Journal of Computational Physics 229 (2010) 1130–1149

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

A conservative high order semi-Lagrangian WENO method for the Vlasov equation [☆]

Jing-Mei Qiu [a,*], Andrew Christlieb [b]

[a] Department of Mathematical and Computer Science, Colorado School of Mines, Golden, CO 80401, United States
[b] Department of Mathematics, Michigan State University, East Lansing, MI 48824, United States

ARTICLE INFO

Article history:
Received 24 December 2008
Received in revised form 3 September 2009
Accepted 10 October 2009
Available online 22 October 2009

Keywords:
Conservative scheme
Semi-Lagrangian methods
WENO reconstruction
Operator splitting methods
Vlasov equation
Landau damping
Two-stream instability

ABSTRACT

In this paper, we propose a novel Vlasov solver based on a semi-Lagrangian method which combines Strang splitting in time with high order WENO (weighted essentially non-oscillatory) reconstruction in space. A key insight in this work is that the spatial interpolation matrices, used in the reconstruction process of a semi-Lagrangian approach to linear hyperbolic equations, can be factored into right and left flux matrices. It is the factoring of the interpolation matrices which makes it possible to apply the WENO methodology in the reconstruction used in the semi-Lagrangian update. The spatial WENO reconstruction developed for this method is conservative and updates point values of the solution. While the third, fifth, seventh and ninth order reconstructions are presented in this paper, the scheme can be extended to arbitrarily high order. WENO reconstruction is able to achieve high order accuracy in smooth parts of the solution while being able to capture sharp interfaces without introducing oscillations. Moreover, the CFL time step restriction of a regular finite difference or finite volume WENO scheme is removed in a semi-Lagrangian framework, allowing for a cheaper and more flexible numerical realization. The quality of the proposed method is demonstrated by applying the approach to basic test problems, such as linear advection and rigid body rotation, and to classical plasma problems, such as Landau damping and the two-stream instability. Even though the method is only second order accurate in time, our numerical results suggest the use of high order reconstruction is advantageous when considering the Vlasov–Poisson system.

1. Introduction

For collisionless plasmas where the average particle speed is much less than the speed of light, the plasma may be described by the well-known Vlasov–Poisson (VP) system,

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla_{\mathbf{x}} f + \mathbf{E}(t,\mathbf{x}) \cdot \nabla_{\mathbf{v}} f = 0, \tag{1.1}$$

and

$$\mathbf{E}(t,\mathbf{x}) = -\nabla_{\mathbf{x}} \phi(t,\mathbf{x}), \qquad -\triangle_{\mathbf{x}} \phi(t,\mathbf{x}) = \boldsymbol{\rho}(t,\mathbf{x}), \tag{1.2}$$

[☆] Research supported by the Air Force Office of Scientific Research and the Air Force Research Labs (Edward's and Kirtland). Grand Numbers FA9550-07-1-0092 and FA9550-07-1-0144.

[*] Corresponding author.
E-mail addresses: jingqiu@mines.edu (J.-M. Qiu), Christlieb@math.msu.edu (A. Christlieb).

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.10.016

where $\mathbf{x}$ and $\mathbf{v}$ are coordinates in phase space $(\mathbf{x},\mathbf{v}) \in \mathbb{R}^3 \times \mathbb{R}^3$, $\mathbf{E}$ is the electric field, ϕ is the self-consistent electrostatic potential and $f(t,\mathbf{x},\mathbf{v})$ is the probability distribution function which describes the probability of finding a particle with velocity $\mathbf{v}$ at position $\mathbf{x}$ at time t. The probability distribution function couples to the long range fields via the charge density, $\rho(t,\mathbf{x}) = \int_{\mathbb{R}^3} f(t,\mathbf{x},\mathbf{v})d\mathbf{v} - 1$, where we take the limit of uniformly distributed infinitely massive ions in the background. Eqs. (1.1) and (1.2) have been nondimensionalized so that all physical constants are one. When self-consistent magnetic fields need to be considered, the system must be extended to the Vlasov–Maxwell (VM) system. However, in this paper we focus our attention on numerically solving the Vlasov equation.

The Lagrangian particle methods, e.g. the classical Particle-In-Cell (PIC) method [4,20,29,28,21], Particle–Particle–Particle–Mesh (P3M) method [20,1], Complex Particle Kinetics (CPK) method [14,18,19], etc., are based on tracing the motion of a finite number of macro-particles (test particles). The trajectories of particles are traced by solving Newton's equations of motions with self-consistent fields. The key task in developing an efficient Lagrangian method is reducing the field evaluation from $\mathcal{O}(N^2)$ to a more manageable operation count, where N is the number of test particles in the simulation. In PIC and P3M a fixed volumetric mesh is used to compute the fields, and in CPK an inherent assumption of neutrality is built into the particle model. An alternative grid-free approach for efficiently solving for the self-consistent fields is to make use of fast summation algorithms [3,17,23], as in the Boundary Integral/Treecode (BIT) method [11,10,12,24], which solves for the fields in $\mathcal{O}(N\log N)$. The Lagrangian particle methods are very popular in plasma simulations because: (1) they are flexible and easy to implement, (2) it is easy to couple Lagrangian methods with Monte Carlo techniques to approximate the collisional term in the Boltzmann equation, (3) the computational cost is reasonable, especially for high dimensional problems. However, traditional Lagrangian particle methods, such as PIC, P3M and CPK, typically use a statistical approach in their representation of phase space, and as such, the numerical noise in these systems is proportional to $\mathcal{O}(1/\sqrt{N})$. Within the BIT methodology there have been attempts to overcome this by using a systematic collocation over the initial phase space, combined with dynamic point insertion in time, so as to maintain phase space accuracy [12].

On the other hand, Eulerian approaches are based on a fixed numerical mesh. Compared to a Lagrangian approach, a drawback of the grid-based method is its significant increase in computational cost as the problem dimension increases. However, these methods have the advantage of high order accuracy, which leads to a reduction in computational cost when compared with a low order method. The gain in efficiency is because the high order method, on a coarse mesh, can often attain similar or better results than the low order method on a fine mesh. One of the most accurate mesh-based numerical methods applied to the VP system is the non-split high order conservative finite difference WENO method, presented in Refs. [22,2]. The method, which has successfully simulated the Landau damping phenomenon with machine precision [33], is high order in phase space and time, and could resolve sharp interfaces without oscillations by adaptively assigning weights to neighboring reconstruction stencils. Other well-developed methods for the VP system include finite element [32,31] as well as spectral methods. The bottleneck of all of these methods, when applied to the VP system, is the CFL time step restriction.

The Strang splitting semi-Lagrangian methods are one of the most popular algorithms among Eulerian approaches for the VP system. Various implementations have been successfully developed in the Strang splitting semi-Lagrangian framework [8,27,16,30,7,6,25,9]. Despite its success and popularity, a number of deficiencies occur in previous implementations. The semi-Lagrangian method, which evolves point values in Ref. [8], is not written in a conservative form. Because of this, an ad hoc filtering procedure is necessary to remove oscillations around shocks. This filtering procedure has the additional drawback of introducing artificial numerical diffusion. The semi-Lagrangian method, that evolves integrated mass, described in Refs. [27,16,30], is at best second order accurate in space, because the shearing velocity varies with order one within each individual mesh cell. In [7], pointwise WENO (PWENO) interpolations are applied to obtain a semi-Lagrangian and flux balance methods for the dimensional split Vlasov equation. The algorithms developed there appear to be very close to the semi-Lagrangian WENO method proposed in this paper. However, there is an essential difference. Specifically, the PWENO interpolation [7] that evolves point values is not written in a conservative form, and hence it can be shown that the PWENO method is not conservative. The WENO interpolation for the conservative flux balance methods [7] evolves the integrated mass, rather than the point values. As a result, the method is at best second order accurate for multi-dimensional problems. The work in Refs. [25,9] is also only second order accurate in space, because of the interpolation schemes used in the evolution of phase space density.

In this paper, we propose a novel Vlasov solver based on Strang splitting in time and a conservative high order semi-Lagrangian method with WENO reconstruction [22,2] in space. Strang splitting is used to dimensionally split the high dimensional nonlinear Vlasov equation into a sequence of one-dimensional linear hyperbolic problems. The key insight, made in this paper, is that the spatial interpolation matrices, used in the reconstruction process of a semi-Lagrangian approach, can be factored into symmetric right and left flux matrices. As demonstrated in Section 3 of the paper, it is the factoring of the interpolation matrices which makes it possible to apply the WENO methodology in the reconstruction used in the semi-Lagrangian update. For each decoupled one-dimensional hyperbolic equation, the proposed WENO reconstruction is a pointwise conservative semi-Lagrangian algorithm. The fact that it is the point value, instead of integrated mass, that is being evolved, allows a direct extension of the proposed scheme to higher than second order accuracy in space for multi-dimensional problems.

The paper is organized as follows. Section 2 is a brief review of the Strang splitting semi-Lagrangian framework. A conservative form of the semi-Lagrangian method, that evolves point values, is rewritten in Section 3 with coefficients presented for the third, fifth, seventh and ninth order schemes. Section 3 also contains the high order WENO reconstruction procedure, which prevents oscillations around discontinuities in phase space. In Section 4, we present our numerical results for basic test problems, such as linear advection and rigid body rotation, as well as the results from application of the method to classical plasma problems, such as Landau damping and the two-stream instability. In particular, the application to the two-stream instability makes it clear how high order can provide the desired accuracy for a bulk flow on a very coarse mesh. Section 5 provides a summary of the results and conclusions.

2. Strang splitting semi-Lagrangian method for the Vlasov equation

In this section, we briefly review the Strang splitting semi-Lagrangian method for the VP system, originally proposed in Ref. [8]. The Strang splitting method is also known as the 'fractional step method' in the literature. In this paper, we consider the Vlasov equation, Eq. (1.1), with only one position and one velocity axis, i.e. $(x, v) \in \mathbb{R} \times \mathbb{R}$. However, extension to higher dimensions in **x** and **v** is straightforword. The time splitting form of Eq. (1.1) is,

$$\frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} = 0, \tag{2.1}$$

$$\frac{\partial f}{\partial t} + E(t,x)\frac{\partial f}{\partial v} = 0. \tag{2.2}$$

The split form of Eq. (1.1) can be made second order accurate in time by solving Eq. (2.1) for a half time step, then solving Eq. (2.2) for a full time step, followed by solving Eq. (2.1) for a second half time step. The observation that both Eqs. (2.1) and (2.2) are linear hyperbolic equations allows for a simple implementation of the semi-Lagrangian method based on shifting in each direction. Specifically, the numerical update from $f^n(x, v)$ (the solution at $t^n = n\Delta t$) to $f^{n+1}(x, v)$ is as follows:

1. advance a half time step for Eq. (2.1) by shifting in the x-direction,

$$f^*(x, v) = f^n\left(x - \frac{\Delta t}{2}v, v\right), \tag{2.3}$$

2. compute the electric field at the half step by substituting f^* into Eq. (1.2) and solve for $E^*(x)$,
3. advance a full time step for Eq. (2.2) by shifting in the v-direction,

$$f^{**}(x, v) = f^*(x, v - \Delta t E^*(x)), \tag{2.4}$$

4. advance a half time step for Eq. (2.1) by shifting in the x-direction,

$$f^{n+1}(x, v) = f^{**}\left(x - \frac{\Delta t}{2}v, v\right). \tag{2.5}$$

In general, the function $f^n(x, v)$ (or $f^*(x, v)$,$f^{**}(x, v)$) is only available at grid points. Hence, an interpolation algorithm is needed in order to recover the function values at the foot of the characteristics in the shifting process. Indeed, the interpolation algorithms, discussed in detail in the next section, are the key to the quality of the semi-Lagrangian scheme.

Remark 2.1. The fact that the semi-Lagrangian method is not subject to a CFL time step restriction, which is what allows for an extra large numerical time step, is its main advantage over the traditional finite difference method [33]. On the other hand, its convergence rate is at best second order in time.

3. Conservative representation of semi-Lagrangian scheme and WENO reconstruction

In this section, we focus our discussion on solving a one-dimensional linear hyperbolic equation of the form

$$f_t + vf_x = 0, \quad f(x, t = 0) = f_0(x), \quad \text{on } [a, b], \tag{3.1}$$

by a semi-Lagrangian method under the assumption that v is a constant. The method can be used to solve Eqs. (2.1) and (2.2). Hence, this method is well suited for solving the Vlasov equation, when it is cast in the Strang splitting framework.

In this section, we adopt the following notation for the numerical discretization. The domain $[a, b]$ is discretized as

$$a = x_0 < x_1 \cdots < x_N = b. \tag{3.2}$$

We assume uniformly distributed grid points, i.e. $x_i = a + i \cdot \Delta x$ with mesh size $\Delta x = (b - a)/N$. We chose $x_{i+\frac{1}{2}}$ to be the midpoint, $x_{i+\frac{1}{2}} = (x_i + x_{i+1})/2$, and define cells about x_i and the midpoint $x_{i+\frac{1}{2}}$ as $I_i = \left[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\right]$ and $I_{i+\frac{1}{2}} = [x_i, x_{i+1}]$. We further define $xshift$ to be $xshift = v\frac{\Delta t}{\Delta x}$, and let f_i^n denote the point values of the numerical solution at the mesh point $x = x_i$ and time $t = t^n := n\Delta t$.

In Section 3.1, the conservative semi-Lagrangian method with a fixed stencil reconstruction is discussed, while the method with WENO reconstruction, which assures non-oscillatory capture of sharp interfaces, is discussed in Section 3.2. The WENO reconstruction presented in Section 3.2 is developed for the flux-based conservative form of Section 3.1, hence it is important to have a good understanding of the conservative formulation before diving into the WENO method presented in Section 3.2.

3.1. Conservative representation and fixed stencil interpolation

3.1.1. A third order interpolation

In this section, a conservative semi-Lagrangian scheme based on third order fixed stencil interpolation is presented. There are three cases of *xshift* which are discussed in turn: shift to the right by some amount less than half a cell $(xshift \in [0, \frac{1}{2}])$, shift to the left by some amount less than half a cell $(xshift \in [-\frac{1}{2}, 0])$ and shift a distance greater than half a cell $(|xshift| > \frac{1}{2})$. The procedures presented here are the same whether they are in compact non-conservative form, i.e. not expressed as a difference of fluxes, or in conservative form, i.e. expressed as a difference of fluxes. The conservative form allows for the application of WENO reconstruction to the semi-Lagrangian method and is presented in Section 3.2.

Hereafter, we will adopt a matrix notation for presentation of the interpolation scheme. The matrix A will denote the interpolation matrix. We use $A(i,j)$ to denote the element at the ith row and jth column, $A(i,:)$ to denote the ith row of A, and $A(:,j)$ to denote the jth column of A. Matrices B and C in steps three and four are related to the conservative form of the procedure, with matrix C being essential in the construction of the conservative form.

We first consider $xshift \in [0, \frac{1}{2}]$. The algorithm is as below.

1. To compute the solution at t^{n+1}, we start with a reconstruction of the underlying function at t^n using a piecewise cubic approximation $\tilde{f}^n(x)$. Its projection on cell $I_{i-\frac{1}{2}}$ is $\tilde{f}^n_{i-\frac{1}{2}}(\xi)$, which is reconstructed from the stencil $\{f^n_{i-2}, f^n_{i-1}, f^n_i, f^n_{i+1}\}$ and is

$$\tilde{f}^n_{i-\frac{1}{2}}(\xi) = f^n_i + \left(-\frac{1}{6}f^n_{i-2} + f^n_{i-1} - \frac{1}{2}f^n_i - \frac{1}{3}f^n_{i+1}\right)\xi + \left(\frac{1}{2}f^n_{i-1} - f^n_i + \frac{1}{2}f^n_{i+1}\right)\xi^2 + \left(\frac{1}{6}f^n_{i-2} - \frac{1}{2}f^n_{i-1} + \frac{1}{2}f^n_i - \frac{1}{6}f^n_{i+1}\right)\xi^3, \tag{3.3}$$

where $\xi(x) = \frac{x-x_i}{x_{i-1}-x_i} \in [0,1],\ x \in I_{i-\frac{1}{2}}$. Eq. (3.3) can be written in matrix form as,

$$\tilde{f}^n_{i-\frac{1}{2}}(\xi) = \left(f^n_{i-2}, f^n_{i-1}, f^n_i, f^n_{i+1}\right) \cdot A^L_3 \cdot (1, \xi, \xi^2, \xi^3)', \tag{3.4}$$

with matrix

$$A^L_3 = \begin{pmatrix} 0 & -\frac{1}{6} & 0 & \frac{1}{6} \\ 0 & 1 & \frac{1}{2} & -\frac{1}{2} \\ 1 & -\frac{1}{2} & -1 & \frac{1}{2} \\ 0 & -\frac{1}{3} & \frac{1}{2} & -\frac{1}{6} \end{pmatrix}. \tag{3.5}$$

2. An update for the function at the point value x_i at time level t^{n+1} (i.e. f^{n+1}_i) can be obtained by tracing the characteristic back to time $t = t^n$ and evaluating the interpolant $\tilde{f}^n_{i-\frac{1}{2}}(\xi)$ at location $\xi = \xi_0$. We observe that the update for $f^{n+1}_i = \tilde{f}^n_{i-\frac{1}{2}}(\xi_0)$ can be written in a conservative form,

$$f^{n+1}_i = f^n_i - \xi_0\left(\left(f^n_{i-2}, f^n_{i-1}, f^n_i, f^n_{i+1}\right) \cdot B^L_3 \cdot \left(1, \xi_0, \xi_0^2\right)'\right) \tag{3.6}$$

$$= f^n_i - \xi_0\left(\left(f^n_{i-2}, f^n_{i-1}, f^n_i, f^n_{i+1}\right) \cdot \left(\begin{pmatrix} \vec{0} \\ C^L_3 \end{pmatrix} - \begin{pmatrix} C^L_3 \\ \vec{0} \end{pmatrix}\right) \cdot \left(1, \xi_0, \xi_0^2\right)'\right) \tag{3.7}$$

$$= f^n_i - \xi_0\left(\left(f^n_{i-1}, f^n_i, f^n_{i+1}\right) \cdot C^L_3 - \left(f^n_{i-2}, f^n_{i-1}, f^n_i\right) \cdot C^L_3\right) \cdot \left(1, \xi_0, \xi_0^2\right)',$$

where

$$B^L_3 = \begin{pmatrix} \frac{1}{6} & 0 & -\frac{1}{6} \\ -1 & -\frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & 1 & -\frac{1}{2} \\ \frac{1}{3} & -\frac{1}{2} & \frac{1}{6} \end{pmatrix}, \qquad C^L_3 = \begin{pmatrix} -\frac{1}{6} & 0 & \frac{1}{6} \\ \frac{5}{6} & \frac{1}{2} & -\frac{1}{3} \\ \frac{1}{3} & -\frac{1}{2} & \frac{1}{6} \end{pmatrix}. \tag{3.8}$$

3. Finally, we define the flux function,

$$\hat{f}^n_{i-\frac{1}{2}}(\xi) = \left(f^n_{i-2}, f^n_{i-1}, f^n_i\right) \cdot C^L_3 \cdot (1, \xi, \xi^2)', \tag{3.9}$$

so that Eq. (3.6) can be written as

$$f^{n+1}_i = f^n_i - \xi_0\left(\hat{f}^n_{i+\frac{1}{2}}(\xi_0) - \hat{f}^n_{i-\frac{1}{2}}(\xi_0)\right). \tag{3.10}$$

Proposition 3.1. *The scheme in Eq. (3.10) conserves the total mass if periodic boundary conditions are imposed.*

Proof

$$\sum_i f_i^{n+1} = \sum_i \left(f_i^n - \xi_0\left(\hat{f}^n_{i+\frac{1}{2}}(\xi_0) - \hat{f}^n_{i-\frac{1}{2}}(\xi_0)\right)\right) = \sum_i f_i^n - \xi_0 \sum_i \left(\hat{f}^n_{i+\frac{1}{2}}(\xi_0) - \hat{f}^n_{i-\frac{1}{2}}(\xi_0)\right) = \sum_i f_i^n. \qquad \square$$

As to the case when $xshift \in [-\frac{1}{2}, 0)$, the scheme can be written in a conservative form in a fashion similar to the case of $xshift \in [0, \frac{1}{2}]$. Specifically,

$$f_i^{n+1} = f_i^n + \xi_0\left(\hat{f}^n_{i+\frac{1}{2}}(\xi_0) - \hat{f}^n_{i-\frac{1}{2}}(\xi_0)\right), \tag{3.11}$$

where the flux function

$$\hat{f}^n_{i-\frac{1}{2}}(\xi) = \left(f^n_{i-1}, f^n_i, f^n_{i+1}\right) \cdot C_3^R \cdot (1, \xi, \xi^2)'. \tag{3.12}$$

C_3^R is a 3×3 matrix with $C_3^R(i,j) = C_3^L(4-i,j)$. Notice the change in sign in Eq. (3.11) from (3.10). In the case when $|xshift| > \frac{1}{2}$, the left/right shifting exceeds half a single cell, and whole grid shifting is carried out, followed by a final update based on the above procedure for $xshift \in [-\frac{1}{2}, \frac{1}{2}]$.

We remark that Eq. (3.11) is also conservative if periodic boundary conditions are imposed. The conservative form allows for the application of WENO reconstructions on flux functions, while preserving the conservation property.

Notice that the existence of a matrix C_3^L in Eq. (3.8) is essential in the design of the conservative version of the scheme. Proposition 3.2 below assures such existence. Further, the proposition can be easily extended to the scheme with other high order reconstructions.

Proposition 3.2. *There exists a matrix C_3^L in Eq. (3.8), such that the matrix B_3^L in Eq. (3.6) can be split as*

$$\begin{pmatrix} \vec{0} \\ C_3^L \end{pmatrix} - \begin{pmatrix} C_3^L \\ \vec{0} \end{pmatrix}.$$

Proof. Let $B_3^L(i,:) = \vec{b_i}$ and $C_3^L(i,:) = \vec{c_i}$. If there exists a matrix C_3^L s.t.

$$B_3^L = \begin{pmatrix} \vec{0} \\ C_3^L \end{pmatrix} - \begin{pmatrix} C_3^L \\ \vec{0} \end{pmatrix}, \tag{3.13}$$

then

$$\begin{aligned} \vec{c_1} &= -\vec{b_1}, \\ \vec{c_2} &= -\vec{b_1} - \vec{b_2}, \\ \vec{c_3} &= -\vec{b_1} - \vec{b_2} - \vec{b_3} = \vec{b_4}. \end{aligned} \tag{3.14}$$

From Eq. (3.14), existence of the matrix C_3^L can be guaranteed if

$$\sum_{i=1}^{4} \vec{b_i} = \vec{0}. \tag{3.15}$$

By choosing $\vec{f} = (f^n_{i-2}, f^n_{i-1}, f^n_i, f^n_{i+1}) = (1,1,1,1)$, we see that there exists at least one such matrix C_3^L which splits B_3^L as proposed. We note that the same splitting is true for matrix B_3^R. $\square$

Remark 3.3. The scheme described above is obtained through a high order reconstruction, and then shifting procedure. The algorithm can be interpreted from another point of view, as a high order Taylor expansion in the time direction as follows. From Eq. (3.1), we have

$$\frac{\partial f^l}{\partial^l t} = (-v)^l \frac{\partial f^l}{\partial x^l}, \quad l = 1, 2, 3, \ldots. \tag{3.16}$$

Taylor expanding of $f(x = x_i, t)$ about $t = t_n$ gives,

$$f(x = x_i, t) = f_i^n + \sum_{i=1}^{\infty} \frac{1}{i!} \frac{\partial^i f}{\partial t^i}(t - t_n)^i = f_i^n + \sum_{i=1}^{\infty} \frac{(-v)^i}{i!} \frac{\partial^i f}{\partial x^i}(t - t_n)^i.$$

In particular, we may express f_i^{n+1}as,

$$f_i^{n+1} = f_i^n + \sum_{i=1}^{\infty} \frac{(-v)^i}{i!} \frac{\partial^i f}{\partial x^i} \Delta t^i, \tag{3.17}$$

where $\frac{\partial^j f}{\partial x^j}$ is approximated by taking derivatives of Eq. (3.3). This formulation is equivalent to the semi-Lagrangian formulation with a restricted time step.

Remark 3.4. For the linear hyperbolic equation (3.1), the standard finite difference update [22] is equivalent to Taylor expanding $f(x = x_i, t_{n+1})$ about $t = t_n$ with a truncation error of $\mathcal{O}(\Delta t^2)$,

$$f(x = x_i, t_{n+1}) = f_i^n - \Delta t \cdot v \cdot \frac{\partial f}{\partial x} + \mathcal{O}(\Delta t^2). \tag{3.18}$$

Again $\frac{\partial f}{\partial x}$ is approximated by taking the derivative of the Lagrangian interpolant of *f*, where the interpolant is given by Eq. (3.3).

3.1.2. General high order interpolation: fifth, seventh and ninth order

In this subsection, we provide the fifth, seventh and ninth order conservative semi-Lagrangian formulation, following a similar approach to the third order one. The provided coefficients can be used to code up the algorithm directly. Its extension to higher than ninth order interpolation is straight forward. As usual, let $\xi_0 = |xshift|$. Our discussion focuses on the case of $xshift \in [-\frac{1}{2}, \frac{1}{2}]$, since the case of $|xshift| > \frac{1}{2}$ will be handled with a whole grid shift followed by the case of $xshift \in [-\frac{1}{2}, \frac{1}{2}]$ to account for the fractional remainder.

As for the case of $xshift \in [0, \frac{1}{2}]$, the high order conservative semi-Lagrangian method is,

$$f_i^{n+1} = f_i^n - \xi_0 \left(\hat{f}^n_{i+\frac{1}{2}}(\xi_0) - \hat{f}^n_{i-\frac{1}{2}}(\xi_0) \right), \tag{3.19}$$

where the flux functions are respectively defined, for fifth, seventh and ninth order interpolation by,

$$\hat{f}^n_{i-\frac{1}{2}}(\xi) = (f^n_{i-3}, \ldots, f^n_{i+1}) \cdot C_5^L \cdot (1, \ldots, \xi^4)', \tag{3.20a}$$

$$\hat{f}^n_{i-\frac{1}{2}}(\xi) = (f^n_{i-4}, \ldots, f^n_{i+2}) \cdot C_7^L \cdot (1, \xi, \ldots, \xi^6)', \tag{3.20b}$$

$$\hat{f}^n_{i-\frac{1}{2}}(\xi) = (f^n_{i-5}, \ldots, f^n_{i+3}) \cdot C_9^L \cdot (1, \xi, \ldots, \xi^8)', \tag{3.20c}$$

with

$$C_5^L = \begin{bmatrix} \frac{1}{30} & 0 & -\frac{1}{24} & 0 & \frac{1}{120} \\ -\frac{13}{60} & -\frac{1}{24} & \frac{1}{4} & \frac{1}{24} & -\frac{1}{30} \\ \frac{47}{60} & \frac{5}{8} & -\frac{1}{3} & -\frac{1}{8} & \frac{1}{20} \\ \frac{9}{20} & -\frac{5}{8} & \frac{1}{12} & \frac{1}{8} & -\frac{1}{30} \\ -\frac{1}{20} & \frac{1}{24} & \frac{1}{24} & -\frac{1}{24} & \frac{1}{120} \end{bmatrix}, \tag{3.21}$$

$$C_7^L = \begin{bmatrix} -\frac{1}{140} & 0 & \frac{7}{720} & 0 & -\frac{1}{360} & 0 & \frac{1}{5040} \\ \frac{5}{84} & \frac{1}{180} & -\frac{19}{240} & -\frac{1}{144} & \frac{1}{48} & \frac{1}{720} & -\frac{1}{840} \\ -\frac{101}{420} & -\frac{5}{72} & \frac{7}{24} & \frac{11}{144} & -\frac{13}{240} & -\frac{1}{144} & \frac{1}{336} \\ \frac{319}{420} & \frac{49}{72} & -\frac{23}{72} & -\frac{7}{36} & \frac{23}{360} & \frac{1}{72} & -\frac{1}{252} \\ \frac{107}{210} & -\frac{49}{72} & \frac{1}{48} & \frac{7}{36} & -\frac{1}{30} & -\frac{1}{72} & \frac{1}{336} \\ -\frac{19}{210} & \frac{5}{72} & \frac{7}{80} & -\frac{11}{144} & \frac{1}{240} & \frac{1}{144} & -\frac{1}{840} \\ \frac{1}{105} & -\frac{1}{180} & -\frac{1}{90} & \frac{1}{144} & \frac{1}{720} & -\frac{1}{720} & \frac{1}{5040} \end{bmatrix} \tag{3.22}$$

and

$$C_9^L = \begin{bmatrix} \frac{1}{630} & 0 & -\frac{41}{18144} & 0 & \frac{13}{17280} & 0 & -\frac{1}{12096} & 0 & \frac{1}{362880} \\ -\frac{41}{2520} & -\frac{1}{1120} & \frac{2081}{90720} & \frac{7}{5760} & -\frac{1}{135} & -\frac{1}{2880} & \frac{23}{30240} & \frac{1}{40320} & -\frac{1}{45360} \\ \frac{199}{2520} & \frac{17}{1440} & -\frac{281}{2592} & -\frac{89}{5760} & \frac{139}{4320} & \frac{11}{2880} & -\frac{17}{6048} & -\frac{1}{5760} & \frac{1}{12960} \\ -\frac{641}{2520} & -\frac{127}{1440} & \frac{4097}{12960} & \frac{587}{5760} & -\frac{29}{432} & -\frac{41}{2880} & \frac{167}{30240} & \frac{1}{1920} & -\frac{1}{6480} \\ \frac{1879}{2520} & \frac{205}{288} & -\frac{797}{2592} & -\frac{91}{384} & \frac{587}{8640} & \frac{5}{192} & -\frac{19}{3024} & -\frac{1}{1152} & \frac{1}{5184} \\ \frac{275}{504} & -\frac{205}{288} & -\frac{59}{2592} & \frac{91}{384} & -\frac{29}{1080} & -\frac{5}{192} & \frac{25}{6048} & \frac{1}{1152} & -\frac{1}{6480} \\ -\frac{61}{504} & \frac{127}{1440} & \frac{1637}{12960} & -\frac{587}{5760} & -\frac{17}{4320} & \frac{41}{2880} & -\frac{43}{30240} & -\frac{1}{1920} & \frac{1}{12960} \\ \frac{11}{504} & -\frac{17}{1440} & -\frac{491}{18144} & \frac{89}{5760} & \frac{11}{2160} & -\frac{11}{2880} & \frac{1}{6048} & \frac{1}{5760} & -\frac{1}{45360} \\ -\frac{1}{504} & \frac{1}{1120} & \frac{59}{22680} & -\frac{7}{5760} & -\frac{11}{17280} & \frac{1}{2880} & \frac{1}{60480} & -\frac{1}{40320} & \frac{1}{362880} \end{bmatrix}. \tag{3.23}$$

As to the case of $xshift \in [-\frac{1}{2}, 0)$, there is a similar formulation. The high order semi-Lagrangian scheme can be written in a conservative form as

$$f_i^{n+1} = f_i^n + \xi_0\left(\hat{f}^n_{i+\frac{1}{2}}(\xi_0) - \hat{f}^n_{i-\frac{1}{2}}(\xi_0)\right), \tag{3.24}$$

where the flux functions are respectively defined, for fifth, seventh and ninth order interpolation by,

$$\hat{f}^n_{i-\frac{1}{2}}(\xi) = \left(f^n_{i-2}, \ldots, f^n_{i+2}\right) \cdot C_5^R \cdot (1, \ldots, \xi^4)', \tag{3.25a}$$

$$\hat{f}^n_{i-\frac{1}{2}}(\xi) = \left(f^n_{i-3}, \ldots, f^n_{i+3}\right) \cdot C_7^R \cdot (1, \xi, \ldots, \xi^6)', \tag{3.25b}$$

$$\hat{f}^n_{i-\frac{1}{2}}(\xi) = \left(f^n_{i-4}, \ldots, f^n_{i+4}\right) \cdot C_9^R \cdot (1, \xi, \ldots, \xi^8)'. \tag{3.25c}$$

C_p^R is a $p \times p$ matrix with its element given by

$$C_p^R(i,j) = C_p^L(p+1-i,j), \quad p = 5, 7, 9. \tag{3.26}$$

Remark 3.5. By a similar argument to that given in Proposition 3.2, the existence of the matrix C_p^L and C_p^R, for $p = 5, 7, 9$, is guaranteed. Hence, the semi-Lagrangian scheme can be written in a conservative form for fifth, seventh and ninth order. Likewise, this can be extended to the cases of $p = 11, 13, 15, \ldots$

Remark 3.6. The proposed conservative form of the semi-Lagrangian scheme for Eq. (3.1) evolves point values, instead of the integrated mass. Because of this, the Strang splitting formulation allows for a high order extension in space (higher than second order in space) for multi-dimensional problems.

3.2. WENO reconstruction in semi-Lagrangian framework

In general, high order fixed stencil reconstruction of numerical fluxes performs well when the solution is smooth. However, around discontinuities oscillations will be introduced. In this subsection, a nonlinear WENO procedure is introduced for reconstructing $\hat{f}^n_{i-\frac{1}{2}}(\xi)$. By adaptively assigning weights to neighboring candidate stencils, the WENO nonlinear reconstruction preserves high order accuracy of the linear scheme around smooth regions of the solution, while producing a sharp and essentially non-oscillatory capture of discontinuities. The WENO reconstruction could also be understood as a black box filtering procedure, based on the fluxes generated from fixed stencil interpolation.

The original third and fifth order WENO reconstructions, in the finite difference and finite volume framework, were introduced in [22]. The reconstruction procedures are further extended to seventh, ninth and eleventh order in [2]. In this subsection, we give the procedure for embedding the original WENO reconstruction into the proposed conservative semi-Lagrangian framework, along with a third order WENO reconstruction as an example. For other order of reconstructions (fifth, seventh and ninth), which are used in our numerical simulations, we only provide the formulations and suggest that the interested reader look at Refs. [22,2] for the details.

3.2.1. Third order WENO reconstruction

In the following, we illustrate the WENO mechanism through a third order example, following Section 3.1.1. As mentioned in Remark 3.4, the standard finite difference scheme is very closely related to our proposed semi-Lagrangian scheme, when solving a linear hyperbolic equation. Based on this observation, we apply WENO reconstructions to numerical fluxes, e.g. $\hat{f}^n_{i-\frac{1}{2}}(\xi)$ in (3.9), in the third order conservative semi-Lagrangian scheme.

We only discuss the construction of the flux function $\hat{f}^n_{i-\frac{1}{2}}$ when *xshift* $\in \left[0, \frac{1}{2}\right]$. When *xshift* $\in [-\frac{1}{2}, 0]$, the numerical flux could be modified symmetrically with respect to x_i in the WENO fashion. From Eq. (3.9), the point values $\{f^n_{i-2}, f^n_{i-1}, f^n_i\}$ are used to construct the flux function $\hat{f}^n_{i-\frac{1}{2}}(\xi)$. The flux $\hat{f}^n_{i-\frac{1}{2}}(\xi)$ is composed of the information from two potential stencils

$$S_1 = \{f^n_{i-2}, f^n_{i-1}\} \quad \text{and} \quad S_2 = \{f^n_{i-1}, f^n_i\}. \tag{3.27}$$

Intuitively, in regions where the function is smooth, we want to use information from both stencils S_1 and S_2, to obtain a third order approximation. On the other hand, around a big jump, we only want to use one of the stencils. For example, consider the jump $(f^n_{i-2}, f^n_{i-1}, f^n_i) = (0, 0, 1)$, we only want to use the stencil S_1 to construct the flux function $\hat{f}^n_{i-\frac{1}{2}}(\xi)$, since excluding the stencil S_2 will prevent numerical oscillations. In this work, we will choose to only use the WENO mechanism in adaptively reconstructing the coefficients in front of the constant 1 in the equation for $\hat{f}^n_{i-\frac{1}{2}}$, that is Eq. (3.9) (or the first column of matrix C_3^L in Eq. (3.8)), while leaving coefficients for ξ, ξ^2 unchanged (or the second and the third columns of matrix C_3^L). It is difficult to apply the WENO reconstruction to the coefficients for ξ (or ξ^2), because the linear weights from different sub-stencils are not always positive. We note that there are techniques for treating non-positive weights in the WENO reconstruction [26], adaptation to the method presented in this work is beyond the scope of this text. Also recall that we have $\xi \in \left[-\frac{1}{2}, \frac{1}{2}\right]$ in our scheme, therefore applying WENO reconstruction on the coefficient for constant 1 plays a more important role in suppressing oscillations, as observed in our numerical experiments in Section 4.

The general WENO procedure for constructing $\hat{f}^n_{i-\frac{1}{2}}$ is as follows,

1. Compute the linear weights, γ_1 and γ_2, such that

$$\left(f^n_{i-2},f^n_{i-1},f^n_i\right)\cdot C^L_3(:,1)=\gamma_1\left(f^n_{i-2},f^n_{i-1}\right)\cdot\left(-\frac{1}{2},\frac{3}{2}\right)'+\gamma_2\left(f^n_{i-1},f^n_i\right)\cdot\left(\frac{1}{2},\frac{1}{2}\right)', \tag{3.28}$$

   where $\left(f^n_{i-2},f^n_{i-1}\right)\cdot\left(-\frac{1}{2},\frac{3}{2}\right)'$ and $\left(f^n_{i-1},f^n_i\right)\cdot\left(\frac{1}{2},\frac{1}{2}\right)'$ are second order reconstructions of fluxes from two stencils S_1 and S_2 respectively. From Eq. (3.9), $\gamma_1=\frac{1}{3}$ and $\gamma_2=\frac{2}{3}$.
2. Compute the smoothness indicator β_r for each stencil $S_r,\ r=1,2$. The smoothness indicators β_r are designed such that, if the function is smooth over the stencil S_r, then $\beta_r=\mathcal{O}(\Delta x^2)$, but if the function has a discontinuity, then $\beta_r=\mathcal{O}(1)$. For the third order fluxes, the smoothness indicators are,

$$\beta_1=\left(f^n_{i-1}-f^n_{i-2}\right)^2,\qquad \beta_2=\left(f^n_i-f^n_{i-1}\right)^2. \tag{3.29}$$

3. Compute the nonlinear weights w_1 and w_2. Let

$$\tilde{w}_1=\gamma_1/(\epsilon+\beta_1)^2,\qquad \tilde{w}_2=\gamma_2/(\epsilon+\beta_2)^2, \tag{3.30}$$

   where ϵ is a small number to prevent the denominator from becoming zero. In our numerical tests we take ϵ to be 10^{-6}. The resulting nonlinear weights are renormalized as,

$$w_1=\tilde{w}_1/(\tilde{w}_1+\tilde{w}_2),\qquad w_2=\tilde{w}_2/(\tilde{w}_1+\tilde{w}_2). \tag{3.31}$$

4. Compute numerical fluxes constructed in WENO fashion. Define the matrix $\widetilde{C}^L_3$ as,

$$\widetilde{C}^L_3(:,1)=w_1\cdot\left(-\frac{1}{2},\frac{3}{2},0\right)'+w_2\cdot\left(0,\frac{1}{2},\frac{1}{2}\right)',\quad \widetilde{C}^L_3(:,2)=C^L_3(:,2),\quad \widetilde{C}^L_3(:,3)=C^L_3(:,3). \tag{3.32}$$

   The updated numerical flux is computed using $\widetilde{C}^L_3$, i.e.

$$\hat{f}^n_{i-\frac{1}{2}}(\xi_0)=\left(f^n_{i-2},f^n_{i-1},f^n_i\right)\cdot\widetilde{C}^L_3\cdot\left(1,\xi_0,\xi_0^2\right)'. \tag{3.33}$$

   Repeating the above procedure to obtain the equivalent WENO reconstruction of $\hat{f}^n_{i+\frac{1}{2}}(\xi_0)$, the fluxes in the conservative semi-Lagrangian scheme in Eq. (3.10) are reconstructed using Eq. (3.33) as well as the equivalent WENO reconstruction of $\hat{f}^n_{i+\frac{1}{2}}(\xi_0)$.

3.2.2. General WENO reconstruction procedure

In the following, we provide a general WENO reconstruction procedure for the semi-Lagrangian framework. We again focus our discussion on reconstructing the numerical flux $\hat{f}^n_{i-\frac{1}{2}}$ for the case of *xshift* $\in\left[0,\frac{1}{2}\right]$, since the procedure for $\hat{f}^n_{i+\frac{1}{2}}$ is similar. To reconstruct a numerical flux in WENO fashion, the basic idea is to use an adaptive convex combination of linear fluxes from candidate stencils

$$S_r=\{f^n_{i-k+r-2},\ldots,f^n_{i+r-2}\},\quad r=1,\ldots,k+1, \tag{3.34}$$

for a $(2k+1)$th order scheme. The WENO procedure for $\hat{f}^n_{i-\frac{1}{2}}$, and likewise $\hat{f}^n_{i+\frac{1}{2}}$, which yields a $(2k+1)$th order conservative semi-Lagrangian method for solving Eq. (3.1), may be summarized as,

1. Pre-compute constant coefficients, including
   (a) the matrices C^L_{2k+1} and C^R_{2k+1}, which are given in Section 3.1 for $k=1,2,3,4$,
   (b) constant coefficients $c_{rj},\ j=1,\ldots,k+1$, in order to reconstruct the flux from each stencil S_r as,

$$\sum_{j=1}^{k+1}c_{rj}f^n_{i-k+r+j-3}. \tag{3.35}$$

   Details of reconstruction can be found in Ref. [13]. The constants $c_{rj},\ r=1,\ldots,k+1,\ j=1,\ldots,k+1$, for $k=1,2,3,4$, are given in Table 3.1 for implementation convenience.
   (c) construct the linear weights $\gamma_r,\ r=1,\ldots,k+1$, s.t.,

$$\left(f^n_{i-k-1},\ldots,f^n_{i+k-1}\right)\cdot C^L_{2k+1}(:,1)=\sum_{r=1}^{k+1}\gamma_r\left(f^n_{i-k+r-2},\ldots,f^n_{i+r-2}\right)\cdot\left(c_{r1},\ldots,c_{r,k+1}\right)'. \tag{3.36}$$

   For implementation convenience, γ_r are listed in Table 3.2 for $k=1,2,3,4$.

2. Compute *xshift* $=v\frac{\Delta t}{\Delta x}$. Here we assume that *xshift* $\in\left[0,\frac{1}{2}\right]$.
3. Compute the smoothness indicator β_r,

$$\beta_r=\sum_{l=1}^{k}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}\Delta x^{2l-1}\left(\frac{\partial^l p_r(x)}{\partial x^l}\right)^2dx,$$

Table 3.1
The constants c_{rj} for $k = 1, 2, 3, 4$.

k	r	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$
1	1	−1/2	3/2			
	2	1/2	1/2			
2	1	1/3	−7/6	11/6		
	2	−1/6	5/6	1/3		
	3	1/3	5/6	−1/6		
3	1	−1/4	13/12	−23/12	25/12	
	2	1/12	−5/12	13/12	1/4	
	3	−1/12	7/12	7/12	−1/12	
	4	1/4	13/12	−5/12	1/12	
4	1	1/5	−21/20	137/60	−163/60	137/60
	2	−1/20	17/60	−43/60	77/60	1/5
	3	1/30	−13/60	47/60	9/20	−1/20
	4	−1/20	9/20	47/60	−13/60	1/30
	5	1/5	77/60	−43/60	17/60	−1/20

Table 3.2
The linear weights γ_r for $k = 1, 2, 3, 4$.

k	$r = 1$	$r = 2$	$r = 3$	$r = 4$	$r = 5$
1	1/3	2/3			
2	1/10	3/5	3/10		
3	1/35	12/35	18/35	4/35	
4	1/126	10/63	10/21	20/63	5/126

where $p_r(x)$ is the reconstruction polynomial from the stencil S_r. The derivation, as well as the explicit expression, for the fifth order indicator can be found in Ref. [22], and the derivation and expression for the seventh and ninth order cases can be found in Ref. [2]. We have listed explicit forms of the β_r in Appendix A for implementation convenience.

4. Compute the nonlinear normalized weights w_r,

$$w_r = \frac{\tilde{w}_r}{\sum_r \tilde{w}_r}, \quad r = 1, \ldots, k+1,$$

where $\tilde{w}_r = \frac{\gamma_r}{(\epsilon+\beta_r)^2}$.

5. Construct a nonlinear matrix $\widetilde{C}^L_{2k+1}$ as,

$$\widetilde{C}^L_{2k+1}(:,1) = w_1 \cdot (c_{11}, c_{12}, \ldots, c_{1,k+1}, 0, \ldots, 0) + \cdots + w_{k+1} \cdot (0, \ldots, 0, c_{k+1,1}, c_{k+1,2}, \ldots, c_{k+1,k+1}),$$
$$\widetilde{C}^L_{2k+1}(:,2) = C^L_{2k+1}(:,2)$$
$$\vdots$$
$$\widetilde{C}^L_{2k+1}(:,2k+1) = C^L_{2k+1}(:,2k+1).$$

6. Compute the numerical fluxes as

$$\hat{f}^n_{i-\frac{1}{2}}(\xi_0) = (f^n_{i-k-1}, \ldots, f^n_{i+k-1}) \cdot \widetilde{C}^L_{2k+1} \cdot (1, \xi_0, \ldots, \xi_0^{2k})'. \tag{3.37}$$

Finally, the resulting WENO fluxes, $\hat{f}^n_{i-\frac{1}{2}}(\xi_0)$ and $\hat{f}^n_{i+\frac{1}{2}}(\xi_0)$, are used in the update of the conservative semi-Lagrangian scheme in Eq. (3.10).

Table 4.1
Order of accuracy for (4.1) with $u(x, t = 0) = \sin(x)$ at $T = 20$. CFL = 1.2.

Mesh	Third order		Fifth order		Seventh order		Ninth order	
	Error	Order	Error	Order	Error	Order	Error	Order
32	3.27×10^{-2}	–	7.31×10^{-5}	–	2.64×10^{-6}	–	7.74×10^{-9}	–
64	8.55×10^{-3}	1.94	2.23×10^{-6}	5.03	3.84×10^{-8}	6.11	1.32×10^{-11}	9.20
96	3.66×10^{-3}	2.09	2.93×10^{-7}	5.00	3.27×10^{-9}	6.07	3.43×10^{-13}	9.00
128	1.93×10^{-3}	2.23	6.97×10^{-8}	5.00	5.91×10^{-10}	5.95	4.35×10^{-14}	7.18
160	1.11×10^{-3}	2.48	2.28×10^{-8}	5.00	1.55×10^{-10}	5.99	4.29×10^{-14}	–
192	6.81×10^{-4}	2.67	9.16×10^{-9}	5.00	5.15×10^{-11}	6.06	9.74×10^{-14}	–

4. Numerical tests

In this section, we first test the spatial accuracy of the proposed schemes for the simple test cases of linear advection and rigid body rotation, Section 4.1. In Section 4.2, we demonstrate the utility of the new method by applying it to classical problems from plasma physics, such as Landau damping and two-stream instability.

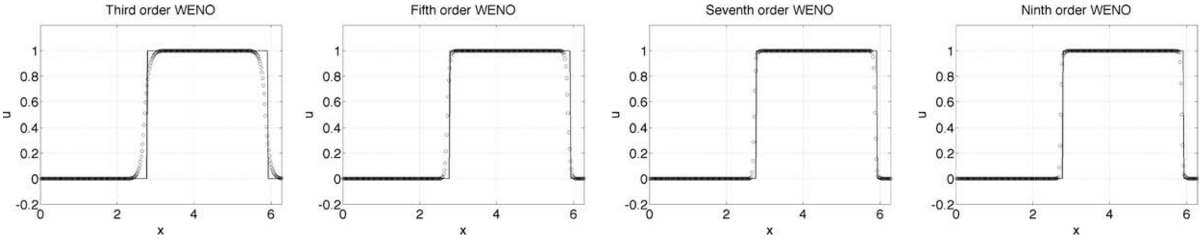


Fig. 4.1. Plots of the numerical solution (circles) versus the exact solution (solid line) of Eq. (4.1) with 320 grid points, CFL = 0.6 at $T = 20$. The initial condition is $u(x, t = 0) = 1$, for $x \in [\pi/2, 3\pi/2]$, and $u(x, t = 0) = 0$, otherwise.

Table 4.2
Order of accuracy for (4.2) with $u(x, y, t = 0) = \sin(x + y)$ at $T = 20$. CFL = 1.2.

Mesh	Third order		Fifth order		Seventh order		Ninth order	
	Error	Order	Error	Order	Error	Order	Error	Order
18	0.38	–	9.08×10^{-3}	–	6.69×10^{-4}	–	1.23×10^{-5}	–
36	0.11	1.79	3.09×10^{-4}	4.88	8.84×10^{-6}	6.24	1.96×10^{-8}	9.29
54	7.48×10^{-2}	0.96	4.06×10^{-5}	5.00	7.75×10^{-7}	6.00	4.68×10^{-10}	9.21
72	4.09×10^{-2}	2.10	9.61×10^{-6}	5.01	1.37×10^{-7}	6.02	3.40×10^{-11}	9.12
90	2.57×10^{-2}	2.08	3.15×10^{-6}	5.00	3.57×10^{-8}	6.04	4.82×10^{-12}	8.75

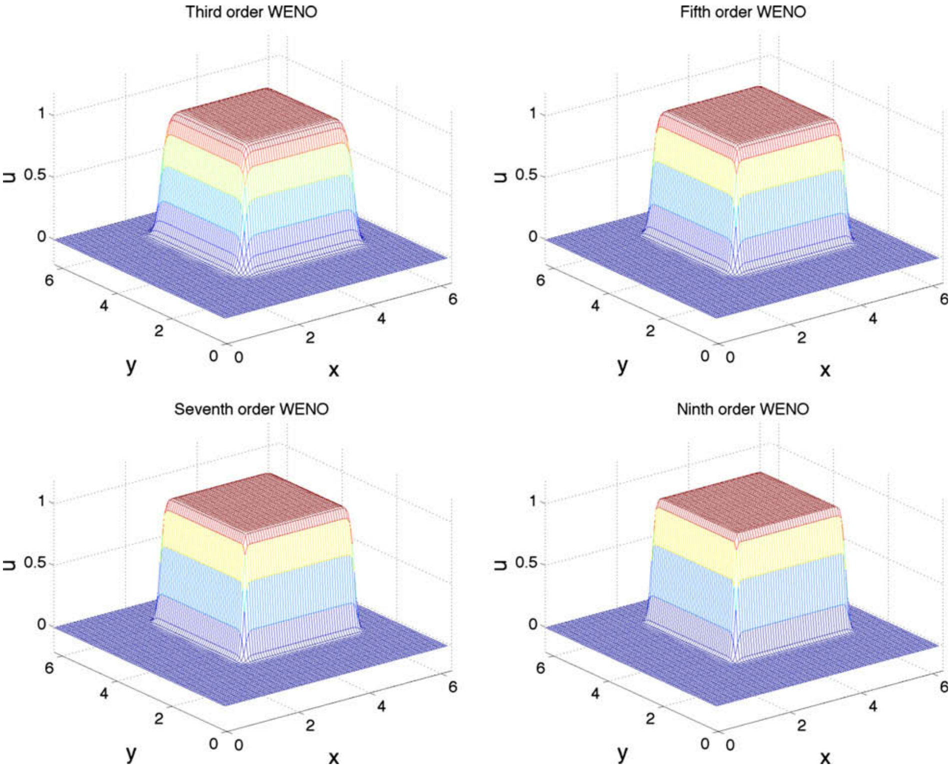


Fig. 4.2. Plots of the numerical solution of Eq. (4.2) with CFL = 0.6 at $T = 0.5$. The initial condition is $u(x, y, t = 0) = 1$, for $x, y \in [\pi/2, 3\pi/2]$, and $u(x, y, t = 0) = 0$, otherwise. The numerical mesh has a resolution of 90 × 90.

4.1. Test examples

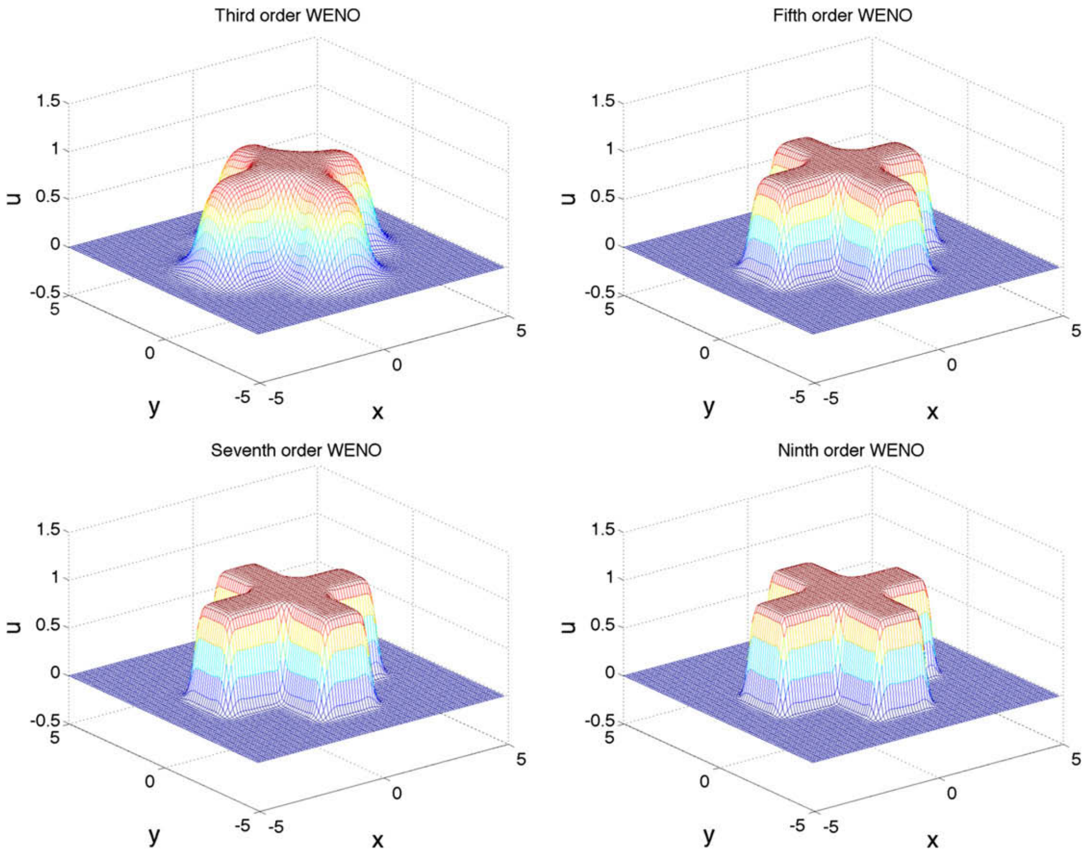
Example 4.1 (*one-dimensional linear translation*).

$$u_t + u_x = 0, \quad x \in [0, 2\pi]. \tag{4.1}$$

The conservative semi-Lagrangian methods with third, fifth, seventh and ninth order WENO reconstruction are used to solve Eq. (4.1). Table 4.1 gives the L^1 error, and the corresponding order of convergence, of the WENO reconstruction when applied to Eq. (4.1) with smooth initial data $u(x,0) = \sin(x)$. Eq. (4.1) has the smooth solution $u(x,t) = \sin(x-t)$, and we therefore expect WENO reconstruction to give high order convergence, which is observed. We note that the schemes with fifth and ninth order WENO reconstruction have a numerical rate of convergence close to the theoretical value. This is in contrast with third and seventh order WENO reconstruction, which does not behave as well. Fig. 4.1 are plots of numerical solutions of a moving rectangular wave. Non-oscillatory numerical capture of discontinuities is observed. Moreover, the data show that as the order of reconstruction increases, the numerical dissipation decreases.

Table 4.3
Order of accuracy for Eq. (4.3) with $u(x,y,t=0) = \exp(x^2 + y^2)$ at $T = 2\pi$, CFL = 1.2.

Mesh	Third order		Fifth order		Seventh order		Ninth order	
	Error	Order	Error	Order	Error	Order	Error	Order
18	1.20×10^{-2}	–	7.33×10^{-3}	–	3.68×10^{-3}	–	3.61×10^{-3}	–
36	4.58×10^{-3}	1.40	7.62×10^{-4}	3.27	2.26×10^{-4}	4.03	8.68×10^{-5}	5.38
54	1.86×10^{-3}	2.23	1.57×10^{-4}	3.89	1.87×10^{-5}	6.14	2.48×10^{-6}	8.77
72	9.25×10^{-4}	2.42	4.55×10^{-5}	4.31	2.45×10^{-6}	7.07	1.77×10^{-7}	9.18
90	5.31×10^{-4}	2.49	1.65×10^{-5}	4.55	5.01×10^{-7}	7.11	2.84×10^{-8}	8.19


Third order WENO

Fifth order WENO

Seventh order WENO

Ninth order WENO

Fig. 4.3. Plots of the numerical solution of Eq. (4.3) with CFL = 1.2 at $T = 2\pi$. The initial condition is $u(x,y,t=0) = 1$, for $(x,y) \in [-1,1] \times [-4,4] \cup [-4,4] \times [-1,1]$, and $u(x,t=0) = 0$, otherwise. The numerical mesh is 90 × 90.

Example 4.2 (*two-dimensional linear transport*).

$$u_t + u_x + u_y = 0, \qquad x \in [0, 2\pi], \quad y \in [0, 2\pi]. \tag{4.2}$$

The Strang splitting conservative semi-Lagrangian method, with third, fifth, seventh and ninth order WENO reconstruction, are applied to the simulation of Eq. (4.2). For any 2D linear transport equation, the semi-Lagrangian method is essentially a shifting procedure. Since the x-shifting and y-shifting operators commute, there is no dimensional splitting error in time and the spatial error is the dominant error. Table 4.2 gives the L^1 error and convergence rate of the scheme when applied to Eq. (4.2) with smooth initial data, $u(x,y,t) = \sin(x + y - 2t)|_{t=0}$. As expected, high order convergence is observed. Similar to the one-dimensional case, schemes with fifth and ninth order WENO reconstructions obtain the expected convergence rates, while in the third and seventh order schemes, a rate lower than the expected rate is observed. Fig. 4.2, the numerical solution of a moving 2D rectangular wave is plotted. As anticipated when using WENO reconstruction, non-oscillatory numerical capture of the discontinuities is observed. Moreover, as the order of the reconstruction increases, the edges of the discontinuities get sharper, i.e. the numerical dissipation decreases with the increase in order.

Example 4.3 (*rigid body rotation*).

$$u_t - yu_x + xu_y = 0, \qquad x \in [-2\pi, 2\pi], \quad y \in [-2\pi, 2\pi]. \tag{4.3}$$

The Strang splitting conservative semi-Lagrangian methods with third, fifth, seventh and ninth order WENO reconstruction are next applied to Eq. (4.3). Table 4.3 gives the L^1 error and convergence rates of the scheme when applied to smooth initial data $u(x,y,0) = \exp(x^2 + y^2)$. We observe that the spatial error still dominates the splitting error in time, which is somewhat unexpected. Further, high order convergence is observed. Fig. 4.3 is a plot of the numerical solution for a rotating cross with sharp edges. Non-oscillatory numerical capture of discontinuities is observed. As in the 2D transport example, the edges of discontinuities of the rotating cross example become sharper as the order of the reconstruction increases.

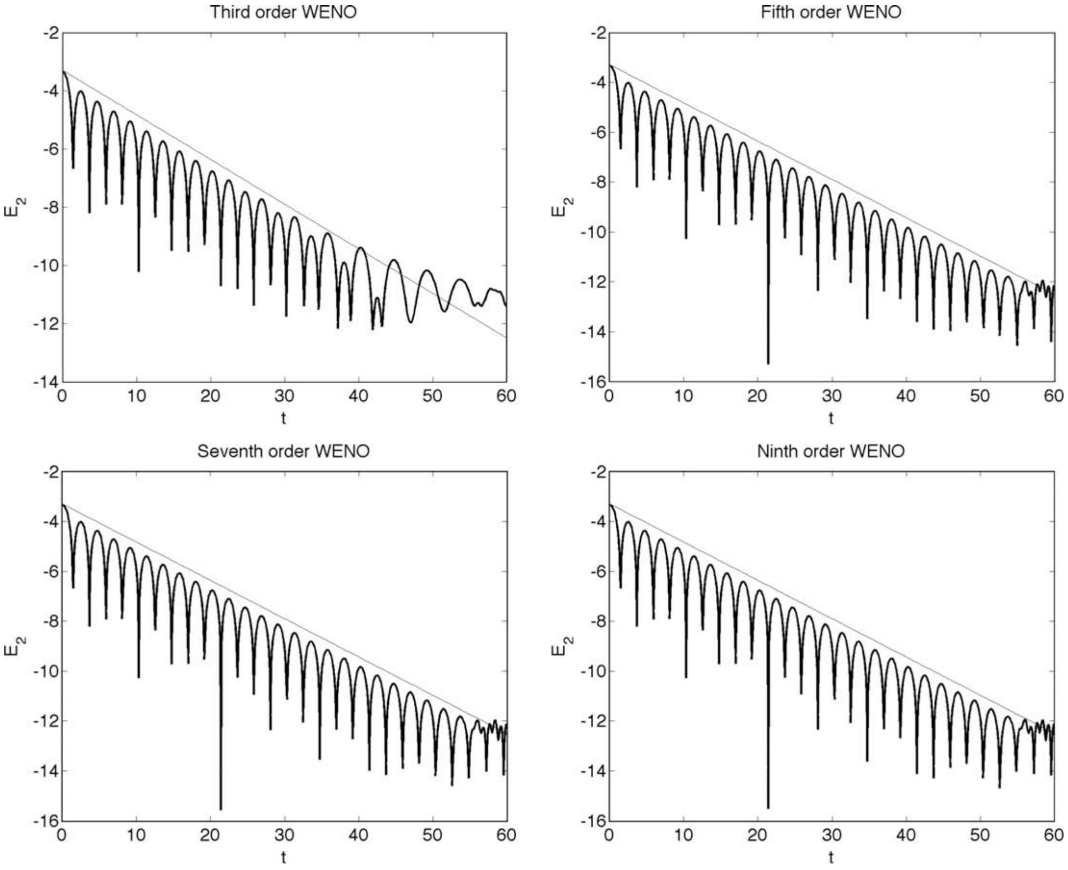


Fig. 4.4. Time evolution of the L^2 norm of the electric field using third, fifth, seventh and ninth order reconstruction.

4.2. The Vlasov–Poisson system

In this subsection, we apply the proposed Strang splitting semi-Lagrangian WENO method to the Vlasov equation in the VP system. Periodic boundary conditions are imposed in the x-direction and zero boundary conditions are imposed in the v-direction for all of our test problems. Because of the periodicity in space, a fast Fourier transform (FFT) is used to solve the 1D Poisson equation. $\rho(x,t)$ is computed by rectangular rule, $\rho(x,t) = \int f(x,v,t)dv = \sum_j f(x,v_j,t)\Delta v$, which is spectrally accurate [5], when the underlying function is smooth enough. Next, we recall some classical results on the VP system.

1. Preservation of the L^p norm, for $1 \leqslant p < \infty$.
$$\frac{d}{dt}\int_v \int_x f(x,v,t)^p dx dv = 0.$$
2. Preservation of the entropy
$$\frac{d}{dt}\int_v \int_x f(x,v,t) ln(f(x,v,t)) dx dv = 0.$$
3. Preservation of the energy
$$\frac{d}{dt}\left(\int_v \int_x f(x,v,t)v^2 dx dv + \int_x E^2(x,t)dx\right) = 0.$$

In our numerical experiments below, we checked the time evolution of these theoretically preserved quantities in the discrete sense. The norms/entropy/energy are numerically approximated by rectangular rule, which is again spectrally accurate, if the integrated function is smooth enough.

Example 4.4 (*Weak Landau damping*). Consider the example of weak Landau damping for the VP system. The initial condition used here is,

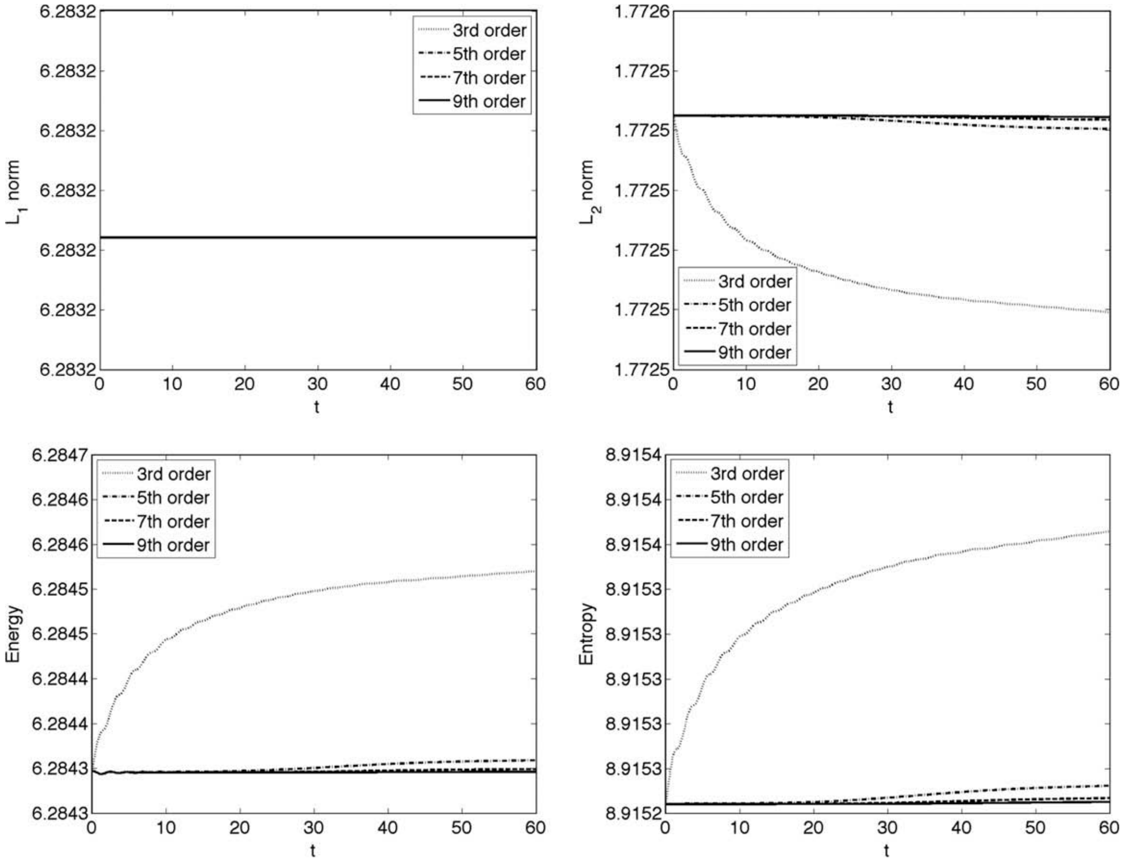


Fig. 4.5. Time evolution of L^1 (upper left) and L^2 (upper right) norms as well as the energy (lower left) and entropy (lower right).

$$f(x,v,t=0)=\frac{1}{\sqrt{2\pi}}(1+\alpha\cos(kx))\exp\left(-\frac{v^2}{2}\right), \tag{4.4}$$

with $\alpha=0.01$ and $k=0.5$. The time evolution of the L^2 norm of the electric field is plotted in Fig. 4.4. The correct damping of the electric field is observed in the plots, benchmarked with the theoretical value $\gamma=0.1533$ [15] (the solid line in Fig. 4.4). We observe that the fifth, seventh and ninth order WENO is better than the third order scheme in recovering the damping rate. The ninth order WENO does not outperform fifth/seventh order WENO, because the numerical mesh no longer supports the waves of filamentation, even though the order of approximation is very high. We remark that the electric field is an important theoretical metric. However, since the electric field is an integrated quantity, it is not a sensitive enough metric to demonstrate the advantage of a method that is high order in phase space. The advantages of our high order methods are better illustrated in preserving the L^1, L^2 norms, energy, entropy and in the example of two-stream instability later in this paper. The evolution of the L^1 and L^2 norm as well as the discrete energy and entropy of the solution are plotted in Fig. 4.5, which shows the advantage of using high order reconstruction in preserving the relevant physical norms.

Example 4.5 (*Strong Landau damping*). The next example we consider is the case of strong Landau damping. We simulate the VP system with the initial condition in Eq. (4.4) with $\alpha=0.5$ and $k=0.5$. Our numerical simulation parameters for all schemes are $v_{max}=5$, $N_x=64$, $N_v=128$ and $\Delta t=\Delta x$; where v_{max} is the maximum velocity on the phase space mesh, N_x is the number of grid points along the x axis, N_v is the number of grid points along the v axis, and Δt is the time step used. In the first two rows of Fig. 4.6, numerical solutions from the proposed Strang splitting conservative semi-Lagrangian scheme with various WENO reconstruction operators are plotted at a final time of $T=30$. In the last row of Fig. 4.6, numerical solu-

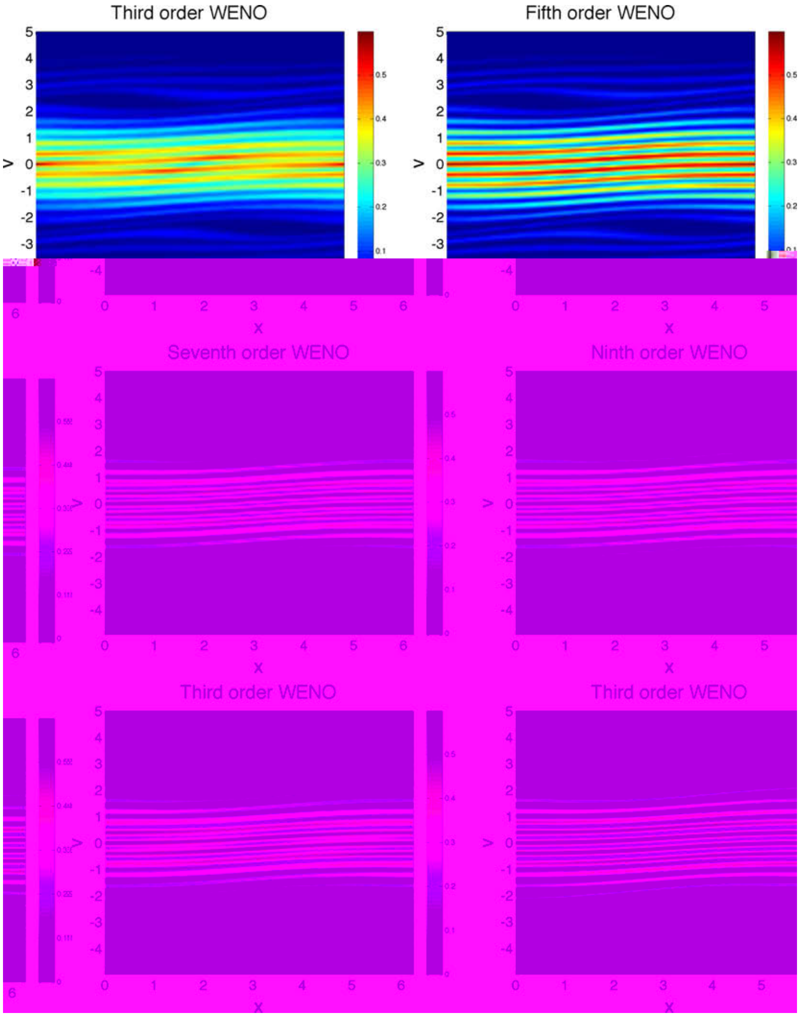


Fig. 4.6. The first two rows are phase space plots of strong Landau damping at $T=30$ using third, fifth, seventh and ninth order reconstruction. The numerical mesh is 64×128. The last row are phase space plots using third order reconstruction, with the numerical mesh being 256×512 (right).

tions from the third order scheme with two times finer and four times finer numerical mesh are plotted. They serve as reference solution, when compared with difference high order schemes on coarse meshes. While decent numerical resolution is obtained with all of the reconstructions, the scheme with a high order WENO reconstruction operator (ninth order) performs better than that with a low order reconstruction operator (third order). The high order reconstruction resolves fine solution structures and filamentation that is not resolved with the low order reconstruction when computed on the same mesh. This is consistent with the fact that, compared with low order schemes, high order schemes converge to the solution faster in smooth solution regions and are less dissipative around discontinuities. Fig. 4.7 shows the time development of the L^2 norm of the electric field from various WENO reconstructions, which is in consistent with the result reported in [15]. As noted in the weak Landau damping example, the electric field is an integrated quantity, and not particularly sensitive to the advantages of high order methods, which are better demonstrated below in the two-stream instability example. The discrete L^1 norm, L^2 norm, energy and entropy for the proposed method with various WENO reconstructions are plotted in Fig. 4.8. It is observed that schemes with higher order WENO reconstruction do a better job of preserving the L^2 norm, entropy and energy than those with lower order reconstruction. The scheme with ninth order does not particularly preserve the L^1 well, which is due to the appearance of some negative values in the numerical solution. The appearance of negative values is because of some small oscillations, especially of higher order schemes. Fig.4.9 gives the solution profile of strong Landau damping at different times using ninth order WENO reconstruction. It is clear that high order WENO reconstruction helps preserve solution structure, even when the details of the solution begin to become under resolved on a given mesh.

Example 4.6 (*Two-stream instability [15]*). Consider the symmetric warm two-stream instability, i.e. the electron distribution function in the VP system is started with the unstable initial condition [15],

$$f(x,v,t=0)=\frac{2}{7\sqrt{2\pi}}(1+5v^2)(1+\alpha((\cos(2kx)+\cos(3kx))/1.2+\cos(kx))\exp\left(-\frac{v^2}{2}\right), \tag{4.5}$$

with $\alpha=0.01,\ k=0.5$, the length of the domain in the x-direction is $L=\frac{2\pi}{k}$ and the background ion distribution function is fixed, uniform and chosen so that the total net charge density for the system is zero. Our numerical simulation parameters

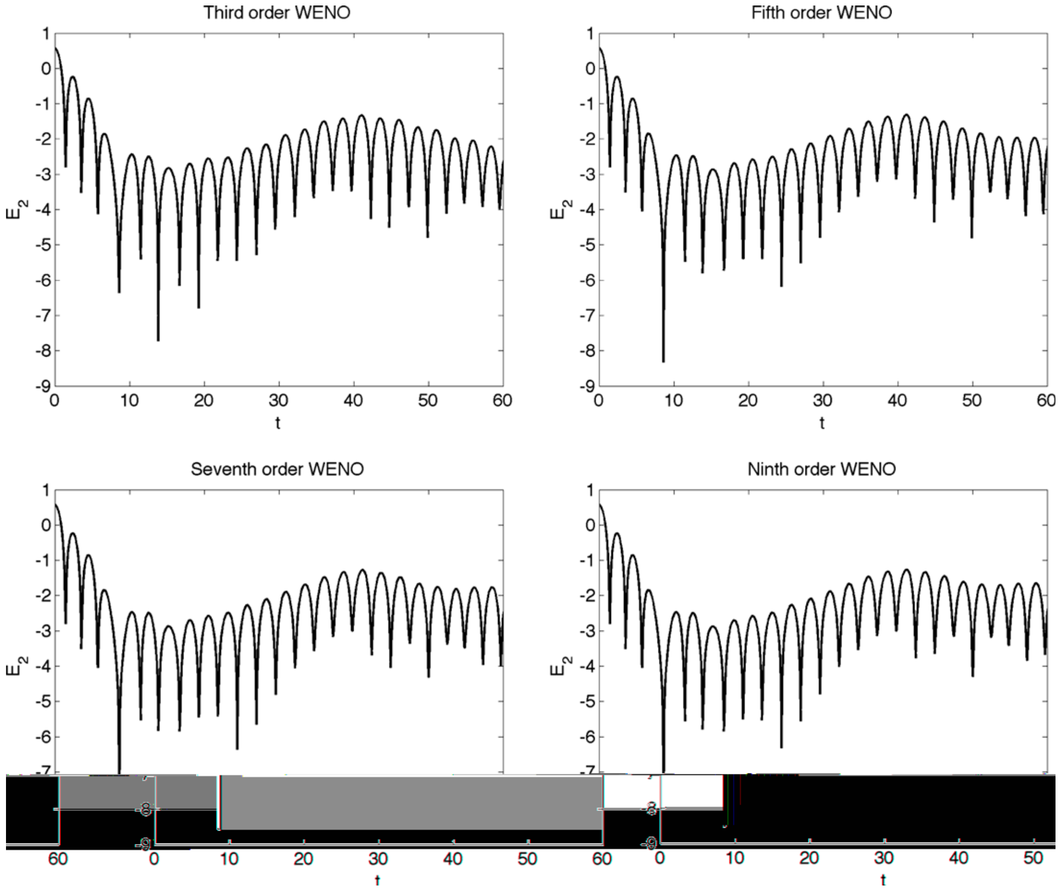


Fig. 4.7. Time evolution of the L^2 norm of the electric field using third, fifth, seventh and ninth order reconstruction.

are $v_{max} = 5,\ N_x = 64,\ N_v = 128,\ \Delta t = \Delta x$ for all schemes. Fig. 4.10 shows numerical solutions of phase space profiles at $T = 53$ from the proposed Strang splitting conservative semi-Lagrangian scheme with various WENO reconstruction operators. While decent numerical resolution is obtained with all of the reconstructions, the scheme with ninth order WENO reconstruction performs better than that with a low order reconstruction operator (third order). As in our previous examples, fine solution structure and filamentation are better resolved with the high order reconstruction. This is again consistent with the fact that high order schemes are not only more accurate but also less dissipative. Fig. 4.11 shows the solution profile of the scheme with third order WENO reconstruction with the resolution of the numerical mesh set to $N_x = 128,\ N_v = 256$ (left) and $N_x = 256,\ N_v = 512$ (right). In order to make a fair comparison in terms of work, the Δt is kept the same as in Fig. 4.10, which is allowed since this is a semi-Lagrangian scheme. It is observed that the resolution of the left plot of Fig. 4.11 is comparable to that of the fifth order scheme from Fig. 4.10, with the computational effort of the third order method in Fig. 4.11 being twice as expensive as the fifth order method in Fig. 4.10. The resolution of the right plot of Fig. 4.11 is comparable to that of the ninth order scheme from Fig. 4.10, with the computational effort of the third order method in Fig. 4.11 being four times as expensive as the ninth order method in Fig. 4.10. Fig. 4.12 shows the time development of the discrete L^2 norm and entropy for the proposed methods with various WENO reconstructions. As expected, the high order schemes are better at preserving the L^2norm and entropy of the system.

5. Conclusions

In this paper, we propose a novel Strang splitting semi-Lagrangian approach for solving the Vlasov equation, which makes it possible to use the WENO methodology in space. The method can be designed to be arbitrarily high order in space in a multi-dimensional setting.

Since the method is a semi-Lagrangian scheme, it does not have a time step restriction, allowing for a larger numerical time step than an explicit scheme, making the method computationally cheaper, and more flexible, than Eulerian formulations of the Vlasov–Poisson system which use explicit time stepping. The method has been tested extensively and for multi-dimensional problems, including transport and rotational problems, our numerical results demonstrate the advantages of high order accuracy combined with the essentially non-oscillatory resolution of sharp interfaces. We also applied the meth-

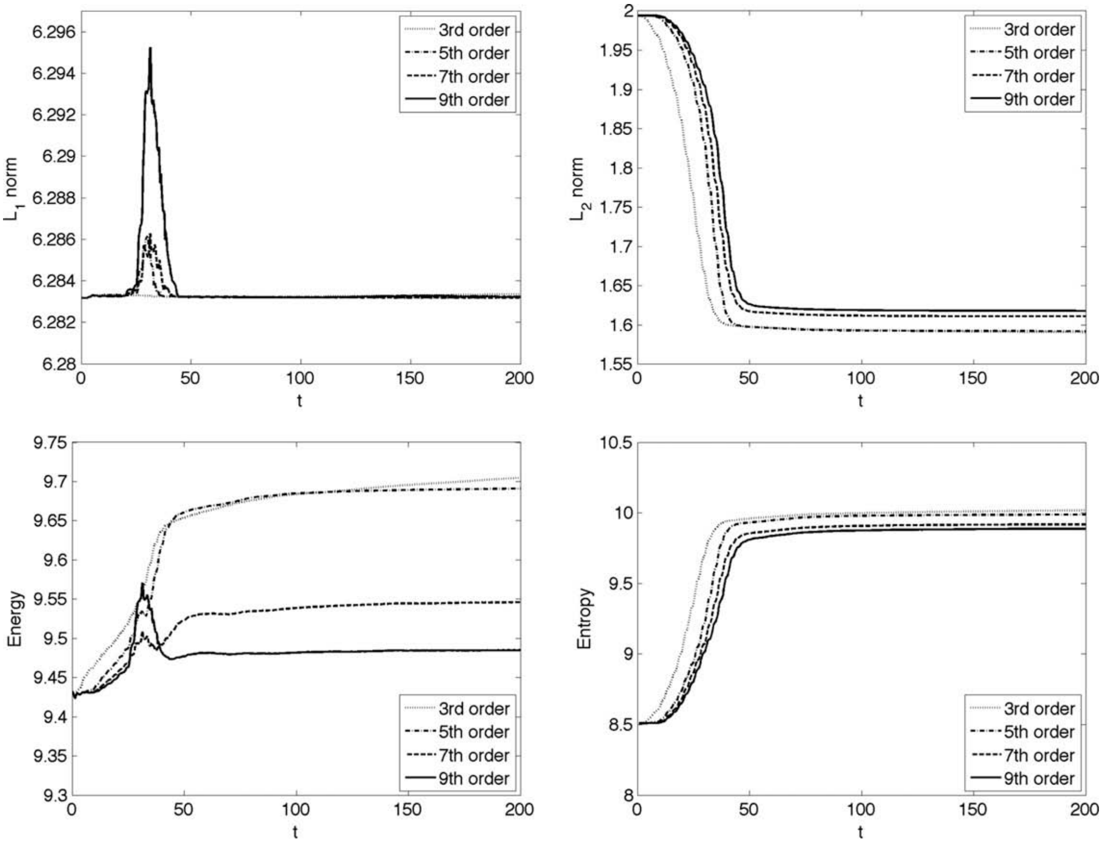


Fig. 4.8. Time evolution of the L^1 (upper left) and L^2 (upper right) norm, discrete energy (lower left) and entropy (lower right).

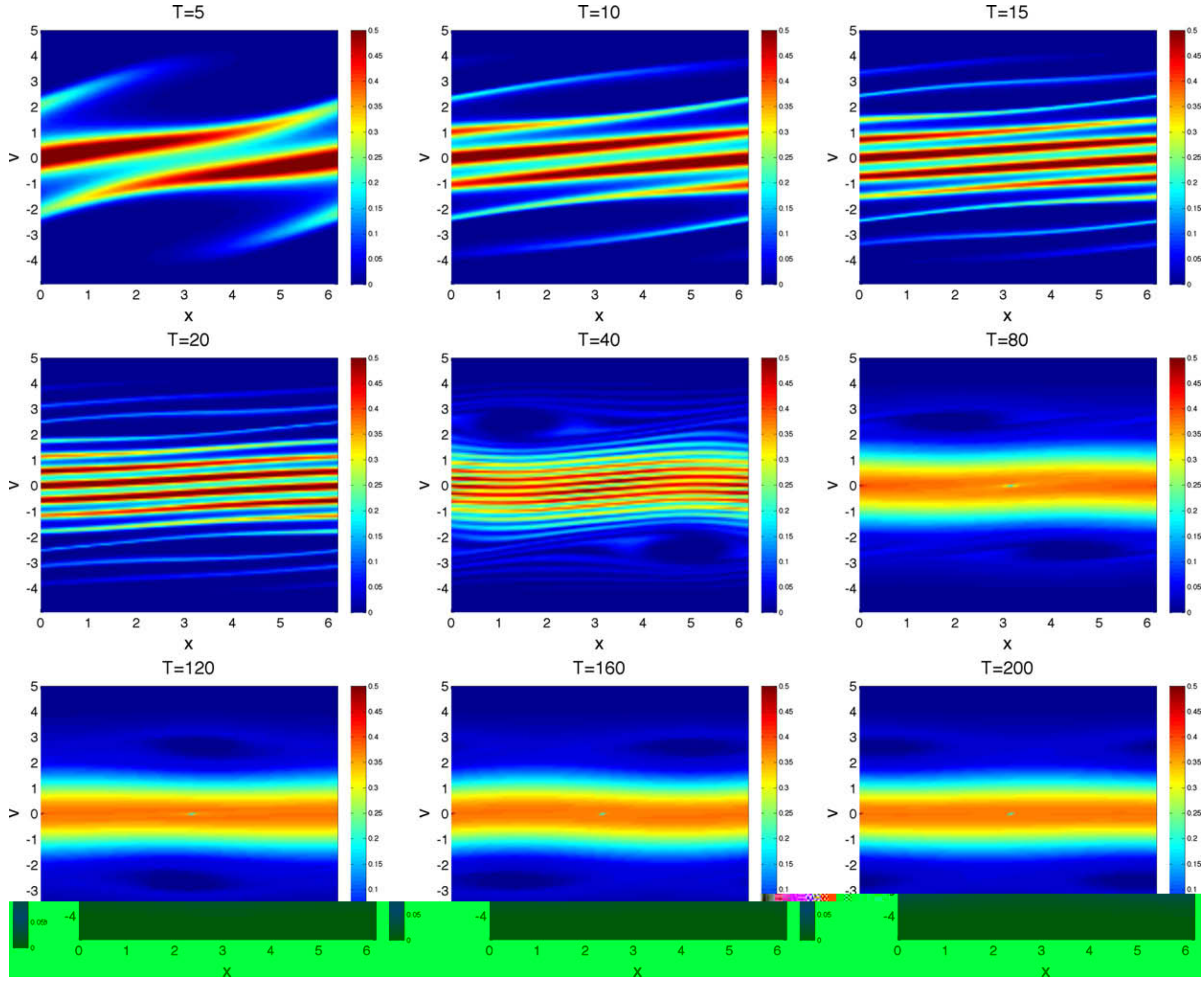


Fig. 4.9. Phase space plots of the time evolution of phase space for the test case of strong Landau damping computed with ninth order WENO reconstruction in the Strang splitting conservative semi-Lagrangian method. The time series is for $T \in \{5, 10, 15, 20, 40, 80, 120, 160, 200\}$.

od to the classical Landau damping and the two-stream instability in plasma physics. Methods with higher order reconstruction have demonstrated their superior abilities in resolving the physical phenomena as opposed to the results generated by low order methods. However, the order of convergence in time has been restricted to at best second order, due to the Strang splitting. A key observation is that, while the method is only second order in time, high order spatial resolution appears to play a more critical role in the quality of the solution, for this class of problems, than does temporal accuracy.

Acknowledgments

The authors would like to thank Dr. David Newman and Dr. Chi-Wang Shu for many helpful discussions. Further, they would like to thank AFOSR-Computational Mathematics and NSF-DMS for their support of this work.

Appendix A. The fifth, seventh and ninth smoothness indicators

For the fifth, seventh and ninth order WENO method, the smoothness indicators, β_r, are, respectively,

fifth order:

$$\beta_1 = \frac{13}{12}\left(f_{i-3}^n - 2f_{i-2}^n + f_{i-1}^n\right)^2 + \frac{1}{4}\left(f_{i-3}^n - 4f_{i-2}^n + 3f_{i-1}^n\right)^2,$$
$$\beta_2 = \frac{13}{12}\left(f_{i-2}^n - 2f_{i-1}^n + f_i^n\right)^2 + \frac{1}{4}\left(f_{i-2}^n - f_i^n\right)^2,$$
$$\beta_3 = \frac{13}{12}\left(f_{i-1}^n - 2f_i^n + f_{i+1}^n\right)^2 + \frac{1}{4}\left(3f_{i-1}^n - 4f_i^n + f_{i+1}^n\right)^2.$$

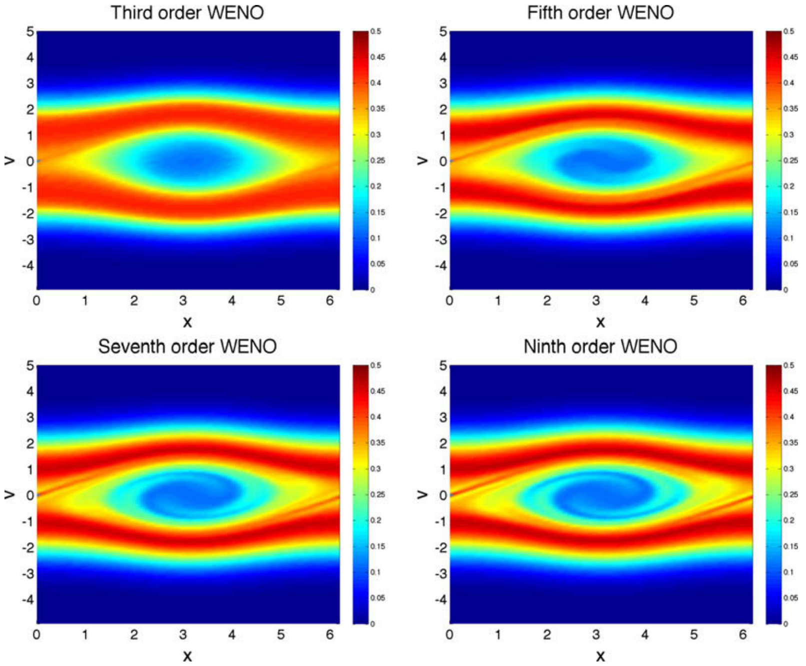


Fig. 4.10. Phase space plots of the two-stream instability $T = 53$ using third, fifth, seventh and ninth order reconstruction. The numerical mesh is 64 × 128.

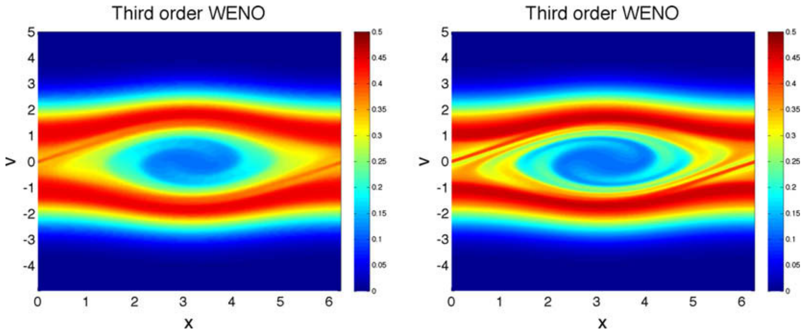


Fig. 4.11. Phase space plots for the two-stream instability $T = 53$ using a third order semi-Lagrangian WENO method. The numerical mesh is 128 × 256 (left) and 256 × 512 (right). The computational cost is 4 and 16 times greater than for the first plot in Fig. 4.10.

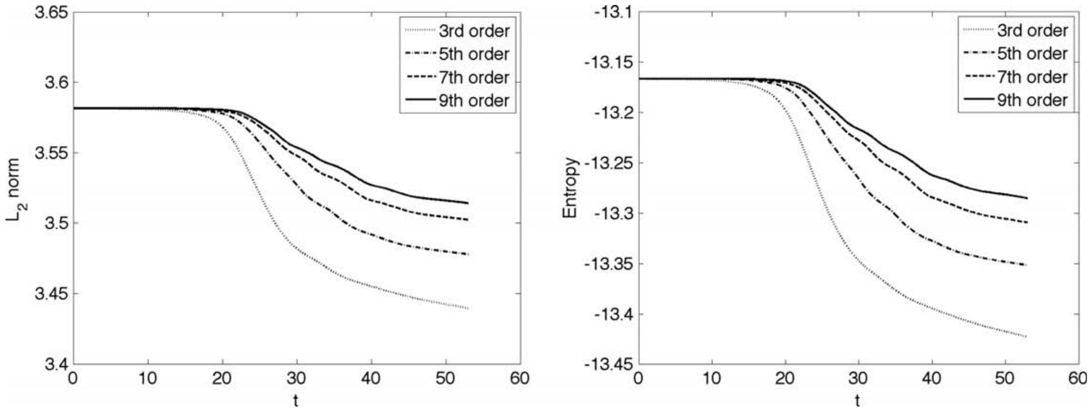


Fig. 4.12. Time development of the numerical L^2 norm (left) and entropy (right) of the two-stream instability.

seventh order:

$$\begin{aligned}\beta_1 =& f^n_{i-4}\left(547f^n_{i-4} - 3882f^n_{i-3} + 4642f^n_{i-2} - 1854f^n_{i-1}\right) + f^n_{i-3}\left(7043f^n_{i-3} - 17246f^n_{i-2} + 7042f^n_{i-1}\right)\\ &+ f^n_{i-2}\left(11003f^n_{i-2} - 9402f^n_{i-1}\right) + 2107\left(f^n_{i-1}\right)^2,\end{aligned}$$

$$\begin{aligned}\beta_2 =& f^n_{i-3}\left(267f^n_{i-3} - 1642f^n_{i-2} + 1602f^n_{i-1} - 494f^n_{i}\right) + f^n_{i-2}\left(2843f^n_{i-2} - 5966f^n_{i-1} + 1922f^n_{i}\right) + f^n_{i-1}\left(3443f^n_{i-1} - 2522f^n_{i}\right)\\ &+ 547\left(f^n_{i}\right)^2,\end{aligned}$$

$$\begin{aligned}\beta_3 =& f^n_{i-2}\left(547f^n_{i-2} - 2522f^n_{i-1} + 1922f^n_{i} - 494f^n_{i+1}\right) + f^n_{i-1}\left(3443f^n_{i-1} - 5966f^n_{i} + 1602f^n_{i+1}\right) + f^n_{i}\left(2843f^n_{i} - 1642f^n_{i+1}\right)\\ &+ 267\left(f^n_{i+1}\right)^2,\end{aligned}$$

$$\begin{aligned}\beta_4 =& f^n_{i-1}\left(2107f^n_{i-1} - 9402f^n_{i} + 7042f^n_{i+1} - 1854f^n_{i+2}\right) + f^n_{i}\left(11003f^n_{i} - 17246f^n_{i+1} + 4642f^n_{i+2}\right) + f^n_{i+1}\left(7043f^n_{i+1} - 3882f^n_{i+2}\right)\\ &+ 547\left(f^n_{i+2}\right)^2.\end{aligned}$$

ninth order:

$$\begin{aligned}\beta_1 =& f^n_{i-5}\left(22658f^n_{i-5} - 208501f^n_{i-4} + 364863f^n_{i-3} - 288007f^n_{i-2} + 86329f^n_{i-1}\right)\\ &+ f^n_{i-4}\left(482963f^n_{i-4} - 1704396f^n_{i-3} + 1358458f^n_{i-2} - 411487f^n_{i-1}\right) + f^n_{i-3}\left(1521393f^n_{i-3} - 2462076f^n_{i-2} + 758823f^n_{i-1}\right)\\ &+ f^n_{i-2}\left(1020563f^n_{i-2} - 649501f^n_{i-1}\right) + 107918\left(f^n_{i-1}\right)^2,\end{aligned}$$

$$\begin{aligned}\beta_2 =& f^n_{i-4}\left(6908f^n_{i-4} - 60871f^n_{i-3} + 99213f^n_{i-2} - 70237f^n_{i-1} + 18079f^n_{i}\right)\\ &+ f^n_{i-3}\left(138563f^n_{i-3} - 464976f^n_{i-2} + 337018f^n_{i-1} - 88297f^n_{i}\right) + f^n_{i-2}\left(406293f^n_{i-2} - 611976f^n_{i-1} + 165153f^n_{i}\right)\\ &+ f^n_{i-1}\left(242723f^n_{i-1} - 140251f^n_{i}\right) + 22658\left(f^n_{i}\right)^2,\end{aligned}$$

$$\begin{aligned}\beta_3 =& f^n_{i-3}\left(6908f^n_{i-3} - 51001f^n_{i-2} + 67923f^n_{i-1} - 38947f^n_{i} + 8209f^n_{i+1}\right)\\ &+ f^n_{i-2}\left(104963f^n_{i-2} - 299076f^n_{i-1} + 179098f^n_{i} - 38947f^n_{i+1}\right) + f^n_{i-1}\left(231153f^n_{i-1} - 299076f^n_{i} + 67923f^n_{i+1}\right)\\ &+ f^n_{i}\left(104963f^n_{i} - 51001f^n_{i+1}\right) + 6908\left(f^n_{i+1}\right)^2,\end{aligned}$$

$$\begin{aligned}\beta_4 =& f^n_{i-2}\left(22658f^n_{i-2} - 140251f^n_{i-1} + 165153f^n_{i} - 88297f^n_{i+1} + 18079f^n_{i+2}\right)\\ &+ f^n_{i-1}\left(242723f^n_{i-1} - 611976f^n_{i} + 337018f^n_{i+1} - 70237f^n_{i+2}\right) + f^n_{i}\left(406293f^n_{i} - 464976f^n_{i+1} + 99213f^n_{i+2}\right)\\ &+ f^n_{i+1}\left(138563f^n_{i+1} - 60871f^n_{i+2}\right) + 6908\left(f^n_{i+2}\right)^2,\end{aligned}$$

$$\begin{aligned}\beta_5 =& f^n_{i-1}\left(107918f^n_{i-1} - 649501f^n_{i} + 758823f^n_{i+1} - 411487f^n_{i+2} + 86329f^n_{i+3}\right)\\ &+ f^n_{i}\left(1020563f^n_{i} - 2462076f^n_{i+1} + 1358458f^n_{i+2} - 288007f^n_{i+3}\right) + f^n_{i+1}\left(1521393f^n_{i+1} - 1704396f^n_{i+2} + 364863f^n_{i+3}\right)\\ &+ f^n_{i+2}\left(482963f^n_{i+2} - 208501f^n_{i+3}\right) + 22658\left(f^n_{i+3}\right)^2.\end{aligned}$$

References

[1] C. Anderson, A method of local corrections for computing the velocity field due to a distribution of vortex blobs, Journal of Computational Physics 62 (1986) 111–123.

[2] D. Balsara, C.-W. Shu, Monotonicity preserving weighted essentially non-oscillatory schemes with increasingly high order of accuracy, Journal of Computational Physics 160 (2000) 405–452.

[3] J. Barnes, P. Hut, A hierarchical O($N\log N$) force-calculation algorithm, Nature (1986) 446–449.

[4] C.K. Birdsall, A.B. Langdon, Plasma Physics via Computer Simulation, Adam Hilger, New York, 1991.

[5] J. Boyd, Chebyshev and Fourier Spectral Methods, Courier Dover Publications, 2001.

[6] J.A. Carrillo, A. Majorana, F. Vecil, A semi-Lagrangian deterministic solver for the semiconductor Boltzmann–Poisson system, Communications in Computational Physics (2007) 1027–1054.

[7] J.A. Carrillo, F. Vecil, Nonoscillatory interpolation methods applied to Vlasov-based models, SIAM Journal on Scientific Computing 29 (2007) 1179.

[8] C. Cheng, G. Knorr, The integration of the Vlasov equation in configuration space, Journal of Computational Physics 22 (1976) 330–351.

[9] A. Christlieb, W. Hitchon, E. Keiter, A computational investigation of the effects of varying discharge geometry for an inductively coupled plasma, IEEE Transactions on Plasma Science 28 (2000) 2214–2231.

[10] A. Christlieb, R. Krasny, J. Verboncoeur, A treecode algorithm for simulating electron dynamics in a Penning–Malmberg trap, Computer Physics Communications 164 (2004) 306–310.

[11] A. Christlieb, R. Krasny, J. Verboncoeur, Efficient particle simulation of a virtual cathode using a grid-free treecode Poisson solver, IEEE Transactions on Plasma Science 32 (2004) 384–389.

[12] A. Christlieb, R. Krasny, J. Verboncoeur, J. Emhoff, I. Boyd, Grid-free plasma simulation techniques, IEEE Transactions on Plasma Science 34 (2006) 149–165.

[13] B. Cockburn, C. Johnson, C.-W. Shu, E. Tadmor, Advanced Numerical Approximation of Nonlinear Hyperbolic Equations, Springer, New York, 1998.

[14] G. Coppa, G. Lapenta, G. Dellapiana, F. Donato, V. Riccardo, Blob method for kinetic plasma simulation with variable-size particles, Journal of Computational Physics 127 (1996) 268–284.

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

[16] F. Filbet, E. Sonnendrücker, P. Bertrand, Conservative numerical schemes for the Vlasov equation, Journal of Computational Physics 172 (2001) 166–187.

[17] L. Greengard, V. Rokhlin, A fast algorithm for particle simulations, Journal of Computational Physics 73 (1987) 325–348.

[18] D. Hewett, Grid and particle hydrodynamics, Journal of Computational Physics 144 (1998) 358–378.

[19] D. Hewett, Fragmentation, merging and internal dynamics for PIC simulation with finite size particles, Journal of Computational Physics 189 (2003) 390–426.

[20] R. Hockney, J. Eastwood, Computer Simulation Using Particles, Institute of Physics Publishing, 1988.

[21] G. Jacobs, J. Hesthaven, High-order nodal discontinuous Galerkin particle-in-cell method on unstructured grids, Journal of Computational Physics 214 (2006) 96–121.

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

[23] K. Lindsay, R. Krasny, A particle method and adaptive treecode for vortex sheet motion in three-dimensional flow, Journal of Computational Physics 172 (2001) 879–907.

[24] K. Matyash, R. Schneider, R. Sydora, F. Taccogna, Application of a grid-free kinetic model to the collisionless sheath, Contributions to Plasma Physics 48 (2008) 116.

[25] G. Parker, W. Hitchon, Convected scheme simulations of the electron distribution function in a positive column plasma, Japanese Journal of Applied Physics 36 (1997) 4799–4807.

[26] J. Shi, C. Hu, C.-W. Shu, A technique of treating negative weights in WENO schemes, Journal of Computational Physics 175 (2002) 108–127.

[27] E. Sonnendruecker, J. Roche, P. Bertrand, A. Ghizzo, The semi-Lagrangian method for the numerical resolution of the Vlasov equation, Journal of Computational Physics 149 (1999) 201–220.

[28] J. Vay, P. Colella, P. Mccorquodale, B. Van Straalen, A. Friedman, D. Grote, Mesh refinement for particle-in-cell plasma simulations: applications to and benefits for heavy ion fusion, Laser and Particle Beams 20 (2003) 569–575.

[29] J. Verboncoeur, Particle simulation of plasmas: review and advances, Plasma Physics and Controlled Fusion 47 (2005) 46.

[30] T. Yabe, F. Xiao, T. Utsumi, The constrained interpolation profile method for multiphase analysis, Journal of Computational Physics 169 (2001) 556–593.

[31] S. Zaki, T. Boyd, L. Gardner, A finite element code for the simulation of one-dimensional Vlasov plasmas. II: Applications, Journal of Computational Physics 79 (1988) 200–208.

[32] S. Zaki, L. Gardner, T. Boyd, A finite element code for the simulation of one-dimensional Vlasov plasmas. I: Theory, Journal of Computational Physics 79 (1988) 184–199.

[33] T. Zhou, Y. Guo, C.-W. Shu, Numerical study on Landau damping, Physica D: Nonlinear Phenomena 157 (2001) 322–333.