



HAL
open science

A Characteristic Mapping Method for Vlasov-Poisson with Extreme Resolution Properties

Philipp Krah, Xi-Yuan Yin, Julius Bergmann, Jean-Christophe Nave, Kai
Schneider

► **To cite this version:**

Philipp Krah, Xi-Yuan Yin, Julius Bergmann, Jean-Christophe Nave, Kai Schneider. A Characteristic Mapping Method for Vlasov-Poisson with Extreme Resolution Properties. *Communications in Computational Physics*, 2024, 35 (4), pp.905-937. 10.4208/cicp.OA-2024-0012 . hal-04574672

HAL Id: hal-04574672

<https://hal.science/hal-04574672>

Submitted on 14 May 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Characteristic Mapping Method for Vlasov–Poisson with Extreme Resolution Properties

Philipp Krah^{*}, Xi-Yuan Yin[†], Julius Bergmann[‡]
Jean-Christophe Nave[§] and Kai Schneider[¶]

Abstract

We propose an efficient semi-Lagrangian characteristic mapping method for solving the one+one-dimensional Vlasov–Poisson equations with high precision on a coarse grid. The flow map is evolved numerically and exponential resolution in linear time is obtained. Global third-order convergence in space and time is shown and conservation properties are assessed. For benchmarking, we consider linear and nonlinear Landau damping and the two-stream instability. We compare the results with a Fourier pseudo-spectral method and results from the literature. The extreme fine-scale resolution features are illustrated showing the method’s capabilities to efficiently treat filamentation in fusion plasma simulations.

Keywords: characteristic mapping method; kinetic equations; plasma physics;

1 Introduction

Flows in the general setting transport quantities, e.g. fluids, plasmas, particles or their probability distribution function (PDF), from one place to another. Typically they generate rich mathematical multi-scale structures even from simple analytical initial conditions and require well-adapted numerical methods to solve the underlying governing partial differential equations. In the present work, we focus on the Vlasov–Poisson (VP) system, modeling particle evolution under their self-consistent electric field, and neglecting their collisions. We consider the equation in $1d+1d$, i.e. in the space of positions and velocities. This transport equation in phase space has numerous applications in plasma and astrophysics [32]. A prominent example is Landau damping in statistical physics of ionized gases.

As a kinetic equation, the VP equation is intrinsically high-dimensional, however, it is especially challenging due to the filamentation of the particle distribution function in phase space. The filamentation is a physical property of the system, that leads to a continuous generation of fine scales with increasing time. Resolving these scales requires computer simulations to accumulate more and more computer storage and resources within time.

^{*}Institut de Mathématiques de Marseille, Aix-Marseille Université, philipp.krah@univ-amu.fr

[†]LMFA-CNRS, Ecole Centrale de Lyon, xi-yuan.yin@ec-lyon.fr

[‡]Institut de Mathématiques de Marseille, Aix-Marseille Université, julius@bei-bergmann.de

[§]McGill University, jcnave@math.mcgill.ca

[¶]Institut de Mathématiques de Marseille, Aix-Marseille Université, kai.schneider@univ-amu.fr

To overcome the filamentation problem in numerical simulations two main approaches have been followed: the Lagrangian approach and the semi-Lagrangian approach. These methods shift the evolution of the PDF towards the evolution of its underlying transport structure. In the purely Lagrangian approach, known as the particle-in-cell method (PIC) (see for review [21, 5, 42]), a set of N_p particles is used to statistically approximate the distribution function in velocity space. The particles move according to their equations of motion in an electromagnetic field that is interpolated from a fixed grid to the particle's position. As the PDF is determined statistically the overall result is subject to statistical noise, with the variance decreasing only slowly with $1/\sqrt{N_p}$. Different approaches for denoising have been proposed, see e.g. [35], among them are wavelet-based density estimation [19, 41].

In order to obtain even more accurate results, while simultaneously respecting the particle's equation of motion, semi-Lagrangian approaches have been developed. This special type of Eulerian method represents the PDF on a fixed grid whereas time is evolved with the help of particle trajectories, known as the characteristic curve. Along these curves, the PDF is conserved and thus constant in time. The PDF is computed by tracing back the trajectory and interpolating its origin back on an Eulerian mesh. The first approach in this direction was already presented in the seventies by Cheng and Knorr [8] and has ever since got continuous attention in research [38, 28, 4, 3, 11, 9, 37, 10, 14].

In [38] Sonnendrücker et al. formalized the semi-Lagrangian approach for solving the VP equations by employing spline interpolations at the feet of the characteristics. While this method is renowned for its high precision, it is accompanied by the need to solve a global tri-diagonal system induced by the spline interpolation scheme. To overcome this computational challenge, [4] introduced local interpolation schemes based on Hermite and Lagrange polynomials, which were shown to exhibit high-order convergence results in space and second-order accuracy in time, as documented by [3]. The introduction of local interpolation schemes enhanced the parallelizability and adaptability of the method to unstructured meshes [4]. However, it introduced the requirement to transport gradients as additional fields. Subsequent developments in this direction focused on maintaining positivity and mass conservation, as presented for the backward [9], and the forward semi-Lagrangian method [11].

Another prominent methodology in the semi-Lagrangian setting is built on discontinuous Galerkin (DG) schemes, which have been explored by several research groups [10, 14, 37, 36]. In DG-based semi-Lagrangian methods, the function is advected and subsequently projected onto the space of piecewise polynomial functions. DG methods offer the advantage of high-order local reconstruction and inherent mass conservation, as well as computational benefits when compared to cubic spline-based semi-Lagrangian methods, as reported by Einkemmer in [13].

Although the scope of this paper lies on a semi-Lagrangian method we also briefly mention other methods like the purely Eulerian methods. They are based on a grid formulation of the distribution function and describe its time evolution on these fixed locations [25, 26, 47, 46, 6, 33, 16]. In these works often the phase space is discretized with a tailored basis, or filtering is applied to handle filamentation. A comparison of semi-Lagrangian and Eulerian methods can be found in [15]. A general overview of numerical techniques to solve VP systems is given in [12].

The characteristic mapping method (CMM) we are developing here for Vlasov–Poisson is a semi-Lagrangian method for solving transport-dominated problems. The origins of CMM were starting with semi-Lagrangian Gradient Augmented Level Set and Jet-Schemes [31]. Then an extension of the characteristic mapping method for linear advection equations was proposed in [27] and in [30]. One of the main features of CMM, which is a geometric approach and fully exploits the characteristic structure of the flow, is that quantities advected by the flow can be constructed as the function composition of the initial condition with the characteristic map. This flow map is a diffeomorphism and for incompressible flows volume preserving. Moreover, it has a one-parameter semi-group structure (time) which allows decomposing the map into sub-maps. In contrast to other refinement methods we thus use compositional refinement. Therefore the PDF can be in principle evaluated at any given resolution at any given time, preserving the features of the initial condition like positivity. This however comes with the cost of storing more and more sub-maps, which may limit the total run-time. More recently CMM has been extended for the incompressible 2d Euler equations [44], and the 3d Euler equations [45]. A generalization for solving flows on manifolds was likewise proposed for instance for computing tracer transport on the sphere [40] and for the incompressible Euler equations on the rotating sphere [39].

The current work aims to extend the CMM approach for the Vlasov–Poisson equation. We note that a related approach exists already in the literature [24], which uses the compositional structure of a flow map similarly to solve Vlasov–Poisson. However, the details of the implementation differ as they are using the symplectic structure derived from a Hamilton formulation of the problem.

In this work, we perform some numerical analysis to assess the convergence order and show the third order in space and time. Code validation is provided to illustrate the theoretically shown convergence properties numerically. Benchmarking and comparison with a classical Fourier pseudo-spectral code yield cross-validation and show the efficiency of CMM with respect to its resolution capabilities. Finally, some high-resolution computations of the two-stream instability and Landau damping provide deep insight into the fine-scale structure of the distribution function.

The remainder of the paper is organized as follows. First, we recall the Vlasov–Poisson equation and summarize its key properties (Section 2). Then we describe the principles of the characteristic mapping method and its adaptation to Vlasov–Poisson, the numerical implementation and some numerical analysis in Section 3. Section 4 presents the pseudo-spectral discretization for Vlasov–Poisson used for comparison and benchmarking with CMM. Different numerical test cases are presented in Section 5. Finally, some conclusion is given in Section 6.

2 The Vlasov–Poisson Equation

We introduce the Vlasov–Poisson equation

$$\partial_t f + v \partial_x f + \partial_x \phi \partial_v f = 0, \quad (1)$$

$$\partial_{xx} \phi + 1 - l \int f \, dv = 0, \quad (2)$$

subject to an initial condition on f and boundary conditions on f and ϕ .

Here $f: \Omega_x \times \Omega_v \times [0, T] \rightarrow \mathbb{R}$ denotes the *particle distribution function* (PDF), $\phi: \Omega_x \times [0, T] \rightarrow \mathbb{R}$

the *electric potential* with corresponding *electric field* $E(x, t) = -\partial_x \phi(x, t)$.

The electron charge density of the plasma is $\rho(x, t) = \int_{\Omega_v} f(x, v, t) dv$.

Definition 2.1 (Mass and momentum). For a solution $f(t, x, v)$ of the Vlasov–Poisson system we define the *mass (0. moment)*

$$\mathcal{M}(t) = \int_{\Omega_x} \rho(x, t) dx = \iint_{\Omega_x \times \Omega_v} f(x, v, t) dx dv,$$

and its *moment (1. moment)*

$$\mathcal{P}(t) = \int_{\Omega_x} j(x, t) dx = \iint_{\Omega_x \times \Omega_v} v f(x, v, t) dx dv.$$

For later reasons we write Eq. (1) in compact form using the Vlasov velocity field $\mathbf{u} = (v, \partial_x \phi)$ and the Vlasov nabla operator $\nabla^{\mathbf{u}} = (\partial_x, \partial_v)$. Thus we can rewrite Eq. (1) in terms of

$$\partial_t f + \mathbf{u} \cdot \nabla^{\mathbf{u}} f = 0.$$

The velocity field $\mathbf{u} = (v, \partial_x \phi)$ is divergence-free, which is straightforward to see:

$$\nabla^{\mathbf{u}} \cdot \mathbf{u} = \partial_x v + \partial_v \partial_x \phi(x, t) = 0.$$

Definition 2.2 (Energy). For a solution $f(t, x, v)$ of the Vlasov–Poisson system we define its *kinetic energy*

$$\mathcal{E}_{\text{kin}}(t) = \frac{1}{2} \int_{\Omega_v} \int_{\Omega_x} f(x, v, t) |v|^2 dx dv,$$

its *potential energy*

$$\mathcal{E}_{\text{pot}}(t) = \frac{1}{2} \int_{\Omega_x} |E(x, t)|^2 dx = \frac{1}{2} \int_{\Omega_x} \rho(t, x) \phi(t, x) dx$$

and *total energy*

$$\mathcal{E}(t) = \mathcal{E}_{\text{pot}}(t) + \mathcal{E}_{\text{kin}}(t).$$

The total mass, momentum and total energy are conserved:

$$\frac{d\mathcal{M}}{dt}(t) = 0, \quad \frac{d\mathcal{P}}{dt}(t) = 0, \quad \frac{d\mathcal{E}}{dt}(t) = 0.$$

3 Characteristic Mapping Method for Vlasov–Poisson

In the following, we describe how the characteristic mapping method can be used to solve the Vlasov–Poisson equation. In Section 3.1 we review the fundamentals of the method for a scalar advection equation with a given velocity field [27, 30]. Thereafter Section 3.2 introduces the velocity field and the boundary treatment that is specific to the Vlasov–Poisson equation.

3.1 Characteristic Mapping Method

The CMM considers a non-linear advection equation

$$\begin{cases} \partial_t \theta + \mathbf{u} \cdot \nabla \theta = 0 & \text{for } (\mathbf{x}, t) \in \Omega \times \mathbb{R}_+ \\ \theta(\mathbf{x}, 0) = \theta_0(\mathbf{x}) & \mathbf{x} \in \Omega, \end{cases} \quad (3)$$

with velocity field $\mathbf{u} \in \mathbb{R}^d$ that may depend on the advected state $\theta: (\mathbf{x}, t): \Omega \times \mathbb{R}_+ \rightarrow \mathbb{R}$ itself. If we follow the points $(\mathbf{x}, t) = (\gamma(t), t)$ that satisfy the ODE:

$$\frac{d\gamma}{dt} = \mathbf{u} \quad \text{with} \quad \gamma(0) = \mathbf{x} \in \Omega \quad (4)$$

we have:

$$\frac{d\theta}{dt}(\gamma(t), t) = \partial_t \theta + \frac{d\gamma}{dt} \cdot \nabla \theta = 0.$$

This means that along these points the solution stays constant in time and can be mapped back to its initial values:

$$\theta(\gamma(t), t) = \theta_0(\mathbf{x}).$$

The curves $\gamma: \mathbb{R}_+ \rightarrow \Omega$ of constant values θ are termed *characteristic curves*. Evolving the solution of Eq. (4) for all $\mathbf{x} \in \Omega$ simultaneously is done using the *characteristic map* $\mathbf{X}: \Omega \times \mathbb{R}_+ \rightarrow \Omega$. Thus the characteristic map $\mathbf{X}(\gamma(t), t) = \gamma(0) = \mathbf{x}$ evolves all characteristic curves consecutively and the solution of Eq. (3) at time t can be computed by the composition with the initial condition

$$\theta(\mathbf{x}, t) = \theta_0(\mathbf{X}(\mathbf{x}, t)).$$

Since $\mathbf{X}(\mathbf{x}, t)$ traces back the characteristic curve to its initial position, it is called *backward characteristic map*. This formulation requires an assumption on the regularity of the velocity field as well as on its divergence (to prevent crossings of characteristic curves) so that we maintain a smooth one-to-one relabeling symmetry between Eulerian positions and Lagrangian particles, i.e. that the characteristic map is a diffeomorphism [1]. This is the case if the velocity field is smooth and divergence-free, i.e.,

$$\nabla \cdot \mathbf{u} = 0.$$

From the definition of the map $\mathbf{X}(\gamma(t), t) = \gamma(0) = \mathbf{x}$ it directly follows that $\frac{d}{dt} \mathbf{X}(\gamma(t), t) = 0$ and thus \mathbf{X} solves

$$\begin{cases} \partial_t \mathbf{X} + \mathbf{u} \cdot \nabla \mathbf{X} = 0 & \text{for } (\mathbf{x}, t) \in \Omega \times \mathbb{R}_+ \\ \mathbf{X}(\mathbf{x}, 0) = \mathbf{x} & \mathbf{x} \in \Omega. \end{cases}$$

Note, that Eq. (3) is non-linear if \mathbf{u} is dependent on the state $\theta(\mathbf{x}, t)$. Therefore, the characteristic map is dependent on the state and thus varies for different initial conditions.

3.2 Advecting the Particle Distribution Function

Starting at Eq. (1) we can use the backward characteristic map $\mathbf{X}(x, v, t) = (X(x, v, t), V(x, v, t))$ to relate the distribution function to its initial condition:

$$f(x, v, t) = f_0(X(x, v, t), V(x, v, t)),$$

by solving

$$0 = \partial_t \mathbf{X} + \mathbf{u} \cdot \nabla^{\mathbf{u}} \mathbf{X}, \quad \text{where } \mathbf{u} = (v, \partial_x \phi)$$

$$\partial_{xx} \phi = l \int f \, dv - 1.$$

Since $\nabla^{\mathbf{u}} \cdot \mathbf{u} = 0$, we can define an associated 2D stream function

$$\psi(x, v, t) = \frac{v^2}{2} - \phi(x, t),$$

such that $\mathbf{u} = \nabla^{\mathbf{u}} \times \psi \mathbf{e}_z$. For convenience ψ is shown in Fig. 1 for the two-stream instability at the initial condition. For implementation reasons of the CMM method, we assume periodic boundary conditions in v additional to x .

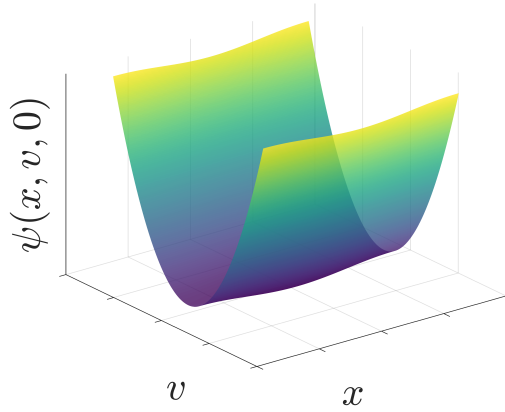


Figure 1: Stream function ψ for the initial condition of the two-stream instability.

3.2.1 Periodic Boundary Conditions in Velocity Space

The map advection assumes periodic boundaries in the x and v direction. For regularity of the map, we therefore have to maintain a periodic continuation of the velocity field $g(v): \mathbb{R} \rightarrow \mathbb{R}$, which is $g(v) = v$ for $v \in \Omega_{v^*} := [-v^*, v^*] \subset \Omega_v$ and a smooth continuation from $-v^*$ to v^* for $v \in \Omega_v \setminus \Omega_{v^*}$. An example of such a function g is visualized in the center of Fig. 2.

We define $g(v)$ as the anti-derivative of $h(v) := \partial_v g(v)$. As seen in the right of Fig. 2 h is a smooth heaviside function with $h(v) = 1$ at $v \in \Omega_{v^*}$. For our purpose we choose:

$$\sigma(v) = 1 - \frac{L_v}{a^2} \eta \left(\frac{|v| - L_v}{a} \right), \quad \text{with } a = 0.1L_v, \quad (5)$$

where

$$\eta(v) = \begin{cases} \frac{1}{I} e^{-1/(1-|v|^2)} & \text{if } |v| < 1 \\ 0 & \text{if } |v| \geq 1 \end{cases},$$

where $I = \int e^{-1/(1-|x|^2)} dx$. Noting the fact that for a periodic domain the total circulation $\int_{\Omega_x \times \Omega_v} \nabla \times \mathbf{u} \, dx \, dv$ of the associated velocity field $\mathbf{u} = (u_1, u_2) = (g(v), \partial_x \phi)$ vanishes. Thus from

$$\nabla \times \mathbf{u} = \partial_v g(v) - \partial_{xx} \phi(x)$$

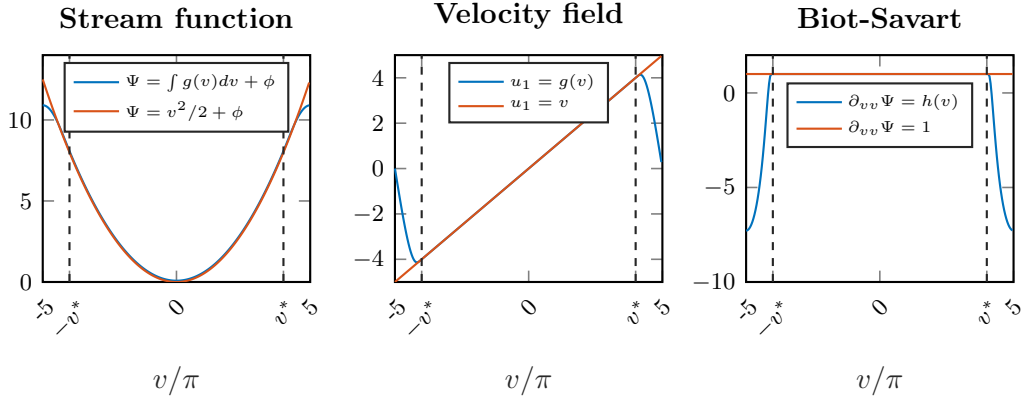


Figure 2: Periodization of the velocity field using a rescaled and shifted smooth Heaviside function $h(v)$. From left to right: resulting periodized stream function, a smooth continuation of the velocity component u_1 at different grid locations v , and the rescaled and shifted Heaviside function.

we see that the total circulation can only vanish if $h(v)$ fulfills

$$\int_{\Omega_v} h(v) dv = 0. \quad (6)$$

To ensure numerically that Eq. (6) and $h(v) = 1$ in $v \in \Omega_{v^*}$ we rescale and shift σ defined in Eq. (5) such that:

$$h(v) = \frac{\sigma - \int_{\Omega_v} \sigma(v) dv}{\max(\sigma - \int_{\Omega_v} \sigma(v) dv)}.$$

The function h is visualized together with the resulting constant velocity component u_1 and the associated stream function $\psi(x, v)$ along the v -direction in Fig. 2. It can be seen that the plotted functions Ψ, g, h intersect with the exact values in the domain Ω_{v^*} . The boundary layer in which the functions differ from the exact values is of size $a = L_v - v^*$. It is a tuning parameter of our algorithm, which will be discussed in the Appendix A.

This boundary issue is an inherent problem for mapping-based methods for kinetic equations. We denote by $\Phi_{[0,t]}$ the exact flow map of the unmodified phase space velocity $\mathbf{u}^e = (v, \partial_x \phi)$. We note that since $u_2 = \partial_x \phi$ is independent of x and is in general non-zero, we have that for any strip $R = S^1 \times [v_0, v_1]$ the image under forward flow will generally exit the strip, i.e. $\Phi_{[0,t]}(R) \not\subset R$ preventing the direct use of a bounded computational domain. However, the relevant error metric is the advection error of the density f , i.e. $\|f_0(\mathbf{X}_{[t,0]}) - f_0(\Phi_{[t,0]})\|_\infty \leq C \|\nabla f_0|_{\Phi_{[t,0]}}(\mathbf{X}_{[t,0]} - \Phi_{[t,0]})\|_\infty$. Therefore, it is sufficient to control the backward map error where ∇f_0 is nonzero. More generally, if one is allowed to assume a compactly supported initial condition, we let $R_0 = \text{supp}(f_0)$ the support of f_0 and R_t the support of $f = f_0 \circ \Phi_{[t,0]}$ at time t . Writing $\mathbf{1}_R$ for the indicator function of the set R , we have that $\mathbf{1}_{R_t} = \mathbf{1}_{R_0} \circ \Phi_{[t,0]}$. The advection error then becomes

$$\begin{aligned} \|f_0(\mathbf{X}_{[t,0]}) - f_0(\Phi_{[t,0]})\|_\infty &\leq C \|\nabla f_0\|_\infty \|(\mathbf{X}_{[t,0]} - \Phi_{[t,0]}) \mathbf{1}_{R_0} \circ \Phi_{[t,0]}\|_\infty \\ &= C \|\nabla f_0\|_\infty \|(\mathbf{X}_{[t,0]} - \Phi_{[t,0]}) \mathbf{1}_{R_t}\|_\infty. \end{aligned}$$

If we further suppose that there exists some strip $\Omega_{v^*} = S^1 \times [-v^*, v^*]$ such that $R_t \subset \Omega_{v^*}$ for all $t \in [0, T]$, we have

$$0 = (\partial_t + \mathbf{u} \cdot \nabla) \mathbf{X}_{[t,0]} - (\partial_t + \mathbf{u}^e \cdot \nabla) \Phi_{[t,0]} = (\partial_t + \mathbf{u}^e \cdot \nabla) (\mathbf{X}_{[t,0]} - \Phi_{[t,0]})$$

inside Ω_{v^*} so that $(\mathbf{X}_{[t,0]} - \Phi_{[t,0]})$ satisfies the advection equation under \mathbf{u}^e in $\Omega_{v^*} \times [0, T]$ with initial condition $\mathbf{X}_{[0,0]} - \Phi_{[0,0]} = 0$ in Ω_{v^*} . It follows from the advection property that this difference remains zero in the set $\{(x, v, t) \in \Omega_{v^*} \times [0, T] \mid \Phi_{[t,0]}(x, v) \in \Omega_{v^*}\} \supset \text{supp}(f)$. Hence $\|f_0(\mathbf{X}_{[t,0]}) - f_0(\Phi_{[t,0]})\|_\infty = 0$.

To apply this flow modification, we must know *a priori* that $\text{supp}(f) \subset \Omega_{v^*}$ for all $t \in [0, T]$. If the support eventually exits the strip, a reinitialization is then required.

3.3 Numerical Implementation

The implementation details closely follow [44] and are based on the gradient-augmented level set method (GALS) and Jet-scheme frameworks. Here we only review the most relevant parts. However, for a detailed explanation, we refer the reader to [44, 43].

Spatial Discretization. The CMM implementation makes use of three different kinds of grids, which are summarized in Table 1.

Grid	Size	Explanation
Coarse grid G	$N \times N$	Stores solution of advection equation
Sampling grid G_f	$N_f \times N_f$	Stores the samples of the initial condition
Velocity grid G_ψ	$N_\psi \times N_\psi$	Velocity samples

Table 1: Overview of the different computational grids used in this work.

The computational domain is discretized with an equidistant grid G . In the initial step of the simulation, the PDF is sampled on the sampling grid G_f of size $N_f \times N_f$. Therefore the map is interpolated using cubic Hermite interpolation and the PDF is evaluated $f = f_0 \circ \mathcal{H}_{G_f}[\mathbf{X}]$ at the corresponding positions. Next, the electric potential is computed on the sampling grid, by integrating over f in space and solving the 1D Poisson equation:

$$\hat{\phi}(k_x, t) = \frac{1}{k_x^2} \mathcal{F}[1 - \int f(\cdot, v, t) dv](k_x), \quad (7)$$

where \mathcal{F} denotes the Fourier transform (FFT) defined in Section 4. Depending on the resolution of the velocity grid G_ψ we perform zero padding, to upsample the solution from the sampling grid G_f to the velocity grid G_ψ . From the solution of Eq. (7), we form the stream function:

$$\psi(x, t) = g(v) - \mathcal{F}^{-1}[\hat{\phi}(\cdot, t)](x)$$

with the precomputed periodic velocity field $g(v)$. The Hermite representation $\mathcal{H}_{G_\psi}[\psi]$ of the stream function ψ involves the derivatives of ψ with respect to x and v direction, as well as the cross derivative xv . Here we make use of $h(v) = \partial_v g(v)$ and $\mathcal{F}^{-1}[\hat{\phi}(k_x, t)](x)$ that have been already computed. Thus, no further 2D FFT is necessary since the cross derivative vanishes. From the obtained stream function in Hermite form we can compute the velocity using the identity:

$$\mathbf{u} = \nabla \times \mathcal{H}_{G_\psi}[\psi].$$

Temporal Discretization. With the given velocity field we integrate Eq. (4) using a 3rd order Runge–Kutta backwards in time.

Compositional Structure and Remapping Criteria. The incompressibility of the velocity field implies that the map is volume preserving, which means that the determinant of its Jacobian equals one. Numerically this property is violated and remapping becomes necessary. The compositional structure of the map, i.e. its decomposition into sub-maps, is hereby helpful and increasingly complicated spatial features in the solution can thus be adaptively resolved. To this end, we monitor the determinant of the Jacobian of the map over time, which should be equal or close to one. We introduce the following error criterion to approximate the a posteriori error estimate [44]:

$$e_{\text{det}}^n = \|\det \nabla \mathbf{X}^n - 1\|_{\infty} \leq \delta_{\text{det}},$$

where δ_{det} is the threshold for the Jacobian error, also called *incompressibility threshold* in the forthcoming. In case that e_{det}^n exceeds this threshold, we store the sub-map \mathbf{X}^n , reinitialize the new map with the identity and proceed with the time evolution. This remapping procedure guarantees volume preservation of the map up to a certain accuracy. Therefore δ_{det} is chosen as small as possible, but large enough, such that all sub-maps can be stored in the active memory. For our fine-scale simulations, we discuss the influence of δ_{det} in Appendix B.

3.4 Error Analysis

The numerical method for the evolution of the characteristic map is the same as the one studied in [44] with a key difference in the definition of the velocity field. In [44], the numerical velocity field $\tilde{\mathbf{u}}$ is obtained from the map \mathbf{X} using the Biot-Savart law on the vorticity field: $\tilde{\mathbf{u}} = -\nabla \times \Delta^{-1} \omega_0(\mathbf{X}(\mathbf{x}, t))$. In the case of Vlasov–Poisson, the x -component of the velocity field is given analytically and is exact within the region $[-v^*, v^*]$. The v -component of the velocity field is obtained from solving a 1D Poisson equation for the advected distribution function:

$$u_v(x) = \partial_x \phi = l \partial_x \Delta_x^{-1} \left(\int f_0(\mathbf{X}(x, v, t)) dv - 1 \right).$$

The error term for the velocity field is given by

$$u_v - u_v^* \approx l \partial_x \Delta_x^{-1} \left(\int \nabla f_0 \cdot (\mathbf{X}(x, v, t) - \mathbf{X}^*(x, v, t)) dv \right),$$

for smooth initial condition f_0 . We note that the velocity error is a 1st anti-derivative of the map error in the x component and hence can be bounded by the map error. This is enough to close the inequalities in the error bounds as seen in [44] and [45]. Therefore, we get the same error estimates as in [44].

Using Hermite cubic interpolation coupled with 3^{rd} order in time local Lagrange extrapolation and 3^{rd} order Runge–Kutta integrators for the one-step maps, we expect global 3^{rd} order for the map error:

$$\|\mathbf{X}(\cdot, t^n) - \mathbf{X}^n\|_{\infty} = \mathcal{O}(\Delta x^3 + \Delta t^3).$$

Assuming smooth initial condition, this also implies a 3^{rd} order convergence for the distribution function f . For mass conservation, we have

$$\mathcal{M} = \iint f_0 \, dx \, dv = \iint f_0 \circ \mathbf{X}^n \det(\nabla \mathbf{X}^n) \, dx \, dv$$

by a change of variables for the integration. It follows that the mass conservation error is bounded by the minimum of the distribution function error and the volume conservation error. The conservation errors for momentum and total energy are also controlled by the distribution function error and are therefore 3^{rd} order. In summary, we define the following errors:

$$\Delta \mathbf{X} = \|\mathbf{X}^{\text{ref}}(\cdot, t^n) - \mathbf{X}^n\|_\infty \quad (\text{map error}), \quad (8)$$

$$\Delta f = \|f^{\text{ref}}(\cdot, t^n) - f^n\|_\infty \quad (\text{distribution function error}), \quad (9)$$

$$\Delta \mathcal{M} = |\mathcal{M}(t^n) - \mathcal{M}(t^0)| \quad (\text{mass conservation error}), \quad (10)$$

$$\Delta \mathcal{P} = |\mathcal{P}(t^n) - \mathcal{P}(t^0)| \quad (\text{momentum conservation error}), \quad (11)$$

$$\Delta \mathcal{E} = |\mathcal{E}(t^n) - \mathcal{E}(t^0)| \quad (\text{energy conservation error}), \quad (12)$$

that are of 3^{rd} order and will be studied numerically in Section 5.

4 Pseudo-Spectral Discretization of Vlasov–Poisson

Assuming the same periodized velocity field as introduced in Section 3.2.1 we can utilize a classical Fourier pseudo-spectral method [22]. Such a periodization approach where a known profile has been substrated and the perturbation becomes periodic has been used in [18] in the context of combustion problems. The Fourier projection of the density distribution function f is given by

$$P_N f(\mathbf{x}) = f_N(\mathbf{x}) = \sum_{|\mathbf{k}| \lesssim N/2} \hat{f}_{\mathbf{k}} e^{i\mathbf{k} \cdot \mathbf{x}}, \quad \hat{f}_{\mathbf{k}} = \frac{1}{(2\pi)^d} \int_{\mathbb{T}^d} f(\mathbf{x}) e^{-i\mathbf{k} \cdot \mathbf{x}} \, d\mathbf{x}$$

where $\mathbb{T} = \mathbb{R}/(2\pi\mathbb{Z})$ and d is the dimension. Note that $|\mathbf{k}| \lesssim N/2$ is understood in the sense $-N/2 \leq k < N/2$ and correspondingly in higher dimensions for each component of \mathbf{k} .

Applying the spectral discretization to the one+one-dimensional Vlasov–Poisson equation ($d = 2$) with periodic boundary conditions and suitable initial condition $f(\mathbf{x}, t = 0) = f_0(\mathbf{x})$ yields the Galerkin scheme

$$\partial_t f_N + (P_N(\mathbf{u}_N \cdot \nabla^u f_N)) = 0 \quad \text{for } \mathbf{x} \in \mathbb{T}^2 \quad \text{and } t > 0,$$

which corresponds to a nonlinear system of $N \times N$ coupled ODEs for $\hat{f}_{\mathbf{k}}(t)$ with $|\mathbf{k}| \lesssim N/2$. A pseudo-spectral evaluation of the nonlinear term is utilized, and the product in physical space is fully dealiased. In other words, the Fourier modes retained in the expansion of the solution are such that $|\mathbf{k}| \leq k_C$, where k_C is the desired cut-off wave number, but the grid has $N = 3k_C$ points in each direction, versus $N = 2k_C$ for a non-dealiased, critically sampled product. This dealiasing makes the pseudo-spectral scheme equivalent

to a Fourier–Galerkin scheme up to round-off errors [22], and is thus conservative for the L_2 norm.

For time discretization of the resulting ODE systems we use a classical Runge–Kutta scheme of order 3. For details on the convergence and stability of the above spectral schemes, we refer to [2].

Note that Eq. (2) can be efficiently solved in Fourier space assuming periodic boundary conditions. Therefore we discretize Eq. (1) in space using pseudo-spectral scheme [22]

$$\begin{aligned}\hat{\phi}(k_x, t) &= \mathcal{F}[\phi(\cdot, t)](k_x) = \int \phi(x, t) e^{-ik_x x} dx, \\ \hat{f}(k_x, k_v, t) &= \mathcal{F}[f(\cdot, \cdot, t)](k_x, k_v) = \int \int f(x, v, t) e^{-i(k_x x + k_v v)} dx dv,\end{aligned}$$

and the inverse:

$$\begin{aligned}\phi(x, t) &= \frac{1}{2\pi} \int \hat{\phi}(k_x, t) e^{ik_x x} dk_x, \\ f(x, v, t) &= \frac{1}{4\pi^2} \int \int \hat{f}(k_x, k_v, t) e^{i(k_x x + k_v v)} dk_x dk_v.\end{aligned}$$

Setting this into Eq. (2) yields

$$\begin{aligned}\hat{\phi}(k_x, t) &= \frac{1}{k_x^2} \mathcal{F}[(1 - l \int f dv)](k_x) = \frac{1}{k_x^2} \left(\mathcal{F}[1](k_x) - \underbrace{l \mathcal{F}[\int f dv]}_{\hat{f}(k_x, 0, t)}(k_x) \right) \\ &= \frac{1}{k_x^2} (\delta(k_x) - l \hat{f}(k_x, 0, t)).\end{aligned}\tag{13}$$

After solving Eq. (13) we solve Eq. (1) with a pseudo-spectral method. To avoid aliasing errors we remove 1/3 of the Fourier modes, i.e., the high frequencies. This filtering is known as the 2/3 rule and has proven to lead to the conservation of the L_2 norm of the solution [2]. Furthermore, we remark that the total mass \mathcal{M} is conserved over time since it is equal to the first Fourier coefficient that is unchanged during time integration.

5 Numerical Studies

In the following, we study Landau damping in the linear and non-linear regime and the two-stream instability. Convergence tests are carried out and the CMM is assessed and benchmarked with respect to the pseudo-spectral method and results in the literature.

With this publication, we provide an open-source MATLAB code as well as a C++ CUDA single GPU implementation ¹. The MATLAB repository also includes the implementation

¹Code repository <https://github.com/orgs/CharacteristicMappingMethod/repositories>

of the pseudo-spectral code that we use for the convergence analysis. The MATLAB convergence tests are computed on a Intel 11th Gen i7-11850H 8-core CPU platform.

In addition to the convergence tests, we run high resolution simulations for all three test cases to access the fine-scale features of the solution. The simulations are performed with a CUDA implementation that is run on NVIDIA A100 SXM4 80 GB GPU.

We define experimental order of convergence (EOC) for two consecutive simulations with coarse map $N_c = 2N$ and $N_c = N$ as:

$$\text{EOC}(N) = \frac{\log \frac{\|f - f_{2N}\|}{\|f - f_N\|}}{\log 2} \quad (14)$$

for a given norm $\|\cdot\|$.

5.1 Landau Damping

In this section we study Landau damping initialized with a perturbed Maxwellian distribution:

$$f_0(x, v) = (1 + \epsilon \cos(kx)) \frac{1}{\sqrt{2\pi}} e^{-v^2/2}. \quad (15)$$

All relevant physical parameters are stated in Table 2. Two perturbation regimes are studied:

1. linear Landau damping regime with $\epsilon = 0.05$
2. non-linear Landau damping regime with $\epsilon = 0.5$

name	value
spatial domain	$\Omega_x \in [0, 2\pi/k]$
velocity domain	$\Omega_v \in [-4\pi, 4\pi]$
effective vel. domain	$\Omega_{v^*} \in [-3.8\pi, 3.8\pi]$
wave number k	0.5
ϵ (linear)	5×10^{-2}
ϵ (non linear)	5×10^{-1}

Table 2: Parameters taken from [34] for the Landau damping.

5.1.1 Linear Landau Damping

We initiate our analysis by conducting a convergence assessment of our CMM algorithm, employing the simulation parameters outlined in Table 2.

Convergence Tests. To evaluate the spatial and temporal convergence properties of our proposed numerical algorithm, we vary the discretization parameters detailed in Table 3.

For the spatial convergence evaluations, we conduct simulations with varying spatial resolutions for the characteristic map \mathbf{X} , integrating them in time up to $T = 10$, while keeping all other parameters constant. It is worth noting that we disable remapping during these convergence tests.

Table 3: Grid parameters of the convergence tests.

name	value	name	value
N coarse grid G	32, 64, 128, 256, 512	N grid resolutions G	1024
N_f sampling grid	1024	N_f sampling grid	1024
N_ψ velocity grid	1024	N_ψ velocity grid	1024
Δt time step size	L/1024	Δt time step sizes	$2^{-7}, 2^{-6}, 2^{-5}, 2^{-4}$
N^{ref} ref grid	1024	ref time step size	2^{-8}

(a) Convergence tests in space
(b) Convergence tests in time

Upon reaching the final time point of $T = 10$, we calculate errors as specified in Eqs. (8) to (12) on the fine grid G_f . As a reference, we use the simulation performed with $N = 1024$. This self-comparison is done to assess the error associated with the map \mathbf{X} .

As elaborated in Section 3.4, we anticipate third-order convergence in space for both the L_∞ error associated with the map and the Probability Density Function (PDF), as well as third-order conservation errors. These expectations are substantiated by our numerical experiments, as illustrated in Fig. 3, and the experimental convergence order provided in Table 4.

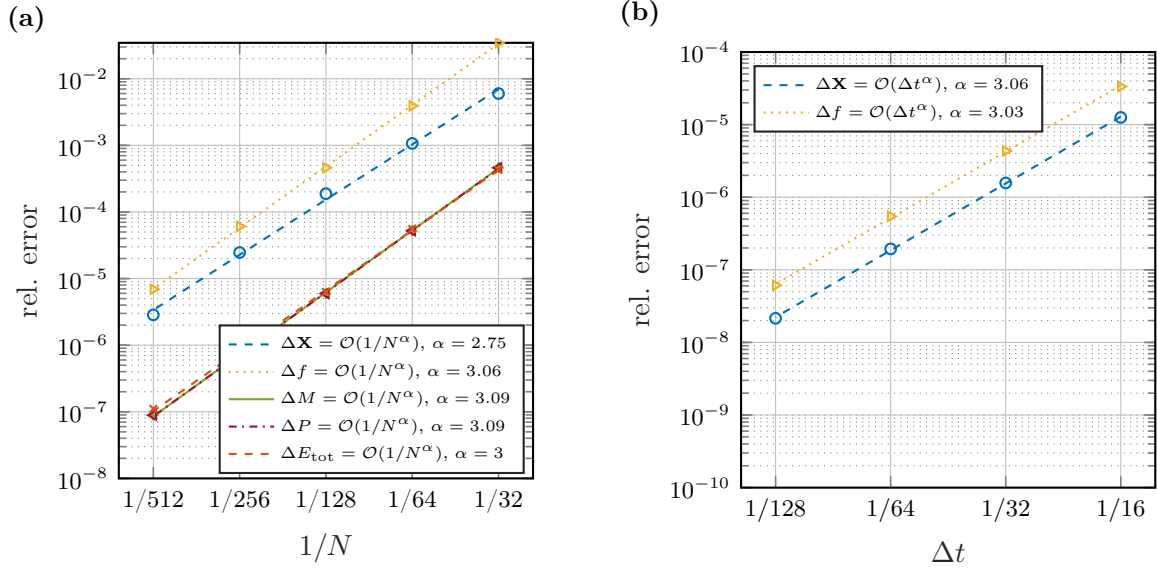


Figure 3: Convergence studies for the linear Landau damping test case at time $t = 10$ with physical parameters listed in Table 2. Shown are the relative errors that correspond to the error measures defined in Eqs. (8) to (12).

(a) Spatial convergence of evolutionary quantities f , \mathbf{X} and the conserved quantities $\Delta \mathcal{M}, \Delta \mathcal{P}, \Delta \mathcal{E}$ using the parameters listed in Table 3a.

(b) Temporal convergence with parameters stated in Table 3b.

Furthermore, we can affirm third-order spatial convergence when comparing the proposed method to a reference solution of a pseudo-spectral computation conducted on a compu-

tational grid with dimensions of 512×512 . The results are succinctly summarized in the final column of Table 4 and visually represented in Fig. 4.

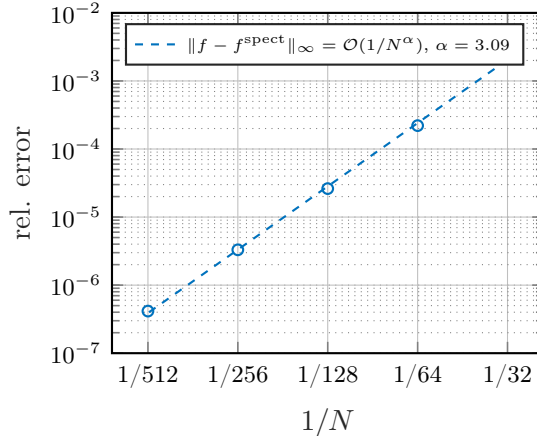


Figure 4: Spatial convergence of the linear Landau damping test case. Relative error as a function of N of CMM with respect to the Fourier–Galerkin reference solution computed at resolution 512^2 .

$1/N$	Δf	$\Delta \mathbf{X}$	ΔM	ΔP	$\Delta \mathcal{E}$	$\ f - f^{\text{spectr}}\ _{\infty}$
1/32	-	-	-	-	-	
1/64	3.1	2.5	3.2	3.1	3.0	3.4
1/128	3.1	2.5	3.1	3.1	3.1	3.1
1/256	2.9	2.9	3.1	3.1	3.0	3.0
1/512	3.1	3.1	3.0	3.0	2.8	3.0

Table 4: Spatial experimental order of convergence EOC as defined in Eq. (14) for the linear Landau damping test case.

Similarly, we observe third-order temporal convergence, as evidenced by the results showcased in Fig. 3b and detailed in Table 5. This assessment of temporal convergence is achieved by comparing simulations conducted with various time steps integrated up to $T = 10$. Importantly, for the temporal convergence examinations, we maintain a consistent spatial resolution as outlined in Table 3b.

Δt	Δf	$\Delta \mathbf{X}$
1/16	-	-
1/32	2.9	3.0
1/64	3.0	3.0
1/128	3.2	3.2

Table 5: Temporal order of convergence EOC as defined in Eq. (14) for the Landau damping test case.

Damping Rate, Sub-grid Resolution and Recurrence Time. A major feature of our method is that the local time stepping is not restricted by the spatial representation of the solution itself. Therefore fine-scale spatial properties can be transported with a larger time step to high accuracy without losing stability. This fact can be exploited to accurately predict the damping rate of linear Landau damping even for highly frequent initial perturbations Eq. (15). One challenge that occurs in regular grid-based, periodic-in- x numerical schemes of Vlasov Poisson is to overcome the *velocity space filamentation problem*. This is a well-known problem in numerical plasma simulations reported since the 1980s [25, 8]. Here, the presence of minor initial spatial perturbations gives rise to oscillations in the velocity distribution, and the frequency of these oscillations increases as time progresses. Consequently, at a certain stage, the velocity grid becomes insufficient to accurately capture these increasingly rapid oscillations. This leads to an aliasing effect, wherein the numerical computation of charge density interprets high-frequency oscillations as if they were of lower frequency. The time at which this artifact comes to play was first discovered 1976 in [8] and is known as the *recurrence time* $\tau_{\text{rec}} = 2\pi/(k\Delta v)$. Mehrenberger et al. showed in [29] that τ_{rec} is linked to the accuracy of the velocity quadrature when computing the charge density in the Poisson equation Eq. (2). This suggests integrating Eq. (2) on a finely resolved grid, which however limits the time step size for conventional numerical schemes.

As we show in Fig. 5, we can overcome this problem when using a high-resolved sampling grid and a coarse map grid. Figure 5 compares the Landau damping for the potential energy at $k = 0.5$ using pseudo-spectral and CMM simulations. For both methods the recurrence time τ_{rec} is clearly visible in Fig. 5a at the time when the exponential damping of the potential energy saturates. With decreasing lattice spacing the recurrence time decreases as expected in the pseudo-spectral code. We note that increasing N in CMM has no effect on the recurrence time. However as expected, when the resolution of the sampling grid is increased, the recurrence time increases. Furthermore in general the recurrence time seems to be larger for the CMM code compared to the spectral code when using the same resolution.

Note that the complexity for solving Eq. (2) is the same in the pseudo-spectral and CMM case if the sampling grid and pseudo-spectral resolution are identical. However, since the time steps are not restricted by the resolution of the sampling grid, the CMM method saves several orders of magnitude when comparing the CPU time to our pseudo-spectral code, shown in Fig. 5b. This is especially visible for increasing $N = N_\psi$. Therefore, the savings are purely contributed to solving the advection part.

The CMM method provides an elegant way of circumventing the filamentation problem for numerical simulations, because of its sub-grid properties. As shown in the radial Fourier spectra in Fig. 6 the spectral content of the solution evaluated at a resolution of 256×256 at time $t = 25$ can be resolved even with a coarse map grid of size 64×64 . Note that the spectrum for the spectral simulation is cut at 85 because of the 2/3 rule.

As a result of the sub-grid property, we can resolve the finest oscillations in the solution, i.e. we can extract damping rates γ and oscillation frequencies ω_r of the Landau damping even for large k . To obtain the damping rates we fit the peaks of the potential energy. We include all peaks after the first peak at $t = 0$ that are above a certain threshold, here $\mathcal{E}_{\text{pot}}(t) \geq 10^{-17}$. From the distance of the peaks, we calculate the mean frequency, which is shown in Fig. 7 together with the damping rate. We compare our results to the

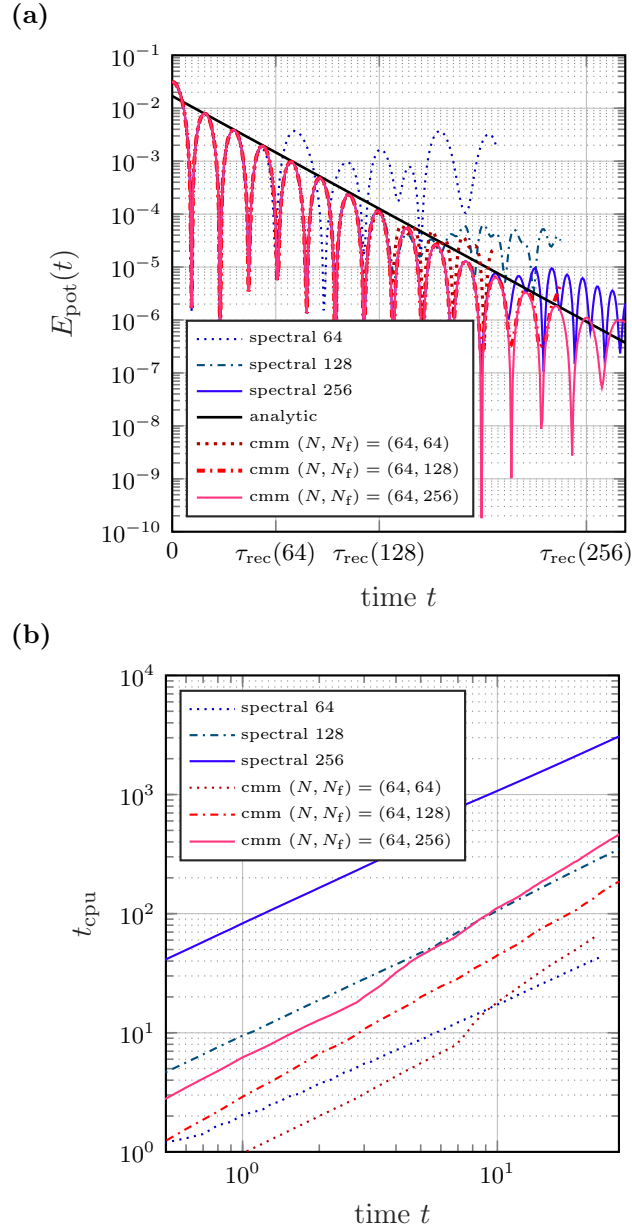


Figure 5: Comparison of the damping of the potential energy for $k = 0.5$. (a). Comparison of CPU-time for CMM and the spectral method for different grids (b).

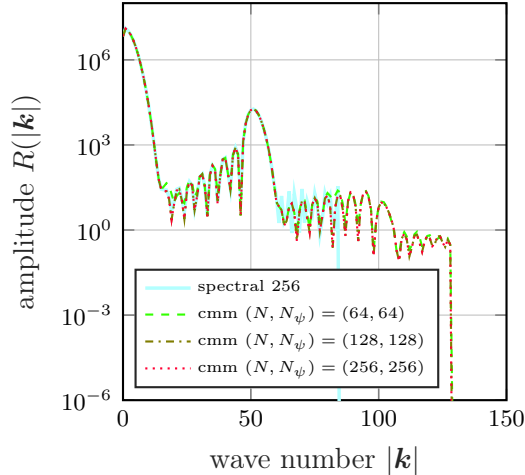


Figure 6: Radial Fourier spectra $R(|\mathbf{k}|) = \int_{|\mathbf{k}|-\Delta k}^{|\mathbf{k}|+\Delta k} |\mathcal{F}[f](\mathbf{k})| d\mathbf{k}$ of the Fourier transformed snapshot $f(x, v, t)$ at time $t = 25$ for the Landau damping test case. The coarsest CMM resolution can recover the finest spectral resolution.

Canosa root finding data [7] as it is reported to yield more accurate results than other approximations [17]. In fact [7] claims an accuracy to the 5th digit. The quantitative results and relative errors compared to [7] are reported in Table 6. As indicated by the relative errors in Table 6, we obtain the most accurate results for large values of k . This error behavior was also observed in [17].

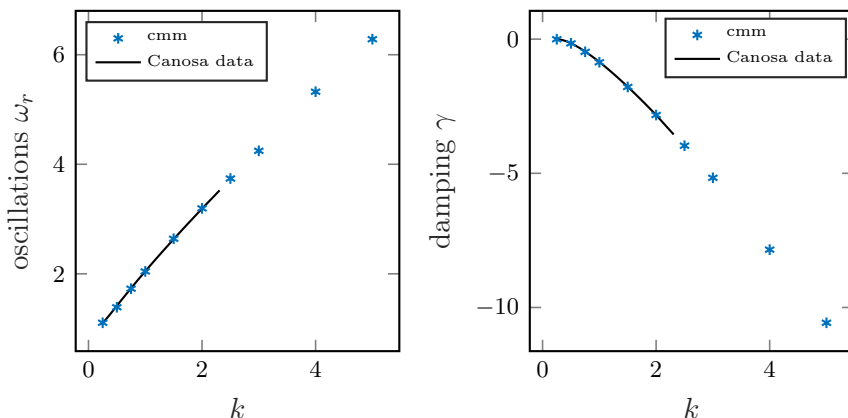


Figure 7: Comparison of the fitted damping rates and oscillation frequencies to the Canosa root finding data [7]. Here we use $(N, N_\psi) = (64, 1024)$, $\delta_{\text{det}} = 0.05$.

Lastly, we show in Fig. 8 that we can resolve the potential energy until machine precision when the sampling grid is increased massively. This computation utilizes the grid dimensions $(N, N_f, N_\psi) = (1024, 4096, 4096)$, and was performed with our CUDA graphics card implementation. The result shows, that we are able to go beyond classical limits given by the recurrence time and filamentation in plasma does not pose a major challenge to our methodology.

k	γ	rel. err	ω_r	rel. err
0.2500	-0.0032	0.4530	1.1062	0.0005
0.5000	-0.1569	0.0233	1.3891	0.0187
0.7500	-0.4726	0.0231	1.7283	0.0051
1.0000	-0.8609	0.0113	2.0433	0.0013
1.5000	-1.7812	0.0031	2.6400	0.0029
2.0000	-2.8274	0.0001	3.1948	0.0018

Table 6: Numerical values of the Landau damping rate γ , oscillation frequency ω_r and their relative error compared to [7] for $(\epsilon, k) = (0.05, 5)$.

name	value
(N, N_f, N_ψ)	(1024, 4096, 4096)
inc. threshold δ_{det}	$5 \cdot 10^{-3}$
stencil distance of GALS	$5 \cdot 10^{-3}$
time step size (CFL) Δt	$1/(N_\psi L_v)$

Table 7: Parameters of the CMM for the fine-scale computations.

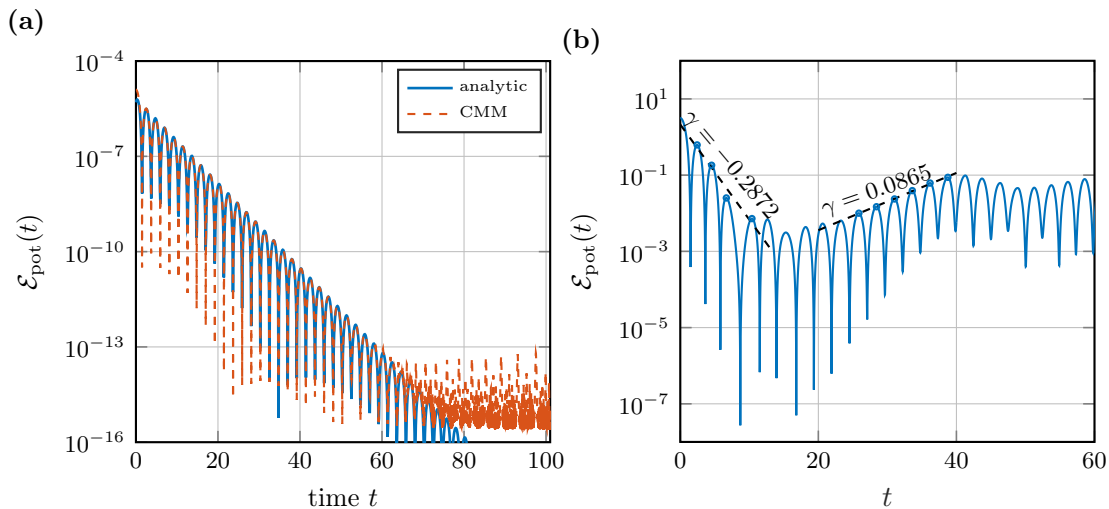


Figure 8: Potential energy of linear ($\epsilon = 0.001$) and non-linear ($\epsilon = 0.5$) Landau damping at $k = 0.5$ using a fine grid computation $(N, N_f, N_\psi) = (1024, 4096, 4096)$.

a) Linear Landau damping resolved down to machine precision without recurrence phenomena.

b) Non-linear Landau damping and the different decay rates. The decay rates are determined by fitting the maxima that are plotted as circles.

5.1.2 Nonlinear Landau Damping

Next, we present simulations of nonlinear Landau damping and compare the results to [33, 20]. For this simulation, we use the same CMM settings listed in Table 7, as they have shown to be free of recurrence time aliasing effects. The solution is initialized as in Eq. (15) and all the physical parameters are set as listed in Table 2.

As the perturbation is increased to $\epsilon = 0.5$ the higher Fourier modes in the electric field get excited. We show the first four Fourier modes that correspond to the wave numbers $k = 0.5, 1, 1.5, 2$ in Fig. 9 sampled till $T = 250$. Confirming the observations of [20, 33] we see decreasing maximal amplitudes of the Fourier modes approximately until $t \approx 15$. Thereafter the amplitudes of all four modes increase until they reach their maxima at $t \approx 40$. The PDF is shown for four different time instances in Fig. 10. We note that by closer examination the general behavior of the four modes is closer to the simulations shown in [33]. From the fits in Fig. 8b we can determine the two characteristic damping values $\gamma_1 = -0.2872$ and $\gamma_2 = 0.0865$ that are in agreement with the values in the literature $\gamma_1 = -0.287$ estimated by [20], $(\gamma_1, \gamma_2) = (-0.281, 0.084)$ estimated by [8].

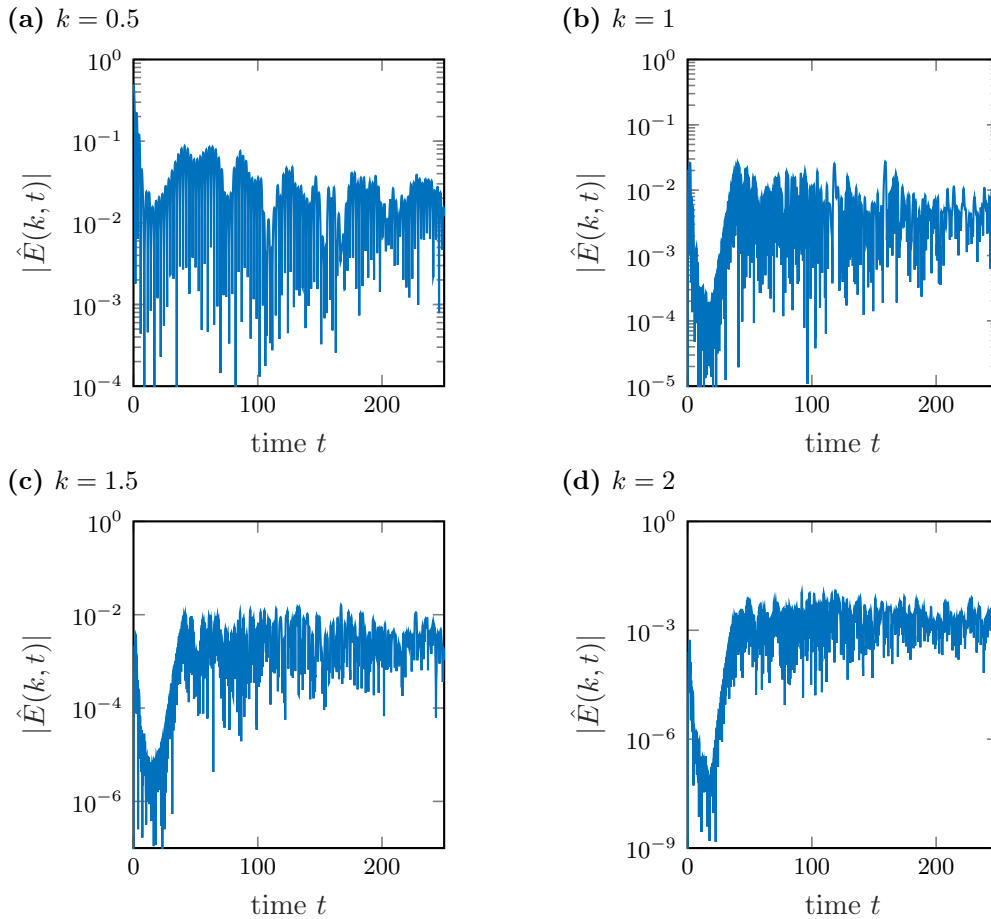


Figure 9: Amplitude $|\hat{E}(k, t)|$ of the first four modes of the electric field of the non-linear Landau damping.

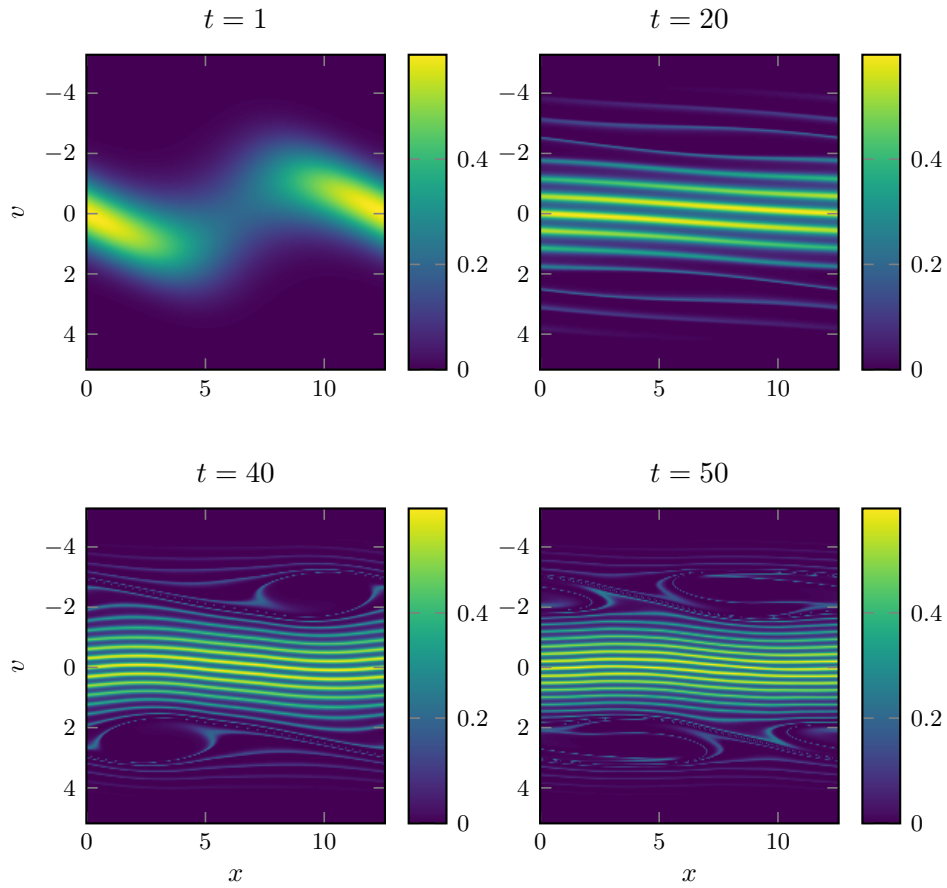


Figure 10: PDF of the non-linear Landau damping for four different time instances. Fine-scale computation with parameters listed in Table 7.

5.2 Two-Stream Instability

The two-stream instability with an initial condition consisting of two counter-propagating cold electron beams initially located at $v_0 = 3$ is considered in the following.

The initial condition is given by

$$f_0(x, v) = (1 + \epsilon \cos(kx)) \frac{1}{2\sqrt{2\pi}} (e^{-(v-v_0)^2/2} + e^{-(v+v_0)^2/2})$$

and the parameters are listed in Table 8.

name	value
spatial domain	$\Omega_x \in [0, 2\pi/k]$
velocity domain	$\Omega_v \in [-5\pi, 5\pi]$
effective vel. domain	$\Omega_{v^*} \in [-4.75\pi, 4.75\pi]$
wave number k	0.2
ϵ	5×10^{-2}
v_0	3
T	80

Table 8: Parameters taken from [34] for the two-stream instability.

Convergence Tests. We first perform basic convergence tests in space and time, similar to the ones in Section 5.1.1. For the convergence in time, we vary the spatial grid size of the map, while keeping all other maps at the same resolution. After $t = 10$ we stop the simulation and compare it to the reference solution. Convergence in time is investigated by comparing a reference solution at $\Delta t = 1/256$ to consecutive decreasing time step sizes. The different grid and time step sizes for our studies are listed in Table 3. The results of the spatial convergence shown in Table 10 as well as the temporal convergence Table 9 indicate that the method is of 3rd order in space and time.

For a sanity check, we also compare the results to the spectral method introduced in Section 4. We see 3rd order convergence when comparing the CMM computations to the reference solution of the spectral method that is computed on a 512×512 grid.

Fine-scale Structures and Zoom Properties. One advantage of the CMM method compared to the pseudo-spectral and other grid-based methods is that fine scales are not advected directly, but carried throughout the map. Thus the fine scales do not need to be

Δt	Δf	$\Delta \mathbf{X}$
$L/16$	-	-
$L/32$	2.8	3.0
$L/64$	2.9	3.0
$L/128$	2.8	2.7

Table 9: Temporal order of convergence EOC as defined in Eq. (14) for the two-stream test case. The reference is CMM with $L/256$.

$1/N$	Δf	$\Delta \mathbf{X}$	ΔM	ΔP	$\Delta \mathcal{E}$	$\ f - f^{\text{spect}}\ _\infty$
1/32	-	-	-	-	-	-
1/64	3.2	1.8	2.9	2.9	2.9	3.1
1/128	2.9	2.0	2.9	2.9	2.9	2.8
1/256	3.1	2.4	3.1	3.1	3.0	2.9
1/512	3.4	2.9	3.3	3.3	2.9	3.0

Table 10: Experimental order of convergence EOC as defined in Eq. (14) for the two-stream instability.

resolved during the advection, allowing for a coarser map. This is illustrated in Fig. 11. The figure shows a time evolution of the PDF for the two-stream instability. It directly compares the pseudo-spectral to the CMM computation in the physical and in the Fourier domain. Over time, small-scale patterns form, due to plasma filamentation, filling up the Fourier spectra. At $t = 40$, all Fourier modes are filled, causing spurious oscillations in the spectral simulation. Instead, CMM allows for an accurate representation even so the fine grid has a full Fourier spectrum because the grid of the sub-maps remains well-resolved as indicated by the fast decay of the Fourier spectra, shown on the right column of Fig. 11.

To showcase the capability of the method we simulated the two-stream instability on a $(N, N_f, N_\psi) = (1024, 4096, 4096)$ configuration up to time $t = 80$ and performed a zoom on the resulting solution, by interpolating the map at the corresponding zoom window and evaluating the initial condition on this window. Figure 12 shows the ability to resolve even the finest structures, using a zoom of factor 2^{10} compared to the initial domain size.

6 Conclusion

We developed the characteristic mapping method for the one+one dimensional Vlasov–Poisson equation. The method is an extension of the previously developed algorithm for the incompressible Euler equations [44, 45]. Differences exist concerning the transport term, which also contains divergence-free velocities in phase space, but these are no longer periodic in all directions. Thus we have used a periodization trick to be able to apply the CMM directly. Numerical analysis of the method showed third-order convergence in space and time. Numerical experiments confirmed our theoretical results for different test cases, including linear and non-linear Landau damping and the two-stream instability. We performed extensive benchmarking and compared the CMM results with those published in the pertinent literature and moreover with a Fourier pseudo-spectral method. Thus the validity of CMM has been shown and its efficiency has been assessed.

The semi-Lagrangian structure of CMM plays a pivotal role in addressing the issue of filamentation, particularly with regard to the temporal discretization induced by the CFL criteria. By carefully adjusting the sampling grid resolution, we have demonstrated that CMM can effectively eliminate filamentation problems. Notably, our investigations have revealed that the method is capable of resolving Landau damping to machine precision, all while remaining immune to recurrence time effects. In comparison to other semi-Lagrangian methods, CMM stands out for its inherent bounds preservation. Unlike competing methods that often require additional measures such as positivity limiters or filters, CMM excels

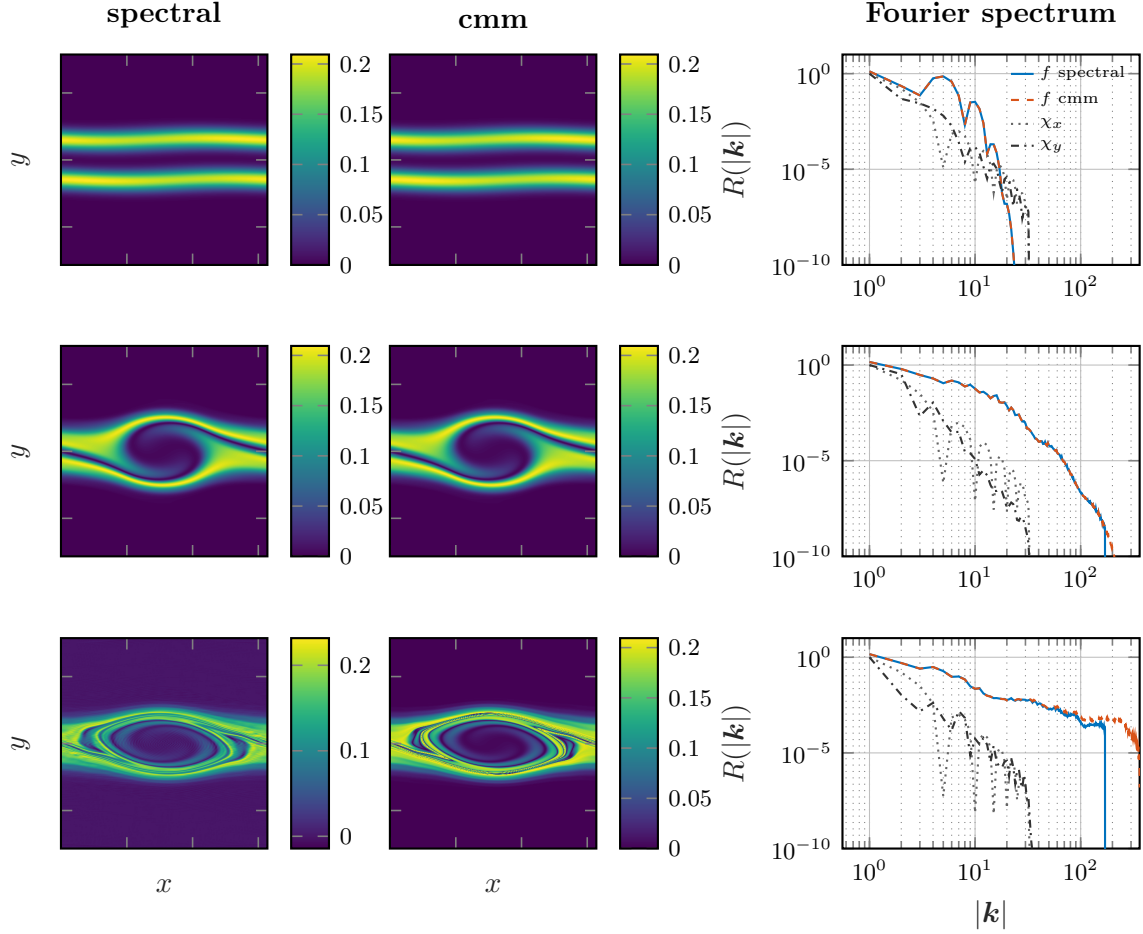


Figure 11: Time evolution of f for the two-stream instability, at $t = 1, 15, 40$. Resolution: spectral 512, CMM $N = 64, N_f = N_\psi = 512$, right: Radial Fourier spectra $R(|\mathbf{k}|) = \int_{|\mathbf{k}|-\Delta k}^{|\mathbf{k}|+\Delta k} |\mathcal{F}[g](\mathbf{k})| d\mathbf{k}$, where g is either the snapshot of the current submap $\chi_{x/y}$ in x/y direction or the PDF.

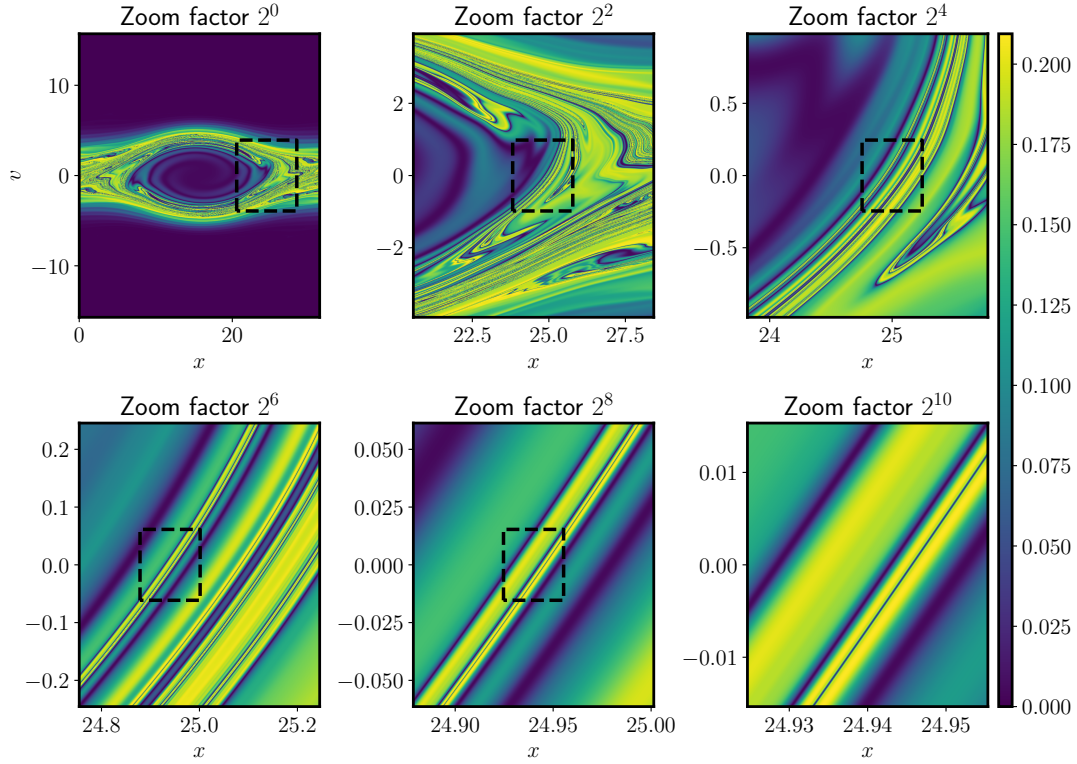


Figure 12: Two-stream instability computed with CMM, $N = 1024$, $N_f = N_\psi = 4096$. Successive zooms of the distribution function f at $t = 80$ to illustrate the multitude of resolved scales. The region inside the dotted black square is successively amplified by a factor 4 in each direction up to a factor of 2^{10} .

in preserving the integrity of physical bounds without the need for such supplements [48, 49, 9]. CMM’s compositional structure is another remarkable aspect of the method. It enables us to resolve fine-scale structures at varying resolutions through the application of zooms, a unique feature not shared by any other semi-Lagrangian approach. However, it is worth noting that this advantage comes with the requirement of storing sub-maps and executing compositions during runtime. Our implementation of CMM also demonstrates computational efficiency, yielding time savings of one order of magnitude when compared to a pseudo-spectral implementation of the method. These findings underscore the practical benefits of adopting CMM for numerical simulations. Furthermore, our research indicates that CMM exhibits excellent conservation properties with third-order convergence. While the method showcases strong conservation characteristics, full numerical conservation of mass, energy, and momentum is a subject of future work. Furthermore, future work will address VP using CMM in higher dimensions.

Author Contribution Statement (CRediT)

In the following, we declare the author’s contributions to the publication:

Philipp Krah:	implementation MATLAB & C++/CUDA code, writing initial draft, numerical studies, visualization
Xi-Yuan Yin:	implementation MATLAB, numerical analysis, boundary periodification, writing initial draft, reviewing & editing
Julius Bergmann:	implementation C++/CUDA code, reviewing & editing
Jean-Christophe Nave:	initial idea, editing and supervision
Kai Schneider:	initial idea, writing the initial draft, editing and supervision, funding acquisition, project administration

Conflict of Interest

The authors declare that they have no conflict of interest.

Code and Data Availability

All scripts to reproduce the results are available at:

<https://github.com/orgs/CharacteristicMappingMethod/repositories>.

Acknowledgement

The authors were granted access to the HPC resources of IDRIS under allocation No. AD012A01664R1 attributed by Grand Équipement National de Calcul Intensif (GENCI). Centre de Calcul Intensif d’Aix-Marseille is acknowledged for granting access to its high-performance computing resources. The authors acknowledge partial funding from the Agence Nationale de la Recherche (ANR), project CM2E, grant ANR-20-CE46-0010-01. KS acknowledges financial support from the French Federation for Magnetic Fusion Studies (FR-FCM) and the Eurofusion consortium, funded by the Euratom Research and Training Programme under Grant Agreement No.633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

References

- [1] V. I. Arnold. *Mathematical methods of classical mechanics*, volume 60. Springer Science & Business Media, 2013. 5
- [2] C. Bardos and E. Tadmor. Stability and spectral convergence of Fourier method for nonlinear problems: on the shortcomings of the 2/3 de-aliasing method. *Numerische Mathematik*, 129:749–782, 2015. 11
- [3] N. Besse and M. Mehrenberger. Convergence of classes of high-order semi-Lagrangian schemes for the Vlasov–Poisson system. *Mathematics of Computation*, 77(261):93–123, 2008. 2
- [4] N. Besse and E. Sonnendrücker. Semi-Lagrangian schemes for the Vlasov equation on an unstructured mesh of phase space. *Journal of Computational Physics*, 191(2):341–376, 2003. 2
- [5] C. K. Birdsall and A. B. Langdon. *Plasma physics via computer simulation*. CRC press, 2018. 2
- [6] Z. Cai, R. Li, and Y. Wang. Solving Vlasov equations using NRxx method. *SIAM Journal on Scientific Computing*, 35(6):A2807–A2831, 2013. 2
- [7] J. Canosa. Numerical solution of Landau’s dispersion equation. *Journal of Computational Physics*, 13(1):158–160, 1973. 17, 18
- [8] C.-Z. Cheng and G. Knorr. The integration of the Vlasov equation in configuration space. *Journal of Computational Physics*, 22(3):330–351, 1976. 2, 15, 19
- [9] N. Crouseilles, M. Mehrenberger, and E. Sonnendrücker. Conservative semi-Lagrangian schemes for Vlasov equations. *Journal of Computational Physics*, 229(6):1927–1953, 2010. 2, 25
- [10] N. Crouseilles, M. Mehrenberger, and F. Vecil. Discontinuous Galerkin semi-Lagrangian method for Vlasov-Poisson. In *ESAIM: Proceedings*, volume 32, pages 211–230. EDP Sciences, 2011. 2
- [11] N. Crouseilles, T. Respaud, and E. Sonnendrücker. A forward semi-Lagrangian method for the numerical solution of the Vlasov equation. *Computer Physics Communications*, 180(10):1730–1745, 2009. 2
- [12] G. Dimarco and L. Pareschi. Numerical methods for kinetic equations. *Acta Numerica*, pages 369–520, 2014. 2
- [13] L. Einkemmer. A performance comparison of semi-Lagrangian discontinuous Galerkin and spline based Vlasov solvers in four dimensions. *Journal of Computational Physics*, 376:937–951, 2019. 2
- [14] L. Einkemmer and A. Ostermann. Convergence analysis of a discontinuous Galerkin/Strang splitting approximation for the Vlasov–Poisson equations. *SIAM Journal on Numerical Analysis*, 52(2):757–778, 2014. 2

- [15] F. Filbet and E. Sonnendrücker. Comparison of Eulerian Vlasov solvers. *Computer Physics Communications*, 150(3):247–266, 2003. 2
- [16] F. Filbet and T. Xiong. Conservative discontinuous Galerkin/Hermite spectral method for the Vlasov–Poisson system. *Communications on Applied Mathematics and Computation*, pages 1–26, 2020. 2
- [17] D. S. Finn, M. G. Knepley, J. V. Pusztaý, and M. F. Adams. A numerical study of Landau damping with PETSC-PIC. *Communications in Applied Mathematics and Computational Science*, 18(1):135–152, 2023. 17
- [18] J. Fröhlich and K. Schneider. An adaptive wavelet Galerkin algorithm for one and two dimensional flame computations. *European Journal of Mechanics, B/Fluids*, 13:439–471, 1994. 10
- [19] S. Gassama, E. Sonnendrücker, K. Schneider, M. Farge, and M. O. Domingues. Wavelet denoising for postprocessing of a 2d particle-in-cell code. In *ESAIM: Proceedings*, volume 16, pages 195–210. EDP Sciences, 2007. 2
- [20] R. E. Heath, I. M. Gamba, P. J. Morrison, and C. Michler. A discontinuous Galerkin method for the Vlasov–Poisson system. *Journal of Computational Physics*, 231(4):1140–1174, 2012. 19
- [21] R. W. Hockney and J. W. Eastwood. *Computer simulation using particles*. CRC Press, 2021. 2
- [22] M. Y. Hussaini and T. A. Zang. Spectral methods in fluid dynamics. *Annual Review of Fluid Mechanics*, 19(1):339–367, 1987. 10, 11
- [23] S. G. Johnson. Saddle-point integration of c_∞ "bump" functions. *arXiv preprint arXiv:1508.04376*, 2015. 30
- [24] M. Kirchhart and R.-P. Wilhelm. The numerical flow iteration for the Vlasov-Poisson equation. *arXiv preprint arXiv:2301.05581*, 2023. 3
- [25] A. J. Klimas. A method for overcoming the velocity space filamentation problem in collisionless plasma model solutions. *Journal of computational physics*, 68(1):202–226, 1987. 2, 15
- [26] A. J. Klimas and W. M. Farrell. A splitting algorithm for Vlasov simulation with filamentation filtration. *Journal of computational physics*, 110(1):150–163, 1994. 2
- [27] H. Kohno and J.-C. Nave. A new method for the level set equation using a hierarchical-gradient truncation and remapping technique. *Computer Physics Communications*, 184(6):1547–1554, 2013. 3, 4
- [28] A. Mangeney, F. Califano, C. Cavazzoni, and P. Travnicek. A numerical scheme for the integration of the Vlasov–Maxwell system of equations. *Journal of Computational Physics*, 179(2):495–538, 2002. 2
- [29] M. Mehrenberger, L. Navoret, and N. Pham. Recurrence phenomenon for Vlasov-Poisson simulations on regular finite element mesh. *Communications in Computational Physics*, 2020. 15

- [30] O. Mercier, X.-Y. Yin, and J.-C. Nave. The characteristic mapping method for the linear advection of arbitrary sets. *SIAM Journal on Scientific Computing*, 42(3):A1663–A1685, 2020. [3](#), [4](#)
- [31] J.-C. Nave, R. R. Rosales, and B. Seibold. A gradient-augmented level set method with an optimally local, coherent advection scheme. *Journal of Computational Physics*, 229(10):3802–3827, 2010. [3](#)
- [32] M. Palmroth, U. Ganse, Y. Pfau-Kempf, M. Battarbee, L. Turc, T. Brito, M. Grandin, S. Hoilijoki, A. Sandroos, and S. von Althaus. Vlasov methods in space physics and astrophysics. *Living Reviews in Computational Astrophysics*, 4(1):1, 2018. [1](#)
- [33] J. T. Parker and P. J. Dellar. Fourier-Hermite spectral representation for the Vlasov-Poisson system in the weakly collisional limit. *Journal of Plasma Physics*, 81(2):305810203, Apr. 2015. arXiv:1407.1932 [physics]. [2](#), [19](#)
- [34] N. Pham, P. Helluy, and A. Crestetto. Space-only hyperbolic approximation of the Vlasov equation. *ESAIM: Proceedings*, 43, Dec. 2013. [12](#), [21](#)
- [35] M. J. Picklo, Q. Tang, Y. Zhang, J. K. Ryan, and X.-Z. Tang. Denoising particle-in-cell data via smoothness-increasing accuracy-conserving filters with application to bohm speed computation. *Journal of Computational Physics*, 502:112790, 2024. [2](#)
- [36] J.-M. Qiu and C.-W. Shu. Positivity preserving semi-Lagrangian discontinuous Galerkin formulation: theoretical analysis and application to the Vlasov–Poisson system. *Journal of Computational Physics*, 230(23):8386–8409, 2011. [2](#)
- [37] J. A. Rossmannith and D. C. Seal. A positivity-preserving high-order semi-Lagrangian discontinuous Galerkin scheme for the Vlasov–Poisson equations. *Journal of Computational Physics*, 230(16):6203–6232, 2011. [2](#)
- [38] E. Sonnendrücker, J. Roche, P. Bertrand, and A. Ghizzo. The semi-Lagrangian method for the numerical resolution of the Vlasov equation. *Journal of Computational Physics*, 149(2):201–220, 1999. [2](#)
- [39] S. Taylor and J.-C. Nave. A characteristic mapping method for incompressible hydrodynamics on a rotating sphere. *arXiv preprint arXiv:2302.01205*, 2023. [3](#)
- [40] S. Taylor and J.-C. Nave. A projection-based characteristic mapping method for tracer transport on the sphere. *Journal of Computational Physics*, 477:111905, 2023. [3](#)
- [41] R. van yen Nguyen, D. del Castillo-Negrete, K. Schneider, M. Farge, and G. Chen. Wavelet-based density estimation for noise reduction in plasma simulations using particles. *Journal of Computational Physics*, 229(8):2821–2839. [2](#)
- [42] J. P. Verboncoeur. Particle simulation of plasmas: review and advances. *Plasma Physics and Controlled Fusion*, 47(5A):A231, 2005. [2](#)
- [43] X.-Y. Yin. *Characteristic mapping framework with applications to density transport and fluid dynamics*. PhD thesis, McGill University (Canada), 2021. [8](#)
- [44] X.-Y. Yin, O. Mercier, B. Yadav, K. Schneider, and J.-C. Nave. A characteristic mapping method for the two-dimensional incompressible Euler equations. *Journal of Computational Physics*, 424:109781, 2021. [3](#), [8](#), [9](#), [22](#)

- [45] X.-Y. Yin, K. Schneider, and J.-C. Nave. A characteristic mapping method for the three-dimensional incompressible euler equations. *Journal of Computational Physics*, 477:111876, 2023. [3](#), [9](#), [22](#)
- [46] S. Zaki, T. Boyd, and L. Gardner. A finite element code for the simulation of one-dimensional Vlasov plasmas. II. Applications. *Journal of Computational Physics*, 79(1):200–208, 1988. [2](#)
- [47] S. Zaki, L. Gardner, and T. Boyd. A finite element code for the simulation of one-dimensional Vlasov plasmas. I. Theory. *Journal of Computational Physics*, 79(1):184–199, 1988. [2](#)
- [48] M. Zerroukat, N. Wood, and A. Staniforth. A monotonic and positive-definite filter for a semi-Lagrangian inherently conserving and efficient (slice) scheme. *Quarterly Journal of the Royal Meteorological Society: A journal of the atmospheric sciences, applied meteorology and physical oceanography*, 131(611):2923–2936, 2005. [25](#)
- [49] M. Zerroukat, N. Wood, and A. Staniforth. The parabolic spline method (PSM) for conservative transport problems. *International journal for numerical methods in fluids*, 51(11):1297–1318, 2006. [25](#)

Appendix

A Periodic Velocity Extension

We want to define a periodic velocity field $u_1(v)$ on the velocity interval $v \in [-L_v, L_v] := \Omega_v$ such that on the support of f , with $\text{supp}(f) \subset \Omega_{v^*} := (-v^*, v^*) \subset \Omega_v$, we have $u_1(v) = v$. It follows that the derivative $\partial_v u_1$, the u_1 contribution to vorticity must be an approximation of the box function

$$B(v) = 1 - \frac{L_v}{L_v - v^*} \mathbf{1}_{|v| \in [v^*, L_v]},$$

such that the integral of B vanishes in order to satisfy the periodicity condition on u_1 . In doing so, we add a padding region $[v^*, L_v]$ corresponding to a horizontal jet in order to make u_1 vanish smoothly at $v = L_v$. The length of the extension, $a = L_v - v^*$, can be seen as the amount of wasted computational resource and should be minimized subject to error tolerances. The function B is only a generic example of a periodic extension, smoother extensions exist with the only requirement being that $B(v) \equiv 1$ in Ω_{v^*} and the integral of B is 0. We note that due to the requirement that B is constant in an open interval, analytic functions are out of question and the best extensions we can hope for are in the C^∞ class. The regularity of the extension will affect the decay of the coefficients of the Fourier transform and thereby the numerical error introduced. The general optimization problem is as follows: Given a family of extension $B(v; a)$ parametrized by a (v^* given, thereby also parametrized by L_v), for each computational grid size N , we minimize the total numerical error

$$\mathcal{E}(N, a) = \mathcal{D} \left(\frac{N}{v^* + a} \right) + \mathcal{C}(a, N),$$

where \mathcal{D} is the spatial error of the numerical scheme which depends on the Nyquist frequency N/L and \mathcal{C} is the numerical error introduced by the velocity modification which we assume to be on the order of the $k^{-1} \hat{B}(N; a)$, the size of the first truncated term in terms of the Fourier transform. We note that this is an over-estimation of $\mathcal{C}(a, N)$, indeed, assuming $B \equiv 1$ in Ω_{v^*} the extension only introduces error outside the support of f which then propagates towards the support itself through successive time stepping.

We consider the simple extension using a bump function:

$$B(v; a) = 1 - \frac{L}{a^2} \eta \left(\frac{|x| - L_v}{a} \right),$$

where η is the standard mollifier $\eta(x) = \exp\left(\frac{-1}{1-x^2}\right) / I$ which satisfies $\text{supp}(\eta) = (-1, 1)$ and $\int_{-1}^1 \eta dx = 1$. The technical note [23], gave a very accurate asymptotic decay rate of $L_v a^{-7/4} k^{-3/4} \exp(-\sqrt{ak})$ for these mollifiers, from which we derive an estimate for the error introduced by the extension:

$$\mathcal{C}(a, N) \propto L_v \left(\frac{aN}{v^* + a} \right)^{-7/4} \exp \left(-\sqrt{\frac{aN}{v^* + a}} \right).$$

For a Hermite cubic implementation of the characteristic mapping method, the local spatial truncation error is 4th order, i.e. assuming $\mathcal{D} \propto \left(\frac{N}{a+v^*}\right)^{-4}$, we obtain the following

approximate optimality condition for a for a given fixed N :

$$0 = \frac{4(a + v^*)^3}{N^4} - (N_a)^{-\frac{7}{4}} \exp\left(-\sqrt{N_a}\right) \left(\frac{v^*}{4a} \left(2\sqrt{N_a} + 7\right) - 1\right), \quad (16)$$

where $N_a = \frac{aN}{a+v^*}$ is the number of wasted grid points. We solve this numerically and show the size of the optimized extension and its computational efficiency $\frac{v^*}{a+v^*}$ in Fig. 13.

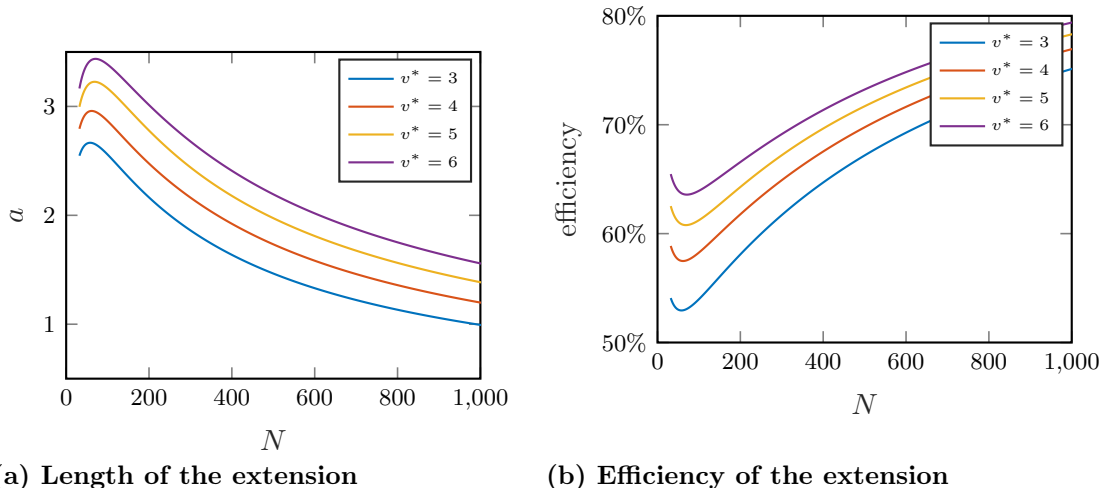


Figure 13: Choice of extension region optimized according to Eq. (16). The efficiency is defined as the ratio of the number of grid points in the support against the total number of grid points.

B Statistics of Fine-Scale Computations

For the linear Landau damping, non-linear Landau damping and two-stream instability test case, we have performed fine scale CMM computations with algorithmic parameters listed in Table 7 and physical parameters in Tables 2 and 8. Results have been shown for the linear and non-linear Landau damping in Figs. 8 to 10 and the zooms of the two-stream instability in Fig. 12. In Fig. 14 we show the time history of the conserved quantities $\mathcal{M}, \mathcal{E}, \mathcal{P}$ and additional quantities like the compressibility error, the number of sub-maps utilized during the computations and the overall GPU-time. For all fine-scale simulations, we use the same incompressibility threshold $\delta_{\text{det}} = 0.005$. However, we include one fine-scale simulation of the two-stream instability using $\delta_{\text{det}} = 0.001$ to visualize the difference in performance. Figure 14 shows that all conserved quantities grow until a saturated state. Remappings affect the simulation results as can be seen from the direct comparison of the two-stream instability computations. After the first remapping of the simulation with $\delta_{\text{det}} = 0.001$ at $t \approx 7$, the results compared to the simulation with $\delta_{\text{det}} = 0.005$ differ. However, the higher number of remappings in the case of $\delta_{\text{det}} = 0.001$ does not imply a significant rise in GPU time. Nevertheless, it leads to higher consumption in memory, which is the limiting factor when choosing small values of δ_{det} .

Additionally to the CMM computations, we show the statistics of the pseudo-spectral computations in Fig. 15. These are the statistics of the simulation shown in Fig. 11. Note that the total mass is conserved up to machine precision. L_2 -norm of f is preserved until the first oscillations in the simulation occur.

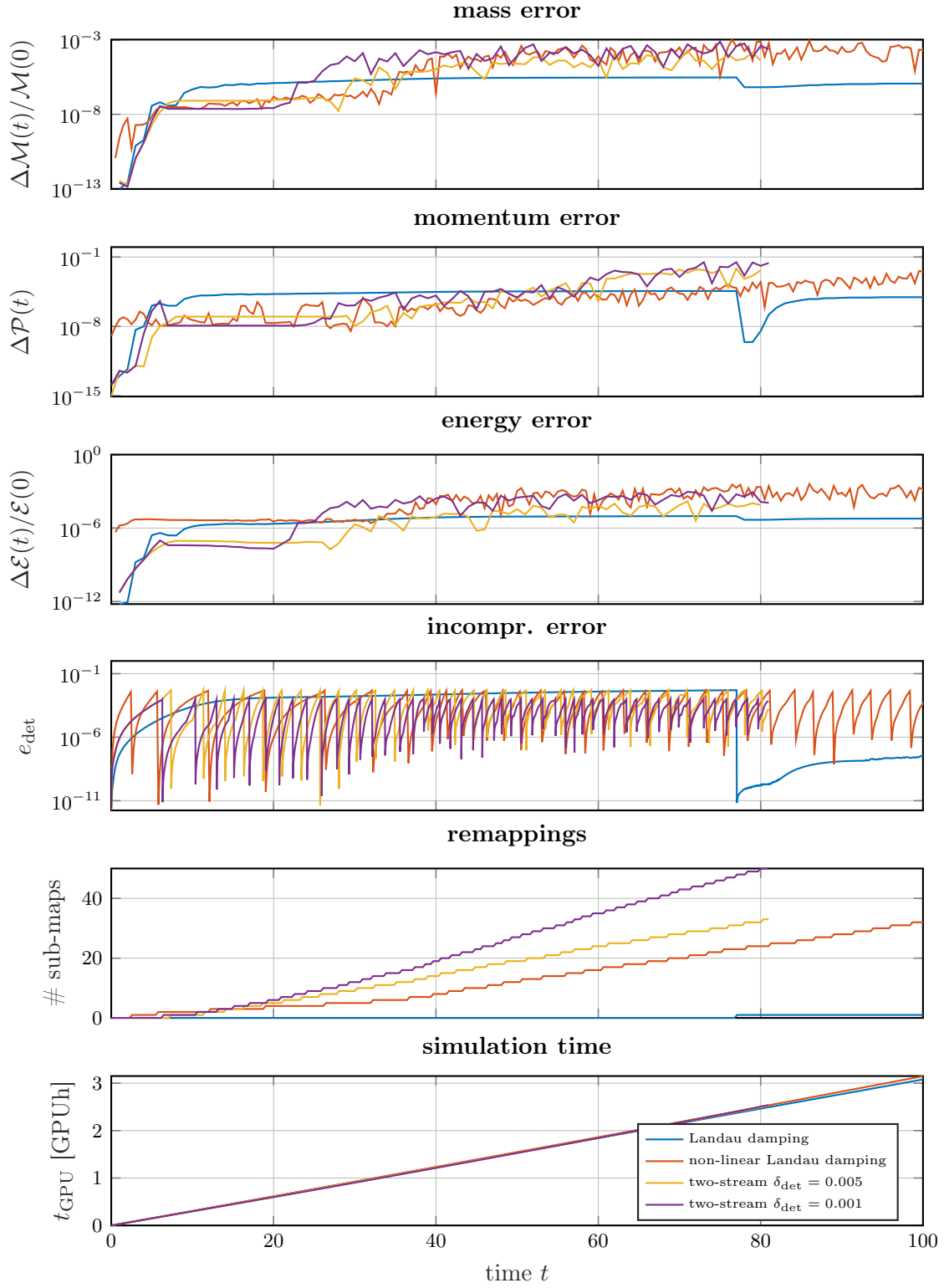


Figure 14: Statistical quantities of the fine resolved CMM computations: Landau damping, non-linear Landau damping and the two-stream instability. The legend is given in the lowest figure. The simulation parameters are listed in Table 7. Shown are (from top to bottom) relative mass error, momentum error, relative energy error, incompressibility error of the map, number of sub-maps, GPU time in hours on a NVIDIA A100 SXM4 80 GB GPU.

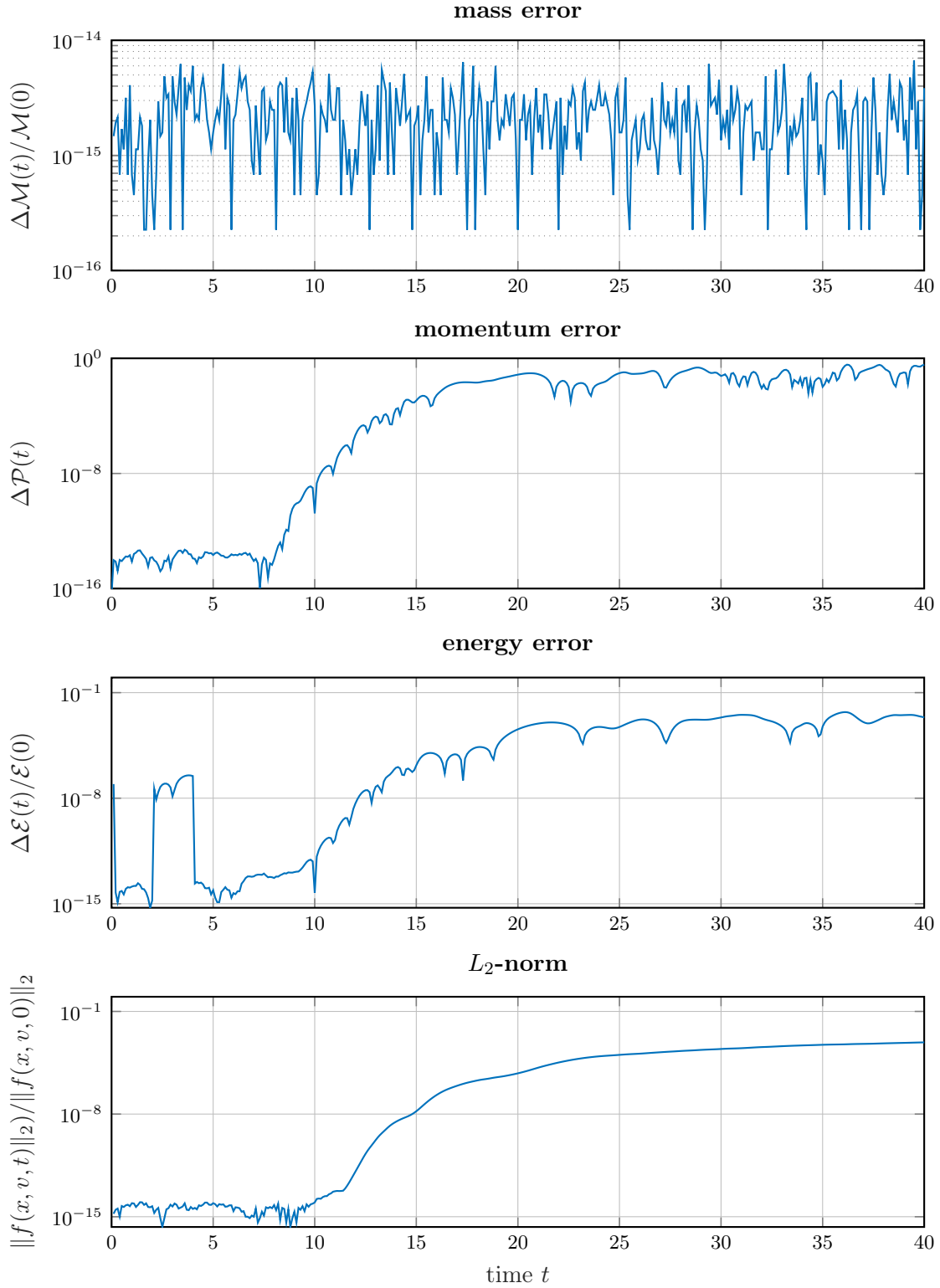


Figure 15: Statistics of spectral computation of the two-stream instability at $N \times N = 512 \times 512$. Shown are (from top to bottom) relative mass error, momentum error, relative energy error and L_2 -error of f .