

PARTICLE-IN-WAVELETS SCHEME FOR THE 1D VLASOV-POISSON EQUATIONS ^{*}, ^{**}

ROMAIN NGUYEN VAN YEN¹, ÉRIC SONNENDRÜCKER²¹, KAI SCHNEIDER³ AND MARIE FARGE

Abstract. A new numerical scheme called particle-in-wavelets is proposed for the Vlasov-Poisson equations, and tested in the simplest case of one spatial dimension. The plasma distribution function is discretized using tracer particles, and the charge distribution is reconstructed using wavelet-based density estimation. The latter consists in projecting the Delta distributions corresponding to the particles onto a finite dimensional linear space spanned by a family of wavelets, which is chosen adaptively. The stability and accuracy of the scheme is supported by numerical computations of Landau damping and of the two-stream instability. By direct comparison with a reference solution obtained by a very precise semi-Lagrangian method, we show that the precision is improved roughly by a factor 3 compared to a classical PIC scheme, for a given number of particles.

Résumé. L'objectif de ce travail est de proposer un nouveau schéma numérique, appelé particle-in-wavelets, pour résoudre les équations de Vlasov-Poisson, et de le tester dans le cas le plus simple où il n'y a qu'une seule dimension d'espace. La fonction de distribution du plasma est discrétisée à l'aide d'un ensemble de particules, et la charge peut alors être reconstruite par la technique d'estimation de la densité en ondelettes. Cette dernière consiste à projeter la distribution de Dirac de chaque particule sur un espace de dimension finie engendré par une famille d'ondelettes choisie de façon adaptative. La stabilité et la précision du schéma sont confirmées par des calculs numériques portant sur l'amortissement Landau et sur l'instabilité double faisceau. En comparant directement les résultats avec une solution de référence obtenue par une méthode semi-lagrangienne très précise, nous mettons en évidence une diminution de l'erreur d'un facteur de l'ordre de 3 par rapport à la méthode PIC classique, pour un nombre de particules donné.

INTRODUCTION

The state of a hot plasma out of local thermodynamic equilibrium is characterized by a particle distribution function depending on position, velocity, and time. When the effects of collisions are neglected, this distribution

^{*} The authors would like to thank Diego del-Castillo-Negrete for his inspiration in the pursuit of PIC denoising, and Matthieu Haefele for developing and supporting the VLASY platform.

^{**} The authors acknowledge financial support from the Euratom-CEA association and the French Federation for Fusion Studies. This work, supported by the European Communities under the contract of Association between EURATOM, CEA and the French Research Federation for fusion studies, was carried out within the framework of the European Fusion Development Agreement. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

¹ LMD-CNRS-IPSL, École Normale Supérieure, 24 rue Lhomond, 75231 Paris Cedex 5, France

² IRMA-CNRS, Université de Strasbourg, 7 rue René-Descartes, 67084 Strasbourg Cedex, France

³ M2P2-CNRS and CMI, Université d'Aix-Marseille, 38 rue Frédéric Joliot-Curie, 13451 Marseille Cedex 13, France

satisfies the partial differential equation (PDE) known as the Vlasov equation. In the presence of simplifying hypotheses and/or special symmetries, the complexity of the problem can be reduced to permit a numerical resolution of the Vlasov equation using the Eulerian PDE discretization approach which requires a mesh in position and velocity. Recent progresses have pushed back the limit of the feasible to include for example the 5D gyrokinetic approximation [11, 15, 27]. However the full equation with six variables plus time remains a major computational challenge, which could take as much advantage of better numerical methods as of further machine development and parallelization.

In the past, even more stringent limitations in memory size and computing power have driven physicists to develop specific simulation methods, which approximate the plasma by a collection of macro-particles [3, 14] and can be linked to Monte-Carlo methods [1]. Although the precision of Monte-Carlo methods is limited by sampling error, they are known to be competitive for problems involving integrals over many dimensions. For the specific problem of solving the Vlasov equations, it is well known that particle methods offer a more straightforward implementation and an easier parallelization than their Eulerian counterparts. However, the debate over the best compromise in terms of overall development and simulation cost versus precision remains unsettled. Current outstanding applications of particle methods include magnetic confinement fusion [13, 16], and laser-plasma acceleration [5]. But in the last decade, there has been a tendency to challenge some results obtained using legacy particle methods on the grounds that the simulations were underresolved [20].

Plasma simulation using particles is dominated by derivatives of the particle-in-cell (PIC) method [3, 14], to whom a vast amount of research has afforded the status of a reference. PIC schemes are obtained by convolving the empirical density corresponding to the distribution of marker particles by a fixed kernel, also known as charge assignment function [14]. Their numerical properties are well understood empirically and they also benefit from an elegant physical interpretation [17]. The most widely used PIC method uses a piecewise linear kernel. Smoother kernels have been studied in the past [22] but computational limitations have made their use relatively rare, except for specific applications where precision and reduction of noise are of high concern.

In [21], we have studied another way of estimating the density from the particle positions, called wavelet-based density estimation (WBDE). WBDE was first introduced in the more general context of nonparametric density estimation [9], as an alternative to the classical kernel density estimation (KDE) method. Instead of being convolved with a kernel, the empirical density is projected onto a finite dimensional linear space spanned by a family of orthogonal wavelets. The projection space is determined from the data itself, which allows for a refined representation around sharp features, and could make the method more precise than PIC for a given computational cost.

The aim of this paper is to present a new scheme for the 1D Vlasov-Poisson equations, that we call particle-in-wavelets (PIW) because it relies crucially on wavelet expansions of the Dirac delta functions corresponding to every particle. WBDE is used for the density estimation step in the PIW scheme. The information pertaining to each particle thus gets spread among neighboring wavelets, in the same manner as it gets spread among neighboring grid points in the PIC case. The general philosophy of this work has been to build the scheme from scratch, taking as much advantage of wavelets as possible, so as to have a proof of principle of various properties specific to wavelets. We are focusing mostly on understanding the precision of the scheme and the conservation properties, as opposed to optimizing numerical efficiency.

Other methods have been proposed, and sometimes successfully implemented, to improve PIC simulations, but they all rely on the application of a linear operator to the particle distribution function in order to reduce the noise. Linear denoising operators are optimal when the regularity of the signal to be denoised is homogeneous in space. However, the distribution function within collisionless plasmas is known to develop sharp features. Therefore a nonlinear thresholding approach could be more efficient, and this work is a first step in this direction.

We first recall some background, on the Vlasov-Poisson equations in one space dimension on the one hand, and on wavelets on the other hand. Reference semi-Lagrangian and PIC schemes are also described. After that we proceed to define the various steps of the PIW (particle-in-wavelets) wavelet-based scheme. Finally, numerical results from the three schemes are presented and the properties of the new PIW scheme are assessed in detail.

1. BACKGROUND

1.1. Vlasov-Poisson equations

The Vlasov equation governs the evolution of the particle distribution function $f(x, v, t)$ of a given species in a collisionless plasma. In the 1D case which we focus on, v varies in \mathbb{R} , and for simplicity, we assume periodicity in the x direction, i.e. $x \in \mathbb{T} := \frac{\mathbb{R}}{l\mathbb{Z}}$, where l is the period. $\Gamma = \mathbb{T} \times \mathbb{R}$ is the phase space for one particle in the plasma. We restrict ourselves to the case where the ion density is assumed to remain constant, and the magnetic field is neglected. The system is then fully described by the Vlasov-Poisson equations, which can be written in non-dimensional form as follows:

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

$$\partial_{xx} \phi + 1 - l \int_{\mathbb{R}} f(x, v, t) dv = 0 \quad (2)$$

where $\phi(x, t)$ is the electric potential. In the following, we denote by $E(x, t) = -\partial_x \phi(x, t)$ the electric field, and by $\rho(x, t) = \int_{\mathbb{R}} f(x, v, t) dv$ the normalized electron density. Note that we have adopted here the following slightly uncommon normalizations for ρ and f :

$$\int_{\Gamma} f(x, v, t) dx dv = 1, \quad (3)$$

$$\int_{\mathbb{T}} \rho(x, t) dx = 1. \quad (4)$$

We shall also need the definitions of two physical quantities associated to the Vlasov-Poisson problem, namely the electrostatic energy

$$\mathcal{V}(t) = \frac{1}{2} \int_{\mathbb{T}} |E(x, t)|^2 \quad (5)$$

and the total energy

$$\mathcal{E}(t) = \frac{1}{2} \int_{\Gamma} v^2 f(x, v, t) dx dv + \mathcal{V}(t), \quad (6)$$

the latter being constant when f is a solution of the Vlasov-Poisson equations.

The goal of this paper is to describe a new numerical scheme for solving (1-2) together with an initial condition for f . We will use the fact that any solution of (1) is constant along the characteristic trajectories defined by:

$$\begin{cases} x'(t) = v(t) \\ v'(t) = -E(x(t), v(t), t) \end{cases} \quad (7)$$

Existing approaches to the same problem are divided into two main categories, that were already mentioned in the introduction:

- particle solvers, which integrate the characteristic equations (7) and discretize the potential on a grid in the x variable,
- Eulerian solvers where the full distribution function f is discretized on a phase space mesh or grid.

The first category is especially appealing when the phase space has dimension 4 or more, since it is then very costly to use an Eulerian approach. However, the discretization of f using particles is subject to statistical sampling noise which normally makes it of rather low precision [20]. In particular, for the 1D problem that we are considering here, we expect Eulerian solvers to be much more precise for a given computational cost. This allows us to take the solution obtained with a high precision Eulerian solver as a reference for measuring the convergence of particle methods. We have chosen a well validated semi-Lagrangian (SL) solver using cubic spline interpolation [25].

$h_0 = 0.03858077774789$	$h_2 = -0.07716155549577$	$h_4 = 0.74568755893443$
$h_1 = -0.12696912539621$	$h_3 = 0.60749164138568$	$h_5 = 0.22658426519707$

TABLE 1. Scaling function filter coefficients for the C^1 Coiflet family with 2 vanishing moments introduced in [6] and used throughout this paper.

It is a reasonable requirement that any new particle method should offer some improvement over the legacy PIC approach. As a contestant to compare with our newly proposed PIW method, we therefore take a PIC solver, which uses a triangular, piece-wise linear charge assignment function to estimate the charge density, and Fourier transforms to solve the Poisson equation.

When assessing the convergence of particle solvers, the loading scheme is especially important since it controls the initial amplitude of the noise. Provided that the initial data f_0 is factorized

$$f_0(x, v) = f_{0,x}(x)f_{0,v}(v)$$

we ensure that the initial noise is small using the following classical approach. First, compute a 2D low discrepancy sequence $(a_i, b_i)_{i \in \mathbb{N}}$ using the Hammersley algorithm, [12], which yields numbers that are more evenly distributed than random numbers. Then set as initial particle positions

$$\begin{aligned} x_i(0) &= F_{0,x}^{-1}(a_i) \\ v_i(0) &= F_{0,v}^{-1}(b_i) \end{aligned}$$

where $F_{0,x}^{-1}$ and $F_{0,v}^{-1}$ are the inverse mappings corresponding to the cumulative distribution functions $F_{0,x}$ and $F_{0,v}$ associated respectively to $f_{0,x}$ and $f_{0,v}$, and are in practice approximated using a pre-generated lookup table and polynomial interpolation. With this initialization scheme, called “quiet start”, the initial error on the particle distribution function is expected to scale like N_p^{-1} , as opposed to $N_p^{-1/2}$ for a random start.

1.2. Wavelets

In this section we recall some notions on wavelets. We keep the same notations as in [21]. More background may be found in the literature (see e.g. [18]). The construction is done in $\mathbb{R}/l\mathbb{Z}$ with $l = 1$, and the adjustments for $l \neq 1$ will be made further down the moment they are required. Let ψ be a 1-periodic wavelet generating an orthogonal basis of $L^2(\mathbb{R}/\mathbb{Z})$, and φ be the associated scaling function.

The dilated and translated scaling functions and wavelets are denoted

$$\varphi_\lambda(x) = 2^{\frac{j_\lambda}{2}} \varphi(2^{j_\lambda} x - i_\lambda) \quad (8)$$

$$\psi_\lambda(x) = 2^{\frac{j_\lambda}{2}} \psi(2^{j_\lambda} x - i_\lambda) \quad (9)$$

where $\lambda = (j_\lambda, i_\lambda)$ is a multi-index characterizing the scale j_λ and position i_λ of a wavelet. For any positive integer J , we denote by V_J the 2^J -dimensional linear space spanned by $(\varphi_{(J,i)})_{0 \leq i \leq 2^J - 1}$. The wavelet basis is entirely characterized by the following two-scale recurrence equation [18] satisfied by the scaling function φ :

$$\varphi(t) = \sqrt{2} \sum_{i=0}^{S-1} h_i \varphi(2t - i), \quad (10)$$

where S is an integer and $(h_i)_{0 \leq i \leq S-1}$ is called the scaling function filter.

i	0	1	2	3	4
$r_i^{(1)}$	0	-0.619368727403235	-1.403784906417689	2.427684222146337	-0.839028531550099
$r_i^{(2)}$	0.622642595976898	-0.389477243531951	0.179082676222473	-0.326853461357942	0.225926730678971

TABLE 2. Stencils for first and second derivative operators in the R-Coiflet 6 scaling functions basis. Only the values for positive i are given. The values for negative i are obtained from the conditions that $r^{(1)}$ is odd and $r^{(2)}$ is even.

The scaling function and wavelet coefficients of a function $u \in L^2(\mathbb{T})$ are defined by

$$\bar{u}_\lambda = \langle u \mid \varphi_\lambda \rangle \tag{11}$$

$$\tilde{u}_\lambda = \langle u \mid \psi_\lambda \rangle \tag{12}$$

where $\langle \cdot \mid \cdot \rangle$ denotes the usual scalar product in $L^2(\mathbb{T})$. For any positive integer L , u can be decomposed as follows:

$$u = \sum_{i=0}^{2^L-1} \bar{u}_{(L,i)} \varphi_{(L,i)} + \sum_{j=L}^{\infty} \sum_{i=0}^{2^j-1} \tilde{u}_{(j,i)} \psi_{(j,i)} \tag{13}$$

where the first sum is the projection of u on V_L , and the second sum contains all the additional details needed for the full reconstruction of u .

In the following we shall also need to represent first and second order differential operators in the space V_J . The results necessary to do so were derived in [2], and we summarize them here for completeness. For any $u \in V_J$ we can, using (13), express the derivative of u as follows:

$$u' = \sum_{i=0}^{2^J-1} \bar{u}_{(J,i)} \varphi'_{(J,i)},$$

but u' is in general not in V_J , so that a Galerkin projection is necessary to obtain a discrete representation of the derivative operator. The Galerkin representation of the first order derivation operator in V_J is defined by the matrix \mathcal{G} such that:

$$\begin{aligned} \mathcal{G}_{i_1, i_2} &= \langle \varphi_{(J, i_1)} \mid \varphi'_{(J, i_2)} \rangle = 2^J \int_{\mathbb{T}} dx \varphi(2^J x - i_1) \varphi'(2^J x - i_2) \\ &= \int_{\mathbb{T}} dy \varphi(y + i_2 - i_1) \varphi'(y) := r_{i_2 - i_1}^{(1)} \end{aligned}$$

In [2], it is shown that the matrix elements $r_{i_2 - i_1}^{(1)}$ can be determined by solving a linear system with coefficients depending only on the h_i , and that they satisfy the antisymmetry relation $r_{-i}^{(1)} = -r_i^{(1)}$, so that applying the matrix \mathcal{G} is numerically equivalent to applying a centered finite difference operator. \mathcal{G} is thus an antisymmetric matrix. The same procedure can be followed to obtain the Galerkin representation of the second derivative in V_J , defined by the matrix \mathcal{L} such that:

$$\mathcal{L}_{i_1, i_2} = \langle \varphi_{(J, i_1)} \mid \varphi''_{(J, i_2)} \rangle$$

and which corresponds to a symmetric filter $r_i^{(2)}$.

In the following we make use of only one orthogonal wavelet basis, namely the C^1 Coiflets with $S = 6$ introduced at the end of [6], which we dub "R-Coiflet 6". This wavelet is more regular than the normal Coiflet 6 wavelet, hence the R which stands for "regular". The scaling function filter coefficients (h_i) are recalled in

Table 1. Except otherwise noticed, the results presented below also hold for any other orthogonal wavelet basis. The filters $r^{(1)}$ and $r^{(2)}$ corresponding to the R-Coiflet 6 wavelet basis are provided in Table 2 for completeness. Note that 6 is the minimal filter length for which both $r^{(1)}$ and $r^{(2)}$ are well defined, which is why we have chosen this wavelet family. Another candidate having the same filter length was the Daubechies minimal phase family, but we have discarded it because of its lack of symmetry.

2. DESCRIPTION OF THE PIW SCHEME

2.1. Wavelet-based density estimation

Let us briefly recall the wavelet-base density estimation (WBDE) algorithm for estimating the charge density from the position of a finite number N_p of particles. A more detailed account, illustrated by several applications, may be found in [21]. WBDE, first introduced in [9] belongs to the class of density estimators constructed from orthogonal projections [24]. Its main originality is that the members of the orthogonal family on which the density is projected are chosen in an adaptive manner, depending on the observed particle positions.

In this work, we have focused on the Vlasov-Poisson problem, for which only the spatial distribution of particles matters when estimating the electric field, while the velocity distribution is irrelevant. Therefore, we apply the WBDE algorithm to the spatial distribution only, which is one dimensional. In the units we are working with, this spatial distribution is identical to the normalized electron density ρ . Defining the empirical density associated to the particle positions $(x_n)_{1 \leq n \leq N_p}$ by

$$\rho^\delta(x) = \frac{1}{N_p} \sum_{n=1}^{N_p} \delta(x - x_n) \quad (14)$$

where δ is the Dirac measure, the idea of the WBDE algorithm is to project ρ^δ onto the finite dimensional subspace spanned by all the wavelets at scales coarser than L and part of the wavelets at scales j such that $L \leq j \leq J$, where the scales L and J are defined as follows:

$$L = \left\lfloor \log_2 N_p^{\frac{1}{1+2r_0}} \right\rfloor \quad (15)$$

$$J = \left\lceil \log_2 \frac{N_p}{\log_2 N_p} \right\rceil \quad (16)$$

In these formulas, $\lfloor \cdot \rfloor$ and $\lceil \cdot \rceil$ denote respectively the lower and upper integer parts, and r_0 is the order of regularity of the wavelet, which is 1 in our case. The wavelets that are kept for $L \leq j \leq J$ are determined according to the amplitude of the corresponding coefficients in ρ^δ , in the spirit of the classical wavelet denoising algorithms, see [8, 9, 18]. The threshold at scale j is defined by:

$$\Theta_j = K \sqrt{\frac{j}{N_p}}$$

where K is a constant which in principle depends on the regularity of the solution [9]. In the following, we have fixed $K = 1$.

The scales L and J and the thresholds Θ_j were derived rigorously in [9]. As detailed in [21], the rationale for the different treatments of the fine and coarse scales is as follows. At coarse scales ($j < L$), the estimation error is dominated by the bias term, so that all the information that is available should be retained. At finer scales ($L \leq j \leq J$), the wavelets start to resolve statistical fluctuations in the particle distribution, so that only large coefficients which stand out of these fluctuations and indicate rapid variations in the underlying density should be retained. This is the principle of nonlinear denoising. The $N_p^{-\frac{1}{2}}$ dependence of the threshold can

be justified heuristically [21] from the asymptotic distribution of the wavelet coefficients when estimating a probability density from N_p independent realizations. At even finer scales ($j > J$), the wavelets start to resolve individual particles and the coefficients therefore do not carry any information related to the underlying density, so that they should be immediately discarded.

To treat the case $l \neq 1$, we can rescale ρ^δ to obtain a normalized distribution on $\frac{\mathbb{R}}{2}$:

$$x \rightarrow l^{-1}x \quad (17)$$

$$\rho^\delta \rightarrow l\rho^\delta \quad (18)$$

and notice that this change of variables leaves the scaling function and wavelet coefficients (11-12) unchanged. Therefore at this stage the algorithm can be applied harmlessly as if $l = 1$.

In summary, the following steps are required to actually implement the estimation:

- (1) compute some approximation $\bar{\rho}_{(J,i)}^S$ of the scaling function coefficients of ρ^δ at scale J , by which the projection of ρ^δ on the space V_J can be expressed:

$$\rho^S = \sum_{i=0}^{2^J-1} \bar{\rho}_{(J,i)}^S \varphi_{(J,i)} \quad (19)$$

- (2) apply the fast wavelet transform algorithm to compute the wavelet coefficients of ρ^S at scales j such that $L \leq j \leq J$:

$$\rho^S = \sum_{i=0}^{2^L-1} \bar{\rho}_{(L,i)}^S \varphi_{(L,i)} + \sum_{j=L}^{J-1} \sum_{i=0}^{2^j-1} \tilde{\rho}_{(j,i)}^S \psi_{(j,i)} \quad (20)$$

- (3) apply the nonlinear threshold operator :

$$\rho^W = \sum_{i=0}^{2^L-1} \bar{\rho}_{(L,i)}^S \varphi_{(L,i)} + \sum_{j=L}^{J-1} \sum_{i=0}^{2^j-1} \mathbb{1}_{[0,+\infty[} \left(|\tilde{\rho}_{(j,i)}^S| - \Theta_j \right) \tilde{\rho}_{(j,i)}^S \psi_{(j,i)} \quad (21)$$

where $\mathbb{1}_{[0,+\infty[}$ is the characteristic function of the interval $[0, +\infty[$.

The remaining difficulty in the above procedure is its first step, to which we dedicate the next section entirely. If steps (2-3) are omitted, we obtain a scheme which we call Linear PIW (L-PIW), and that we will use below for comparison purposes. In that case, the value of L has no influence, and we allow J to be chosen as a free parameter.

2.2. Approximation of empirical scaling function coefficients

To perform step (1) of the above WBDE algorithm, some approximate values $\bar{\rho}_\lambda^S$ of the coefficients $\bar{\rho}_\lambda^\delta$ must be defined. According to (11) and (14), their exact values are:

$$\bar{\rho}_\lambda^\delta = \frac{1}{N_p} \sum_{n=1}^{N_p} \varphi_\lambda(x_n), \quad (22)$$

which is difficult to compute because there is no analytical expression available for φ_λ . In [21] we proposed to use the approximation

$$\bar{\rho}_{(J,i)}^S = 2^{-\frac{J}{2}} \rho^H(2^{-J}i),$$

where ρ^H is a histogram constructed over a grid of size 2^J in each direction. To increase the accuracy in the computation, we adopt here the following alternative. We know that the scaling function φ is supported on

the interval $[0, S - 1]$, which can be discretized using $2^{J_{LT}}$ regularly spaced points

$$s_k = 2^{-J_{LT}}(S - 1)k,$$

at which we can compute approximations of the values of φ

$$\check{\varphi}_k \simeq \varphi(s_k)$$

using for example the iterative algorithm described in [7]. The $\check{\varphi}_k$ constitute a look-up table of approximations to the $\varphi(s_k)$, which needs to be computed only once for any chosen wavelet basis. Now assume that a particle is located at position x and that we want to approximate

$$\varphi_{(J,i)}(x) = 2^{\frac{j}{2}}\varphi(2^j x - i).$$

This can be nonzero only if $0 \leq 2^j x - i \leq S - 1$, that is, for S values of i . For each of these values of i we find the unique integer k such that

$$s_k \leq 2^j x - i < s_{k+1},$$

we let

$$\alpha = \frac{2^j x - i - x_k}{x_{k+1} - x_k}$$

and we compute a linear interpolation from the values stored in the lookup table,

$$\check{\varphi}_\lambda(x) = (1 - \alpha)\check{\varphi}_k + \alpha\check{\varphi}_{k+1}.$$

There remains to sum over all the particles to compute the scaling function coefficient of the normalized electron density which we use in the following:

$$\bar{\rho}_\lambda^S = \frac{1}{N} \sum_{n=1}^{N_p} \check{\varphi}_\lambda(x_n). \quad (23)$$

For each particle, we need to make S lookup operations in the table, so that the overall cost is proportional to SN_p . For the size of the table, we have used $J_{LT} = 18$. Note that when generalizing this algorithm to d dimensions, the required lookup table will remain one dimensional because the d -dimensional scaling functions are factorized into products of 1D scaling functions. Therefore there will be no increase in memory requirements for this stage of the algorithm.

2.3. Wavelet-Galerkin Poisson solver

Once the wavelet coefficients of the electron density ρ have been obtained by WBDE, the electric potential needs to be computed by solving the Poisson equation (2). A wavelet preconditioning technique has been previously developed to solve (2) in 3D using finite differences in the context of a PIC code [26]. Here, we prefer to use a full Galerkin discretization of (2) in the wavelet basis, as introduced in Section 1.2:

$$l^{-2} \mathcal{L} \bar{\phi}^S = 1 - \bar{\rho}^W \quad (24)$$

where $\bar{\phi}^S$ and $\bar{\rho}^W$ stand respectively for the column vectors $(\bar{\phi}_\lambda^S)$ and $(\bar{\rho}_\mu^W)$, ϕ^S stands for the unknown electric potential, and the l^{-2} prefactor comes from the rescaling of the domain (Eq. 17).

In order to better exhibit the properties of the scheme as it is likely to be implemented in the future for two or three spatial dimensions, or in the case of adaptive grids, we have decided to use an iterative method to solve (24), even though a direct method would be faster in the present one-dimensional case. As such, this system is

ill-conditioned and solving it directly using an iterative method would be inefficient. Fortunately, there is a well known technique to improve the condition number. The main idea is to solve for the wavelet coefficients instead of the scaling function coefficients, and to use a diagonal preconditioner in wavelet space. First (24) is rewritten

$$l^{-2}\mathcal{F}\mathcal{L}\mathcal{F}^\dagger\tilde{\phi}^S = \mathcal{F}(1 - \bar{\rho}^W) \quad (25)$$

where \mathcal{F} is the matrix form of the wavelet transform operation, and \cdot^\dagger stands for transposition. In practice, \mathcal{F} and its transpose are applied using the fast wavelet transform algorithm [18]. Now (25) can be efficiently solved using the conjugate gradient method with the diagonal preconditioner defined by:

$$\mathcal{D}_{\lambda,\mu} = \delta_{\lambda\mu}2^{j\lambda}$$

The wavelet coefficients of the potential at the previous time step are used as initial guess to speed-up the convergence. Once $\tilde{\phi}^S$ has been obtained, an inverse wavelet transform is applied to yield $\bar{\phi}^S$.

Note that an alternative and non equivalent approach would be to write down a Galerkin discretization of (2) in the basis made of the wavelets whose coefficients are non-zero in the expansion (21) for ρ^W . We would thus obtain an approximation ϕ^W to the potential which has the same support in wavelet space, allowing for an adaptive strategy. In the present approach, ϕ^S has a priori a full support in wavelet space. As we will see below, this has an important consequence on the self-force.

2.4. Interpolation and particle push

Like in the standard PIC method, it is essential that the interpolation method be compatible with the charge assignment scheme in order to avoid self-forces [3]. The scaling function coefficients of the electric field can be obtained by Galerkin projection of the gradient operator in the scaling function basis (see Sec. 1.2):

$$\bar{E}_\lambda^S = (\varphi_\lambda, \varphi'_\mu) \bar{\phi}_\mu^S = l\mathcal{G}\bar{\phi}^S$$

A natural first attempt is to directly interpolate E^S at the particle positions to obtain the forces exerted on the particles. For this purpose we could use the approximation:

$$E^S(x) = \sum_{\lambda \in \Lambda} \bar{E}_\lambda^S \tilde{\varphi}_\lambda(x) \quad (26)$$

but as we now show this does not guarantee the vanishing of self-forces. Indeed, let us assume for the rest of this section that there is a single particle located at position x . From (23), we see that, prior to the nonlinear thresholding step, the approximation by the scheme of the electron density is simply given by $\bar{\rho}_\lambda^S = \tilde{\varphi}_\lambda(x)$. Hence using the matrix notations defined above we have

$$E^S(x) = l\bar{\rho}^{S\dagger}\mathcal{G}\bar{\phi}^S = l^{-1}\bar{\rho}^{S\dagger}\mathcal{G}\mathcal{L}^{-1}\bar{\rho}^W. \quad (27)$$

Now since \mathcal{G} and \mathcal{L} are both circulating matrices, they commute, and using the fact that \mathcal{G} is antisymmetric and \mathcal{L} is symmetric, we obtain that $\mathcal{G}\mathcal{L}^{-1}$ is also antisymmetric. If we would replace ρ^W by ρ^S above, we would immediately get that $E^S(x) = 0$, but in general $\rho^W \neq \rho^S$ because of the nonlinear thresholding step. Fortunately a simple remedy exists to enforce the vanishing of self-forces in all cases. Denote by \mathcal{M} the matrix of the linear operator defined diagonally in wavelet space by the mask corresponding to the thresholding operation. By definition we have that

$$\bar{\rho}^W = \mathcal{M}\bar{\rho}^S$$

so that (27) is equivalent to

$$E^S(x) = l^{-1}\bar{\rho}^{S\dagger}\mathcal{G}\mathcal{L}^{-1}\mathcal{M}\bar{\rho}^S$$

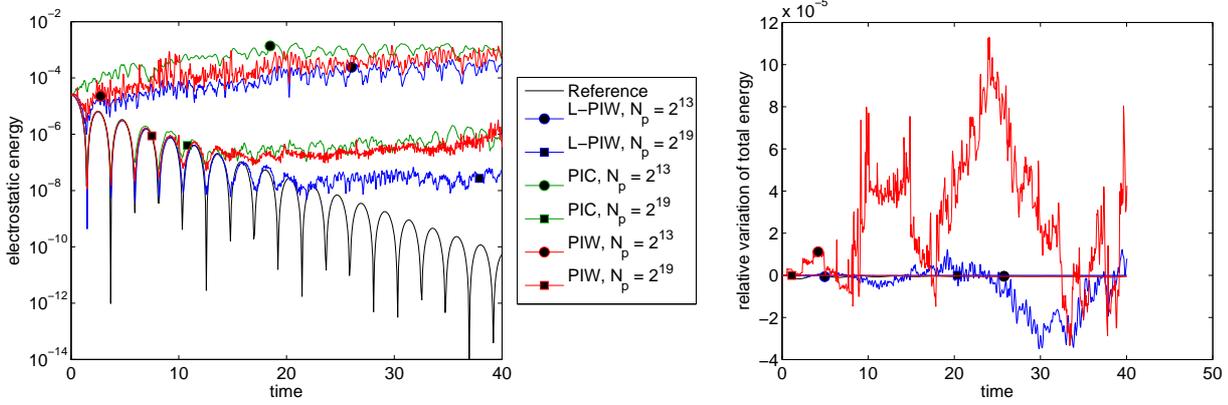


FIGURE 1. Comparison of PIW, PIC and reference solvers for the Landau damping test case. Left: electrostatic energy as a function of time. Right: kinetic energy as a function of time.

which leads us to define a new candidate for the electric field reconstruction:

$$\bar{E}^W = l\mathcal{M}\mathcal{G}\bar{\phi}^S \quad (28)$$

that is, E^W is obtained from E^S by discarding the same wavelet coefficients that were discarded when going from ρ^S to ρ^W . The self-force corresponding to E^W is obtained as before:

$$E^W(x) = l^{-1}\bar{\rho}^{S\dagger}\mathcal{M}\mathcal{G}\mathcal{L}^{-1}\mathcal{M}\bar{\rho}^S$$

but now, using the fact that \mathcal{M} is symmetric and $\mathcal{G}\mathcal{L}^{-1}$ is antisymmetric, we obtain as desired

$$E^W(x) = 0.$$

Once we know how to interpolate the electric field at the particle positions, the advancement in time of (7) is done using a classical Verlet integrator.

3. NUMERICAL RESULTS

3.1. Landau damping

To begin with we consider a Landau damping test case, for which the initial condition is given by

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

where $k = 0.5$, $l = 4\pi$, $\varepsilon = 10^{-3}$. The small value of ε implies that the evolution is dominated by the linear transport effect. The reference solution used for comparison was obtained with the SL solver, using 4097 grid points in the x and v directions. The timestep was $\delta t = 0.025$ for the reference solution, and $\delta t = 0.05$ for the others.

The time evolution of the electrostatic energy $\mathcal{V}(t)$ is shown in Fig. 1 (left). Thanks to the high order interpolation properties of the wavelets, the PIW solution matches the decay of the electrostatic energy better than the PIC one, although the behaviors of PIC and PIW at later times are identical. The time evolution of total energy $\mathcal{E}(t)$ for the PIW and reference scheme is shown in Fig. 1 (right). Note that the kinetic energy

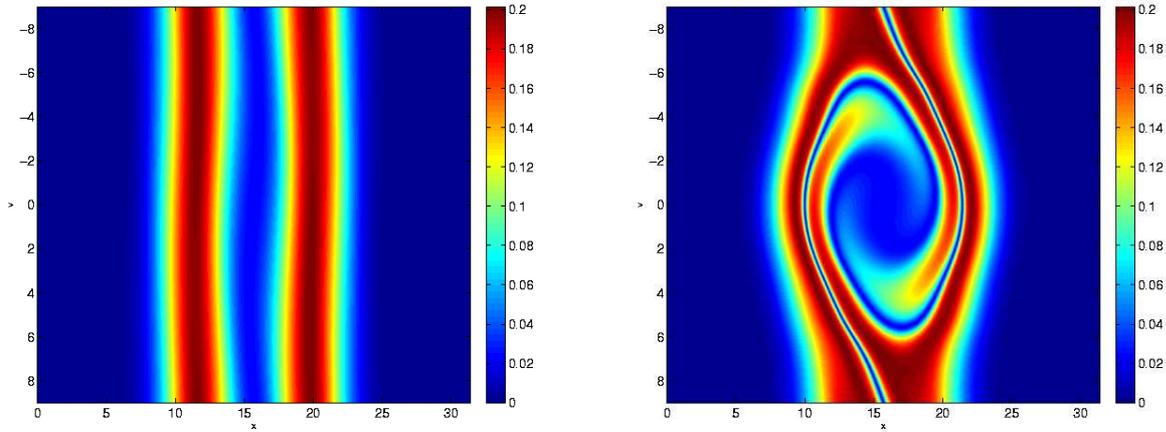


FIGURE 2. Two streams in stability test case. Particle distribution function obtained using the reference solver at $t = 10$ (left) and $t = 30$ (right).

appearing in $\mathcal{E}(t)$ is approximated by a discrete sum over the particles. Variations remain small for the two schemes, and are especially low for the PIW scheme with 2^{19} particles.

This classical test case serves to validate the PIW method, but since the solution remains very smooth it is not very well suited to provide more insight into the effect of nonlinear thresholding. Indeed, the linear L-PIW method performs better than the nonlinear PIW method. We therefore move to the more challenging two-stream instability.

3.2. Two-streams instability

The initial condition for this test case is

$$f_0(x, v) = l^{-1} (1 + \varepsilon \cos(kx)) \frac{1}{\sqrt{2\pi}} \left(e^{-\frac{(v-v_0)^2}{2}} + e^{-\frac{(v+v_0)^2}{2}} \right),$$

where $k = 0.2$, $l = 10\pi$, $\varepsilon = 10^{-2}$, and $v_0 = 2.4$. The reference distribution function is shown at $t = 10$ and $t = 30$ in Fig. 2. It was obtained using the SL solver with 4097 grid points in the x and v directions. The timestep was $\delta t = 0.0125$ for the reference solution, and $\delta t = 0.05$ for the others.

For this test case the electrostatic energy (Fig. 3, left) first oscillates for a few periods, roughly up to $t = 10$, and then undergoes an exponential growth which leads to a strongly nonlinear behavior. The PIC and PIW schemes with 2^{19} particles allow for a good approximation of the electrostatic energy up to $t = 30$. Moreover, the PIW schemes conserve the total energy up to a precision of the order of 10^{-4} even in the nonlinear stage (Fig. 3, right) of the instability, which is similar to the reference semi-Lagrangian scheme.

To assess the quality of the approximations of the electric field in the PIC and PIW schemes, we then focus on the relative L^2 error:

$$\delta_E(t) = \frac{\|E(x, t) - E_{ref}(x, t)\|}{\|E_{ref}(x, t)\|},$$

where $E_{ref}(x, t)$ is the electric field computed using the reference SL solver. To allow for the computation of $\delta_E(t)$, $E(x, t)$ is interpolated on the grid corresponding to $E_{ref}(x, t)$. In the PIW case, $E(x, t)$ is first reconstructed on a grid from its scaling function coefficients using the methods presented in [10].

$\delta_E(t)$ is shown in Fig. 4 (left) at $t = 30$, as a function of the number of particles. For all considered methods, the scaling lies between N_p^{-1} and $N_p^{-\frac{1}{2}}$. PIW offers an improvement of roughly a factor 3 over PIC, over a wide

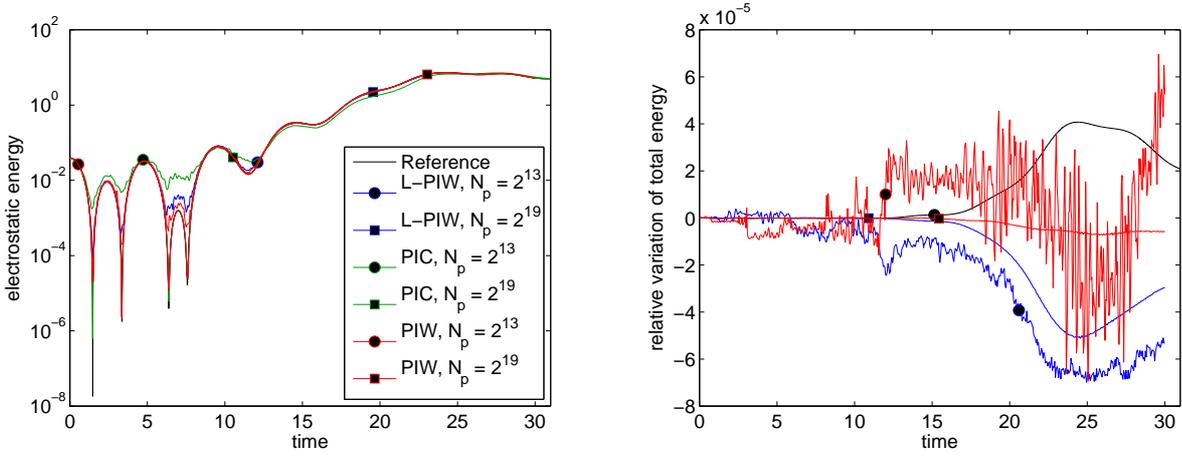


FIGURE 3. Comparison of PIW, PIC and reference solvers for the two-stream instability test case. Left: electrostatic energy as a function of time (Eq. 5). Right: total energy as a function of time (Eq. 6).

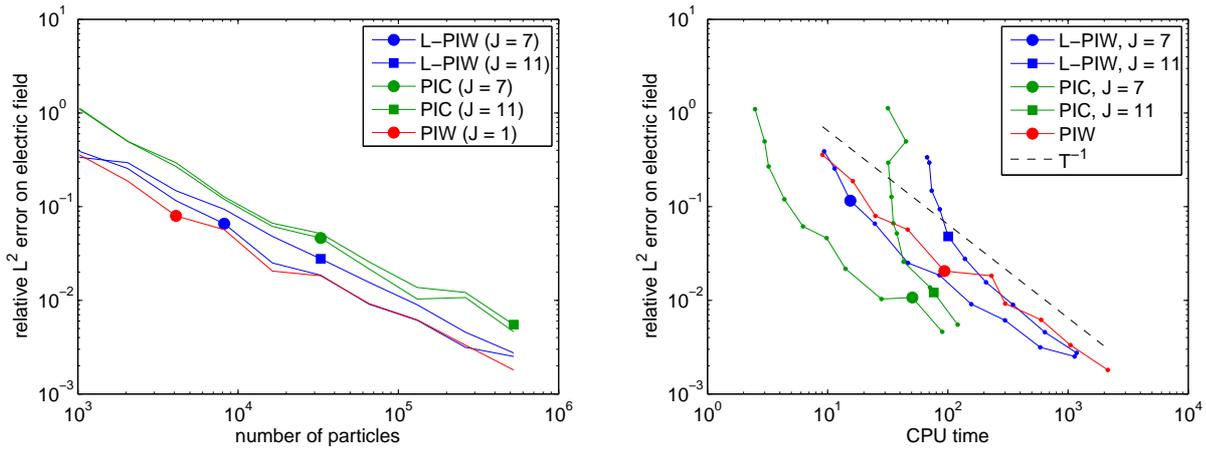


FIGURE 4. Two streams instability test case. L^2 error on the electric field at $t = 30$, as a function of number of particles (left) and computing time (right).

range of particle numbers. Linear PIW with $J = 7$ is a good candidate but starts to saturate for $N_p = 2^{19}$, probably because the spatial discretization error is reached. In contrast, nonlinear PIW remains uniformly better for any number of particles, thanks to its adaptive properties. In the right panel of Fig. 4, $\delta_E(t)$ is plotted versus the total CPU time (in seconds) needed for the integration of the equations on the time interval $[0, 30]$. The same computer was used in all cases, but the comparison between the absolute values of the CPU time should be made with caution since the languages that were used are different (Fortran for the PIC code, and C++ for the PIW code). Anyway, an interesting observation which can be made is that the decrease of the error with computing time for the PIW code is uniformly N_p^{-1} over the whole range of considered parameters. For the PIC and L-PIW methods, in contrast, a change of slope is observed when the number of particles is too low for a given spatial resolution.

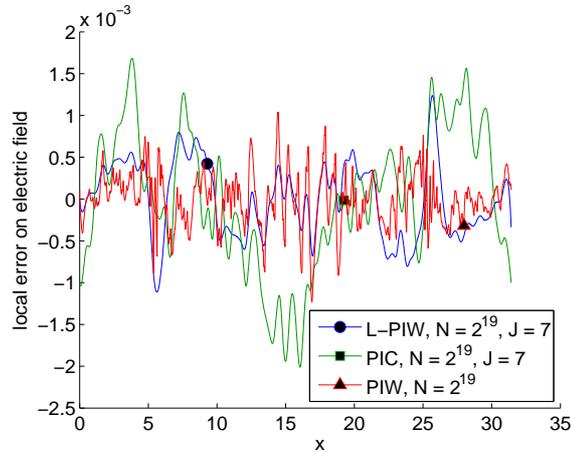


FIGURE 5. Two streams in stability test case. Error on the electric field as a function of space at $t = 30$.

These results indicate that the increased computational cost of PIW scheme as compared to PIC, which is inevitable given the length of the wavelet stencil, is only partially balanced by its improved precision. Indeed, for a given error on the electric field, the PIC scheme outperforms the PIW scheme by being roughly 5 times faster. However, in terms of memory usage the situation is more favorable to the PIW scheme, since fewer particles are necessary to reach the same precision.

To get a feeling of how the error we make on the electric field, we plot it as a function of x for $t = 30$ (Fig. 5). It seems that the error is still dominated by fine scale oscillations which appear spread out over the whole domain. Future studies will be necessary to assess if the WBDE thresholding algorithm can be adjusted in order to denoise the electric field in a more efficient way.

4. DISCUSSION

The PIW approach was proposed as an alternative to PIC for plasma simulation using particles, and successfully implemented to solve the simple one dimensional Vlasov-Poisson toy model. The main conceptual difference between PIC and PIW is that the empirical distribution of particles is convolved with a kernel for the former, whereas it is projected onto a finite dimensional linear space for the latter. Thanks to the use of wavelets to generate the projection space, the denoising step which can be included in PIW and which we have studied is genuinely different from those that are developed for PIC schemes. Indeed, the wavelet representation can adapt locally to the regularity of the density, whereas linear approaches such as Krook operators [19], Fourier filtering [16] or coarse graining [4], cannot.

The adaptive denoising strategy is especially useful when the density develops sharp features, which is known to occur generically as soon as nonlinear effects become important. To better understand this effect, we have compared a linear Landau damping test case, where the density remains homogeneously smooth, to a two stream instability test case. We have shown that for the first, PIW behaves as a higher order PIC approach, offering a constant improvement in terms of L^2 error, but no improvement in rate of convergence. In contrast, for the second test case, fine resolution was required to resolve sharp features developing in the electric field as a result of nonlinear effects. The PIW scheme is able to automatically increase the resolution locally when the number of particles is sufficient to guarantee that the estimation of the density will not get corrupted by noise. This adaptivity guarantees that the error decays like the inverse of computing time over a wide range of particle numbers. The topic of discretization error control using wavelets has been studied elsewhere, see e.g. [23] for a review in the context of computational fluid dynamics.

For the simple 1D case considered here, the improvement observed was roughly a factor 3 of precision. The increase in computational cost of the PIW scheme with respect to the PIC scheme is proportional to the length of the wavelet filter, which we have fixed to 6 in this study. Extending the full PIW scheme to more realistic 2D or 3D geometries is a much more ambitious challenge. A first undertaking, complementary to the present study, could be to interface wavelet denoising using WBDE in an existing production code in order to further support the possible benefits of the approach.

REFERENCES

- [1] A. Y. AYDEMIR. A unified Monte Carlo interpretation of particle simulations and applications to non-neutral plasmas. *Phys. Plasmas*, **1**, no. 4, (1994), 822–831.
- [2] G. BEYLKIN. On the representation of operators in bases of compactly supported wavelets. *SIAM J. Numer. Anal.*, **29**, no. 6, (1992), 1716–1740.
- [3] C. K. BIRDSALL and A. B. LANGDON. *Plasma Physics via Computer Simulation*. McGraw-Hill, New-York (1985).
- [4] Y. CHEN and S. E. PARKER. Coarse-graining phase space in delta f particle-in-cell simulations. *Phys. Plasmas*, **14**, no. 8, (2007), 082301.
- [5] E. CORMIER-MICHEL, B. A. SHADWICK, C. G. R. GEDDES, E. ESAREY, C. B. SCHROEDER and W. P. LEEMANS. Unphysical kinetic effects in particle-in-cell modeling of laser wakefield accelerators. *Phys. Rev. E*, **78**, no. 1, (2008), 016404.
- [6] I. DAUBECHIES. Orthonormal bases of compactly supported wavelets II. Variations on a theme. *J. Math. Anal.*, **24**, no. 2, (1993), 499–519.
- [7] I. DAUBECHIES and J. C. LAGARIAS. Two-scale difference equations. I. Existence and global regularity of solutions. *SIAM Journal on Mathematical Analysis*, **22**, no. 5, (1991), 1388–1410.
- [8] D. DONOHO and I. JONHSTONE. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, **81**, (1994), 425–455.
- [9] D. L. DONOHO, I. M. JOHNSTONE, G. KERYACHARIAN and D. PICARD. Density estimation by wavelet thresholding. *Ann. Statist.*, **24**, no. 2, (1996), 508–539.
- [10] L. GENOVESE, A. NEELOV, S. GOEDECKER, T. DEUTSCH, S. A. GHASEMI, A. WILLAND, D. CALISTE, O. ZILBERBERG, M. RAYSON, A. BERGMAN and R. SCHNEIDER. Daubechies wavelets as a basis set for density functional pseudopotential calculations. *J. Comp. Phys.*, **129**, no. 1, (2008), 014109–+.
- [11] V. GRANDGIRARD, M. BRUNETTI, P. BERTRAND, N. BESSE, X. GARBET, P. GHENDRIH, G. MANFREDI, Y. SARAZIN, O. SAUTER, E. SONNENRÜCKER, J. VACLAVIK and L. VILLARD. A drift-kinetic Semi-Lagrangian 4D code for ion turbulence simulation. *J. Comp. Phys.*, **217**, (2006), 395–423.
- [12] J. M. HAMMERSLEY, D. C. HANDSCOMB and G. WEISS. *Monte Carlo Methods*. Taylor & Francis (1975).
- [13] J. HEIKKINEN, S. JANHUNEN, T. KIVINIEMI and F. OGANDO. Full f gyrokinetic method for particle simulation of tokamak transport. *J. Comp. Phys.*, **227**, no. 11, (2008), 5582–5609.
- [14] R. W. HOCKNEY and J. W. EASTWOOD. *Computer Simulation using Particles*. IOP, Bristol, Philadelphia (1988).
- [15] Y. IDOMURA, M. IDA, S. TOKUDA and L. VILLARD. New conservative gyrokinetic full-f Vlasov code and its comparison to gyrokinetic δ -f particle-in-cell code. *J. Comp. Phys.*, **226**, no. 1, (2007), 244–262.
- [16] S. JOLLIET, A. BOTTINO, P. ANGELINO, R. HATZKY, T. TRAN, B. MCMILLAN, O. SAUTER, K. APPERT, Y. IDOMURA and L. VILLARD. A global collisionless PIC code in magnetic coordinates. *Comput. Phys. Commun.*, **177**, no. 5, (2007), 409–425.
- [17] A. B. LANGDON and C. K. BIRDSALL. Theory of plasma simulation using finite-size particles. *Phys. Fluids*, **13**, no. 8, (1970), 2115.
- [18] S. MALLAT. *A wavelet tour of signal processing*. Academic Press (1999).
- [19] B. F. MCMILLAN, S. JOLLIET, T. M. TRAN, L. VILLARD, A. BOTTINO and P. ANGELINO. Long global gyrokinetic simulations: Source terms and particle noise control. *Phys. Plasmas*, **15**, no. 5, 052308.
- [20] W. M. NEVINS, G. W. HAMMETT, A. M. DIMITS, W. DORLAND and D. SHUMAKER. Discrete particle noise in particle-in-cell simulations of plasma microturbulence. *Phys. Plasmas*, **12**, (2005), 122305.
- [21] R. NGUYEN VAN YEN, D. DEL CASTILLO-NEGRETE, K. SCHNEIDER, M. FARGE and G. CHEN. Wavelet based density estimation for noise reduction in plasma simulation using particles. *J. Comp. Phys.*, **229**, no. 8, (2010), 2821–2839.
- [22] H. OKUDA, A. T. LIN, C. C. LIN and J. M. DAWSON. Splines and high order interpolations in plasma simulations. *Comput. Phys. Commun.*, **17**, no. 3, (1979), 227–231.
- [23] K. SCHNEIDER and O. VASILYEV. Wavelet methods in computational fluid dynamics. *Annu. Rev. Fluid Mech.*, **42**, (2010), 473–503.
- [24] B. W. SILVERMAN. *Density estimation for statistics and data analysis*. Chapman and Hall (1986).
- [25] E. SONNENRÜCKER, J. ROCHE, P. BERTRAND and A. GHIZZO. The semi-Lagrangian method for the numerical resolution of the Vlasov equation. *J. Comp. Phys.*, **149**, (1999), 201–220.
- [26] B. TERZIĆ, I. V. POGORELOV and C. L. BOHN. Particle-in-cell beam dynamics simulations with a wavelet-based Poisson solver. *Phys. Rev. ST Accel. Beams*, **10**, (2007), 034201.

- [27] W. X. WANG, Z. T. W. M. LIN, W. W. LEE, S. ETHIER, J. L. V. LEWANDOWSKI, G. REWOLDT, T. S. HAHM and M. J. Gyro-kinetic simulation of global turbulent transport properties in tokamak experiments. *Phys. Plasmas*, **13**, (2006), 092505.