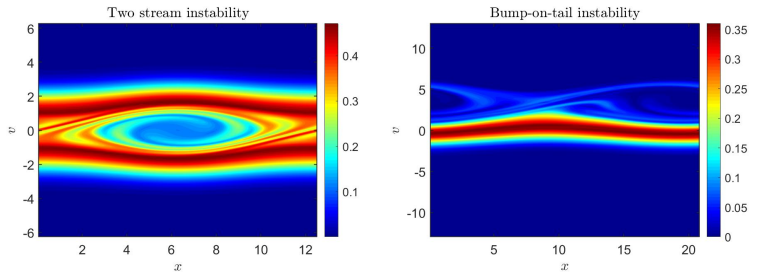


Figure 14: Numerical solution of the non-splitting SL FV WENO scheme for the two stream instability at $T = 53$ (left) and bump-on-tail instability (right) at $T = 40$.

4.3. Guiding center Vlasov model

The guiding center Vlasov model describes a highly magnetized plasma in the transverse of a tokamak [39, 4]. It can be written as

$$\rho_t + \nabla \cdot \left(\mathbf{E}^{\perp}\rho\right) = 0, \tag{59}$$

$$-\Delta\Phi = \rho, \quad \mathbf{E}^{\perp} = (-\Phi_y, \Phi_x), \tag{60}$$

where $\rho(x, y, t)$ represents the charge density and $\mathbf{E}$ is the electric field.

For the 2-D guiding center Vlasov model (59)-(60), we apply the same procedure introduced in (53). The only difference is the simulation of the velocity field. Similar to (53), we denote a numerical velocity field obtained by a FV solution $\overline{\rho}^{(k)}$ by $\mathbf{V}\left(\overline{\rho}^{(k)}\right) := \left(\widetilde{E}_1(x, y), \widetilde{E}_2(x, y)\right)$. We briefly summarize the procedure to construct $\mathbf{V}\left(\overline{\rho}^{(k)}\right)$ as follows.

1. Compute the nodal values of the charge density $\{\rho_{i,j}\}$ at (x_i, y_j) by $\{\overline{\rho}_{i,j}^{(k)}\}$ with fourth-order accuracy:

   $$\rho_{i,j} = P(x_i, y_j), \tag{61}$$

   where $P \in P^3(I_{i,j})$ is obtained with the same stencil as $q_0(x, y)$ introduced in Section 2.2.
2. Solve the Poisson090005s equation (60) and compute the nodal values of the velocity field $\{(E_1, E_2)\}$ at $\{(x_i, y_j)\}$ by a FFT solver.
3. Reconstruct two piecewise P^3 polynomial $\left(\widetilde{E}_1(x, y), \widetilde{E}_2(x, y)\right)$ with the same stencil of $q_0(x, y)$ introduced in Section 2.2 except that we use the nodal values here.

For this problem, we set $N_x = 256$, $N_y = 256$, and

$$\Delta t = \mathrm{CFL}/\left(\max\{|E_1|\}/\Delta x + \max\{|E_2|\}/\Delta y\right).$$

Example 4.6. *(Kelvin-Helmholtz instability problem). Consider the guiding center Vlasov model with initial condition*

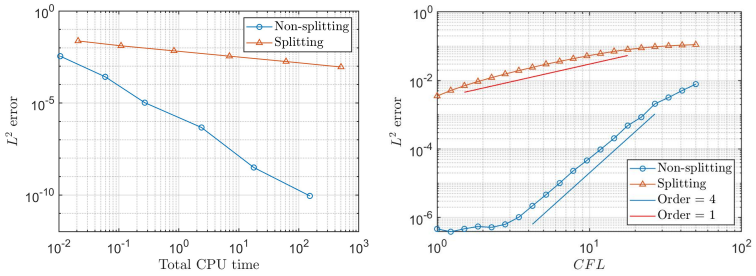
$$u(x, y, 0) = sin(y) + 0.015cos(kx), \quad x \in [0, 4\pi], \quad y \in [0, 2\pi], \tag{62}$$

where $k = 0.5$, and with the periodic boundary condition. In Table 5, we present the L^2 errors and corresponding orders of accuracy of the non-splitting and splitting-based SL FV WENO schemes at $T = 5$ with CFL = 10.2. For this problem, the reference solutions are obtained in the same way as in strong Landau damping. We observe that the order of accuracy for the non-splitting SL FV WENO scheme is fourth as expected. However, the convergence rate of the splitting-based SL FV scheme is reduced to first order. In Figure 15a, we show the log-log plot of the CPU times vs. the L^2 errors of the non-splitting and splitting-based SL FV WENO schemes with the same settings in Table 5. For this problem, the non-splitting scheme is observed to be much more efficient than the splitting-based scheme. Figure 15b shows the log-log plot of the CFL numbers vs. the L^2 errors of both schemes with a fixed mesh of 128×128 at $T = 5$. As shown, the temporal order of the non-splitting SL FV WENO scheme is fourth and that of the splitting scheme is first.

Table 5: (Kelvin-Helmholtz instability problem). L^2 errors and corresponding orders of accuracy of the non-splitting and splitting-based SL FV WENO schemes for Kelvin-Helmholtz instability problem at $T = 5$ with CFL = 1.

	Non-splitting		**Splitting**	
mesh	L^2 error	order	L^2 error	order
16× 16	3.49E-03	—	2.35E-02	—
32× 32	2.64E-04	3.73	1.28E-02	0.88
64× 64	1.04E-05	4.66	6.80E-03	0.91
128× 128	4.65E-07	4.49	3.51E-03	0.95
256× 256	3.12E-09	7.22	1.79E-03	0.97
512× 512	8.89E-11	5.14	9.03E-04	0.99

In Figure 16, we provide the contour plots of the numerical solution of the non-splitting SL FV WENO scheme with CFL = 1 and CFL = 10.2 at



(a) (b)

Figure 15: (Kelvin-Helmholtz instability problem). Left: log-log plot of the CPU times vs. the L^2 errors of the two schemes with the same settings in Table 5. Right: log-log plot of the CFL numbers vs. the L^2 errors of the two schemes with a fixed mesh of 128×128 for Kelvin-Helmholtz instability problem at $T = 5$.

$T = 40$. *The shapes of the two results are consistent with the existing results in the literature [36, 21]. In Figure 17, we present the contour plots of the numerical solution of the splitting-based SL FV WENO scheme with CFL = 0.1 and CFL = 1 at* $T = 40$. *We observe that the numerical solution is similar to the ones in Figure 16 when CFL = 0.1. When CFL = 1, the solution from the splitting-based scheme is observed to have significant deviation from the reference solution.*

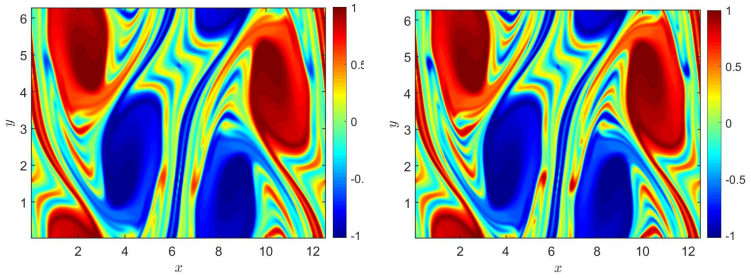


Figure 16: (Kelvin-Helmholtz instability problem). Contour plots of the numerical solution of the non-splitting SL FV WENO scheme with CFL = 1 (left) and with CFL = 10.2 (right) at $T = 40$.

It is well known that the 2-D guiding center Vlasov model has three conservative physical quantities over time, i.e. mass, energy, and enstrophy

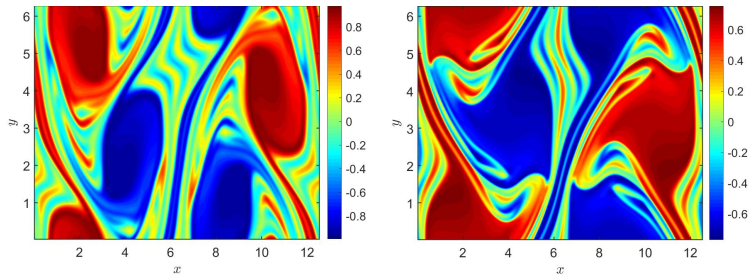


Figure 17: (Kelvin-Helmholtz instability problem). Contour plots of the numerical solution of the splitting-based SL FV WENO scheme with CFL = 0.1 (left) and CFL = 1 (right) at $T = 40$.

[21, 36]. For mass conservation, both the non-splitting and splitting-based SL FV WENO schemes are mass conservative by $O(10^{-13})$ magnitude deviation. We skip this for saving space. We show the relative deviation of energy and enstrophy for the two schemes in Figure 18 with CFL = 10.2. We observe that the performance of the non-splitting SL FV WENO scheme is better than the splitting-based SL FV WENO scheme.

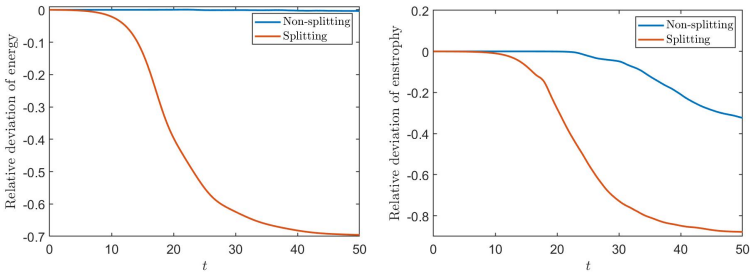


Figure 18: (Kelvin-Helmholtz instability problem). Relative Deviation of energy (left) and enstrophy (right) for the non-splitting and splitting-based SL FV WENO schemes with CFL = 10.2.

4.4. Incompressible Euler equation

The 2-D incompressible Euler equation in vorticity-stream function formulation reads

$$\omega_t + \nabla \cdot (\mathbf{u}\omega) = 0, \tag{63}$$

$$\Delta\psi = \omega, \quad \mathbf{u} = (-\psi_y, \psi_x), \tag{64}$$

where $\omega(x, y, t)$ is the vorticity of the fluid, ψ is the stream-function determined by Poisson090005s equation, and $\mathbf{u} := (u_1, u_2)$ is the velocity field. The form of (63)-(64) is almost the same with (59)-(60) except the sign of the Poisson090005s equation. Hence, the procedure for solving (63)-(64) is similar to the one of solving the guiding center Vlasov model. We skip the description of this procedure for brevity. Similar to the guiding center Vlasov model, the mass, energy, and enstrophy are conserved for the 2-D incompressible Euler equation in vorticity-stream function formulation [21, 36]. For this problem, we set $N_x = 256$, $N_y = 256$, CFL $= 10.2$, and set the time step as $\Delta t = \text{CFL}/\left(\max\{u_1\}/\Delta x + \max\{u_2\}/\Delta y\right)$.

Example 4.7. *(Vortex patch problem). Consider the incompressible Euler equations on the domain* $[0, 2\pi] \times [0, 2\pi]$ *with the initial condition*

$$u(x, y, 0) = \begin{cases} -1, & \text{if } (x, y) \in [\frac{\pi}{2}, \frac{3\pi}{2}] \times [\frac{\pi}{4}, \frac{3\pi}{4}], \\ 1, & \text{if } (x, y) \in [\frac{\pi}{2}, \frac{3\pi}{2}] \times [\frac{5\pi}{4}, \frac{7\pi}{4}], \\ 0, & \text{otherwise}, \end{cases} \tag{65}$$

and the periodic boundary condition.

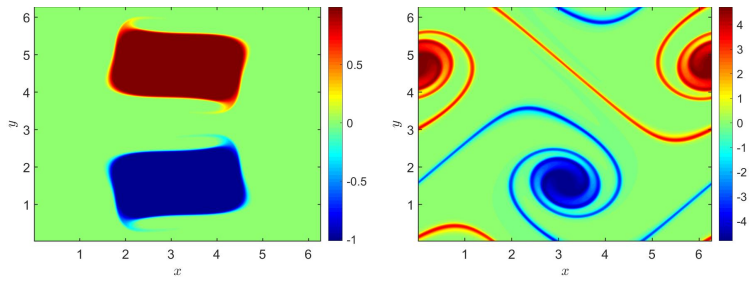
In Figure 19a, we show the contour plot of the numerical solution of the non-splitting SL FV WENO scheme at $T = 10$*. The numerical solution is non-oscillatory thanks to the essentially non-oscillatory nature of the WENO reconstruction procedure in space. This result is comparable to the existing ones in the literature [17, 36, 21].*

In Figure 20, We present the performance of preserving energy and enstrophy for the non-splitting and splitting-based SL FV WENO schemes. We observe that the non-splitting scheme performs better.

Example 4.8. *(Shear flow problem). Consider the incompressible Euler equations in the domain* $[0, 2\pi] \times [0, 2\pi]$ *with the initial condition*

$$u(x, y, 0) = \begin{cases} \delta cos(x) - \frac{1}{\rho} sech^2\left(\frac{y - \pi/2}{\rho}\right), & \text{if } y \le \pi, \\ \delta cos(x) + \frac{1}{\rho} sech^2\left(\frac{3\pi/2 - y}{\rho}\right), & \text{if } y > \pi, \end{cases} \tag{66}$$

where $\delta = 0.05$ *and* $\rho = \pi/15$*, and the periodic boundary condition.*



(a) (b)

Figure 19: Contour plots of the numerical solution of the non-splitting SL FV WENO scheme with CFL = 10.2 for vortex patch problem at $T = 10$ (left) and shear flow problem at $T = 8$ (right).

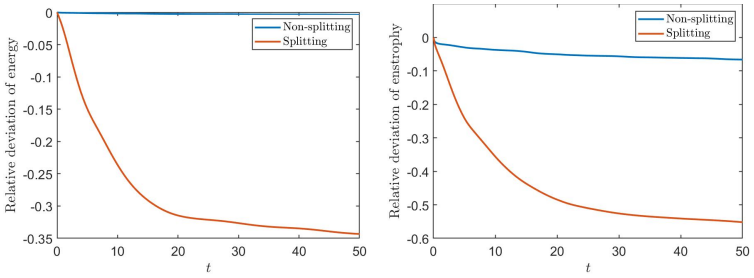


Figure 20: (Vortex patch problem). Relative Deviation of energy (left) and enstrophy (right) for the non-splitting and splitting-based SL FV WENO schemes with CFL = 10.2.

In Figure 19b, we provide the contour plot of the numerical solution of the non-splitting SL FV WENO scheme at $T = 8$. The performance is similar to the results in [17, 36, 21]. In Figure 21, the time history of the relative deviation of the energy and enstrophy for the non-splitting and splitting-based SL FV WENO schemes is provided. The non-splitting SL FV WENO scheme has better performance compared with the splitting-based SL FV WENO scheme.

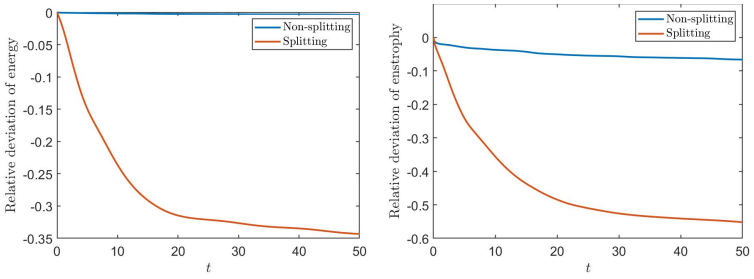


Figure 21: (Shear flow problem). Relative Deviation of energy (left) and enstrophy (right) for the non-splitting and splitting-based SL FV WENO schemes with CFL = 10.2.

5. Conclusion

In this paper, we introduce a fourth-order SL FV WENO scheme without operator splitting for 2-D linear transport equations, the nonlinear Vlasov-Poisson (VP) system, the guiding center Vlasov model and the incompressible Euler equation in the vorticity-stream function formulation. The proposed SL FV WENO scheme is mass conservative, PP, unconditionally stable under linearized settings, and fourth-order accurate in space and time. Numerical performances are observed through an extensive set of test problems.

6. Acknowledgements

The work of the first and fourth authors was partially supported by National Natural Science Foundation (China) [grant number 12071392]. The work of the second author was partially support by the Foundation of Beijing Normal University [grants numbers 28704-310432105]. The work of the third author was partially support by NSF [grants numbers NSF-DMS-1818924

and NSF-DMS-2111253], Air Force Office of Scientific Research FA9550-18-1-0257.

Appendices

A. Constructing a cubic curve

Assume that the coordinate of $\{v_k^\star\}$ in $x-y$ space is $\{(x_k, y_k)\}$. Denote the cubic curve (6) by $\mathcal{C}$. Then, we do the following steps to construct (6) (see Figure A.22).

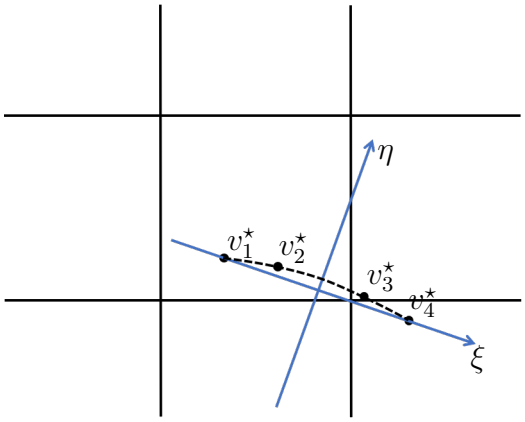


Figure A.22: The dashed line represents $\mathcal{C}$; the blue arrows represent the new coordinate $\xi-\eta$; the black dots are the four characteristic feet $\{v_k^\star\}$.

(a) Based on $v_1^\star$ and $v_4^\star$, we construct an affine coordinate transformation from $x-y$ to $\xi-\eta$ such that (x_1, y_1) and (x_4, y_4) are $(-1, 0)$ and $(1, 0)$ in $\xi-\eta$ space, respectively:

$$\begin{cases} \xi(x,y) = \quad ax + by + c, \\ \eta(x,y) = -bx + ay + d, \end{cases} \tag{A.1}$$

where

$$a = \frac{2(x_4 - x_1)}{(x_1 - x_4)^2 + (y_1 - y_4)^2},$$

$$b = \frac{2(y_4 - y_1)}{(x_1 - x_4)^2 + (y_1 - y_4)^2},$$
$$c = \frac{x_1^2 - x_4^2 + y_1^2 - y_4^2}{(x_1 - x_4)^2 + (y_1 - y_4)^2},$$
$$d = \frac{2(x_1 y_4 - x_4 y_1)}{(x_1 - x_4)^2 + (y_1 - y_4)^2}.$$

The reverse transformation of (A.1) can be constructed accordingly:

$$\begin{cases} x = \frac{x_4 - x_1}{2}\xi - \frac{y_4 - y_1}{2}\eta + \frac{x_4 + x_1}{2}, \\ y = \frac{y_4 - y_1}{2}\xi + \frac{x_4 - x_1}{2}\eta + \frac{y_4 + y_1}{2}. \end{cases} \tag{A.2}$$

(b) Get the $\xi - \eta$ coordinates of $v_2^\star$ and $v_3^\star$ by (A.1) as (ξ_2, η_2) and (ξ_3, η_3). Based on $(-1, 0)$, (ξ_2, η_2), (ξ_3, η_3) and $(1, 0)$, we construct a cubic interpolation

$$\eta(\xi) = \eta_2 \frac{(\xi^2 - 1)(\xi - \xi_3)}{(\xi_2^2 - 1)(\xi_2 - \xi_3)} + \eta_3 \frac{(\xi^2 - 1)(\xi - \xi_2)}{(\xi_3^2 - 1)(\xi_3 - \xi_2)}. \tag{A.3}$$

(c) Substituting (A.3) into (A.2), we obtain the parametric equation (6) of $\mathcal{C}$.

B. The Clipping procedure

In Appendix B.1, we first introduce the procedure to determine the intersection points between a cubic curve and the Eulerian mesh. Then, in Appendix B.2 and Appendix B.3, we introduce methods to determine the outer and inner integral segments based on the intersection points.

B.1. Determining intersection points

Take $\mathcal{C}$ as an example, we do the following steps to determine all the intersection points between $\mathcal{C}$ and the Eulerian mesh (see Figure B.23 (a)).

(a) By solving $x'(\xi) = 0$ and $y'(\xi) = 0$, we find the minimum and maximum values of $\mathcal{C}$ in x and y directions and determine the mesh lines intersecting $\mathcal{C}$ (see Figure B.23 (a)).

(b) If $x = x_{i-\frac{1}{2}}$ intersect $\mathcal{C}$, we solve

$$x(\xi) - x_{i-\frac{1}{2}} = 0, \tag{B.1}$$

and choose the real single roots in $[-1, 1]$ to determine the points in $x = x_{i-\frac{1}{2}}$. We do the same operation for all the lengthways mesh lines determined in step (a). We define such intersection points as type-1 intersection points.

(c) If $y = y_{j-\frac{1}{2}}$ intersect $\mathcal{C}$, we solve

$$y(\xi) - y_{j-\frac{1}{2}} = 0, \tag{B.2}$$

and choose the real single roots in $[-1, 1]$ to the intersection points in $y = y_{j-\frac{1}{2}}$. Similarly, we do the same thing for all the widthways mesh lines determined in step (a). We define such intersections as type-2 intersection points.

(d) Order all the intersection points by ξ from small to large.

B.2. Determining outer integral segments

For a single cubic curved edge, taking $\mathcal{C}$ as an example, the procedure to obtain the outer integral segments are shown as follows.

(a) We connect $v_1^\star$, the intersections between $v_1^\star$ and $v_4^\star$, and $v_4^\star$ in turn based on their ξ values from small to large along the cubic curve $\mathcal{C}$. Then, we define these segments as the outer segments in $\mathcal{C}$, denoted as $\{\mathcal{C}_k\}$ (see Figure B.23 (b)). Finally, we determine the location or index of a outer segment by the information of the start and end point of this segment.

(b) In a given cubic-curved quadrilateral upstream cell, we redefine the direction of these outer segments as counterclockwise with respect to $\widetilde{I}_{i,j}^\star$ and accomplish the determination of $\{\mathcal{L}_{i,j;p,q}^k\}$.

B.3. Determining inner integral segments

We determine the inner integral segments, $\{\mathcal{S}_{i,j;p,q}^k\}$, of $\widetilde{I}_{i,j}^\star$ by the following steps.

(a) Determine the mesh lines intersecting $\widetilde{I}^{\star}_{i,j}$ by the the maximum and minimum values of $\widetilde{I}^{\star}_{i,j}$.

(b) If $x = x_{i-\frac{1}{2}}$ intersects $\widetilde{I}^{\star}_{i,j}$, we first collect all the intersection points in $x = x_{i-\frac{1}{2}}$ of the four edges of $\widetilde{I}^{\star}_{i,j}$.

(c) After all the intersection points in $x = x_{i-\frac{1}{2}}$ are collected, we order the collected intersections and vertexes by their y coordinates from small to large, denoted as $\{p_k\}$. Then, we define $\overline{p_1p_2}$ and $\overline{p_3p_4}$, if they exist, as the super inner segments of $\widetilde{I}^{\star}_{i,j}$ in $x = x_{i-\frac{1}{2}}$ (see Figure B.24).

(d) For a given lengthways super inner segment, say $\overline{p_1p_2}$, we determine the inner integral segments to be the subsegments of $\overline{p_1p_2}$ belong to specific Eulerian cells. The location or index and the direction of a given inner integral segments can be easily determined by the Eulerian cell it belongs to (see Figure B.24 (a)).

(e) We do **step (b)-(d)** for all lengthways mesh lines determined in **step (a)** and determine all the lengthways inner integral segments of $\widetilde{I}^{\star}_{i,j}$.

(f) Then, we perform similar procedures to determine all the widthways inner segments of $\widetilde{I}^{\star}_{i,j}$.

In practical programming, two implementation issues arise due to the truncation error of the floating point system. Firstly, an intersection point near one end of a cubic curve can be incorrectly determined when this end is very close to an Eulerian mesh line. Secondly, when a cubic curve and a mesh line almost overlap, the locations or indices of outer segments can be incorrectly determined. Both the two kinds of mistakes cause chaos in latter integral procedure. Here, we just remind readers the two phenomena and do not state a standard way to treat them, since they are not the key points of this paper.

C. Proof of Proposition 3.9

Proof. Consider (2) with constant coefficients, $a(x, y, t) \equiv a$, $b(x, y, t) \equiv b$, and periodic boundary condition. We define that $\kappa_1 = \frac{a\Delta t}{\Delta x}$ and $\kappa_2 = \frac{b\Delta t}{\Delta x}$. Without loss of generality, we define that $a > 0$, $b > 0$, $0 \leq \kappa_1 \leq 1$, and $0 \leq \kappa_2 \leq 1$. When κ_1 or κ_2 is greater than 1, the linear scheme could reduce

to solution shifting on a uniform mesh together with the scheme having $0 \le \kappa_1 \le 1$ and $0 \le \kappa_2 \le 1$.

Then (17) with linear reconstruction is summarized as

$$\begin{aligned}\overline{u}_{i,j}^{n+1} = \frac{1}{\Delta x \Delta y}\Bigg[& \int_{x_{i-\frac{1}{2}}-\kappa_1\Delta x}^{x_{i-\frac{1}{2}}} \int_{y_{j-\frac{1}{2}}-\kappa_2\Delta y}^{y_{j-\frac{1}{2}}} \widetilde{u}^{(i-1,j-1)}(x,y)dxdy \\ & + \int_{x_{i-\frac{1}{2}}}^{x_{i-\frac{1}{2}}+(1-\kappa_1)\Delta x} \int_{y_{j-\frac{1}{2}}-\kappa_2\Delta y}^{y_{j-\frac{1}{2}}} \widetilde{u}^{(i,j-1)}(x,y)dxdy \\ & + \int_{x_{i-\frac{1}{2}}-\kappa_1\Delta x}^{x_{i-\frac{1}{2}}} \int_{y_{j-\frac{1}{2}}}^{y_{j-\frac{1}{2}}+(1-\kappa_2)\Delta y} \widetilde{u}^{(i-1,j)}(x,y)dxdy \\ & + \int_{x_{i-\frac{1}{2}}}^{x_{i-\frac{1}{2}}+(1-\kappa_1)\Delta x} \int_{y_{j-\frac{1}{2}}}^{y_{j-\frac{1}{2}}+(1-\kappa_2)\Delta y} \widetilde{u}^{(i,j)}(x,y)dxdy\Bigg].\end{aligned} \tag{C.1}$$

We prove the proposition via von Neumann stability analysis by assuming

$$\overline{u}_{p,q}^{n} = \overline{u}^{n} e^{I\xi_1 p\Delta x} e^{I\xi_2 q\Delta y} \quad \text{for} \quad p = i-3, i-2, \ldots, i+2;\ q = j-3, j-2, \ldots, j+2 \tag{C.2}$$

and

$$\overline{u}_{i,j}^{n+1} = \overline{u}^{n+1} e^{I\xi_1 i\Delta x} e^{I\xi_2 j\Delta y}, \tag{C.3}$$

where $I = \sqrt{-1}$. Substituting (C.2) and (C.3) into (C.1), we have

$$\overline{u}^{n+1} = A(\kappa_1, \kappa_2, \zeta_1, \zeta_2)\overline{u}^{n}, \tag{C.4}$$

where $\zeta_1 = \xi_1\Delta x$, $\zeta_2 = \xi_2\Delta y$, and $A(\kappa_1, \kappa_2, \zeta_1, \zeta_2)$ is the amplification factor. The explicit expression of $A(\kappa_1, \kappa_2, \zeta_1, \zeta_2)$ is extremely complicated. Hence, we skip this expression for brevity. Now, it is sufficient to verify that $|A(\kappa_1, \kappa_2, \zeta_1, \zeta_2)| \le 1$ for any $\kappa_1,\ \kappa_2 \in [0,1]$, and $\zeta_1,\ \zeta_2 \in [0, 2\pi]$. Since we can not determine the maximum and minimum values of $|A(\cdot,\cdot,\cdot,\cdot)|$ theoretically, we numerically verify this relation by sampling 1000 uniform points in κ_1, κ_2, ζ_1, and ζ_2 domains, respectively. We find that all the modulus of A computed by the sampled points are not greater than 1, which validates Proposition 3.9.

□

[1] P. H. Lauritzen, R. D. Nair, P. A. Ullrich, A conservative semi-Lagrangian multi-tracer transport scheme (CSLAM) on the cubed-sphere grid, Journal of Computational Physics 229 (2010) 1401–1424.

[2] W. Guo, R. D. Nair, J. M. Qiu, A Conservative Semi-Lagrangian Discontinuous Galerkin Scheme on the Cubed-Sphere, Monthly Weather Review 142 (2014) 457–475.

[3] F. Filbet, E. Sonnendrcker, P. Bertrand, Conservative Numerical Schemes for the Vlasov Equation, Journal of Computational Physics 172 (2001) 166–187.

[4] N. Crouseilles, M. Mehrenberger, E. Sonnendrcker, Conservative semi-Lagrangian schemes for Vlasov equations, Journal of Computational Physics 229 (2010) 1927–1953.

[5] J. A. Rossmanith, D. C. Seal, A positivity-preserving high-order semi-Lagrangian discontinuous Galerkin scheme for the VlasovPoisson equations, Journal of Computational Physics 230 (2011) 6203–6232.

[6] T. F. Russell, M. A. Celia, An overview of research on EulerianLagrangian localized adjoint methods (ELLAM), Advances in Water Resources 25 (2002) 1215–1231.

[7] J. Carpio, J. Prieto, An anisotropic, fully adaptive algorithm for the solution of convection-dominated equations with semi-Lagrangian schemes, Computer Methods in Applied Mechanics and Engineering 273 (2014) 77–99.

[8] A. Puigferrat, M. Mas, I. de Pouplana, G. Casas, E. Oate, Semi-lagrangian formulation for the advectiondiffusionabsorption equation, Computer Methods in Applied Mechanics and Engineering 380 (2021) 113807.

[9] F. X. Giraldo, The LagrangeGalerkin Spectral Element Method on Unstructured Quadrilateral Grids, Journal of Computational Physics 147 (1998) 114–146.

[10] F. Giraldo, J. Perot, P. Fischer, A spectral element semi-Lagrangian (SESL) method for the spherical shallow water equations, Journal of Computational Physics 190 (2003) 623–650.

[11] H. Natarajan, P. Popov, G. Jacobs, A high-order semi-Lagrangian method for the consistent Monte-Carlo solution of stochastic Lagrangian driftdiffusion models coupled with Eulerian discontinuous spectral element method, Computer Methods in Applied Mechanics and Engineering 384 (2021) 114001.

[12] M. Restelli, L. Bonaventura, R. Sacco, A semi-Lagrangian discontinuous Galerkin method for scalar advection by incompressible flows, Journal of Computational Physics 216 (2006) 195–215.

[13] J.-M. Qiu, C.-W. Shu, Positivity preserving semi-Lagrangian discontinuous Galerkin formulation: Theoretical analysis and application to the VlasovPoisson system, Journal of Computational Physics 230 (2011) 8386–8409.

[14] X. Cai, W. Guo, J.-M. Qiu, A high order conservative semi-Lagrangian discontinuous Galerkin method for two-dimensional transport simulations, Journal of Scientific Computing 73 (2017) 514–542.

[15] J. A. Carrillo, F. Vecil, Nonoscillatory Interpolation Methods Applied to Vlasov-Based Models, SIAM Journal on Scientific Computing 29 (2007) 1179–1206.

[16] E. Cristiani, M. Falcone, Fast Semi-Lagrangian Schemes for the Eikonal Equation and Applications, SIAM Journal on Numerical Analysis 45 (2007) 1979–2011.

[17] J.-M. Qiu, C.-W. Shu, Conservative high order semi-Lagrangian finite difference WENO methods for advection in incompressible flow, Journal of Computational Physics 230 (2011) 863–889.

[18] T. N. Phillips, A. J. Williams, A Semi-Lagrangian Finite Volume Method for Newtonian Contraction Flows, SIAM Journal on Scientific Computing 22 (2001) 2152–2177.

[19] A. Christlieb, W. Guo, M. Morton, J.-M. Qiu, A high order time splitting method based on integral deferred correction for semi-Lagrangian Vlasov simulations, Journal of Computational Physics 267 (2014) 7–27.

[20] F. Huot, A. Ghizzo, P. Bertrand, E. Sonnendrcker, O. Coulaud, Instability of the time splitting scheme for the one-dimensional and relativistic

VlasovMaxwell system, Journal of Computational Physics 185 (2003) 512–531.

[21] T. Xiong, G. Russo, J. Qiu, Conservative Multi-dimensional Semi-Lagrangian Finite Difference Scheme: Stability and Applications to the Kinetic and Fluid Simulations, Journal of Scientific Computing (2019).

[22] J. Cheng, C.-W. Shu, A high order accurate conservative remapping method on staggered meshes, Applied Numerical Mathematics 58 (2008) 1042–1060.

[23] P. H. Lauritzen, R. D. Nair, P. A. Ullrich, A conservative semi-Lagrangian multi-tracer transport scheme (CSLAM) on the cubed-sphere grid, Journal of Computational Physics 229 (2010) 1401–1424.

[24] X. Cai, W. Guo, J.-M. Qiu, A high order semi-Lagrangian discontinuous Galerkin method for VlasovPoisson simulations without operator splitting, Journal of Computational Physics 354 (2018) 529–551.

[25] J. Zhu, J. Qiu, A new fifth order finite difference WENO scheme for solving hyperbolic conservation laws, Journal of Computational Physics 318 (2016) 110–121.

[26] X.-D. Liu, S. Osher, T. Chan, Weighted essentially non-oscillatory schemes, Journal of Computational Physics 115 (1994) 200–212.

[27] G.-S. Jiang, C.-W. Shu, Efficient implementation of weighted ENO schemes, Journal of Computational Physics 126 (1996) 202–228.

[28] C.-W. Shu, Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws, in: Advanced numerical approximation of nonlinear hyperbolic equations, Springer, 1998, pp. 325–432.

[29] A. Burbeau, P. Sagaut, A dynamic p-adaptive Discontinuous Galerkin method for viscous flow with shocks, Computers & Fluids 34 (2005) 401–417.

[30] L. Wang, D. J. Mavriplis, Adjoint-based h-p adaptive discontinuous Galerkin methods for the 2D compressible Euler equations, Journal of Computational Physics 228 (2009) 7643–7661.

[31] D. Kuzmin, A vertex-based hierarchical slope limiter for p-adaptive discontinuous Galerkin methods, Journal of Computational and Applied Mathematics 233 (2010) 3077–3085.

[32] X. Zhang, C. W. Shu, On maximum-principle-satisfying high order schemes for scalar conservation laws, Journal of Computational Physics 229 (2010) 3091–3120.

[33] E. Celledoni, A. Marthinsen, B. Owren, Commutator-free Lie group methods, Future Generation Computer Systems 19 (2003) 341–352.

[34] X. Zhang, C.-W. Shu, Maximum-principle-satisfying and positivity-preserving high-order schemes for conservation laws: survey and new developments, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 467 (2011) 2752–2776.

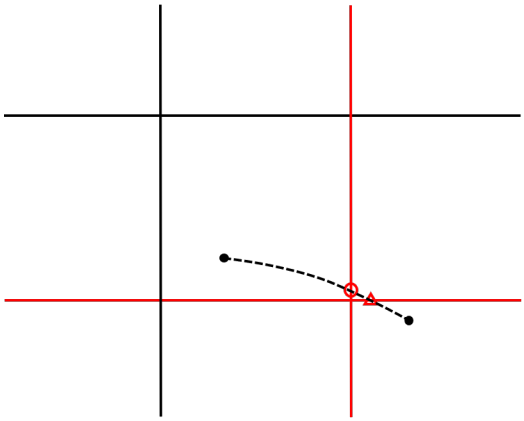
[35] N. Zheng, X. Cai, J.-M. Qiu, J. Qiu, A Conservative Semi-Lagrangian Hybrid Hermite WENO Scheme for Linear Transport Equations and the Nonlinear Vlasov–Poisson System, SIAM Journal on Scientific Computing 43 (2021) A3580–A3606.

[36] X. Cai, S. Boscarino, J.-M. Qiu, High order semi-lagrangian discontinuous Galerkin method coupled with Runge-Kutta exponential integrators for nonlinear Vlasov dynamics, Journal of Computational Physics 427 (2021) 110036.

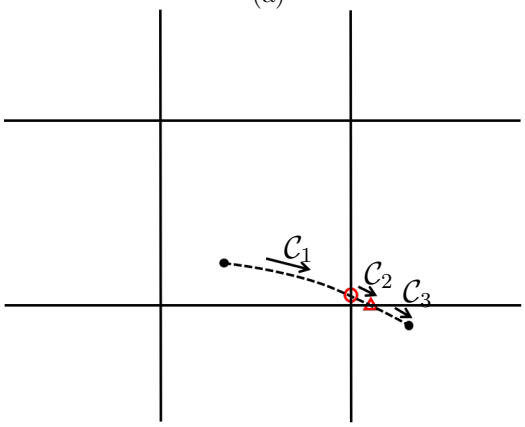
[37] F. Filbet, E. Sonnendrücker, Comparison of Eulerian Vlasov solvers, Computer Physics Communications 150 (2003) 247–266.

[38] T. Arber, R. Vann, A Critical Comparison of Eulerian-Grid-Based Vlasov Solvers, Journal of Computational Physics 180 (2002) 339–357.

[39] M. M. Shoucri, A twolevel implicit scheme for the numerical solution of the linearized vorticity equation, International Journal for Numerical Methods in Engineering 17 (2010) 1525–1538.



(a)



(b)

Figure B.23: Left: the red straight lines represent the mesh lines intersecting $\mathcal{C}$; the red circle represents a type-1 intersection point; the red triangle represents a type-2 intersection point. Right: the black arrows represent the directions of the outer segments.

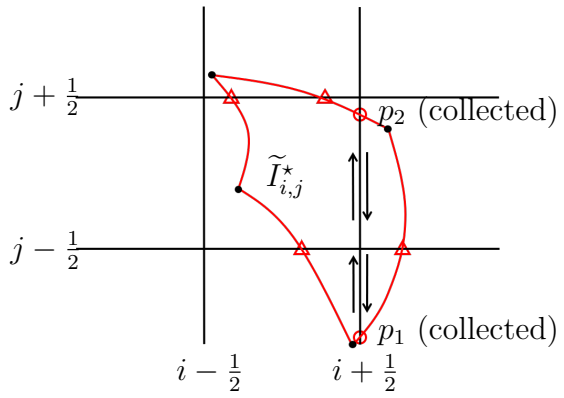


Figure B.24: Schematic illustration for determining inner integral segments.