Version of Record: https://www.sciencedirect.com/science/article/pii/S0010465518303606
Manuscript_5527deff848fbf974640dd1f7eedfc2a

Projective multiscale time-integration for electrostatic particle-in-cell methods.

P. Cazeaux[a], J.S. Hesthaven[b]

[a] *Department of Mathematics, University of Kansas, Lawrence, KS 66049, USA*
[b] *Ecole Fédérale Polytechnique de Lausanne, CH-1015 Lausanne, Switzerland*

Abstract

The simulation of problems in kinetic plasma physics are often challenging due to strongly coupled phenomena across multiple scales. In this work, we propose a wavelet-based coarse-grained numerical scheme, based on the framework of Equation-Free Projective Integration, for a kinetic plasma system modeled by the Vlasov-Poisson equations. A kinetic particle-in-cell (PIC) code is used to simulate the meso scale dynamics for short time intervals. This allows the extrapolation over long time-steps of the behavior of a coarse wavelet-based discretization of the system. To validate the approach and the underlying concepts, we perform two 1D1V numerical experiments: nonlinear propagation and steepening of an ion wave, and the expansion of a plasma slab in vacuum. The direct comparisons to resolved PIC simulations show good agreement. We show that the speedup of the projective integration scheme over the full particle scheme scales linearly with the system size, demonstrating efficiency while taking into account fully kinetic, non-Maxwellian effects. This suggests that the approach is potentially interesting for kinetic plasma problems with a large separation of scales.

Keywords: Plasma physics, Coarse-grained time integration, Multiscale numerical methods, Equation Free Projective Integration, Particle In Cell methods

1. Introduction

The importance and complexity of plasma physics makes its simulation a crucial challenge across various domains of science, from astrophysics (stars, solar wind) to efforts in developing nuclear fusion reactors (tokamaks) to everyday human environment (lighting, industrial processes). Plasma phenomena are typically characterized by a complex multiscale character, as physics at scales separated by multiple orders of magnitude in space and time directly influences the global behavior of the plasma. Examples of such phenomenons include magnetic reconnection in the solar wind, in which large amounts of magnetic energy are released into the plasma by the breaking and reforming of magnetic field lines at extremely small length scales; or meso-scale plasma turbulence, which is known to have a significant influence on the global transport properties of plasmas in laboratory tokamaks. The direct modeling of such problems still defies numerical methods developed by computational scientists.

In this paper we study the application of an original multiscale computational technique, the Equation-Free Projective Integration method (EFPI) [1, 2, 3, 4], and its application to the simulation of plasmas. A commonly used approach to simulate strongly multiscale systems is to derive and discretize a set of reduced equations for macro-scale phenomenons by analytically averaging the contributions of smaller scales, e.g., using homogenization theory. This procedure typically relies on strong structural assumptions such as

Email addresses: pcazeaux@ku.edu (P. Cazeaux), jan.hesthaven@epfl.ch (J.S. Hesthaven)

© 2018 published by Elsevier. This manuscript is made available under the Elsevier user license
https://www.elsevier.com/open-access/userlicense/1.0/

periodicity [5], or stationary randomness. In the context of plasmas, reduced fluid models such as magneto-hydrodynamics can sometimes be used depending on the physical setting. The EFPI framework, on the other hand, does not rely on discretizing a closed set of reduced equations. Rather, full-scale and highly resolved simulations of the system are computed for short period of times, allowing the recovery of the slow macroscopic behavior which is then projected forward in time with large timesteps. This approach seeks to account for the effect of meso-scale physics on the macroscopic behavior even when no satisfying upscaling theory is known, assuming only that there is a sufficient separation of scales. Whether this approach can provide accurate, efficient simulations for kinetic plasma problems remains an open question and a question that we shall consider in this work.

In this paper we are particularly concerned with the simulation of collisionless plasmas for which a kinetic description is required. In this context, particle methods, in particular Particle-In-Cell methods [6] have traditionally been preferred to Eulerian (grid-based) methods as they allow a coarse, but reasonably precise description of phase-space. However, standard explicit PIC schemes impose stability conditions to guarantee that space and time steps be sufficiently small to resolve the Debye length and plasma periods which are the typical space and time scales at which the electrostatic force tends to restore local charge neutrality in the plasma. For the majority of applications, these are extremely small when compared to the typical scales of the plasma under investigation. The development of implicit PIC methods, which are free of such constraints, has been the focus point of many recent efforts. To avoid the expensive fully implicit resolution of the particle positions and the fields, alternative algorithms have been developed, e.g., the direct implicit method [7, 8, 9] where a two-step predictor-corrector approximation is implemented or semi-implicit methods [10, 11, 12]. In the implicit moment methods [13, 14], moment equations are used to predict the value of the fields. Alternatives include the hybrid method, see e.g. [15], where a particle model is used for some of the particle species and a fluid model for others; or the so-called *multiscale* PIC methods [16, 17] which use different step sizes for different spatial regions, depending on the electric gradient scale in each region. Recently, fully implicit PIC methods based on Jacobian-free iterative algorithms have been developed [18, 19, 20, 21, 22] and allow to nonlinear converge particle and fields, resulting in an overall computational speedup. Further acceleration in such schemes is possible when using moment equations as a reduced model to advance the fields e.g. for preconditioning [23], upscaling effective closures from the underlying particle model. Another recent development is the asymptotic-preserving methods [24, 25] that allow to better account for the stiffness due to the quasineutrality constraint.

We shall pursue the development of a different idea, utilizing the equation-free projective integration method with an explicit PIC code as the fine-scale solver. The basic idea is to advance a macroscopic, or coarse, representation of the solution by initializing a consistent mesoscopic, or fine-scale, state of the system (lifting operation). This fine description is then advanced for a number of time-steps of a fine-scale solver, here the PIC code, recording the evolution of the system at the coarse level by means of a restriction operator. A linear least squares fit is then computed for each coarse variable to evaluate its temporal derivative. The coarse representation is then projected forward in time using a much larger time step.

A first attempt at implementing an equation free procedure in this context was the EFREE procedure developed by Shay *et al.* [26] and applied to the nonlinear propagation and steepening of an ion-acoustic wave. In EFREE, both electron and ion distribution functions are assumed to be Maxwellian and the electrons are adiabatic. The coarse description is then simply the ion density, mean velocity and pressure discretized on a coarse grid. Despite neglecting all kinetic effects, the EFREE results are promising as the wave propagation is correctly reproduced during the initial phase. However as the wave steepens, kinetic aspects such as trapping and non-Maxwellian phase space structures appear and the assumptions of the EFREE procedure fails to adequately account for these. An effort to include such effects was proposed in [27], by using a coarse discretization of the full inverse cumulative ion distribution function. However, this approach did not show a strong linear scaling for the speedup with the ion acoustic wavelength as the EFREE procedure achieved. Achieving such a scaling appears necessary to establish the usability of the equation free projection integration paradigm in kinetic plasma simulations.

In this paper we propose a new method based on representing the ion distribution function by the use of a multiresolution wavelet analysis. More precisely, the distribution function is discretized using a coarse grid in physical space and a wavelet basis in velocity space. This allows to fully account for kinetic

effects by tracking the evolution of macro-scale phase-space structures. The motivation for introducing a multiresolution wavelet analysis, widely used in signal processing, is its multiscale nature which allows the local adaption driven by the smoothness of the function to be approximated [28]. One of their most successful applications has been signal denoising [29]. Wavelet-based density estimation (WBDE) has recently been applied to the estimation of particle distribution functions obtained from plasma simulations with PIC methods [30].

The noise reduction is typically obtained by thresholding the expansion coefficients in a wavelet basis, based on the fact that this expansion is assumed to be sparse, or more precisely, the signal is assumed to be well-approximated by a small number of large expansion coefficients. Note that the sparsity of the representation is naturally connected to the coarse-graining of the data in phase-space. In this first step of the wavelet-based projective integration method, we limit attention to a non-adaptive approach, i.e., the wavelet representation is truncated at some pre-defined level of resolution. In the future, an adaptative denoising algorithm could be used to improve on this aspect, allowing for data-driven improvement of the resolution in specific regions of the simulation grid [31].

The two fundamental properties, sparsity and denoising, appear particularly well suited for application in the equation-free projective integration framework. First, depending on the bin size used for estimation of the ion distribution function, important random fluctuations can appear in the measured coefficients, especially in phase space regions with low particle count. The smoothing operation from fine to coarse grid is typically not sufficient to correct for the strong noise in the simulations [26, 27]. On the other hand, the wavelet representation enables the computation of a much smoother approximation as we will show.

In addition, an important concern is the handling of PIC-generated statistical noise by the projective integration procedure. We show that by combining the restriction on a coarse grid in physical space, wavelet thresholding techniques in velocity space, quiet starts and projection along approximated characteristics, our proposed method reduces noise dramatically.

We test numerically our proposed method on two well-known one-dimensional problems: the propagation and steepening of an ion acoustic wave, as in [26, 27]; and the expansion of a Gaussian plasma in a vacuum, following [32]. Let us note that at the moment, our method is not necessarily more efficient for either of these problems than a traditional implicit or multiscale PIC methods described above. Rather, its potential lies in the potential of its extension to strong multiscale problems for which more conventional methods fail. However the method must first be developed and evaluated in simple settings, for which trusted results exist. Our numerical experiments confirm that reference solutions obtained by brute-force high resolution PIC simulations are well reproduced by the proposed wavelet-based method. By increasing resolution, the ion pressure and phase space features converge correctly unlike previous attempts [26].

While we have restricted here our application to one-dimensional examples, the method is in principle not restricted to such cases. However, beating the curse of dimensionality of the 6 dimensional Vlasov system will require new developments such as adaptativity in phase space, or the use of novel low-rank tensor formats recently introduced with grid-based Eulerian or semi-Lagrangian Vlasov solvers [33, 34]. Bypassing the use of a fine grid discretization for sampling and loading particles could also prove necessary, although this can be done in parallel for each spatial grid cell. Note that when high-frequency structure start to dominate the physics, separation of scales does not apply. This could be detected by the adaptative filtering, at which point one should revert to traditional approach, e.g. pure particle-in-cell with no upscaling.

The paper is organized as follows. In Section 2, we present the two-species Vlasov-Poisson model and the classical PIC scheme. In Section 3, we recall the equation-free projective integration (EFPI) framework and the original EFREE method developed in [26]. From this approach, we develop our proposed wavelet-based method. Section 4 is devoted to the presentation and discussion of the numerical results and a comparison between the brute-force explicit PIC and our proposed schemes is provided. These results show that the scheme is stable and able to deal with multiscale kinetic problems where the stiffness comes from the fast electron scale.

2. Presentation of the problem

2.1. The Vlasov-Poisson equations

The theoretical modeling of a two-species plasma, comprising electrons and one ion species, usually begins with the two-species Vlasov–Poisson system. Each species of ions and electrons is described by a distribution function, respectively $f^{\mathrm{i}}(x,v,t)$ and $f^{\mathrm{e}}(x,v,t)$. The superscripts i and e will refer to the ions and electrons, respectively, throughout the whole paper. The position and velocity variables x, v are such that $(x,v) \in \Omega \times \mathbb{R}^d$ with $\Omega \subset \mathbb{R}^d$, $d \leq 3$, and $t \geq 0$ is the time variable. We restrict our attention to the one dimensional case, with $d = 1$, $\Omega = [0, L]$ where $L > 0$ is the size of the system, and periodic boundary conditions are imposed. However, all the developments proposed here can be generalized to the full six dimensional Vlasov–Poisson problem.

The Vlasov-Poisson equations is expressed as

$$\begin{cases} \partial_t f^{\mathrm{i}} + v \cdot \nabla_x f^{\mathrm{i}} - \dfrac{e}{m_{\mathrm{i}}} (\nabla_x \phi) \cdot \nabla_v f^{\mathrm{i}} = 0, \\ \partial_t f^{\mathrm{e}} + v \cdot \nabla_x f^{\mathrm{e}} + \dfrac{e}{m_{\mathrm{e}}} (\nabla_x \phi) \cdot \nabla_v f^{\mathrm{e}} = 0, \end{cases} \tag{2.1}$$

where we denote $e > 0$ as the positive elementary charge, by m_{i} and m_{e}, respectively, the ion and electron masses, and by ϕ the electric potential. The electrostatic potential ϕ is recovered by

$$-\Delta \phi = \frac{e}{\varepsilon_0} (n^{\mathrm{i}} - n^{\mathrm{e}}) \tag{2.2}$$

where ε_0 is the vacuum permittivity, and n^{i}, n^{e} are the ion and electron densities, respectively, given as

$$n^{\mathrm{i}}(x,t) = \int_{\mathbb{R}^d} f^{\mathrm{i}}(x,v,t) \mathrm{d}v, \qquad n^{\mathrm{e}}(x,t) = \int_{\mathbb{R}^d} f^{\mathrm{e}}(x,v,t) \mathrm{d}v.$$

Two very important physical scales for this model are the Debye length λ and the electron plasma frequency ω_p, given as:

$$\lambda = \sqrt{\frac{\varepsilon_0 k_B T_0^{\mathrm{e}}}{e^2 n_0}} \qquad \omega_p = \sqrt{\frac{n_0 e^2}{\varepsilon_0 m_{\mathrm{e}}}}, \tag{2.3}$$

where n_0 is the plasma density scale, T_0^{e} is the electron temperature scale and k_B is the Boltzmann constant. In problems of interest, both the Debye length and the electron plasma period become very small compared to the macroscopic scales.

2.2. Classical particle-in-cell scheme

In order to numerically solve the Vlasov-Poisson equations, the explicit particle-in-cell (PIC) method, discussed at length in [6], is widely used due to its strong parallel scaling and relatively low computational cost compared to traditional Eulerian methods. This is particularly true for problems in high dimensions. The basic idea of the PIC method is to discretize the phase-space distribution function $f^{i,e}$ with weighted macro-particles:

$$f^{i,e}(t,x,v) \approx f^{i,e}_{N_p} = \sum_{j=1}^{N_p} \omega_j^{i,e} \delta(v - V_j^{i,e}) \psi(h^{-1}(x - X_j^{i,e})), \tag{2.4}$$

where N_p is the number of particles, ψ is a shape function, h is a spatial scale length of the support of the macro-particles, $X_j^{i,e}(t)$, $V_j^{i,e}(t)$ is the location of the j-th particle in phase space and $\omega_j^{i,e}$ is a weight which is defined at initialization. One should not consider each particle as a physical particle but rather as particle clouds. Each of these particles follows Newton's equations:

$$\begin{aligned} \dot{X}_j^{\mathrm{i}}(t) &= V_j^{\mathrm{i}}(t), \quad \dot{V}_j^{\mathrm{i}}(t) = -\frac{e}{m_{\mathrm{i}}} \nabla_x \phi_{N_p,h} \left(X_j^{\mathrm{i}}(t), t \right), \\ \dot{X}_j^{\mathrm{e}}(t) &= V_j^{\mathrm{i}}(t), \quad \dot{V}_j^{\mathrm{e}}(t) = \frac{e}{m_{\mathrm{e}}} \nabla_x \phi_{N_p,h} \left(X_j^{\mathrm{e}}(t), t \right). \end{aligned} \tag{2.5}$$

These equations are then discretized in time. The potential $\phi_{N_p,h}$, approximating the electrostatic potential ϕ, is determined at the particle locations by the following procedure, repeated at each time step. Given a fixed spatial grid of space step h, the nodal values of the density of each species is computed by projecting every particle to the grid nodal locations with appropriate weights, given by the shape of the macroparticle [6]. The total charge density is then used to determine the nodal values of the electrostatic potential on the mesh by solving the Poisson equation (2.2) numerically, using either finite differences or the fast Fourier transform. Field values at each particle location are then determined by interpolation.

2.3. Advantages and limitations

Due to their efficiency and simplicity, PIC-based numerical methods are a dominating tool used in the modeling of kinetic plasma physics phenomena. While the mathematical literature on PIC methods is limited, some a priori convergence results have been obtained, see e.g. [35, 36, 37]. Typically, the error scales with the number of particles, as $1/\sqrt{N_p}$ as would be expected. Nevertheless, it is well-known that to ensure stability, explicit PIC codes must resolve the local Debye length and the electron plasma period everywhere in the computational domain [6].

The main drawback of PIC methods is the well-known statistical sampling noise due to the low number of numerical particles in each cell [38] - a number that by design is orders of magnitudes smaller than the true number of particles in the plasma. This so-called *shot noise* has a significant impact on the use of PIC solvers as a fine-scale solver in the projective integration framework [26, 27]. Various noise-reduction techniques have been developed to reduce its impact on computations. We will in particular make use of the classical quiet start procedure [6, 37], which helps to ensure that the initial noise is small. Instead of sampling randomly, similar to that of a Monte Carlo method, or on a regular grid in phase space as studied in [35], a given number of particles are initialized in each cell: the (conditional) inverse cumulative distribution function of the distribution is used to assign the particle velocities, and the particle locations inside the cell are scrambled using a pseudo-random sequence to reduce correlations.

3. Projective integration scheme

In this section, we present the equation-free projective integration (EPFI) framework and we develop a new method for the solution of plasma problems. The major issues in the design of such a multiscale scheme, based on the PIC method, are the choice of a set of appropriate macroscopic variables, and the handling of the PIC-generated statistical noise.

3.1. Presentation of EFPI schemes

The simulation of strongly coupled multiscale systems remains a major challenge due to the coupling of physics, often known only at the meso-scale, and the dynamics of interest which may emerge at the macroscale. Often, direct computation of the solution of the original problem is out of reach due to the necessity for the discretization to resolve the smallest or fastest scales in the system. Traditional approaches for the simulation of systems that exhibit a significant separation of scales usually involve two stages. First, a set of reduced equations is derived to describe the system at the coarse or macroscopic scale. These reduced equations are then solved numerically and their solutions analyzed. For plasmas, such equations exist in the form of fluid closures which reduce the dimensionality of the problem, such as the equations of magneto-hydrodynamics. However, these rely on assumptions which are not always satisfied and leave out important mesoscale physics [39].

When such closed-form equations are not available, the equation-free approach, originally proposed by Kevrekidis *et al.* [1, 2, 3, 4], sidesteps the need for an explicit reduced model. Instead, the approach relies on short bursts of fine-scale simulations to determine the time derivatives of a coarse representation of the system. Even though macro-scale level equations may not be necessary, it is worth stressing that an efficient application of the equation-free procedure requires the system to be parameterizable by a set of coarse variables for which the evolution evolves on a slow manifold [2]. However, if the dynamic behavior of the system can be coarse-grained by a representation that is smooth at macroscopic scales in time and space,

then it is conceivable that the full system need only be simulated for a small number of fine-scale time steps to advance the coarse variables over a much larger time interval.

The success of the method depends critically on the correct choice of the coarse description of the system, as we discuss below in more detail. Low dimensionality is clearly important as the primary purpose is computational speedup. A *restricting* operator, allowing the transition between the fine to the coarse representation of the data, as well as a *lifting* operator to go from coarse to fine scale, must be determined. Of course, the use of a coarse representation implies that some fine scale information is lost through the restricting process. This data must be reconstructed when lifting to initialize the fine-scale solver, typically through interpolation techniques and physical knowledge of the system.

Remark 3.1. *Ultimately, a goal should be to develop error estimators to validate the fidelity of the coarse representation during the time integration. This is in particular the case in the context of plasma simulations, since one may not be able to assume that the relaxation properties on which the EFPI framework is based [2, 3] hold at all times. For example, quasineutrality or Boltzmann equilibria can be violated as shocks develop or kinetic effects become dominant. Such techniques would allow the adaption of the macroscopic projection timestep as well as the coarse resolution, or revert to full-scale simulations when, for example, short transitions are detected during which no scale separation can be exploited.*

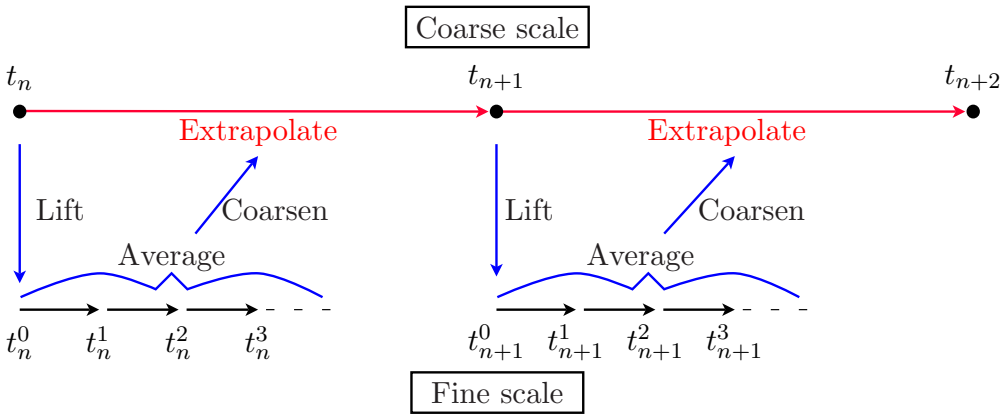


Figure 1: Equation free projective integration cycle

The basic forward Euler projective integration cycle is presented in Figure 1. This cycle comprises three main parts:

Initialization. At the beginning of the step, only the coarse representation of the system is available. This data is used to recover, by the lifting operator, a consistent fine state, which is used to initialize the fine-scale time stepper.

Micro-scale evolution. After its initialization, the fine-scale time solver is advanced forward N_e steps of size δt. After each step, the corresponding coarse variables are obtained by restricting the data.

Extrapolation. At the completion of the fine-scale steps, the evolution of the coarse variables is known at $N_e + 1$ instants corresponding to the fine-scale time steps. For each of the degrees of freedom, the best fit for the rate of change of the corresponding variable ψ is computed e.g. by a least squares algorithm. The macroscopic variables are projected forward by the macroscopic timestep Δt by the forward Euler formula

$$f(t + \Delta t) = f(t) + \Delta t \dot{f}(t). \tag{3.1}$$

Remark 3.2. *We note that it is possible to adapt this procedure to increase the accuracy of the time integration, i.e., we could also consider a trapezoidal leapfrog method as described in [26].*

3.2. Choice of macroscopic variables: EFREE approach

The core of the method is the distinction between a coarse-scale (macroscopic) and a fine-scale (mesoscopic) model representations. In this study as in [26, 27], the fine-scale representation consists in a fully kinetic description of the plasma, resolving the smallest relevant scales of space and time and using an explicit PIC scheme. A coarse representation must then be carefully chosen to approximate the system.

We first discuss the original so-called *EFREE* approach proposed in [26], to model ion-acoustic wave propagation and steepening. In EFREE, at the coarse-scale level of representation both ion and electron distribution functions are assumed to be shifted Maxwellian distributions, the electrons are assumed to be adiabatic, and the plasma is quasineutral. Thanks to these physical assumptions, the system is parameterized by the ion and electron densities $n^{\mathrm{i,e}}$, mean velocities $V^{\mathrm{i,e}}$ and temperatures $T^{\mathrm{i,e}}$ (or equivalently the pressures $P^{\mathrm{i,e}} = n^{\mathrm{i,e}} T^{\mathrm{i,e}}$):

$$f^{\mathrm{i,e}}(x,v) = C^{\mathrm{i,e}} n^{\mathrm{i,e}}(x) \exp\left(-\frac{1}{2} \frac{m_{\mathrm{i,e}} \left(v - V^{\mathrm{i,e}}(x)\right)^2}{k_B T^{\mathrm{i,e}}(x)} \right). \tag{3.2}$$

The ion variables are chosen as *active* variables, i.e. directly integrated in time. Electron variables are *passive* variables, determined from the ion variables. Due to adiabaticity, the electron temperature is assumed to be constant at all times:

$$T^{\mathrm{e}} = T_0^{\mathrm{e}}. \tag{3.3}$$

The electron density is a function of the electrostatic potential ϕ:

$$n^{\mathrm{e}} = n_0 \exp\left(\frac{e\phi}{k_B T_0^{\mathrm{e}}} \right), \tag{3.4}$$

where n_0 is a renormalization constant. Assuming that $n^{\mathrm{i}} \approx n^{\mathrm{e}}$ by quasineutrality, one deduces from (3.4) the potential ϕ as a function of n^{i}. Using (2.2) the electron density is then given as

$$\begin{cases} \phi = \dfrac{k_B T_0^{\mathrm{e}}}{e} \log\left(\dfrac{n^{\mathrm{i}}}{n_0} \right), \\ n^{\mathrm{e}} = n^{\mathrm{i}} - \dfrac{\varepsilon_0}{e} \Delta\phi, \end{cases} \tag{3.5}$$

i.e.

$$n^{\mathrm{e}} = n^{\mathrm{i}} - \lambda^2 \frac{\partial}{\partial x} \left(\frac{n_0}{n^{\mathrm{i}}} \frac{\partial n^{\mathrm{i}}}{\partial x} \right). \tag{3.6}$$

Finally the electron mean velocity is recovered by

$$V^{\mathrm{e}} = V^{\mathrm{i}}. \tag{3.7}$$

The spatial discretization of the *active* variables (in EFREE, n^{i}, V^{i} and P^{i}) uses a two-scale grid structure. The coarse representation of the data is defined as the values of the ion variables at nodes of a coarse grid, which resolves the macroscopic phenomenon. The data is then lifted to the fine grid which corresponds to the mesh employed for the PIC algorithm. This grid typically has a much finer resolution as it must resolve the Debye length λ. The *passive* electron variables are then determined using Eqs. (3.3–3.7), and the particle positions are initialized by the PIC code. During the fine-scale evolution step of the EFPI procedure, the variables are projected on the fine-scale grid using the usual PIC weighting scheme.

Increasing and decreasing the resolution by moving the data between the coarse and fine grids is realized in two steps:

Lifting operator. A linear interpolation scheme is employed to lift the data from the coarse to the fine grid. Let n_c be the number of nodes in the coarse grid, and n_f the number of nodes in the fine grid. For simplicity, both numbers are assumed to be powers of 2, and the interpolation procedure is done in successive factors of 2, repeated over l levels such that $n_f = 2^l n_c$. Let $f_k = f_k^{(0)}$, $k = 1, \ldots, n_c$ be the values of a variable f at the coarse grid points. The first level of interpolation is:

$$f_{2k}^{(1)} = f_k^{(0)}, \qquad f_{2k-1}^{(1)} = 0.5\left(f_{k-1}^{(0)} + f_k^{(0)}\right), \qquad \forall k = 1, \ldots, n_c, \tag{3.8}$$

where we assume periodic boundary conditions: $f_0^{(0)} = f_{n_c}^{(0)}$. This step is repeated l times to obtain the fine-scale data $f_k^{(l)}$, $k = 1, \ldots, n_f$.

Restriction. To restrict the data obtained by projection from the particle positions on the fine grid, a linear smoothing scheme is employed. Knowing the values $f_k^{(l)}$, $k = 1, \ldots, n_f = 2^l n_c$ of the variable f at the fine grid points, the first level of restriction gives the values

$$f_k^{(l-1)} = 0.25\left(f_{2k-1}^{(l)} + 2f_{2k}^{(l)} + f_{2k+1}^{(l)}\right), \qquad \forall k = 1, \ldots, 2^{l-1} n_c, \tag{3.9}$$

also assuming periodic boundary conditions $f_{n_f+1}^{(l)} = f_1^{(l)}$. This operation is repeated until the coarse level is reached.

Remark 3.3. *Restriction of the data to a coarse grid is essential to realize a speedup through EFPI integration [26]. Here, it also plays a crucial role by smoothing the statistical fluctuations of the variables inherent in the PIC scheme for the fine-scale kinetic representation of the data. As we will illustrate below, random fluctuations in the computed quantities appear to be substantially less pronounced when the EFPI-accelerated schemes are used as compared to a traditional PIC schemes. However, it should be noted that levels of fluctuations in the net charge $n^{\mathrm{i}} - n^{\mathrm{e}}$, the electric field, or the potential remain too important to allow their use as active variables.*

We refer to [26] for more details on this EFREE procedure. The numerical results presented in that work show good agreement with fully resolved PIC simulations for the propagation of an ion acoustic wave. In particular, the propagation speed of the wave is very well reproduced and the numerical speedup obtained with the EFREE procedure is promising, particularly since a macroscopic CFL condition, related only to the ion sound speed c_s and the coarse grid step Δx_c, appears as the limiting factor for the coarse time step Δt:

$$2c_s \Delta t \lesssim \Delta x_c. \tag{3.10}$$

This property enables a linear scaling as one increases the system size, i.e. the wavelength of the ion acoustic wave.

However, problems appear as the wave steepens, a shock forms and the ion distribution exhibits strong non-Maxwellian features due to ion trapping. In particular, large differences in the ion pressure appear relatively fast. As already noted in [26], it is necessary to generalize the description to handle non-Maxwellian distribution functions to address these phenomena.

3.3. A wavelet-based equation-free scheme

To capture the (x, v) phase-space structures emerging during the time evolution of the kinetic PIC experiments, we propose to represent at the coarse level of discretization the full ion distribution function f^{i}, using a coarse-grained wavelet approximation in the velocity space. As in the EFREE method presented in Sec. 3.2, we assume that the electrons are adiabatic and that their distribution function is well approximated by a shifted Maxwellian. However, we lift the requirements of quasi-neutrality and, most importantly, the assumption of Maxwellian-distributed ions.

Choice of macroscopic variables and parameterization. The ion distribution is approximated using a two-scale phase space grid structure. Let V_{min}, V_{max} be an appropriately chosen bounds for the ion velocities. A fine phase space rectangular grid is then constructed using $n_{x,\mathrm{f}}$ points on the position axis and $n_{v,\mathrm{f}} = 2^{J_\mathrm{f}}$ points on the velocity axis. Let $\delta x = L/n_{x,\mathrm{f}}$ and $\delta v = (V_{max} - V_{min})/n_{v,\mathrm{f}}$ be the grid steps in position and velocity respectively. The ion distribution function is then approximated at this fine level as continuous and piecewise-linear in x and piecewise-constant in v, i.e.

$$f^{\mathrm{i}}(x,v) = \sum_{k_x=1}^{n_{x,\mathrm{f}}} \sum_{k_v=1}^{n_{v,\mathrm{f}}} f^{\mathrm{i,f}}_{k_x,k_v}\ \chi_{k_x}\left(\frac{x}{\delta x} - k_x\right) \xi_{k_v}\left(\frac{v - V_{min}}{\delta v} + \frac{1}{2} - k_v\right), \tag{3.11}$$

where

$$\chi_k(x) = \begin{cases} 1 - |x| & \text{if } |x + j n_{x,\mathrm{f}}| \leq 1 \text{ for some } j \in \mathbb{N}; \\ 0 & \text{else,} \end{cases} \qquad \text{and} \qquad \xi_k(v) = \begin{cases} 1 & \text{if } |v| \leq 1/2; \\ 0 & \text{else.} \end{cases} \tag{3.12}$$

The fine discrete representation is then composed of the $n_{x,\mathrm{f}} \times n_{v,\mathrm{f}}$ coefficients,

$$f^{\mathrm{i,f}} \equiv \left\{ f^{\mathrm{i,f}}_{k_x,k_v} \right\}_{\substack{1 \leq k_x \leq n_{x,\mathrm{f}} \\ 1 \leq k_v \leq n_{v,\mathrm{f}}}}. \tag{3.13}$$

A coarse representation can then be obtained efficiently as follows:

- The data is first restricted to the coarse spatial resolution with $n_{x,\mathrm{c}}$ points using the scheme (3.9), leading to an intermediate set of coefficients $f^{\mathrm{i},(l)} \equiv \left\{ f^{\mathrm{i},(l)}_{k_x,k_v} \right\}_{\substack{1 \leq k_x \leq n_{x,\mathrm{c}} \\ 1 \leq k_v \leq n_{v,\mathrm{f}}}}$ in the notation of Sec. 3.2.
- For a given position index $1 \leq k_x \leq n_{x,\mathrm{c}}$, the corresponding column of values is assimilated to a set of scaling coefficients in a wavelet basis at the scale J_f (where $n_{v,\mathrm{f}} = 2^{J_\mathrm{f}}$). The (periodized) discrete wavelet transform is then applied to compute the scaling coefficients at a coarse scale $J_\mathrm{c} \leq J_\mathrm{f}$ [40, 30]. All coefficients for scales strictly finer than J_c are then discarded, leading to retain only $n_{v,\mathrm{c}} = 2^{J_\mathrm{c}}$ coefficients for each position index k_x. We obtain thus a coarse representation,

$$f^{\mathrm{i,c}} \equiv \left\{ f^{\mathrm{i,c}}_{k_x,k_v} \right\}_{\substack{1 \leq k_x \leq n_{x,\mathrm{c}} \\ 1 \leq k_v \leq n_{v,\mathrm{c}}}}. \tag{3.14}$$

In this work, we employ the R-Coiflet wavelet family of order 4, chosen for its regularity and symmetry properties [40] in addition to having four vanishing moments. For completeness, some notions on wavelets and the wavelet transform are recalled in Appendix A.

The scaling coefficients $f^{\mathrm{i,c}}$ are chosen as the *active* variables, leading to a wavelet-based equation-free projective integration method.

Remark 3.4. *Note that non-physical negative values of the distribution function can be obtained due to the extrapolation procedure, or due to the non-positivity of the oscillating wavelet functions as observed in [30]. We propose and detail in Appendix C a procedure designed to correct this defect and allow resampling from the approximated distribution function, while conserving mass and momentum.*

To complete the description of the system, we determine the electron *passive* variables n^{e}, V^{e} and T^{e} by first computing the ion density and velocity n^{i} and V^{i} by integrating numerically (3.11) along the velocity coordinate. Using the assumption of adiabatic electrons (3.4), the electrostatic potential satisfies the nonlinear Poisson-Boltzmann equation:

$$-\Delta\phi = \frac{e}{\varepsilon_0}\left(n^{\mathrm{i}} - n_0 \exp\left(\frac{e\phi}{k_B T^{\mathrm{e}}_0}\right)\right). \tag{3.15}$$

Equivalently, we solve self-consistently for the rescaled potential $\widetilde{\phi} = e\phi/(k_B T_0^{\mathrm{e}})$ such that

$$-\lambda^2 \Delta \widetilde{\phi} + \exp(\widetilde{\phi}) = n^{\mathrm{i}}/n_0. \tag{3.16}$$

In practice, this equation is discretized on the coarse grid and solved iteratively using a Newton method. Then, the values for the electron variables are recovered from Eqs. (3.3), (3.4) and (3.7):

$$n^{\mathrm{e}} = n_0 \exp(\widetilde{\phi}), \qquad V^{\mathrm{e}} = V^{\mathrm{i}}, \qquad T^{\mathrm{e}} = T_0^{\mathrm{e}}. \tag{3.17}$$

Restriction operator. The process for restricting the PIC data to the coarse level is in principle quite simple: the histogram of the empirical ion distribution function (3.13) is constructed by projecting each particle using the PIC weighting scheme in space and a nearest-grid-point scheme for the velocity. However, to further reduce the error associated with the projective integration, it is known that one can take advantage of an appropriate choice of *co-evolving frame* associated with the solution [4].

We propose here to project the particles by transporting the phase space grid backwards along approximated characteristics from time t_{n+1} to $t_n + p\delta t$, where $t_n = n\Delta t$. An equivalent description is that the solution is approximately advected forward along the characteristics up to time t_{n+1}. This original procedure is detailed in Appendix B.

In practice, we sample for a few fine time steps the histogram of the approximately advected solution $\widetilde{f}^{\mathrm{i,f}}_{(n,n+1)}$ at times $t_n + p\delta t$, $0 \le p \le N_e$ (see Appendix B for details). The coarse representation $\widetilde{f}^{\mathrm{i,c}}_{(n,n+1)}$ of these coefficients is obtained at each step by the coarsening process described above. Note that by construction,

$$\widetilde{f}^{\mathrm{i,c}}_{(n,n+1)}(t_{n+1}) = f^{\mathrm{i,c}}(t_{n+1}).$$

Remark 3.5. *Note that this is a different approach than used in [26, 18, 41], where the distribution function is simply shifted by the wave velocity at the initialization stage. In our numerical experiments, this simple shifting did not yield satisfactory results due to the phase-space (as opposed to moments) representation, see also the discussion in Section 4.1.3.*

In the case of a strongly hyperbolic problem such as the Vlasov-Poisson problem, traveling with an approximate flow results in big improvements in terms of noise and accuracy since the derivative which is estimated is smaller in the new frame. In addition, our approach allows to naturally take into account the *transport component* of the equation.

Extrapolation. After completion of the fine-scale steps by the PIC integrator, the coarse variables obtained by the restriction operator described above are projected forward using a forward Euler scheme as in Eq. (3.1):

$$f^{\mathrm{i,c}}(t_{n+1}) = \widetilde{f}^{\mathrm{i,c}}_{(n,n+1)}(t_{n+1}) = \widetilde{f}^{\mathrm{i,c}}_{(n,n+1)}(t_n + N_e\delta t) + (\Delta t - N_e \delta t)\left(\frac{\mathrm{d}}{\mathrm{d}t}\widetilde{f}^{\mathrm{i,c}}_{(n,n+1)}\right). \tag{3.18}$$

In this expression, the derivative is estimated by a least-squares estimator, which writes

$$\frac{\mathrm{d}}{\mathrm{d}t}\widetilde{f}^{\mathrm{i,c}}_{(n,n+1)} = \sum_{p=0}^{\lfloor N_e/2 \rfloor} \frac{6}{N_e(N_e+1)(N_e+2)} \frac{(N_e-2i)}{\delta t}\left[\widetilde{f}^{\mathrm{i,c}}_{(n,n+1)}(t_n + (N_e-p)\delta t) - \widetilde{f}^{\mathrm{i,c}}_{(n,n+1)}(t_n + p\delta t)\right]. \tag{3.19}$$

Remark 3.6. *This projection scheme conserves the total charge. Indeed, the total charge is proportional to the sum of all coefficients of f^{i} or all the weights of the particles at any given step. This number is conserved by the PIC time-stepper and the restriction operator, and thus also equal to the sum of all coefficients of $\widetilde{f}^{\mathrm{i,c}}_{(n,n+1)}$ for $p = 0 \dots N_e$. The sum of all entries of $\frac{\mathrm{d}}{\mathrm{d}t}\widetilde{f}^{\mathrm{i,c}}_{(n,n+1)}$ is then zero by* (3.19)*, thus the total charge is conserved by the projection step* (3.18)*.*

Remark 3.7. *The forward Euler scheme* (3.18) *is only first order accurate. It is possible to use higher order timestepping for the projection step in EFPI, e.g. leap-frog or Runge-Kutta methods [2, 3]. However in numerical experiments for the wavelet-based EFPI scheme, implementing a higher order integrator did not yield any improvements in accuracy or stability, pointing to the fact that the timestepping error may not be the main component of the overall EFPI scheme error, but rather other error sources such as statistical sampling or the coarsening and lifting steps.*

Lifting and initialization of the next macro-step. The lifting step consists of three important stages:

- Increasing the phase space resolution of the data using the linear interpolation (3.8) and the inverse Discrete Wavelet Transform in (3.11);
- Loading the particles using the quiet start procedure.

Note that the loading scheme is particularly important as it controls the initial level of the noise. We detail this procedure in Appendix C.

4. Test cases

We now illustrate the performance of the proposed approach for solving the Vlasov-Poisson system by modeling two different test cases, and seek to compare the results obtained with the classical PIC scheme and with the proposed coarse time-stepping algorithm. Simulations with the original EFPI algorithm proposed by Shay *et. al.* [26] are also considered in the first test case, comprising an ion acoustic wave, for a comparison. The second test case is a Gaussian plasma expansion in a vacuum.

4.1. Ion acoustic wave test-case

The first test case is the propagation and nonlinear steepening of an ion acoustic wave in a Maxwellian two-species plasma. In previous studies [26, 27], other versions of EFPI-accelerated PIC-based codes were applied to this test case and we refer in particular to [26] for a detailed account of the theory and initial setup of the wave. We initialize the Vlasov-Poisson system with

$$f^{\mathrm{i,e}}(x,v,t=0) = C^{\mathrm{i,e}}\left(1+\delta n\ cos\left(2\pi x/L\right)\right)\exp\left(-\frac{1}{2}\left(\frac{v-\delta v\ cos\left(2\pi k\cdot x/L\right)}{v_{\mathrm{th}}^{i,e}}\right)^2\right), \tag{4.1}$$

where $C^{i,e}$ are renormalization constants, $\delta n = 0.2$ and $\delta v = 0.2$ control the initial amplitude of the perturbation. The initial thermal velocities are chosen as $v_{\mathrm{th}}^{\mathrm{i}} = 0.22$ and $v_{\mathrm{th}}^{\mathrm{e}} = 42.5$, since the condition $T_0^{\mathrm{e}} \gg T_0^{\mathrm{i}}$ is necessary to minimize Landau damping of the wave.

The set of parameters normalized for the XES1-based PIC code [6] are $\epsilon_0 = 1$, $\omega_p^{\mathrm{e}} = 1527.35$, $\omega_p^{\mathrm{i}} = 36$, $q_{\mathrm{e}}/m_{\mathrm{e}} = -1800$, $q_{\mathrm{i}}/m_{\mathrm{i}} = 1$. The time step is chosen as $\delta t = 0.0001667 \approx 0.25/\omega_p^{\mathrm{e}}$. The number of mesh cells $nx = n_f$ is adjusted depending on the system length L for a constant grid step $h = 1/128 \approx 0.28\lambda$. These parameters correspond to the same case as in [26]. Under these conditions, both the fast space and time scales (the Debye length and the electron plasma period) are resolved, and the PIC method is stable.

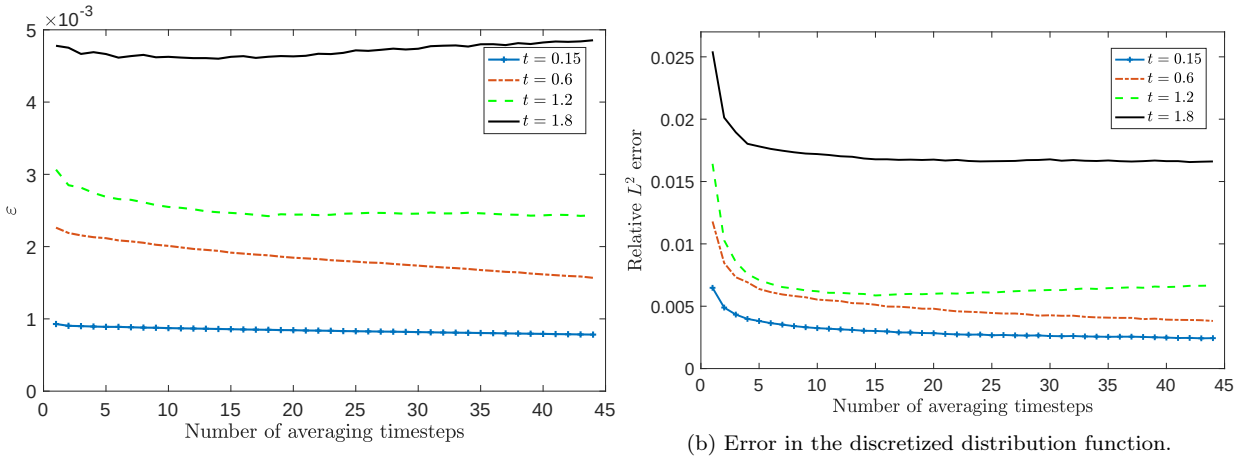
Parameters for the simulation runs used for this study are shown in Table 1.

Run	Type	L	N_{ppc}	$n_{x,\mathrm{f}}$	$n_{x,\mathrm{c}}$	$n_{v,\mathrm{f}}$	$n_{v,\mathrm{c}}$	$\Delta t/\delta t$
1	PIC	4	8192	512	–	–	–	–
2	EFPI	4	4096	512	512	2048	64	150
3	EFPI	4	2048	512	128	2048	64	250
4	EFPI	4	1024	512	32	2048	64	600
5	PIC	8	8192	1024	–	–	–	–
6	EFPI	8	4096	1024	1024	2048	64	150
7	EFPI	8	2048	1024	256	2048	64	250
8	EFPI	8	1024	1024	64	2048	64	600

Table 1: Simulation runs presented for this test case. L is the domain length, N_{ppc} the number of particles per species per cell for the underlying PIC code, $n_{x,\mathrm{f}}$ and $n_{x,\mathrm{c}}$ ($n_{v,\mathrm{f}}$ and $n_{v,\mathrm{c}}$) respectively the number of fine and coarse space (velocity) grid points, δt and Δt respectively the fine and coarse time steps.

Note that the number of particles per grid cell for each species is chosen as 8192 for the reference PIC runs, while it is reduced for the EFPI runs to 4096 particles per *coarse* grid cell to account for the better signal-to-noise ratio achieved through coarsening and short PIC runs with quiet starts.

4.1.1. Choice of EFPI numerical parameters



(a) Error in the density, ε.

(b) Error in the discretized distribution function.

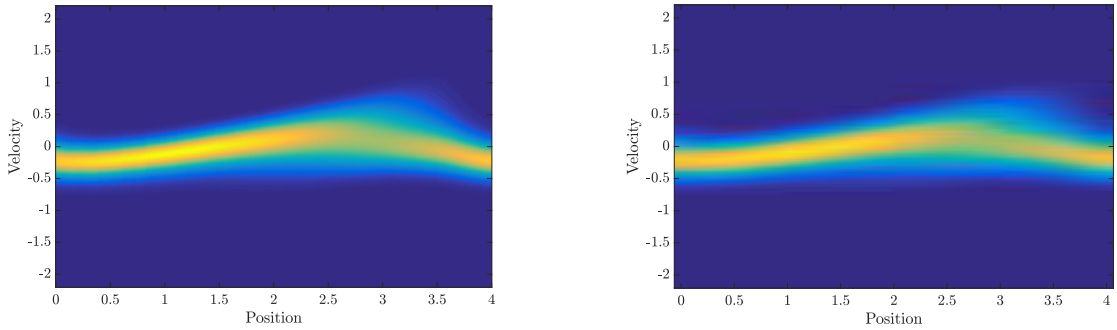
Figure 2: Extrapolation error as a function of the number of fine-scale time steps for each macro step projection cycle, Run 2.

The EFPI algorithm presents two main parameters, both of which require some tuning to obtain optimal results: the coarse time step Δt and the number of fine-scale time steps needed during each projective integration cycle N_e. As noted in [26], in the first case Δt is controlled by a Courant-type condition which is well understood.

However, the choice of N_e is more delicate. In [26], various choices ranging from $N_e = 10$ to $N_e = 20$ are analyzed numerically showing a strong effect on the error, with the simulation diverging quickly for $N_e \leq 14$ and almost indistinguishable results for $N_e \geq 16$. We investigate here the influence of N_e on both the relative L^2 error in the ion density function, noted ε (see also Eq. (33) in [26]), and the relative L^2 error in the coefficients of the discretized distribution function over the whole phase space, which are defined as:

$$\varepsilon = \sqrt{\frac{\sum_{j=1}^{n_{x,c}} (n_j^{EFPI} - n_j^{PIC})^2}{\sum_{j=1}^{n_{x,c}} (n_j^{EFPI})^2 + (n_j^{PIC})^2}}, \qquad \left\| f^{EFPI} - f^{PIC} \right\|_{L^2} = \sqrt{\frac{\sum_{k_x,k_v} (f^{\mathrm{i,c},EFPI}_{k_x,k_v} - f^{\mathrm{i,c},PIC}_{k_x,k_v})^2}{\sum_{k_x,k_v} (f^{\mathrm{i,c},EFPI}_{k_x,k_v})^2 + (f^{\mathrm{i,c},PIC}_{k_x,k_v})^2}}. \tag{4.2}$$

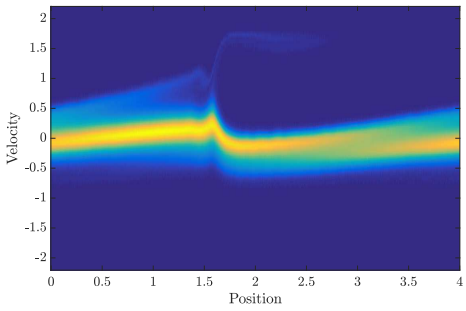
Figure 2 shows both error functions for Run 2 for various times up to $t = 1.8$ as a function of N_e. We observe that the method is stable for all choices of $N_e \geq 1$. Furthermore, the error stagnates for $N_e \geq 6$, a value which was stable with respect to the number of particles used. This value of $N_e = 6$ is used in all results presented in this section (Runs 1–8).



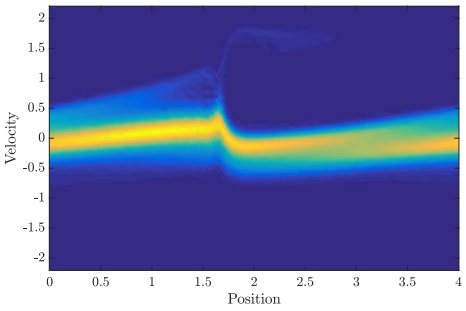
(a) Fully resolved PIC solution (Run 1)

(b) EFPI solution (Run 4, $n_c = 32$)

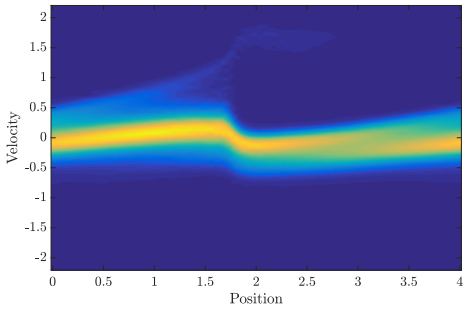
Figure 3: Ion distribution function plots, Runs 1 and 4, $t = 1.5$.



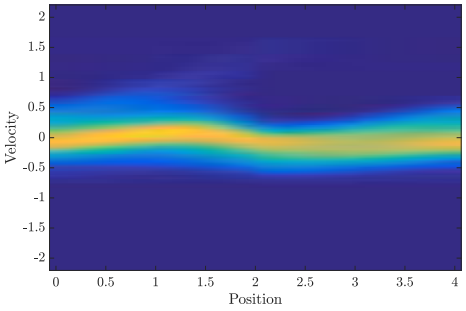
(a) Fully resolved PIC solution (Run 1)



(b) EFPI solution (Run 2, $n_c = 512$)



(c) EFPI solution (Run 3, $n_c = 128$)



(d) EFPI solution (Run 4, $n_c = 32$)

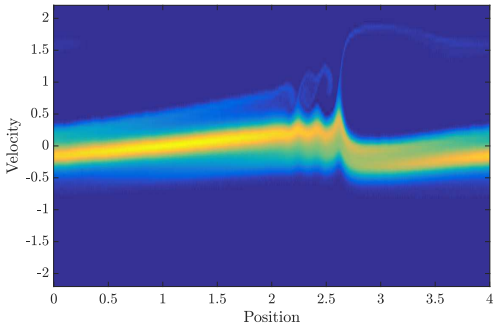
Figure 4: Ion distribution function plots, Runs 1–4, $t = 3.5$.

4.1.2. Numerical results

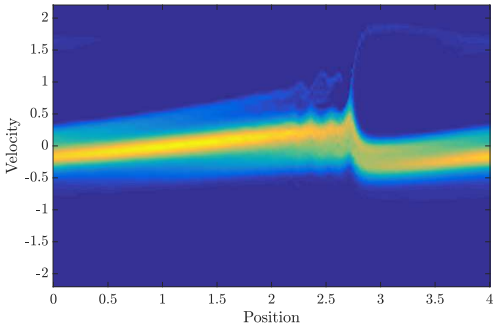
Figures 3, 4 and 5 show results for runs 1–4 (case $L = 4$) at times $t = 1.5$, 3.5 and $t = 4.5$, respectively. We show the unnormalized ion distribution function for the PIC scheme (run 1) and our proposed wavelet-based projective method (run 2–4). The highly resolved PIC simulation is taken as the reference. We can see that as the wave propagates, it deforms and steepens as a shock develops.

At time $t = 1.5$, the ion distribution function is very smooth. The different schemes are in good agreement with each other and all main features are well represented. The propagation speed of the ion wave is very well matched by the proposed method. Note that the upper velocity tail starts to deviate from the Maxwellian distribution. This result shows that the (linear) thresholding of the wavelet coefficients is very efficient and already very well adapted to use in the EFPI algorithm.

At time $t = 3.5$, a ion shock has fully developed. As can be seen on Figure 4, the accuracy of the wavelet-based EFPI solution is very dependent on the level of coarsening in space. Phase-space features of

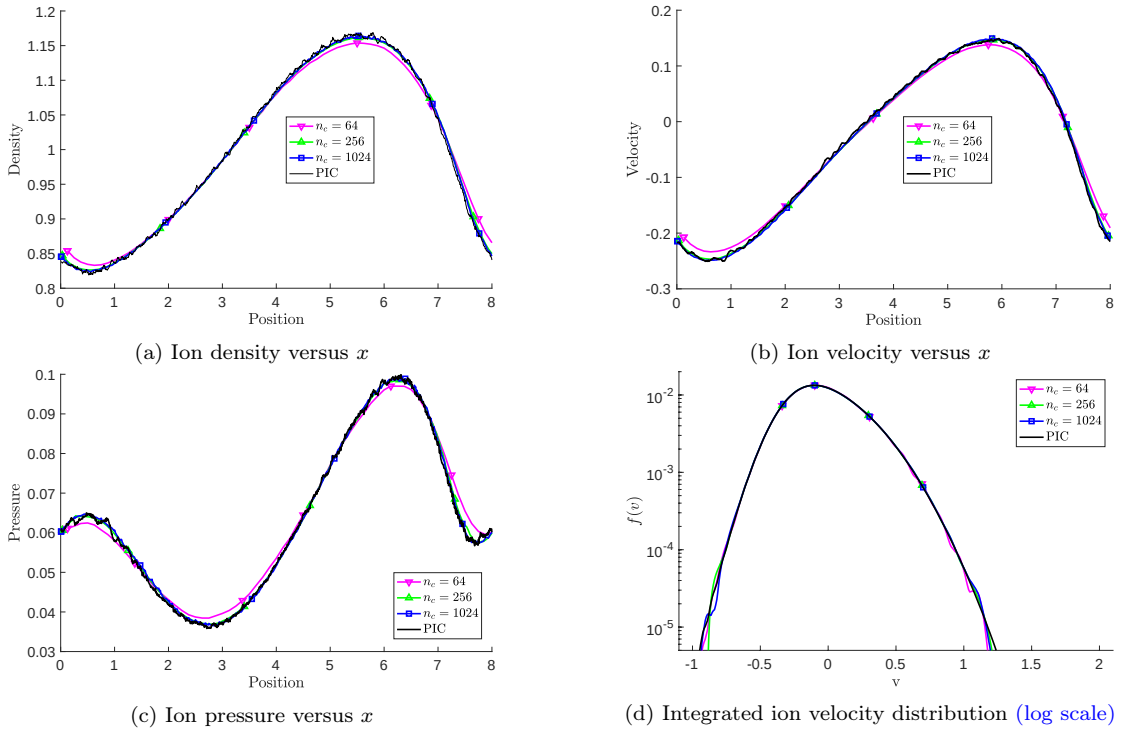


(a) Fully resolved PIC solution (Run 1)



(b) EFPI solution (Run 2, $n_c = 512$)

Figure 5: Ion distribution function plots, Runs 1–2, $t = 4.5$.



(a) Ion density versus x

(b) Ion velocity versus x

(c) Ion pressure versus x

(d) Integrated ion velocity distribution (log scale)

Figure 6: Comparison between moments obtained with PIC (Run 5) and EFPI (Runs 6–8) at $t = 3$.

the shock are very well reproduced when 512 grid points are used for the projective integration (no spatial coarsening), but are completely smoothed out when the solution is coarsened using 32 points.

At the final time $t = 4.5$, the reflected ions as well as smaller trailing shocks form distinctive features as seen on Figure 5, which are again well reproduced in the finer EFPI solution.

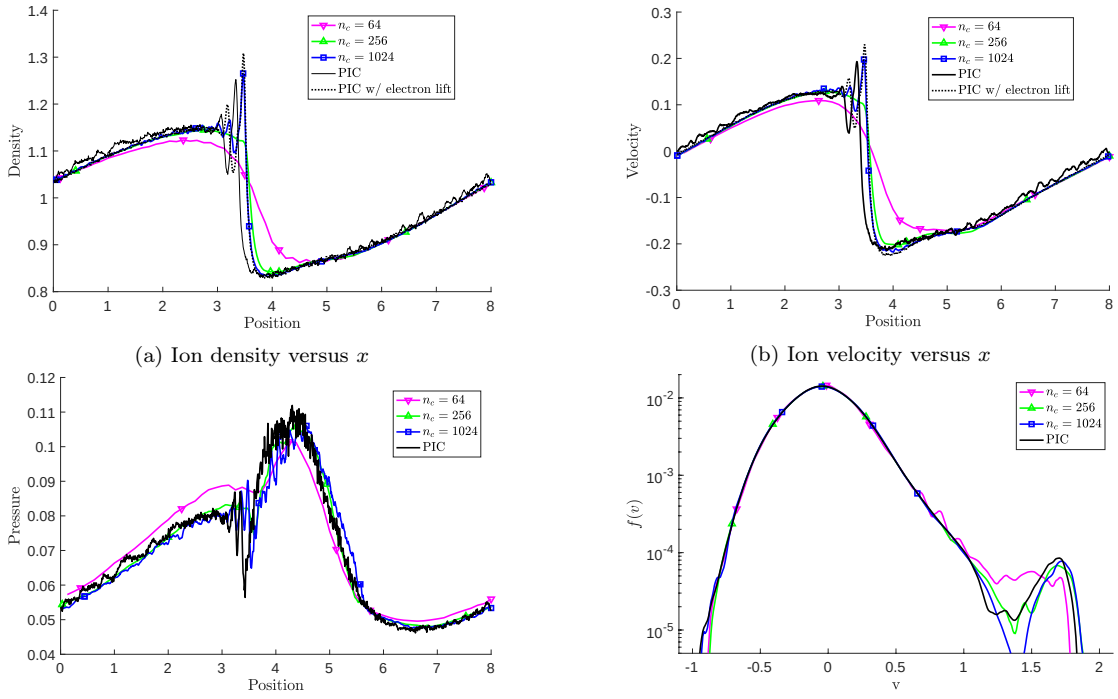
Remark 4.1. *This level of diffusion was not present in the EFREE study [26], where knowledge of the ion sound speed is used to track the ion sound wave in a uniformly co-moving frame. However this transformation did not yield more accurate results with the present method which tracks the full phase-space.*

Next, we compare the moments and the velocity distribution obtained for the case of a larger system length ($L = 8$) at times $t = 3$ and $t = 7$. At time $t = 3$, Figure 6, there is reasonable agreement between all the schemes, although the coarsest solution with $n_{\mathrm{x,c}} = 64$ shows some diffusion. In particular, the anharmonic shape for the ion pressure and the asymmetric velocity distribution are well reproduced.

At time $t = 7$, Figure 7, the accuracy of the EFPI solution is highly dependent on the resolution which is chosen. In particular, the narrow density and velocity peaks at the shock are well reproduced by the fine EFPI solution (1024 grid points), but absent from the coarser solutions. The overall shape is still well reproduced using only 256 grid points, but clearly overdamped by numerical diffusion when using only 64 points. The ion velocity distribution shows the same behavior, with the second peak corresponding to ions reflected by the shock well reproduced for the finer EFPI solutions.

4.1.3. Discussion

Strongly non-Maxwellian effects appear progressively during the ion acoustic wave propagation. In particular one observes reflected particles and a double-peaked ion distribution function at the shock. The simple system representation by shifted Maxwellians in the original EFREE implementation [26] lead to noticeable differences with the reference simulation. In particular, the ion pressure diverges quickly and does not develop the correct strongly anharmonic shape. We have illustrated here how the wavelet-based equation free method substantially improves on these results. Our projective integration scheme gives



(a) Ion density versus x

(b) Ion velocity versus x

(c) Ion pressure versus x

(d) Integrated ion velocity distribution (log scale)

Figure 7: Comparison between moments obtained with PIC (Run 5) and EFPI (Runs 6–8) at $t = 7$.

accurate results even in the strongly nonlinear regime, after the creation of the shock, while using a large macroscopic time step. This shows that the EFPI framework can be effective in a kinetic setting, provided a non-parametric representation of the general non-maxwellian ion distribution function is used.

Convergence analysis and modeling error. We present on Figure 8 the density error ε, as defined in (4.2), as a function of time for Runs 6–8. Solid lines indicate the error relative to the reference PIC solution (Run 5). The error increases slowly with time and is quite well controlled, after the initial period where noise dominates. Note that increasing the resolution from $n_{x,\mathrm{c}} = 256$ to 1024 grid points has little effect on the error apart from the very beginning, but the highly underresolved solution with $n_{x,\mathrm{c}} = 64$ is as expected

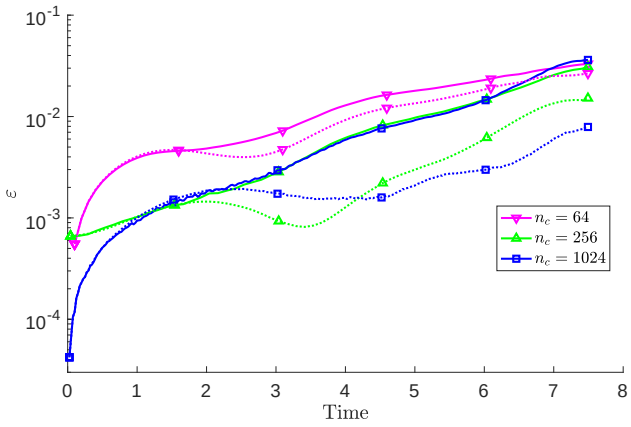


Figure 8: Error ε in the ion density compared to the fully resolved PIC solutions without (solid lines) and with (dotted lines) periodic reset of the electrons to the self-consistent adiabatic Maxwell-Boltzmann equilibrium.

Number of grid points	Maximum stable timestep for EFPI	
	Integration along characteristics	No integration along characteristics
512	0.028 ($170\delta t$)	0.005 ($30\delta t$)
128	0.058 ($350\delta t$)	0.012 ($75\delta t$)
32	0.104 ($625\delta t$)	0.050 ($300\delta t$)

Table 2: Maximum projective time steps ensuring stability over the time interval $0 \leq t \leq 5$ for the case $L = 4$.

much less accurate overall.

An important component of the error is that the wave speed obtained with EFPI is slightly too high. Comparing the density peaks at the shock front for the case where $L = 8$, we observe a position difference of approximately $+0.12$ with respect to the reference PIC solution at time $t = 7$ (see Fig. 7a). It is interesting to note that this difference can be completely explained by the assumption of Boltzmann-distributed Maxwellian electrons which is implicit in the EFPI method. Indeed, resonant electrons are trapped by the shock potential, inducing a slight deviation from the Maxwellian distribution [42]. This effect is neglected by the EFPI representation. To show this, we perform a full PIC simulation with the same parameters as Run 5, where in addition the electrons particles are periodically reset to the self-consistent Boltzmann equilibrium. More precisely, every 150 PIC timesteps, the electron phase space distribution is recalculated from the ion density using equations (3.16) and (3.17). The result is plotted in dotted lines on Figures 7a and 7b, and shows an very good agreement with the finest EFPI solution.

In addition, we plot on Figure 8 the density error with respect to this modified PIC solution, in dotted lines. We see that the error is much improved, especially at later times where Debye-scale structures appear at the shock front. At the final time $t = 8$, with a fully developed shock, the EFPI solution converges to this new modified PIC solution as the macro grid step and the macro time step are reduced.

Stability and the need for a particle deposition scheme with integration along approximate characteristics. Maximum stable time steps, listed in Table 2, show that the method can be used to directly exploit the large difference between electron and ion timescales, without the need for coarsening. When a larger macro grid step can be chosen, this can be exploited to use even bigger time steps.

An interesting test is to project $f^{\mathrm{i,c}}$ directly instead of the integration along the characteristics (3.18) employed here, using a straightforward deposition of the particles on the fine grid. As seen in Table 2, this approach requires the use of a much smaller time step, especially at higher resolution where the electric field becomes very strong at the shock front.

Remark 4.2. *Due to the use of a phase space grid, the proposed method resembles a Eulerian grid-based first-order timestepping scheme applied to the nonlinear kinetic transport system (2.1). The corresponding local Courant stability condition* $\frac{|v|\Delta t}{\Delta x} + \frac{|\nabla_x \phi|\Delta t}{\Delta v} \lesssim 1$ *naturally involves the local electric field, which grows nonlinearly as the shock front develops. Since numerical diffusion tends to smooth the resulting electric field peak, depending on the coarsening level, the resulting stability condition is not as easy to formulate as in the EFREE case [26] where the use of a spatial-only grid seems to yield a Courant condition based solely on the macroscopic ion sound wave speed. However, with the particle deposition scheme by integration along the characteristics the proposed method may be interpreted as an approximate semi-Lagrangian scheme for which a rigorous analysis may be quite difficult.*

Particle noise. Numerical experiments show that random-like fluctuations due to the PIC codes necessitate a careful treatment. As we have shown, the level of fluctuations in the active (ion) variables has been successfully reduced by our proposed wavelet-based approach due to (a) linear smoothing from fine to coarse grid, (b) denoising by linear or nonlinear thresholding of wavelet coefficients, (c) projection along approximate characteristics, (d) quiet start. This last factor appears particularly efficient since the fine-scale integration is only realized for a small number of timesteps, enabling the PIC error to scale as $\log(N_p)/N_p$ instead of $1/\sqrt{N_p}$ [37].

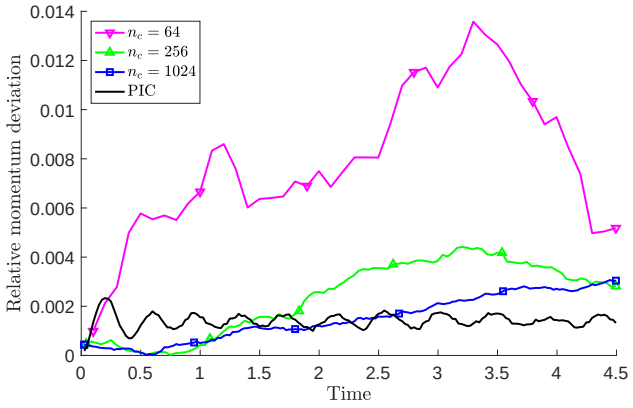


Figure 9: Relative total ion momentum deviation (Runs 5–8)

Conservation of moments. As noted in Remark 3.6, charge is conserved by the EFPI approach. This is not however the case for the momentum or energy. Indeed, the implicit representation of electrons makes this a difficult task since ion momentum alone is not a conserved quantity. We plot in Figure 9 the relative deviation of the total ion momentum with respect to its initial value. The reference curve oscillates around 0.02, as momentum is transfered to trapped electrons. This effect cannot be reproduced by the EFPI solutions which show larger deviations, but stay within reasonable bounds.

4.2. One-dimensional plasma expansion test-case

As a second test case, we consider a one-dimensional expansion problem, the expansion of a Gaussian plasma in vacuum [43, 44, 45, 46, 32]. Note that this is similar to the case of a thin uniform slab described in [47, 48], used as a test case e.g. in [25]. The ions initially uniformly occupy a Gaussian slab of characteristic radius $r_0 \ll L$, while the electrons are initialized as the adiabatic Maxwell-Boltzmann equilibrium in a self-consistent potential. The test problem requires the observation of the expansion of the ion slab, initialized with:

$$f^i(x, v, t = 0) = C^{\mathrm{i}} \exp\left(-\frac{1}{2} \left(\frac{v}{v^{\mathrm{i}}_{\mathrm{th}}} \right)^2 - \left(\frac{x}{r_0} \right)^2 \right), \tag{4.3}$$

where C^{i} is a renormalization constant, and the electron distribution function is deduced by solving the nonlinear Poisson-Boltzmann equation (3.16) and hypothesis (3.17). Note that this is the same procedure used when lifting the coarse ion data in the approach developed in this paper. In the quasineutral limit, an analytic self-similar solution exists [43, 45, 46] and the electron distribution function keeps a Maxwellian form with a time-dependent homogeneous temperature, an important fact for the applicability of the method presented in this paper.

The initial electron temperature is 1000 times higher than the initial ion temperature: the thermal velocities are set as $v^i_{\mathrm{th}} = 0.0316$, $v^{\mathrm{e}}_{\mathrm{th}} = 42.84857$. The set of numerical parameters normalized for the PIC code are, in code units, $\epsilon_0 = 1$, $\omega^{\mathrm{e}}_p = 42.8486$, $q_{\mathrm{e}}/m_{\mathrm{e}} = -1.0$, $\omega^{\mathrm{i}}_p = 1$, $q_{\mathrm{i}}/m_{\mathrm{i}} = 0.000554466$. The system length is $L = 1000$, and the initial value for the characteristic radius is $r_0 = 40$. Simulation parameters are

Run	Type	L	r_0	N_{ppc}	$n_{x,\mathrm{f}}$	$n_{x,\mathrm{c}}$	$n_{v,\mathrm{f}}$	$n_{v,\mathrm{c}}$	$\Delta t/\delta t$
9	PIC	1000	40	16384	4096	–	–	–	–
10	EFPI	1000	40	4096	4096	4096	2048	512	250
11	EFPI	1000	40	2048	4096	2048	2048	512	400

Table 3: Simulation runs presented for this test case. L is the domain length, r_0 the initial slab width, N_{ppc} the number of particles per species per cell, $n_{x,\mathrm{f}}$ and $n_{x,\mathrm{c}}$ ($n_{v,\mathrm{f}}$ and $n_{v,\mathrm{c}}$) respectively the number of fine and coarse space (velocity) grid points, δt and Δt respectively the fine and coarse time steps.

summarized in Table 3. As in the previous test case, the electron motion is much faster than the ion motion and we can exploit this separation of scales with EFPI, since the expansion occurs roughly at the speed of the ion acoustic wave, $r(t) \approx \sqrt{r_0^2 + 2c_{s0}^2 t^2}$ [32].

The Debye length is initially $\lambda = v_{th}^e/\omega_p^e = 1$ at the center of the plasma slab. The PIC time step is chosen as $\delta t = 0.005 \approx 0.21/\omega_p^e$ and the grid step $h = L/4096 \approx 0.244\lambda$, allowing to resolve the fast space and time scales.

4.2.1. Differences in the setup

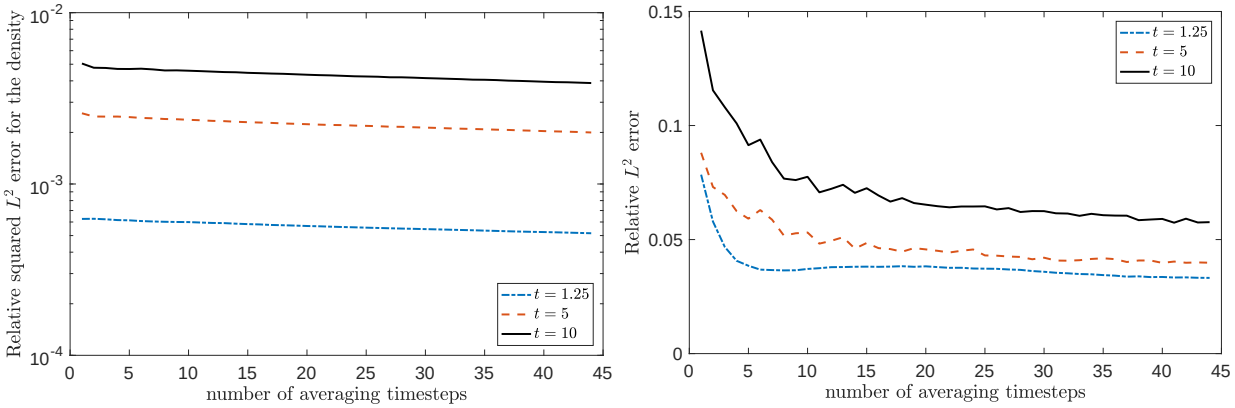
The boundary conditions in this example are different from the previous example: we simulate only a half-domain with a purely absorbing right boundary condition and an axis of symmetry on the left of the domain, as for example in [25]. In addition, the global kinetic energy of the electrons is now projectively integrated as an active variable to account for the progressive decrease in temperature as the expansion expands and freezes, as well as the total charge of each species to account for the possible escape of particles through the right boundary.

4.2.2. Choice of EFPI numerical parameters and numerical results

As in the ion acoustic wave test case, numerical experiments show that the coarse time step Δt is controlled by a Courant-type condition. To choose the number N_e of fine-scale time steps computed for each projective integration cycle, we repeat the analysis from the previous test case by measuring empirically the relative L^2 error in the ion density and discretized distribution functions depending on N_e, using the parameters from run 10. Results are shown in Fig. 10. The method is stable for all values of N_e, and the choice of $N_e = 10$ seems to give good results, and is used in all simulations presented in this section.

Figures 11 and 12 show the results for runs 9–11, respectively, at times $t = 20$ and $t = 50$. We plot the unnormalized ion distribution function, the ion density and the local charge density due to both ion and electrons. As time advances, the ion slab expands roughly at the speed of the ion acoustic wave [49].

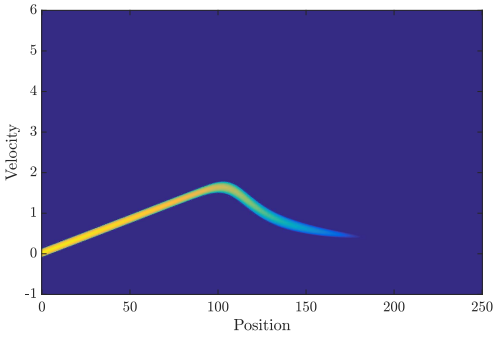
First, we observe that the proposed method gives a correct account of the speed of expansion of the ion acoustic wave, thus verifying that it correctly captures this wave as also seen earlier. As it expands, the plasma develops a double layer at the edges of the expansion which is accurately reproduced by the EFPI solution. At later times (see Fig. 12, $t = 50$), the solution exhibits wave breaking and the ion distribution function becomes clearly non-maxwellian, an effect which is also reproduced by the EFPI method. We note also that data has the same or lower level of noise compared to the reference PIC simulation, even as it uses 4 times less particles (see Table 3).



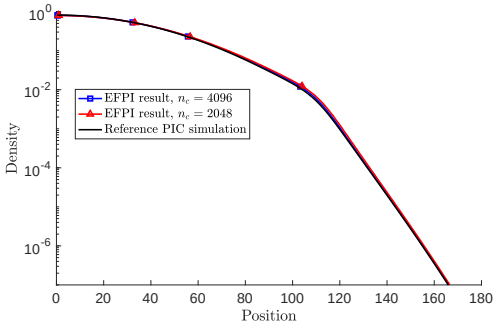
(a) Error in the density, ε.

(b) Error in the discretized distribution function.

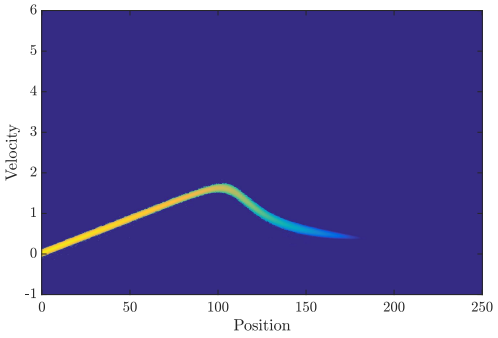
Figure 10: Extrapolation error as a function of the number of fine-scale timesteps per projective step. Parameters are otherwise chosen as for run 7 in Table 3.



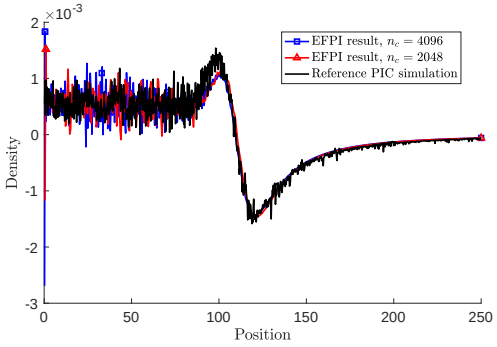
(a) Fully resolved PIC solution (Run 9), logarithmic color scale



(b) Ion density (logarithmic scale) versus x

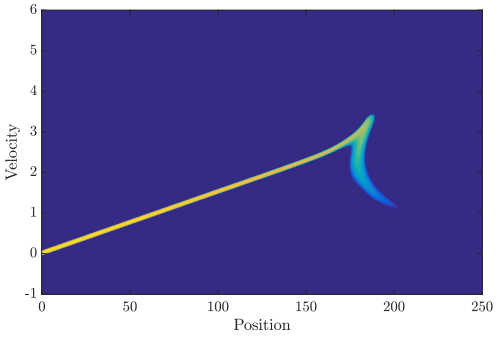


(c) EFPI solution (Run 10, $n_c = 4096$), logarithmic color scale

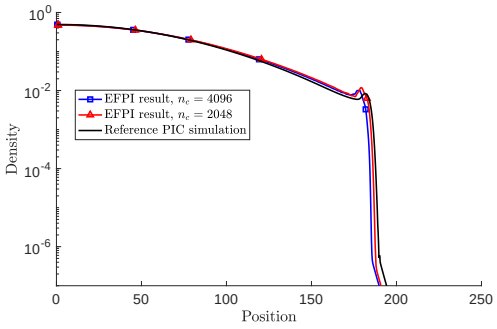


(d) Local charge density versus x

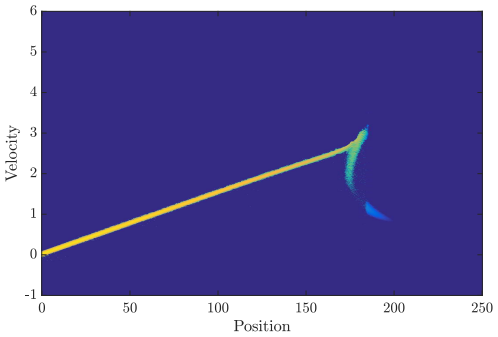
Figure 11: Plasma expansion at $t = 20$: PIC (Run 9) and EFPI (Runs 10 and 11, $n_c = 4096$ and 2048).



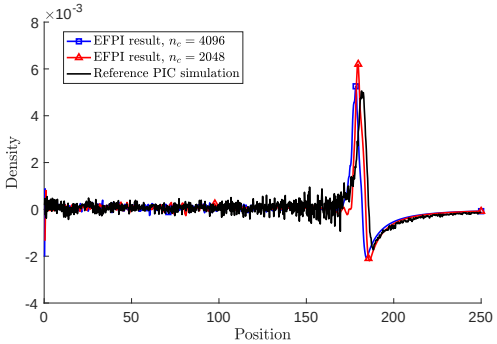
(a) Fully resolved PIC solution (Run 9), logarithmic color scale



(b) Ion density (logarithmic scale)



(c) EFPI solution (Run 10, $n_c = 4096$), logarithmic color scale



(d) Local charge density

Figure 12: Plasma expansion at $t = 50$: PIC (Run 9) and EFPI (Runs 10 and 11, $n_c = 4096$ and 2048).

4.3. Discussion: computational speedup

Finally, let us discuss the computational gains which have been realized in both test cases. The proposed equation-free method has been shown to remain accurate compared to a full-scale PIC simulation, even as highly kinetic phenomena becomes important. To be considered an interesting candidate for multi-scale simulations, such a method must also provide an appreciable run time speedup compared over the classical PIC method.

As noted in [26], the majority of the computational time used to run the equation-free methods is used during the time-stepping of fine-scale PIC kinetic simulations. The remaining parts of the EFPI cycle (lifting, restricting, projecting) represents a small overhead, especially as the system grows bigger and needs a large numbers of particles. Then, the speedup mainly depends on the ability to keep the number of kinetic fine-scale time steps N_e small while using large projective time steps Δt.

Based on the numerical experiments presented here, we observe that, for both the original EFREE [26] and the wavelet-based method presented here, the macro time step is governed by the stability condition (3.10) i.e. a macro-scale physical effect. On the other hand, the number of necessary fine-scale time steps per cycle N_e has been shown to depend on fine-scale effects such as the shot noise and the electron plasma period. It is thus independent of macro-scale parameters, in particular the wavelength of the ion acoustic wave. This allows for substantial CPU time speedups, summarized in Table 4 for the simulation runs which we have presented.

Type	Ion acoustic wave, $L = 4$				Ion acoustic wave, $L = 8$				Plasma expansion, $L = 2000$		
Run	1	2	3	4	5	6	7	8	9	10	11
Speedup	1	14.5	49	140	1	13	45	200	1	25	80

Table 4: Observed speedup between identical EFPI and full PIC runs 1–11 in Tables 1 and 3. All simulations were performed on a single core of a 2.7Ghz Intel Core i7.

As can be seen from runs 1–8, an approximate linear scaling is achieved with the size of the coarse grid step, even as the ratio $\Delta t/\delta t$ increases slower than linearly. The quiet start procedure implemented in the proposed method is probably an important factor for this result as the statistical particle noise scales as $\log(N_{pc})/N_{pc}$ and not as $1/\sqrt{N_{pc}}$ for the very short PIC runs that are used during each EFPI cycle. Here N_{pc} is the number of particles per coarse grid cell. Reasoning as in [26], one estimates the signal-to-noise ratio of the least squares fit used to determine the time derivative as $\xi = \sigma_f/M_t$, where σ_f the fitting error due to shot noise and M_t the time derivative. If we estimate the fitting error as $\sigma_f \sim \log(N_{pc})/(N_{pc}\sqrt{N_e}) \sim \log(L)/\left(L\sqrt{N_e}\right)$ (from the least squares derivation) and assume $M_t \sim 1/L$, this gives $\xi \sim \log(L)\left(N_e\right)^{-1/2}$. As a consequence, a constant signal-to-noise ratio can be obtained with a very weak dependence of N_e on L or N_{pc}. This is consistent with the numerical observations made above.

Finally, the noise reduction achieved in our proposed method allows the use of much lower number of particles than in corresponding full PIC simulations with no perceptible noise increase or accuracy loss. This was illustrated by the plasma expansion simulations where the number of particles is reduced by 8 using the projective integration approach. This is possibly also due to the better scaling achieved by the use of a quiet start for very short runs. In this case, it contributes to the high speedup demonstrated in run 11, with a gain of almost 2 orders of magnitude in runtime. This finding illustrates a promising aspect of the application of EFPI algorithms based on PIC fine-scale simulators.

5. Conclusions

In this paper we have presented a novel equation-free projective integration method for the Vlasov-Poisson equation. At the coarse level, the ion distribution function is represented on a non-parametric wavelet basis, and the electrons assumed to behave passively as an adiabatic fluid. At the fine-scale level, a particle-in-cell description is implemented for both ions and electrons. A lifting operator and a restriction operator are used to go back and forth consistently between these descriptions. During each macro time-step, an explicit PIC code is stepped forward for a small number of fine-scale time steps to determine the

rate of change in the macroscopic variables, which are then projected forward. The effects of statistical shot noise are controlled using both wavelet-based techniques, such as linear thresholding of coefficients, and more usual techniques in the EFPI framework such as coarsening in space and integration with the flow.

To validate the method and investigate its potential for simulating plasma systems, we applied it to two test cases. The first, a well known problem of the nonlinear propagation and steepening of an ion acoustic wave, was proposed in [26] to test the EFREE method as a first attempt at implementing the projective integration framework to model kinetic plasma phenomena. We also proposed a second test case, new for projective integration methods: the expansion of an ion slab in a vacuum [25]. These tests have confirmed that the method is stable and allows the use of time and space steps which are much larger than for the standard PIC method, while lifting numerous restrictions of the original EFREE method, most notably the assumption of a Maxwellian ion distribution function. We observed computational speedups between one to two orders of magnitude in our test runs. While these numbers depend highly on implementation and code optimization, such results are certainly encouraging.

To further increase the accuracy and speedup of the method, the next step is to implement a non-linear thresholding algorithm to adaptively select the relevant wavelet coefficients for the coarse description of the ion distribution function. This will allow the exploitation of the sparsity of the discretization of smooth ion distribution functions in a wavelet basis and reduce the size of the coarse discretization. This will be essential in higher dimensions. This approach should also be combined with a framework for adapting the coarse resolution in time and space. To realize this, new a posteriori error estimators should be developed to measure deviations between the coarse model and the underlying multiscale plasma system. The numerous algorithms which have been developed for applying wavelets to signal analysis and adaptive denoising offer many interesting possibilities for this new application. These various possible research directions show that the method still has a lot of potential for improvement.

Ultimately, the goal will be to apply the projective integration paradigm to more challenging plasma problems for which direct simulations by ordinary explicit, implicit or hybrid methods is out of reach due to the need to resolve meso-physics with a complex influence on the macroscopic behavior, such as transport driven by meso-scale turbulence.

Acknowledgment

This work was partially supported by the Department of Energy, Grant No. DE-SC0008553.

References

[1] C. Gear, I. Kevrekidis, and C. Theodoropoulos, Computational Chemical Engineering **26**, 941 (2002).
[2] I. G. Kevrekidis, C. W. Gear, J. M. Hyman, P. G. Kevrekidis, O. Runborg, C. Theodoropoulos, et al., Communications in Mathematical Sciences **1**, 715 (2003).
[3] C. Gear and I. Kevrekidis, SIAM Journal of Scientific Computing **24**, 1091 (2002).
[4] M. E. Kavousanakis, R. Erban, A. G. Boudouvis, C. W. Gear, and I. G. Kevrekidis, Journal of Computational Physics **225**, 382 (2007).
[5] G. Papanicolau, A. Bensoussan, and J. Lions, *Asymptotic analysis for periodic structures* (Elsevier, 1978).
[6] C. K. Birdsall and A. B. Langdon, *Plasma physics via computer simulation* (Taylor & Francis, 2004).
[7] J. Denavit, Journal of Computational Physics **42**, 337 (1981).
[8] B. I. Cohen, A. B. Langdon, and A. Friedman, Journal of Computational Physics **46**, 15 (1982).
[9] A. B. Langdon, B. I. Cohen, and A. Friedman, Journal of Computational Physics **51**, 107 (1983).
[10] P. Ricci, G. Lapenta, and J. Brackbill, Geophysical research letters **29** (2002).
[11] G. Lapenta, J. Brackbill, and P. Ricci, Physics of Plasmas (1994-present) **13**, 055904 (2006).
[12] S. Markidis, G. Lapenta, et al., Mathematics and Computers in Simulation **80**, 1509 (2010).
[13] R. J. Mason, Journal of Computational Physics **41**, 233 (1981).
[14] R. J. Mason, Journal of Computational Physics **51**, 484 (1983).
[15] A. S. Lipatov, *The hybrid multiscale simulation technology: an introduction with application to astrophysical and laboratory plasmas* (Springer, 2002).
[16] A. Friedman, S. Parker, S. Ray, and C. Birdsall, Journal of Computational Physics **96**, 54 (1991).
[17] S. Parker, A. Friedman, S. Ray, and C. Birdsall, Journal of Computational Physics **107**, 388 (1993).
[18] G. Chen, L. Chacón, and D. C. Barnes, Journal of Computational Physics **230**, 7018 (2011).
[19] S. Markidis and G. Lapenta, Journal of Computational Physics **230**, 7037 (2011).

[20] G. Chen and L. Chacon, Computer Physics Communications **185**, 2391 (2014).
[21] G. Chen and L. Chacon, Computer Physics Communications **197**, 73 (2015).
[22] L. Chacón and G. Chen, Journal of Computational Physics **316**, 578 (2016).
[23] G. Chen, L. Chacón, C. Leibs, D. Knoll, and W. Taitano, Journal of Computational Physics **258**, 555 (2014).
[24] P. Degond, F. Deluzet, and L. Navoret, Comptes Rendus Mathématique **343**, 613 (2006).
[25] P. Degond, F. Deluzet, L. Navoret, A.-B. Sun, and M.-H. Vignal, Journal of Computational Physics **229**, 5630 (2010).
[26] M. A. Shay, J. F. Drake, and B. Dorland, Journal of Computational Physics **226**, 571 (2007).
[27] A. M. Maluckov, S. Ishiguro, and M. M. Škoric, Communications in Computational Physics **4**, 556 (2008).
[28] S. Mallat, *A wavelet tour of signal processing* (Academic press, 1999).
[29] D. L. Donoho, I. M. Johnstone, G. Kerkyacharian, and D. Picard, The Annals of Statistics pp. 508–539 (1996).
[30] R. Nguyen Van Yen, D. del Castillo-Negrete, K. Schneider, M. Farge, and G. Chen, Journal of Computational Physics **229**, 2821 (2010).
[31] F. Luisier, C. Vonesch, T. Blu, and M. Unser, Signal Processing **90**, 415 (2010).
[32] P. Mora, Physics of Plasmas **12**, 112102 (2005).
[33] K. Kormann, SIAM Journal on Scientific Computing **37**, B613 (2015).
[34] L. Einkemmer and C. Lubich (2018), arXiv preprint, arXiv:1801.01103.
[35] G.-H. Cottet and P.-A. Raviart, SIAM journal on numerical analysis **21**, 52 (1984).
[36] K. Ganguly and H. Victory, Jr, SIAM journal on numerical analysis **26**, 249 (1989).
[37] S. Wollman and E. Ozizmir, SIAM journal on numerical analysis **33**, 1377 (1996).
[38] W. Nevins, G. Hammett, A. Dimits, W. Dorland, and D. Shumaker, Physics of Plasmas **12**, 122305 (2005).
[39] N. A. Krall and A. W. Trivelpiece, *Principles of Plasma Physics* (San Francisco Press, Inc., San Francisco, CA, 1986).
[40] I. Daubechies, SIAM Journal on Mathematical Analysis **24**, 499 (1993).
[41] W. T. Taitano, D. A. Knoll, L. Chacón, and G. Chen, SIAM Journal on Scientific Computing **35**, S126 (2013).
[42] H. Schamel, Journal of Plasma Physics **9**, 377 (1973).
[43] D. Dorozhkina and V. Semenov, Physical Review Letters **81**, 2691 (1998).
[44] A. Baitin and K. Kuzanyan, Journal of Plasma physics **59**, 83 (1998).
[45] V. Kovalev, V. Bychenkov, and V. Tikhonchuk, Journal of Experimental and Theoretical Physics **95**, 226 (2002).
[46] V. Kovalev and V. Y. Bychenkov, Physical Review Letters **90**, 185004 (2003).
[47] T. Grismayer, Ph.D. thesis, Ecole Polytechnique, Palaiseau, France (2006).
[48] T. Grismayer, P. Mora, J. Adam, and A. Héron, Physical Review E **77**, 066407 (2008).
[49] P. Mora, Physical Review E **72**, 056401 (2005).
[50] J. Hammersley, Annals of the New York Academy of Sciences **86**, 844 (1960).

Appendix

A. A primer on wavelets

For completeness, let us recall some notions on wavelets, see e.g. [28] for further details.

Continuous decomposition. For any function $f \in L^2(\mathbb{R})$ and a starting resolution level J_f, representation in the wavelet basis is given by the reconstruction formula:

$$f(y) = \sum_{k=-\infty}^{\infty} \overline{f}_{J_\mathrm{f},k}\phi_{J_\mathrm{f},k}(y) + \sum_{j=J_\mathrm{f}}^{\infty}\sum_{k=-\infty}^{\infty} \widetilde{f}_{j,k}\psi_{j,k}(y) \tag{A.1}$$

where

$$\begin{aligned} \phi_{j,k}(y) &= 2^{j/2}\phi\left(2^j y - k\right), \\ \psi_{j,k}(y) &= 2^{j/2}\psi\left(2^j y - k\right), \end{aligned} \tag{A.2}$$

are scaled and translated versions of the father wavelet (or scaling function) $\phi(y)$ and the mother wavelet $\psi(y)$. We will assume that ψ has zero average and compact support, and that the wavelet family $(\psi_{j,k}(y))_{j\in\mathbb{N},k\in\mathbb{Z}}$ is an orthonormal family. One can show existence of the scaling function ϕ, defined such that the orthogonal complement in $L^2(\mathbb{R})$ of the linear space spanned by the wavelets is itself orthogonally spanned by the translates of the function ϕ.

The coefficients $\widetilde{f}_{j,k} = \langle f|\psi_{j,k}\rangle = \int_\mathbb{R} f\psi_{j,k}$ of a function f in this wavelet family, called detail coefficients, describe fluctuations of the function f at scale 2^{-j} around the position $k/2^j$. Large values of j correspond to fine scales, small values to coarse scales. The coefficients $\overline{f}_{j,k} = \langle f|\phi_{j,k}\rangle$ are called the scaling coefficients. The first sum on the right-hand side of formula (A.1) forms a smooth approximation of f at the coarse

scale 2^{-J_f}, and the second sum represents a correction formed by the addition of details at successively finer scales.

In this work, we employ the R-Coiflet wavelet family of order 4, chosen for its regularity and symmetry properties [40]:

$$\int_{\mathbb{R}} y^m \psi(y) = 0, \qquad 0 \le m < 4, \qquad \int_{\mathbb{R}} y^m \phi(y) = 0, \qquad 1 \le m < 4.$$

This implies that the first four moments of the distribution function depend only on its scaling coefficients, and not on the detail coefficients.

Periodization. Let us note that the reconstruction formula (A.1) involves an infinite sum over the position index k. In practice we are interested only in functions defined over a bounded domain, and the commonly used alternative is to periodize the wavelet transform over this domain [28]. Assuming that the coordinates are rescaled so that all particles lie in $[0, 1]$, wavelets and scaling functions are replaced by their periodic counterparts:

$$\begin{aligned} \phi_{j,k}(y) &\longrightarrow \sum_{i\in\mathbb{Z}} \phi_{j,k}(y+l), \\ \psi_{j,k}(y) &\longrightarrow \sum_{i\in\mathbb{Z}} \psi_{j,k}(y+l). \end{aligned} \tag{A.3}$$

Throughout this paper we use such periodic wavelets.

Discrete wavelet decomposition. In practice, a distribution function f is typically represented numerically by its histogram f^H on a regular grid with 2^{J_f} points $x_k = 2^{-J_\mathrm{f}} k$, $0 \le k \le 2^{J_\mathrm{f}} - 1$. Wavelet-based density estimation algorithms [29, 30] proceed by approximating the scaling coefficients at the finest scale J_f by:

$$\overline{f}_{J_\mathrm{f},k} \approx 2^{-J_\mathrm{f}/2} f^H(2^{-J_\mathrm{f}} k). \tag{A.4}$$

The fast Discrete Wavelet Transform [28] can then be used to deduce the wavelet coefficients at coarser scales ($j \le J_\mathrm{f}$). Filtering algorithms can then choose coefficients to be discarded according to some linear or nonlinear criteria [29].

B. Particle sampling and co-evolving frame integration

We describe here the co-evolving frame projection which is used in the restriction step of our proposed method, see Section 3.3. Let Δt the macroscopic time step, $t_n = n\Delta t$. Suppose that at time $s \in [t_n, t_{n+1})$, we know the ion distribution function $f^\mathrm{i}(x, v, s)$. The exact solution f^i solution of the system (2.1–2.2) is constant along the characteristics: for a given point (X_0, V_0) in phase space, let $(X(t), V(t))$ be given by Newton's law,

$$\begin{aligned} \frac{\mathrm{d}X}{\mathrm{d}t} = V, \qquad \frac{\mathrm{d}V}{\mathrm{d}t} = -\frac{e}{m_\mathrm{i}} \nabla_x \phi(X, t), \\ X(s) = X_0, \quad V(s) = V_0. \end{aligned} \tag{B.1}$$

We have that $f^\mathrm{i}(X(t), V(t), t) = f^\mathrm{i}(X_0, V_0, s)$. Now the self-consistent potential ϕ is typically slow-moving and quite smooth. Assuming that this is the case and $\phi(x, t_n) \equiv \phi_n(x)$ is a good approximation for $\phi(x, s)$ for $s \in [t_n, t_{n+1}]$, a simple and explicit approximation to the characteristics is obtained by

$$\widetilde{X}_{(n,n+1)}(s,t) = X_0 + (t-s)\, V_0, \qquad \widetilde{V}_{(n,n+1)}(s,t) = V_0 - \frac{e}{m_\mathrm{i}} (t-s)\, \nabla_x \widetilde{\phi}_n(X_0). \tag{B.2}$$

Let us introduce finally the approximately integrated solution $\widetilde{f}^\mathrm{i}_{(n,n+1)}(x, v, s)$ defined implicitely by

$$\widetilde{f}^\mathrm{i}_{(n,n+1)} \left(\widetilde{X}_{(n,n+1)}(s, t_{n+1}), \widetilde{V}_{(n,n+1)}(s, t_{n+1}), s \right) = f^\mathrm{i}(X_0, V_0, s). \tag{B.3}$$

Note that if $t \mapsto (\widetilde{X}_{(n,n+1)}(s,t), \widetilde{V}_{(n,n+1)}(s,t))$ were the true characteristics then $\widetilde{f}^{\mathrm{i}}_{(n,n+1)}(s)$ would be constant to $f^{\mathrm{i}}(t_{n+1})$. In any case, it holds $\widetilde{f}^{\mathrm{i}}_{(n,n+1)}(s = t_{n+1}) = f^{\mathrm{i}}(t_{n+1})$. Thus we will reduce the projection error by computing and projecting forward a coarse discretization $\widetilde{f}^{\mathrm{i,c}}_{(n,n+1)}$ of $\widetilde{f}^{\mathrm{i}}_{(n,n+1)}$, instead of projecting directly the coefficients $f^{\mathrm{i,c}}$. To achieve this, we propose the following restriction scheme.

1. At initialization of the fine-scale PIC time stepper, e.g. at time $t_n = n\Delta t$, we store the electric field $E_n = -\nabla_x \phi_n$.

2. Let δt be the fine-scale time step. For each fine scale step, $p = 0, \ldots, N_e$, the coarse description of the ion distribution function is computed as follows. Iterating through all particles representing ions, the phase space position of each is integrated on the approximate characteristics (B.2), with $s = t_n + p\delta t$:

$$\begin{cases} \widetilde{X^{\mathrm{i}}_j} = X^{\mathrm{i}}_j + (\Delta t - p\delta t) \cdot V^{\mathrm{i}}_j, \\ \widetilde{V^{\mathrm{i}}_j} = V^{\mathrm{i}}_j + \dfrac{e}{m_{\mathrm{i}}} (\Delta t - p\delta t) \cdot E_n \left(X^{\mathrm{i}}_j\right). \end{cases} \tag{B.4}$$

   The particle is deposited on the grid by updating the values of the histogram on the fine grid:

$$\begin{cases} k_x = \left\lfloor \widetilde{X^{\mathrm{i}}_j}/h \right\rfloor, \qquad \theta_x = \widetilde{X^{\mathrm{i}}_j}/h - k_x, \qquad k_v = \left\lfloor (\widetilde{V^{\mathrm{i}}_j} - V_{min})/\delta v \right\rfloor, \\ [\widetilde{f}^{\mathrm{i}}_{(n,n+1)}]_{k_x,k_v} \leftarrow [\widetilde{f}^{\mathrm{i}}_{(n,n+1)}]_{k_x,k_v} + (1-\theta_x) \cdot \omega^{\mathrm{i}}_j, \\ [\widetilde{f}^{\mathrm{i}}_{(n,n+1)}]_{k_x+1,k_v} \leftarrow [\widetilde{f}^{\mathrm{i}}_{(n,n+1)}]_{k_x+1,k_v} + \theta_x \cdot \omega^{\mathrm{i}}_j. \end{cases} \tag{B.5}$$

3. Finally, the linear smoothing scheme (3.9) and the discrete wavelet transform are used to coarsen the phase space resolution of the data as described above.

C. Quiet start particle loading and momentum-conserving histogram reconstruction

At the particle loading stage of the projection method presented in Section 3.3, the ion distribution function f^{i} is known by its scaling coefficients on the fine grid. Using the approximation (A.4), we recover the approximated histogram of the ion distribution function:

$$f^{\mathrm{i,f}} \equiv \left\{ f^{\mathrm{i,f}}_{k_x,k_v} \right\}_{\substack{1 \le k_x \le n_{x,\mathrm{f}} \\ 1 \le k_v \le n_{v,\mathrm{f}}}}.$$

Due to the extrapolation process or the properties of the wavelets, some values of the approximate histogram may be negative. This results in a non-monotone CCDF, which is nonphysical and must be corrected. We propose here a simple reconstruction method which conserves momentum (except in pathological cases) and is found to give good results in practice.

Remark C.1. *Note that the reconstruction algorithm described hereafter can also be applied after lifting the histogram to the fine grid in velocity, but before lifting it to the fine grid in space. Since it is applied to the velocity histogram for each grid point in space, this can save some computational time if this process becomes a major factor in the total computational time (this was not the case in our numerical experiments).*

Let us compute for a given index of the spatial grid $1 \le k_x \le n_{x,\mathrm{f}}$ the corresponding value of the ion density,

$$n^{\mathrm{i,f}}_{k_x} = \sum_{k_v=1}^{n_{v,\mathrm{f}}} f^{\mathrm{i,f}}_{k_x,k_v}, \tag{C.1}$$

and the values of the normalized conditional cumulative distribution function F_v at points $0 \le k_v \le n_{v,\mathrm{f}}$:

$$F_v[k_v \mid k_x] = \frac{1}{n^{\mathrm{i}}_{k_x}} \sum_{l=1}^{k_v} f^{\mathrm{i,f}}_{k_x,k_v}. \tag{C.2}$$

For simplicity, we drop the dependence of $n^{\mathrm{i,f}}$ and F_v on k_x in the rest of the paragraph. Note that $F_v[0] = 0$ and $F_v[n_{v,\mathrm{f}}] = 1$. In a first step, we modify the entries of F_v to ensure that $0 \le F_v \le 1$ for all indexes; in a second step, we ensure that the resulting F_v is monotone. Let us describe these two stages of the algorithm in details.

Edge reconstruction. It is possible that some entries of the array F_v fall outside of the range $[0, 1]$ because of overshoot. Let us describe here a procedure to correct the case where some values are negative, which can be easily tranposed to the case of values larger than 1.

Step 1: Find k_0, the largest index such that $F_v[k_0] < 0$. Set $k^- = k_0 - 1$, $k^+ = k_0 + 1$.

Step 2: Compute the momentum-conserving slope (see Lemma C.2),

$$\mathrm{d}F = \frac{2\sum_{k=1}^{k^+-1} F_v[k]}{(k^+ - k^-)(k^+ - k^- - 1)}. \tag{C.3}$$

- If $0 \le (k^+ - k^- - 1)\mathrm{d}F \le F_v[k^+]$, then go to the next step;
- Otherwise, extend the range of indexes between k^- and k^+ and recompute $\mathrm{d}F$ accordingly: If $k^- > 0$, then $k^- \leftarrow k^- - 1$. Otherwise let $k^+ \leftarrow k^+ + 1$ and $k^- \leftarrow k_0 - 1$.

Step 3: Finally, modify the array F_v by

$$F_v[k] \leftarrow 0 \text{ for } 0 \le k < k^-, \qquad F_v[k] \leftarrow (k - k^-)\mathrm{d}F \text{ for } k^- \le k < k^+. \tag{C.4}$$

Interior reconstruction. Let us suppose now that $0 \le F_v \le 1$. Suppose that $k_v \mapsto F_v[k_v]$ is not monotone.

Step 1: Find k_0, the smallest index such that $F_v[k_0] > F_v[k_0 + 1]$. Let k_1 be the smallest index such that $F_v[k_0] \le F_v[k_1]$ and

$$k^+ = \operatorname*{argmin}_{k_0+1 \le k \le k_1} F[k] \qquad \text{and} \qquad k_- = \max\left\{0 \le k \le k_0 - 1 \mid F_v[k] \le F[k^+]\right\}. \tag{C.5}$$

Step 2: Compute the momentum-conserving slope:

$$\mathrm{d}F = \frac{2}{k^+ - k^- - 1}\left(\frac{\sum_{k=k^-}^{k^+-1} F_v[k]}{k^+ - k^-} - F_v[k_-]\right). \tag{C.6}$$

- If $0 \le (k^+ - k^- - 1)\mathrm{d}F \le F_v[k^+]$, then go to the next step;
- Otherwise, set $k^+ \leftarrow k^+ + 1$ and recompute $\mathrm{d}F$ accordingly.

Step 3: Finally, modify the array F_v by

$$F_v[k] \leftarrow F_v[k_-] + (k - k^-)\mathrm{d}F \qquad \text{for } k^- < k < k^+. \tag{C.7}$$

Now F_v is monotone up to $k^+ > k_0$. Proceeding recursively, we obtain in all cases a corrected monotone array F_v^+ where the $+$ denotes that the correction is computed from the lower to the upper indexes. To ensure symmetry of the reconstruction, it is necessary to compute another corrected array F_v^-, computed in similar fashion but starting from the higher indexes. Finally, we define the reconstructed array $\mathcal{R}F_v$ as:

$$\mathcal{R}F_v = (F_v^+ + F_v^-)/2. \tag{C.8}$$

Lemma C.2. *The slope defined by Eq.* (C.6) *(resp. Eq.* (C.3)*) is designed so that the reconstruction in Eq.* (C.7) *(resp. Eq.* (C.4)*) conserves the first two moments of the histogram $f_k = F_v[k] - F_v[k-1]$,*

$$M_0 = \sum_{k=1}^{n_{v,\mathrm{f}}} (F_v[k] - F_v[k-1]) \qquad \textit{and} \qquad M_1 = \sum_{k=1}^{n_{v,\mathrm{f}}} k(F_v[k] - F_v[k-1]).$$

Proof. The moment of order zero is always conserved, since $M_0 = F_v[n_{v,\mathrm{f}}] - F_v[0] = 1$. For the moment of order one, we investigate only Eq. (C.7) since the proof is the same. The change in M_1 can be computed by

$$\begin{aligned} M_1^{before} - M_1^{after} &= \sum_{k^- < k < k^+} (k - k^+) \left(F_v[k] - F_v[k-1] - \mathrm{d}F\right) \\ &= \sum_{k^- < k < k^+} (k - k^+) F_v[k] - \sum_{k^- \le k < k^+} (k + 1 - k^+) F_v[k] - \left(\sum_{k'=1}^{k^+ - k^- - 1} -k' \right) \mathrm{d}F \\ &= \frac{(k^+ - k^-)(k^+ - k^- - 1)}{2} \mathrm{d}F - (k^- - k^+) F_v[k^-] - \sum_{k=k^-}^{k^+ - 1} F_v[k] = 0, \end{aligned}$$

where we have used the fact that $(k + 1 - k^+)F_v[k] = 0$ for $k = k^+ - 1$ and also Eq. C.6. □

Particle loading. Once all the reconstructed arrays $(\mathcal{R}F_v[\,\cdot\,|k_x])_{1 \le k_x \le n_{x,\mathrm{f}}}$ have been computed, the quiet loading of the ion particles proceeds as follows. Recall that each particle is parameterized by its position X_j^{i}, velocity V_j^{i} and weight ω_j^{i}. Let $(a_j, b_j)_{j \in \mathbb{N}} \subset [0,1]^2$ be a low discrepancy sequence (e.g. the Hammersley sequence [50]), which yields numbers that are more evenly distributed than a randomly generated sequence. Then, we set the initial particle parameters as

$$\begin{aligned} X_i^{\mathrm{i}} &= L \cdot a_i, \\ V_i^{\mathrm{i}} &= (\mathcal{R}F_v)^{-1}(b_i \mid x = X_i^{\mathrm{i}}), \\ \omega_i^{\mathrm{i}} &= n^{\mathrm{i}}(X_i^{\mathrm{i}}), \end{aligned} \tag{C.9}$$

where all values are linearly interpolated between the two closest points of the fine grid. The inverse mapping $(\mathcal{R}F_v)^{-1}(\,\cdot\,|x = X_i^{\mathrm{i}})$ is evaluated by lookup and linear interpolation. To finish initializing the PIC code, the electrons are loaded using the normal quiet start procedure for a shifted Maxwellian function.