



HAL
open science

A δf PIC method with forward–backward Lagrangian reconstructions

M. Campos Pinto, M. Pelz, P.-H. Tournier

► **To cite this version:**

M. Campos Pinto, M. Pelz, P.-H. Tournier. A δf PIC method with forward–backward Lagrangian reconstructions. *Physics of Plasmas*, 2023, 30 (3), pp.033902. 10.1063/5.0129989 . hal-04041230

HAL Id: hal-04041230

<https://hal.science/hal-04041230v1>

Submitted on 6 Sep 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.



Distributed under a Creative Commons Attribution 4.0 International License

RESEARCH ARTICLE | MARCH 10 2023

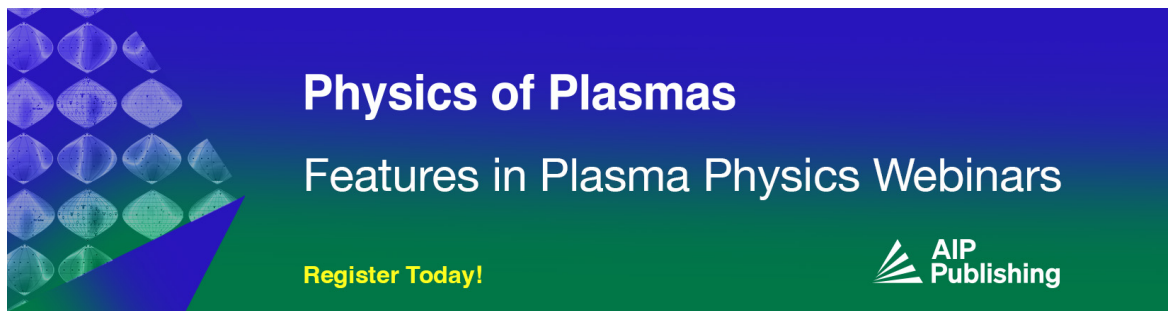
A δf PIC method with forward–backward Lagrangian reconstructions

M. Campos Pinto   ; M. Pelz  ; P.-H. Tournier 




Phys. Plasmas 30, 033902 (2023)

<https://doi.org/10.1063/5.0129989>



Physics of Plasmas
Features in Plasma Physics Webinars

Register Today!



A δf PIC method with forward-backward Lagrangian reconstructions

Cite as: Phys. Plasmas **30**, 033902 (2023); doi: 10.1063/5.0129989

Submitted: 8 October 2022 · Accepted: 15 February 2023 ·

Published Online: 10 March 2023



View Online



Export Citation



CrossMark

M. Campos Pinto,^{1,a)}  M. Pelz,²  and P.-H. Tournier³ 

AFFILIATIONS

¹Max-Planck-Institut für Plasmaphysik, Boltzmannstraße 2, D-85748 Garching b. München, Germany

²Department of Mathematics, University of British Columbia, Vancouver, British Columbia V6T 1Z2, Canada

³Sorbonne Université, Université Paris-Diderot SPC, CNRS, Inria, Laboratoire Jacques-Louis Lions, équipe Alpines, F-75005 Paris, France

^{a)} Author to whom correspondence should be addressed: martin.campos-pinto@ipp.mpg.de

ABSTRACT

In this work, we describe a δf particle simulation method where the bulk density is periodically remapped on a coarse spline grid using a forward-backward Lagrangian approach. This method is designed to handle plasma regimes where the densities strongly deviate from their initial state and may evolve into general profiles. We describe the method in the case of an electrostatic particle-in-cell scheme and validate its qualitative properties using a classical two-stream instability subject to a uniform oscillating drive.

© 2023 Author(s). All article content, except where otherwise noted, is licensed under a Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>). <https://doi.org/10.1063/5.0129989>

I. INTRODUCTION

In order to reduce the statistical noise in numerical simulations of kinetic plasma problems, particle-in-cell (PIC) methods often follow a so-called δf approach,^{1–3} which consists of decomposing the transported density into two parts, a bulk density f_0 given by an analytical formula and a variation δf represented with numerical particles. In Ref. 4, this approach was revisited as a variance reduction method in the scope of Monte Carlo algorithms, with f_0 playing the role of a control variate, and since then, several techniques have been devised to improve the reduction of statistical error, in particular for gyrokinetic simulation models^{5–8} and collisional models.^{9–11}

In many practical problems, the plasma either remains close to an equilibrium state^{5,12} or evolves as a small perturbation of some analytically known background,^{13–15} which can be used as a bulk density. In some cases, however, the plasma evolves in an unpredictable way and f_0 needs to be updated by a self-consistent algorithm to better follow the total density.

Typical examples are problems where full- f simulations are needed. In the physics of tokamak plasmas, one instance is the modeling of $E \times B$ staircases,¹⁶ which are long-lived patterns of quasiregular step-like profiles that develop spontaneously in turbulent plasmas. As they slowly move in the radial direction over large time scales, the separation assumption between an analytical background and fluctuations

is no longer valid after some time and simulations need to involve full- f methods such as the semi-Lagrangian scheme used in the GYSELA code.¹⁷ Another example is the modeling of the tokamak edge region where the plasma density may strongly deviate from local Maxwellian distributions with large and intermittent fluctuations. This has motivated the development of various Eulerian full- f schemes such as those presented in Refs. 18 and 19, where large plasma blob structures can be seen propagating from the core region toward the tokamak edge.

If one desires to model such problems with a δf PIC method, it is, thus, necessary to allow for general updates of the bulk density over time. An interesting approach in this direction was proposed in Ref. 20: It consisted of projecting the particle density δf on a coarse spline basis and adding the resulting smoothed distribution to the bulk density.

In this article, we consider a variant of this approach where the bulk density is updated using a semi-Lagrangian approach based on the forward-backward Lagrangian (FBL) method.²¹

Inspired by Ref. 22, the core of the FBL method is to compute backward trajectories on arbitrary nodes by local inversions of the particle trajectories: As these describe the forward transport flow in phase space and are naturally provided by the PIC code, their local inversion allows to perform semi-Lagrangian updates of a smooth density represented on a coarse grid. In particular, the novelty of our approach is that it does not primarily rely on an accurate particle approximation

of the density itself but rather rely on an accurate description of the particle trajectories. As these are, in general, much less noisy than the phase space density, we believe that this new paradigm can lead to efficient low-noise particle methods.

The outline is as follows: In Sec. II, we present our general ansatz for the discrete density, which may be seen as a hybrid discretization between particle and semi-Lagrangian density representations. In Sec. III, we recall the key steps of electrostatic particle-in-cell approximations, and in Sec. IV, we describe the δf PIC method with FBL remappings of the bulk density. The proposed method is summarized in Sec. V, and in Sec. VI, we present a series of numerical results involving two-stream instability (TSI) test cases to illustrate the enhanced denoising properties of our approach. We conclude in Sec. VII with a summary of the proposed method, a discussion on its novelty, and two perspectives for future research.

II. APPROXIMATION ANSATZ

Our ansatz for the general density at a discrete time t^n is a sum of two terms

$$f^n := f_*^n + \delta f^n \tag{1}$$

(we shall often use $:=$ to highlight a definition), where the first one will be seen as the bulk density, a smooth approximation to the full solution, and the second one as the fine scale variations. Following the general principle of δf methods, we require f_*^n to have a simple expression that is easy to evaluate at arbitrary positions in a general d -dimensional phase space, and we represent the variation δf^n as an unstructured collection of numerical particles with coordinates z_k^n and weights δw_k^n ,

$$\delta f^n(z) := \sum_{k=1}^{N_p} \delta w_k^n \varphi_\varepsilon(z - z_k^n). \tag{2}$$

Here, φ_ε is a smooth shape function of scale ε , and $z = (z_1, \dots, z_d)$ is a phase-space coordinate. In typical problems where the transported density slightly deviates from an initial profile, the bulk density is often set to this initial value, $f_*^n = f^0$, or to some analytical equilibrium.^{5,12} In this note, we investigate an alternate approach where the bulk density is represented as an arbitrary collection of B-splines on a coarse grid with the mesh size $h_* > \varepsilon$, in the spirit of Ref. 20. This yields an expression that is formally similar to that of δf^n ,

$$f_*^n(z) = \sum_{j \in \mathbb{Z}^d} w_{*,j}^n \varphi_{h_*}(z - jh_*), \tag{3}$$

where $\varphi_\varepsilon(\cdot - jh_*)$ is now the coarse B-spline shape centered on a general d -dimensional grid node jh_* (\mathbb{Z} is the set of integers), and $w_{*,j}^n$ is its weight. In practice, only a finite number N_* of such nodes is used, and as outlined in the introduction, we will periodically update the coefficients of this spline bulk density using forward-backward Lagrangian (FBL) reconstructions, which involves a relatively small set of passive markers designed to track the characteristic flow in phase space.

A. Main numerical parameters and limit regimes

The main numerical parameters are as follows:

- N_r is the remapping period, i.e., the number of time steps between two updates of the bulk density. The limit value

$N_r = \infty$ corresponds to a frozen bulk density, namely, $f_*^n = f_*^0$ for every time step n .

- N_* is the number of coarse splines used to represent the bulk density in the computational domain $\Omega \subset \mathbb{R}^d$, and it is on the order of $\text{Vol}(\Omega)(h_*)^{-d}$. The limit value of $N_* = 0$ (an empty grid) corresponds to a “full- f ” particle approximation.
- N_p is the number of numerical particles describing the fine scale structures. The limit value of $N_p = 0$ corresponds to a semi-Lagrangian ansatz²³ where the full density is represented on a structured grid and can be updated in time with an ad hoc scheme such as the FBL method described in Ref. 21.

B. Weighted collections of spline shape functions

For simplicity, we consider spline shape functions for both the bulk density and the fine scale variations. Specifically, we set

$$\varphi_\varepsilon(z) := \left(\frac{1}{\varepsilon}\right)^d \varphi\left(\frac{z}{\varepsilon}\right), \quad z \in \mathbb{R}^d$$

with a reference shape function φ defined as a centered cardinal B-spline of degree p ,

$$\varphi(z) := \prod_{i=1}^d B_p(z_i) \quad \text{with support} \quad \left[-\frac{p+1}{2}, \frac{p+1}{2}\right]^d$$

involving standard univariate B-splines defined recursively by

$$B_0(x) := \mathbb{1}_{[-\frac{1}{2}, \frac{1}{2}]}(x) \quad \text{and} \quad B_p(x) := \int_{x-\frac{1}{2}}^{x+\frac{1}{2}} B_{p-1} \quad \text{for } p \geq 1.$$

In the sequel, it will be convenient to denote an arbitrary collection of weighted splines as

$$\Phi_\varepsilon[\mathbf{W}, \mathbf{Z}](z) := \sum_{k=1}^N w_k \varphi_\varepsilon(z - z_k),$$

where $\mathbf{W} = (w_k)_{k=1 \dots N}$, $\mathbf{Z} = (z_k)_{k=1 \dots N}$. With this convention, the two components of our general ansatz (1) read

$$f_*^n = \Phi_{h_*}[\mathbf{W}_*^n, \mathbf{Z}_*^n] \quad \text{and} \quad \delta f^n = \Phi_\varepsilon[\delta \mathbf{W}^n, \mathbf{Z}^n], \tag{4}$$

where $\mathbf{Z}_* := (jh_*)_{j \in \mathbb{Z}^d}$ are the nodes of the spline grid.

III. PARTICLE APPROXIMATIONS TO TRANSPORT EQUATIONS

Our method may be described for general non-linear transport problems of the form

$$\partial_t f(t, z) + U[f] \cdot \nabla_z f(t, z) = 0, \tag{5}$$

where $z \in \mathbb{R}^d$ is the phase-space variable, and $U[f]$ is the generalized velocity field associated with the solution f .

A. Characteristic flows

The characteristic trajectories associated with Eq. (5) are the curves $Z(t) = Z(t; s, z) \in \mathbb{R}^d$, solution to the ordinary differential equations (ODEs)

$$\frac{d}{dt}Z(t) = U[f](t, Z(t)), \quad Z(s) = z$$

for arbitrary $s, t \in [0, T]$ and $z \in \mathbb{R}^d$, see, e.g., Ref. 24. The (forward) characteristic flow between two times $t^n = n\Delta t$ and $t^{n+1} = (n + 1)\Delta t$ is then defined as

$$\mathcal{F}_{\text{ex}}^{n,n+1}(z) := Z(t^{n+1}; t^n, z), \quad (6)$$

and the inverse mapping $\mathcal{B}_{\text{ex}}^{n,n+1} := (\mathcal{F}_{\text{ex}}^{n,n+1})^{-1}$ is the backward flow

$$\mathcal{B}_{\text{ex}}^{n,n+1}(z) = Z(t^n; t^{n+1}, z). \quad (7)$$

Using the backward flow, we can write the solution to Eq. (5) over the time interval $[t^n, t^{n+1}]$ as

$$f(t^{n+1}, z) = f(t^n, \mathcal{B}_{\text{ex}}^{n,n+1}(z)). \quad (8)$$

Here, we may restrict ourselves to divergence-free velocity fields: $\text{div}_z U[f] = 0$. The characteristic flows are then measure-preserving, and the transport is conservative.

B. 1D Vlasov–Poisson equation

A simple example is the periodic Vlasov–Poisson equation in a two-dimensional phase space, i.e., $z = (x, v)$ with a periodic space coordinate $x \in [0, L]$ and velocity $v \in \mathbb{R}$,

$$\partial_t f(t, x, v) + v\partial_x f(t, x, v) - E(t, x)\partial_v f(t, x, v) = 0 \quad (9)$$

for $t \geq 0$, $(x, v) \in [0, L] \times \mathbb{R}$ with a normalized initial density $\int_0^L \int_{\mathbb{R}} f^0(x, v) dv dx = 1$ and a periodic electric field $E = E[f]$ defined by

$$\begin{cases} E(t, x) = -\partial_x \phi(t, x), \\ -\Delta \phi(t, x) = \rho(t, x) = \frac{1}{L} \int_{\mathbb{R}} f(t, x, v) dv. \end{cases} \quad (10)$$

This corresponds to an electrostatic, normalized ($\epsilon_0 = q_e = m = 1$) periodic electron plasma in 1D with constant neutralizing background ion density, so that $\int_0^L \rho dx = 0$. Here, the generalized velocity field is

$$U[f](t, z) = (v, -E(t, x)) \quad \text{with } z = (x, v).$$

Due to the non-linear nature of this transport equation, the characteristic flow has no explicit expression.

C. Full- f particle approximation

Particle approximations represent the transported density $f^n(z) \approx f(t^n, z)$ as a sum of numerical particles of the form

$$f^n(z) = \Phi_\varepsilon[\mathbf{W}^n, \mathbf{Z}^n](z) \quad (11)$$

with weights initially set to

$$w_k^0 := \frac{f^0(z_k^0)}{N_p g^0(z_k^0)}, \quad k = 1, \dots, N_p, \quad (12)$$

where g^0 is the sampling distribution of the initial markers $\mathbf{Z}^0 = (z_k^0)_{k=1, \dots, N_p}$, see, e.g., Ref. 11. As the problem is conservative, the weights are kept constant in time, $\mathbf{W}^{n+1} = \mathbf{W}^n$, and the markers are pushed forward

$$\mathbf{Z}^{n+1} = \mathcal{F}^n(\mathbf{Z}^n) \quad (13)$$

using some approximation to the forward flow in Eq. (6), which takes as parameter the numerical solution at time t^n ,

$$\mathcal{F}^n(\mathbf{Z}) = \mathcal{F}_{\Delta t}[\mathbf{W}^n, \mathbf{Z}^n](\mathbf{Z}).$$

D. Electrostatic full- f leap-frog flow

For the Vlasov–Poisson equation [Eq. (9)], a standard numerical flow is given by a leap-frog (Strang splitting) scheme

$$\mathcal{F}_{\Delta t, \Delta x}^{\text{lf, pic}}[\mathbf{W}^n, \mathbf{Z}^n](\mathbf{Z}) = \mathcal{F}_{\frac{\Delta t}{2}}^x \circ \mathcal{F}_{\Delta t, \Delta x}^{\text{v, pic}}[\mathbf{W}^{n+\frac{1}{2}}, \mathbf{Z}^{n+\frac{1}{2}}] \circ \mathcal{F}_{\frac{\Delta t}{2}}^x(\mathbf{Z}), \quad (14)$$

where \circ is the usual function composition operator associated with a grid with M points and the step size $\Delta x = L/M$. The first split flow reads

$$\mathcal{F}_{\frac{\Delta t}{2}}^x(x, v) := \left(x + \frac{\Delta t}{2}v, v\right), \quad (15)$$

and the second one takes the form

$$\mathcal{F}_{\Delta t, \Delta x}^{\text{v, pic}}[\mathbf{W}^{n+\frac{1}{2}}, \mathbf{Z}^{n+\frac{1}{2}}](x, v) := \left(x, v - \Delta t E^{n+\frac{1}{2}}(x)\right). \quad (16)$$

Here, we have set

$$\mathbf{W}^{n+\frac{1}{2}} := \mathbf{W}^n, \quad \mathbf{Z}^{n+\frac{1}{2}} := \mathcal{F}_{\frac{\Delta t}{2}}^x(\mathbf{Z}^n), \quad (17)$$

and the electric field is computed from the particles with a discrete Poisson solver such as the one described just below, which takes as input the particle weights and positions,

$$E^{n+\frac{1}{2}}(x) := \mathbb{E}_{\Delta x}^{\text{pic}}[\mathbf{W}^{n+\frac{1}{2}}, \mathbf{Z}^{n+\frac{1}{2}}](x). \quad (18)$$

E. Electrostatic full- f PIC field solver

To clarify how the particles are coupled with the field in Eq. (18), we recall the precise form of a basic Finite Difference (FD) Poisson solver for a general collection of weighted particles, i.e.,

$$\Phi_\varepsilon[\mathbf{W}, \mathbf{Z}] \mapsto \mathbb{E}_{\Delta x, \varepsilon}^{\text{pic}}[\mathbf{W}, \mathbf{Z}]. \quad (19)$$

We note that for periodic systems spectral solvers involving fast Fourier transforms are often used, see, e.g., Ref. 25, Sec. 8. In both cases, the charge deposition algorithm consists of computing grid values for the charge density

$$\rho_i = \frac{1}{L} - \int_{\mathbb{R}} \Phi_\varepsilon[\mathbf{W}, \mathbf{Z}](i\Delta x, v) dv = \frac{1}{L} - \sum_{k=1}^{N_p} w_k \varphi_\varepsilon(i\Delta x - x_k) \quad (20)$$

for $i = 1, \dots, M$. A discrete Poisson equation is then solved. With a FD scheme, this reads

$$E_i = -\frac{\phi_{i+1} - \phi_{i-1}}{2\Delta x}, \quad -\frac{\phi_{i+1} - 2\phi_i + \phi_{i-1}}{\Delta x^2} = \rho_i$$

for $i = 1, \dots, M$. The electric field (19) is then defined using the same spline shape functions as the particles but a different normalization and scaled with the FD grid as follows:

$$E_{\Delta x, \varepsilon}^{\text{pic}}(x) := \sum_{i=1}^M E_i \tilde{\varphi}_{\Delta x}(x - i\Delta x), \quad \text{where} \quad \tilde{\varphi}_{\Delta x}(x) = \varphi\left(\frac{x}{\Delta x}\right).$$

In the usual case where $\varepsilon = \Delta x$, we write $E_{\Delta x}^{\text{pic}} = E_{\Delta x, \Delta x}^{\text{pic}}$.

IV. A δf PIC SCHEME WITH REMAPPINGS

We now describe the main steps of a δf method involving a decomposition of the form [Eq. (1)]. Denoting by N_r the remapping period, we first observe that the bulk part is frozen between two remapping steps. This yields, in particular,

$$f_*^n = f_*^m, \quad \text{where} \quad m = m(n) := \left\lfloor \frac{n}{N_r} \right\rfloor N_r. \quad (21)$$

The δf part is then evolved as follows.

A. δf particle approximation: Basic steps

A typical δf time step consists of (i) pushing the markers forward to follow the characteristic curves as in Eq. (13)

$$\mathbf{Z}^{n+1} = \mathcal{F}^n(\mathbf{Z}^n), \quad (22)$$

and (ii) updating the weights. Indeed, since the numerical particles now represent a variation $\delta f^n = f^n - f_*^n$ with frozen f_*^n on the time step, their weights no longer carry a conserved density and they must be evolved in time. In the *direct* δf method (see, e.g., Ref. 11, Sec. 3), they are set as

$$\delta w_k^n := \frac{f_k^n - f_*^n(z_k^n)}{N_p g_k^n} \approx \frac{\delta f(t^n, z_k^n)}{N_p g(t^n, z_k^n)}, \quad k = 1, \dots, N_p, \quad (23)$$

where

$$f_k^n \approx f(t^n, z_k^n) \quad \text{and} \quad g_k^n \approx g(t^n, z_k^n) \quad (24)$$

correspond to the transported density and the markers density probability, respectively, evaluated at the particle positions. Since both values are constant along exact trajectories, a reasonable choice is

$$f_k^n := f^0(z_k^0) \quad \text{and} \quad g_k^n := g^0(z_k^0). \quad (25)$$

Up to these constant values, we see that the δf weights only depend on the bulk density, which is characterized by the weights $\mathbf{W}_*^n = \mathbf{W}_*$, at the marker positions \mathbf{Z}^n . Hence, we may summarize the weighting scheme in Eqs. (23)–(25) as

$$\delta \mathbf{W}^n := \delta \mathcal{W}(\mathbf{W}_*^n, \mathbf{Z}^n). \quad (26)$$

B. Electrostatic δf -PIC flow

In the case of a δf ansatz, the leap-frog flow reads

$$\mathcal{F}_{\Delta t, \Delta x}^{\text{lf}, \delta f}[\mathbf{W}_*^m, \mathbf{Z}^n] = \mathcal{F}_{\frac{\Delta t}{2}}^x \circ \mathcal{F}_{\Delta t, \Delta x}^{\text{v}, \delta f}[\mathbf{W}_*^m, \mathbf{Z}^{n+\frac{1}{2}}] \circ \mathcal{F}_{\frac{\Delta t}{2}}^x \quad (27)$$

with split flows defined similarly as in Eqs. (15)–(16) and a field solver adapted to the ansatz from Eq. (1), namely,

$$\mathcal{F}_{\Delta t, \Delta x}^{\text{v}, \delta f}[\mathbf{W}_*^m, \mathbf{Z}^{n+\frac{1}{2}}](x, v) := \left(x, v - \Delta t E^{n+\frac{1}{2}}(x)\right) \quad (28)$$

with

$$E^{n+\frac{1}{2}} = E_*^m + E_{\delta f}^{n+\frac{1}{2}} = E_{\Delta x, h_*}[\mathbf{W}_*^m, \mathbf{Z}^{n+\frac{1}{2}}]. \quad (29)$$

Since both parts of the density are formally written as weighted collections of spline shapes, we may use a standard PIC solver for each parts. The field induced by the bulk density f_*^m is

$$E_*^m := E_{\Delta x, h_*}^{\text{pic}}[\mathbf{W}_*^m, \mathbf{Z}_*^m], \quad (30)$$

and the field corresponding to the variation $\delta f^{n+\frac{1}{2}} = \Phi_\varepsilon[\delta \mathbf{W}^{n+\frac{1}{2}}, \mathbf{Z}^{n+\frac{1}{2}}]$, with scaling $\varepsilon = \Delta x$, reads

$$E_{\delta f}^{n+\frac{1}{2}} = E_{\Delta x}^{\text{pic}}[\delta \mathbf{W}^{n+\frac{1}{2}}, \mathbf{Z}^{n+\frac{1}{2}}] \quad \text{with} \quad \begin{cases} \delta \mathbf{W}^{n+\frac{1}{2}} := \delta \mathcal{W}(\mathbf{W}_*^m, \mathbf{Z}^{n+\frac{1}{2}}), \\ \mathbf{Z}^{n+\frac{1}{2}} := \mathcal{F}_{\frac{\Delta t}{2}}^x(\mathbf{Z}^n), \end{cases} \quad (31)$$

see (17), (26), and (23). The resulting field solver corresponding to (29) may then be written as

$$E_{\Delta x, h_*}[\mathbf{W}_*^m, \mathbf{Z}^{n+\frac{1}{2}}] := E_{\Delta x, h_*}^{\text{pic}}[\mathbf{W}_*^m, \mathbf{Z}_*^m] + E_{\Delta x}^{\text{pic}}[\delta \mathcal{W}(\mathbf{W}_*^m, \mathbf{Z}^{n+\frac{1}{2}}), \mathbf{Z}^{n+\frac{1}{2}}].$$

Since the field E_*^m is constant between two remappings, regular time steps only involve a deposition of δf particles to update the $E_{\delta f}$ part. Note that even if $|\delta f^n| \ll |f_*^m|$, there is no reason why $E_{\delta f}$ should be small compared to E_*^m . In particular, we have $E_*^m \approx 0$ when f_*^m is close to an equilibrium density.

C. FBL remappings with auxiliary markers

For the remappings, namely, the updates of the bulk density $f_*^m \mapsto f_*^n$ with $n = m + N_r$, we resort to forward-backward Lagrangian (FBL) reconstructions.²¹ As mentioned above, this involves a collection of auxiliary markers that have been reset on the Cartesian grid at the previous remapping step,

$$\tilde{\mathbf{Z}}^m := \mathbf{Z}_*, \quad \text{i.e.,} \quad \tilde{z}_j^m := j h_* \quad \forall j \in \mathbb{Z}^d, \quad (32)$$

and pushed forward similarly as the classical ones in Eq. (22),

$$\tilde{\mathbf{Z}}^n = \mathcal{F}^{n-1}(\dots \mathcal{F}^m(\mathbf{Z}_*) \dots) \quad (33)$$

in order to track the forward characteristic flow. The FBL method then performs a semi-Lagrangian step

$$f_*^n := A_* \mathcal{T}_{\text{fbl}}[\tilde{\mathbf{Z}}^n] f_*^m, \quad (34)$$

where A_* is a spline interpolation or quasi-interpolation operator on the h_* grid, and $\mathcal{T}_{\text{fbl}}[\tilde{\mathbf{Z}}^n]: f \mapsto f \circ \mathcal{B}_{\text{fbl}}^{m,n}$ is a transport operator approximating the exact one in Eq. (8), based on the FBL backward flow

$$\mathcal{B}_{\text{fbl}}^{m,n} = \mathcal{B}_{\text{fbl}}[\tilde{\mathbf{Z}}^n] \approx \mathcal{B}_{\text{ex}}^{m,n}. \quad (35)$$

As a result, we obtain a numerical scheme to update the coefficients of the bulk density, which we summarize as

$$\mathbf{W}_*^n := \text{Remap}[\tilde{\mathbf{Z}}^n](\mathbf{W}_*^m), \quad (36)$$

where n is a multiple of N_r and $m = n - N_r$.

In Sec. IV D, we give some details about the FBL algorithm.

D. FBL density reconstructions

The FBL scheme in Eq. (34) performs a semi-Lagrangian approximation with a backward flow derived from the position of the auxiliary markers $\tilde{\mathbf{Z}}^m$ in Eq. (33), as follows:

- (i) For any index $j \in \mathbb{Z}^d$, let \tilde{z}_i^m with $i = i(j)$ be an FBL marker close to the node jh_* ;
- (ii) Compute the quadratic local backward flow associated with \tilde{z}_i^m , namely,

$$B_i^{m,n}(z) = z_i^* + D_i^n(z - \tilde{z}_i^n) + \frac{1}{2}(z - \tilde{z}_i^n)^T Q_i^n(z - \tilde{z}_i^n), \quad (37)$$

where $z_i^* = ih_*$ and D_i^n, Q_i^n correspond, respectively, to the Jacobian matrix and Hessian tensor of the backward flow at $z = \tilde{z}_i^n$, computed from the positions of the logical neighbors of \tilde{z}_i^n (for more details see Ref. 21);

- (iii) Reconstruct the transported density at node $z_j^* = jh_*$ using this approximated flow

$$f_{*,j}^n := f_*^m(B_i^{m,n}(z_j^*)). \quad (38)$$

- (iv) Compute the new weights \mathbf{W}_*^n using a spline approximation operator A^* .

For the operator A_* , we can use a standard interpolation scheme or even a local quasi-interpolation scheme, see, e.g., Ref. 26. Both compute high-order B-spline approximations, but the latter only involves local pointwise evaluations of the target function. The resulting approximation takes the form

$$A_*g(z) := \sum_{j \in \mathbb{Z}^d} w_j(g) \varphi_{h_*}(z - jh_*)$$

with

$$w_j(g) := h_*^d \sum_{\|i\|_\infty \leq s_p} a_i g((j+i)h_*) \quad (39)$$

with coefficients $a_i := a_{i_1} \cdots a_{i_d}$ on a local stencil of size s_p , as given by the symmetric method ($a_i = a_{-i}$) in Ref. 26, Sec. 6.

V. OVERVIEW OF THE δf PARTICLE SCHEME WITH FBL REMAPPINGS

Gathering the above steps, we may write the solution at time t^n as

$$f^n = f_*^m + \delta f^n \quad \text{with} \quad \begin{cases} f_*^m = \Phi_{h_*}[\mathbf{W}_*^m, \mathbf{Z}_*], \\ \delta f^n = \Phi_{\tilde{z}}[\delta \mathbf{W}^n, \mathbf{Z}^n], \end{cases} \quad (40)$$

where we remind that $\mathbf{Z}_* = (jh_*)_{j \in \mathbb{Z}^d}$ are the coarse grid nodes and $m = m(n)$ is the last remapping step preceding n , see Eq. (21). The proposed scheme can then be summarized by the flow chart in Fig. 1 and reads as follows.

A. Initialization

- Compute the weights \mathbf{W}_*^0 of the initial bulk density $f_*^0 = \Phi_{h_*}[\mathbf{W}_*^0, \mathbf{Z}_*]$ with

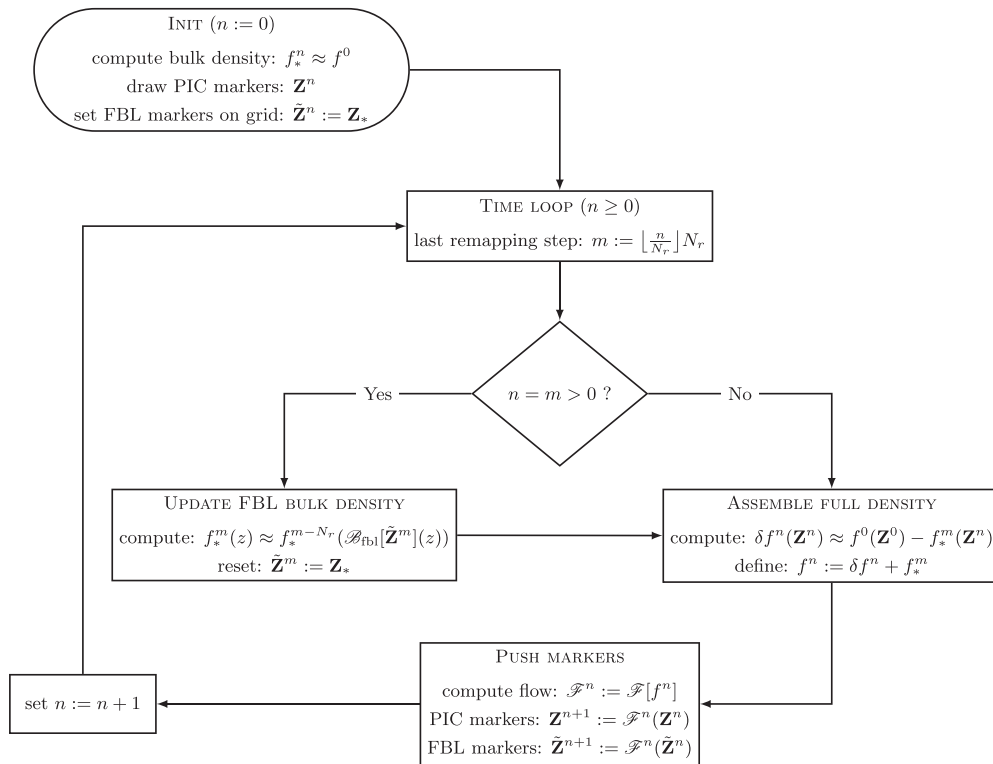


FIG. 1. Flow chart summarizing the main steps of the FBL- δf scheme.

$$f_*^0 := A_* f^0;$$

- Use a spline approximation scheme such as the one in Eq. (39);
- Draw the δf markers according to some initial sampling distribution

$$\mathbf{Z}^0 \sim g^0;$$

- Initialize the auxiliary FBL markers on the h_* grid (32),

$$\tilde{\mathbf{Z}}^0 := \mathbf{Z}_*.$$

B. Time loop

For $n = 0, \dots, N_t - 1$, let $m := \lfloor \frac{n}{N_r} \rfloor N_r$ denote the previous remapping step as in Eq. (21), and do

- If $n = m > 0$, **remap**
 - Update the bulk density weights (coefficients) with the FBL method described in Sec. IV D

$$\mathbf{W}_*^m := \text{Remap}[\tilde{\mathbf{Z}}^m](\mathbf{W}_*^m). \tag{41}$$

- Reset the FBL markers with (32), $\tilde{\mathbf{Z}}^m := \mathbf{Z}_*$.
- **Push** the markers forward

$$\begin{cases} \mathbf{Z}^{n+1} := \mathcal{F}^n(\mathbf{Z}^n), \\ \tilde{\mathbf{Z}}^{n+1} := \mathcal{F}^n(\tilde{\mathbf{Z}}^n) \end{cases} \tag{42}$$

using some discrete flow $\mathcal{F}^n = \mathcal{F}_{\Delta t}^{\delta f}[\mathbf{W}_*^m, \mathbf{Z}^n]$ such as the leap-frog scheme in Eq. (27), which involves deposition and field solver steps as detailed in Secs. IV B and III E.

C. FBL variant

A semi-Lagrangian FBL variant of the above scheme can be obtained by (i) setting $N_p = 0$ and (ii) treating the FBL markers $\tilde{\mathbf{Z}}^n$ as “active” particles in the field solver.

Specifically, this amounts to discarding the δf part ($\mathbf{Z}^n = \emptyset$) so that the full density is represented on the structured spline grid on the successive remapping steps and to defining the electric field in Eqs. (27) and (28) by

$$E^{n+\frac{1}{2}} = E_{\Delta x}^{\text{pic}}[\mathbf{W}_*^{n+\frac{1}{2}}, \tilde{\mathbf{Z}}^{n+\frac{1}{2}}] \tag{43}$$

as one would do with a standard full- f PIC scheme, see (16)–(18), using the weighted FBL markers as standard particles. Note that

here the weights are those set at the last remapping step (41), i.e., $\mathbf{W}_*^{n+\frac{1}{2}} = \mathbf{W}_*^m$ with m as in (21).

VI. NUMERICAL RESULTS

In this section, we assess the proposed method on the periodic 1D1V Vlasov–Poisson system (9) and (10), where we compare it with other PIC and semi-Lagrangian schemes.

A. Numerical parameters for the tested methods

For the purpose of comparison, we use different values for the main parameters listed in Sec. II A, including some limiting values, which allow to emulate full- f and δf PIC schemes, as well as a semi-Lagrangian FBL scheme. All the methods use cubic splines for particle shape functions.

As the solutions are typically supported in a region occupying roughly one half of the computational phase-space domain, for each grid resolution $N_x \times N_v$, we indicate as $N_*^{\text{eff}} \approx N_*/2$ with $N_* = N_x N_v$, the approximate number of active spline coefficients on the grid, corresponding to the “effective” grid resolution. The resulting methods are summarized in Table I.

Here, the active/passive/frozen status of FBL markers $\tilde{\mathbf{Z}}^n$ refers to their role in the scheme and, in particular, in the field solver that is involved in the push step in Eq. (42): *Active* markers (used in the FBL variant) carry a charge which is deposited as explained in Sec. V C, while *passive* markers (used in the proposed FBL- δf scheme) do not, as described in Sec. IV B. Finally, *frozen* markers are passive and never pushed: In the absence of remappings, only their initial position on the grid is used to represent the bulk density $f_*^n = f_*^0$.

We point out that in this study we do not attempt to assess the computational cost of various methods, as the testcases are very simple and our implementation is far from being optimal. In particular, one should keep in mind that full- f and basic δf methods, even if noisier, may be quicker to run and easier to parallelize than their FBL counterparts. Efficient implementations of the latter are left for future works.

B. Test cases

Our numerical experiments are based on a textbook 1D1V two-stream instability test case where the initial distribution is of the form

$$f^0(x, v) = (1 + \alpha \cos(kx)) \frac{v^2}{\sqrt{2\pi}} e^{-\frac{v^2}{2}} \tag{44}$$

with a wave number of $k = \frac{1}{2}$ and a perturbation amplitude of $\alpha = 0.01$. This solution is known to develop thin filaments in the

TABLE I. Numerical parameters used for the different methods, as described in the text.

Method	N_p	Grid (FBL)	N_*^{eff}	N_r	FBL markers
Full- f PIC, coarse	20 000	0×0	0	∞	\emptyset
Full- f PIC, fine	320 000	0×0	0	∞	\emptyset
δf PIC	18 752	50×50	1250	∞	frozen
FBL, coarse	0	200×200	20 000	20	active
FBL, fine	0	800×800	320 000	20	active
FBL- δf PIC	18 752	50×50	1250	20	passive

phase space that are difficult to resolve numerically, see, e.g., Refs. 27 and 28. Here, we will simulate two-stream instabilities on the time interval $[0, 100]$. We shall consider two versions of the problem:

- (i) the basic Two-Stream Instability (TSI) test case where the solution is evolved with the self-consistent Vlasov–Poisson system (9) and (10);
- (ii) a Driven Two-Stream Instability (DTSI) test case where an oscillating external field

$$E_{\text{ext}}(t, x) = \frac{1}{10} \cos\left(\frac{\pi t}{50}\right) \tag{45}$$

is added to the self-consistent field from Eq. (10). The effect of this external drive is to add a periodic perturbation to TSI trajectories, uniform in x and v ,

$$(x, v) \mapsto \left(x + \frac{250}{\pi^2} \left(1 - \cos\left(\frac{\pi t}{50}\right)\right), v + \frac{5}{\pi} \sin\left(\frac{\pi t}{50}\right)\right),$$

so that the solutions to both test cases coincide for $t = 100$.

The computational phase-space domain is set to

$$\Omega = [0, L] \times [-v_{\text{max}}, v_{\text{max}}]$$

with $L = \frac{2\pi}{k}$ and $v_{\text{max}} = 10$. Note that this value is significantly larger than what is usually seen in the literature, in order to correctly represent the variations in velocity induced by the oscillating drive. Finally, the initial sampling distribution for the particles is taken as $g^0 = (\int f^0(z) dz)^{-1} f^0$ for the full- f PIC scheme and as

$$g^0(x, v) = \frac{1}{L} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{v^2}{2\sigma^2}} \tag{46}$$

with $\sigma = 2$ for the δf and FBL- δf PIC methods.

C. Phase-space density plots

In this section, we assess the accuracy and denoising capabilities of the basic δf PIC and FBL- δf PIC methods by plotting various densities in phase space. In Figs. 2–9, the axes show $x \in [0, L]$ and $v \in [-v_{\text{max}}, v_{\text{max}}]$ with L and v_{max} as indicated above, and the same color range is used for the purpose of comparison.

In Figs. 2 and 3, we first plot the densities f^n corresponding to different methods listed in Table I for both the TSI and DTSI test cases, at the time $t^n = 100$ where the exact solutions coincide.

For both test cases, we observe very noisy (oscillating) profiles for the coarse full- f PIC solutions, which use about 20 000 particles, and smoother profiles for the fine PIC solutions, which use 16 times as many particles. FBL solutions are essentially free of spurious oscillations with many fine structures well resolved in the fine solutions compared to the coarse ones. We also note that the TSI and DTSI solutions are in very good agreement at $t^n = 100$ for these full- f numerical simulations.

Regarding the δf solutions, we observe that in the TSI test case (Fig. 2) both the basic δf PIC and the FBL- δf PIC methods allow to strongly reduce the level of oscillations compared to the coarse PIC simulation—all three using approximately 20 000 active particles. In the DTSI case, however (Fig. 3), we see a significant reduction in the accuracy of the basic δf solution, where the global central vortex suffers from a visible deterioration and spurious streams of particles have appeared at high velocities. In contrast, the accuracy of the FBL- δf PIC method in the DTSI case remains very comparable to that of the TSI case. In particular, we observe that the new method is able to capture several low-density filaments, i.e., fine regions with zero or very low density distinct from the central vortex. In the basic δf simulation, these fine structures are poorly resolved, as we can see from Fig. 3 where the low-density filaments on the left of the central vortex are virtually merged with the latter. To better visualize the difference in

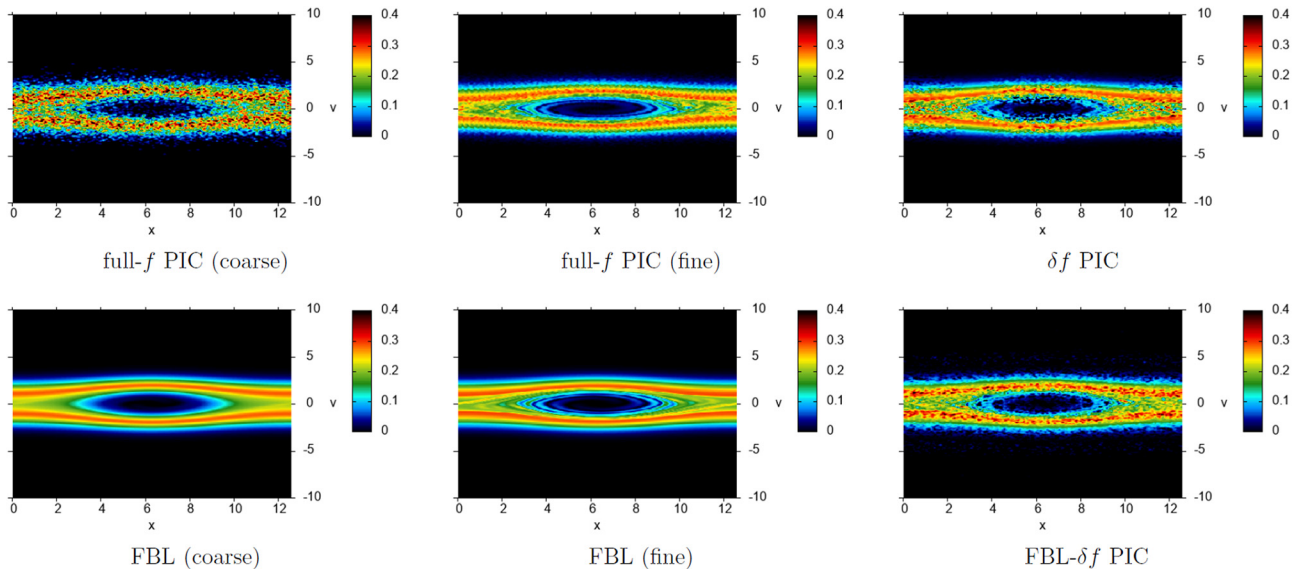


FIG. 2. Phase-space densities f^n obtained for the TSI test case at $t^n = 100$ with the methods listed in Table I. Here, the basic δf and FBL- δf methods use about 18 750 standard particles and 1250 FBL markers, which is comparable to the 20 000 particles (or active markers) used in the coarse PIC and FBL methods. Fine PIC and FBL simulations use 16 as many particles.

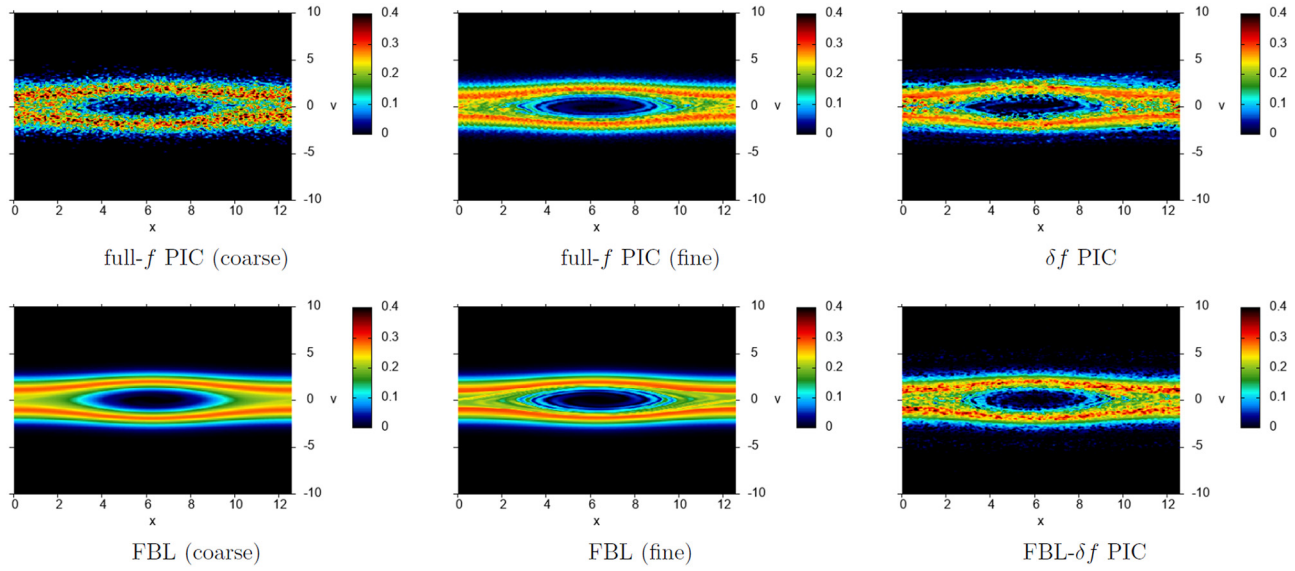


FIG. 3. Densities obtained for the DTSI test case at $t^n = 100$, where the exact solution coincides with the TSI one. As in Fig. 2, the different methods are those listed in Table I.

the two methods, we next show in Figs. 4–9 a series of plots corresponding to snapshots of various densities at successive times $t^n = 0, 12.5, \dots, 87.5$ for the DTSI test case: In Figs. 4 and 5, we show the evolution of the δf and FBL- δf solutions, completed by that of a reference scheme (an FBL simulation using a 1024×1024 grid) in Fig. 6. In Fig. 7, we then show the evolution of the bulk density f_*^n computed by the FBL- δf PIC method, which we remind is periodically remapped on a 50×50 grid of cubic splines, see Table I. Finally, the evolution of the residual δf^n densities is shown in Fig. 8 for the basic δf method and in Fig. 9 for the FBL- δf method.

From these plots, we see that the density strongly deviates from its initial value due to the external drive, which clearly challenges the basic assumption of a standard δf approach. Here, we have chosen a deviation so large that the bulk density cannot be compensated by a proper δf weighting of the particles, leading to significant errors in the full f^n density visible in Fig. 4. As expected, the solution computed with the FBL- δf PIC method in Fig. 5 does not show this erratic behavior. The main reason for this is visible in Fig. 7 where we see that the coarse bulk density computed by the FBL- δf scheme properly follows the total density. As a result, the δf particles can effectively

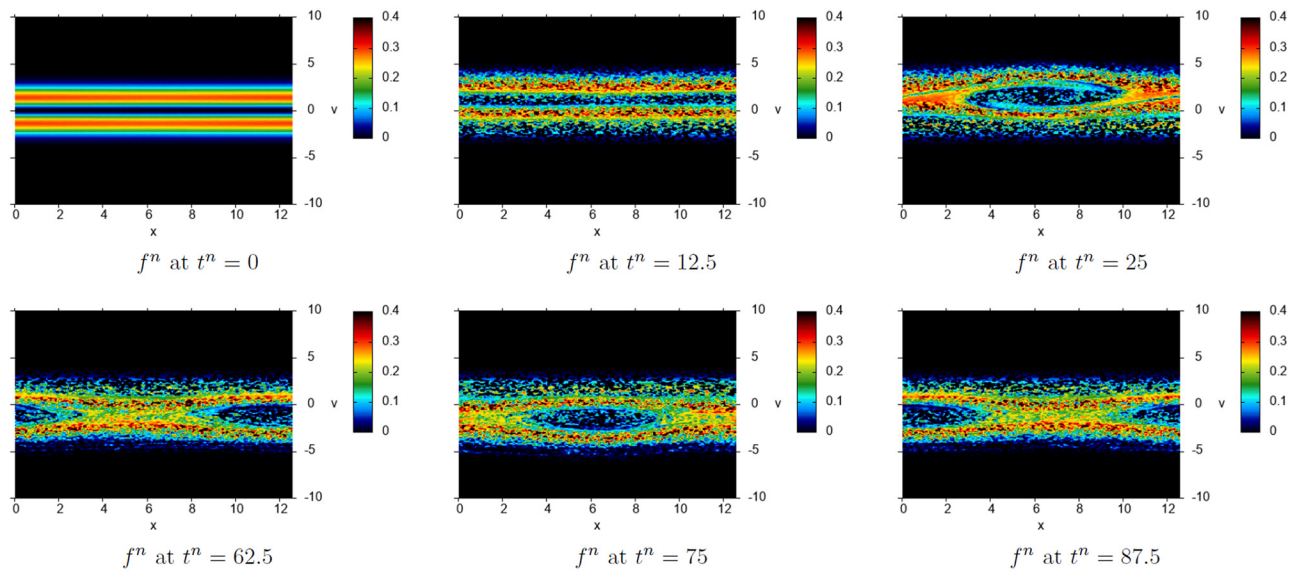


FIG. 4. Snapshots of the density f^n computed by the δf PIC scheme at various times t^n , for the DTSI test case. At time $t^n = 100$, the corresponding density is shown in the upper-right plot of Fig. 3.

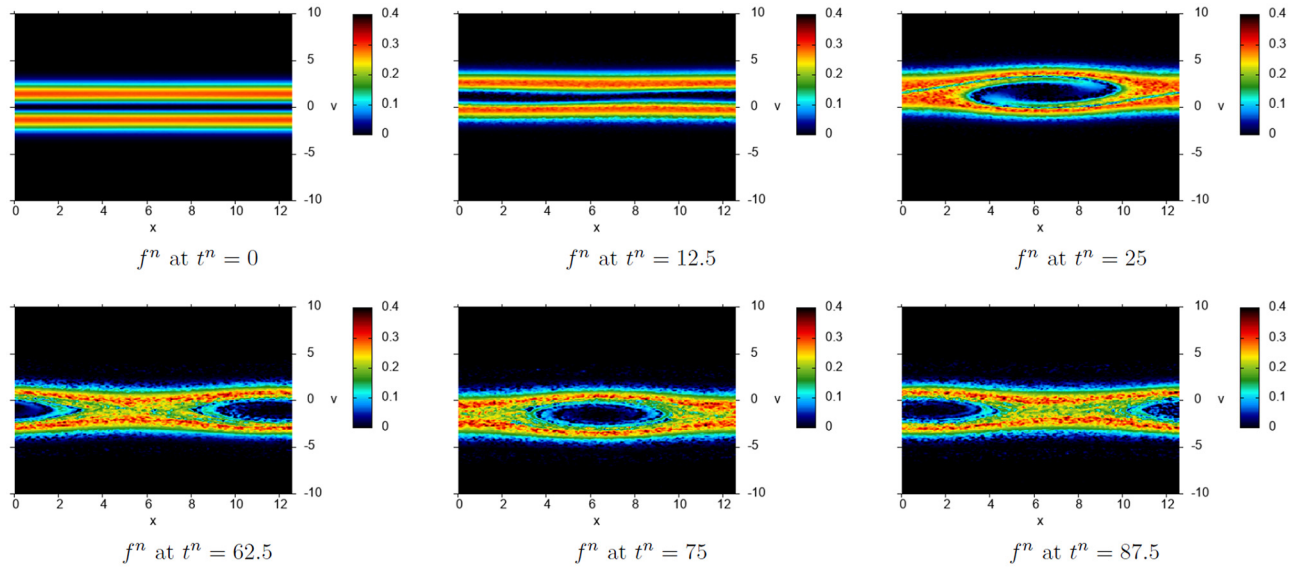


FIG. 5. Snapshots of the density f^n computed by the FBL- δf PIC scheme at various times t^n , for the DTSI test case. At time $t^n = 100$, the corresponding density is shown in the lower-right plot of Fig. 3.

represent small scale variations with respect to the main part of the plasma density. This is made evident in Figs. 8 and 9 where we see that the amplitude of the δf particle density for the FBL- δf scheme is significantly lower than that with a basic δf method.

In Fig. 7, we also observe that the numerical bulk density is subject to a visible numerical diffusion. This is an expected consequence of the periodic coarse-grid interpolations involved in the semi-Lagrangian updates, and over moderate times, it should not pose a serious issue since the total density is described by the sum of the bulk and the perturbed

particle density. Over long simulation times, however, this numerical diffusion will lead to bulk densities that do not approximate well the total f . In such cases, a proper re-initialization of the bulk density f_b will have to be devised, based on the available representation of the total density, and various methods do exist in this direction such as adaptive kernel density estimation algorithms^{29–31} or adaptive sparse-grid methods.³² We note that such an algorithm is likely to be more costly than a simple update of the coarse bulk density, but this should be mitigated by the fact that it will be applied on much larger time periods.

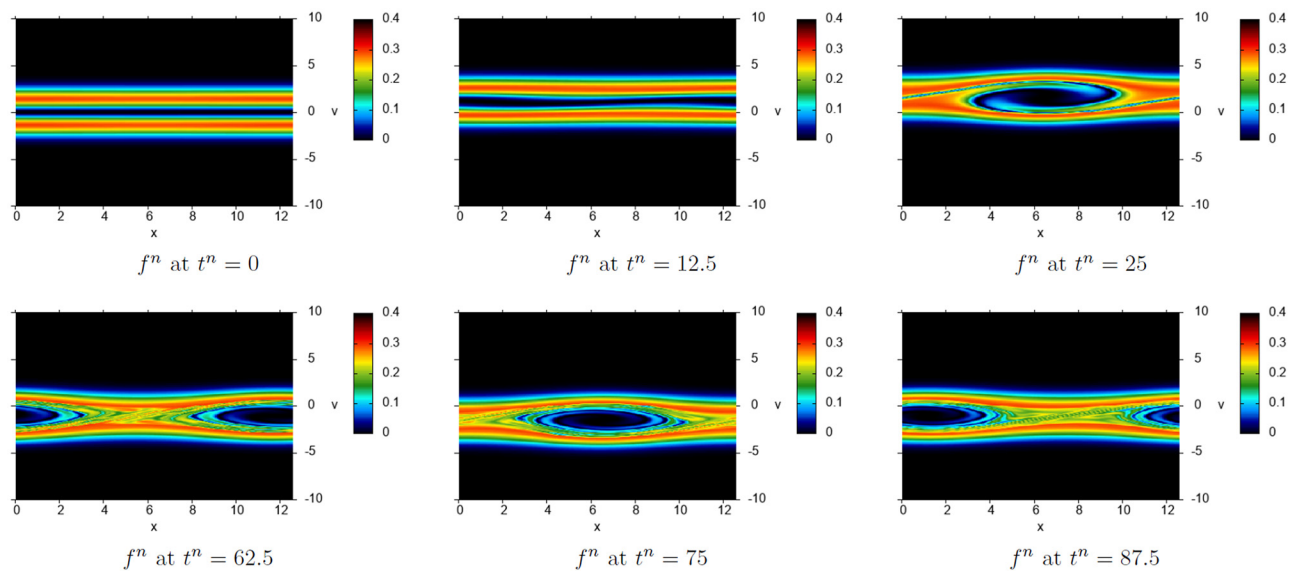


FIG. 6. Snapshots of the density f^n obtained with a reference method (an FBL simulation using a 1024×1024 grid) for the same times as in Figs. 4 and 5, again for the DTSI test case.

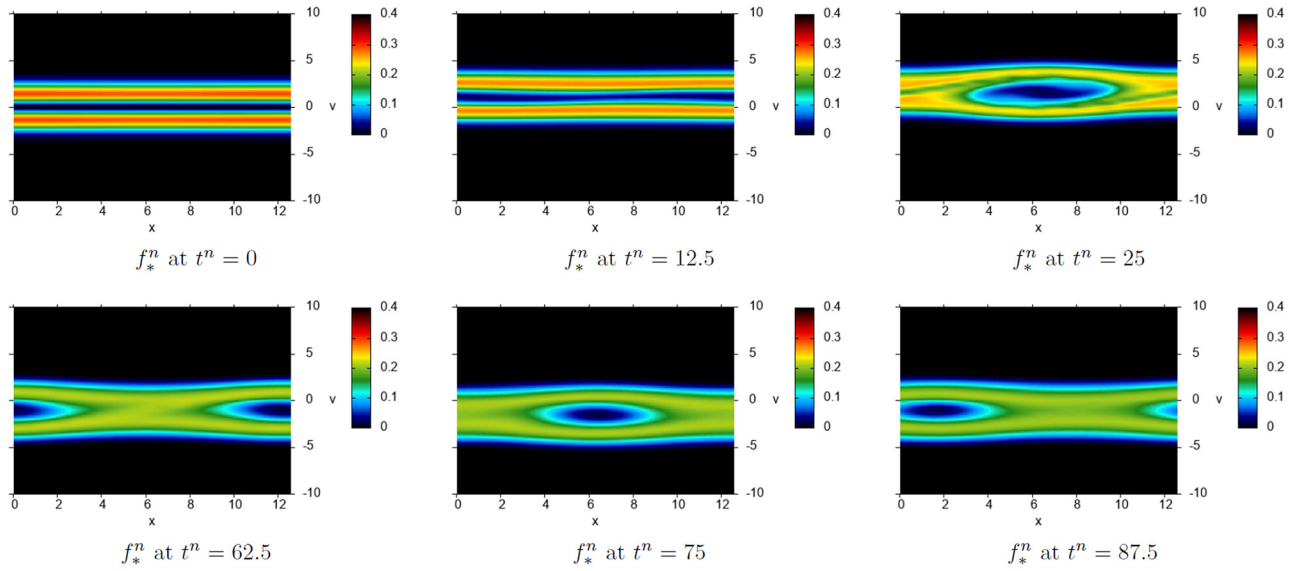


FIG. 7. Snapshots of the bulk density f_*^n corresponding to the FBL- δf PIC simulation shown in Fig. 5.

D. Density and field plots in physical space

In Figs. 10 and 11, we next plot the snapshots of the deposited charge densities ρ^n for the DTSI test case, corresponding to δf and FBL- δf solutions shown in Figs. 4 and 5. For a better assessment of the accuracy, we also plot in dashed line a reference charge density computed by a fine semi-Lagrangian scheme. The associated E^n fields are plotted in Figs. 12 and 13.

These plots show that despite the velocity averaging, the high level of noise visible in the δf phase-space densities is also present in

the deposited charge densities, which results in significant errors in the self-consistent electric field.

E. Approximation of momentum and energy

We next assess the accuracy of the different methods in approximating the system momentum and energy, namely,

$$\int_{\Omega} v f(z) dz \quad \text{and} \quad \frac{1}{2} \int_{\Omega} v^2 f(z) dz + \frac{1}{2} \int_0^L E(x)^2 dx$$

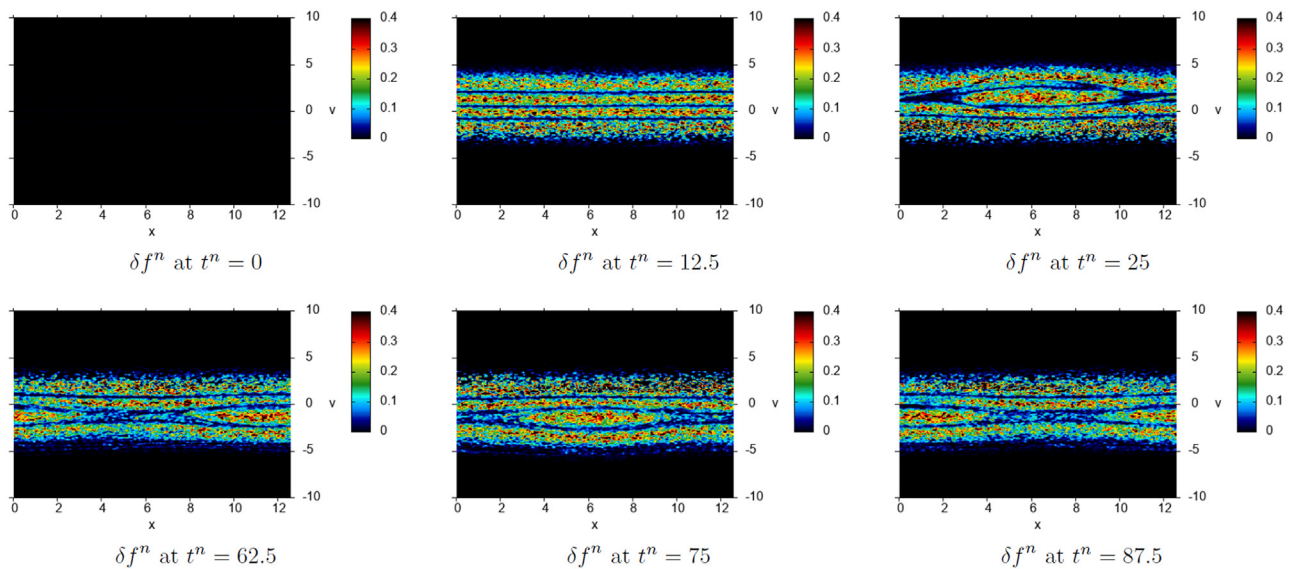


FIG. 8. Snapshots of the residual density δf^n corresponding to the δf PIC simulation shown in Fig. 4.

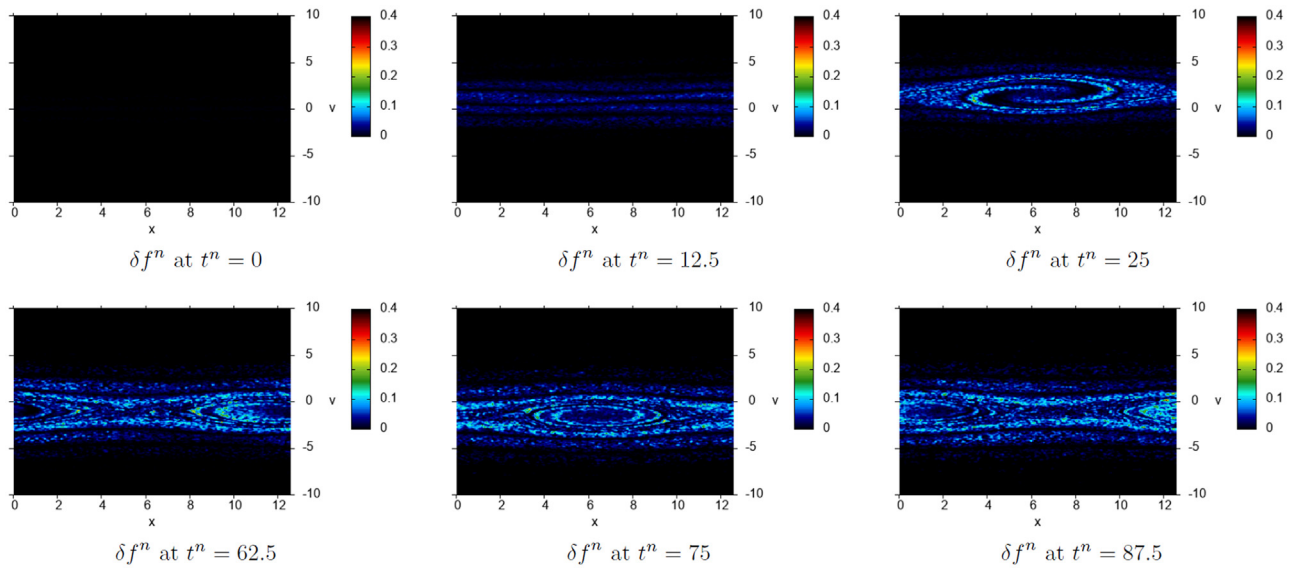


FIG. 9. Snapshots of the residual density δf^n corresponding to the FBL- δf PIC simulation shown in Fig. 5.

with $z = (x, v) \in \Omega = [0, L] \times [-v_{\max}, v_{\max}]$ as above. In Fig. 14, we plot the time evolution of these discrete quantities for the TSI test-case, where they are invariants of the exact solution (assuming $f = 0$ for $|v| > v_{\max}$). The first finding is that both the full- f PIC and the FBL scheme (for the coarse and fine meshes) preserve these quantities with very good accuracy, while visible errors accumulate for the two δf

methods. More precisely, we observe oscillating errors for both δf schemes, with no clear winner: Although the oscillations are clearly of larger amplitude with the basic δf scheme, a secular growth is visible on the numerical energy for the FBL- δf method, which may be due to the numerical diffusion of the bulk density (visible in Fig. 7) caused by the periodic coarse updates as discussed above. We note, however, that

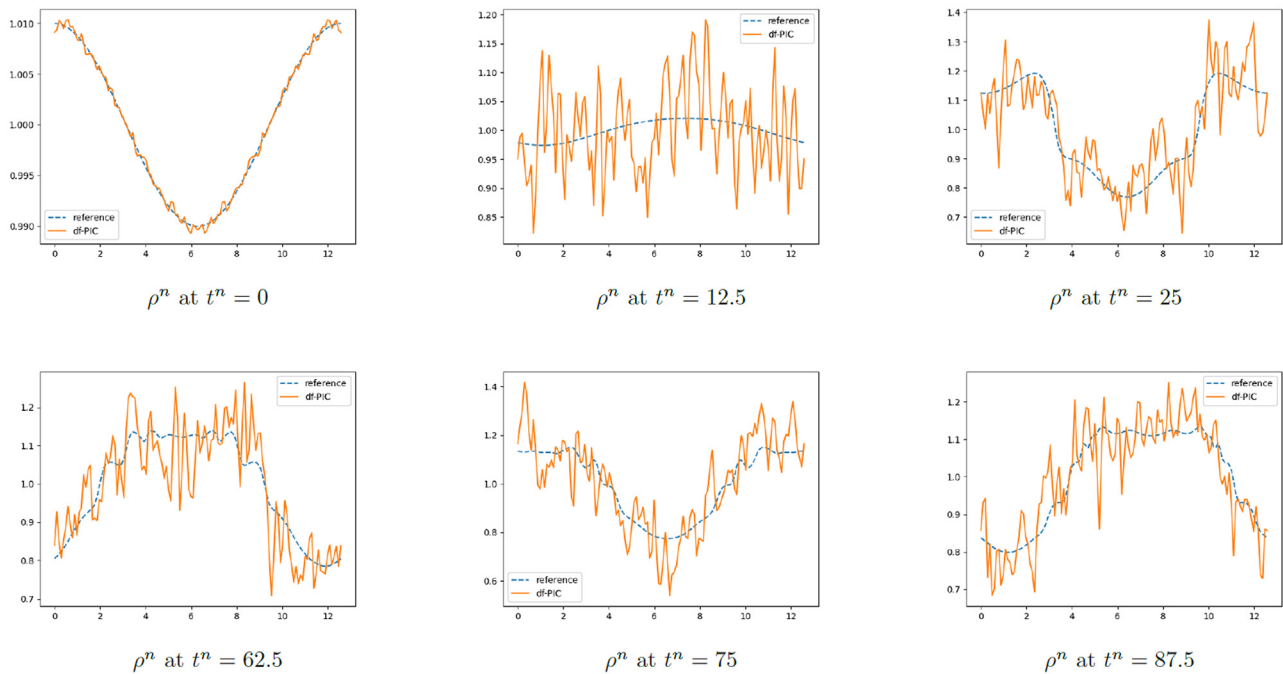


FIG. 10. Snapshots of the charge density ρ^n as a function of x for various time steps, deposited with the δf PIC scheme (with the TSI test case). These space densities correspond to the phase-space densities shown in Fig. 4.

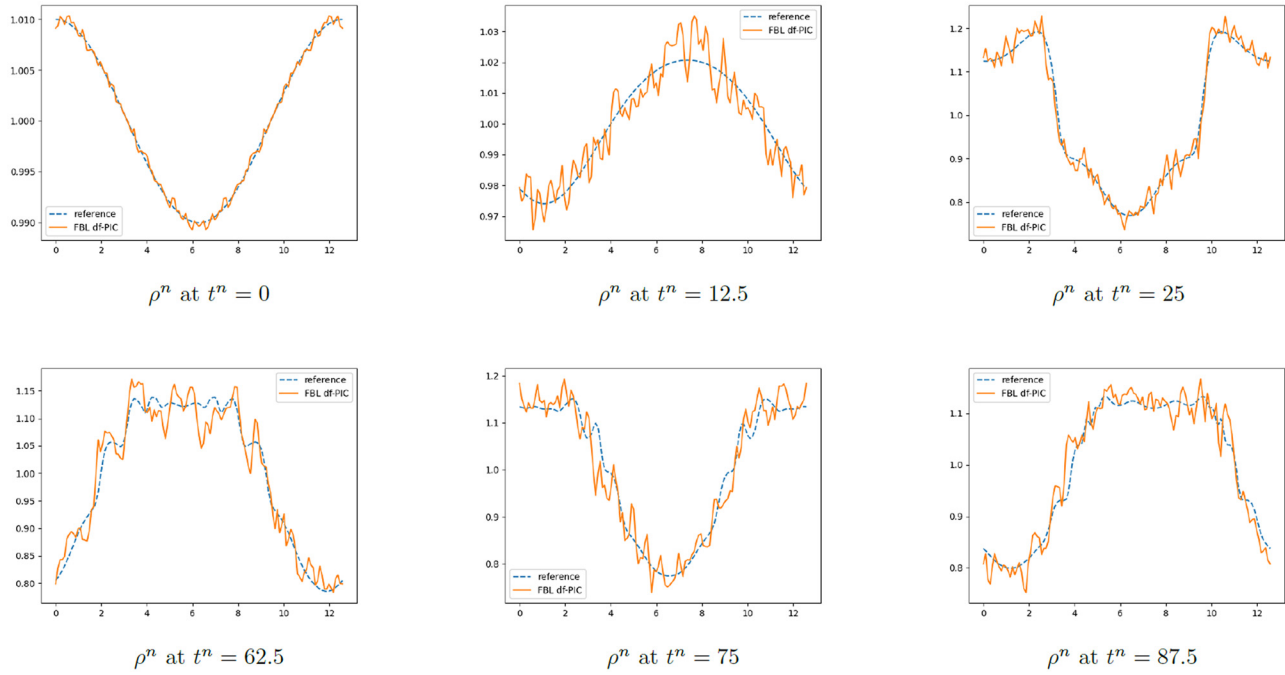


FIG. 11. Snapshots of the charge density ρ^n as a function of x for various time steps, deposited with the FBL- δf PIC scheme (with the DTSL test case). These space densities correspond to the phase-space densities shown in Fig. 5.

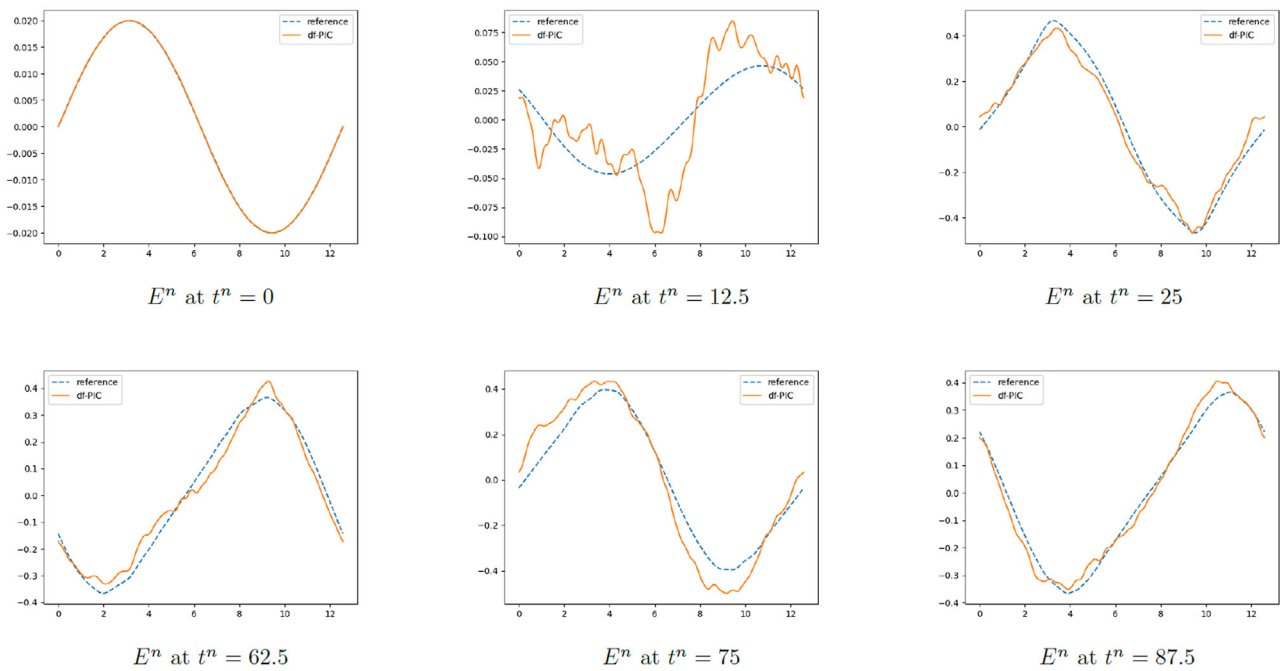


FIG. 12. Snapshots of the self-consistent field E^n as a function of x for various time steps, computed by the δf PIC scheme (DTSL). Like the charge densities in Fig. 10, these fields correspond to the phase-space densities shown in Fig. 4.

06 September 2024 12:56:09

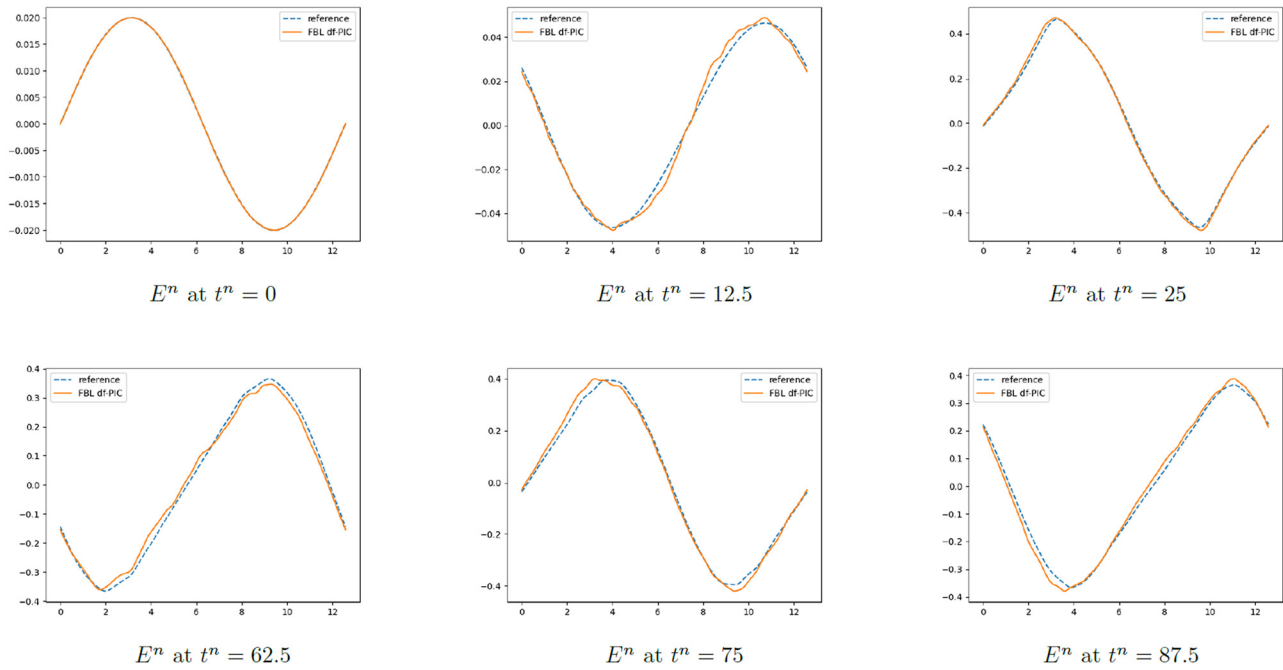


FIG. 13. Snapshots of the self-consistent field E^n as a function of x for various time steps, computed by the FBL- δf PIC scheme (DTSI). Like the charge densities in Fig. 11, these fields correspond to the phase-space densities shown in Fig. 5.

for the TSI test-case both δf schemes did also show similar accuracy in the density plots of Fig. 2, so that their similar performance in the conservation properties may not come as a big surprise.

For the DTSI test-case, the total momentum and energy are no longer system invariants due to the external drive. To compare the performance of the different methods, we, thus, plot in Fig. 15 the energy of the self-consistent electric field, for both the TSI (left) and DTSI (right) test-cases. Here, the first observation is that the results present a high level of variability for the different methods. However, not all methods perform equally. If we consider that in both test-cases,

the fine FBL method (purple curve) can be used as reference, then we find that the curves of the FBL- δf scheme (in red) are of reasonable accuracy. More precisely, they are less damped than those of the coarse FBL scheme (in green), and they show less oscillations than those of the coarse full- f PIC scheme (in blue). In the TSI test case, the curve of the basic δf scheme (in orange) shows a similar behavior with slightly more oscillations, while in the DTSI test case, the oscillations are of very high amplitude (close to the actual level of the energy itself). As a result, we, thus, find that for the DTSI test case the new scheme also shows a superior accuracy in terms of energy approximation.

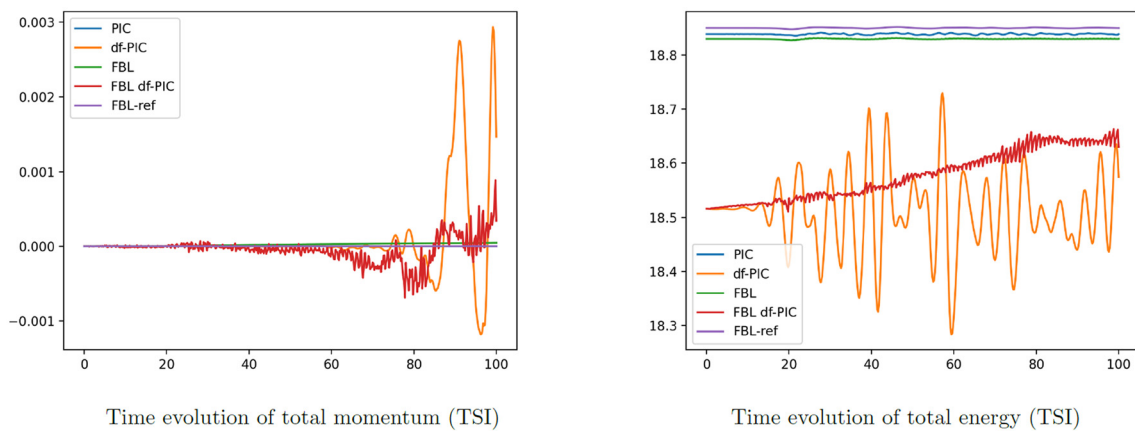


FIG. 14. Evolution of the total momentum (left) and energy (right), plotted as a function of time t , for the TSI test case.

06 September 2024 12:56:09

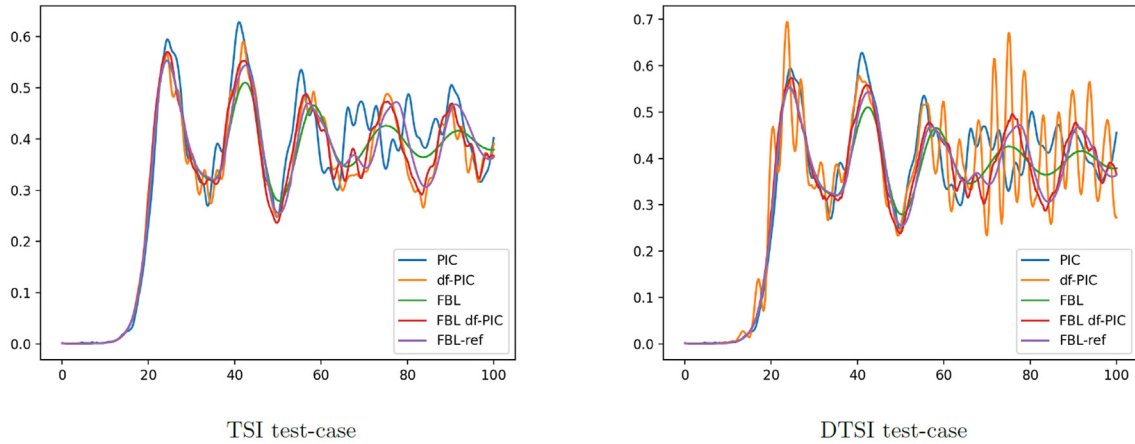


FIG. 15. Evolution of the self-consistent electric energy, plotted as a function of time t , for the TSI (left) and DTSI (right) test cases.

F. Reduction of the statistical errors

We conclude our numerical experiments by considering the empirical variances for the total particle number and current, as defined by Eqs. (4.80) and (4.81) in Ref. 8, namely,

$$\sigma_N^2 = \frac{N_p^2}{N_p - 1} \left[\sum_{k=1}^{N_p} (\delta w_k)^2 - \left(\sum_{k=1}^{N_p} \delta w_k \right)^2 \right] \tag{47}$$

and

$$\sigma_J^2 = \frac{N_p^2}{N_p - 1} \left[\sum_{k=1}^{N_p} (v_k \delta w_k)^2 - \left(\sum_{k=1}^{N_p} v_k \delta w_k \right)^2 \right], \tag{48}$$

respectively. We note that here the weight normalization differs from that of Ref. 8: For δf and full- f simulations (where the weights are constant), we have

$$\delta w_k^n \approx \frac{1}{N_p} \frac{\delta f^n(z_k)}{g^n(z_k)} \quad \text{and} \quad w_k^n = w_k^0 = \frac{1}{N_p} \frac{f^0(z_k)}{g^0(z_k)},$$

respectively, see (23) and (12). In particular, $\sigma_N = 0$ for full- f runs using a sampling distribution $g^0 = f^0 / (\int f^0(z) dz)$.

Following the interpretation of the δf method as a variance reduction technique,⁴ these respective quantities may be used as quantitative indicators of the statistical errors associated with the Monte Carlo approximation of the total density $\int \int f(t, x, v) dx dv$ and normalized current $\int \int f(t, x, v) v dx dv$, see Ref. 8, Sec. 2.

In Fig. 16, we plot these quantities for the two test cases, and for the three coarse PIC schemes, namely, the coarse full- f PIC, the basic δf , and the FBL- δf scheme. For the TSI test case corresponding to the top plots, we observe that the δf schemes indeed result in a reduction in the current variance σ_J , of a factor between two and three (the full- f density variance is zero as previously noticed). We also note a secular growth in the case of the FBL- δf scheme: This is probably due to the slow diffusion of the bulk density caused by the periodic remappings and should be mitigated by proper re-initialization methods for the

bulk density. We refer to the perspective section below for a discussion on possible improvements.

For the DTSI test case, we observe similar values for the current variances σ_J of the full- f FBL- δf PIC simulations (with values around 20 and 10, respectively). In contrast, strong oscillations are visible in the variances of the basic δf PIC scheme, which are closely correlated with the deviation in velocity already seen in Fig. 4, and the associated offset with the initial density. The amplitude of these oscillations (around 40) confirms our previous observations that a basic δf approach is no longer justified when the density deviates significantly from its initial profile and further validates the proposed method in such regimes.

VII. CONCLUSION

A. Summary

In this article, we have presented a δf PIC method where the bulk density is represented by smooth B-splines on a coarse grid and periodically updated with a forward-backward Lagrangian (FBL) method. This method amounts in updating the spline coefficients of the bulk density with a backward semi-Lagrangian method, where the backward trajectories are obtained by local inversions of the forward trajectories provided by the PIC code. In these inversions, the local backward flow is represented as a quadratic mapping whose Jacobian and Hessian matrices are computed from the relative displacements of regular passive markers pushed forward by the PIC code. After describing the representation of the bulk and variation density in terms of weighted collections of shape functions with variable width and recalling the general form of full- f and δf particle approximations to transport problems, we have provided a detailed description of a remapped δf scheme where the bulk density updates are performed using the FBL method. Finally, we have studied the basic performance of the method on two textbook examples involving one-dimensional (i.e., 1D1V phase-space) electrostatic problems: a simple two-stream instability where the deviation from the initial density is mild and does not justify any update of the bulk density, and a driven one where an oscillating electric field induces strong displacements of the phase-space density in velocity. In the latter case, the basic equilibrium/deviation separation assumption is no longer valid and the δf method fails as

06 September 2024 12:56:09

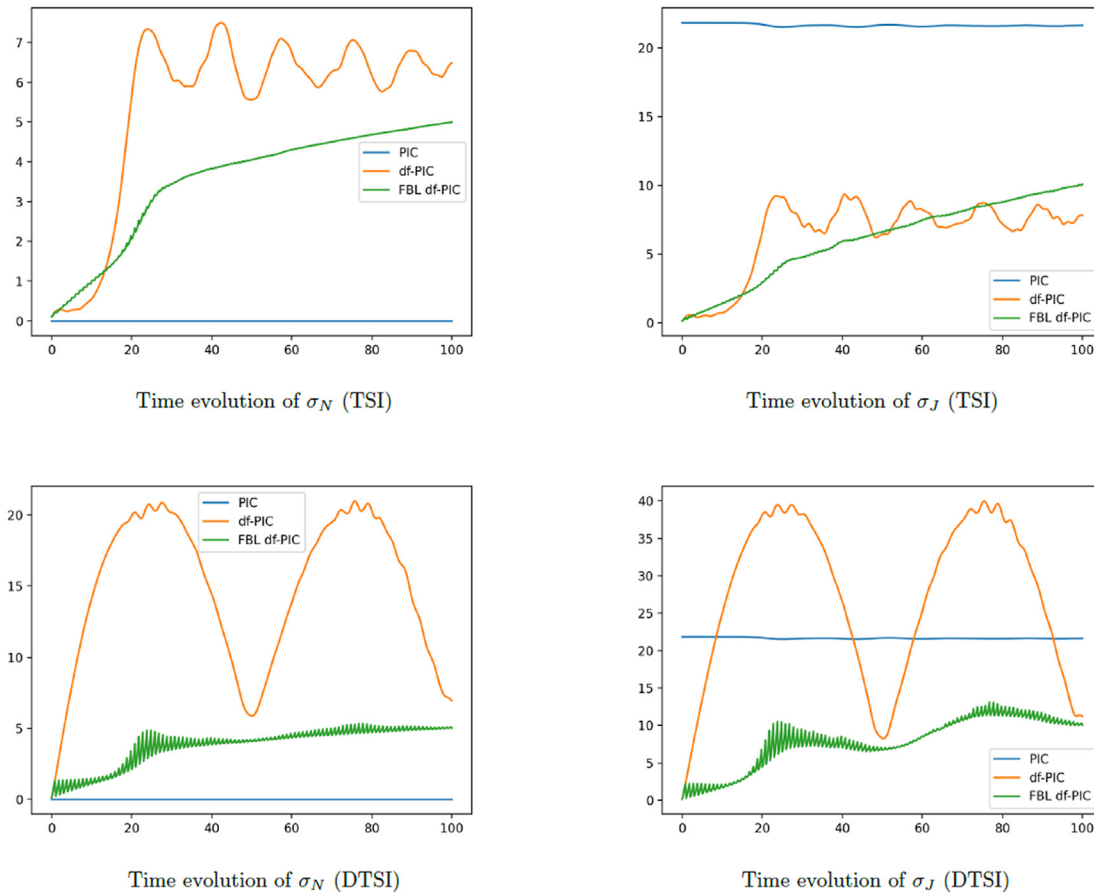


FIG. 16. Evolution of the empirical variances, plotted as a function of time t , for the TSI (top row) and DTSI (bottom row) test cases.

expected to correctly resolve the solution. In contrast, the FBL- δf method shows good results and a measurable reduction of the statistical error.

B. Novelty of the proposed approach

Although we did not compare the numerical performance of our method with the direct projection scheme of Ref. 20, we believe that opting for semi-Lagrangian updates of the bulk density is a significant novelty which has the key feature that it does not rely on an accurate particle approximation of the total density, but rather on accurate particle trajectories. Two typical configurations may illustrate why this difference is meaningful: The first one is a regime where a localized plasma blob moves through an otherwise empty phase space. In a region crossed by the blob, the bulk density should ideally go from zero to some positive density and then go back to zero as the blob leaves the region. With a direct projection scheme, however, this cannot happen: As the δf particles enter the region, the bulk density will be updated by their coarse projection and take a positive value, but as they leave, they will not be able to exactly compensate the smooth bulk density (and neither will their projection), because they have precisely moved away from it. As a result, the updated bulk density will retain a

spurious residual of that fine discrepancy, which will remain there since the δf particles have now left. Another problematic scenario is the opposite one, where both the density and the transport flow are smooth in phase space. If the bulk density grid is not too coarse, then one will always find some phase-space cells devoid of markers: On such cells, the projected δf density will be zero by construction, leading to strong oscillations. Because the forward-backward Lagrangian approach relies on the particle trajectories and a smooth grid-based bulk density, it is exempt of both these pitfalls: For the case of a moving blob, this has already been seen on the DTSI test case where the external drive moves the plasma density over large portions of the phase space, with Fig. 7 showing a clean transport of the bulk density (e.g., between times $t'' = 0$ and 25). For the second one, the key point is that a smooth flow leads to accurate backward trajectories, according to the analysis of Ref. 21. On every grid node, the new value will, thus, correspond to that of the previous bulk density at a point where the smooth spline density should be also accurate, independent of the markers resolution.

We also point out that this approach not only allows to approximate general bulk densities, but it can be applied to other representations, as long as their evaluation is easily computable: Instead of B-splines, one could, for instance, use piecewise polynomials or a collection of Fourier modes. The semi-Lagrangian updates would then

involve specific algorithms (such as interpolation or FFT) for updating the weights, based on nodal evaluations of the previous density at the feet of the backward characteristics computed by the FBL scheme.

C. Perspectives

Moving forward, an important direction of research is to extend the present method to more challenging problems such as higher-dimensional ones. We are, in particular, interested in gyrokinetic models where the transport phase space has typically four dimensions. In their current form, the various algorithms (namely, the FBL flow reconstruction and the spline quasi-interpolation) can be directly applied to a 4D phase space; however, preliminary numerical simulations have shown that typical transport flows in higher dimensions have an anisotropic smoothness which is not well captured by our isotropic algorithms. Our plan is, thus, to design extensions of the current approach that will be able to efficiently reconstruct high-dimensional characteristic flows with anisotropic smoothness.

Another improvement will be to design good re-initialization schemes for the bulk density that could be applied on larger time scales to counter the numerical diffusion inherent to the coarse semi-Lagrangian updates, as well as the secular growths in the energy error and in the variance, observed in Figs. 14 and 16. Adaptive approximation methods do exist which have already shown good results for particle denoising, such as adaptive wavelet or Fourier filters^{29,30} kernel density estimation algorithms,³¹ which automatically select the optimal width for the particle shape functions, or sparse-grid adaptive methods,³² which have also proven effective for particle denoising in higher dimensions.

ACKNOWLEDGMENTS

The authors thank Roman Hatzky for fruitful discussions on statistical error reduction, as well as the anonymous reviewers for their comments, which have improved the presentation of our results. This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014–2018 and 2019–2020 under grant agreement (No. 633053). The views and opinions expressed herein do not necessarily reflect those of the European Commission.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Martin Campos Pinto: Conceptualization (lead); Methodology (equal); Software (lead); Supervision (lead); Writing – original draft (lead); Writing – review & editing (equal). **Merlin Pelz:** Conceptualization (supporting); Methodology (equal); Software (supporting); Writing – original draft (supporting); Writing – review & editing (equal). **Pierre-Henri Tournier:** Conceptualization (supporting); Methodology (equal); Software (supporting); Writing – original draft (supporting); Writing – review & editing (equal).

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

- M. Kotschenreuther, *Bull. Am. Phys. Soc.* **33**, 2017 (1988).
- A. M. Dimits and W. W. Lee, *J. Comput. Phys.* **107**, 309–323 (1993).
- S. E. Parker and W. W. Lee, *Phys. Fluids B* **5**, 77–86 (1993).
- A. Y. Aydemir, *Phys. Plasmas* **1**, 822–831 (1994).
- R. Hatzky, T. M. Tran, A. Könies, R. Kleiber, and S. J. Allfrey, *Phys. Plasmas* **9**, 898 (2002).
- A. Mishchenko, R. Hatzky, and A. Könies, *Phys. Plasmas* **11**, 5480 (2004).
- A. Mishchenko, A. Bottino, R. Hatzky, E. Sonnendrücker, R. Kleiber, and A. Könies, *Phys. Plasmas* **24**, 081206 (2017).
- R. Hatzky, R. Kleiber, A. Könies, A. Mishchenko, M. Borchardt, A. Bottino, and E. Sonnendrücker, *J. Plasma Phys.* **85**, 905850112 (2019).
- Y. Chen and S. E. Parker, *Phys. Plasmas* **14**, 2301 (2007).
- T. Vernay, S. Brunner, L. Villard, B. F. McMillan, S. Jolliet, T. M. Tran, A. Bottino, and J. P. Graves, *Phys. Plasmas* **17**, 122301 (2010).
- E. Sonnendrücker, A. Wachter, R. Hatzky, and R. Kleiber, *J. Comput. Phys.* **295**, 402 (2015).
- G. Brochard, R. Dumont, H. Lütjens, and X. Garbet, *Nucl. Fusion* **60**, 086002 (2020).
- E. Lanti, N. Ohana, N. Tronko, T. Hayward-Schneider, A. Bottino, B. F. McMillan, A. Mishchenko, A. Scheinberg, A. Biancalani, P. Angelino, S. Brunner, J. Dominski, P. Donnel, C. Gheller, R. Hatzky, A. Jocksch, S. Jolliet, Z. X. Lu, J. P. Martin Collar, I. Novikau, E. Sonnendrücker, T. Vernay, and L. Villard, *Comput. Phys. Commun.* **251**, 107072 (2020).
- A. Biancalani, A. Bottino, A. Di Siena, Ö. Gürçan, T. Hayward-Schneider, F. Jenko, P. Lauber, A. Mishchenko, P. Morel, I. Novikau, F. Vannini, L. Villard, and A. Zocco, *Plasma Phys. Controlled Fusion* **63**, 065009 (2021).
- B. Rettino, T. Hayward-Schneider, A. Biancalani, A. Bottino, P. Lauber, I. Chavdarovski, M. Weiland, F. Vannini, and F. Jenko, *Nucl. Fusion* **62**, 076027 (2022).
- G. Dif-Pradalier, G. Hornung, P. Ghendrih, Y. Sarazin, F. Claret, L. Vermare, P. Diamond, J. Abiteboul, T. Cartier-Michaud, C. Ehrlacher, D. Estève, X. Garbet, V. Grandgirard, Ö. Gürçan, P. Hennequin, Y. Kosuga, G. Latu, P. Maget, P. Morel, C. Nordsieck, R. Sabot, and A. Storelli, *Phys. Rev. Lett.* **114**, 085004 (2015).
- V. Grandgirard, J. Abiteboul, J. Bigot, T. Cartier-Michaud, N. Crouseilles, G. Dif-Pradalier, C. Ehrlacher, D. Esteve, X. Garbet, P. Ghendrih, G. Latu, M. Mehrenberger, C. Nordsieck, C. Passeron, F. Rozar, Y. Sarazin, E. Sonnendrücker, A. Strugarek, and D. Zarzoso, *Comput. Phys. Commun.* **207**, 35 (2016).
- M. Dorf and M. Dorr, *Contrib. Plasma Phys.* **60**, e201900113 (2020).
- N. R. Mandell, A. Hakim, G. W. Hammett, and M. Francisquez, *J. Plasma Phys.* **86**, 905860109 (2020).
- S. J. Allfrey and R. Hatzky, *Comput. Phys. Commun.* **154**, 98–104 (2003).
- M. Campos Pinto and F. Charles, *SMAI J. Comput. Math.* **4**, 121 (2018).
- S. Colombi and C. Alard, *J. Plasma Phys.* **83**, 705830302 (2017).
- E. Sonnendrücker, J. Roche, P. Bertrand, and A. Ghizzo, *J. Comput. Phys.* **149**, 201 (1999).
- P.-A. Raviart, *Numerical Methods in Fluid Dynamics*, Lecture Notes in Mathematics (Springer, Berlin, 1985), pp. 243–324.
- C. K. Birdsall and A. Langdon, *Plasma Physics via Computer Simulation* (Adam Hilger, IOP Publishing, 1991).
- C. Chui and H. Diamond, *Numer. Math.* **57**, 105 (1990).
- Z. Cheng and G. Knorr, *J. Comput. Phys.* **22**, 330–351 (1976).
- J. A. Rossmannith and D. C. Seal, *J. Comput. Phys.* **230**, 6203 (2011).
- R. N. van den, D. del Castillo-Negrete, K. Schneider, M. Farge, and G. Chen, *J. Comput. Phys.* **229**, 2821 (2010).
- Y. Gao, M. Ku, T. Qian, and J. Wang, *J. Comput. Appl. Math.* **324**, 204 (2017).
- W. Wu and H. Qin, *Phys. Plasmas* **25**, 102107 (2018).
- S. Muralikrishnan, A. J. Cerfon, M. Frey, L. F. Ricketson, and A. Adelman, *J. Comput. Phys.: X* **11**, 100094 (2021).