

Energy-conserving time propagation for a structure-preserving particle-in-cell Vlasov–Maxwell solver

Katharina Kormann ^{a,b,*}, Eric Sonnendrücker ^{a,b}

^a Max Planck Institute for Plasma Physics, Boltzmannstr. 2, 85748 Garching, Germany

^b Technical University of Munich, Department of Mathematics, Boltzmannstr. 3, 85748 Garching, Germany

ARTICLE INFO

Article history:

Received 5 August 2019

Received in revised form 25 September 2020

Accepted 29 September 2020

Available online 2 October 2020

Keywords:

Particle-in-cell

Geometric numerical methods

Discrete gradient

Vlasov–Maxwell

ABSTRACT

This paper discusses energy-conserving time-discretizations for finite element particle-in-cell discretizations of the Vlasov–Maxwell system. A geometric spatially discrete system can be obtained using a standard particle-in-cell discretization of the particle distribution and compatible finite element spaces for the fields to discretize the Poisson bracket of the Vlasov–Maxwell model (see Kraus et al. (2017) [1]). In this paper, we derive energy-conserving time-discretizations based on the discrete gradient method applied to an antisymmetric splitting of the Poisson matrix. Firstly, we propose a semi-implicit method based on a splitting that yields constant Poisson matrices in each substep. Moreover, we devise an alternative discrete gradient that yields a time discretization that can additionally conserve Gauss' law. Finally, we explain how substepping for fast species dynamics can be incorporated.

© 2020 The Author(s). Published by Elsevier B.V.

1. Introduction

Particle-in-cell simulations are widely used in the plasma community to solve the Vlasov–Maxwell system due to their ease of implementation and their favorable scaling properties in higher dimensions. Recently, a systematic derivation of geometric particle-in-cell methods has been proposed by Kraus, Kormann, Morrison, & Sonnendrücker [1]. A similar model has also been obtained by He, Sun, Qin, & Liu [2]. The derivation is based on compatible finite elements for the fields and a standard particle-in-cell ansatz for the distribution function. The derived semi-discrete system conserves Casimir invariants of the system such as discrete versions of $\nabla \cdot \mathbf{B} = 0$ and $\nabla \cdot \mathbf{E} = \rho$. For the time discretization, a Hamiltonian splitting method was proposed following [3], which preserves the Poisson structure and yields an explicit scheme for Cartesian coordinates. In particular, Gauss' law is conserved over time, but only a modified energy. While one step of this method is very efficient, the time step is restricted by stability constraints. In particular, the explicit solution of the vacuum Maxwell equations yields a stability limit depending on the speed of light (cf. Appendix A.2). Moreover, the plasma frequency needs to be resolved (cf. Appendix A.1). This motivates our investigations on alternative time-stepping schemes for the geometric electromagnetic particle-in-cell method that allow for larger time steps, while still preserving structure as much as possible.

In the geometric integration community, two categories of numerical schemes have been developed: symplectic or Poisson integrators and energy-conserving methods. While the Hamiltonian splitting falls into the first category, this paper explores the possibilities of energy-conserving schemes. Note that energy-conserving numerical methods cannot be sym-

* Corresponding author at: Max Planck Institute for Plasma Physics, Boltzmannstr. 2, 85748 Garching, Germany.

E-mail address: katharina.kormann@ipp.mpg.de (K. Kormann).

plectic for general Hamiltonians (see [4]). This work devises a temporal discretizations based on discrete gradients. The discrete gradient method is a general framework to design energy-conserving time discretizations for conservative partial differential equations in skew-symmetric form and was first introduced by McLachlan, Quispel, & Robidoux [5]. Several special cases have been devised, in particular the average-vector-field method introduced by Celledoni et al. [6]. Applying discrete gradients to the full Vlasov–Maxwell system results in a heavily nonlinear scheme. On the other hand, the method of discrete gradients can be applied after a splitting of the equations that respects the skew-symmetry without loss of energy conservation. As a first scheme, we propose a semi-implicit method that applies the discrete gradient method to the subsystems that cannot be solved analytically. The scheme is only implicit in the field solver, whence the computational overhead is relatively small compared to the explicit method. On the other hand, this method does not conserve Gauss’ law. We therefore devise a second scheme where we reduce the splitting and derive an approximation of the time-dependent Poisson matrix that conserves Gauss’ law in addition to the energy. The resulting system nonlinearly couples the particle and field equations and therefore needs to be solved in a nonlinear iteration. In a simulation with fast electrons and slower ions, substepping for the trajectories of the faster species can be crucial to reflect the multiscale nature of the system. Such a substepping technique can be incorporated in our implicit scheme.

In the plasma physics community, energy-conserving particle-in-cell methods have also been developed, mostly with a finite difference description of the fields. Markidis & Lapenta [7] have devised the so-called EC-PIC method for the Vlasov–Maxwell system that is fully nonlinear: The method uses a finite difference description of the fields on a Yee grid and employs differencing by the implicit midpoints rule both in space and time. With some rearrangements of the equations they yield an implicit formulation for the update of the fields and an average velocity. The method conserves energy but Gauss’ law is not preserved over time. Later Lapenta derived a semi-implicit version, the so-called energy-conserving semi-implicit particle-in-cell method (ECSIM) [8], that only requires an implicit field solver. When transferring the ECSIM method to a finite element formulation of the fields, the resulting method is very close to our energy-conserving semi-implicit scheme derived from the discrete gradient method. This has been investigated in [9]. Compared to the ECSIM method, we follow a more systematic derivation that ensures second-order accuracy which is lost in one part of the ECSIM method as pointed out in [9]. Similar to the semi-implicit method derived in this work, the ECSIM method is energy-conserving but does not satisfy Gauss’ law. On the other hand, Gauss’ law can be reinforced by different Lagrange multiplier techniques as proposed by Marder [10], Langdon [11] and by Munz, Omnes, Schneider, Sonnendrücker, & Voss [12]. However, these techniques are not compatible with the energy conservation in our scheme. Recently, Chen & Tóth [13] have therefore proposed a different procedure for the ECSIM algorithm that uses a correction of the particle positions instead and respects the energy conservation. However, this method is built on linearized shape functions and is therefore only a first order correction.

Chen, Chacón, & Barnes have developed fully implicit particle-in-cell methods that conserve both energy and Gauss’ law for the electrostatic Vlasov–Ampère model [14] as well as the reduced electromagnetic Vlasov–Darwin model [15]. Chen & Chacón [16] have also proposed an energy- and charge-conserving scheme for the relativistic Vlasov–Maxwell system. The scheme has similarities with our implicit scheme but is different in two aspects: Most importantly it is semi-implicit in the sense that it treats the curl-part of Maxwell’s equation explicitly. For this reason, the time step is restricted by the same stability condition that is derived for the explicit Hamiltonian splitting in Appendix A.2. Moreover, the charge-conserving current deposition is given in a special form for linear shape functions. There are also specific issues related to the relativistic Vlasov–Maxwell model and the correct propagation of light waves addressed in this paper that are not relevant for us as our target application is a fully kinetic (at least for ions) simulation of Tokamak plasmas in regimes where the gyrokinetic approximation typically used is not valid any more.

The outline of the paper is as follows: In the next section, we introduce the Vlasov–Maxwell model and the geometric electromagnetic particle-in-cell (GEMPIC) framework for its spatial semi-discretization. Section 3 introduces the discrete gradient method and a splitting of the Poisson matrix of the semi-discrete Vlasov–Maxwell system and Section 4 devises an energy-conserving semi-implicit discrete gradient method for the split equations. In the subsequent section 5, we explain how the scheme can be modified to conserve as well Gauss’ law. Numerical experiments on simple test problems presented in Section 6 confirm the conservation properties of the new methods.

2. Geometric electromagnetic particle in cell

In the GEMPIC framework [1], the Vlasov–Maxwell equations are discretized by a standard particle-in-cell ansatz for the distribution function and compatible finite elements for the fields. The spatial semi-discretization is derived from a semi-discretization of the Hamiltonian and the Poisson bracket. In the following, we summarize this semi-discretization as a starting point for the time-discretizations proposed in this paper.

2.1. The Vlasov–Maxwell system

A kinetic description of a collisionless plasma models a species s of particles with charge q_s and mass m_s by a distribution function f_s in phase-space that evolves according to the Vlasov equation

$$\frac{\partial f_s}{\partial t} + \mathbf{v} \cdot \nabla_{\mathbf{x}} f_s + \frac{q_s}{m_s} (\mathbf{E} + \mathbf{v} \times \mathbf{B}) \cdot \nabla_{\mathbf{v}} f_s = 0,$$

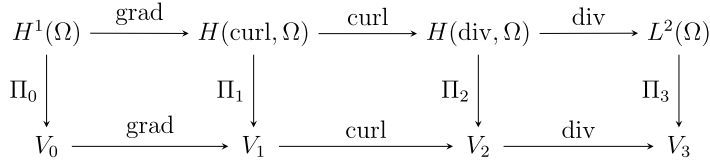


Fig. 1. Discrete de Rham complex for the spaces of electromagnetics.

where \mathbf{E} and \mathbf{B} denote the external and self-consistent electric and magnetic fields. The advection equation is coupled to Maxwell's equations for the self-consistent fields

$$\frac{\partial \mathbf{E}}{\partial t} - \nabla \times \mathbf{B} = -\mathbf{J}, \quad (1a)$$

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \times \mathbf{E} = 0, \quad (1b)$$

$$\nabla \cdot \mathbf{E} = \rho, \quad (1c)$$

$$\nabla \cdot \mathbf{B} = 0, \quad (1d)$$

where the charge density ρ and the current density \mathbf{J} are defined as velocity moments of the distribution functions

$$\rho = \sum_s q_s \int f_s \, d\mathbf{v}, \text{ and } \mathbf{J} = \sum_s q_s \int \mathbf{v} f_s \, d\mathbf{v}.$$

As for any hyperbolic conservation law, the solution of the Vlasov equation stays constant along the characteristic equations, which are defined as the following system of ordinary differential equations,

$$\frac{d\mathbf{x}}{dt} = \mathbf{v}, \quad \frac{d\mathbf{v}}{dt} = \frac{q_s}{m_s} (\mathbf{E}(\mathbf{x}, t) + \mathbf{v} \times \mathbf{B}(\mathbf{x}, t)). \quad (2)$$

The following Hamiltonian defines the total energy of the system

$$\mathcal{H} = \sum_s \frac{m_s}{2} \int |\mathbf{v}|^2 f_s(t, \mathbf{x}, \mathbf{v}) \, d\mathbf{x} \, d\mathbf{v} + \frac{1}{2} \int (|\mathbf{E}(t, \mathbf{x})|^2 + |\mathbf{B}(t, \mathbf{x})|^2) \, d\mathbf{x}. \quad (3)$$

2.2. Compatible finite element discretization of the field equations

The Maxwell equations themselves possess a rich structure of conservation properties. In particular, the fields \mathbf{E} and \mathbf{B} and potentials ϕ and \mathbf{A} involved in Maxwell's equations naturally fit into a de Rham complex (the continuous structure is represented on the first line in Fig. 1). We have $\phi \in H^1(\Omega)$, $\mathbf{A}, \mathbf{E} \in H(\text{curl}, \Omega)$, $\mathbf{B}, \mathbf{J} \in H(\text{div}, \Omega)$, and $\rho \in L^2(\Omega)$. The theory of finite element exterior calculus [17,18] explains how these properties can be preserved in a finite element discretization: The discrete spaces are chosen in such a way that the finite element approximation spaces of each of the quantities also form a de Rham complex (second line in Fig. 1) and that the projections Π_i from the continuous to the discrete spaces are such that the diagram shown in Fig. 1 is commuting. We use $\mathbf{E}_h \in V_1$ and $\mathbf{B}_h \in V_2$. Let $3N_1$ be the dimension of V_1 and denote by $\Lambda_i^{1,k}$, $i = 1, \dots, N_1$, $k = 1, 2, 3$, the vector-valued basis functions that are only non-zero in component k . Analogously, we denote the $3N_2$ basis functions of V_2 by $\Lambda_i^{2,k}$, $i = 1, \dots, N_2$, $k = 1, 2, 3$. Then, we get the following discrete representation of the fields

$$\mathbf{E}_h(\mathbf{x}) = \sum_{i=1}^{N_1} \sum_{k=1}^3 e_{i,k}(t) \Lambda_i^{1,k}(\mathbf{x}), \quad \mathbf{B}_h(\mathbf{x}) = \sum_{i=1}^{N_2} \sum_{k=1}^3 b_{i,k}(t) \Lambda_i^{2,k}(\mathbf{x}), \quad (4)$$

where $\mathbf{e} = (e_{i,k})$ and $\mathbf{b} = (b_{i,k})$ are the degrees of freedom in the semi-discretization. A compatible finite element discretization of Maxwell's equation can be obtained based on the ansatz (4) treating (1a) and (1c) in weak and (1b) and (1d) in strong form,

$$\mathbb{M}_1 \frac{d\mathbf{e}}{dt} - \mathbb{C}^\top \mathbb{M}_2 \mathbf{b} = -\mathbf{j},$$

$$\frac{d\mathbf{b}}{dt} + \mathbb{C} \mathbf{e} = 0,$$

$$\mathbb{G}^\top \mathbb{M}_1 \mathbf{e} = \mathbf{q},$$

$$\mathbb{D} \mathbf{b} = 0,$$

where $\mathbb{M}_{1/2}$ are the finite element mass matrices for the basis functions $\Lambda^{1/2}$, respectively, with elements $(\mathbb{M}_{1/2})_{(i_1,k_1),(i_2,k_2)} = \int \Lambda_{i_1}^{1/2,k_1}(\mathbf{x})^\top \Lambda_{i_2}^{1/2,k_2}(\mathbf{x}) d\mathbf{x}$, and \mathbb{D} , \mathbb{G} , and \mathbb{C} represent the discrete divergence, gradient, and curl operators which satisfy $\text{Im}\mathbb{G} \subset \text{Ker}\mathbb{C}$ and $\text{Im}\mathbb{C} \subset \text{Ker}\mathbb{D}$ as their continuous counterparts. There are various types of compatible finite element discretizations. In our work, we use spline finite elements of various order as proposed by Buffa, Rivas, Sangalli, & Vázquez [19]. The charge and current density are tested with the corresponding basis functions to obtain the vectors \mathbf{q} and \mathbf{j} for the right-hand side.

2.3. Coupling to the particles

A particle discretization represents the distribution function f_s by a number N_{ps} of particles which in turn are represented by the dynamic variables $(\mathbf{x}_a, \mathbf{v}_a)$, its coordinates in phase-space, as well as a weight w_a which is fixed over time in our formulation. The particle distribution is reconstructed by the Klimontovich distribution

$$f_{s,h}(\mathbf{x}, \mathbf{v}, t) = \sum_{a=1}^{N_{ps}} w_a \delta(\mathbf{x} - \mathbf{x}_a(t)) \delta(\mathbf{v} - \mathbf{v}_a(t)). \quad (5)$$

This representation is suitable in a finite-element discretization where the velocity moments are only needed in the weak form. In the resulting semi-discretized system, the dynamic variables $\mathbf{u}^\top = (\mathbf{X}^\top, \mathbf{V}^\top, \mathbf{e}^\top, \mathbf{b}^\top)$ are given by the phase space positions of the particles (for all species), $(\mathbf{X}^\top, \mathbf{V}^\top)$, and the degrees of freedom of the fields, $\mathbf{e} = (e_i)$ and $\mathbf{b} = (b_i)$. The integrals over the test functions times the charge and current density, that are needed for the right-hand side of the Ampère law and the electric Gauss law, can be computed as follows

$$\rho_{(i,k)} = \int \Lambda_i^{1,k}(\mathbf{x}) \rho(\mathbf{x}) d\mathbf{x} = \sum_s q_s \int \Lambda_i^{1,k}(\mathbf{x}) f_{s,h}(\mathbf{x}, \mathbf{v}, t) d\mathbf{x} d\mathbf{v} = \sum_s q_s \sum_{N_{ps}} w_a \Lambda_i^{1,k}(\mathbf{x}_a)$$

and analogously

$$\mathbf{j}_{(i,k)} = \int \Lambda_i^{2,k}(\mathbf{x}) \mathbf{J}(\mathbf{x}) d\mathbf{x} = \sum_s q_s \sum_{N_{ps}} \mathbf{v}_a w_a \Lambda_i^{2,k}(\mathbf{x}_a).$$

The phase-space coordinates evolve in time according to the characteristic equations (2) which are semi-discretized as

$$\frac{d\mathbf{x}_a}{dt} = \mathbf{v}_a, \quad \frac{d\mathbf{v}_a}{dt} = \frac{q_a}{m_a} (\mathbf{E}_h(\mathbf{x}_a) + \widehat{\mathbf{B}}_h(\mathbf{x}_a, t) \mathbf{v}_a),$$

where the matrix $\widehat{\mathbf{B}}_h(\mathbf{x}_a, t)$ represents the cross product of a vector with $\mathbf{B}_h(\mathbf{x}_a, t)$ such that $\widehat{\mathbf{B}}_h(\mathbf{x}_a) \mathbf{v}_a = \mathbf{v}_a \times \mathbf{B}_h(\mathbf{x}_a)$ and can be computed as

$$\widehat{\mathbf{B}}_h(\mathbf{x}_a, t) = \sum_{i=1}^{N_2} \begin{pmatrix} 0 & b_{i,3}(t) \Lambda_i^{2,3}(\mathbf{x}_a) & -b_{i,2}(t) \Lambda_i^{2,2}(\mathbf{x}_a) \\ -b_{i,3}(t) \Lambda_i^{2,3}(\mathbf{x}_a) & 0 & b_{i,1}(t) \Lambda_i^{2,1}(\mathbf{x}_a) \\ b_{i,2}(t) \Lambda_i^{2,2}(\mathbf{x}_a) & -b_{i,1}(t) \Lambda_i^{2,1}(\mathbf{x}_a) & 0 \end{pmatrix}.$$

In order to write the full semi-discrete system in matrix-vector form, we define the following matrices: The diagonal matrices $\mathbb{M}_q \in \mathbb{R}^{3N_p \times 3N_p}$ and $\mathbb{M}_m \in \mathbb{R}^{3N_p \times 3N_p}$ collect the particle charges $q_s w_a$ or particles masses $m_s w_a$ on the diagonal and $\Lambda^{1/2}(\mathbf{X}) \in \mathbb{R}^{3N_p \times 3N_{1/2}}$ denotes the matrix containing the value of all basis functions at each particle position. Further, we denote by $\mathbb{B}(\mathbf{X}, \mathbf{b}) \in \mathbb{R}^{(3N_p) \times (3N_p)}$ the matrix that consists of one 3×3 block $\widehat{\mathbf{B}}_h(\mathbf{x}_a, t)$ for each particle. This yields the following set of semi-discrete equations of motion

$$\dot{\mathbf{X}} = \mathbf{V}, \quad (6a)$$

$$\dot{\mathbf{V}} = \mathbb{M}_p^{-1} \mathbb{M}_q (\Lambda^1(\mathbf{X}) \mathbf{e} + \mathbb{B}(\mathbf{X}, \mathbf{b}) \mathbf{V}), \quad (6b)$$

$$\dot{\mathbf{e}} = \mathbb{M}_1^{-1} (\mathbb{C}^\top \mathbb{M}_2 \mathbf{b} - \Lambda^1(\mathbf{X})^\top \mathbb{M}_q \mathbf{V}), \quad (6c)$$

$$\dot{\mathbf{b}} = -\mathbb{C} \mathbf{e}. \quad (6d)$$

Moreover, the semi-discretization of the Hamiltonian reads

$$\hat{\mathcal{H}}(\mathbf{u}) = \frac{1}{2} \mathbf{V}^\top \mathbb{M}_p \mathbf{V} + \frac{1}{2} \mathbf{e}^\top \mathbb{M}_1 \mathbf{e} + \frac{1}{2} \mathbf{b}^\top \mathbb{M}_2 \mathbf{b}.$$

In Section 4 of [1], these semi-discrete equations are derived from a semi-discretization of the Poisson bracket for the Vlasov–Maxwell system. This derivation leads to a representation of the equations of motion as a Hamiltonian system of the form

$$\dot{\mathbf{u}} = \mathcal{J}(\mathbf{u}) \mathbf{D}_{\mathbf{u}} \hat{\mathcal{H}}(\mathbf{u}), \quad (7)$$

where the skew-symmetric so-called Poisson matrix $\mathcal{J}(\mathbf{u})$ is given as

$$\mathcal{J}(\mathbf{u}) = \begin{pmatrix} 0 & \mathbb{M}_p^{-1} & 0 & 0 \\ -\mathbb{M}_p^{-1} & \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{B}(\mathbf{X}, \mathbf{b}) \mathbb{M}_p^{-1} & \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{A}^1(\mathbf{X}) \mathbb{M}_1^{-1} & 0 \\ 0 & -\mathbb{M}_1^{-1} \mathbb{A}^1(\mathbf{X})^\top \mathbb{M}_q \mathbb{M}_p^{-1} & 0 & \mathbb{M}_1^{-1} \mathbb{C}^\top \\ 0 & 0 & -\mathbb{C} \mathbb{M}_2^{-1} & 0 \end{pmatrix}. \quad (8)$$

The total derivative of the Hamiltonian can be computed to be

$$\mathbf{D}_{\mathbf{u}} \hat{\mathcal{H}}(\mathbf{u}) = \left(0, (\mathbb{M}_p \mathbf{v})^\top, (\mathbb{M}_1 \mathbf{e})^\top, (\mathbb{M}_2 \mathbf{b})^\top \right)^\top.$$

In section 3, we will construct energy-conserving time discretizations based on this special form of the semi-discretization.

2.4. The explicit Hamiltonian splitting

Finally the system of equations (6) has to be discretized in time. In [1], the temporal discretization is based on the form (7) of the evolution equation combined with a splitting of the equations by splitting the Hamiltonian as

$$\hat{\mathcal{H}}_{p_k} = \frac{1}{2} m_s \sum_{a=1}^{N_p} w_p v_{k,a}^2, \quad k = 1, 2, 3, \quad \hat{\mathcal{H}}_{\mathbf{e}} = \frac{1}{2} \mathbf{e}^\top \mathbb{M}_1 \mathbf{e}, \quad \hat{\mathcal{H}}_{\mathbf{b}} = \frac{1}{2} \mathbf{b}^\top \mathbb{M}_2 \mathbf{b}.$$

This yields five sets of explicit equations (cf. [1, Sec. 5.1]). The splitting has originally been proposed in [3,20] as a Hamiltonian splitting and later been constructed from a fully discrete action principle in [21]. The discrete system then still conserves Gauss' law, however, only a modified energy. Xiao and coworkers [22] have also shown a local energy conservation law in the context of a Whitney form discretization of the fields. On the other hand, the equations become only explicit because the directions are decoupled by splitting the kinetic energy into the three subsystems of different directions. This separation is, however, limited to orthogonal grids. Otherwise, the kinetic energy can be kept together which yields a scheme that is implicit in the coordinates of the individual particles (cf. [23]). In this paper, we study an alternative approach for the time discretization based on the form (7) and the discrete gradient method.

3. Implicit time stepping based on the discrete gradient method

3.1. Discrete gradient time stepping

The discrete gradient method was proposed by McLachlan et al. [5] as a general method to construct energy conserving time stepping for conservative ordinary differential equations in skew-symmetric form, i.e. for a semi-discretization of the form

$$\dot{\mathbf{u}} = \mathcal{J}(\mathbf{u}) \cdot \mathbf{D}_{\mathbf{u}} \hat{\mathcal{H}}(\mathbf{u}) \quad \text{with} \quad \mathcal{J}(\mathbf{u})^\top = -\mathcal{J}(\mathbf{u}).$$

The discrete gradient $\bar{\nabla} \mathcal{H}(\mathbf{u}^m, \mathbf{u}^{m+1})$ for time step $[t_m, t_{m+1}]$ shall then satisfy

$$(\mathbf{u}^{m+1} - \mathbf{u}^m)^\top \bar{\nabla} \hat{\mathcal{H}}(\mathbf{u}^m, \mathbf{u}^{m+1}) = \hat{\mathcal{H}}(\mathbf{u}^{m+1}) - \hat{\mathcal{H}}(\mathbf{u}^m).$$

For any skew-symmetric approximation $\tilde{\mathcal{J}}(\mathbf{u}^m, \mathbf{u}^{m+1})$ of $\mathcal{J}(\mathbf{u})$, the following implicit scheme is then energy conserving,

$$\frac{\mathbf{u}^{m+1} - \mathbf{u}^m}{\Delta t} = \tilde{\mathcal{J}}(\mathbf{u}^m, \mathbf{u}^{m+1}) \bar{\nabla} \hat{\mathcal{H}}(\mathbf{u}^m, \mathbf{u}^{m+1}). \quad (9)$$

Energy conservation can be easily seen by the following calculation,

$$\begin{aligned} \hat{\mathcal{H}}(\mathbf{u}^{m+1}) - \hat{\mathcal{H}}(\mathbf{u}^m) &= (\mathbf{u}^{m+1} - \mathbf{u}^m)^\top \bar{\nabla} \hat{\mathcal{H}}(\mathbf{u}^m, \mathbf{u}^{m+1}) \\ &= \Delta t \bar{\nabla} \hat{\mathcal{H}}(\mathbf{u}^m, \mathbf{u}^{m+1})^\top \tilde{\mathcal{J}}^\top(\mathbf{u}^m, \mathbf{u}^{m+1}) \bar{\nabla} \hat{\mathcal{H}}(\mathbf{u}^m, \mathbf{u}^{m+1}) \\ &= -\Delta t \bar{\nabla} \hat{\mathcal{H}}(\mathbf{u}^m, \mathbf{u}^{m+1})^\top \tilde{\mathcal{J}}(\mathbf{u}^m, \mathbf{u}^{m+1}) \bar{\nabla} \hat{\mathcal{H}}(\mathbf{u}^m, \mathbf{u}^{m+1}) = 0. \end{aligned}$$

There is some freedom in the choice of the discrete gradient method. Since our energy is quadratic—and, hence, $\mathbf{D}_{\mathbf{u}} \hat{\mathcal{H}}(\mathbf{u})$ is linear—the midpoint rule is a second order discrete gradient

$$\bar{\nabla} \mathcal{H}(\mathbf{u}^m, \mathbf{u}^{m+1}) = \mathbf{D}_{\mathbf{u}} \hat{\mathcal{H}}(\mathbf{u})|_{\mathbf{u} = \frac{\mathbf{u}^m + \mathbf{u}^{m+1}}{2}}. \quad (10)$$

Note that the standard schemes for constructing discrete gradients like the average vector field method [6] or the Gonzalez discrete gradient [24] all boil down to the midpoint rule for quadratic energies. Hence, it remains to define a suitable discretization of the Poisson matrix.

3.2. Discrete gradients and antisymmetric splitting of the Poisson matrix

The discrete gradient method applied to the full Vlasov–Maxwell system yields a heavily non-linear system for the time discretization. In order to simplify the system, we may split the Poisson matrix into several antisymmetric submatrices. Then, we can apply the discrete gradient method separately to each subsystem and combine the solutions of the subsystems in a splitting method, e.g., a Lie splitting for first order, Strang splitting for second order, or composition methods for higher order (see [25] for splitting methods of various orders). Note that discrete gradient methods are typically of order two, whence a Strang splitting is the natural choice.

The Poisson matrix (8) can for instance be split into the following four antisymmetric matrices:

$$J_1 := \begin{pmatrix} 0 & \mathbb{M}_p^{-1} & 0 & 0 \\ -\mathbb{M}_p^{-1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \quad (11a)$$

$$J_2 := \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{B}(\mathbf{X}, \mathbf{b}) \mathbb{M}_p^{-1} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \quad (11b)$$

$$J_3 := \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbb{M}_1^{-1} \mathbb{C}^\top \\ 0 & 0 & -\mathbb{C} \mathbb{M}_1^{-1} & 0 \end{pmatrix}. \quad (11c)$$

$$J_4 := \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{A}^1(\mathbf{X}) \mathbb{M}_1^{-1} & 0 \\ 0 & -\mathbb{M}_1^{-1} \mathbb{A}^1(\mathbf{X})^\top \mathbb{M}_q \mathbb{M}_p^{-1} & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \quad (11d)$$

This yields the following four subsystems,

1. System 1: $\dot{\mathbf{X}} = \mathbf{V}$.
2. System 2: $\dot{\mathbf{V}} = \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{B}(\mathbf{X}, \mathbf{b}) \mathbf{V}$.
3. System 3: $\dot{\mathbf{e}} = \mathbb{M}_1^{-1} \mathbb{C}^\top \mathbb{M}_2 \mathbf{b}$, $\dot{\mathbf{b}} = -\mathbb{C} \mathbf{e}$.
4. System 4: $\dot{\mathbf{V}} = \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{A}^1(\mathbf{X}) \mathbf{e}$, $\dot{\mathbf{e}} = -\mathbb{M}_1^{-1} \mathbb{A}^1(\mathbf{X})^\top \mathbb{M}_q \mathbf{V}$.

We note that J_1 and J_3 are independent of \mathbf{u} . Moreover, J_2 only depends on \mathbf{X} and \mathbf{b} which remain unchanged in system 2. In the same way, J_4 is only dependent on \mathbf{X} which is again unchanged in this step. Hence, the Poisson matrix is constant in each of the substeps and there is one unique definition of \bar{J}_i , $i = 1, 2, 3, 4$. On the other hand, we have the option to solve subsystems 1 and 2 analytically instead of by a discrete gradient.

4. A semi-implicit discrete gradient discretization

In this section, we construct a time-stepping scheme based on the splitting introduced in (11). We will use the discrete gradient method to solve the subsystems 3 and 4 and solve subsystems 1 and 2 analytically. In the following, we will give the explicit formulas for each subsystem and discuss how to efficiently solve them.

4.1. Solution of system 1

In the first subsystem, only the position is updated and the right-hand side is independent of \mathbf{X} . The solution advancing the equation from time t_0 to t is hence given as

$$\mathbf{X}(t) = \mathbf{X}(t_0) + (t - t_0) \mathbf{V}.$$

Note that this yields the same discrete scheme in time as the midpoint-rule discrete gradient would yield.

4.2. Solution of system 2

The second system only updates the velocity and it decomposes into one equation for each particle, namely

$$\frac{d}{dt} \mathbf{v}_a = \frac{q_s}{m_s} \widehat{\mathbf{B}}_h(\mathbf{x}_a) \mathbf{v}_a = \frac{q_s}{m_s} \mathbf{v}_a \times \mathbf{B}_h(\mathbf{x}_a), \quad a = 1, \dots, N_p.$$

This is a rotation round the magnetic axis $\tilde{\mathbf{b}} = \frac{\mathbf{B}_h(\mathbf{x}_a)}{\|\mathbf{B}_h(\mathbf{x}_a)\|_2}$ (that is fixed over time in this subsystem) with an angle $\alpha = \Delta t \frac{q_s}{m_s} \|\mathbf{B}(\mathbf{x}_a)\|_2$ depending on the time step $\Delta t = t - t_0$. The solution is given by

$$\mathbf{v}_a(t) = R(\mathbf{B}_h(\mathbf{x}_a))\mathbf{v}_a(t_0),$$

where $R(\mathbf{B}_h(\mathbf{x}_a))$ is the rotation matrix

$$R(\mathbf{B}_h(\mathbf{x}_a)) = \begin{pmatrix} \tilde{b}_1^2 + (\tilde{b}_2^2 + \tilde{b}_3^2) \cos(\alpha) & \tilde{b}_3 \sin(\alpha) + \tilde{b}_2 \tilde{b}_1 (1 - \cos(\alpha)) & -\tilde{b}_2 \sin(\alpha) + \tilde{b}_3 \tilde{b}_1 (1 - \cos(\alpha)) \\ -\tilde{b}_3 \sin(\alpha) + \tilde{b}_2 \tilde{b}_1 (1 - \cos(\alpha)) & \tilde{b}_2^2 + (\tilde{b}_1^2 + \tilde{b}_3^2) \cos(\alpha) & \tilde{b}_1 \sin(\alpha) + \tilde{b}_3 \tilde{b}_2 (1 - \cos(\alpha)) \\ \tilde{b}_2 \sin(\alpha) + \tilde{b}_3 \tilde{b}_1 (1 - \cos(\alpha)) & -\tilde{b}_1 \sin(\alpha) + \tilde{b}_3 \tilde{b}_2 (1 - \cos(\alpha)) & \tilde{b}_3^2 + (\tilde{b}_2^2 + \tilde{b}_1^2) \cos(\alpha) \end{pmatrix}. \quad (12)$$

This system could also be discretized based on a discrete gradient. Again this yields a separate linear system of size 3×3 for each particle. Since the computational effort would hence be comparable, we prefer an exact solution of this substep.

4.3. Solution of system 3

For system 3, we apply the midpoint discrete gradient with the constant Poisson matrix. This results in the following system of linear equations for the unknown coefficients $\mathbf{e}^{m+1}, \mathbf{b}^{m+1}$,

$$\begin{pmatrix} \mathbb{M}_1 & -\frac{\Delta t}{2} \mathbb{C}^\top \mathbb{M}_2 \\ \frac{\Delta t}{2} \mathbb{M}_2 \mathbb{C} & \mathbb{M}_2 \end{pmatrix} \begin{pmatrix} \mathbf{e}^{m+1} \\ \mathbf{b}^{m+1} \end{pmatrix} = \begin{pmatrix} \mathbb{M}_1 & \frac{\Delta t}{2} \mathbb{C}^\top \mathbb{M}_2 \\ -\frac{\Delta t}{2} \mathbb{M}_2 \mathbb{C} & \mathbb{M}_2 \end{pmatrix} \begin{pmatrix} \mathbf{e}^m \\ \mathbf{b}^m \end{pmatrix}.$$

Note that this corresponds to the implicit midpoint rule applied to the vacuum Maxwell equation. The system can, of course, be solved in this form. For this, an iterative GMRES solver can for instance be used. For increasing degree of the spline basis, the system, however, gets rather ill-conditioned and we need a good preconditioner to solve the system. A simple preconditioner would be to split the equations into two explicit equations, i.e., to first solve for \mathbf{b}^{m+1} for given \mathbf{e}^m and then to solve for \mathbf{e}^{m+1} (or vice versa).

On the other hand, the equations for \mathbf{e}^{m+1} and \mathbf{b}^{m+1} can be decoupled using the Schur complement $S_3 = \mathbb{M}_1 + \frac{\Delta t^2}{4} \mathbb{C}^\top \mathbb{M}_2 \mathbb{C}$:

$$\begin{pmatrix} \mathbb{M}_1 & -\frac{\Delta t}{2} \mathbb{C}^\top \mathbb{M}_2 \\ \frac{\Delta t}{2} \mathbb{M}_2 \mathbb{C} & \mathbb{M}_2 \end{pmatrix}^{-1} = \begin{pmatrix} I & 0 \\ -\frac{\Delta t}{2} \mathbb{C} & I \end{pmatrix} \begin{pmatrix} S_3^{-1} & 0 \\ 0 & \mathbb{M}_2^{-1} \end{pmatrix} \begin{pmatrix} I & \frac{\Delta t}{2} \mathbb{C}^\top \\ 0 & I \end{pmatrix}.$$

With this expression for the matrix inverse, we get the following two equations:

$$\mathbf{e}^{m+1} = S_3^{-1} \left(\left(\mathbb{M}_1 - \frac{\Delta t^2}{4} \mathbb{C}^\top \mathbb{M}_2 \mathbb{C} \right) \mathbf{e}^m + \Delta t \mathbb{C}^\top \mathbb{M}_2 \mathbf{b}^m \right), \quad (13a)$$

$$\mathbf{b}^{m+1} = \mathbf{b}^m - \frac{\Delta t}{2} \mathbb{C} (\mathbf{e}^m + \mathbf{e}^{m+1}). \quad (13b)$$

Hence, we only need to solve the system $S_3 \mathbf{e}^{m+1} = \mathbf{f}$ for \mathbf{e}^{m+1} with given right-hand side \mathbf{f} and the magnetic field can then be updated by an explicit equation.

Let us now consider the implicit equation a bit more in detail. For this, we split the equation into three parts for each of the components of the field. The discrete mass matrix has block-diagonal form

$$\mathbb{M}_2 = \begin{pmatrix} M_{21} & 0 & 0 \\ 0 & M_{22} & 0 \\ 0 & 0 & M_{23} \end{pmatrix}$$

and the discrete curl matrix has the following block structure

$$\begin{pmatrix} 0 & -D_3 & D_2 \\ D_3 & 0 & -D_1 \\ -D_2 & D_1 & 0 \end{pmatrix},$$

where $D_k, k = 1, 2, 3$, denotes the derivative matrix along direction k . With this notation, we have the following expression for $\mathbb{C}^\top \mathbb{M}_2 \mathbb{C}$:

$$\mathbb{C}^\top \mathbb{M}_2 \mathbb{C} = \begin{pmatrix} D_3^\top M_{22} D_3 + D_2^\top M_{23} D_2 & -D_2^\top M_{23} D_1 & -D_3^\top M_{22} D_1 \\ -D_1^\top M_{23} D_2 & D_3^\top M_{21} D_3 + D_1^\top M_{23} D_1 & -D_3^\top M_{21} D_2 \\ -D_1^\top M_{22} D_3 & -D_2^\top M_{21} D_3 & D_2^\top M_{21} D_2 + D_1^\top M_{22} D_1 \end{pmatrix}.$$

Componentwise the equation therefore reads

$$\begin{aligned} & \left(M_{11} + \frac{\Delta t^2}{2} (D_3^\top M_{22} D_3 + D_2^\top M_{23} D_2) \right) \mathbf{e}_1 - \frac{\Delta t^2}{2} D_2^\top M_{23} D_1 \mathbf{e}_2 - \frac{\Delta t^2}{2} D_3^\top M_{22} D_1 \mathbf{e}_3 = \mathbf{f}_1, \\ & -\frac{\Delta t^2}{2} D_1^\top M_{23} D_2 \mathbf{e}_1 + \left(M_{12} + \frac{\Delta t^2}{2} (D_3^\top M_{21} D_3 + D_1^\top M_{23} D_1) \right) \mathbf{e}_2 - \frac{\Delta t^2}{2} D_3^\top M_{21} D_2 \mathbf{e}_3 = \mathbf{f}_2, \\ & -\frac{\Delta t^2}{2} D_1^\top M_{22} D_3 \mathbf{e}_1 - \frac{\Delta t^2}{2} D_2^\top M_{21} D_3 \mathbf{e}_2 + \left(M_{13} + \frac{\Delta t^2}{2} (D_2^\top M_{21} D_2 + D_1^\top M_{22} D_1) \right) \mathbf{e}_3 = \mathbf{f}_3. \end{aligned}$$

4.3.1. Direct inversion in Fourier space for periodic boundary conditions

In this paper, we consider the special case of a periodic box. Then, all the one-dimensional matrices are circulant and can thus be diagonalized by Fourier transformation. The derivative matrix is given by the circulant matrix

$$D = \frac{1}{\Delta x} \begin{pmatrix} 1 & & & -1 \\ -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{pmatrix}.$$

The eigenvalues of the one dimensional building blocks of the matrix $\mathbb{C}^\top M_2 \mathbb{C}$ of size $n \times n$ can therefore computed to be

- $D: \lambda_\ell^+ = \frac{1}{\Delta x} \left(1 - \exp\left(-\frac{2\pi i \ell}{n}\right) \right)$, $\ell = 0, \dots, n-1$.
- $D^\top: \lambda_\ell^- = \frac{1}{\Delta x} \left(1 - \exp\left(\frac{2\pi i \ell}{n}\right) \right)$, $\ell = 0, \dots, n-1$.
- M with row $c_p, \dots, c_0, \dots, c_p$ (p order of the spline): $\lambda_\ell^{(p)} = c_0 + \sum_{j=1}^p c_j 2 \cos\left(\frac{2\pi \ell j}{n}\right)$, $\ell = 0, \dots, n-1$.

After Fourier transformation, we end up with a 3×3 system for each Fourier mode which can be solved analytically.

The direct solution based on Fourier transforms is very efficient, since it only requires a Fourier transform but no iterative solver. A similar construction is possible for other basis functions on periodic Cartesian grids. For more complex domains, an iterative solution of the system is necessary. For spline finite elements of higher order, already the mass matrices are ill-conditioned and a good preconditioner is necessary. In [26], the Fourier solver for the Cartesian case has been used successfully as a preconditioner for the linear systems for the curvilinear case.

4.4. Solution of system 4

Again, we apply the midpoint rule discrete gradient and use that the Poisson matrix is constant in this substep. Then, the discrete gradient discretization yields the following linear system for the coefficients $(\mathbf{v}^{m+1}, \mathbf{e}^{m+1})$

$$\begin{pmatrix} I & -\frac{\Delta t}{2} \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{A}^1(\mathbf{X}) \\ \frac{\Delta t}{2} \mathbb{A}^1(\mathbf{X})^\top \mathbb{M}_q & \mathbb{M}_1 \end{pmatrix} \begin{pmatrix} \mathbf{v}^{m+1} \\ \mathbf{e}^{m+1} \end{pmatrix} = \begin{pmatrix} I & \frac{\Delta t}{2} \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{A}^1(\mathbf{X}) \\ -\frac{\Delta t}{2} \mathbb{A}^1(\mathbf{X})^\top \mathbb{M}_q & \mathbb{M}_1 \end{pmatrix} \begin{pmatrix} \mathbf{v}^m \\ \mathbf{e}^m \end{pmatrix}.$$

Defining the Schur complement $S_4 = \mathbb{M}_1 + \frac{\Delta t^2}{4} \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{A}^1(\mathbf{X})^\top \mathbb{M}_q \mathbb{A}^1(\mathbf{X})$, we get the following expression for the inverse of the left-hand-side matrix

$$\begin{pmatrix} I & -\frac{\Delta t}{2} \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{A}^1(\mathbf{X}) \\ \frac{\Delta t}{2} \mathbb{A}^1(\mathbf{X})^\top \mathbb{M}_q & \mathbb{M}_1 \end{pmatrix}^{-1} = \begin{pmatrix} I & \frac{\Delta t}{2} \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{A}^1(\mathbf{X}) \\ 0 & I \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & S_4^{-1} \end{pmatrix} \begin{pmatrix} I & 0 \\ -\frac{\Delta t}{2} \mathbb{A}^1(\mathbf{X})^\top \mathbb{M}_q & I \end{pmatrix}.$$

Hence, the system can be solved in three steps:

1. $\mathbf{v}^* = \mathbf{v}^m + \frac{\Delta t}{2} \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{A}^1(\mathbf{X}) \mathbf{e}^m$,
 $\mathbf{e}^* = \left(\mathbb{M}_1 - \frac{\Delta t^2}{4} \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{A}^1(\mathbf{X})^\top \mathbb{M}_p \mathbb{A}^1(\mathbf{X}) \right) \mathbf{e}^m - \Delta t \mathbb{A}^1(\mathbf{X})^\top \mathbb{M}_q \mathbf{v}^m$.
2. $\mathbf{e}^{m+1} = S_4^{-1} \mathbf{e}^*$.
3. $\mathbf{v}^{m+1} = \mathbf{v}^* + \frac{\Delta t}{2} \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{A}^1(\mathbf{X}) \mathbf{e}^{m+1}$.

Note that the implicit part (step 2) is reduced to the field equations and the particle equations can be solved explicitly. Moreover, we note that all appearing matrices are block-diagonal in the three pairs $(\mathbf{v}_k, \mathbf{e}_k)$, $k = 1, 2, 3$, where \mathbf{v}_k collects the k th component of the velocity of all particles and \mathbf{e}_k the degrees of freedom of the electric field that belong to the basis functions $\mathbf{A}_i^{1,k}$, $i = 1, \dots, N_1$. This block-diagonal structure is due to the fact that all basis functions have only one non-zero component. As a consequence, the implicit equation $\mathbf{e}^{m+1} = S_4^{-1} \mathbf{e}^*$ decomposes to three systems of size $N_1 \times N_1$.

4.4.1. Preconditioning the linear solver

Since the matrix S_4 is a symmetric matrix, we can solve the system in step 2 using the conjugate gradient method. However, also S_4 is increasingly ill-conditioned for higher degree of the splines so that we need a good preconditioner. Let us consider the matrix $N := \mathbb{A}^1(\mathbf{X})^\top \mathbb{M}_q \mathbb{A}^1(\mathbf{X})$ with elements

$$N_{(i,k_1),(j,k_2)} = q \sum_a w_a \mathbb{A}_i^{1,k_1}(\mathbf{x}_a)^\top \mathbb{A}_j^{1,k_2}(\mathbf{x}_a).$$

We note that this is a Monte Carlo approximation of the integral $\int_\Omega \rho(\mathbf{x}) \mathbb{A}_i^{1,k_1}(\mathbf{x})^\top \mathbb{A}_j^{1,k_2}(\mathbf{x}) d\mathbf{x}$ in a multispecies simulation. In a simulation of electrons with a neutralizing ion background, N only contains the electron distribution function, and hence the elements of N approximate the integral $\int_\Omega (1 - \rho(\mathbf{x})) \mathbb{A}_i^{1,k_1}(\mathbf{x})^\top \mathbb{A}_j^{1,k_2}(\mathbf{x}) d\mathbf{x}$. In case $\rho(\mathbf{x}) = 1$, this would be a Monte Carlo approximation of the mass matrix in the one-species case. We therefore refer to N as *particle sampled mass matrix*.

Oftentimes $\rho(\mathbf{x}) \ll 1$ holds true so that N is very small in a multispecies simulation or close to \mathbb{M}_1 in an electron simulation. Therefore, \mathbb{M}_1 is a good preconditioner for S_4 in a multispecies simulation and $(1 + \frac{\Delta t^2}{4} \frac{q^2}{m}) \mathbb{M}_1$ in an electron simulation. For the periodic box, the mass matrix is circulant and can be inverted in Fourier space (cf. Sec. 4.3). Clearly, this preconditioner works the better the smaller Δt , the smaller ρ , and the more particles per cell are used (since the latter improves the Monte-Carlo approximation).

Remark 1. Instead of using \mathbb{M}_1 or $(1 + \frac{\Delta t^2}{4} \frac{q^2}{m}) \mathbb{M}_1$, respectively, as a preconditioner, we could also replace the Schur complement by \mathbb{M}_1 or $(1 + \frac{\Delta t^2}{4} \frac{q^2}{m}) \mathbb{M}_1$. Since assembling N is the step of highest computational complexity in this propagator, this yields a considerable saving in compute time. On the other hand, this approximation is not energy-conserving so that we would loose the desirable property of our algorithm.

4.5. Summary of the semi-implicit discrete gradient time stepping

The proposed semi-implicit and energy-conserving scheme, called semi-implicit or energy-conserving discrete gradient scheme in the following, is composed of the four operators resulting from the splitting of the Poisson matrix (11). For the time-stepping, the following four operators are combined:

1. Operator 1: $\mathbf{X}(t) = \mathbf{X}(t_0) + (t - t_0)\mathbf{V}$.
2. Operator 2: $\mathbf{v}_a(t) = R(\mathbf{B}_h(\mathbf{x}_a))\mathbf{v}_a(t_0)$ with the rotation matrix defined by eqn. (12).
3. Operator 3:

$$\mathbf{e}(t) = S_3^{-1} \left(\left(\mathbb{M}_1 - \frac{\Delta t^2}{4} \mathbb{C}^\top \mathbb{M}_2 \mathbb{C} \right) \mathbf{e}(t_0) + \Delta t \mathbb{C}^\top \mathbb{M}_2 \mathbf{b}(t_0) \right), \quad (14)$$

$$\mathbf{b}(t) = \mathbf{b}(t_0) - \frac{\Delta t}{2} \mathbb{C} (\mathbf{e}(t_0) + \mathbf{e}(t)). \quad (15)$$

4. Operator 4:

- (a) $\mathbf{V}^* = \mathbf{V}(t_0) + \frac{\Delta t}{2} \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{A}^1(\mathbf{X}) \mathbf{e}(t_0)$,
 $\mathbf{e}^* = \left(\mathbb{M}_1 - \frac{\Delta t^2}{4} \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{A}^1(\mathbf{X})^\top \mathbb{M}_p \mathbb{A}^1(\mathbf{X}) \right) \mathbf{e}(t_0) - \Delta t \mathbb{A}^1(\mathbf{X})^\top \mathbb{M}_q \mathbf{V}(t_0)$.
- (b) $\mathbf{e}(t) = S_4^{-1} \mathbf{e}^*$, $S_4 = \mathbb{M}_1 + \frac{\Delta t^2}{4} \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{A}^1(\mathbf{X})^\top \mathbb{M}_q \mathbb{A}^1(\mathbf{X})$.
- (c) $\mathbf{V}(t) = \mathbf{V}^* + \frac{\Delta t}{2} \mathbb{M}_p^{-1} \mathbb{M}_q \mathbb{A}^1(\mathbf{X}) \mathbf{e}(t)$.

The four operators can now be combined in various ways to build the full time step. The first order Lie splitting is build upon full time steps of each of the operators one after the other. Second order can be obtained when combining two Lie splitting steps with opposite ordering of the operators of half a time step each. In this case, the last operator of the first Lie step and the first operator of the second Lie step are the same so that we can instead place a full time step of one operator in the middle. Clearly, for a large number of particles per cell, the fourth operator is the most expensive one. Therefore, the complexity of the algorithm is reduced when placing this operator in the middle and calling it only once.

A Strang splitting of the following form yields the shortest run time: half time step of operator 3, half time step of operator 1, half time step of operator 2, full time step with operator 4, half time step with operator 2, half time step with operator 1, half time step with operator 3. With this ordering we only have to apply the most expensive operator 4 once and we can merge the updates of operator 1 and 2 (that do not touch the fields) into the particle loops of operator 4. The number of loops over all particles is minimized in order to maximize the arithmetic intensity of the algorithm. The computational complexity is then dominated by the assembling of the particle sampled mass matrix. This results in the algorithm outlined in Algorithm 1. We note that the ordering of operator 1 and 2 could also be exchanged without changing the structure of the loops.

```

for  $m = 1, \dots$  do
   $\mathbf{e}^* = S_3^{-1} \left( \left( \mathbb{M}_1 - \frac{\Delta t^2}{16} \mathbb{C}^\top \mathbb{M}_2 \mathbb{C} \right) \mathbf{e}^m + \Delta t \mathbb{C}^\top \mathbb{M}_2 \mathbf{b}^m \right);$ 
   $\mathbf{b}^* = \mathbf{b}^m - \frac{\Delta t}{4} \mathbb{C} (\mathbf{e}^m + \mathbf{e}^*);$ 
   $\mathbf{j} = 0;$ 
   $N = 0;$ 
  for  $a = 1, \dots, N_p$  do
     $\mathbf{x}_a^* = \mathbf{x}_a^m + \frac{\Delta t}{2} \mathbf{v}_a^m;$ 
     $\mathbf{v}_a^* = R(\mathbf{b}^*) \mathbf{v}_a^m;$ 
     $\mathbf{v}_a^{**} = \mathbf{v}^* + \frac{\Delta t}{2} \frac{q_s}{m_s} \mathbb{A}^1(\mathbf{x}_a^*) \mathbf{e}^*;$ 
     $\mathbf{j} = \mathbf{j} + \Delta t q_s w_a \mathbb{A}^1(\mathbf{x}_a^*) \mathbf{v}_a^*;$ 
     $N = N + \frac{\Delta t^2}{4} q_s w_a \mathbb{A}^1(\mathbf{x}_a^*)^\top \mathbb{A}^1(\mathbf{x}_a^*);$ 
  end
   $\mathbf{e}^{**} = (\mathbb{M}_1 - N) \mathbf{e}^* - \mathbf{j};$ 
  for  $a = 1, \dots, N_p$  do
     $\mathbf{v}_a^{***} = \mathbf{v}^{**} + \frac{\Delta t}{2} \frac{q_s}{m_s} \mathbb{A}^1(\mathbf{x}_a^*) \mathbf{e}^{**};$ 
     $\mathbf{v}_a^{m+1} = R(\mathbf{b}^*) \mathbf{v}_a^{***};$ 
     $\mathbf{x}_a^{m+1} = \mathbf{x}_a^* + \frac{\Delta t}{2} \mathbf{v}_a^{m+1};$ 
  end
   $\mathbf{e}^{m+1} = S_3^{-1} \left( \left( \mathbb{M}_1 - \frac{\Delta t^2}{16} \mathbb{C}^\top \mathbb{M}_2 \mathbb{C} \right) \mathbf{e}^{**} + \Delta t \mathbb{C}^\top \mathbb{M}_2 \mathbf{b}^* \right);$ 
   $\mathbf{b}^{m+1} = \mathbf{b}^* - \frac{\Delta t}{4} \mathbb{C} (\mathbf{e}^{**} + \mathbf{e}^{m+1});$ 
end

```

Algorithm 1: Second order semi-implicit discrete gradient scheme.

On the other hand, we can also ask which ordering gives best accuracy. In our examples, we found that better accuracy is sometimes achieved when placing operator 3 after the operators 1 and 2. However, in this case we need to traverse all particles two times more per time step (or one time if we fuse this step between two time steps in a “first same as last” procedure).

The computational overhead compared to the explicit Hamiltonian splitting is rather limited, namely we need to solve the curl-part of Maxwell's equation implicitly (which is, however, a small problem compared to the particle loops) and we need the particle sampled mass matrix. On the other hand, no integrals need to be computed for the deposition of the current.

The method (with any ordering of the operators) is energy-conserving, since the single operators are solved either exactly or based on an energy-conserving discrete-gradient discretization. On the other hand, the Casimir invariants can be destroyed by the splitting of the Poisson matrix. In particular, Gauss' law is not conserved over time with this time stepping method as will be discussed in the next section.

Compared to the explicit Hamiltonian splitting method, this method allows for larger time steps. In particular, the time step is not restricted by either light waves in the Maxwell's equation or Langmuir waves as the stability analysis in the Appendix shows.

5. Conservative implicit discrete gradient method

In the previous section, we have derived an energy-conserving semi-implicit time propagator. However, it was shown in [1, Sec. 4.6] that the semi-discrete equations of motion of the GEMPIC framework (cf. (6)) satisfy Gauss' law which the semi-implicit discrete gradient scheme does not as we will show in the following. In this section, we derive an alternative discrete gradient method that—in addition to energy—also conserves Gauss' law.

5.1. Conservation of Gauss' law

In order to satisfy Gauss' law, it is important how the current is accumulated in Ampère's law,

$$\mathbb{M}_1 \mathbf{e}^{m+1} = \mathbb{M}_1 \mathbf{e}^m - \int_{t_m}^{t_{m+1}} \mathbb{A}^1(\mathbf{X}(\tau))^\top \mathbb{M}_q \mathbf{V}(\tau) d\tau. \quad (16)$$

Applying \mathbb{G}^\top to (16) yields,

$$\mathbb{G}^\top \mathbb{M}_1 \mathbf{e}^{m+1} = \mathbb{G}^\top \mathbb{M}_1 \mathbf{e}^m - \int_{t_m}^{t_{m+1}} \mathbb{G}^\top \mathbb{A}^1(\mathbf{X}(\tau))^\top \mathbb{M}_q \mathbf{V}(\tau) d\tau.$$

If $\frac{d\mathbf{X}(\tau)}{d\tau} = \mathbf{V}(\tau)$ holds true, we can use the chain rule to identify the integrand as a time derivative, namely

$$\mathbb{G}^\top \mathbb{A}^1(\mathbf{X}(\tau))^\top \mathbb{M}_q \mathbf{V}(\tau) = \frac{d}{d\tau} \mathbb{A}^0(\mathbf{X}(\tau))^\top \mathbb{M}_q \mathbb{1}_{N_p}.$$

Hence, the integral over t can be evaluated as

$$\begin{aligned} \mathbb{G}^\top \mathbb{M}_1 \mathbf{e}^{m+1} &= \mathbb{G}^\top \mathbb{M}_1 \mathbf{e}^m - \int_{t_m}^{t_{m+1}} \mathbb{G}^\top \mathbb{A}^1(\mathbf{X}(\tau))^\top \mathbb{M}_q \mathbf{V}(\tau) d\tau \\ &= \mathbb{G}^\top \mathbb{M}_1 \mathbf{e}^m - \mathbb{A}^0(\mathbf{X}(t_{m+1}))^\top \mathbb{M}_q \mathbb{1}_{N_p} + \mathbb{A}^0(\mathbf{X}(t_m))^\top \mathbb{M}_q \mathbb{1}_{N_p}. \end{aligned}$$

This means that, if the discrete version of Gauss' law $\mathbb{G}^\top \mathbb{M}_1 \mathbf{e}^m = -\mathbb{A}^0(\mathbf{X}(t_m))^\top \mathbb{M}_q \mathbb{1}_{N_p}$ holds at time t_m , it also holds at time t_{m+1} . In the semi-implicit method, the current is accumulated in subsystem 4, while the particle trajectory is updated in subsystem 1. Hence, the time derivative of the particle trajectories is zero in subsystem 4 and we would need to combine these two subsystems in order to achieve conservation of Gauss' law.

We note that the curl-part of Ampère's law does not change the divergence of the electric field, since the discrete gradient and curl operators respect the relation $\mathbb{C}\mathbb{G} = 0$: Multiplying the curl-part of Ampère's law by \mathbb{G}^\top , we get

$$\mathbb{G}^\top \mathbb{M}_1 \dot{\mathbf{e}} - \frac{\Delta t}{2} \mathbb{G}^\top \mathbb{C}^\top \mathbb{M}_2 \mathbf{b} = 0,$$

with the second term being zero, since $\mathbb{G}^\top \mathbb{C}^\top = (\mathbb{C}\mathbb{G})^\top = 0$. Hence the divergence of the electric field does not change with time. In particular, the divergence is not changed in the solution of System 3, where the electric field is updated by equation (13a). Multiplying the equation by \mathbb{G}^\top yields

$$\mathbb{G}^\top \left(\mathbb{M}_1 + \frac{\Delta t^2}{4} \mathbb{C}^\top \mathbb{M}_2 \mathbb{C} \right) \mathbf{e}^{m+1} = \mathbb{G}^\top \left(\mathbb{M}_1 - \frac{\Delta t^2}{4} \mathbb{C}^\top \mathbb{M}_2 \mathbb{C}^\top \right) \mathbf{e}^m + \Delta t \mathbb{G}^\top \mathbb{C}^\top \mathbb{M}_2 \mathbf{b}^m.$$

Again exploiting $\mathbb{G}^\top \mathbb{C}^\top = (\mathbb{C}\mathbb{G})^\top = 0$ yields

$$\mathbb{G}^\top \mathbb{M}_1 \mathbf{e}^{m+1} = \mathbb{G}^\top \mathbb{M}_1 \mathbf{e}^m.$$

5.2. Alternative discrete gradient that conserves Gauss' law

In this section, we construct a discrete gradient solution of the combined subsystem

$$\dot{\mathbf{u}} = (J_1 + J_4) \mathbb{D}_{\mathbf{u}} \hat{\mathcal{H}}(\mathbf{u}), \quad (17)$$

that defines the approximation $\overline{J_1 + J_4}$ such that Gauss' law is satisfied.

The mid-point discrete gradient defines

$$\frac{\mathbf{X}^{m+1} - \mathbf{X}^m}{\Delta t} = \frac{\mathbf{V}^m + \mathbf{V}^{m+1}}{2}.$$

Then, we can construct the time-continuous trajectory

$$\mathbf{X}(\tau) = ((t_{m+1} - \tau)\mathbf{X}^m + (\tau - t_m)\mathbf{X}^{m+1})/\Delta t, \tau \in [t_m, t_{m+1}]$$

with constant time derivative

$$\frac{d\mathbf{X}}{d\tau} = \frac{\mathbf{X}^{m+1} - \mathbf{X}^m}{\Delta t} = \frac{\mathbf{V}^m + \mathbf{V}^{m+1}}{2}.$$

With this definition, we can construct the following discrete gradient scheme

$$\frac{\mathbf{X}^{m+1} - \mathbf{X}^m}{\Delta t} = \frac{\mathbf{V}^m + \mathbf{V}^{m+1}}{2}, \quad (18a)$$

$$\frac{\mathbf{V}^{m+1} - \mathbf{V}^m}{\Delta t} = \mathbb{M}_p^{-1} \mathbb{M}_q \frac{1}{\Delta t} \int_{t_m}^{t_{m+1}} \mathbb{A}^1(\mathbf{X}(\tau)) d\tau \left(\frac{\mathbf{e}^m + \mathbf{e}^{m+1}}{2} \right), \quad (18b)$$

$$\frac{\mathbf{e}^{m+1} - \mathbf{e}^m}{\Delta t} = -\mathbb{M}_1^{-1} \frac{1}{\Delta t} \int_{t_m}^{t_{m+1}} \mathbb{A}^1(\mathbf{X}(\tau))^\top d\tau \mathbb{M}_q \left(\frac{\mathbf{V}^m + \mathbf{V}^{m+1}}{2} \right), \quad (18c)$$

which corresponds to the mid-point gradient and a Poisson matrix

$$\overline{J_1 + J_4} := \begin{pmatrix} 0 & \mathbb{M}_p^{-1} & 0 & 0 \\ -\mathbb{M}_p^{-1} & 0 & \mathbb{M}_p^{-1} \mathbb{M}_q \int_{t_m}^{t_{m+1}} \mathbb{A}^1(\mathbf{X}(\tau)) \mathbb{M}_1^{-1} d\tau & 0 \\ 0 & -\mathbb{M}_1^{-1} \int_{t_m}^{t_{m+1}} \mathbb{A}^1(\mathbf{X}(\tau))^\top \mathbb{M}_q \mathbb{M}_p^{-1} d\tau & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \quad (19)$$

Since $\overline{J_1 + J_4}$ is antisymmetric, energy is conserved. Moreover, the scheme conserves Gauss' law by the calculations in the previous section, since the term $\frac{\mathbf{v}^m + \mathbf{v}^{m+1}}{2}$ on the right-hand side of (18c) corresponds to the time derivative of $\mathbf{X}(\tau)$.

5.2.1. Summary of the conservative discrete gradient time stepping

Scheme (18) can be combined with the analytic solution of System 2 and the discrete gradient solution of System 3 (as discussed in Section 4.3) to an implicit propagator that conserves both energy and Gauss' law.

System (18) is now not only linearly implicit (as it was the case when the \mathbf{X} propagation was separated from \mathbf{V} and \mathbf{e}) so that we need to use an iterative algorithm to solve this nonlinear equation. In our implementation, we use the semi-implicit scheme to produce a good starting point for the nonlinear iteration and then continue to improve the approximation with Picard iterations until a predefined tolerance is met. Note that this scheme also comes with a memory overhead, since we need to store a second particle position and velocity for all particles during the nonlinear iteration. Algorithm 2 summarizes the proposed conservative discrete gradient scheme in a second order Strang splitting with minimal number of particle loops. However, we note that we need one particle loop per Picard iteration in this case. Reordering operator 2 and 3 and spending one (or two) extra particle loops per time step will only marginally increase the computing time for each time step and can therefore easily pay-off if accuracy is increased.

In the same way as for the semi-implicit scheme, a von Neumann analysis shows that this scheme has no stability restrictions by light waves or Langmuir waves.

5.2.2. Exact numerical evaluation of the integrals

Note that we have to evaluate the integrals of the form $\int_0^1 \Lambda_j^{1,k}(\mathbf{X}(\tau)) d\tau$, where $\Lambda_j^{1,k}$ is a tensor product of splines of a certain degree in each of the three variables. As a function of τ the integrand is then a polynomial of degree $p_1 p_2 p_3$ (where p_k denotes the degree of the spline in each direction $k = 1, 2, 3$) locally in each cell of the three dimensional domain. In order to solve the integral exactly (which is necessary to conserve Gauss' law), we can use Gauss-Legendre quadrature with $\lceil \frac{p_1 p_2 p_3 + 1}{2} \rceil$ points separately in each cell crossed by the line integral. Note that, while the degree of the quadrature is generally quite high, we only need to perform one-dimensional quadratures. As an example, if we use splines of degree 3 for the 0-form, i.e., we have two times degree 3 and once degree 2 in the 1-form, we need to use a 10-point-quadrature. For splines of degree 2, on the other hand, we only need 3 points, so the complexity rapidly increases with the degree of the splines.

5.3. Substepping

For the accurate simulation of low frequency phenomena, where the fields are slowly varying in time but on the grid scale in space, particle substepping is another useful feature that can allow for larger global time steps. In this subsection, we therefore adapt the method proposed by Chen, Chacón, & Barnes [14] to our context. It is our goal to combine a few small time steps in the particle push with a larger global time step for the update of the electric field in the conservative discretization of the combined subsystem 1 and 4. The algorithm proposed below preserves both energy and Gauss' law, however, it cannot be classified as a discrete gradient method as defined in (9).

Let us split the time step into N_{sub} subintervals $[t_\nu, \tau_{\nu+1}]$, $\nu = 0, \dots, N_{sub}$, of not necessarily identical length $\Delta\tau_\nu = \tau_{\nu+1} - \tau_\nu$ with $\tau_0 = t_m$ and $\tau_{N_{sub}} = t_{m+1}$: $\Delta t = t_{m+1} - t_m = \sum_{\nu=0}^{N_{sub}-1} (\tau_{\nu+1} - \tau_\nu)$. Then, keeping the electric field constant over all the substeps, we push the particles according to $\mathbf{X}_m^0 = \mathbf{X}^m$, $\mathbf{V}_m^0 = \mathbf{V}^m$ and for $\nu = 0$ to $N_{sub} - 1$

$$\frac{\mathbf{X}_m^{\nu+1} - \mathbf{X}_m^\nu}{\Delta\tau_\nu} = \frac{\mathbf{V}_m^\nu + \mathbf{V}_m^{\nu+1}}{2}, \quad (20a)$$

$$\frac{\mathbf{V}_m^{\nu+1} - \mathbf{V}_m^\nu}{\Delta\tau_\nu} = \mathbb{M}_p^{-1} \mathbb{M}_q \frac{1}{\Delta\tau_\nu} \int_{\tau_\nu}^{\tau_{\nu+1}} \mathbb{A}^1(\mathbf{X}_m(\tau)) d\tau \left(\frac{\mathbf{e}^m + \mathbf{e}^{m+1}}{2} \right), \quad (20b)$$

with $\mathbf{X}_m(\tau) = ((\tau_{\nu+1} - \tau)\mathbf{X}_m^\nu + (\tau - \tau_\nu)\mathbf{X}_m^{\nu+1})/\Delta\tau_\nu$. The current for the update of the electric field then also needs to be updated as a sum over the contributions of each substep following the discrete gradient formulation in (18c) by

$$\frac{\mathbf{e}^{m+1} - \mathbf{e}^m}{\Delta t} = -\mathbb{M}_1^{-1} \frac{1}{\Delta t} \sum_{\nu=0}^{N_{sub}-1} \left(\int_{\tau_\nu}^{\tau_{\nu+1}} \mathbb{A}^1(\mathbf{X}_m(\tau))^\top d\tau \right) \mathbb{M}_q \left(\frac{\mathbf{V}_m^\nu + \mathbf{V}_m^{\nu+1}}{2} \right). \quad (21)$$

```

for  $m = 1, \dots$  do
   $\mathbf{e}^* = S_3^{-1} \left( (\mathbb{M}_1 - \frac{\Delta t^2}{16} \mathbb{C}^\top \mathbb{M}_2 \mathbb{C}) \mathbf{e}^m + \Delta t \mathbb{C}^\top \mathbb{M}_2 \mathbf{b}^m \right)$ ;
   $\mathbf{b}^* = \mathbf{b}^m - \frac{\Delta t}{4} \mathbb{C} (\mathbf{e}^m + \mathbf{e}^*)$ ;
   $\mathbf{j} = 0$ ;
   $N = 0$ ;
  for  $a = 1, \dots, N_p$  do
     $\mathbf{v}_a^* = R(\mathbf{b}^*) \mathbf{v}_a^m$ ;
     $\mathbf{x}_a^{\text{iter}} = \mathbf{x}_a^m + \frac{\Delta t}{2} \mathbf{v}_a^*$ ;
     $\mathbf{v}_a^{**} = \mathbf{v}^* + \frac{\Delta t}{2} \frac{q_s}{m_s} \mathbb{A}^1(\mathbf{x}_a^{\text{iter}}) \mathbf{e}^*$ ;
     $\mathbf{j} = \mathbf{j} + \Delta t q_s w_a \mathbb{A}^1(\mathbf{x}_a^{\text{iter}}) \mathbf{v}_a$ ;
     $N = N + \frac{\Delta t^2}{4} q_s w_a \mathbb{A}^1(\mathbf{x}_a^{\text{iter}})^\top \mathbb{A}^1(\mathbf{x}_a^{\text{iter}})$ ;
  end
   $\mathbf{e}^{\text{iter}} = (\mathbb{M}_1 - N) \mathbf{e}^* - \mathbf{j}$ ;
  for  $a = 1, \dots, N_p$  do
     $\mathbf{v}_a^{\text{iter}} = \mathbf{v}_a^{**} + \frac{\Delta t}{2} \frac{q_s}{m_s} \mathbb{A}^1(\mathbf{x}_a^{\text{iter}}) \mathbf{e}^{\text{iter}}$ ;
     $\mathbf{x}_a^{\text{iter}} = \mathbf{x}_a^{\text{iter}} + \frac{\Delta t}{2} \mathbf{v}_a^{\text{iter}}$ ;
  end
   $\text{residual} = \text{tolerance} + 1$ ;
  while  $\text{residual} < \text{tolerance}$  do
     $\mathbf{j} = 0$ ;
    for  $a = 1, \dots, N_p$  do
       $\tilde{\mathbf{v}}_a = \frac{1}{2} (\mathbf{v}_a^* + \mathbf{v}_a^{\text{iter}})$ ;
       $\mathbf{x}_a^{\text{iter}} = \mathbf{x}_a^* + \Delta t \tilde{\mathbf{v}}_a$ ;
       $\mathbf{j} = \mathbf{j} + \Delta t q_s w_a \int_0^1 \mathbb{A}^1(\mathbf{x}_a^* + \tau \tilde{\mathbf{v}}_a) d\tau \mathbf{v}_a$ ;
       $\mathbf{v}_a^{\text{iter}} = \mathbf{v}^* + \frac{\Delta t}{2} \int_0^1 \frac{q_s}{m_s} \mathbb{A}^1(\mathbf{x}_a^* + \tau \tilde{\mathbf{v}}_a) d\tau \mathbf{e}^{**}$ ;
    end
     $\mathbf{e}^{\text{old}} = \mathbf{e}^{\text{iter}}$ ;
     $\mathbf{e}^{\text{iter}} = \mathbf{e}^* - M^{-1} \mathbf{j}$ ;
     $\text{residual} = \|\mathbf{e}^{\text{iter}} - \mathbf{e}^{\text{old}}\|_2$ ;
  end
  for  $a = 1, \dots, N_p$  do
     $\mathbf{v}_a^{m+1} = R(\mathbf{b}^*) \mathbf{v}_a^{\text{iter}}$ ;
     $\mathbf{x}_a^{m+1} = \mathbf{x}_a^{\text{iter}} + \frac{\Delta t}{2} \mathbf{v}_a^{m+1}$ ;
  end
   $\mathbf{e}^{m+1} = S_3^{-1} \left( (\mathbb{M}_1 - \frac{\Delta t^2}{16} \mathbb{C}^\top \mathbb{M}_2 \mathbb{C}) \mathbf{e}^{\text{iter}} + \Delta t \mathbb{C}^\top \mathbb{M}_2 \mathbf{b}^* \right)$ ;
   $\mathbf{b}^{m+1} = \mathbf{b}^* - \frac{\Delta t}{4} \mathbb{C} (\mathbf{e}^{\text{iter}} + \mathbf{e}^{m+1})$ ;
end

```

Algorithm 2: Second order conservative discrete gradient scheme.

For this version with substepping, we have two nested nonlinear systems. The sub-iteration (20) is a nonlinear system in $(\mathbf{X}_m^{\nu+1}, \mathbf{V}_m^{\nu+1})$ which decomposes into separate systems for each particle which can be included into the overall iteration that is similar to the algorithm without substepping. In our implementation, we use Picard iterations for both nonlinear systems. A Strang splitting version with substepping is shown in Algorithm 3.

5.3.1. Conservation properties

Despite of not being a discrete gradient scheme, the scheme with substepping is energy conserving as can be seen from the following calculation: Setting $\mathbf{V}^{m+1} = \mathbf{V}_m^{N_\nu}$, we get

$$\begin{aligned}
 (\mathbf{V}^{m+1})^\top \mathbb{M}_p \mathbf{V}^{m+1} - (\mathbf{V}^m)^\top \mathbb{M}_p \mathbf{V}^m &= \sum_{\nu=0}^{N_{\text{sub}}-1} \left((\mathbf{V}_m^{\nu+1})^\top \mathbb{M}_p \mathbf{V}_m^{\nu+1} - (\mathbf{V}_m^\nu)^\top \mathbb{M}_p \mathbf{V}_m^\nu \right) \\
 &\stackrel{(20b)}{=} \sum_{\nu=0}^{N_{\text{sub}}-1} \left((\mathbf{V}_m^{\nu+1} + \mathbf{V}_m^\nu)^\top \mathbb{M}_q \int_{\tau_\nu}^{\tau_{\nu+1}} \mathbb{A}_i^1(\mathbf{X}_m(\tau)) d\tau \right) \left(\frac{\mathbf{e}^m + \mathbf{e}^{m+1}}{2} \right) \\
 &\stackrel{(21)}{=} - \left((\mathbf{e}^{m+1})^\top \mathbb{M}_1 \mathbf{e}^{m+1} - (\mathbf{e}^m)^\top \mathbb{M}_1 \mathbf{e}^m \right).
 \end{aligned}$$

Hence, the difference in the kinetic energy equals the negative difference in the potential energy (since the magnetic field is not changed in this system) and thus the total energy is conserved.

Moreover, Gauss' law is respected over time, since we have in each subinterval $\tau \in [\tau_\nu, \tau_{\nu+1}]$ that

$$\mathbf{X}_m(\tau) = ((\tau_{\nu+1} - \tau) \mathbf{X}_m^\nu + (\tau - \tau_\nu) \mathbf{X}_m^{\nu+1}) / \Delta \tau_\nu = \mathbf{X}_m^\nu + \frac{\mathbf{V}_m^\nu + \mathbf{V}_m^{\nu+1}}{2} (\tau - \tau_\nu),$$

```

for  $m = 1, \dots$  do
   $\mathbf{e}^* = S_3^{-1} \left( \left( \mathbb{M}_1 - \frac{\Delta t^2}{16} \mathbb{C}^\top \mathbb{M}_2 \mathbb{C} \right) \mathbf{e}^m + \Delta t \mathbb{C}^\top \mathbb{M}_2 \mathbf{b}^m \right);$ 
   $\mathbf{b}^* = \mathbf{b}^m - \frac{\Delta t}{4} \mathbb{C} (\mathbf{e}^m + \mathbf{e}^*);$ 
   $\mathbf{j} = 0;$ 
   $N = 0;$ 
  for  $a = 1, \dots, N_p$  do
     $\mathbf{v}_a^* = R(\mathbf{b}^*) \mathbf{v}_a^m;$ 
     $\mathbf{x}_a^{\text{iter}} = \mathbf{x}_a^m + \frac{\Delta t}{2} \mathbf{v}_a^*;$ 
     $\mathbf{v}_a^{**} = \mathbf{v}^* + \frac{\Delta t}{2} \frac{q_s}{m_s} \mathbb{A}^1(\mathbf{x}_a^{\text{iter}}) \mathbf{e}^*;$ 
     $\mathbf{j} = \mathbf{j} + \Delta t q_s w_a \mathbb{A}^1(\mathbf{x}_a^{\text{iter}}) \mathbf{v}_a;$ 
     $N = N + \frac{\Delta t^2}{4} q_s w_a \mathbb{A}^1(\mathbf{x}_a^{\text{iter}})^\top \mathbb{A}^1(\mathbf{x}_a^{\text{iter}});$ 
  end
   $\mathbf{e}^{\text{iter}} = (\mathbb{M}_1 - N) \mathbf{e}^* - \mathbf{j};$ 
  for  $a = 1, \dots, N_p$  do
     $\mathbf{v}_a^{\text{iter}} = \mathbf{v}_a^{**} + \frac{\Delta t}{2} \frac{q_s}{m_s} \mathbb{A}^1(\mathbf{x}_a^{\text{iter}}) \mathbf{e}^{\text{iter}};$ 
     $\mathbf{x}_a^{\text{iter}} = \mathbf{x}_a^{\text{iter}} + \frac{\Delta t}{2} \mathbf{v}_a^{\text{iter}};$ 
  end
   $\text{residual} = \text{tolerance} + 1;$ 
  while  $\text{residual} < \text{tolerance}$  do
     $\mathbf{j} = 0;$ 
     $\tilde{\mathbf{e}} = \frac{\mathbf{e}^* + \mathbf{e}^{\text{iter}}}{2};$ 
    for  $a = 1, \dots, N_p$  do
       $\mathbf{x}_a^{\text{before}} = \mathbf{x}_a^*;$ 
       $\mathbf{v}_a^{\text{before}} = \mathbf{v}_a^*;$ 
      for  $v = 1, \dots, N_v$  do
         $\text{subresidual} = \text{subtolerance} + 1;$ 
         $\mathbf{x}_a^{\text{old}} = \mathbf{x}_a^{\text{before}};$ 
         $\mathbf{v}_a^{\text{old}} = \mathbf{v}_a^{\text{before}};$ 
        while  $\text{subresidual} < \text{subtolerance}$  do
           $\tilde{\mathbf{v}} = \frac{1}{2} (\mathbf{v}_a^{\text{before}} + \mathbf{v}_a^{\text{iter}});$ 
           $\mathbf{x}_a^{\text{iter}} = \mathbf{x}_a^{\text{before}} + \Delta t \tilde{\mathbf{v}};$ 
           $\mathbf{v}_a^{\text{iter}} = \mathbf{v}_a^{\text{before}} + \frac{\Delta t}{2} \int_0^1 \frac{q_s}{m_s} \mathbb{A}^1(\mathbf{x}_a^{\text{before}} + \tau \tilde{\mathbf{v}}) d\tau \tilde{\mathbf{e}};$ 
           $\text{subresidual} = \max(\|\mathbf{x}_a^{\text{iter}} - \mathbf{x}_a^{\text{old}}\|_\infty, \|\mathbf{v}_a^{\text{iter}} - \mathbf{v}_a^{\text{old}}\|_\infty);$ 
           $\mathbf{x}_a^{\text{old}} = \mathbf{x}_a^{\text{iter}};$ 
           $\mathbf{v}_a^{\text{old}} = \mathbf{v}_a^{\text{iter}};$ 
        end
         $\mathbf{j} = \mathbf{j} + \Delta t q_s w_a \int_0^1 \mathbb{A}^1(\mathbf{x}_a^{\text{old}} + \tau \tilde{\mathbf{v}}) d\tau \tilde{\mathbf{v}};$ 
         $\mathbf{x}_a^{\text{before}} = \mathbf{x}_a^{\text{iter}};$ 
         $\mathbf{v}_a^{\text{before}} = \mathbf{v}_a^{\text{iter}};$ 
      end
    end
     $\mathbf{e}^{\text{old}} = \mathbf{e}^{\text{iter}};$ 
     $\mathbf{e}^{\text{iter}} = \mathbf{e}^* - M^{-1} \mathbf{j};$ 
     $\text{residual} = \|\mathbf{e}^{\text{iter}} - \mathbf{e}^{\text{old}}\|_2;$ 
  end
  for  $a = 1, \dots, N_p$  do
     $\mathbf{v}_a^{m+1} = R(\mathbf{b}^*) \mathbf{v}_a^{\text{iter}};$ 
     $\mathbf{x}_a^{m+1} = \mathbf{x}_a^{\text{iter}} + \frac{\Delta t}{2} \mathbf{v}_a^{m+1};$ 
  end
   $\mathbf{e}^{m+1} = S_3^{-1} \left( \left( \mathbb{M}_1 - \frac{\Delta t^2}{16} \mathbb{C}^\top \mathbb{M}_2 \mathbb{C} \right) \mathbf{e}^{\text{iter}} + \Delta t \mathbb{C}^\top \mathbb{M}_2 \mathbf{b}^* \right);$ 
   $\mathbf{b}^{m+1} = \mathbf{b}^* - \frac{\Delta t}{4} \mathbb{C} (\mathbf{e}^{\text{iter}} + \mathbf{e}^{m+1});$ 
end

```

Algorithm 3: Second order conservative discrete gradient scheme with substepping.

and hence

$$\frac{d\mathbf{X}_m(\tau)}{d\tau} = \frac{\mathbf{V}_m^\nu + \mathbf{V}_m^{\nu+1}}{2},$$

which corresponds to the form of $\mathbf{V}(\tau)$ in (21).

6. Numerical experiments

In this section, we verify the conservation properties of our new time discretization methods for a number of test problems. We first study the reduced model in 1d2v phase-space with a perturbation along x_1 , a magnetic field along x_3 ,

Table 1

Weibel instability in 1d2v phase space: Comparison of the conservation properties for various integrators. For the two discrete gradient method, we show results for two different orderings of the individual operators (the first set of experiments refers to the case with the ordering shown in the Algorithms).

Method	Δt	Gauss	Energy	Error magn. energy
explicit	0.025	2.39E-15	3.48E-07	5.12E-06
Hamiltonian splitting	0.05	2.39E-15	1.39E-06	1.88E-05
	0.1	—	—	—
semi-implicit	0.025	1.25E-07	6.60E-15	1.45E-05
discrete gradient	0.05	2.68E-07	3.08E-14	1.53E-05
(O3,O1,O2,O4)	0.1	1.10E-06	1.11E-14	4.15E-05
	0.2	5.80E-06	8.76E-15	1.24E-04
semi-implicit	0.025	8.77E-08	7.44E-15	5.06E-06
discrete gradient	0.05	3.38E-07	1.97E-14	2.02E-05
(O1,O2,O3,O4)	0.1	1.93E-06	8.81E-15	4.06E-05
	0.2	7.23E-06	1.03E-14	1.12E-04
implicit conservative	0.025	2.28E-15	1.56E-13	8.22E-06
discrete gradient	0.05	2.32E-15	2.60E-14	1.50E-05
(O3,O2,O4)	0.1	2.14E-15	1.66E-13	4.11E-05
	0.2	2.09E-15	5.48E-14	1.20E-04
implicit conservative	0.025	2.72E-15	1.01E-13	5.68E-06
discrete gradient	0.05	2.23E-15	5.92E-15	1.69E-05
(O2,O3,O4)	0.1	2.24E-15	1.81E-13	3.99E-05
	0.2	2.24E-15	2.15E-14	1.09E-04

and an electric field along the x_1 and x_2 directions. Moreover, we assume that the distribution function is independent of v_3 . For this example, we report results on a Weibel instability, the two-stream instability, and a two-species simulation of an ion-acoustic wave. Also, we demonstrate the absence of grid heating. Finally, we show also results for the Weibel instability simulated in full 3d3v phase-space.

In all experiments reported, we have used a second order Strang splitting method and third order splines for the 0-forms. For the iterative linear solvers, the tolerance is set to 10^{-15} and for the nonlinear iteration in the discrete gradient method we use a tolerance of 10^{-12} and a tolerance of 10^{-10} for the subiterations when they exist. Note that the tolerance of the linear solver is applied to the residual while the tolerance in the nonlinear iteration is directly applied to the fields. In order to balance the errors, we therefore use a more restrictive tolerance for the linear solvers. In the subiterations, only particle positions are involved which is why the tolerance can be chosen less restrictive. The implementation is based on the Fortran libraries SeLaLib¹ and PLAF.²

6.1. Weibel instability in 1d2v

As a first example, we consider the Weibel instability [27] in 1d2v. We use the same parameters that had already been considered with the Hamiltonian splitting time discretization in [1]. The initial distribution and fields are of the form

$$f(x, \mathbf{v}, t = 0) = \frac{1}{2\pi\sigma_1\sigma_2} \exp\left(-\frac{1}{2}\left(\frac{v_1^2}{\sigma_1^2} + \frac{v_2^2}{\sigma_2^2}\right)\right) (1 + \alpha \cos(kx)), \quad x \in [0, 2\pi/k),$$

$$B_3(x, t = 0) = \beta \cos(kx),$$

$$E_2(x, t = 0) = 0,$$

and $E_1(x, t = 0)$ is computed from Poisson's equation. In our simulations, we use the following choice of parameters, $\sigma_1 = 0.02/\sqrt{2}$, $\sigma_2 = \sqrt{12}\sigma_1$, $k = 1.25$, $\alpha = 0$ and $\beta = 10^{-4}$ and simulate for 500 time units. We use 100,000 particles and 32 grid points.

We run the simulation with the semi-implicit discrete gradient and the Gauss-conserving discrete gradient method for various time steps and compare to the Hamiltonian splitting. Table 1 shows the conservation properties of the various runs. The numerical experiments verify energy conservation of the new semi-implicit methods and conservation of Gauss' law for the discrete gradient method and the Hamiltonian splitting. In particular, we note that the error in Gauss' law is satisfied up to round-off errors in the Hamiltonian splitting as well as the conservative discrete gradient method. Moreover, the implicit methods conserve energy to the accuracy of the iterative field solver. Hence, the level of accuracy in the conservation properties only depends on the tolerance applied in the linear and nonlinear iterations and not on the spatial

¹ <http://selalib.gforge.inria.fr/>.

² <http://jorek.gforge.inria.fr/documentation/plaf/html/index.html>.

Table 2

Two-stream instability 1d2v phase space: Comparison of the conservation properties for various integrators.

Method	Δt	Gauss	Energy
explicit	0.025	4.77E-15	7.37E-04
Hamiltonian splitting	0.05	4.11E-15	2.79E-03
	0.1	4.77E-15	9.37E-03
	0.2	4.44E-15	4.81E-01
	0.4	—	—
	0.4	—	—
semi-implicit	0.025	1.26E-03	6.71E-12
discrete gradient	0.05	5.17E-03	5.68E-12
	0.1	2.56E-02	6.08E-12
	0.2	2.07E-01	5.49E-12
	0.4	1.30E-00	5.68E-12
	0.4	1.30E-00	5.68E-12
implicit conservative	0.025	4.55E-15	9.69E-12
discrete gradient	0.05	3.89E-15	8.38E-11
	0.1	3.55E-15	5.68E-12
	0.2	4.33E-15	1.68E-11
	0.4	4.72E-15	2.71E-11
	0.4	4.72E-15	2.71E-11

or temporal resolution of the numerical approximation. We can also see that the semi-implicit methods allow for larger time steps where the Hamiltonian splitting becomes unstable due to the stability constraint $\Delta t \leq \sqrt{\frac{17}{42}} \Delta x \approx 0.099935$ due to the explicit scheme for Maxwell's equations.

The number of nonlinear iterations in the discrete gradient scheme are about 4 ($\Delta t = 0.025$), 5 ($\Delta t = 0.05$), 6 ($\Delta t = 0.1$), and 8 ($\Delta t = 0.2$). Hence, we see only a moderate increase in the number of iterations. Roughly speaking the cost of the implicit method is a factor “number of iterations” more expensive than the explicit Hamiltonian splitting. This shows that the computational costs of our implicit and semi-implicit schemes are comparable to the costs that are required for the EC-PIC and EC-SIM algorithms (cf. [8]).

In this example, the error of the various methods is comparable as we see for the maximum error over time in the magnetic energy shown in the last column of Table 1. The error in the magnetic energy is computed compared to a solution with a time step $\Delta t = 0.0125$ and the same method.

6.2. Two-stream instability

As a second example, we look at a classical electrostatic test case known as the two-stream instability with the following initial value

$$f(x, \mathbf{v}, t = 0) = \frac{1}{4\pi} \left(e^{-0.5(v_1 - 2.4)^2} + e^{-0.5(v_1 + 2.4)^2} \right) e^{-0.5v_2^2}, \quad x \in [0, 10\pi),$$

$$B_3(x, t = 0) = 0,$$

$$E_2(x, t = 0) = 0,$$

and E_1 computed from Poisson's equation. We have simulated the problem over 200 time units with 64 grid points and 64,000 particles with the various integrators for time steps of 0.025, 0.05, 0.1, 0.2, and 0.4. Again the time step for the explicit scheme is restricted due to the Maxwell's equations by $\Delta t \leq \sqrt{\frac{17}{42}} \Delta x \approx 0.19882$. We note that larger time steps would therefore be possible for this electrostatic example if we solve the Vlasov-Ampère equation instead. For the discrete gradient methods the operators are ordered as shown in the algorithms. The simulations with the discrete gradient method took the following number of iterations on average: 7 ($\Delta t = 0.025$), 7 ($\Delta t = 0.05$), 9 ($\Delta t = 0.1$), 11 ($\Delta t = 0.1$), 16 ($\Delta t = 0.4$).

The conservation properties of the various simulations are summarized in Table 2. All simulations with time step smaller than 0.4 capture the linear growth rate quite accurately and show only small variations in the nonlinear phase. For a time step of 0.4, the stability condition of the Hamiltonian splitting is violated, the semi-implicit discrete gradient was still able to capture the linear growth rate but showed large deviation in the nonlinear phase while the results of the discrete gradient scheme remains very accurate. Moreover, the initially random electric field deviates considerably for the semi-implicit discrete gradient method with time steps of 0.2 and 0.4 which can be explained by the fact that Gauss' law is violated to a very large degree in these simulations. Fig. 2 shows the electric energy as a function of time for the discrete gradient method at a time step of 0.025 as a reference and the simulations with $\Delta t = 0.2$ and $\Delta t = 0.4$.

Even though we have demonstrated that large time steps are possible with the implicit methods for the examples of the two-stream and the Weibel instability, large time steps are not particularly beneficial for these examples due to accuracy. As a next example we will therefore study a multi-species problem where physical effects take place on the time-scale of the ions so that large time steps are interesting for the electron dynamics.

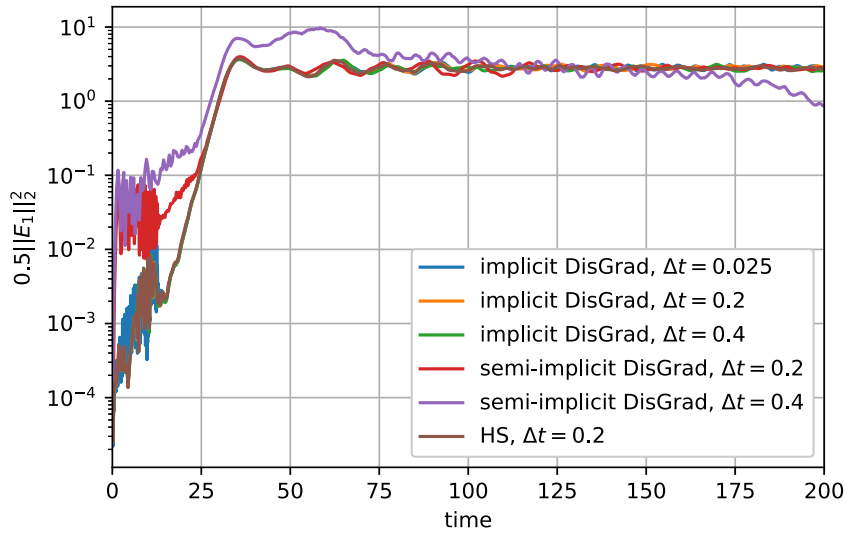


Fig. 2. Two-stream instability: Time evolution of the first component of the electric energy for various configurations. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

6.3. Ion acoustic wave

As a third example, we consider the ion acoustic wave excited by an ion density perturbation. The example is electrostatic and involves electrons and ions. We normalize mass and temperature to their values of the electrons. The initial distributions are then given by

$$f_e(x, v, t=0) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{v^2}{2}\right),$$

$$f_i(x, v, t=0) = \frac{1}{\sqrt{2\pi} \frac{T_i}{m_i}} \exp\left(-\frac{v^2}{2\sqrt{\frac{T_i}{m_i}}}\right) \left(1 + \alpha \cos\left(\frac{2\pi}{L}x\right)\right).$$

We use the following parameters $T_i = 10^{-4}$, $m_i = 200$, $\alpha = 0.2$, $L = 10$.

The ion acoustic wave problem is solved with 32 grid points and 128,000 particles per species with the explicit Hamiltonian splitting, the semi-implicit discrete gradient method, the conservative discrete gradient method, and the substepping method for various time steps. In this case, we use the ordering that places operator 3 after operators 1 and 2, since this gives more accurate results in the present example. Table 3 shows the conservation properties of the various methods for different time steps. As expected Gauss' law is satisfied to machine precision for the Hamiltonian splitting as well as the discrete gradient method. On the other hand, the error in Gauss' law is considerable for the semi-implicit method and increases with the time step. The discrete gradient methods conserve energy—in contrast to the Hamiltonian splitting—up to the tolerance of the iterative solvers.

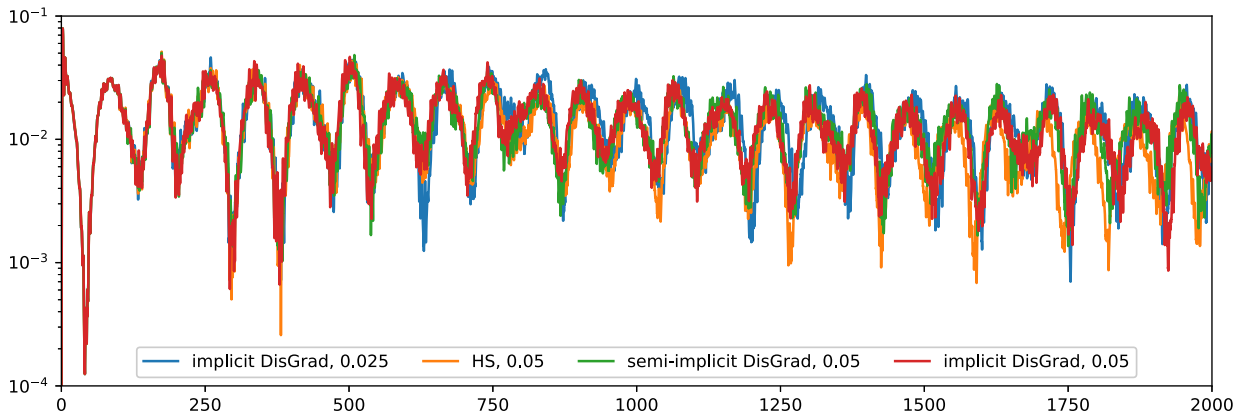
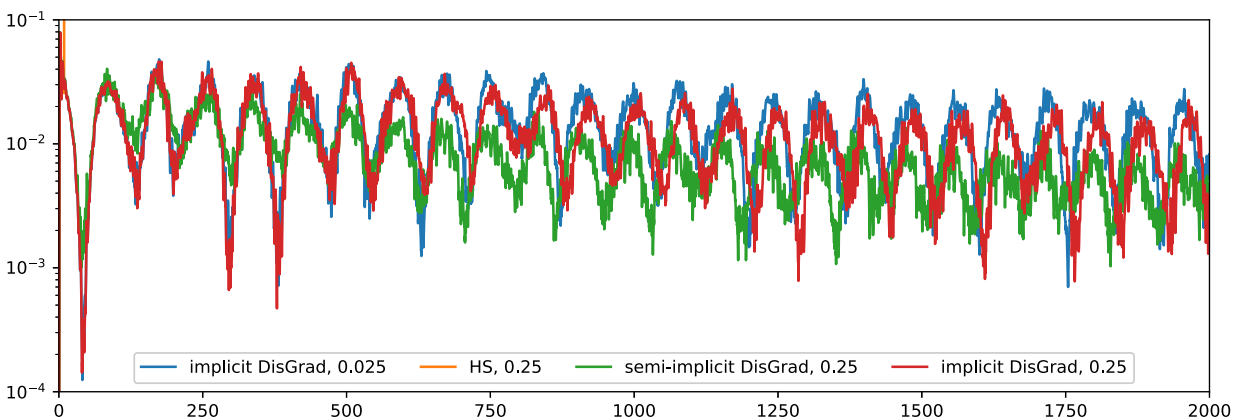
In order to judge the quality of the solution, we look at the evolution of the first component of the electric energy over time. As a reference, we show the solution of the discrete gradient method with a time step of $\Delta t = 0.025$. Fig. 3 shows the results for the Hamiltonian splitting, the semi-implicit, and the conservative discrete gradient solution with a time step of $\Delta t = 0.05$. We can see that all three time stepping schemes give quite good results for this relatively small time step. Next we increase the time step to $\Delta t = 0.25$ in Fig. 4 where the explicit method becomes unstable and the conservative discrete gradient method gives clearly better results than the semi-implicit method. Finally, Fig. 5 shows that the solution of the discrete gradient method becomes clearly worse when increasing the time step from $\Delta t = 0.25$ to $\Delta t = 1.0$. When introducing a substepping scheme with four substeps for the electrons only, the quality of the solution at $\Delta t = 1$ and $\Delta \tau = 0.25$ for the electrons is comparable to the discrete gradient method with a total time step of $\Delta t = 0.25$ for both species.

The efficiency of the conservative discrete gradient method is hampered by the number of iterations needed for the nonlinear solution to converge. In particular, for the smallest time step considered ($\Delta t = 0.025$), we already need about 6 iterations on average which renders the method uncompetitive compared to the explicit one. When increasing the time step, the iteration count increases to about 7 iterations for $\Delta t = 0.05$, 12 iterations for $\Delta t = 0.25$, 17 for $\Delta t = 0.5$, and 32 for $\Delta t = 1.0$. We can see that the increase of iterations needed between $\Delta t = 0.05$ and $\Delta t = 0.25$ is quite small compared to the increase in the time step. Then, the iteration count starts increasing at a higher rate. However, the quality of the

Table 3

Ion acoustic wave: Comparison of the conservation properties for various integrators.

Method	Δt	Gauss	Energy
explicit	0.025	9.10E-15	1.57E-05
Hamiltonian splitting	0.05	9.96E-15	8.93E-05
semi-implicit	0.025	3.07E-4	4.28E-13
discrete gradient	0.05	1.78E-03	3.27E-13
	0.25	7.78E-02	2.75E-13
	0.5	2.31E-01	2.58E-13
	1.0	3.93E-01	2.36E-13
implicit conservative	0.025	1.11E-14	9.34E-12
discrete gradient	0.05	1.22E-14	2.96E-12
	0.25	1.28E-14	2.96E-12
	0.5	1.49E-14	4.52E-12
	1.0	1.53E-14	1.85E-11
substepping(4,1)	1.0	1.71E-14	5.45E-13

**Fig. 3.** Ion acoustic wave: Time evolution of the first component of the electric energy with various time propagation schemes at $\Delta t = 0.05$ compared to a reference simulation with $\Delta t = 0.025$.**Fig. 4.** Ion acoustic wave: Time evolution of the first component of the electric energy with various time propagation schemes at $\Delta t = 0.25$ compared to a reference simulation with $\Delta t = 0.025$.

solution with such large time steps is not very good either so that the high value of the iteration tolerance is questionable in those simulations. The substepping method still produces good results for $\Delta t = 1.0$ and four substeps for the electrons and no substeps for the ions. In this case, about 16 outer iterations are needed per time step and about 5 inner iterations for both electrons (per substep) and ions.

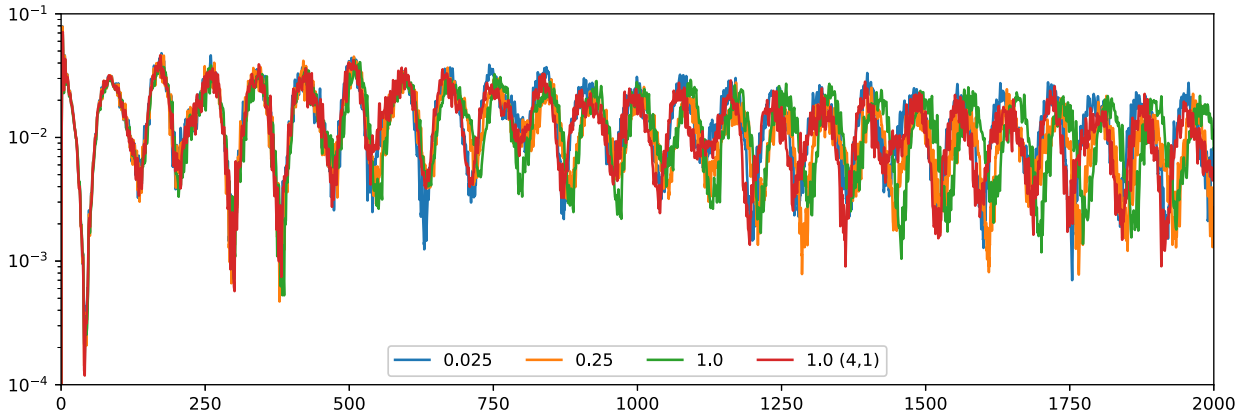


Fig. 5. Ion acoustic wave: Time evolution of the first component of the electric energy comparing the discrete gradient method without sub-stepping and a time step of $\Delta t = 0.025$, $\Delta t = 0.25$, and $\Delta t = 1.0$ compared to the discrete gradient method with $\Delta t = 1.0$ and four substeps for the electrons.

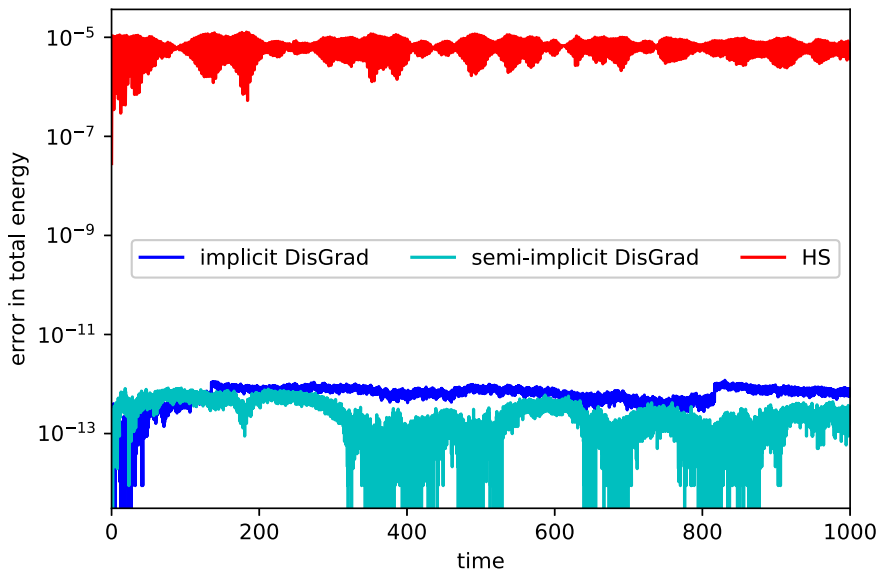


Fig. 6. Grid heating: Time evolution of the absolute error in total energy for simulation with resolution > 10 times the Debye length.

Note that a linear dispersion analysis shows that the electric energy should oscillate with no damping in this case. Our results, however, show a slight damping. This is not an effect of the time stepping scheme but of the spatial discretization. Adding more particles or smoothing the fields and the current improves the results.

6.4. Grid heating

For numerical schemes that lack energy conservation, artificial heating occurs if the Debye length is not resolved on the spatial grid. Let us consider a Maxwellian initial condition of the form

$$f(x, v_1, v_2) = \frac{1}{2\pi\sigma} \exp\left(-0.5 \left(\frac{v_1^2 + v_2^2}{\sigma^2}\right)\right)$$

in a periodic box of length $L = 50\pi$. In our normalized units the plasma frequency is equal to one and the Debye length then takes the value σ of the thermal velocity. We choose a spatial resolution of 64 grid points, i.e., $\Delta x \approx 2.4$, and a thermal velocity of $\sigma = 0.2$. In this case, the Debye length is a factor 10 larger than the spatial resolution and a standard explicit method would suffer from grid heating. On the contrary, we do not see a grid heating for any of our methods. Fig. 6 shows the evolution of the absolute error in the total energy over time for simulations with a very small time step of $\Delta t = 0.05$. The value of the total energy is around 12.6. In particular, the energy is conserved over time up to the solver tolerance of 10^{-12} for the implicit method. For the explicit Hamiltonian splitting the energy is not conserved but it shows

Table 4

Weibel instability in 3d3v phase space: Comparison of the conservation properties for various integrators.

Method	Δt	Gauss	Energy	Error magn. energy
explicit	0.05	5.80E-16	6.70E-06	1.23E-03
Hamiltonian splitting	0.1	5.70E-16	2.72E-05	5.03E-03
	0.2	—	—	—
	0.4	—	—	—
semi-implicit	0.05	3.47E-07	2.59E-13	8.65E-04
discrete gradient	0.1	1.60E-06	3.76E-14	3.63E-03
	0.2	6.00E-06	2.85E-14	1.35E-02
	0.4	2.27E-05	6.88E-14	2.91E-02
	0.4	2.27E-05	6.88E-14	2.91E-02
implicit conservative	0.05	1.27E-15	3.53E-13	8.61E-04
discrete gradient	0.1	1.20E-15	1.50E-14	3.63E-03
	0.2	1.06E-15	2.22E-13	1.33E-02
	0.4	1.03E-15	8.22E-15	2.92E-02
	0.4	1.03E-15	8.22E-15	2.92E-02

an oscillatory behavior instead of the steady increase that is referred to as grid heating. We note that the energy error of the Hamiltonian splitting is solely caused by the splitting in time and its error is a function of Δt only. The spatial discretization is energy-conserving which is why this explicit scheme does not suffer from grid heating.

6.5. Weibel instability in 3d3v

In order to demonstrate the potential of the algorithm also in full phase space, we consider the Weibel test case in a simulation in the full six-dimensional phase space. The initial distribution is of the form

$$f(\mathbf{x}, \mathbf{v}, t = 0) = \frac{1}{(2\pi)^{3/2} \sigma_1 \sigma_2^2} \exp \left(-\frac{1}{2} \left(\frac{v_1^2}{\sigma_1^2} + \frac{v_2^2 + v_3^2}{\sigma_2^2} \right) \right), \quad \mathbf{x} \in [0, 2\pi/k]^3,$$

$$\mathbf{B}(\mathbf{x}, t = 0) = (0, 0, \beta \cos(kx_1))^T,$$

and the electric field at time zero is computed from Poisson's equation. In our simulation, we choose the parameters $\sigma_1 = 0.02/\sqrt{2}$, $\sigma_2 = \sqrt{12}\sigma_1$, $k = 1.25$, and $\beta = 0.01$ and simulate for 250 time units. Note that the problem is mainly depending on the variable x_1 which is why we resolve the spatial grid more along this direction, namely we choose a resolution of $16 \times 8 \times 8$ grid points. The simulation uses $N_p = 100,000$ particles.

Table 4 shows the conservation properties of the various integrators with different time steps. The results here are shown for the standard ordering of the operators that maximizes the arithmetic intensity. Again the conservation properties are verified. For this case, the conservative discrete gradient method needs about 5 ($\Delta t = 0.05$), 6 ($\Delta t = 0.1$), 9 ($\Delta t = 0.2$), or 14 ($\Delta t = 0.4$) iterations per time step on average. The error in the magnetic energy is computed compared to a solution with the same method and a time step of $\Delta t = 0.0125$. The accuracy is comparable for all methods with a slight disadvantage for the Hamiltonian splitting.

7. Conclusions and outlook

We have described a general procedure to derive energy-conserving time-stepping methods for the geometric electromagnetic particle-in-cell method based on discrete gradients and an antisymmetric splitting of the Poisson matrix. In particular, we derived a semi-implicit scheme, which yields good results with little computational overhead per time step compared to the Hamiltonian splitting when conservation of Gauss' law is not critical and, in the same time, allows for larger time steps. Moreover, we have derived an implicit method that conserves both energy and Gauss' law. Due to the fact that a nonlinear iteration that couples the particle and field degrees of freedom is necessary, the method comes with a considerable computational overhead. On the other hand, combined with a substepping for the fast species, it yields quite accurate results for large time steps and has therefore the potential to be more efficient in multiscale simulations with realistic mass ratio between electron and ion species. Finally, we have also shown the absence of grid heating for the GEMPIC semi-discretization independent of the time stepping scheme.

CRedit authorship contribution statement

Katharina Kormann: Methodology, Software, Formal analysis, Investigation, Writing - original draft. **Eric Sonnendrücker:** Conceptualization, Methodology, Writing - review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

The authors acknowledge discussions with Michael Kraus, Philip J. Morrison, Benedikt Perse, Jalal Lakhili, and Ahmed Ratnani. K.K. is grateful to Elena Celledoni and Brynjulf Owren for discussions during the program “Geometry, compatibility and structure preservation in computational differential equations” at the Isaac Newton Institute for Mathematical Sciences, Cambridge. This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training program 2014-2018 and 2019-2020 under grant agreement No 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

Appendix A. Stability analysis

A comprehensive stability analysis of the geometric particle-in-cell method is quite hard to achieve due to the fact that the scheme is highly nonlinear. A rather general stability analysis for electromagnetic particle-in-cell schemes was provided by Godfrey [28], however, it does not directly apply to our methods, since it is based on a staggered time step while we use a more complex splitting of the equations. On the other hand, the stability limits of explicit particle-in-cell schemes are mostly related to the way the curl-part in Maxwell’s equations is solved and the propagation of electrostatic Langmuir waves. In this section, we will therefore perform a stability analysis for the curl-part in Maxwell’s equation in 1d and for Langmuir waves.

A.1. Stability analysis for Langmuir waves

In this section, we perform a stability analysis for Langmuir waves as in [8] following the theory developed in [29]. For the electrostatic case, both implicit schemes in a Strang splitting combination read as follows

$$\mathbf{X}^{m+1/2} = \mathbf{X}^{m-1/2} + \Delta t \mathbf{V}^m, \quad \mathbf{V}^{m+1} = \mathbf{V}^m + \frac{\Delta t}{2} \frac{q_s}{m_s} (\mathbf{E}^m + \mathbf{E}^{m+1}). \quad (\text{A.1})$$

For the von Neumann stability analysis, the time evolution is assumed to be harmonic, i.e.,

$$\mathbf{x}_p^m = \bar{\mathbf{x}}_p e^{i\omega m \Delta t}, \quad \mathbf{v}_p^m = \bar{\mathbf{v}}_p e^{i\omega m \Delta t}, \quad \mathbf{E}_p^m = \bar{\mathbf{E}}_p e^{i\omega m / \Delta t}.$$

Inserting the time harmonic ansatz into the scheme (A.1), we find

$$\begin{aligned} 2i \sin\left(\frac{\Delta t}{2} \omega\right) \bar{\mathbf{x}}_p &= \Delta t \bar{\mathbf{v}}_p, \\ 2i \sin\left(\frac{\Delta t}{2} \omega\right) \bar{\mathbf{v}}_p &= \frac{q_s}{m_s} \Delta t \cos\left(\frac{\Delta t}{2} \omega\right) \bar{\mathbf{E}}_p. \end{aligned}$$

In Fourier space the electric field for cold plasma Langmuir waves can be related to the displacement by the electron plasma frequency ω_{pe} as (cf. [8])

$$\frac{q_s}{m_s} \bar{\mathbf{E}}_p = -\omega_{pe}^2 \bar{\mathbf{x}}.$$

With this assumption, we obtain

$$\begin{aligned} 2i \sin\left(\frac{\Delta t}{2} \omega\right) \bar{\mathbf{x}}_p - \Delta t \bar{\mathbf{v}}_p &= 0, \\ \cos\left(\frac{\Delta t}{2} \omega\right) \Delta t \omega_{pe}^2 \bar{\mathbf{x}}_p + 2i \sin\left(\frac{\Delta t}{2} \omega\right) \bar{\mathbf{v}}_p &= 0. \end{aligned}$$

This system of linear equations has a solution if the determinant of the matrix vanishes, i.e., if

$$-4 \sin^2\left(\frac{\Delta t}{2} \omega\right) + \Delta t^2 \omega_{pe}^2 \cos\left(\frac{\Delta t}{2} \omega\right) = 0.$$

For this case, we hence get the same stability condition as for the semi-implicit scheme introduced by Lapenta in [8]. The equation has real solutions independent of Δt and, hence, the scheme is unconditionally stable.

The Hamiltonian splitting method with second order Strang splitting, on the other hand, yields the standard explicit leap frog scheme for electrostatics

$$\mathbf{X}^{m+1/2} = \mathbf{X}^{m-1/2} + \Delta t \mathbf{V}^m, \quad \mathbf{V}^{m+1} = \mathbf{V}^m + \Delta t \frac{q_s}{m_s} \mathbf{E}^{m+1/2}.$$

In this case the determinant condition reads

$$-4 \sin^2 \left(\frac{\Delta t}{2} \omega \right) + \Delta t^2 \omega_{pe}^2 = 0.$$

For $\omega_{pe} \Delta t > 2$, there are only complex conjugate solutions, i.e., we have one growing solution which leads to numerical instabilities.

A.2. Curl-part of Maxwell's equations (vacuum Maxwell's equation)

Let us now consider the curl-part of Maxwell's equations in 1d with a second order Strang splitting propagator. Let us denote $\tilde{\mathbf{e}} = \mathbb{M}_1 \mathbf{e}$ and $\tilde{\mathbf{b}} = \mathbb{M}_2 \mathbf{b}$. With this notation the explicit version of the Maxwell equations with a finite element description reads for degree of freedom j

$$\begin{aligned} \tilde{e}_j^{n+1/2} &= \tilde{e}_j^n + \frac{\alpha}{2} (\tilde{b}_j^n - \tilde{b}_{j+1}^n), \\ b_j^{n+1} &= b_j^n - \alpha (e_j^{n+1/2} - e_{j-1}^{n+1/2}), \\ \tilde{e}_j^{n+1} &= \tilde{e}_j^{n+1/2} + \frac{\alpha}{2} (\tilde{b}_j^{n+1} - \tilde{b}_{j+1}^{n+1}), \end{aligned}$$

where $\alpha = \frac{\Delta t}{\Delta x}$.

For the k th Fourier mode, we use the ansatz

$$e_j^n = \bar{e} \xi^n \exp \left(i \frac{2\pi k x_j}{L} \right), \quad b_j^n = \bar{b} \xi^n \exp \left(i \frac{2\pi k x_j}{L} \right),$$

where L denotes the length of the domain. Then, we have the following relation after multiplication with the mass matrices

$$\tilde{e}_j^n = \lambda_k^{(p)} e_j^n, \quad \tilde{b}_j^n = \lambda_k^{(p-1)} b_j^n,$$

where $\lambda_k^{(q)}$ denotes the k th eigenvalue of the mass matrix for q th order splines. The Fourier transformed difference equations then have the following form

$$\underbrace{\begin{pmatrix} (\xi - 1) \lambda_k^{(p)} & -(\xi + 1) \frac{\alpha}{2} \left(1 - e^{i \frac{2\pi \Delta x k}{L}} \right) \lambda_k^{(p-1)} \\ \alpha \left(1 - e^{-i \frac{2\pi \Delta x k}{L}} \right) \lambda_k^{(p)} & (\xi - 1) \lambda_k^{(p)} + \alpha^2 \lambda_k^{(p-1)} \left(1 - \cos \left(\frac{2\pi \Delta x k}{L} \right) \right) \end{pmatrix}}_{:=D} \begin{pmatrix} \bar{e} \\ \bar{b} \end{pmatrix} = 0.$$

To find a solution, we compute ξ such that the determinant is zero

$$\det(D) = \lambda_k^{(p)} \left(\lambda_k^{(p)} (\xi - 1)^2 + \alpha^2 2 \left(1 - \cos \left(\frac{2\pi \Delta x k}{L} \right) \right) \xi \lambda_k^{(p-1)} \right) = \lambda_k^{(p)} \left(\xi^2 - 2q\xi + 1 \right),$$

where $q = 1 - \frac{\lambda_k^{(p-1)}}{\lambda_k^{(p)}} \alpha^2 \left(1 - \cos \left(\frac{2\pi \Delta x k}{L} \right) \right)$. The roots of the equation $\det(D) = 0$ can be expressed as

$$\xi_{+/-} = q \pm \sqrt{q^2 - 1}.$$

For stability, we need to have $|\xi| \leq 1$ and thus $|q| \leq 1$ which yields the condition

$$0 \leq \frac{\lambda_k^{(p-1)}}{\lambda_k^{(p)}} \alpha^2 \left(1 - \cos \left(\frac{2\pi \Delta x k}{L} \right) \right) \leq 2,$$

for all values of k . For $p = 1, 2, 3$ this yields the following conditions on α :

- $p = 1$: $\alpha \leq \sqrt{\frac{1}{3}}$,
- $p = 2$: $\alpha \leq \sqrt{\frac{2}{5}}$,
- $p = 3$: $\alpha \leq \sqrt{\frac{17}{42}}$.

We note that a Lie splitting would yield the same determinant (up to the multiplicative factor $\lambda_k^{(p)}$) and, hence, yields the same stability limit.

Next, we consider the implicit variant. Since we will show that the scheme is unconditionally stable, it suffices to consider the Lie splitting

$$\begin{aligned} b_j^{n+1} &= b_j^n - \frac{\alpha}{2} \left(e_j^n + e_j^{n+1} - e_{j-1}^n - e_{j-1}^{n+1} \right), \\ \tilde{e}_j^{n+1} &= \tilde{e}_j^n + \frac{\alpha}{2} \left(\tilde{b}_j^n + \tilde{b}_j^{n+1} - \tilde{b}_{j-1}^n - \tilde{b}_{j-1}^{n+1} \right). \end{aligned}$$

With the same ansatz, we now get the following equation for mode k

$$\begin{pmatrix} \xi - 1 & \frac{\alpha}{2} \left(1 - e^{-i \frac{2\pi \Delta x k}{L}} \right) (\xi + 1) \\ -\frac{\alpha}{2} \left(1 - e^{i \frac{2\pi \Delta x k}{L}} \right) \lambda_k^{(p-1)} (\xi + 1) & (\xi - 1) \lambda_k^{(p)} \end{pmatrix} \begin{pmatrix} \bar{b} \\ \bar{e} \end{pmatrix} = 0.$$

This yields the following expression for the determinant

$$\begin{aligned} \det(D) &= \left(\lambda_k^{(p)} + \frac{\alpha^2}{2} \left(1 - \cos \left(\frac{2\pi \Delta x k}{L} \right) \right) \lambda_k^{(p-1)} \right) \xi^2 + \left(\lambda_k^{(p)} - \frac{\alpha^2}{2} \left(1 - \cos \left(\frac{2\pi \Delta x k}{L} \right) \right) \lambda_k^{(p-1)} \right) \xi + \\ &\quad \left(\lambda_k^{(p)} + \frac{\alpha^2}{2} \left(1 - \cos \left(\frac{2\pi \Delta x k}{L} \right) \right) \lambda_k^{(p-1)} \right). \end{aligned}$$

Solving the equation $\det(D) = 0$ for ξ yields

$$\xi_{+/-} = q \pm \sqrt{q^2 - 1},$$

where $q = \frac{\lambda_k^{(p)} - \frac{\alpha^2}{2} \left(1 - \cos \left(\frac{2\pi \Delta x k}{L} \right) \right) \lambda_k^{(p-1)}}{\lambda_k^{(p)} + \frac{\alpha^2}{2} \left(1 - \cos \left(\frac{2\pi \Delta x k}{L} \right) \right) \lambda_k^{(p-1)}}$. In this case, it holds that $q^2 - 1 \leq 0$ independent of k , Δx , p and α , since both $\lambda_k^{(p)} \geq 0$ and $\frac{\alpha^2}{2} \left(1 - \cos \left(\frac{2\pi \Delta x k}{L} \right) \right) \lambda_k^{(p-1)} \geq 0$. Hence, the scheme is unconditionally stable.

References

- [1] M. Kraus, K. Kormann, P.J. Morrison, E. Sonnendrücker, GEMPIC: geometric electromagnetic particle-in-cell methods, *J. Plasma Phys.* 83 (4) (2017), <https://doi.org/10.1017/S002237781700040X>.
- [2] Y. He, Y. Sun, H. Qin, J. Liu, Hamiltonian particle-in-cell methods for Vlasov–Maxwell equations, *Phys. Plasmas* 23 (9) (2016) 092108, <https://doi.org/10.1063/1.4962573>.
- [3] Y. He, H. Qin, Y. Sun, J. Xiao, R. Zhang, J. Liu, Hamiltonian integration methods for Vlasov–Maxwell equations, *Phys. Plasmas* 22 (2015) 124503, <https://doi.org/10.1063/1.4938034>.
- [4] G. Zhong, J.E. Marsden, Lie–Poisson Hamilton–Jacobi theory and Lie–Poisson integrators, *Phys. Lett. A* 133 (3) (1988) 134–139.
- [5] R.I. McLachlan, G.R.W. Quispel, N. Robidoux, Geometric integration using discrete gradients, *Philos. Trans. R. Soc. Lond. A, Math. Phys. Eng. Sci.* 357 (1754) (1999) 1021–1045, <https://doi.org/10.1098/rsta.1999.0363>.
- [6] E. Celledoni, V. Grimm, R. McLachlan, D. McLaren, D. O’Neale, B. Owren, G. Quispel, Preserving energy resp. dissipation in numerical (PDEs) using the “average vector field” method, *J. Comput. Phys.* 231 (20) (2012) 6770–6789, <https://doi.org/10.1016/j.jcp.2012.06.022>, <http://www.sciencedirect.com/science/article/pii/S0021999112003373>.
- [7] S. Markidis, G. Lapenta, The energy conserving particle-in-cell method, *J. Comput. Phys.* 230 (18) (2011) 7037–7052, <https://doi.org/10.1016/j.jcp.2011.05.033>.
- [8] G. Lapenta, Exactly energy conserving semi-implicit particle in cell formulation, *J. Comput. Phys.* 334 (2017) 349–366, <https://doi.org/10.1016/j.jcp.2017.01.002>, <http://www.sciencedirect.com/science/article/pii/S0021999117300128>.
- [9] B. Perse, Energy-conserving implicit time discretization for the GEMPIC framework, Master’s thesis, Technical University of Munich, Garching, 2017.
- [10] B. Marder, A method for incorporating Gauss’ law into electromagnetic PIC codes, *J. Comput. Phys.* 68 (1) (1987) 48–55, [https://doi.org/10.1016/0021-9991\(87\)90043-X](https://doi.org/10.1016/0021-9991(87)90043-X).
- [11] A.B. Langdon, On enforcing Gauss’ law in electromagnetic particle-in-cell codes, *Comput. Phys. Commun.* 70 (3) (1992) 447–450, [https://doi.org/10.1016/0010-4655\(92\)90105-8](https://doi.org/10.1016/0010-4655(92)90105-8).
- [12] C.-D. Munz, P. Omnes, R. Schneider, E. Sonnendrücker, U. Voß, Divergence correction techniques for Maxwell solvers based on a hyperbolic model, *J. Comput. Phys.* 161 (2) (2000) 484–511, <https://doi.org/10.1006/jcph.2000.6507>.
- [13] Y. Chen, G. Tóth, Gauss’s law satisfying energy-conserving semi-implicit particle-in-cell method, *J. Comput. Phys.* 386 (2019) 632–652, <https://doi.org/10.1016/j.jcp.2019.02.032>, <http://www.sciencedirect.com/science/article/pii/S0021999119301494>.
- [14] G. Chen, L. Chacón, D. Barnes, An energy- and charge-conserving, implicit, electrostatic particle-in-cell algorithm, *J. Comput. Phys.* 230 (18) (2011) 7018–7036, <https://doi.org/10.1016/j.jcp.2011.05.031>, <http://www.sciencedirect.com/science/article/pii/S0021999111003421>.
- [15] G. Chen, L. Chacón, A multi-dimensional, energy- and charge-conserving, nonlinearly implicit, electromagnetic Vlasov–Darwin particle-in-cell algorithm, *Comput. Phys. Commun.* 197 (2015) 73–87, <https://doi.org/10.1016/j.cpc.2015.08.008>, <http://www.sciencedirect.com/science/article/pii/S0010465515002921>.
- [16] G. Chen, L. Chacón, L. Yin, B. Albright, D. Stark, R. Bird, A semi-implicit, energy- and charge-conserving particle-in-cell algorithm for the relativistic Vlasov–Maxwell equations, *J. Comput. Phys.* 407 (2020) 109228, <https://doi.org/10.1016/j.jcp.2020.109228>.
- [17] D.N. Arnold, R.S. Falk, R. Winther, Finite element exterior calculus, homological techniques, and applications, *Acta Numer.* 15 (2006) 1–155, <https://doi.org/10.1017/S0962492906210018>.

- [18] D.N. Arnold, R.S. Falk, R. Winther, Finite element exterior calculus: from Hodge theory to numerical stability, *Bull. Am. Math. Soc.* 47 (2010) 281–354, <https://doi.org/10.1090/S0273-0979-10-01278-4>.
- [19] A. Buffa, J. Rivas, G. Sangalli, R. Vázquez, Isogeometric discrete differential forms in three dimensions, *SIAM J. Numer. Anal.* 49 (2011) 818–844, <https://doi.org/10.1137/100786708>.
- [20] J. Xiao, H. Qin, J. Liu, Y. He, R. Zhang, Y. Sun, Explicit high-order non-canonical symplectic particle-in-cell algorithms for Vlasov–Maxwell systems, *Phys. Plasmas* 22 (2015) 112504, <https://doi.org/10.1063/1.4935904>.
- [21] J. Xiao, H. Qin, J. Liu, Structure-preserving geometric particle-in-cell methods for Vlasov–Maxwell systems, *Plasma Sci. Technol.* 20 (2018) 110501, <https://doi.org/10.1088/2058-6272/aac3d1>.
- [22] J. Xiao, H. Qin, J. Liu, R. Zhang, Local energy conservation law for a spatially-discretized Hamiltonian Vlasov–Maxwell system, *Phys. Plasmas* 24 (6) (2017) 062112, <https://doi.org/10.1063/1.4986097>.
- [23] J. Squire, H. Qin, W.M. Tang, Geometric integration of the Vlasov–Maxwell system with a variational particle-in-cell scheme, *Phys. Plasmas* 19 (2012) 084501, <https://doi.org/10.1063/1.4742985>.
- [24] O. Gonzalez, Time integration and discrete Hamiltonian systems, *J. Nonlinear Sci.* 6 (5) (1996) 449, <https://doi.org/10.1007/BF02440162>.
- [25] E. Hairer, C. Lubich, G. Wanner, *Geometric Numerical Integration*, Springer, 2006.
- [26] B. Perse, K. Kormann, E. Sonnendrücker, Geometric particle-in-cell simulations of the Vlasov–Maxwell system in curvilinear coordinates, *arXiv:2002.09386*, 2020.
- [27] E.S. Weibel, Spontaneously growing transverse waves in a plasma due to an anisotropic velocity distribution, *Phys. Rev. Lett.* 2 (1959) 83–84, <https://doi.org/10.1103/PhysRevLett.2.83>.
- [28] B.B. Godfrey, Canonical momenta and numerical instabilities in particle codes, *J. Comput. Phys.* 19 (1) (1975) 58–76, [https://doi.org/10.1016/0021-9991\(75\)90116-3](https://doi.org/10.1016/0021-9991(75)90116-3).
- [29] A.B. Langdon, Analysis of the time integration in plasma simulation, *J. Comput. Phys.* 30 (2) (1979) 202–221.