Semi-implicit Particle-In-Cell methods embedding sparse grid reconstructions

Clément Guillet

SEMI-IMPLICIT PARTICLE-IN-CELL METHODS EMBEDDING SPARSE GRID RECONSTRUCTIONS

C. GUILLET*

Abstract. In this article, we introduce semi-implicit Particle-In-Cell (PIC) methods based on a discretization of the Vlasov-Maxwell system in the electrostatic regime and embedding sparse grid reconstructions: the Semi-Implict Sparse-PIC (SISPIC-sg) scheme, its standard extension (SISPIC-std) and the Energy-Conserving Sparse-PIC (ECSPIC) scheme. These schemes are inspired by the Energy-Conserving Semi-Implicit Method (ECSIM) introduced in [39]. The particle equations are linearized so that the particle response to the field can be computed by solving a linear system with a stiffness matrix. The methods feature the three following properties: the scheme is unconditionally stable with respect to the plasma period; the finite grid instability is eliminated, allowing the user to use any desired grid discretization; the statistical error is significantly reduced compared to semi-implicit and explicit schemes with standard grid for the same number of particles. The ECSPIC scheme conserves exactly the discrete total energy of the system but we have experienced numerical instability related to the loss of the field energy non-negativity genuine to the sparse grid combination technique. The SISPIC methods are exempted from this instability and is unconditionally stable with respect to the time and spatial discretization, but does not conserve exactly the discrete total energy. The methods have been investigated on a series of two dimensional test cases and gains in term of memory storage and computational time compared to explicit and existing semi-implicit methods have been observed. These gains are expected to be larger for three dimensional computations for which the full potential of sparse grid reconstructions can be achieved.

Key words. plasma physics, particle-in-cell, electrostatic, sparse grid combination technique, energy conserving

MSC codes. 65N75

1. Introduction. Particle-In-Cell (PIC) method is one of the most widely spread numerical method for the simulation of kinetic plasmas [7, 22, 23, 29]. It is based on a discretization of the Vlasov-Maxwell system, or a subset thereof for electrostatic regimes. Vlasov equation describes the evolution of the probability density function of species of particles in the phase-space while Maxwell's equations (Ampere equation in this paper) characterize the evolution of the electromagnetic field. Ampere's equation is conducted by the moment of the particle distribution and Vlasov equation characteristics are self-consistently determined by the field. It results in a tightly coupled non-linear system whose solutions are proven to be challenging to determine. The specificity of PIC methods is the mixed discretization, made of both an Eulerian grid for the moments of the particle distribution and fields, in conjunction with individual Lagrangian particles in continuous phase space.

The solutions of Vlasov-Maxwell equations verify some conservation properties, such as the conservation of the total energy and momentum of the system. The charge continuity equation is also a consequence of the Vlasov equations (moment of order 0). The question of conservation of these physical quantities in numerical simulations has been very popular for years. Explicit formulations of PIC methods, based on an explicit time integration of the characteristics of the Vlasov equation, are usually momentum-conserving but not energy-conserving. Conversely, PIC implementations based on an implicit formulation can be energy-conserving but not momentum-conserving. The question whether a numerical scheme preserving both energy and momentum is possible or not is addressed in [8].

Originally, and still in most applications, PIC implementations are based on an explicit time discretization of the Vlasov equation, *e.g.* by means of a leap-frop scheme. An explicit time integration benefits from simplicity of implementation, as well as a poor computational cost per iteration. Nonetheless, explicit approaches suffer from temporal stability constraints,

*Corresponding author
Université de Toulouse; UPS, INSA, UT1, UTM, Institut de Mathématiques de Toulouse
LAPLACE, Université de Toulouse, CNRS, INPT, UPS, 118 Route de Narbonne, 31062 Toulouse, France

imposing a limit on the time-step discretization, to resolve the fastest wave. In addition, these approaches usually feature spatial stability constraints, manifested by numerical instabilities as aliasing or finite grid instability [37, 35] occurring when the grid discretization (grid cell size) is equal to or superior to the Debye length of the plasma. Therefore, the application of explicit approaches to multidimensional problems, especially for three dimensional geometries or large plasma densities, can be very computationally demanding and cumbersome.

In response to these issues, implicit formulations of PIC schemes have emerged [27] and received a lot of attention, particularly thanks to their stability properties. Indeed, (semi-)implicit PIC methods such as the implicit-moment method [9, 41], direct implicit method [38, 17, 33] and their developments alleviate the numerical constraints, preserving stability with larger time-steps and grid discretizations. Ideally, in implicit formulations, the particle equations and the field equations shall be non-linearly coupled, requiring Newton or Picard iterations. Because of solver efficiency limitations at the early development of implicit methods, linear approximations have been favored at the expense of numerical approximations producing violation of the energy conservation and resulting in significant artificial plasma heating or cooling. The methods using a linearization of the particle-field coupling are named semi-implicit methods. Recently, a fully implicit approach [14], based on Newton–Krylov methods, in which field-particle couplings are converged to a tight nonlinear tolerance has been developed. In addition to the elimination of both temporal and spatial stability constraints, the scheme offers valuable conservation properties, such as the exact conservation of the discrete energy of the system and consistency with the charge continuity equation. Nonetheless, the method requires the resolution of a non-linear system for the particles and field, which can be very computationally expensive, especially for multidimensional computations. A few years ago, a semi-implicit method preserving exactly the discrete total energy of the system [39] has been developed. This Energy-Conserving Semi-Implicit Method (ECSIM) retains the simplicity of explicit schemes, *i.e.* it advances the particles first and then the fields without any iteration, and conserves discrete energy exactly. In this approach, the particle-field coupling is partially linearized, meaning that a part of the particle response to the field is computed thanks to a mass matrix, ensuring an exact discrete energy-conservation. Compared to the previous semi-implicit methods, namely the implicit-moment method and the direct implicit method, the particle pusher and the derivation of the field equation are different. The mover does not require any inner iteration and its complexity is similar to that of explicit formulations. Nonetheless, the field matrix presents a significantly more complex structure in comparison to that of explicit schemes in order to conserve energy to round-off errors. The major advantage over fully-implicit schemes is the reduced complexity of the algorithm, allowing development of the method for three dimensional simulations. Since then, the method has been applied extensively to large-scale kinetic simulations [46, 45, 13, 21]. However, the method is not consistent with the charge continuity equation as the fully-implicit method does. Therefore the error of conservation, or equivalently the consistency with the Gauss law for Vlasov-Ampere (VA) formulations, shall be corrected throughout the simulation in order to avoid a non-physical evolution of the plasma. Since then, developments addressing this charge continuity issue have been introduced. In [16], a correction inspired of the Boris ($\nabla \cdot \mathbf{E}$) correction, but operating on the particles instead of the field in order to preserve energy conservation is proposed. The method uses local linearization of the particle shape functions and requires the resolution of an under-determined system on the particles with Lagrange multiplier method. Besides, a prediction-correction scheme [12] inspired both of the ECSIM scheme and of a charge-conserving scheme based on an averaging of grid quantities over interpolated trajectories of particles has been proposed.

Particle-In-Cell schemes also contain a major weakness: the statistical error originating from the sampling of the probability density function by a finite number of numerical particles.

This numerical noise decreases slowly with the increase of the average number of particles per cell, scaling as the inverse square root of the mean number of particles per cell. Therefore, a large number of particles may be required for some simulations, necessitating tremendous computational resources. Noise reduction strategies aim at maintaining the accuracy of computations with a reduced set of particles. They have therefore received a lot of attention with, for instance, variance reduction methods such as the δf method [28] or the quiet start initialization procedure [44] as well as filtering methods in either Fourier domain [7], wavelet domain [30], and micro-macro decomposition [20, 19, 18].

Sparse grid reconstructions in PIC methods aim at reducing the statistical error resulting from the particle sampling. Specifically, the particle distribution moments are computed on a hierarchy of component grids with a coarse resolution. Compared to standard grids, the mean number of particles per cell is larger for any of the component grids. This crucial feature offers either a mitigation of the statistical noise or a decrease of the total number of numerical particles required for a precision comparable to discretizations on a standard grid. The method has already been applied to explicit PIC discretizations of the Vlasov-Poisson model in two dimensions [26], and three dimensions [43, 42, 24, 25]. Substantial gains in term of memory consumption as well as computational time have been pointed out, by two or three orders of magnitude in comparison to approaches with standard grids. Besides, sparse grid reconstructions have proven to preserve exact momentum conservation of explicit formulations. These observations call for the development of an implicit formulation embedding sparse grid reconstructions with improved spatial and temporal stability properties and, if possible, discrete total energy conservation.

This paper focuses on the development of semi-implicit schemes incorporating sparse grid reconstructions. Its contribution lies in the introduction of three new semi-implicit methods (SISPIC-std, SISPIC-sg, ECSPIC) for PIC simulations, proposed as enhancements over the ECSIM method and the explicit sparse-PIC method introduced in the key references [39] and [26]. The schemes presented here are inspired by the ECSIM scheme and are based on a linearization of the equations to obtain the implicit particle response to the electric field by solving a linear system. All three schemes differ from the ECSIM scheme in that they consider an electrostatic regime where Maxwell's equations reduce to the Ampere equation without a magnetic field. To our knowledge, all ECSIM methods described in the literature are based on the Vlasov-Maxwell system and include a self-consistent magnetic field. The aim here is to address the difficulty associated with computing an electromagnetic field free from any solenoidal (or inductive) component. The schemes introduced here are based on an electrostatic Vlasov-(div)Ampere formulation, ensuring they are genuinely consistent with this property: they do not require any post-processing typically implemented in this context. The resulting linear system in each of our approaches is constructed from one stiffness matrix (instead of $3v$ mass matrices in ECSIM), and its size is reduced by 6 compared to the ECSIM scheme since the unknowns are scalar. To summarize the various improvements and trade-offs offered by the new schemes compared to ECSIM and the explicit sparse-PIC methods, we provide Table 1 summarizing the following properties verified by the schemes:

- P_1: The scheme is consistent with an electrostatic formulation.
- P_2: The scheme is unconditionally stable with respect to the plasma period: the time step can be chosen irrespective of this value.
- P_3: Aliasing or finite grid instability is eliminated, allowing grid discretization without constraints related to the Debye length.
- P_4: The statistical error is significantly reduced compared to standard PIC schemes carried out with a Cartesian grid of comparable resolution and the same number of particles.
- P_5: The discrete total energy of the system is exactly conserved for any discretization

parameters.

P_6: The total momentum of the system is exactly conserved for any discretization parameters.

P_7: The scheme is exempt from numerical instabilities.

Table 1: Properties P_1-P_7 of the newly introduced schemes (SISPIC-std, SISPIC-sg, ECSPIC) and the reference schemes (explicit sparse-PIC [26], ECSIM [39]).

scheme	P_1	P_2	P_3	P_4	P_5	P_6	P_7
explicit sparse-PIC	✓	×	×	✓	×	✓	×
ECSIM	×	✓	✓	×	✓	×	✓
SISPIC-std	✓	✓	✓	×	✓	×	✓
SISPIC-sg	✓	✓	✓	✓	×	×	✓
ECSPIC	✓	✓	✓	✓	✓	×	×

For P_4, the reduction of statistical noise is achieved either through sparse grid reconstructions, the Vlasov-(div)Ampere formulation, or both. This contribution is significant because, as a consequence of their typical Vlasov-Ampere formulation, semi-implicit methods tend to produce more statistical noise than explicit methods. P_7 is not satisfied by the ECSPIC scheme. Indeed, instabilities were observed during numerical investigations, attributed to the absence of sparse grid reconstructions to maintain the non-negativity of the solution.

This paper is organized as follows. In section 2, the general framework of the article is introduced: Vlasov-(div)Ampere formulation and PIC methods. In section 3, the sparse grid reconstructions are introduced, as well as the SISPIC-sg method which is compared to existing semi-implicit methods (direct implicit, moment implicit, ECSIM). Some insights of the gains provided by the method are also discussed. In section 4, the SISPIC-std and ECSPIC methods are introduced and the computational complexity, memory requirements and properties (energy conservation, loss of non-negativity) of the latter are discussed. Finally, in section 5, the methods are investigated and compared to existing methods (sparse and standard explicit schemes, ECSIM) on two-dimensional classical test cases: Landau damping and two-streams instability.

2. General framework.

2.1. Notations. Let $d \in \mathbb{N}^*$ be the dimension of the problem and let the spatial domain be the d-dimensional periodic unit interval $\Omega = (\mathbb{R}/\mathbb{Z})^d$. For multi-indexes $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_d) \in \mathbb{N}^d$ and $\boldsymbol{\beta} = (\beta_1, ..., \beta_d) \in \mathbb{N}^d$, let us define order relations by:

$$\boldsymbol{\alpha} \leq \boldsymbol{\beta} \;\Leftrightarrow\; \forall i \in \{1, ...d\}\ \alpha_i \leq \beta_i, \tag{2.1}$$

$$\boldsymbol{\alpha} < \boldsymbol{\beta} \;\Leftrightarrow\; \boldsymbol{\alpha} \leq \boldsymbol{\beta} \text{ and } \exists\, i \in \{1, ..., d\}\ s.t.\ \alpha_i < \beta_i, \tag{2.2}$$

and introduce the notations:

$$\boldsymbol{\alpha}\boldsymbol{\beta} = (\alpha_1\beta_1, ..., \alpha_d\beta_d), \qquad \boldsymbol{\alpha}^{-1} = \frac{1}{\alpha_1 ... \alpha_d}. \tag{2.3}$$

The l^1 norm for a multi-index $\boldsymbol{\alpha} \in \mathbb{N}^d$ is also introduced:

$$|\boldsymbol{\alpha}|_1 := \sum_{i=1}^{d} |\alpha_i|. \tag{2.4}$$

2.2. Electrostatic Vlasov-(div)Ampere (VdA) formulation. The semi-implicit PIC methods introduced in this paper are based on a discretization of Vlasov-Ampere system in an electrostatic regime, assuming a vanishing magnetic field $\mathbf{B} = 0$. In this regime, the following Vlasov-Ampere (VA) formulation is considered:

$$(VA): \begin{cases} \dfrac{\partial f_s}{\partial t}(\mathbf{x},\mathbf{v},t) + \mathbf{v}\cdot\nabla_{\mathbf{x}} f_s(\mathbf{x},\mathbf{v},t) + \dfrac{q_s}{m_s}\mathbf{E}(\mathbf{x},t)\cdot\nabla_{\mathbf{v}} f_s(\mathbf{x},\mathbf{v},t) = 0, \\ \nabla\times\mathbf{E}(\mathbf{x},t) = 0, \\ \epsilon_0 \dfrac{\partial \mathbf{E}}{\partial t}(\mathbf{x},t) = -\mathbf{J}(\mathbf{x},t). \end{cases} \tag{2.5}$$

The system is defined for $(\mathbf{x},\mathbf{v},t) \in \Omega\times\mathbb{R}^d\times\mathbb{R}^+$. In this problem, $f_s(\mathbf{x},\mathbf{v},t)$ is the phase-space distribution function attached to the species s; q_s, m_s are the corresponding charge and mass, ϵ_0 is the vacuum permittivity, $\mathbf{E}$ is the electric field and $\mathbf{J}$ is the plasma current density obtained from the phase-space distribution of each species:

$$\mathbf{J}(\mathbf{x},t) = \sum_s \mathbf{J}_s(\mathbf{x},t) = \sum_s q_s \int \mathbf{v} f_s(\mathbf{x},\mathbf{v},t) d\mathbf{v}. \tag{2.6}$$

The electric field is initialized with the Gauss law and requires the resolution of a Poisson equation for the electric potential, denoted Φ:

$$\mathbf{E}(\mathbf{x},0) = -\boldsymbol{\nabla}\Phi(\mathbf{x},0), \qquad -\varepsilon_0\Delta\Phi(\mathbf{x},0) = \rho(\mathbf{x},0), \tag{2.7}$$

where $\rho(\mathbf{x},0)$ is the plasma charge density at initialization defined from the initial distribution of each species:

$$\rho(\mathbf{x},t) = \sum_s \rho_s(\mathbf{x},t) = \sum_s q_s n_s, \quad n_s = \int f_s(\mathbf{x},\mathbf{v},t) d\mathbf{v}. \tag{2.8}$$

Remark 2.1. Provided that the charge continuity equation is verified:

$$\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla}\cdot\mathbf{J} = 0, \tag{2.9}$$

the Vlasov-Ampere formulation (2.5) is equivalent to a Vlasov-Poisson formulation:

$$(VP): \begin{cases} \dfrac{\partial f_s}{\partial t}(\mathbf{x},\mathbf{v},t) + \mathbf{v}\cdot\nabla_{\mathbf{x}} f_s(\mathbf{x},\mathbf{v},t) + \dfrac{q_s}{m_s}\mathbf{E}(\mathbf{x},t)\cdot\nabla_{\mathbf{v}} f_s(\mathbf{x},\mathbf{v},t) = 0, \\ -\epsilon_0\Delta\Phi(\mathbf{x},t) = \rho(\mathbf{x},t), \quad \mathbf{E}(\mathbf{x},t) = -\boldsymbol{\nabla}\Phi(\mathbf{x},t). \end{cases} \tag{2.10}$$

By considering the charge continuity equation (2.9), one can derive from the Ampere equation an evolution equation for the electric potential Φ:

$$\varepsilon_0 \frac{\partial \Delta\Phi}{\partial t}(\mathbf{x},t) = \boldsymbol{\nabla}\cdot\mathbf{J}(\mathbf{x},t). \tag{2.11}$$

The equation can alternatively be obtained by considering the divergence of the Ampere equation (third equation of the system (2.5)) and the Gauss law ($\epsilon_0\nabla\cdot\mathbf{E} = \rho$). From this equation, a multi-dimensional electrostatic Vlasov-(div)Ampere formulation is considered:

$$(VdA): \begin{cases} \dfrac{\partial f_s}{\partial t}(\mathbf{x},\mathbf{v},t) + \mathbf{v}\cdot\nabla_{\mathbf{x}} f_s(\mathbf{x},\mathbf{v},t) + \dfrac{q_s}{m_s}\mathbf{E}(\mathbf{x},t)\cdot\nabla_{\mathbf{v}} f_s(\mathbf{x},\mathbf{v},t) = 0, \\ \epsilon_0 \dfrac{\partial \Delta\Phi}{\partial t}(\mathbf{x},t) = \boldsymbol{\nabla}\cdot\mathbf{J}(\mathbf{x},t), \quad \mathbf{E}(\mathbf{x},t) = -\boldsymbol{\nabla}\Phi(\mathbf{x},t). \end{cases} \tag{2.12}$$

The formulation is equivalent to the first one (2.5) in multi-dimensions if the charge continuity equation (or Gauss law) is verified. Indeed, since the electric field is derived from a potential, its curl vanishes:

$$\boldsymbol{\nabla} \times \mathbf{E} = -\boldsymbol{\nabla} \times \boldsymbol{\nabla} \Phi = 0. \tag{2.13}$$

In this article, our goal is to derive semi-implicit methods which are based on discretizations of the (VdA) system.

2.3. PIC discretizations. The distribution of particles (f_s) is represented by a collection of macro-particles. A macro-particle, also called numerical particle, refers to a heap of physical particles of the same species (electrons, ions, etc.). Let N_s denotes the number of macro-particles attached to the species s and N the total number of particles. The positions and velocities of a particle at time t are denoted $(\mathbf{x}_p(t), \mathbf{v}_p(t))$, $p = 1, ..., N_s$ being the index of the particles. We assume that all the numerical particles of one species have the same weight, defined by the ratio of physical particles (n_s) per numerical particle (N_s):

$$\omega = \frac{\int_\Omega n_s d\mathbf{x}}{N_s}, \tag{2.14}$$

and the same charge and mass:

$$q_p = q_s \omega, \quad m_p = m_s \omega, \qquad \forall p = 1, ..., N_s. \tag{2.15}$$

A shape function, denoted W_h, dependent on the grid discretization parameter h and defined when the latter is specified, is associated to the particles in order to accumulate the particle properties (charge density, current density) onto the mesh. The current density accumulation is defined by:

$$\mathbf{J}_h(\mathbf{x}, t) := \sum_s \mathbf{J}_{s,h}(\mathbf{x}, t) := \sum_s \frac{1}{h^d} \sum_{p=1}^{N_s} q_p \mathbf{v}_p(t) W_h(\mathbf{x} - \mathbf{x}_p(t)), \tag{2.16}$$

where the quantity $1/h^d$ is the volume of the grid cells. The electric field is then computed on the mesh with Ampere (or Poisson) equation and interpolated at the particles positions. Finally the particles are advanced by considering the characteristics of Vlasov equation.

3. Semi-implicit PIC methods.

3.1. Implicit formulation and linearization. Let Δt be the step of the time discretization and let us denote a quantity by the superscript $k \in \mathbb{N}$ the evaluation of this quantity at iteration k:

$$t^k := k\Delta t, \quad \mathbf{x}_p^k := \mathbf{x}_p(t^k), \quad \mathbf{v}_p^k := \mathbf{v}_p(t^k), \text{ etc.} \tag{3.1}$$

Let us introduce our implicit scheme used for time discretization of the electrostatic (VdA) system, where the position is staggered half a time step with respect to the velocities and the fields as follows:

$$\begin{cases} \mathbf{x}_p^{k+\frac{1}{2}} = \mathbf{x}_p^{k-\frac{1}{2}} + \Delta t \mathbf{v}_p^{k+\frac{1}{2}} \\ \mathbf{v}_p^{k+1} = \mathbf{v}_p^k + \Delta t \dfrac{q_p}{m_p} \mathbf{E}_h^{k+\frac{1}{2}} \left(\mathbf{x}_p^{k+\frac{1}{2}} \right) \\ \Delta_h \Phi_h^{k+1} - \Delta_h \Phi_h^k = \dfrac{\Delta t}{\varepsilon_0} \nabla_h \cdot \mathbf{J}_h^{k+\frac{1}{2}} \\ \mathbf{E}_h^{k+1} = -\nabla_h \Phi_h^{k+1} \end{cases}, \quad k \in \mathbb{N}. \tag{3.2}$$

The spatial discretization, designated by the subscript h which corresponds to the size of the grid cells, is not detailed here and shall be precised later. The operators $\nabla_h, \nabla_h \cdot, \Delta_h$ are defined in the appendix (equations (A.6), (A.7) and (A.8)). The following averaged quantities have been introduced:

$$\mathbf{v}_p^{k+\frac{1}{2}} = \frac{1}{2}\left(\mathbf{v}_p^{k+1} + \mathbf{v}_p^k\right), \quad \mathbf{E}_h^{k+\frac{1}{2}}\left(\mathbf{x}_p^{k+\frac{1}{2}}\right) := \frac{1}{2}\left[\mathbf{E}_h^{k+1}\left(\mathbf{x}_p^{k+\frac{1}{2}}\right) + \mathbf{E}_h^k\left(\mathbf{x}_p^{k+\frac{1}{2}}\right)\right]. \tag{3.3}$$

The electric field is interpolated from the grid to the particle positions according to:

$$\mathbf{E}_h^{k+1}\left(\mathbf{x}_p^{k+\frac{1}{2}}\right) = \left(\mathcal{I}_{V_h}\mathbf{E}_h^{k+1}\right)\left(\mathbf{x}_p^{k+\frac{1}{2}}\right). \tag{3.4}$$

The interpolation operator $\mathcal{I}_{V_h}$ is not explicitly defined at this point, as it depends on the grid discretizations and will vary depending on the schemes introduced in the following sections. The implicit current density is defined by:

$$\mathbf{J}_h^{k+\frac{1}{2}}(\mathbf{x}) = \frac{1}{h^d}\sum_s \sum_{p=1}^{N_s} q_p \mathbf{v}_p^{k+\frac{1}{2}} W_h\left(\mathbf{x} - \mathbf{x}_p^{k+\frac{1}{2}}\right). \tag{3.5}$$

The scheme introduced in equation (3.2) is fully implicit because of the coupling between the particles and the field. It requires the resolution of a non-linear system. Semi-implicit approaches, such as the moment implicit method [9, 41], the direct implicit method [38, 34], etc. or the method introduced here are based on a linearization of the equations. Nonetheless, the method introduced in this paper is different from the direct implicit method (in which the shape function W_h is linearized using Taylor expansions) and the moment implicit method (in which the linearization is obtained from moment of the vlasov equation). The method is inspired of the ECSIM method introduced in [39]. The linearization is obtained directly in the particle equations by updating the position of the particles with a known velocity, *i.e.* the first equation in (3.2) becomes:

$$\mathbf{x}_p^{k+\frac{1}{2}} = \mathbf{x}_p^{k-\frac{1}{2}} + \frac{\Delta t}{2}\left[\mathbf{v}_p^k + \Delta t \frac{q_p}{m_p}\mathbf{E}_h^{k+\frac{1}{2}}\left(\mathbf{x}_p^{k+\frac{1}{2}}\right) + \mathbf{v}_p^k\right] \tag{3.6}$$

$$= \mathbf{x}_p^k + \Delta t \mathbf{v}_p^k + O\left(\Delta t^2\right). \tag{3.7}$$

As a result, the implicit model has become linear so that the implicit contribution of the electric field (evaluated at the known particle position) may be obtained from the divergence of Ampere equation (third equation of (3.2)) by solving a linear system. The particle equations are then:

$$\begin{cases} \mathbf{x}_p^{k+\frac{1}{2}} = \mathbf{x}_p^{k-\frac{1}{2}} + \Delta t \mathbf{v}_p^k \\ \mathbf{v}_p^{k+1} = \mathbf{v}_p^k + \Delta t \dfrac{q_p}{m_p}\mathbf{E}_h^{k+\frac{1}{2}}\left(\mathbf{x}_p^{k+\frac{1}{2}}\right) \end{cases}, \quad k \in \mathbb{N}. \tag{3.8}$$

3.2. Sparse grid reconstructions. In this section, the sparse grid notations are introduced in the framework of the so-called sparse grid combination technique [31].

Definition 3.1 (Component indices). *Let $\mathbb{L}_n$ be a set of indices, called the component indices, with respect to the discretization parameter $n \in \mathbb{N}$ and defined by:*

$$\mathbb{L}_n := \bigcup_{i \in [\![0, d-1]\!]} \mathbb{L}_{n,i}, \quad \mathbb{L}_{n,i} := \{\boldsymbol{l} \in \mathbb{N}^d \mid |\boldsymbol{l}|_1 = n + d - 1 - i,\ \boldsymbol{l} \geq \boldsymbol{1}\}, \tag{3.9}$$

Let us consider the family of d-dimensional anisotropic grids on the space domain indexed by the component indices and called component grids:

Definition 3.2 (Component grids). *The component grids are defined for* $\boldsymbol{l} \in \mathbb{L}_n$ *by:*

$$\Omega_{h_{\boldsymbol{l}}} := \left\{ \boldsymbol{j} h_{\boldsymbol{l}} \,|\, \boldsymbol{j} \in I_{h_{\boldsymbol{l}}} \right\} \in \Omega, \quad I_{h_{\boldsymbol{l}}} := [\![0, h_{l_1}^{-1} - 1]\!] \times ... \times [\![0, h_{l_d}^{-1} - 1]\!] \subset \mathbb{N}^d, \tag{3.10}$$

where:

$$h_{\boldsymbol{l}} := (h_{l_1}, ..., h_{l_d}) \in \mathbb{R}^d, \;\; h_l = 2^{-l} \text{ for } l \in \mathbb{L}_n \tag{3.11}$$

is called the grid discretization and corresponds to the cell grid width.

The number of component grids is given by:

$$|\mathbb{L}_n| := \text{ Card}(\mathbb{L}_n) = \sum_{i=0}^{d-1} \binom{n+d-2-i}{d-1} = O\left(|\log h_n|^{d-1} \right), \tag{3.12}$$

where $\binom{i}{j}$, for $i \geq j$ integers, is the binomial coefficient defined as:

$$\binom{i}{j} := \frac{i!}{j!(i-j)!}. \tag{3.13}$$

Let us also consider a regular isotropic grid, named Cartesian grid, corresponding to a component grid of level $\mathbf{l} = n \cdot \mathbf{1}$ with uniform discretization h_n for any direction:

Definition 3.3 (Cartesian grid). *The Cartesian grid, denoted* $\Omega_{h_n}^{(\infty)}$*, is defined by:*

$$\Omega_{h_n}^{(\infty)} := \left\{ \boldsymbol{j} h_n \,|\, \boldsymbol{j} \in I_{h_n} \right\} \subset \Omega, \quad I_{h_n} := [\![0, h_n^{-1} - 1]\!]^d \subset \mathbb{N}^d. \tag{3.14}$$

Let $\mathbf{l} \in \mathbb{L}_n$, $\mathbf{j} \in I_{h_{\mathbf{l}}}$ be multi-indexes associated to a component grid and consider basis functions defined by tensor products of one-dimensional functions as follows:

$$W_{h_{\mathbf{l}};\mathbf{j}}(\mathbf{x}) := W_{h_{\mathbf{l}}}(\mathbf{x} - \mathbf{j} h_{\mathbf{l}}) = \left(\bigotimes_{i=1}^{d} W_{h_{l_i}} \right) (\mathbf{x} - \mathbf{j} h_{\mathbf{l}}), \;\; W_{h_l}(x) := W\left(h_l^{-1}(x) \right), \tag{3.15}$$

where $h_{\mathbf{l}}$ is the (component) grid discretization and W is the so-called hat function defined:

$$W(x) = \max\left(1 - |x|, 0\right). \tag{3.16}$$

These functions verify a partition of unity property:

$$\sum_{\mathbf{j} \in I_{h_{\mathbf{l}}}} W_{h_{\mathbf{l}};\mathbf{j}}(\mathbf{x}) = 1. \tag{3.17}$$

The space of d-dimensional hat functions with respect to the component grid $\Omega_{h_{\mathbf{l}}}$ is denoted $V_{h_{\mathbf{l}}}$ and defined by:

$$V_{h_{\mathbf{l}}} := \text{span}\{W_{h_{\mathbf{l}};\mathbf{j}} \mid \mathbf{j} \in I_{h_{\mathbf{l}}}\}, \tag{3.18}$$

where $\{W_{h_{\mathbf{l}};\mathbf{j}} \mid \mathbf{j} \in I_{h_{\mathbf{l}}}\}$ is called the nodal basis of the space $V_{h_{\mathbf{l}}}$ and $I_{h_{\mathbf{l}}}$ the nodal basis index set. Each function $v_{h_{\mathbf{l}}} \in V_{h_{\mathbf{l}}}$ can be represented in the basis of $V_{h_{\mathbf{l}}}$ as follows:

$$v_{h_{\mathbf{l}}} = \sum_{\mathbf{j} \in I_{h_{\mathbf{l}}}} \alpha_{\mathbf{l},\mathbf{j}} W_{h_{\mathbf{l}};\mathbf{j}}, \tag{3.19}$$

where $\alpha_{\mathbf{l},\mathbf{j}}$ are the coefficients of $v_{h_\mathbf{l}}$ in the nodal basis which are the nodal values of the function $v_{h_\mathbf{l}}$ (because the basis functions are multi-linear). We introduce the space of d-dimensional piecewise linear functions with respect to the Cartesian grid:

$$V_{h_n}^{(\infty)} = \text{span}\{W_{h_\mathbf{n};\mathbf{j}} \mid \mathbf{j} \in \mathbb{N}^d \mid \mathbf{0} \leq \mathbf{j} \leq h_\mathbf{n}^{-1}\}. \tag{3.20}$$

Eventually, for u a smooth function, we introduce the linear interpolation operators in nodal basis associated to these spaces defined by:

$$\mathcal{I}_{V_{h_\mathbf{l}}} u = \sum_{\mathbf{j} \in I_{h_\mathbf{l}}} u(\mathbf{j}h_\mathbf{l}) W_{h_\mathbf{l};\mathbf{j}}, \quad \mathcal{I}_{V_{h_n}^{(\infty)}} u = \sum_{\mathbf{j} \in I_{h_n}} u(\mathbf{j}h_n) W_{h_n;\mathbf{j}}. \tag{3.21}$$

3.2.1. Sparse grid combination technique. The sparse grid combination technique [32, 31, 11] is a method of interpolation using evaluations of the function on the nodes of component grids. The sparse grid interpolant is obtained by a linear combination of partial representations of the function on the component grids. Let u be a function and $u_{h_\mathbf{l}}$ an approximation of this function in the space $V_{h_\mathbf{l}}$ (*e.g.* $\mathcal{I}_{V_{h_\mathbf{l}}} u$), then a sparse grid reconstruction is defined by linear combination of the contributions $u_{h_\mathbf{l}}$ of each component grid:

$$u_{h_n} := \sum_{\mathbf{l} \in \mathbb{L}_n} c_\mathbf{l} u_{h_\mathbf{l}}, \quad \text{where } c_\mathbf{l} = (-1)^i \binom{d-1}{i} \text{ if } \mathbf{l} \in \mathbb{L}_{n,i}. \tag{3.22}$$

Under some assumptions on the component grid approximations [31], the error between the sparse grid reconstruction and the solution scales as :

$$\|u_{h_n} - u\|_\infty = O\left(h_n^2 |\log h_n|^{d-1}\right). \tag{3.23}$$

The combination technique is remarkable for the reduction of the number of interpolation nodes:

$$|\Omega_{h_n}^{(\infty)}| = O\left(h_n^{-d}\right), \quad \sum_{\mathbf{l} \in \mathbb{L}_n} |\Omega_{h_\mathbf{l}}| = O\left(h_n^{-1} |\log h_n|^{d-1}\right), \tag{3.24}$$

while achieving accuracy comparable to the standard interpolation (with a negligible multiplicative term $|\log h_n|^{d-1}$). Indeed, for standard interpolation on the Cartesian grid with basis functions of degree one, *i.e.* hat functions, the interpolation error scales as $O\left(h_n^2\right)$.

Remark 3.4. The sparse grid reconstruction of a non-negative function is not non-negative.

3.3. Derivation of the Semi-Implicit Sparse-PIC (SISPIC-sg) scheme. Let us now detail how the implicit contribution of the electric field can be obtained by solving a linear system. The current density can be recast into two components, designated as the explicit and the implicit (with $\sim$ notation) contributions:

$$\mathbf{J}_h^{k+\frac{1}{2}}(\mathbf{x}) = \underbrace{\mathbf{J}_h^k(\mathbf{x})}_{\text{explicit}} + \underbrace{\tilde{\mathbf{J}}_h^{k+1}(\mathbf{x})}_{\text{implicit}}, \tag{3.25}$$

which are defined by:

$$\mathbf{J}_h^k(\mathbf{x}) = \frac{1}{h^d} \sum_s \sum_{p=1}^{N_s} q_p \mathbf{v}_p^k W_h\left(\mathbf{x} - \mathbf{x}_p^{k+\frac{1}{2}}\right), \tag{3.26}$$

$$\tilde{\mathbf{J}}_h^{k+1}(\mathbf{x}) = \frac{\Delta t}{h^d} \sum_s \sum_{p=1}^{N_s} \frac{q_p^2}{4m_p} \left[\mathbf{E}_h^{k+1}\left(\mathbf{x}_p^{k+\frac{1}{2}}\right) + \mathbf{E}_h^k\left(\mathbf{x}_p^{k+\frac{1}{2}}\right)\right] W_h\left(\mathbf{x} - \mathbf{x}_p^{k+\frac{1}{2}}\right). \tag{3.27}$$

Let us now explicit the spatial discretization, *i.e.* define the notation h. In this article, we introduce three distinct schemes. They vary depending on the discretization of each of the explicit and implicit components of the current density. Specifically, if both components are discretized on the Cartesian grid, as defined in equation (3.14), the scheme is termed SISPIC-std. If the two components are discretized on the set of component grids, as defined in equation (3.10), the scheme is referred to as ECSPIC. Finally, if they are discretized on different spatial meshes, the resulting scheme is named SISPIC-sg. We first introduce the latter, while the other two schemes will be presented in section 4. The choice of this approach is motivated by a tradeoff between the computational advantages offered by component grid discretizations (better statistical resolution and fewer grid nodes) and the stability resulting from discretizing the implicit component on a single grid. This latter reason is discussed in more detail in section 4.2. Within the SISPIC-sg scheme, the explicit part of the current density is discretized on the set of component grids and combined onto the Cartesian grid with the combination technique; the implicit component is discretized directly on the Cartesian grid. All the details of the spatial discretization are given in the appendix A.1. The explicit contribution of the current is defined on the Cartesian grid by:

$$\mathbf{J}^k_{h_n;\mathbf{j}^y} = \sum_{\mathbf{l}\in\mathbb{L}_n} c_{\mathbf{l}} \left(\mathcal{I}_{V_{h_{\mathbf{l}}}} \mathbf{J}^k_{h_{\mathbf{l}}}\right)(\mathbf{j}^y), \tag{3.28}$$

where the notation $\mathbf{j}^y$, corresponding to the Cartesian grid index $\mathbf{j} \in I_{h_n}$, is introduced in the appendix A.1 and refers to the grid nodes on the staggered Yee mesh. An explicit contribution of the current is defined independently on each component grid by:

$$\mathbf{J}^k_{h_{\mathbf{l}};\mathbf{j}^y} = h_{\mathbf{l}}^{-1} \sum_s \sum_{p=1}^{N_s} q_p \mathbf{v}^k_p W_{h_{\mathbf{l}};p\mathbf{j}^y}, \tag{3.29}$$

where the following shortcut notation has been introduced:

$$W_{h_{\mathbf{l}};p\mathbf{j}^y} := W_{h_{\mathbf{l}}}\left(\mathbf{j}^y h_{\mathbf{l}} - \mathbf{x}^{k+\frac{1}{2}}_p\right). \tag{3.30}$$

$h_{\mathbf{l}}^{-1}$ is the volume of the component grid. The implicit contribution of the current is discretized on the Cartesian grid and defined for each Cartesian grid index $\mathbf{j} \in I_{h_n}$, by:

$$\tilde{\mathbf{J}}^{k+1}_{h_n;\mathbf{j}^y} = \frac{\Delta t}{h_n^d} \sum_s \sum_{p=1}^{N_s} \frac{q_p^2}{4m_p} \left[\mathbf{E}^{k+1}_{h_n}\left(\mathbf{x}^{k+\frac{1}{2}}_p\right) + \mathbf{E}^k_{h_n}\left(\mathbf{x}^{k+\frac{1}{2}}_p\right)\right] W_{h_n;p\mathbf{j}^y}. \tag{3.31}$$

Let us recall that the electric field is interpolated at the particle positions according to:

$$\mathbf{E}^{k+1}_{h_n}\left(\mathbf{x}^{k+\frac{1}{2}}_p\right) = \sum_{\mathbf{j}\in I_{h_n}} \mathbf{E}^{k+1}_{h_n;\mathbf{j}^y} W_{h_n;p\mathbf{j}^y}, \qquad \mathbf{E}^{k+1}_{h_n;\mathbf{j}^y} = -\boldsymbol{\nabla}_{h_n} \Phi^{k+1}_{h_n;\mathbf{j}}, \tag{3.32}$$

where the discrete gradient operator notation is introduced in appendix A.1. Introducing the following notation:

$$W_{h_n;\mathbf{j}\mathbf{j}'} = W_{h_n;\mathbf{j}'\mathbf{j}} := W_{h_n}(\mathbf{j}' h_n - \mathbf{j} h_n), \tag{3.33}$$

and decomposing the electric potential onto the basis of $V^{(\infty)}_{h_n}$:

$$\Phi^{k+1}_{h_n;\mathbf{j}} = \sum_{\mathbf{j}'\in I_{h_n}} \Phi^{k+1}_{h_n;\mathbf{j}'} W_{h_n;\mathbf{j}\mathbf{j}'}, \tag{3.34}$$

the implicit contribution of the current can be recast into:

$$\tilde{\mathbf{J}}^{k+1}_{h_n;\mathbf{j}^y} = -\frac{\Delta t}{h_n^d}\sum_s\sum_{p=1}^{N_s}\frac{q_p^2}{4m_p}\sum_{\tilde{\mathbf{j}},\mathbf{j}'\in I_{h_n}}\left(\Phi^{k+1}_{h_n;\mathbf{j}'}+\Phi^{k}_{h_n;\mathbf{j}'}\right)\boldsymbol{\nabla}_{h_n}W_{h_n;\tilde{\mathbf{j}}\mathbf{j}'}W_{h_n;p\tilde{\mathbf{j}}^y}W_{h_n;p\mathbf{j}^y}. \tag{3.35}$$

Let us now introduce the discretization of the (div)Ampere equation on the Cartesian grid. From equation (3.34) and applying the discrete Laplacian operator, one gets:

$$\sum_{\mathbf{j}'\in I_{h_n}}\left(\Phi^{k+1}_{h_n;\mathbf{j}'}-\Phi^{k}_{h_n;\mathbf{j}'}\right)\Delta_{h_n}W_{h_n;\mathbf{j}\mathbf{j}'} = \frac{\Delta t}{\varepsilon_0}\boldsymbol{\nabla}_{h_n}\cdot\mathbf{J}^{k+\frac{1}{2}}_{h_n;\mathbf{j}^y}, \tag{3.36}$$

where $\mathbf{J}^{k+1}_{h_n;\mathbf{j}^y}$ is defined from equations (3.25), (3.28) and (3.35).

The contribution of the electric potential at time $k+1$ can thus be obtained by solving a linear system. The matrix corresponding to the linear system is called the stiffness matrix, because it discretizes the Laplacian operator; it is denoted $\mathbb{S}_{h_n}$ and is defined by:

$$\mathbb{S}_{h_n} = \mathbb{S}^{(1)}_{h_n}+\mathbb{S}^{(2)}_{h_n}, \quad \text{where } \forall(\mathbf{j},\mathbf{j}')\in I_{h_n}\times I_{h_n} \tag{3.37}$$

$$(\mathbb{S}^{(1)}_{h_n})_{\mathbf{j}\mathbf{j}'} = \Delta_{h_n}W_{h_n;\mathbf{j}\mathbf{j}'}, \tag{3.38}$$

$$(\mathbb{S}^{(2)}_{h_n})_{\mathbf{j}\mathbf{j}'} = \sum_s\sum_{p=1}^{N_s}\beta_p\sum_{\tilde{\mathbf{j}}\in I_{h_n}}W_{h_n;p\tilde{\mathbf{j}}^y}\boldsymbol{\nabla}_{h_n}W_{h_n;\tilde{\mathbf{j}}\mathbf{j}'}\cdot\boldsymbol{\nabla}_{h_n}W_{h_n;p\mathbf{j}^y}, \tag{3.39}$$

and $\beta_p = q_p^2\Delta t^2/(4\varepsilon_0 m_p)$. Two stiffness matrices have been introduced: the first matrix corresponds to the discretization of the Laplacian operator; the second matrix is the linear term of the particle response to the electric potential. Note that the stiffness matrix depends on the particle positions, and thus is different from one iteration to another. Therefore it has to be computed at each iteration.

Remark 3.5. The SISPIC-sg scheme is second order accurate in time.

3.4. Comparison with the existing semi-implicit methods.

3.4.1. Direct implicit. The motivation of this section is to situate the present method among the state-of-the-art of the semi-implicit methods. In the direct implicit method [34], the particles are advanced to an intermediate position and velocity using the known electric field in a PREPUSH step:

$$\begin{cases}\tilde{\mathbf{v}}_p^{k+\frac{1}{2}} = \mathbf{v}_p^{k-\frac{1}{2}}+\Delta t\dfrac{q_p}{m_p}\mathbf{E}_h^k\left(\mathbf{x}_p^k\right)\\ \tilde{\mathbf{x}}_p^{k+1} = \mathbf{x}_p^k+\Delta t\tilde{\mathbf{v}}_p^{k+\frac{1}{2}}\end{cases}, \quad k\in\mathbb{N}. \tag{3.40}$$

Similarly to our approach, the sources in the fields equations are decomposed into an explicit contribution and an implicit contribution:

$$\mathbf{J}_h^{k+\frac{1}{2}}(\mathbf{x}) = \underbrace{\mathbf{J}_h^k(\mathbf{x})}_{\text{explicit}}+\underbrace{\tilde{\mathbf{J}}_h^{k+1}(\mathbf{x})}_{\text{implicit}}. \tag{3.41}$$

The implicit contribution is different from our approach in that the linearization of the shape function introduces two matrices, which are named implicit susceptibility matrices:

$$\tilde{\mathbf{J}}_h^{k+1}(\mathbf{x}) = \boldsymbol{\chi}_h^k(\mathbf{x})\cdot\mathbf{E}_h^{k+1}(\tilde{\mathbf{x}}_p^{k+1})-\Delta t\boldsymbol{\nabla}\times\boldsymbol{\xi}_h^k\cdot\mathbf{E}_h^{k+1}(\tilde{\mathbf{x}}_p^{k+1}). \tag{3.42}$$

The implicit susceptibility matrices are approximated by:

$$\boldsymbol{\chi}_h^k(\mathbf{x}) \approx \sum_s \frac{\Delta t}{4}\frac{q_s}{m_s}\rho_{s,h}^k(\mathbf{x})\boldsymbol{\alpha}_s^k, \quad \boldsymbol{\xi}_h^k(\mathbf{x}) \approx \sum_s \frac{\Delta t}{8}\frac{q_s}{m_s}\mathbf{J}_{s,h}^k(\mathbf{x})\boldsymbol{\alpha}_s^k, \tag{3.43}$$

where $\boldsymbol{\alpha}_s^k$ is the rotation matrix [34, 47, 39]. It is defined from the magnetic field and falls down to the identity matrix in the electrostatic regime. In the electrostatic version, the second implicit susceptibility matrix is zero. These matrices can be read as a discretization of the continuous quantity $d\mathbf{J}/dt$. Within the SISPIC-sg scheme, a different discretization of this quantity than the direct implicit method is proposed: the first term in equation (3.42) corresponding to:

$$\boldsymbol{\chi}_{h;\mathbf{j}}^k \cdot \mathbf{E}^{k+1}(\mathbf{x}_p^k) \approx \sum_{\tilde{\mathbf{j}}\in I_h} \mathbf{E}_{h;\tilde{\mathbf{j}}}^{k+\frac{1}{2}} \sum_s \frac{q_s}{4m_s}\frac{\Delta t}{h^d}\sum_p W_{h;p\mathbf{j}^y} W_{h;p\tilde{\mathbf{j}}^y}, \tag{3.44}$$

and the second term being neglected as a $O\left(\Delta t^2\right)$ correction. After the advanced field has been computed in the direct implicit method, the particles are advanced to the next time step in a FINALPUSH step:

$$\begin{cases} \mathbf{x}_p^{k+1} = \tilde{\mathbf{x}}_p^{k+1} + \dfrac{\Delta t^2}{2}\dfrac{q_p}{m_p}\mathbf{E}_h^{k+1}(\tilde{\mathbf{x}}_p^{k+1}) \\ \mathbf{v}_p^{k+\frac{1}{2}} = \tilde{\mathbf{v}}_p^{k+\frac{1}{2}} + \dfrac{\Delta t}{2}\dfrac{q_p}{m_p}\mathbf{E}_h^{k+1}\left(\tilde{\mathbf{x}}_p^{k+1}\right) \end{cases}, \quad k \in \mathbb{N}. \tag{3.45}$$

3.4.2. Moment implicit. In the moment implicit method [41, 9], the non-linearity coupling the fields and the particles equations is overcome by considering instead the resolution of a set of coupled fluid moment and field equations. Indeed, thanks to a representation of the implicit sources of the field equations using the moment equations instead of the particle equations directly, the non linear coupling is approximated so that, once the field equations are solved within this approximation, the rest of the steps can be completed directly without iterations. Specifically, in [47], the particle equations are as follows:

$$\begin{cases} \mathbf{x}_p^{k+1} = \mathbf{x}_p^k + \dfrac{\Delta t}{2}\left(\mathbf{v}_p^{k+1} + \mathbf{v}_p^k\right) \\ \mathbf{v}_p^{k+1} = \mathbf{v}_p^k + \Delta t\dfrac{q_p}{m_p}\mathbf{E}_h^{k+\frac{1}{2}}\left[\dfrac{1}{2}\left(\mathbf{x}_p^{k+1} + \mathbf{x}_p^k\right)\right] \end{cases}, \quad k \in \mathbb{N}. \tag{3.46}$$

and the implicit current density, which is the source of the Ampere equation, is approximated by:

$$\mathbf{J}_h^{k+\frac{1}{2}}(\mathbf{x}) \approx \sum_s \boldsymbol{\alpha}_s^k \mathbf{J}_{s,h}^k(\mathbf{x}) - \frac{\Delta t}{2}\boldsymbol{\mu}_s^k(\mathbf{x}) \cdot \boldsymbol{\alpha}_s^k \mathbf{E}_h^{k+\frac{1}{2}}(\mathbf{x}) - \frac{\Delta t}{2}\nabla \cdot \boldsymbol{\Pi}_s^k(\mathbf{x}), \tag{3.47}$$

where $\boldsymbol{\alpha}_s^k$ is the rotation matrix, $\boldsymbol{\Pi}_s^k$ is the rotationed pressure matrix and $\boldsymbol{\mu}_s^k$ is defined as the effective dielectric matrix which express the response of the electric field to the plasma current:

$$\boldsymbol{\Pi}_s^k(\mathbf{x}) = \frac{1}{h^d}\sum_{p=1}^{N_s} q_p \boldsymbol{\alpha}_s^k \mathbf{v}_p^k \boldsymbol{\alpha}_s^k \mathbf{v}_p^k W_h(\mathbf{x} - \mathbf{x}_p^k), \quad \boldsymbol{\mu}_s^k(\mathbf{x}) = \frac{-q_s\rho_s^k(\mathbf{x})}{m_s}\boldsymbol{\alpha}_s^k. \tag{3.48}$$

The first term in the equation (3.47) corresponds to the explicit contribution of the current density and the two last terms correspond to the implicit one, for which the SISPIC-sg and the direct implicit methods provide a different approximation.

3.4.3. ECSIM. The SISPIC-sg method introduced in this paper is inspired by the ECSIM scheme [39]. The non-linear coupling between the field and the particle equations is approximated directly in the particle equations similarly for both methods. Let us emphasize the differences between the two methods:

- The first difference between the traditional ECSIM scheme [39] (as well as its extensions [16, 12]) and our approach is the electrostatic regime. As a result, the formulation of the problem is different between the two approaches and, within the SISPIC-sg method, the Ampere equation is substituted by the divergence of the Ampere equation. The resulting electrostatic stiffness matrix contains terms depending on the product of the basis functions discrete gradients (with larger supports than the basis functions) instead of products of the basis functions theirselves within the ECSIM scheme (for which the matrix is more similar to a mass matrix). Therefore the stiffness matrix of the SISPIC-sg scheme has more non-zeros entries (21 non-zero terms per row) than the mass matrices of the ECSIM scheme (9 non-zero terms per row). In addition, there is no self-consistent magnetic field in our approach and therefore the rotation matrix used in the traditional ECSIM scheme [39] does not have to be computed. It results in an unique stiffness matrix to compute at each time step, instead of $3v$ mass matrices (where v is the dimension of the velocity domain) for the ECSIM scheme. In addition, the unknown for the electrostatic approach is scalar, so that the size of the linear system is reduced by six in comparison to the ECSIM scheme (Φ versus $E_x, E_y, E_z, B_x, B_y, Bz$). Note also that, contrary to the SISPIC-sg approach, the electrostatic condition $\nabla \times \mathbf{E}$ is not verified in the ECSIM scheme.
- The second major difference between the two approaches is the embedding of the sparse grid combination technique. Indeed, within the SISPIC-sg scheme, the explicit contribution of the current density is accumulated onto each component grid, and eventually combined onto the Cartesian grid. The motivation for this choice of discretization is related to complexity issues and explained in the following section.
- Contrary to the ECSIM scheme and caused by the sparse grid reconstruction, the SISPIC-sg scheme does not conserve exactly the discrete total energy of the system. This is a consequence of the choice of the interpolation basis functions which are different from the shape functions used for the current density accumulation in the SISPIC-sg scheme. Indeed, the supports of the functions are either based on the Cartesian grid (for interpolation) or on the component grids (for current accumulation).

3.5. Discussion on complexity. The size of the linear system, *i.e.* the size of the stiffness matrix, is related to the Cartesian grid discretization and grows exponentially with the dimension of the problem; specifically, the number of nodes of the linear system scales as $O\left(h_n^{-d} \times h_n^{-d}\right)$. Nonetheless, thanks to the locality of the shape functions W_{h_n}, the stiffness matrix has many zero entries (21 non-zero terms per row), the total number of non-zero terms of the stiffness matrix being:

$$non\ \ zero\ terms = 21 * 2^{dn}. \tag{3.49}$$

It is widely known that PIC schemes contain a significant weakness: the statistical error originating from the sampling of the distribution function by a limited number of numerical particles. This numerical noise decreases slowly with the increase of the average number of particles per cell (in $O\left(1/\sqrt{P_c}\right)$, where $P_c = N h_n^d$ is the mean number of particles per cell. Within the SISPIC-sg method, the explicit contribution of the current density is accumulated onto the mesh thanks to the sparse grid combination technique.

PROPOSITION 3.6. *Assuming enough smoothness on the solution $\boldsymbol{J}$, the local error between the explicit contribution of the current density and the solution can be recast into a grid-based error and a particle sampling error (noise):*

$$\boldsymbol{J}^k_{h_n} - \boldsymbol{J} = \underbrace{\mathrm{Bias}(\boldsymbol{J}^k_{h_n})}_{\text{grid-based error}} + \underbrace{\mathcal{V}(\boldsymbol{J}^k_{h_n})}_{\text{particle sampling error}}, \tag{3.50}$$

where

$$\left\|\mathrm{Bias}(\boldsymbol{J}^k_{h_n})\right\|_\infty \leq O\left(h_n^2 |\log h_n|^{d-1}\right), \quad \left\|\mathbb{V}\left(\mathcal{V}(\boldsymbol{J}^k_{h_n})\right)^{\frac{1}{2}}\right\|_\infty \leq O\left(\frac{|\log h_n|^{d-1}}{(N h_n)^{\frac{1}{2}}}\right) \tag{3.51}$$

Proof. The proof is similar to the one provided for the charge density in [26]. ∎

These estimations shall be compared to the ones obtained for a standard accumulation onto the Cartesian grid (without sparse grid technique):

$$\left\|\mathrm{Bias}(\mathbf{J}^{k,std}_{h_n})\right\|_\infty = O\left(h_n^2\right), \quad \left\|\mathbb{V}\left(\mathcal{V}(\mathbf{J}^{k,std}_{h_n})\right)^{\frac{1}{2}}\right\|_\infty = O\left(\left(\frac{1}{N h_n^d}\right)^{\frac{1}{2}}\right). \tag{3.52}$$

Specifically, the combination technique provides an accurate representation of the current density approximated on a variety of grids with coarse discretizations: the component grids. One crucial feature of the method is the large size of the component grid cells in comparison to those of the Cartesian grid, resulting in a significant increase of the number of particles per cell. This entails an improvement of the statistical resolution (reduced particle sampling error), without increasing the overall number of particles. Practically, the total number of particles in order to achieve a given statistical resolution is reduced in comparison to PIC schemes discretized on standard mesh. As a result, the gain in term of memory footprint of the method compared to standard implicit methods is manifest, particularity for three dimensional simulations with refined mesh [24, 25].

4. Extension to exact energy conservation: SISPIC-std, ECSPIC schemes. It has already been noted that the SISPIC-sg scheme does not exactly conserve the discrete total energy of the system. Nonetheless, it can be adapted to a specific configuration in which the total energy is conserved exactly. Specifically, if the explicit contribution of the current density is directly accumulated onto the Cartesian grid (i.e., both the explicit and implicit contributions of the current density are discretized on the Cartesian grid), then the scheme conserves total energy exactly. However, it no longer benefits from the sparse grid discretization. This scheme can thus be regarded as an electrostatic version of the ECSIM scheme and is termed SISPIC-std.

The goal of this section is to derive a sparse grid scheme conserving exactly the total energy. Let us first introduce the discrete total energy of the system at time k defined from the kinetic energy of the particles and the field energy:

$$\mathcal{E}^k_{\mathcal{T}} := \underbrace{\mathcal{E}^k_{\mathcal{K}}}_{\text{kinetic energy}} + \underbrace{\mathcal{E}^k_{\mathcal{F}}}_{\text{field energy}}, \tag{4.1}$$

which are defined by:

$$\mathcal{E}^k_{\mathcal{K}} := \frac{1}{2}\sum_s \sum_{p=1}^{N_s} m_p \left(\mathbf{v}^k_p\right)^2, \quad \mathcal{E}^k_{\mathcal{F}} := \frac{\epsilon_0}{2}\sum_{\mathbf{j}\in I_{h_n}} \left(\mathbf{E}^k_{h_n;\mathbf{j}}\right)^2. \tag{4.2}$$

In order to have exact conservation of the total energy, the method used to accumulate the current density and the method used to interpolate the electric field at the particle positions shall be, in a way, similar. Therefore, in our approach, the electric field shall be interpolated at the particle positions with the sparse grid combination technique.

4.1. Derivation of the Energy-Conserving semi-implicit Sparse-PIC (ECSPIC) scheme. The method uses the same particle mover as the SISPIC-sg scheme, in which the particle positions are updated with a known velocity:

$$\begin{cases} \mathbf{x}_p^{k+\frac{1}{2}} = \mathbf{x}_p^{k-\frac{1}{2}} + \Delta t \mathbf{v}_p^k \\ \mathbf{v}_p^{k+1} = \mathbf{v}_p^k + \Delta t \dfrac{q_p}{m_p} \mathbf{E}_{h_n}^{k+\frac{1}{2}} \left(\mathbf{x}_p^{k+\frac{1}{2}} \right) \end{cases}, \quad k \in \mathbb{N}. \tag{4.3}$$

In the above, the electric field reconstruction evaluated at the particle positions is constructed, according to the combination technique, from the electric field contributions of all component grids, averaged between time k and $k+1$:

$$\mathbf{E}_{h_n}^{k+\frac{1}{2}} \left(\mathbf{x}_p^{k+\frac{1}{2}} \right) = \sum_{\mathbf{l} \in \mathbb{L}_n} c_{\mathbf{l}} \mathcal{I}_{V_{h_{\mathbf{l}}}} \left(\mathbf{E}_{h_{\mathbf{l}};\mathbf{j}^y}^{k+\frac{1}{2}} \right) \left(\mathbf{x}_p^{k+\frac{1}{2}} \right), \tag{4.4}$$

$$\text{where } \mathbf{E}_{h_{\mathbf{l}};\mathbf{j}^y}^{k+\frac{1}{2}} = \frac{1}{2} \left(\mathbf{E}_{h_{\mathbf{l}};\mathbf{j}^y}^{k+1} + \mathbf{E}_{h_{\mathbf{l}};\mathbf{j}^y}^{k} \right). \tag{4.5}$$

We recall that the notation $\mathcal{I}_{V_{h_{\mathbf{l}}}}$, introduced in equation (3.21), stands for the interpolation onto the space associated to the component grid of discretization $h_{\mathbf{l}}$ and spanned by basis functions with support depending on $h_{\mathbf{l}}$. The electric field, derived from the electric potential according to equation (3.32), is obtained from the implicit current density using the divergence of the Ampere equation. This equation is discretized on each component grid according to the relation:

$$\sum_{\mathbf{j}' \in I_{h_{\mathbf{l}}}} \left(\Phi_{h_{\mathbf{l}};\mathbf{j}'}^{k+1} - \Phi_{h_{\mathbf{l}};\mathbf{j}'}^{k} \right) \Delta_{h_{\mathbf{l}}} W_{h_{\mathbf{l}};\mathbf{j}\mathbf{j}'} = \frac{\Delta t}{\epsilon_0} \boldsymbol{\nabla}_{h_{\mathbf{l}}} \cdot \mathbf{J}_{h_{\mathbf{l}};\mathbf{j}^y}^{k+\frac{1}{2}}. \tag{4.6}$$

The current density is decomposed into an explicit contribution and an implicit contribution, both defined on each component grid:

$$\mathbf{J}_{h_{\mathbf{l}};\mathbf{j}^y}^{k+\frac{1}{2}} = \underbrace{\mathbf{J}_{h_{\mathbf{l}};\mathbf{j}^y}^{k}}_{\text{explicit}} + \underbrace{\tilde{\mathbf{J}}_{h_{\mathbf{l}};\mathbf{j}^y}^{k+1}}_{\text{implicit}}, \quad \forall \mathbf{j} \in I_{h_{\mathbf{l}}}, \tag{4.7}$$

by the relations:

$$\mathbf{J}_{h_{\mathbf{l}};\mathbf{j}^y}^{k} = \sum_s h_{\mathbf{l}}^{-1} \sum_{p=1}^{N_s} q_p \mathbf{v}_p^k W_{h_{\mathbf{l}};p\mathbf{j}^y}, \tag{4.8}$$

$$\tilde{\mathbf{J}}_{h_{\mathbf{l}};\mathbf{j}^y}^{k+1}(\mathbf{x}) = \sum_s h_{\mathbf{l}}^{-1} \sum_{p=1}^{N_s} \frac{q_p^2 \Delta t}{2 m_p} \mathbf{E}_{h_n}^{k+\frac{1}{2}} \left(\mathbf{x}_p^{k+\frac{1}{2}} \right) W_{h_{\mathbf{l}};p\mathbf{j}^y}. \tag{4.9}$$

Introducing the equation (4.4) in the last equation (4.9), using the decomposition introduced in equation (3.34) and substituting them into the divergence of Ampere equation (4.6), the electric potential at time $k+1$ can be obtained by solving a linear system. The matrix of the linear system, denoted by $\mathbb{S}_{\mathbb{L}_n}$, is named the global stiffness matrix and constructed by

assembling local stiffness matrices. The local stiffness matrices are defined for couples of component indices $(\mathbf{l}, \tilde{\mathbf{l}}) \in \mathbb{L}_n^2$ by:

$$\mathbb{S}^{(1)}_{h_{\mathbf{l}}} := \left(s^{(1)}_{\mathbf{j},\mathbf{j}'}\right)_{(\mathbf{j},\mathbf{j}')\in I^2_{h_{\mathbf{l}}}}, \qquad \text{where } s^{(1)}_{\mathbf{j},\mathbf{j}'} = \Delta_{h_{\mathbf{l}}} W_{h_{\mathbf{l}};\mathbf{j}\mathbf{j}'}$$

$$\mathbb{S}^{(2)}_{h_{\mathbf{l}},h_{\tilde{\mathbf{l}}}} := \left(s^{(2)}_{\mathbf{j},\mathbf{j}'}\right)_{(\mathbf{j},\mathbf{j}')\in I_{h_{\mathbf{l}}}\times I_{h_{\tilde{\mathbf{l}}}}}, \text{ where } s^{(2)}_{\mathbf{j},\mathbf{j}'} = \sum_s \sum_{p=1}^{N_s} \beta_p \sum_{\tilde{\mathbf{j}}\in I_{h_{\tilde{\mathbf{l}}}}} W_{h_{\tilde{\mathbf{l}}};p\tilde{\mathbf{j}}^y} \boldsymbol{\nabla}_{h_{\tilde{\mathbf{l}}}} W_{h_{\tilde{\mathbf{l}}};\tilde{\mathbf{j}}\mathbf{j}'} \cdot \boldsymbol{\nabla}_{h_{\mathbf{l}}} W_{h_{\mathbf{l}};p\mathbf{j}^y},$$

and $\beta_p = q_p^2 \Delta t^2/(4\varepsilon_0 m_p)$. The first local matrix is a discretization of the Laplacian operator on the component grid $\Omega_{h_{\mathbf{l}}}$ and shall be named the local Laplacian matrix. The local Laplacian matrices are square, symmetric and of size depending on the number of nodes from the component grid. The second local matrix, which shall be referred to as the local stiffness matrix, translates the energy exchange between the particles and the field. Specifically, the local stiffness matrix represents the effect on the electric potential (computed on the grid $\Omega_{h_{\mathbf{l}}}$) of the electric potential (computed on the grid $\Omega_{h_{\tilde{\mathbf{l}}}}$) response to the particles. The local stiffness matrices are rectangular and of size depending on the number of nodes from each component grid in the couple. Let us recall $|\mathbb{L}_n| := Card(\mathbb{L}_n)$ the number of component grids. There are $|\mathbb{L}_n|^2$ local stiffness matrices but note that only a few more than a half of them shall be computed thanks to the symmetry. Let us numerate all the component grid levels $\mathbb{L}_n = (\mathbf{l}_1, ..., \mathbf{l}_{|\mathbb{L}_n|})$ and let $\Phi^k_{\mathbb{L}_n}$ ($\boldsymbol{\nabla}_{h_{\mathbf{l}}} \cdot \mathbf{J}^k_{\mathbb{L}_n}$ resp.) be a global vector corresponding to the electric potential (divergence of the current density resp.) approximations on all the component grids at time k. From these local matrices, we construct by blocks two global matrices, containing all the component grid contributions:

$$\mathbb{S}^{(1)}_{\mathbb{L}_n} = \begin{pmatrix} \mathbb{S}^{(1)}_{h_{\mathbf{l}_1}} & 0 & \cdots & 0 \\ 0 & \mathbb{S}^{(1)}_{h_{\mathbf{l}_2}} & \cdots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \mathbb{S}^{(1)}_{h_{\mathbf{l}_{|\mathbb{L}_n|}}} \end{pmatrix}. \tag{4.10}$$

(4.11)

$$\mathbb{S}^{(2)}_{\mathbb{L}_n} = \begin{pmatrix} c_{\mathbf{l}_1} h^{-1}_{\mathbf{l}_1} \mathbb{S}^{(2)}_{h_{\mathbf{l}_1},h_{\mathbf{l}_1}} & c_{\mathbf{l}_2} h^{-1}_{\mathbf{l}_1} \mathbb{S}^{(2)}_{h_{\mathbf{l}_1},h_{\mathbf{l}_2}} & \cdots & c_{\mathbf{l}_{|\mathbb{L}_n|}} h^{-1}_{\mathbf{l}_1} \mathbb{S}^{(2)}_{h_{\mathbf{l}_1},h_{\mathbf{l}_{|\mathbb{L}_n|}}} \\ c_{\mathbf{l}_1} h^{-1}_{\mathbf{l}_2} \mathbb{S}^{(2)}_{h_{\mathbf{l}_2},h_{\mathbf{l}_1}} & c_{\mathbf{l}_2} h^{-1}_{\mathbf{l}_2} \mathbb{S}^{(2)}_{h_{\mathbf{l}_2},h_{\mathbf{l}_2}} & \cdots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ c_{\mathbf{l}_1} h^{-1}_{\mathbf{l}_{|\mathbb{L}_n|}} \mathbb{S}^{(2)}_{h_{\mathbf{l}_{|\mathbb{L}_n|}},h_{\mathbf{l}_1}} & \cdots & c_{\mathbf{l}_{|\mathbb{L}_n|-1}} h^{-1}_{\mathbf{l}_{|\mathbb{L}_n|}} \mathbb{S}^{(2)}_{h_{\mathbf{l}_{|\mathbb{L}_n|}},h_{\mathbf{l}_{|\mathbb{L}_n|-1}}} & c_{\mathbf{l}_{|\mathbb{L}_n|}} h^{-1}_{\mathbf{l}_{|\mathbb{L}_n|}} \mathbb{S}^{(2)}_{h_{\mathbf{l}_{|\mathbb{L}_n|}},h_{\mathbf{l}_{|\mathbb{L}_n|}}} \end{pmatrix}.$$

Two matrices are then constructed from these matrices:

$$\mathbb{S}_{\mathbb{L}_n} = \mathbb{S}^{(1)}_{\mathbb{L}_n} + \frac{1}{2}\mathbb{S}^{(2)}_{\mathbb{L}_n}, \quad \mathbb{S}^{\star}_{\mathbb{L}_n} = \mathbb{S}^{(1)}_{\mathbb{L}_n} - \frac{1}{2}M^{(2)}_{\mathbb{L}_n}, \tag{4.12}$$

which are square and of order:

$$Order(\mathbb{S}_{\mathbb{L}_n}) = Order(\mathbb{S}^{\star}_{\mathbb{L}_n}) = \sum_{r=0}^{d-1} 2^{n+d-1-r} \binom{n+d-2-r}{d-1}. \tag{4.13}$$

Note that the two global matrices are not symmetric because the global stiffness matrix is not. The system can thus be rewritten as a linear system:

$$\mathbb{S}_{\mathbb{L}_n}\Phi^{k+1}_{\mathbb{L}_n} = \frac{\Delta t}{\epsilon_0}\boldsymbol{\nabla}_{h_\mathbf{l}}\cdot\mathbf{J}^k_{\mathbb{L}_n} + \mathbb{S}^\star_{\mathbb{L}_n}\Phi^k_{\mathbb{L}_n}. \tag{4.14}$$

In order to solve the linear system (4.14) and obtain the updated electric potential at time $k+1$, the local stiffness matrices need to be computed at each time step because of their dependence on the position of the particles. The resulting global stiffness matrix (and global matrix) is then different at each iteration.

Within the ECSPIC scheme, the electric field is decomposed into contributions which are defined separately on the component grids; it is then reconstructed at the particle positions with the sparse grid combination technique. The electric field is never defined on the Cartesian grid so that the field energy definition in (4.2) is not relevant. Let us define the discrete field energy, in the context of sparse grid reconstruction schemes, which is defined by the combination of the electric field energy from all the component grids:

$$\mathcal{E}^k_{\mathcal{F}} := \frac{\epsilon_0}{2}\sum_{\mathbf{l}\in\mathbb{L}_n} c_\mathbf{l} h_\mathbf{l}^{-1}\sum_{\mathbf{j}\in I_{h_\mathbf{l}}}\left(\mathbf{E}^k_{h_\mathbf{l};\mathbf{j}}\right)^2. \tag{4.15}$$

Proposition 4.1 (Exact energy conservation). *The total energy of the system is exactly conserved, i.e.*

$$\mathcal{E}^{k+1}_{\mathcal{T}} = \mathcal{E}^k_{\mathcal{T}}. \tag{4.16}$$

Proof of proposition 4.1. The difference in the particle kinetic energy between two consecutive steps k and $k+1$ is expressed by:

$$\mathcal{E}^{k+1}_{\mathcal{K}} - \mathcal{E}^k_{\mathcal{K}} = \sum_s\sum_{p=1}^{N_s}\frac{1}{2}m_p\left[\left(\mathbf{v}_p^{k+1}\right)^2 - \left(\mathbf{v}_p^k\right)^2\right] \tag{4.17}$$

$$= \Delta t\sum_s\sum_{p=1}^{N_s} q_p\mathbf{v}_p^{k+1/2}\mathbf{E}^{k+1/2}_{h_n}\left(\mathbf{x}_p^{k+1/2}\right) \tag{4.18}$$

$$= \Delta t\sum_s\sum_{p=1}^{N_s} q_p\mathbf{v}_p^{k+1/2}\sum_{\mathbf{l}\in\mathbb{L}_n} c_\mathbf{l}\sum_{\mathbf{j}\in I_{h_\mathbf{l}}}\mathbf{E}^{k+1/2}_{h_\mathbf{l};\mathbf{j}^y}W_{h_\mathbf{l};p\mathbf{j}^y} \tag{4.19}$$

$$= \Delta t\sum_{\mathbf{l}\in\mathbb{L}_n} c_\mathbf{l} h_\mathbf{l}^{-1}\sum_{\mathbf{j}\in I_{h_\mathbf{l}}}\mathbf{E}^{k+1/2}_{h_\mathbf{l};\mathbf{j}^y}\cdot\mathbf{J}^{k+1/2}_{h_\mathbf{l};\mathbf{j}^y}. \tag{4.20}$$

$$= -\Delta t\sum_{\mathbf{l}\in\mathbb{L}_n} c_\mathbf{l} h_\mathbf{l}^{-1}\sum_{\mathbf{j}\in I_{h_\mathbf{l}}}\boldsymbol{\nabla}_{h_\mathbf{l}}\Phi^{k+1/2}_{h_\mathbf{l};\mathbf{j}}\cdot\mathbf{J}^{k+1/2}_{h_\mathbf{l};\mathbf{j}^y}. \tag{4.21}$$

In the above, the particle velocity equation in (3.2) has been used, as well as the definition of the recombined electric field (4.4) and the current density (4.7)-(4.9). The last equation is obtained because the electric field derives from the electric potential. From lemma A.1, a discrete integration by parts on (4.21) yields:

$$\mathcal{E}^{k+1}_{\mathcal{K}} - \mathcal{E}^k_{\mathcal{K}} = \Delta t\sum_{\mathbf{l}\in\mathbb{L}_n} c_\mathbf{l} h_\mathbf{l}^{-1}\sum_{\mathbf{j}\in I_{h_\mathbf{l}}}\Phi^{k+1/2}_{h_\mathbf{l};\mathbf{j}}\boldsymbol{\nabla}_{h_\mathbf{l}}\cdot\mathbf{J}^{k+1/2}_{h_\mathbf{l};\mathbf{j}^y}. \tag{4.22}$$

Then, using the divergence of Ampere equation (4.6), the difference in the kinetic energy is equal to:

$$\mathcal{E}_{\mathcal{K}}^{k+1} - \mathcal{E}_{\mathcal{K}}^{k} = -\epsilon_0 \sum_{\mathbf{l}\in\mathbb{L}_n} c_{\mathbf{l}} h_{\mathbf{l}}^{-1} \sum_{\mathbf{j}\in I_{h_{\mathbf{l}}}} \Phi_{h_{\mathbf{l}};\mathbf{j}}^{k+1/2} \boldsymbol{\nabla}_{h_{\mathbf{l}}} \cdot \left(\mathbf{E}_{h_{\mathbf{l}};\mathbf{j}^y}^{k+1} - \mathbf{E}_{h_{\mathbf{l}};\mathbf{j}^y}^{k} \right). \tag{4.23}$$

Eventually, a discrete integration by parts (see lemma A.1) gives the result:

$$\mathcal{E}_{\mathcal{K}}^{k+1} - \mathcal{E}_{\mathcal{K}}^{k} = \epsilon_0 \sum_{\mathbf{l}\in\mathbb{L}_n} c_{\mathbf{l}} h_{\mathbf{l}}^{-1} \sum_{\mathbf{j}\in I_{h_{\mathbf{l}}}} \boldsymbol{\nabla}_{h_{\mathbf{l}}} \Phi_{h_{\mathbf{l}};\mathbf{j}}^{k+1/2} \cdot \left(\mathbf{E}_{h_{\mathbf{l}};\mathbf{j}^y}^{k+1} - \mathbf{E}_{h_{\mathbf{l}};\mathbf{j}^y}^{k} \right) \tag{4.24}$$

$$= -\epsilon_0 \sum_{\mathbf{l}\in\mathbb{L}_n} c_{\mathbf{l}} h_{\mathbf{l}}^{-1} \sum_{\mathbf{j}\in I_{h_{\mathbf{l}}}} \mathbf{E}_{h_{\mathbf{l}};\mathbf{j}^y}^{k+1/2} \cdot \left(\mathbf{E}_{h_{\mathbf{l}};\mathbf{j}^y}^{k+1} - \mathbf{E}_{h_{\mathbf{l}};\mathbf{j}^y}^{k} \right) \tag{4.25}$$

$$= -\left(\mathcal{E}_{\mathcal{F}}^{k+1} - \mathcal{E}_{\mathcal{F}}^{k} \right). \ \square \tag{4.26}$$

Remark 4.2. The sparse grid discrete field energy, defined in equation (4.15) is not non-negative.

4.2. Discussion about complexity and non-negativity. The major difference with the SISPIC-sg scheme is related to the embedding of sparse grid reconstructions, not only restricted to the computation of the current density explicit contribution as in SISPIC-sg, but used also within the resolution of the linear system. It results in the computation of numerous blocks (local stiffness matrices) to assemble the system matrix. In the SISPIC-sg scheme, an unique stiffness matrix shall be computed. The number of component grids and thus of stiffness matrices to compute for the ECSPIC method depends on the dimension and the grid discretization, thus scaling as $O\left(|\log h_n|^{d-1}\right)$. The determination of a stiffness matrix can be expansive because of the computation of the particle interactions requiring a loop spanning the particle population. Nonetheless, the local stiffness matrices are less expensive to compute for sparse grid reconstructions than for standard discretizations since the number of particles is significantly reduced and, the array of a local stiffness matrix fits more easily in the cache memory of the CPU than the global stiffness matrix for standard grids. As demonstrated in [24], a good cache memory management is a capital feature increasing the performance of these kind of algorithms and enabling an efficient parallelization.

Another significant difference between the two approaches introduced in this paper is the profile of the global stiffness matrix. Indeed, one major advantage of the SISPIC-sg scheme (and ECSIM scheme) is the sparsity of the stiffness matrix. Because of the very localized support of the basis functions (hat function with one grid cell width support), the number of non-zero terms of the stiffness matrix is 21 per row (9 for the ECSIM scheme's mass matrix). We recall that the total number of non-zero terms in the global matrix for the SISPIC-sg method is:

$$non\ zero\ terms = 21 * 2^{dn}. \tag{4.27}$$

Conversely, the matrix profile of the ECSPIC scheme is a bit more complex. The global matrix is constructed by assembling blocks of local matrices, each corresponding to a couple of component grids. For couples of the same component grids (corresponding to the diagonal blocks in the global matrix), the profile of the local stiffness matrix is similar to the standard matrix one with the same sparsity (only 21 non-zero terms per row). On the contrary, for couples of component grids with different discretizations, the profile may not be sparse at

all. *E.g.* consider a couple of grids with levels $\mathbf{l}_1 = (1, n)$ and $\mathbf{l}_2 = (n, 1)$, for all grid nodes $\mathbf{j}_1 \in I_{h_{\mathbf{l}_1}}$ and $\mathbf{j}_2 \in I_{h_{\mathbf{l}_2}}$ the support of the basis functions is not disjoint (see figure 1):

$$Supp(W_{h_{\mathbf{l}_1},\mathbf{j}_1}) \cap Supp(W_{h_{\mathbf{l}_2},\mathbf{j}_2}) \neq \emptyset. \tag{4.28}$$

In that configuration, all the entries in the local stiffness matrix are non-zeros. Nonetheless,

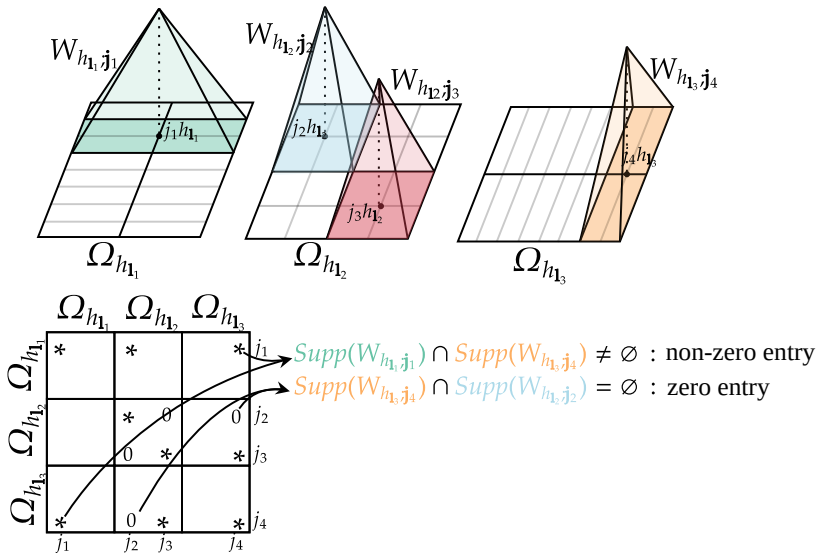

Fig. 1: Support of basis functions for component grids and corresponding non-zero and zero entries in the global stiffness matrix.

thanks to the sparse grid properties, the size of the linear system, *i.e.* the order of the global stiffness matrix, is reduced in comparison to the SISPIC schemes:

$$Order(\mathbb{S}) = \begin{cases} \displaystyle\sum_{r=0}^{d-1} 2^{n+d-1-r} \binom{n+d-2-r}{d-1} & \text{(ECSPIC)}, \\ 2^{dn} & \text{(SISPIC-sg, SISPIC-std, ECSIM)}. \end{cases} \tag{4.29}$$

The scheme ensures an exact energy conservation, *i.e.* the change in the particle energy is equal to the change in the field energy (see proposition 4.1). Usually, since these two quantities are non-negative, this property guarantees some stability for the scheme. Indeed, if the two energy contributions (particle and field) are non-negative, then, by induction, the following bounds hold:

$$\mathcal{E}_{\mathcal{K}}^k < \mathcal{E}_{\mathcal{T}}^0, \quad \mathcal{E}_{\mathcal{F}}^k < \mathcal{E}_{\mathcal{T}}^0, \quad \forall k \in \mathbb{N}. \tag{4.30}$$

Both the kinetic energy of the particles and the field energy are bounded by the initial total energy of the system; *i.e.* the velocity of the particles and the electric field are bounded. Nonetheless, within the ECSPIC scheme, the field energy, which is defined by the combination of the component grid contributions, is not a non-negative quantity, the loss of non-negativity being a genuine property of the combination technique (see remark 3.4). It results from

this that numerical instabilities can arise in some contexts. *E.g.* let us assume a decreasing negative field energy:

$$\mathcal{E}_{\mathcal{F}}^{k} < 0, \quad \mathcal{E}_{\mathcal{F}}^{k} - \mathcal{E}_{\mathcal{F}}^{k+1} = \eta(k) > 0, \quad \forall k \in \mathbb{N} \tag{4.31}$$

then the kinetic energy verifies:

$$\mathcal{E}_{\mathcal{K}}^{k} \geq \mathcal{E}_{\mathcal{K}}^{0} + \sum_{i=0}^{k-1} \eta(i). \tag{4.32}$$

Since the field energy is not non-negative, the quantities $\eta(i)$ are not controlled and can be very large, leading to an instability manifested by increasing particle velocities and electric field. Note that if the field energy is non-negative, then

$$\sum_{i=0}^{k-1} \eta(i) \leq \mathcal{E}_{\mathcal{F}}^{0}, \tag{4.33}$$

and one recovers the bounds in (4.30).

5. Numerical results. In this section, the methods introduced in this paper are implemented in a sequential program written in Fortran 90 and compared to each other in a series of numerical tests: a Landau damping and a two-streams instability test cases. The SISPIC-sg and ECSPIC schemes are investigated, as well as their standard version (which are similar) which shall be named SISPIC-std, and compared to existing PIC schemes. The PIC explicit scheme [26] is considered both with standard grid (Ex-std) and sparse grid reconstructions (Ex-sg). The ECSIM scheme [39] is also considered as a comparison. The simulations aim at demonstrating the conservation and stability properties of the schemes introduced in this paper, as well as establishing the correctness of the methods in different configurations. Although the benefits of sparse grid reconstruction methods have proven to be significantly larger for three dimensional computations, the methods have been implemented in two dimensions of space and three dimensions of velocity (2d-3v) for simplicity. The author emphasizes on the large gap in term of computational time reduction gained thanks to sparse grid reconstructions between two and three dimensional computations. Indeed, in two dimensional simulations, sparse grid reconstructions for explicit schemes shows a slight reduction of the computational times in comparison to standard grid methods [26], whereas the reduction has proven to be substantial for three dimensional computations [24, 25]. This is a consequence of the significant particle sampling error difference between the sparse grid and the standard methods for two dimensional and three dimensional computations illustrated by the equations (3.51) and (3.52).

The computational times reported in this section are not necessarily representative of the exact efficiency of the methods since some of our implementations are known to not be optimal (ECSIM, ECSPIC): *e.g.* the ECSIM and Ex-sg schemes have not been implemented with iterative method but are expected to be more efficient than direct methods, especially for three dimensional computations. The execution times reported are rather meant to give an indication of the order of magnitude between the methods.†

†All the following simulations are performed on a laptop equipped with Intel® Core™ i9-10885H CPU with 8 cores @2.40 GHz sharing a L3 cache memory of 16MB and L1 cache memory of 32KB specific to each core. The Random-Access Memory (RAM) size is 32GB. The compilers used for the hardware is the GNU Fortran version 9.4.0 with options -cpp and optimizations -Ofast. All reported computational time are provided with frequency boost disabled. The linear systems are solved either with LU decomposition method from the MUMPS [2] library or with GMRES iterative method and algebraic multigrid preconditionner from the PETSc [3] library. In all the simulations, the tolerance of the iterative GMRES method is chosen so that the residual error is around 10^{-15}.

The domain is a periodic square $\Omega = (\mathbb{R}/L\mathbb{Z})^2$, of dimension $L \in \mathbb{R}_+^*$. Dimensionless variables are considered, the reference length and time units being the Debye length and the plasma period, defined by:

$$\lambda_D = \sqrt{\varepsilon_0 T_e / q_e n_0}, \qquad \omega_p^{-1} = 1/\sqrt{q_e n_0 / m_e \varepsilon_0}. \tag{5.1}$$

The electrons are considered immersed in a uniform, immobile, background of ions. Electron mass, temperature and charge are normalized to one. Periodic boundary conditions are considered for the particles and the field.

Throughout this section, we will refer to the mean number of particles per cell, denoted P_c, relating the amount of statistical noise in the simulation. For the standard methods, it depends on the number of particles and the Cartesian grid discretization:

$$P_c = N h_n^2. \tag{5.2}$$

An equivalent quantity for sparse grid reconstruction methods can be defined by considering the number of particles for all component grid cells [43]:

$$P_c = N h_n \left(3n - 1\right)^{-1}. \tag{5.3}$$

The results are compared without any filtering methods for all schemes. Let us now introduce the different diagnostics and error definitions. The momentum error in the simulation at iteration k is measured by the sum of the momentum error vector:

$$\varepsilon_{\mathcal{M}}^k := \left| \sum_{r=1}^{d} \left(\sum_s \frac{1}{N_s} \sum_{p=1}^{N_s} \frac{m_s \mathbf{v}_p^k - m_s \mathbf{v}_p^0}{m_s v_T} \right)_r \right|, \tag{5.4}$$

where $v_T := \sqrt{2 q_s T_s / m_s}$ is the thermal velocity of the electrons. The total discrete energy error at time iteration k is defined by:

$$\varepsilon_{\mathcal{E}}^k := \left| \sum_{r=1}^{d} \left(\frac{\mathcal{E}_{\mathcal{K}}^k + \mathcal{E}_{\mathcal{F}}^k - \left(\mathcal{E}_{\mathcal{K}}^0 + \mathcal{E}_{\mathcal{F}}^0 \right)}{\mathcal{E}_{\mathcal{K}}^0 + \mathcal{E}_{\mathcal{F}}^0} \right)_r \right|, \tag{5.5}$$

where $\mathcal{E}_{\mathcal{K}}^k$ and $\mathcal{E}_{\mathcal{F}}^k$ are the kinetic and field energy measured at time k.

5.1. Instability of the ECSPIC scheme. While performing numerical investigations on the ECSPIC scheme, we have experienced numerical instabilities not observed with the SISPIC or the ECSIM methods. The numerical instabilities are neither related to the time discretization nor the space discretization. We have found that it might be instead related to the loss of the field energy non-negativity. Indeed, according to the discussion in section 4.2, the reconstructed field energy is not guaranteed to be a non-negative quantity, even though the field energy defined on each component grid is non-negative. This is a drawback genuine to the combination technique: it does not preserve the non-negativity of the solution. This triggers an instability: when the field energy becomes negative the particle velocities increase to offset the loss in the field energy, preserving the system total energy (see proposition 4.1). This formal analysis is echoed by numerical investigations. We have observed that once the field energy is negative, it tends to become more and more negative, entailing the particle velocities increase, leading to numerical instabilities.

The conclusions drawn from these features call for the investigations of semi-implicit sparse-PIC methods that shall conserved a different discrete potential energy. This issue is left to a future work.

5.2. Finite-grid instability. The so-called aliasing or finite grid instability, first studied in [37], is a common numerical instability arising in PIC plasma simulations. It originates from the inconsistency between the discrete Eulerian discretization of the fields on a grid and the continuous discretization of Lagrangian particles in the phase-space [35].

This instability is manifested in simulations by a numerical heating of the plasma [7] related to the numerical parameters. Since the aliasing introduces artificial heat in the system, it is also characterized by a violation of the energy conservation. Usually in PIC simulations, the aliasing instability is avoided by choosing a grid discretization equal to or smaller than the Debye length ($h_n \leq \lambda_D$), including for problems with scales of interest much larger than the Debye length. For example, dense plasmas are well described by the quasi-neutral approximation in most of the domain and the simulation of the plasma physics does not required grid cells smaller than the Debye length. Therefore large gains could be obtained with coarse grid cells that do not resolve the Debye length, especially for three dimensional computations.

Various methods and numerical schemes have been proposed to mitigate this instability, including introducing grid interlacing [15], random jiggling [10], using higher order particle shapes [5] or temporal/spatial filtering [40]. Besides, semi-implicit scheme with exact energy conservation, such as the ECSIM scheme [39] have proven numerically to preserve the simulations from aliasing.

Nonetheless, analysis of the aliasing instability is not straightforward. It has been conducted linearly for stationary plasmas and specific schemes, such as the fully-implicit energy-conserving scheme in [4]. For drifting plasma however, it has been shown that in principle the scheme is not exempted from the instability, but that in practice the scheme is almost always freed from it. In this section, we address to establish numerically that the introduced schemes do not feature finite grid instability in classical configurations where the explicit discretizations does.

An initially Maxwellian and stable plasma is considered with the following distribution of electron:

$$f_v^0(\mathbf{v}) = \left(\frac{1}{\sqrt{\pi} v_T}\right)^3 e^{-\|\mathbf{v}\|_2^2/v_T^2}, \tag{5.6}$$

where $v_T = \sqrt{2T_e q_e/me}$ is the thermal velocity of electrons and $\|\mathbf{v}\|_2^2 = v_1^2 + v_2^2 + v_3^2$. The size of the domain is $L = 5\pi, 10\pi, 15\pi, 20\pi$ and the grid discretization is $h_n = 2^{-5}L$. Some of these configurations shall lead to the development of the finite-grid instability for the explicit schemes, since the grid discretization is larger than the Debye length.

The first dimension of the particle phase space (x_1, v_1) is represented at time $T = 100$ on figure 2 for different schemes and configurations of the domain. The finite grid instability is visible for the explicit schemes for the configurations in which the grid discretization is larger than the Debye length: the velocities of the particles are increased entailing an increase of the system total energy (initially the electric energy is equal to zero). This is a characterization of the numerical heating of particles. The implicit schemes are preserved from the finite grid instability for all configurations. The results for the ECSPIC scheme are not provided since the numerical instability discussed in the previous section prevents us to investigate the finite grid instability. The initially Maxwellian velocity distribution is preserved at any time and for all configurations with the SISPIC-std and SISPIC-sg schemes.

5.3. Landau damping. The first test case considered is the well-known Landau damping [1, 36]. When a plasma is slightly perturbed from an equilibrum state, it returns to its equilibrium with an exponential damping. A perturbation in the electron distribution of an

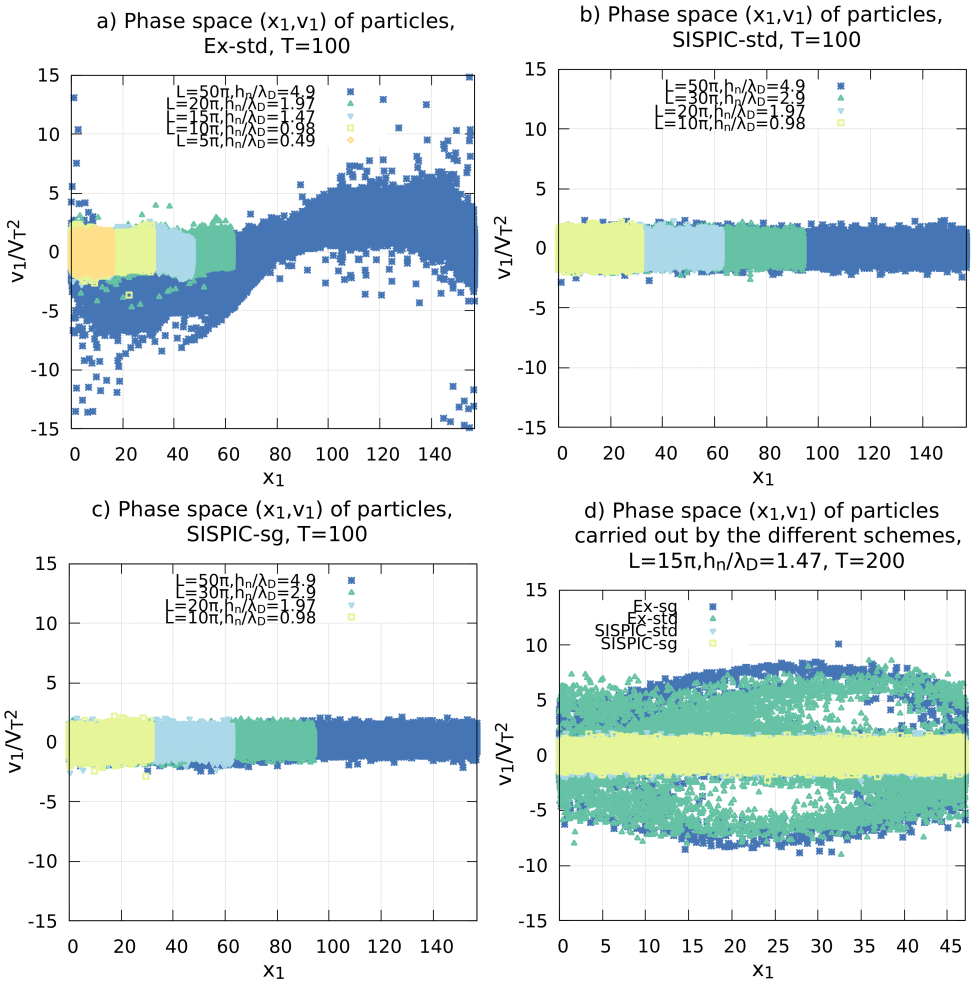

Fig. 2: Finite-grid instability: representation of the phase space (x_1, v_1) of an initially Maxwellian distribution of electron. $P_c = 100$, $\Delta t = 0.1$ for the explicit schemes and $\Delta t = 1$ for the implicit schemes.

equilibrium state is considered:

$$f_e(\mathbf{x}, \mathbf{v}, 0) = f_v^0(\mathbf{v}) f_x^0(\mathbf{x}), \tag{5.7}$$

where the initial velocity distribution is Maxwellian, similar to the previous configuration defined by equation (5.6), and the perturbation has the following form:

$$f_x^0(\mathbf{x}) = (1 + \alpha_1 \cos(k_1 x_1))\,(1 + \alpha_2 \cos(k_2 x_2))\,. \tag{5.8}$$

$\boldsymbol{\alpha}$ is the magnitude and $\mathbf{k}$ is the period of the perturbation. The perturbation is considered uniform in each dimension, *i.e.* $k_i = k$, $k \in \mathbb{R}$ and the domain size depends on the perturbation:

$$L = \frac{2\pi}{k}. \tag{5.9}$$

By considering the roots of the dispersion function ($\varepsilon(\omega, k) = 0$), which is as follows:

$$\frac{1}{\varepsilon_0}\varepsilon(\omega, k) = 1 + \frac{1}{k^2}\left(1 + \frac{\omega}{\sqrt{2}k} Z\left(\frac{\omega}{\sqrt{2}k}\right)\right), \tag{5.10}$$

one can find the damping rate of the plasma ($\Im(\omega)$) for given values of $k \in \mathbb{R}$ [7]. *E.g.* for $k = 0.3$, the root with the largest imaginary part is $\omega = \pm 1.1598 - 0.0126i$, etc.

Let us parametrize the perturbation with $\alpha_1 = \alpha_2 = 0.05$, $k = 0.3$ such that the domain size is $L = 20\pi/3$. The time step is $\Delta t = 3, 1, 0.5, 0.1, 0.005$, depending on the configurations and the final time is $T = 100$. The grid discretization is $h_n = 2^{-5}L$ so that the Debye length is resolved: $h_n \approx 0.65\lambda_D$ and the number of particles per cell ranges from $P_c = 100$ to 1000.

The evolution of the electric field L^2-norm in time is provided on figures 4 and 5 for different configurations described in table 3. For the sparse grid reconstruction schemes, the electric field L^2-norm is computed on the Cartesian grid after combination of the field. The corresponding evolution in time of the total energy, the total momentum, the error with respect to the Gauss' law and the difference in the field energy are represented on figure 6.

We observe on the left panel of figure 4 that the damping rate is not well reproduced for the ECSIM and SISPIC-std schemes with $P_c = 100$. The damping rate is accurately reproduced for the SISPIC-sg and ECSPIC schemes, despite the reduced total number of particles (more than two times fewer particles for the sparse schemes in comparison to standard schemes with equivalent P_c). Nonetheless, around $t \approx 30$, the recombined field energy of ECSPIC scheme becomes negative and the instability described in section 5.1 is observed: the velocities of the particles are significantly increased to balance the loss in the negative field energy (see figure 6, bottom panel). It results from the nonphysical growth of the particle velocities yielding an increase of the electric field L^2-norm (computed after combination on the Cartesian grid, *i.e.* the quantity observed on the figure 4). As a result of round-off errors, the total energy and total momentum are also increased.

On the right panel of figure 4, the SISPIC-sg scheme with $P_c = 50$ provides equivalent results than the SISPIC-std scheme with $P_c = 500$ and slightly better than the ECSIM scheme with $P_c = 1000$. These observations lead to two conclusions. First, the electrostatic Vlasov-(div)Ampere formulation (in which the particle response to the field is computed on the potential Φ) offers a reduction of the statistical noise in PIC discretizations compared to traditional Vlasov-Ampere formulation (in which the particle response to the field is computed on the field $\mathbf{E}$). Second, the results point out the significant gain in term of statistical resolution for semi-implicit methods embedding sparse grid reconstructions over standard semi-implicit methods. Indeed, the SISPIC-sg simulation on figure 4 (right panel) is performed with roughly 25 times fewer particles than the ECSIM simulation (with $P_c = 1000$) and 12 times fewer than the SISPIC-std simulation.

The SISPIC-sg scheme is compared to the explicit schemes on figure 5. On panel left, the damping rate is more accurate for SISPIC-sg scheme with $\Delta t = 0.5$, $P_c = 100$ than the explicit schemes with time steps 2,10 and 20 times smaller ($\Delta t = 0.1, 0.05, 0.025$), with $P_c = 500$. The stability, according to the time discretization, is investigated on figure 5 (right panel). All of the semi-implicit schemes, excepted the ECSPIC scheme, are stable for $\Delta t = 1, 3$. As a comparison, the explicit scheme, which is not unconditionally stable is represented with $\Delta t = 1$.

Note that, as depicted in the bottom right panel of Figure 6, the consistency with Gauss' law is maintained to 1E−02, 1E−03 for the SISPIC schemes and remains bounded over time. This implies that our Vlasov-Ampere formulation is approximately equivalent to the Vlasov-Poisson system of the explicit schemes.

The computational time of each step of an iteration and the total are provided in the table 7 for the different schemes. The field solve computational time of the ECSIM scheme is about

Table 3: Landau damping: configurations of the simulations (figure 9), $k = 0.05$.

scheme	Δt	h_n	P_c	N	figure (color)	comp. time one iteration [s]
Ex-std	0.05	2^{-5}	500	5.1E+05	5 (left)	4.2E−02(×1)
Ex-sg	0.05	2^{-5}	500	2.2E+05	5 (left)	5.0E−02 (×1.2)*
ECSIM	0.5	2^{-5}	1000	1.2E+06	4 (right)	3.1E−00 (×73)*
SISPIC-std	0.5	2^{-5}	500	5.1E+05	4 (right)	7.4E−02 (×1.7)
SISPIC-sg	0.5	2^{-5}	100	4.4E+04	4 (left)	7.6E−02 (×1.8)
ECSPIC	0.5	2^{-5}	100	4.4E+04	4 (left)	5.1E−01 (×12)*

*the implementation is not optimal

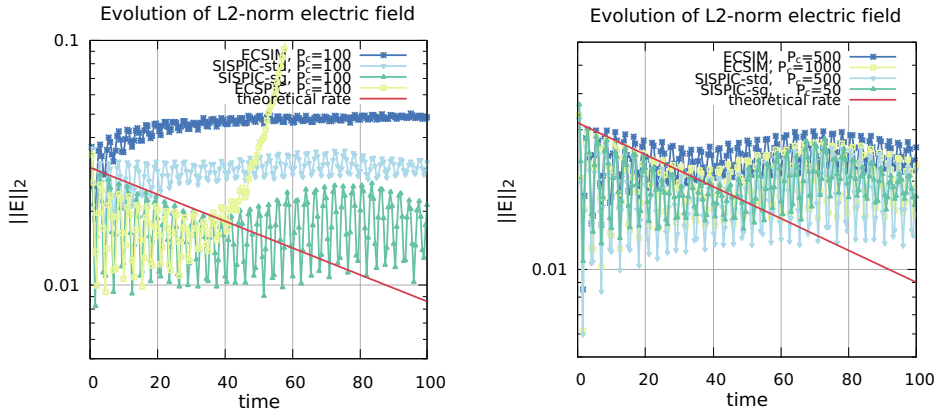

Fig. 4: Landau damping: evolution of the electric field L^2-norm $\|\mathbf{E}_{h_n}\|_{L^2}$ (on the Cartesian grid) in time, $k = 0.3$, $\Delta t = 0.5$ (*cf.* table 3).

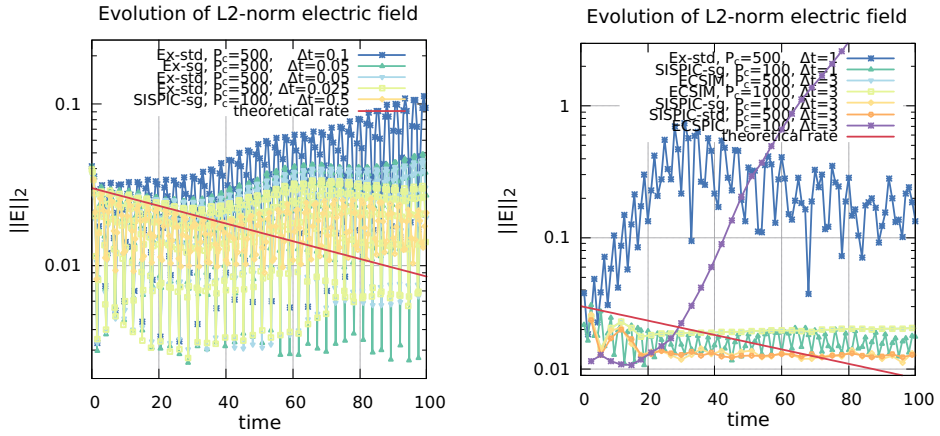

Fig. 5: Landau damping: evolution of the electric field L^2-norm $\|\mathbf{E}_{h_n}\|_{L^2}$ (on the Cartesian grid) in time, $k = 0.3$ (*cf.* table 3).

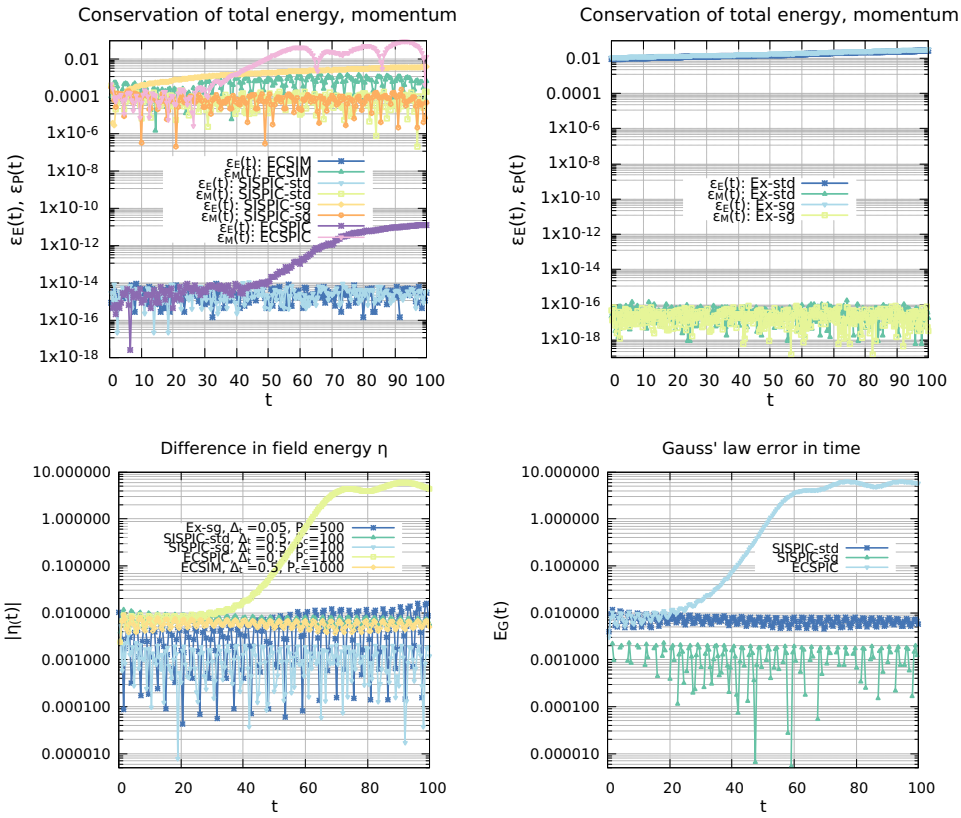

Fig. 6: Landau damping: evolution of the total energy $\epsilon_{\mathcal{E}}(t)$ and total momentum $\epsilon_{\mathcal{M}}(t)$ conservation error in time, $k = 0.3$. Semi-implicit: $P_c = 100$, $\Delta t = 0.5$ (top left panel); explicit: $P_c = 500$, $\Delta t = 0.05$ (top right panel); difference in field energy $\eta(t) = \sum_{i=0}^{k} \mathcal{E}_{\mathcal{F}}^{i} - \mathcal{E}_{\mathcal{F}}^{i+1}$, where k is such that $t_k = t$ (bottom right panel); Gauss' law error in time $\varepsilon_{\mathcal{G}}(t) := \|(\varepsilon_0 \nabla_h \cdot \mathbf{E}_h - \rho_h)(t)\|_\infty$, $P_c = 100$, $\Delta t = 0.5$ (bottom left panel).

5 times larger than the SISPIC schemes one (with direct method) because the linear system is 6 times larger (see section 3.4). The total computational time of the ECSIM is expected to be much smaller with an iterative method, about six times larger than the SISPIC-std scheme. The difference between SISPIC-std and SISPIC-sg is not significant here because the reduction of particles is not large (roughly 2 times fewer particles). Nonetheless, as already observed in [26, 24], the gains provided by sparse grid reconstructions are significantly larger for finer grid discretizations and three dimensional computations so that the difference between SISPIC-std and SISPIC-sg shall deepen in those configurations. For the configurations investigated here (equivalent P_c), the computational time of the SISPIC schemes are about 2 or 3 times larger than the explicit schemes; the computational time of the ECSPIC scheme is 10 times larger than the explicit sparse scheme. Note that the implementation of the mass matrix computation and linear system solve with iterative method (use of more efficient PETSc objects) for ECSPIC is not optimal; *e.g.* with a better implementation, the field solve computation time with iterative method shall be lower than SISPIC (because the size of the system is smaller).

Table 7: Computational time (mean of 10 iterations) for the Landau damping simulations, $k = 0.3$, $P_c = 100$, $n = 5$.

scheme	total [s] (min)	compute matrix, assembly [s]	field solve [s] direct (LU)	field solve [s] iterative (GMRES)	charge accumulation [s]	field interpolation [s]
Ex-std	2.5E−02	5.8E−04	1.0E−01	2.0E−02	2.0E−03	1.3E−03
Ex-sg	5.3E−02*	1.8E−05	3.1E−02	(not computed)	7.9E−03	1.4E−02
ECSIM	2.5E−00*	1.7E−01	2.4E−00	(not computed)	5.5E−03	6.5E−03
SISPIC-std	7.2E−02	4.6E−02	5.6E−01	1.7E−02	4.4E−03	2.3E−03
SISPIC-sg	7.6E−02	3.7E−02	5.8E−01	1.8E−02	1.7E−02	1.0E−03
ECSPIC	5.0E−01*	2.4E−01	2.0E−01	5.3E−01*	1.7E−02	2.8E−02

*the implementation is not optimal

5.4. Two-streams instability. The two-streams instability [6] configuration consists of two particle beams with opposite mean velocities. The following Maxwellian distribution of electrons is considered:

$$f_e(\mathbf{x}, \mathbf{v}, 0) = f_v^0(\mathbf{v}) f_x^0(\mathbf{x}), \tag{5.11}$$

where the perturbation has the same form than for the Landau damping and the initial velocity distribution is Maxwellian with two beams:

$$f_v^0(\mathbf{v}) = \left(\frac{1}{\sqrt{\pi v_T}}\right)^2 \left(e^{\frac{-\|\mathbf{v}-\mathbf{v}_0\|_2^2}{v_T^2}} + e^{\frac{-\|\mathbf{v}+\mathbf{v}_0\|_2^2}{v_T^2}} \right). \tag{5.12}$$

$\mathbf{v}_0 = (v_0, 0) \in \mathbb{R}^2$ is the mean velocity of the beams in opposite direction and the domain size is:

$$L = \frac{2\pi}{k}, \tag{5.13}$$

where $k_i = k \in \mathbb{R}$, for $i = 1, 2$. Depending on the values of k and v_0, the configuration is stable or unstable. Indeed, when two streams move through each other so that one wavelength is traveled in one cycle of the plasma frequency, the perturbation of one stream is increased by the other stream and the perturbation grows exponentially in time. The linear dispersion relation for this test case is:

$$\frac{1}{\varepsilon_0}\varepsilon(\omega, \mathbf{k}) = 1 - \frac{\omega_p^2}{(\omega - \mathbf{k}\cdot\mathbf{v}_0)^2} - \frac{\omega_p^2}{(\omega + \mathbf{k}\cdot\mathbf{v}_0)^2}. \tag{5.14}$$

The four roots of the linear dispersion relation are [7]:

$$\omega = \pm \left[k^2 v_0^2 + \omega_p^2 \pm \omega_p \left(4k^2 v_0^2 + \omega_p^2 \right)^{\frac{1}{2}} \right]^{\frac{1}{2}}, \tag{5.15}$$

which can be imaginary, and lead to instability, for:

$$0 \leq \frac{k v_0}{\omega_p} \leq \sqrt{2}. \tag{5.16}$$

Let us parametrize the perturbation with $\alpha_1 = \alpha_2 = 0.005$, $k = 0.05$ such that the domain size is $L = 40\pi$. The mean velocity is $v_0 = 12$, the time step is $\Delta t = 0.1$ and the final time $T = 60$, resulting in 600 iterations. The grid discretization is $h_n = 2^{-5}L$ for semi-implicit schemes and $h_n = 2^{-7}L$ for explicit schemes so that the Debye length is resolved for explicit schemes: $h_n \simeq 0.98\lambda_D$. The number of particles per cell is $P_c = 100$ or 500. The evolution of the electric field L^2-norm is provided on figure 9 for different configurations described in table 8.

Table 8: Configurations for the two-streams instability simulations (figure 9), $k = 0.05$.

scheme	Δt	h_n	P_c	N	figure (color)	comp. time one iteration [s]
Ex-std	0.1	2^{-7}	500	8.2E+06	9 ■	2.3E−00 (÷1)
Ex-sg	0.1	2^{-7}	500	1.2E+06	9 ■	1.0E−00 (÷2.3)*
SISPIC-std	0.1	2^{-5}	500	5.1E+05	9 ■	1.5E−01 (÷15.3)
SISPIC-sg	0.1	2^{-5}	100	4.4E+04	9 ■	7.5E−02 (÷30.6)
ECSPIC	0.1	2^{-5}	100	4.4E+04	9 ■	5.3E−01 (÷4.3)*

*the implementation is not optimal

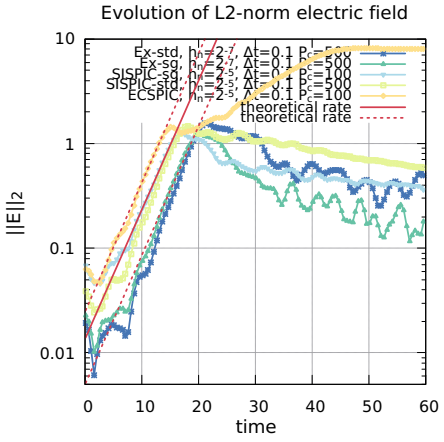

Fig. 9: Two-streams instability: evolution of the electric field L^2-norm $\|\mathbf{E}_{h_n}\|_{L^2}$ (on the Cartesian grid) in time, $k = 0.05$, $\Delta t = 0.1$ (*cf.* table 8).

Here again, the ECSPIC scheme is subject to the numerical instability related to the loss of field energy non-negativity. The instability is triggered at the moment when the field energy become negative (around $t \approx 18$). The theoretical rate is well reproduced for both the explicit and the semi-implicit schemes. The SISPIC schemes are significantly more efficient, in term of computational time, than the explicit schemes, the execution time being 15 or 30 times smaller than for the explicit scheme on standard grid. The ECSPIC scheme is slightly more efficient than the explicit sparse grid scheme but we recall that the implementation is not optimal and gains may be obtained with optimization on the computation of the mass matrix and the field solver with iterative method, as well as the dimensionality of the problem.

Conclusion. In this paper, numerical methods based on an implicit discretization of the Vlasov-Maxwell system in electrostatic regime and embedding sparse grid reconstructions have been introduced: the SISPIC-sg, SISPIC-std and ECSPIC schemes. These methods have been numerically experienced and compared against existing semi-implicit (ECSIM) and explicit PIC methods. Sparse grid reconstructions embedded in PIC discretizations offer a reduction of the memory cost of the method thanks to a better control of the statistical noise which entails a decrease of the particle number. Indeed, we have observed numerically that the number of particles required to obtain a given accuracy can be reduced by a factor of 25 with sparse grid semi-implicit schemes in comparison to standard grid schemes. In addition, the methods are unconditionally stable with respect to the plasma period so that the time step can be chosen irrespective to this value; and the finite grid instability is eliminated, permitting to choose the mesh size according to the physic of interest rather than the Debye length. Nonetheless, we have observed numerical instabilities related to the loss of the field energy non-negativity in the ECSPIC scheme, these instabilities deteriorating significantly the results in course of time. The SISPIC-sg method is exempted from this instability and has proven to be the most efficient method (compared to the explicit schemes, ECSPIC and ECSIM) in term of memory footprint and computational times. The gains, observed on two dimensional simulations, are expected to be larger with three dimensional geometries for which the reduction of particles achieved by the sparse grid reconstruction is more significant (see figure 10, extracted from [26]). Indeed, the full potential of the sparse grid reconstruction method can only be achieved by three dimensional computations. This is strikingly illustrated by the plot of figure 10 relating the memory footprint of the particle data for standard and sparse grid simulations to guarantee a equivalent statistical noise.

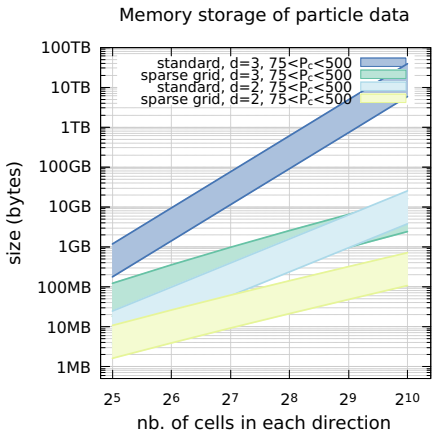

Fig. 10: Memory footprint of the particle data for standard and sparse grid simulations, with $75 \leq P_c \leq 500$.

However, the schemes introduced are not consistent with the charge continuity equation (as Gauss's law is not strictly satisfied), indicating that the Vlasov-Ampere formulation is not equivalent to the Vlasov-Poisson system. Therefore, corrections, such as those introduced in [16] and [12], are planned for future work. We conclude our investigations upon semi-implicit schemes embedding sparse grid reconstructions with the following considerations. The SISPIC-sg method is a good alternative to explicit sparse-PIC schemes for configurations in which the time and space scales of interest are larger than the plasma period or/and

the Debye length. The use of sparse grid techniques shall offer a significant reduction of the computational costs (memory footprint, computational time) for three dimensional geometries. Nonetheless the derivation of a semi-implicit sparse-PIC method conserving the discrete total energy is left to a future work.

Acknowledgements. I am grateful for discussions with Fabrice Deluzet, Gwenael Fubiani, Laurent Garrigues and Jacek Narski.
Clément Guillet benefits from a Université de Toulouse/Région Occitanie PhD grant.
This work has been carried out within the framework of the EUROfusion Consortium, funded by the European Union via the Euratom Research and Training Programme (Grant Agreement No 101052200 — EUROfusion). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the European Commission can be held responsible for them.
This work has been supported by a public grant from the "Laboratoire d'Excellence Centre International de Mathématiques et d'Informatique" (LabEx CIMI) overseen by the French National Research Agency (ANR) as part of the "Investissements d'Avenir" program (reference ANR-11-LABX-0040) in the frame of the PROMETEUS project (PRospect of nOvel nuMerical modEls for elecTric propulsion and low tEmperatUre plaSmas).
This work has been supported by a grant from the French National Research Agency (ANR) project MATURATION (reference ANR-22-CE46-0012)
Support from the FrFCM (Fédération de recherche pour la Fusion par Confinement Magnétique) in the frame of the SPARCLE project (SParse grid Acceleration for the paRticle-in-CelL mEthod) and the BRIDIPIC project "BRIDging Particle-In-Cell methods and low frequency numerical models of plasmas" is also acknowledged.

REFERENCES

[1] 61 - on the vibrations of the electronic plasma. In D. TER HAAR, editor, *Collected Papers of L.D. Landau*, pages 445–460. Pergamon, 1965.

[2] P.R. Amestoy, I. S. Duff, J. Koster, and J.-Y. L'Excellent. A fully asynchronous multifrontal solver using distributed dynamic scheduling. *SIAM Journal on Matrix Analysis and Applications*, 23(1):15–41, 2001.

[3] Satish Balay, Shrirang Abhyankar, Mark F. Adams, Steven Benson, Jed Brown, Peter Brune, Kris Buschelman, Emil M. Constantinescu, Lisandro Dalcin, Alp Dener, Victor Eijkhout, William D. Gropp, Václav Hapla, Tobin Isaac, Pierre Jolivet, Dmitry Karpeev, Dinesh Kaushik, Matthew G. Knepley, Fande Kong, Scott Kruger, Dave A. May, Lois Curfman McInnes, Richard Tran Mills, Lawrence Mitchell, Todd Munson, Jose E. Roman, Karl Rupp, Patrick Sanan, Jason Sarich, Barry F. Smith, Stefano Zampini, Hong Zhang, Hong Zhang, and Junchao Zhang. PETSc Web page. https://petsc.org/, 2022.

[4] D.C. Barnes and L. Chacón. Finite spatial-grid effects in energy-conserving particle-in-cell algorithms. *Computer Physics Communications*, 258:107560, January 2021.

[5] Charles K. Birdsall and Neil Maron. Plasma self-heating and saturation due to numerical instabilities. *Journal of Computational Physics*, 36(1):1–19, June 1980.

[6] C.K. Birdsall. *Interaction Between Two Electron Streams for Microwave Amplification*. Department of Electrical Engineering, Stanford University, 1951.

[7] C.K. Birdsall and A.B Langdon. *Plasma Physics via Computer Simulation*. CRC Press, 0 edition, October 2018.

[8] J.U. Brackbill. On energy and momentum conservation in particle-in-cell plasma simulation. *Journal of Computational Physics*, 317:405–427, July 2016.

[9] J.U Brackbill and D.W Forslund. An implicit method for electromagnetic plasma simulation in two dimensions. *Journal of Computational Physics*, 46(2):271–308, May 1982.

[10] J.W. Brackbill and Giovanni Lapenta. A Method to Suppress the Finite-Grid Instability in Plasma Simulations. *Journal of Computational Physics*, 114(1):77–84, September 1994.

[11] H.-J. Bungartz, M. Griebel, D. Röschke, and C. Zenger. Pointwise Convergence Of The Combination Technique For Laplace's Equation. *East-West J. Numer. Math*, 2:21–45, 1994.

[12] Martin Campos Pinto and Valentin Pagès. A semi-implicit electromagnetic FEM-PIC scheme with exact energy and charge conservation. *Journal of Computational Physics*, 453:110912, March 2022.

[13] G. Chen and L. Chacón. A multi-dimensional, energy- and charge-conserving, nonlinearly implicit, electromagnetic Vlasov–Darwin particle-in-cell algorithm. *Computer Physics Communications*, 197:73–87, December 2015.

[14] G. Chen, L. Chacón, and D.C. Barnes. An energy- and charge-conserving, implicit, electrostatic particle-in-cell algorithm. *Journal of Computational Physics*, 230(18):7018–7036, August 2011.

[15] Liu Chen, A Bruce Langdon, and C.K Birdsall. Reduction of the grid effects in simulation plasmas. *Journal of Computational Physics*, 14(2):200–222, February 1974.

[16] Yuxi Chen and Gábor Tóth. Gauss's Law satisfying Energy-Conserving Semi-Implicit Particle-in-Cell method. *Journal of Computational Physics*, 386:632–652, June 2019.

[17] Bruce I Cohen, A.Bruce Langdon, and A Friedman. Implicit time integration for plasma simulation. *Journal of Computational Physics*, 46(1):15–38, August 1982.

[18] Anaïs Crestetto, Nicolas Crouseilles, and Mohammed Lemou. A particle micro-macro decomposition based numerical scheme for collisional kinetic equations in the diffusion scaling. *Communications in Mathematical Sciences*, 16(4):887–911, 2018.

[19] Anaïs Crestetto, Nicolas Crouseilles, Mohammed Lemou, ,INRIA-Nancy Grand Est, CALVI Project, and IRMA, Université de Strasbourg, 67084, STRASBOURG, ,INRIA-Rennes Bretagne Atlantique, IPSO Project, and IRMAR (Université de Rennes 1), 35042 RENNES, and ,CNRS and IRMAR (Université de Rennes 1), and INRIA-Rennes Bretagne Atlantique, IPSO Project, 35042 RENNES. Kinetic/fluid micro-macro numerical schemes for Vlasov-Poisson-BGK equation using particles. *Kinetic & Related Models*, 5(4):787–816, 2012.

[20] Nicolas Crouseilles, Mohammed Lemou, ,INRIA-Nancy Grand Est and IRMA, Université de Strasbourg, 7, rue René Descartes, 67084 Strasbourg, and ,CNRS and IRMAR, Université de Rennes 1, 263 Avenue du General Leclerc CS74205, 35042 Rennes cedex. An asymptotic preserving scheme based on a micro-macrodecomposition for Collisional Vlasov equations: diffusion and high-field scaling limits. *Kinetic & Related Models*, 4(2):441–477, 2011.

[21] Lars K.S. Daldorff, Gábor Tóth, Tamas I. Gombosi, Giovanni Lapenta, Jorge Amaya, Stefano Markidis, and Jeremiah U. Brackbill. Two-way coupling of a global Hall magnetohydrodynamics model with a local implicit particle-in-cell model. *Journal of Computational Physics*, 268:236–254, July 2014.

[22] John M. Dawson. Particle simulation of plasmas. *Rev. Mod. Phys.*, 55(2):403–447, April 1983. Publisher: American Physical Society.

[23] Pierre Degond, Fabrice Deluzet, and David Doyen. Asymptotic-preserving Particle-In-Cell methods for the Vlasov-Maxwell system near quasi-neutrality. *arXiv:1509.04235 [physics]*, September 2015. arXiv: 1509.04235.

[24] F. Deluzet, G. Fubiani, L. Garrigues, C. Guillet, and J. Narski. Efficient parallelization for 3d-3v sparse grid particle-in-cell: shared memory architectures. Submitted to Journal of Computational physics, 2022.

[25] F. Deluzet, G. Fubiani, L. Garrigues, C. Guillet, and J. Narski. Efficient parallelization for 3d-3v sparse grid particle-in-cell: Single gpu architectures. Submitted to Computer Physics Communications, 2022.

[26] Fabrice Deluzet, Gwenael Fubiani, Laurent Garrigues, Clément Guillet, and Jacek Narski. Sparse grid reconstructions for Particle-In-Cell methods. *ESAIM: M2AN*, 56(5):1809–1841, September 2022.

[27] J Denavit and JM Walsh. Proceedings of the ninth conference on numerical simulation of plasmas. *J. Comput. Phys*, 42:337, 1981.

[28] R. E. Denton and M. Kotschenreuther. {delta}f Algorithm. Technical Report DOE/ET/53088-629; IFSR-629, Texas Univ., Austin, TX (United States). Inst. for Fusion Studies, November 1993.

[29] Laurent Garrigues, Gwénaël Fubiani, and Jean-Pierre Boeuf. Negative ion extraction via particle simulation for fusion: critical assessment of recent contributions. *Nuclear Fusion*, 57(1):014003, January 2017. Publisher: IOP Publishing.

[30] Salimou Gassama, Éric Sonnendrücker, Kai Schneider, Marie Farge, and Margarete Domingues. Wavelet denoising for postprocessing of a 2D Particle - In - Cell code. *http://dx.doi.org/10.1051/proc:2007013*, 16, January 2007.

[31] Michael Griebel. The combination technique for the sparse grid solution of pde's on multiprocessor machines. *Parallel Process. Lett.*, 02(01):61–70, March 1992. Publisher: World Scientific Publishing Co.

[32] Michael Griebel, Micha Schneider, and Christoph Zenger. A combination technique for the solution of sparse grid problems. 1992.

[33] Dennis W Hewett and A Bruce Langdon. Electromagnetic direct implicit plasma simulation. *Journal of Computational Physics*, 72(1):121–155, September 1987.

[34] Dennis W Hewett and A Bruce Langdon. Electromagnetic direct implicit plasma simulation. *Journal of Computational Physics*, 72(1):121–155, 1987.

[35] C.-K. Huang, Y. Zeng, Y. Wang, M.D. Meyers, S. Yi, and B.J. Albright. Finite grid instability and spectral fidelity of the electrostatic Particle-In-Cell algorithm. *Computer Physics Communications*, 207:123–135, October 2016.

[36] Nicholas A. Krall and Alvin W. Trivelpiece. Principles of Plasma Physics. *American Journal of Physics*, 41(12):1380–1381, December 1973. Publisher: American Association of Physics Teachers.

[37] A.Bruce Langdon. Effects of the spatial grid in simulation plasmas. *Journal of Computational Physics*, 6(2):247–267, October 1970.
[38] A.Bruce Langdon, Bruce I Cohen, and Alex Friedman. Direct implicit large time-step particle simulation of plasmas. *Journal of Computational Physics*, 51(1):107–138, 1983.
[39] Giovanni Lapenta. Exactly energy conserving semi-implicit particle in cell formulation. *Journal of Computational Physics*, 334:349–366, April 2017.
[40] H.Ralph Lewis, A Sykes, and J.A Wesson. A comparison of some particle-in-cell plasma simulation methods. *Journal of Computational Physics*, 10(1):85–106, August 1972.
[41] Rodney J Mason. Implicit moment particle simulation of plasmas. *Journal of Computational Physics*, 41(2):233–244, June 1981.
[42] Sriramkrishnan Muralikrishnan, Antoine J. Cerfon, Matthias Frey, Lee F. Ricketson, and Andreas Adelmann. Sparse grid-based adaptive noise reduction strategy for particle-in-cell schemes. *Journal of Computational Physics: X*, 11:100094, June 2021.
[43] L F Ricketson and A J Cerfon. Sparse grid techniques for particle-in-cell schemes. *Plasma Phys. Control. Fusion*, 59(2):024002, February 2017.
[44] R. D. Sydora. Low-noise electromagnetic and relativistic particle-in-cell plasma simulation models. *Journal of Computational and Applied Mathematics*, 109(1):243–259, September 1999.
[45] Gábor Tóth, Yuxi Chen, Tamas I. Gombosi, Paul Cassak, Stefano Markidis, and Ivy Bo Peng. Scaling the Ion Inertial Length and Its Implications for Modeling Reconnection in Global Simulations. *JGR Space Physics*, 122(10), October 2017.
[46] Gábor Tóth, Xianzhe Jia, Stefano Markidis, Ivy Bo Peng, Yuxi Chen, Lars K. S. Daldorff, Valeriy M. Tenishev, Dmitry Borovikov, John D. Haiducek, Tamas I. Gombosi, Alex Glocer, and John C. Dorelli. Extended magnetohydrodynamics with embedded particle-in-cell simulation of Ganymede's magnetosphere. *J. Geophys. Res. Space Physics*, 121(2):1273–1293, February 2016.
[47] H.X. Vu and J.U. Brackbill. Celest1d: an implicit, fully kinetic model for low-frequency, electromagnetic plasma simulation. *Computer Physics Communications*, 69(2):253–276, 1992.
[48] Kane Yee. Numerical solution of initial boundary value problems involving maxwell's equations in isotropic media. *IEEE Transactions on Antennas and Propagation*, 14(3):302–307, 1966.

Appendix A. Appendix.

A.1. Spatial discretization. In this section, the spatial discretization of the physical quantities (electric field, charge current, electric potential, etc.), as well as operators (gradient and divergence), on the component grids are explicited in two and three dimensions.

Let us consider a component grid $\Omega_{h_\mathbf{l}}$ with grid discretization $h_\mathbf{l}$ (or the Cartesian grid, *i.e.* $l_i = n, \forall i$). The scalar quantities, such as the electric potential, are defined at the vertices of the grid cells:

$$\Phi_{h_\mathbf{l};\mathbf{j}} = \Phi_{h_\mathbf{l};i,j,k} \in \Omega_{h_\mathbf{l}}, \quad \text{where } \mathbf{j} = (i,j,k) \in I_{h_\mathbf{l}}. \tag{A.1}$$

The notation $\Phi_{h_\mathbf{l};i,j,k}$ stands for the electric potential approximation at the grid node $\mathbf{j}h_\mathbf{l} = (ih_{l_1}, jh_{l_2}, kh_{l_3})$.

The field quantities, such as the electric field and current density, are defined on the centers and vertices of the grid cells according to the Yee discretization [48]. The Yee discretization of a component grid consists of d staggered component grids, $\Omega_{h_\mathbf{l}}^{y_1}$, $\Omega_{h_\mathbf{l}}^{y_2}$ and $\Omega_{h_\mathbf{l}}^{y_3}$ defined by:

$$\Omega_{h_\mathbf{l}}^{y_i} := \left\{\mathbf{j}^{y_i} h_\mathbf{l} \mid \mathbf{j} \in I_{h_\mathbf{l}}\right\} \subset \Omega, \quad i = 1, ..., d \tag{A.2}$$

and we introduce the notation $\mathbf{j}^y$ for an index $\mathbf{j} \in I_{h_\mathbf{l}}$, defined by:

$$\mathbf{j}^y = (\mathbf{j}^{y_i})_{i=1,...,d}, \qquad \text{where } \mathbf{j}^{y_i} = \mathbf{j} + \mathbf{e}_i/2, \tag{A.3}$$

where $\mathbf{e}_i \in \mathbb{N}^d$ is the index whose value is 1 along the i^{th} coordinate and 0 elsewhere. Specifically, in two and three dimensions, it falls down to:

$$\mathbf{j}^y = \begin{pmatrix} (i+1/2, j) \\ (i, j+1/2) \end{pmatrix} \text{ if } d = 2, \qquad \mathbf{j}^y = \begin{pmatrix} (i+1/2, j, k) \\ (i, j+1/2, k) \\ (i, j, k+1/2) \end{pmatrix} \text{ if } d = 3. \tag{A.4}$$

Let $\Omega^y_{h_\mathbf{l}} = (\Omega^{y_i}_{h_\mathbf{l}})_{i=1,\dots,d}$ denotes the staggered component grids. Then, the electric field and current density discretizations are written as:

$$\mathbf{E}_{h_\mathbf{l};\mathbf{j}^y} = (\mathbf{E}_{h_\mathbf{l};\mathbf{j}^{y_i}})_{i=1,\dots,d} \in \Omega^y_{h_\mathbf{l}}, \quad \mathbf{J}_{h_\mathbf{l};\mathbf{j}^y} = (\mathbf{J}_{h_\mathbf{l};\mathbf{j}^{y_i}})_{i=1,\dots,d} \in \Omega^y_{h_\mathbf{l}}. \tag{A.5}$$

Let us introduce the discrete gradient, discrete divergence, and discrete Laplacian operators defined on the component grids. The discrete gradient is defined from the regular grid to the staggered grid; the discrete divergence from the staggered grid to the regular grid; and the discrete Laplacian from the regular grid to the regular grid by:

$$\left(\boldsymbol{\nabla}_{h_\mathbf{l}} \Phi_{h_\mathbf{l};\mathbf{j}}\right)_{h_\mathbf{l};\mathbf{j}^y} = \left(\frac{\Phi_{h_\mathbf{l};\mathbf{j}+\mathbf{e}_i} - \Phi_{h_\mathbf{l};\mathbf{j}}}{h_{l_i}}\right)_{i=1,\dots,d}, \tag{A.6}$$

$$\left(\boldsymbol{\nabla}_{h_\mathbf{l}} \cdot \mathbf{E}_{h_\mathbf{l};\mathbf{j}^y}\right)_{h_\mathbf{l};\mathbf{j}} = \sum_{i=1}^{d} \frac{\mathbf{E}_{h_\mathbf{l};\mathbf{j}^{y_i}} - \mathbf{E}_{h_\mathbf{l};\mathbf{j}^{y_i}-\mathbf{e}_i}}{h_{l_i}}, \tag{A.7}$$

$$\left(\Delta_{h_\mathbf{l}} \Phi_{h_\mathbf{l};\mathbf{j}}\right)_{h_\mathbf{l};\mathbf{j}} = \nabla_{h_\mathbf{l}} \cdot \nabla_{h_\mathbf{l}} \Phi_{h_\mathbf{l};\mathbf{j}}. \tag{A.8}$$

The motivation for the introduction of the staggered discretization is to retain some properties of the continuum gradient and divergent operators. Specifically, the discrete gradient and discrete divergence operators shall verify a discrete integration by parts for exact conservation of energy.

Lemma A.1 (Discrete integration by parts). *Let $A_{h_l;\boldsymbol{j}}$ be a scalar quantity defined on the component grid Ω_{h_l} and $\boldsymbol{B}$ be a field quantity defined on the staggered component grids $\Omega^y_{h_l}$, then the following discrete integration by parts holds:*

$$\sum_{\boldsymbol{j} \in I_{h_l}} \boldsymbol{\nabla}_{h_l} A_{h_l;\boldsymbol{j}} \cdot \boldsymbol{B}_{h_l;\boldsymbol{j}^y} = - \sum_{\boldsymbol{j} \in I_{h_l}} A_{h_l;\boldsymbol{j}} \boldsymbol{\nabla}_{h_l} \cdot \boldsymbol{B}_{h_l;\boldsymbol{j}^y} \tag{A.9}$$

Proof. The result is obtained with the periodicity of the component grids. ∎