Pseudospectral particle-in-cell formulation with arbitrary charge and current-density time dependencies for the modeling of relativistic plasmas

Olga Shapoval,[1] Edoardo Zoni,[1] Remi Lehe,[1] Maxence Thévenet,[2] and Jean-Luc Vay[1,*]
[1]*Lawrence Berkeley National Laboratory, Berkeley, California 94720, USA*
[2]*Deutsches Elektronen-Synchrotron DESY, Notkestrasse 85, 22607 Hamburg, Germany*

(Received 2 January 2024; revised 10 April 2024; accepted 12 July 2024; published 23 August 2024)

This paper introduces a formulation of the particle-in-cell (PIC) method for the modeling of relativistic plasmas, that leverages the ability of the pseudospectral analytical time-domain solver (PSATD) to handle arbitrary time dependencies of the charge and current densities during one PIC cycle (applied to second-order polynomial dependencies here). The formulation is applied to a modified set of Maxwell's equations that was proposed earlier in the context of divergence cleaning, and to recently proposed extensions of the PSATD-PIC algorithm. Detailed analysis and testings revealed that, under some condition, the formulation can expand the range of numerical parameters under which PIC simulations are stable and accurate when modeling relativistic plasmas such as, e.g., plasma-based particle accelerators.

DOI: 10.1103/PhysRevE.110.025206

I. INTRODUCTION

Simulations of relativistic plasmas often rely on the electromagnetic particle-in-cell (PIC) method [1–3], with variations of the method that have been proposed and are chosen based on the application. For the modeling of plasma-based accelerators [4,5], a variation that has gained in popularity uses the "infinite-order" (in space and time) pseudospectral analytical time-domain solver (PSATD) method [6,7], instead of the (almost universally adopted) second-order (in space and time) finite-difference time-domain (FDTD) "Yee" method [8], to solve Maxwell's equations at discrete points in space and time. In contrast to the Yee solver, the PSATD solver offers no numerical dispersion and no Courant condition on the field solve. Extensions of the PSATD PIC method includes the use of finite-order spatial stencils [9,10], alternating nodal-staggered representations of the field quantities during one PIC loop [11], time-averaging of the fields gathered onto the particles [12], and integration of the equations in a Galilean frame moving at a given velocity (a.k.a. Galilean PSATD PIC or Galilean PIC) [13,14]. The combination of the Galilean PIC method with the other extensions has led to stable modeling of plasma accelerators [13,14], free of the numerical Cherenkov instability (NCI) [15] when using the Lorentz boosted frame method to speed up simulations [16]. In some cases, however, the method, which relies on the user setting a predefined Galilean velocity, can become inaccurate when it cannot be assumed that the local plasma velocity is close to that predefined velocity. As a possible remedy, this paper introduces and starts exploring a formulation of the PIC algorithm where the standard assumption that the current density that is produced by the particles is constant over a time step is relaxed.

*Contact author: jlvay@lbl.gov

The reminder of the paper is organized as follows. The formulation of the algorithm is derived first in Sec. II A, followed by the presentation of its finite-order stencil, alternating nodal-staggered and time-averaged extensions in Sec. II B. The connection between the algorithm and the Galilean PIC formulation is discussed next in Sec. II C. The effectiveness of the algorithm at mitigating the NCI is then explored theoretically and numerically on a simple uniform plasma case in Sec. III A. Finally, the scheme is tested in simulations of laser-plasma accelerators in a Lorentz boosted frame in Sec. III B.

II. NEW PIC-JRhom ALGORITHM

A. Presentation of the algorithm

The following modified system of Maxwell's equations is considered:

$$\frac{\partial \boldsymbol{E}}{\partial t} = c^2 \boldsymbol{\nabla} \times \boldsymbol{B} - \frac{\boldsymbol{J}}{\varepsilon_0} + c^2 \boldsymbol{\nabla} F, \tag{1a}$$

$$\frac{\partial \boldsymbol{B}}{\partial t} = -\boldsymbol{\nabla} \times \boldsymbol{E}, \tag{1b}$$

$$\frac{\partial F}{\partial t} = \boldsymbol{\nabla} \cdot \boldsymbol{E} - \frac{\rho}{\varepsilon_0}. \tag{1c}$$

In addition to the usual Maxwell-Faraday and Ampère-Maxwell equations, the system contains an extra equation for the scalar field F, which propagates deviations to Gauss' law. (Note that, in the case where Gauss' law is verified in the PIC simulation, Eq. (1c) leads to $F = 0$, and Eqs. (1a) and (1b) reduce to the standard Maxwell's equations.) These additional terms were introduced in Ref. [17] from the potential formulation in the Lorentz gauge and used as a propagative divergence cleaning procedure, as an alternate to the Langdon-Marder [18] or Marder [19] diffusive ones. This type of divergence cleaning was also proposed independently and analyzed more formally in Ref. [20]. A connection to the

formulation of Eqs. (1) in potential form, derived more formally than in Ref. [17], is instructive and given in Appendix A.

While the abovementioned earlier work [17,20] considered this formulation in the context of the standard PIC method using FDTD discretization of Eqs. (1), this article focuses on the PSATD [1,6,7] discretization of Eqs. (1), where the equations are integrated analytically over one time step, in Fourier space. The expression of Eqs. (1) in Fourier space reads

$$\frac{\partial \widehat{\boldsymbol{E}}}{\partial t} = ic^2\boldsymbol{k}\times\widehat{\boldsymbol{B}} - \frac{\widehat{\boldsymbol{J}}}{\varepsilon_0} + ic^2\widehat{F}\boldsymbol{k}, \tag{2a}$$

$$\frac{\partial \widehat{\boldsymbol{B}}}{\partial t} = -i\boldsymbol{k}\times\widehat{\boldsymbol{E}}, \tag{2b}$$

$$\frac{\partial \widehat{F}}{\partial t} = i\boldsymbol{k}\cdot\widehat{\boldsymbol{E}} - \frac{\widehat{\rho}}{\varepsilon_0}, \tag{2c}$$

where $\widehat{f}$ denotes the Fourier transform of function f. The analytical integration of Eqs. (2) in time requires an assumption on the time dependency of the current and charge densities $\widehat{\boldsymbol{J}}$ and $\widehat{\rho}$ over the integration interval, i.e., over a time step that goes from $t = n\Delta t$ to $t = (n+1)\Delta t$. In the standard PSATD algorithm [6,7], $\widehat{\boldsymbol{J}}$ is assumed to constant in time, and $\widehat{\rho}$ is assumed to be linear in time, within a given time step Δt.

This paper considers more general time dependencies for $\widehat{\boldsymbol{J}}$ and $\widehat{\rho}$ within one time step, which is divided into m subintervals of equal size $\delta t = \Delta t/m$. During these subintervals, $\widehat{\boldsymbol{J}}$ and $\widehat{\rho}$ are considered to be either piecewise constant, piecewise linear, or piecewise quadratic in time. This is illustrated in Fig. 1. In the rest of this paper, the notation "PIC-JRhom" is used, where J and Rho (J, Rho $\in$ {C (constant), L (linear), Q (quadratic)}) indicate the (piecewise) time dependency of the current density $\widehat{\boldsymbol{J}}$ and charge density $\widehat{\rho}$, respectively, and m is the number of subintervals. For example, "PIC-LL2" refers to the PIC algorithm with linear time dependency of both $\widehat{\boldsymbol{J}}$ and ρ and 2 subintervals. Note that, in this notation, "PIC-CL1" refers to the standard PSATD PIC algorithm [7], where $\widehat{\boldsymbol{J}}$ is constant and $\widehat{\rho}$ is linear in time over one time step.

More specifically for each ℓth time subinterval $\ell \in \mathbb{Z} \cap [0, m-1]$:

(1) When $\widehat{\rho}(t)$ is assumed to be piecewise constant: macroparticles deposit their charge density in the middle of each time subinterval, i.e., at $t_{n+(\ell+1/2)/m} \equiv n\Delta t + (\ell + 1/2)\delta t$, and $\widehat{\rho}$ is then assumed to be constant in each subinterval:

$$\widehat{\rho}(t) = \rho^{n+(\ell+1/2)/m}, \quad t \in [n\Delta t + \ell\delta t, n\Delta t + (\ell+1)\delta t].$$

(2) When $\widehat{\rho}(t)$ is assumed to be piecewise linear: macroparticles deposit their charge density at the edge of each time subinterval, i.e., at $t_{n+\ell/m} \equiv n\Delta t + \ell\delta t$ and $t_{n+(\ell+1)/m} \equiv n\Delta t + (\ell+1)\delta t$, and $\widehat{\rho}$ is then assumed to be linear in each subinterval:

$$\widehat{\rho}(t) = \frac{\widehat{\rho}^{\,n+(\ell+1)/m} - \widehat{\rho}^{\,n+\ell/m}}{\delta t}(t - t_{n+(\ell+1/2)/m}) + \frac{\widehat{\rho}^{\,n+(\ell+1)/m} + \widehat{\rho}^{\,n+\ell/m}}{2},$$
$$t \in [n\Delta t + \ell\delta t, n\Delta t + (\ell+1)\delta t].$$

(3) When $\widehat{\rho}(t)$ is assumed to be piecewise quadratic: macroparticles deposit their charge density at the middle *and* edge of each time subinterval, i.e., at $t_{n+(\ell+1/2)/m}$, and at $t_{n+\ell/m}$ and $t_{n+(\ell+1)/m}$. $\widehat{\rho}$ is then assumed to be quadratic in each subinterval:

$$\widehat{\rho}(t) = \frac{2(\rho^{n+(\ell+1)/m} - 2\widehat{\rho}^{\,n+(\ell+1/2)/m} + \rho^{n+\ell/m})}{\delta t^2} \times (t - t_{n+(\ell+1/2)/m})^2 + \frac{\widehat{\rho}^{\,n+(\ell+1)/m} - \widehat{\rho}^{\,n+\ell/m}}{\delta t} \times (t - t_{n+(\ell+1/2)/m}) + \rho^{n+(\ell+1/2)/m},$$
$$t \in [n\Delta t + \ell\delta t, n\Delta t + (\ell+1)\delta t],$$

with similar definitions for $\widehat{\boldsymbol{J}}$, when $\widehat{\boldsymbol{J}}(t)$ is assumed to be piecewise constant, piecewise linear, or piecewise quadratic, respectively.

Overall, the time dependency of $\widehat{\boldsymbol{J}}$ and $\widehat{\rho}$ can thus be expressed, for $t \in [n\Delta t + \ell\delta t, n\Delta t + (\ell+1)\delta t]$, with $\ell \in [0, m-1]$, as

$$\widehat{\boldsymbol{J}}(t) = \frac{2\boldsymbol{a}_{\boldsymbol{J}}^{\tau}}{\delta t^2}(t - t_{n+(\ell+1/2)/m})^2 + \frac{\boldsymbol{b}_{\boldsymbol{J}}^{\tau}}{\delta t}(t - t_{n+(\ell+1/2)/m}) + \boldsymbol{c}_{\boldsymbol{J}}^{\tau}, \tag{3a}$$

$$\widehat{\rho}(t) = \frac{2a_{\rho}^{\tau}}{\delta t^2}(t - t_{n+(\ell+1/2)m})^2 + \frac{b_{\rho}^{\tau}}{\delta t}(t - t_{n+(\ell+1/2)m}) + c_{\rho}^{\tau}, \tag{3b}$$

where the coefficients of the polynomials are given in Table I.

It is important to note that the particles' momenta are not updated during one time step, i.e., the proposed scheme does *not* involve subcycling of the macroparticles motion. As in standard PSATD PIC, macroparticles move in straight line from their known position at $t_n = n\Delta t$ to time t, using their known momentum at $t_{n+1/2}$:

$$\boldsymbol{x}(t) = \boldsymbol{x}^n + \frac{\boldsymbol{p}^{n+1/2}}{m\sqrt{1 + (\boldsymbol{p}^{n+1/2}/mc)^2}}(t - t_n),$$

where $\boldsymbol{x}^n$ and $\boldsymbol{p}^{n+1/2}$ follow the standard leap-frog time stepping that is commonly used in PIC simulations. Thus, here, even though the charge and current density may be deposited several times per time step Δt, the macroparticles' momentum $\boldsymbol{p}$ is only updated once per time step, and therefore the fields $\boldsymbol{E}$ and $\boldsymbol{B}$ are gathered onto macroparticles to update $\boldsymbol{p}$ only once per time step also.

While charge-conserving deposition can be used readily with schemes CLn, since the time dependency of the current is the derivative of the time dependency of the charge density, as required by the continuity equation, there are no obvious such schemes for CCn, LLn, and QQn. Hence, exact charge conservation is not addressed in the present work and direct deposition of charge and current densities, i.e., $\rho = \sum S(\boldsymbol{x}_p - \boldsymbol{x}_{i,j,k})q_p/V$ and $J = \sum S(\boldsymbol{x}_p - \boldsymbol{x}_{i,j,k})q_p\boldsymbol{v}_p/V$, where q_p, $\boldsymbol{x}_p$, and $\boldsymbol{v}_p$ are, respectively, the particles charge, positions, and velocities and $V = \Delta x\Delta y\Delta z$ is the volume of a grid cell, are used for all simulations.

Using the piecewise definition of $\widehat{\rho}$ and $\widehat{\boldsymbol{J}}$ given in Eqs. (3), Eqs. (2) can be integrated analytically over one time step Δt, i.e., from $t = n\Delta t$ to $t = (n+1)\Delta t$. In practice, this is

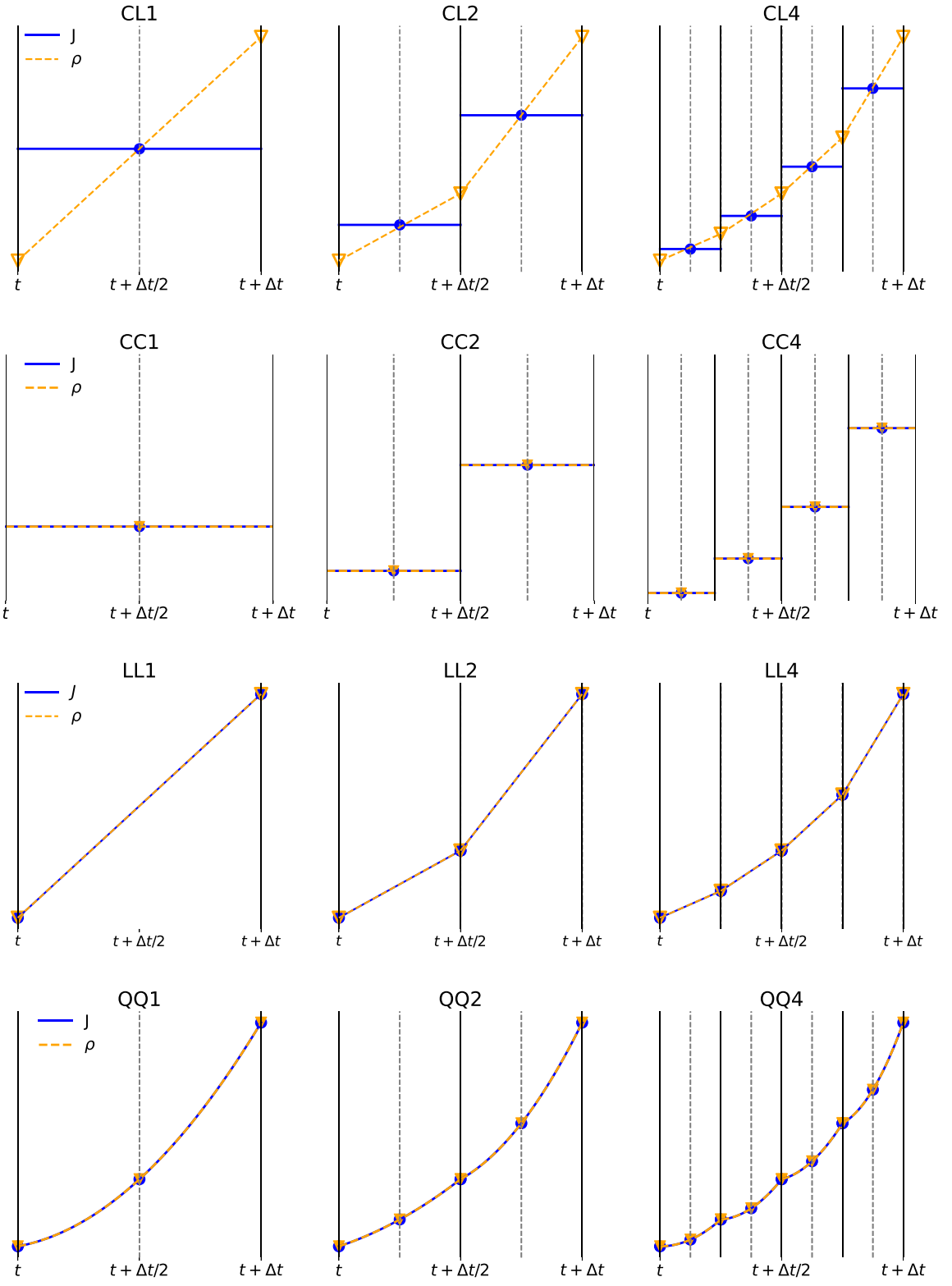

FIG. 1. Diagrams illustrating various time dependencies of the current density $\boldsymbol{J}$ and charge density ρ for constant/linear (CL), both constant (CC), linear (LL), and quadratic (QQ) dependencies with m subintervals: (first column) $m = 1$, (second) $m = 2$, and (third) $m = 4$. CL1 corresponds to the standard PSATD PIC method. The triangle and circle glyphs represent the times at which the macroparticles deposit ρ and $\boldsymbol{J}$ on the grid, respectively. The dashed and solid lines represent the assumed time dependency of ρ and $\boldsymbol{J}$ within one time step, when integrating the Maxwell equations analytically.

TABLE I. Polynomial coefficients (**PC**), based on the time dependency of the current and charge densities $\widehat{\boldsymbol{J}}$ and $\widehat{\rho}$ over ℓth time subinterval $[n\Delta t + \ell\delta t, n\Delta t + (\ell+1)\delta t]$.

	Time dependency of $\widehat{\boldsymbol{J}}$ or $\widehat{\rho}$		
PC	Constant ($\tau = 0$)	Linear ($\tau = 1$)	Quadratic ($\tau = 2$)
$\boldsymbol{a}_J^\tau$	0	0	$\widehat{\boldsymbol{J}}^{n+(\ell+1)/m} - 2\widehat{\boldsymbol{J}}^{n+(\ell+1/2)/m} + \widehat{\boldsymbol{J}}^{n+\ell/m}$
$\boldsymbol{b}_J^\tau$	0	$\widehat{\boldsymbol{J}}^{n+(\ell+1)/m} - \widehat{\boldsymbol{J}}^{n+\ell/m}$	$\widehat{\boldsymbol{J}}^{n+(\ell+1)/m} - \widehat{\boldsymbol{J}}^{n+\ell/m}$
$\boldsymbol{c}_J^\tau$	$\widehat{\boldsymbol{J}}^{n+(\ell+1/2)/m}$	$(\widehat{\boldsymbol{J}}^{n+(\ell+1)/m} + \widehat{\boldsymbol{J}}^{n+\ell/m})/2$	$\widehat{\boldsymbol{J}}^{n+(\ell+1/2)/m}$
$a_{\widehat{\rho}}^\tau$	0	0	$\widehat{\rho}^{\,n+(\ell+1)/m} - 2\widehat{\rho}^{\,n+(\ell+1/2)m} + \widehat{\rho}^{\,n+\ell/m}$
$b_{\widehat{\rho}}^\tau$	0	$\widehat{\rho}^{\,n+(\ell+1)/m} - \widehat{\rho}^{\,n+\ell/m}$	$\widehat{\rho}^{\,n+(\ell+1)/m} - \widehat{\rho}^{\,n+\ell/m}$
$c_{\widehat{\rho}}^\tau$	$\widehat{\rho}^{\,n+(\ell+1/2)/m}$	$(\widehat{\rho}^{\,n+(\ell+1)/m} + \widehat{\rho}^{\,n+\ell/m})/2$	$\widehat{\rho}^{\,n+(\ell+1/2)m}$

done by sequentially integrating these equations over each subinterval $\ell \in [0, m-1]$:

$$\widehat{\boldsymbol{E}}^{n+(\ell+1)/m} = C\widehat{\boldsymbol{E}}^{n+\ell/m} + ic^2\frac{S}{ck}\boldsymbol{k}\times\widehat{\boldsymbol{B}}^{n+\ell/m} + ic^2\frac{S}{ck}\widehat{F}^{n+\ell/m}\boldsymbol{k} + \frac{1}{\varepsilon_0 ck}(Y_3\boldsymbol{a}_J + Y_2\boldsymbol{b}_J - S\boldsymbol{c}_J) + \frac{ic^2}{\varepsilon_0 c^2k^2}(Y_1 a_\rho - Y_5 b_\rho - Y_4 c_\rho)\boldsymbol{k}, \tag{4a}$$

$$\widehat{\boldsymbol{B}}^{n+(\ell+1)/m} = C\widehat{\boldsymbol{B}}^{n+\ell/m} - i\frac{S}{ck}\boldsymbol{k}\times\widehat{\boldsymbol{E}}^{n+\ell/m} - \frac{i}{\varepsilon_0 c^2k^2}\boldsymbol{k}\times(Y_1\boldsymbol{a}_J - Y_5\boldsymbol{b}_J - Y_4\boldsymbol{c}_J), \tag{4b}$$

$$\widehat{F}^{n+(\ell+1)/m} = C\widehat{F}^{n+\ell/m} + i\frac{S}{ck}\boldsymbol{k}\cdot\widehat{\boldsymbol{E}}^{n+\ell/m} + \frac{i}{\varepsilon_0 c^2k^2}\boldsymbol{k}\cdot(Y_1\boldsymbol{a}_J - Y_5\boldsymbol{b}_J - Y_4\boldsymbol{c}_J) + \frac{1}{\varepsilon_0 ck}(Y_3 a_\rho + Y_2 b_\rho - S c_\rho), \tag{4c}$$

where

$$C = \cos(ck\delta t),\ S = \sin(ck\delta t),$$
$$Y_1 = \frac{(1-C)(8-c^2k^2\delta t^2) - 4Sck\delta t}{2c^2k^2\delta t^2},$$
$$Y_2 = \frac{2(C-1) + Sck\delta t}{2ck\delta t},$$
$$Y_3 = \frac{S(8-c^2k^2\delta t^2) - 4ck\delta t(1+C)}{2c^2k^2\delta t^2},$$
$$Y_4 = (1-C),\quad Y_5 = \frac{(1+C)ck\delta t - 2S}{2ck\delta t}. \tag{5}$$

The steps of the PIC-JRhom cycle with sub-time-stepping are summarized in the diagram shown in Fig. 2.

Assuming that the electric and magnetic fields are known from the previous time step at iteration n, the particles velocities are pushed from $n-1/2$ to $n+1/2$, then the positions from n to $n+1$. The charge and current densities are then obtained at every substep using direct deposition, as described above, assuming that the velocities are constant and the positions evolve linearly over the interval $n \to n+1$.

It is important to note that the algorithm preserves the property of absence of spurious self-force when using the same shape factor for field gather as for charge and current deposition, as with standard PIC. Further discussion of the property preservation and tests are given in Appendix B.

B. Extensions

As shown in Refs. [9–12], the PSATD PIC algorithm can be extended to (a) arbitrary-order spatial stencils, (b) a scheme that alternates between nodal and staggered representations of the field components on the simulation grid, and (c) a scheme that averages the fields to be gathered over one timestep. Such extensions are presented in the next sections for the PSATD PIC-JRhom algorithm.

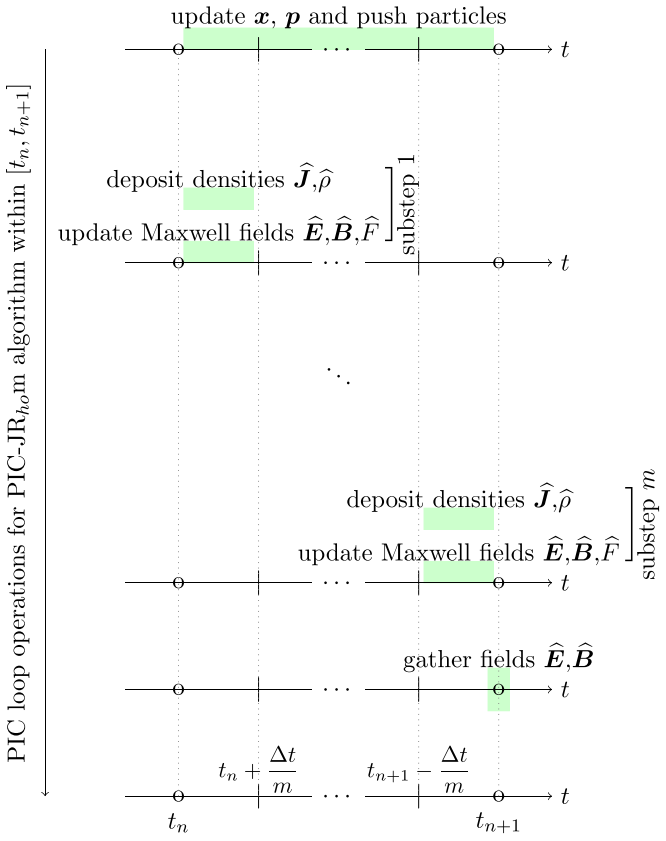

FIG. 2. Diagram of the PIC-JRhom algorithm.

1. Extension to finite-order stencils

When using domain decomposition to run PSATD PIC methods on parallel computers, it is advantageous to alter the wave vector in the Fourier representation of the equations to emulate a finite-difference approximation of the spatial derivatives at a finite order p, since this enhances the locality of the field solvers and thus reduces the required number of guard cells around each subdomain [9,10]. The modified $[k_u^p]$ at order p along the direction $u = x, y, z$ are then given by

$$\left[k_u^p\right]_{\text{nodal}} = \sum_{j=1}^{p/2} \left[\alpha_j^p\right]_{\text{nodal}} \frac{\sin(k_u j\Delta u)}{j\Delta u}, \quad u = x, y, z, \tag{6a}$$

$$\left[k_u^p\right]_{\text{staggered}} = \sum_{j=1}^{p/2} \left[\alpha_j^p\right]_{\text{staggered}} \frac{\sin(k_u (j-1/2)\Delta u)}{(j-1/2)\Delta u}, \quad u = x, y, z, \tag{6b}$$

for a nodal and staggered representation, respectively, with the following Fornberg coefficients [21]:

$$\left[\alpha_j^p\right]_{\text{nodal}} = (-1)^{j+1} \frac{2[(p/2)!]^2}{(p/2-j)!(p/2+j)!}, \tag{7a}$$

$$\left[\alpha_j^p\right]_{\text{staggered}} = (-1)^{j+1} \left[\frac{p!}{2^p (p/2)!}\right]^2 \times \frac{4}{(2j-1)(p/2-j)!(p/2+j-1)!}. \tag{7b}$$

These modified wave numbers can be readily used with the PIC-JRhom algorithm to limit the number of guard cells and enable efficient parallel simulations, just as with other flavors of PSATD PIC algorithms [9,10].

2. Extension to alternating nodal-staggered grids

Just like the standard and averaged formulations of PSATD PIC, the PIC-JRhom algorithm can readily adopt the "hybrid nodal-staggered" scheme presented in Ref. [11] where the field alternate between nodal and staggered representations on the simulation grid. More precisely, the Maxwell solve and guard cell exchanges are performed on a staggered "Yee" grid while the charge/current depositions and fields gather are performed with field quantities on a separate nodal grid. This "hybrid" alternating nodal-staggered extension allows to retain the advantages of low numerical dispersion and compact stencils of the integration of Maxwell's equations on a staggered grid with the stability associated with the interpolation of fields onto the particles from a nodal grid [11] (especially for NCI-prone boosted-frame simulations). The application of the "hybrid" alternating nodal-staggered scheme to PIC-JRhom leads to the steps shown in Fig. 3.

3. Extension to the time-averaged PSATD PIC algorithm

In Ref. [12], an extension to PSATD PIC, named time-averaged PSATD PIC (also labeled as averaged PIC for convenience), is presented that enables stable boosted-frame simulations even when the time step is larger than the Courant condition along a given axis, e.g., $c\Delta t = \Delta z > \Delta x$. With the time-averaged algorithm, the field quantities that are gathered onto the particles are given by time averages of the fields on the grid obtained by analytically integrating the $\widehat{\boldsymbol{E}}$ and $\widehat{\boldsymbol{B}}$ fields from $t = n\Delta t$ to $t = (n+2)\Delta t$. The time-averaged PIC-JRhom algorithm consists of the steps shown in Fig. 4, where the analytical average of $\widehat{\boldsymbol{E}}$ and $\widehat{\boldsymbol{B}}$ at time $t = (n+1)\Delta t$ are

$$\left\langle \widehat{\boldsymbol{E}}^{n+1} \right\rangle = \frac{1}{2\Delta t} \sum_{\ell=0}^{2m-1} \left[\frac{S}{ck} \widehat{\boldsymbol{E}}^{\,n+\ell/m} + \frac{ic^2 Y_4}{c^2k^2} \boldsymbol{k} \times \widehat{\boldsymbol{B}}^{\,n+\ell/m} + \frac{i\boldsymbol{k} Y_4}{2ck\delta t} \widehat{F}^{n+\ell/m} + \frac{1}{\varepsilon_0 c^2 k^2} \left(Y_1 \boldsymbol{a}_J^\tau - Y_5 \boldsymbol{b}_J^\tau - Y_4 \boldsymbol{c}_J^\tau\right) - ic^2 \boldsymbol{k} \left(Y_6 a_\rho^\tau + Y_7 b_\rho^\tau + Y_8 c_\rho^\tau\right) \right], \tag{8a}$$

$$\left\langle \widehat{\boldsymbol{B}}^{n+1} \right\rangle = \frac{1}{2\Delta t} \sum_{\ell=0}^{2m-1} \left[\frac{S}{ck} \widehat{\boldsymbol{B}}^{n+\ell/m} - \frac{iY_4}{c^2k^2} \boldsymbol{k} \times \widehat{\boldsymbol{E}}^{n+\ell/m} + i\boldsymbol{k} \times \left(Y_6 \boldsymbol{a}_J^\tau + Y_7 \boldsymbol{b}_J^\tau + Y_8 \boldsymbol{c}_J^\tau\right) \right]. \tag{8b}$$

For a detailed derivation see Appendix D.

C. Relation to the Galilean PSATD PIC algorithm

This section examines the relationship between the Galilean PIC algorithm, the standard PSATD PIC algorithm and the PIC-JRhom algorithm. To this end, it is instructive to "deconstruct" the Galilean PIC algorithm by separating it in two independent steps: (i) a shift of the quantities to recenter them on a grid moving at $\boldsymbol{v}_{\text{gal}}$, (ii) the integration of the PSATD equations assuming that the current source is constant along the flow moving at the Galilean velocity $\boldsymbol{v}_{\text{gal}}$.

The standard Galilean PIC scheme [13,14] can then be written highlighting terms that arise from step (i) in red in Eqs. (9a) and (9b) and those from step (ii) in blue in Eqs. (10a)–(10d) and (11):

$$\widehat{\boldsymbol{B}}^{n+1} = C\theta^2 \widehat{\boldsymbol{B}}^n - i\frac{S}{\omega} \boldsymbol{k} \times (\theta^2 \widehat{\boldsymbol{E}}^n) + i X_1 \boldsymbol{k} \times \theta \widehat{\boldsymbol{J}}^{n+\frac{1}{2}}, \tag{9a}$$

$$\widehat{\boldsymbol{E}}^{n+1} = C\theta^2 \widehat{\boldsymbol{E}}^n + ic^2 \frac{S}{\omega} \boldsymbol{k} \times (\theta^2 \widehat{\boldsymbol{B}}^n) + X_4 \theta \widehat{\boldsymbol{J}}^{n+\frac{1}{2}} + i\left(X_3 \theta^2 \widehat{\rho}^{\,n} - X_2 \widehat{\rho}^{\,n+1}\right) \boldsymbol{k}, \tag{9b}$$

where the coefficients X_1, X_2, X_3, and X_4 are defined as

$$X_1 := \frac{1}{\varepsilon_0(\omega^2 - \Omega^2)} \left(\theta^* - \theta C + i\Omega\theta \frac{S}{\omega}\right), \tag{10a}$$

$$X_2 := \frac{c^2}{\theta^* - \theta} \left(\theta^* \frac{\chi_1}{\varepsilon_0 \omega^2} - \theta \frac{1-C}{\varepsilon_0 \omega^2}\right), \tag{10b}$$

$$X_3 := \frac{c^2}{\theta^* - \theta} \left(\theta^* \frac{\chi_1}{\varepsilon_0 \omega^2} - \theta^* \frac{1-C}{\varepsilon_0 \omega^2}\right), \tag{10c}$$

$$X_4 := i\Omega X_1 - \frac{\theta}{\varepsilon_0} \frac{S}{\omega}, \tag{10d}$$

with

$$\chi_1 := \frac{\omega^2}{\omega^2 - \Omega^2} \left(\theta^* - \theta C + i\Omega\theta \frac{S}{\omega}\right), \tag{11}$$

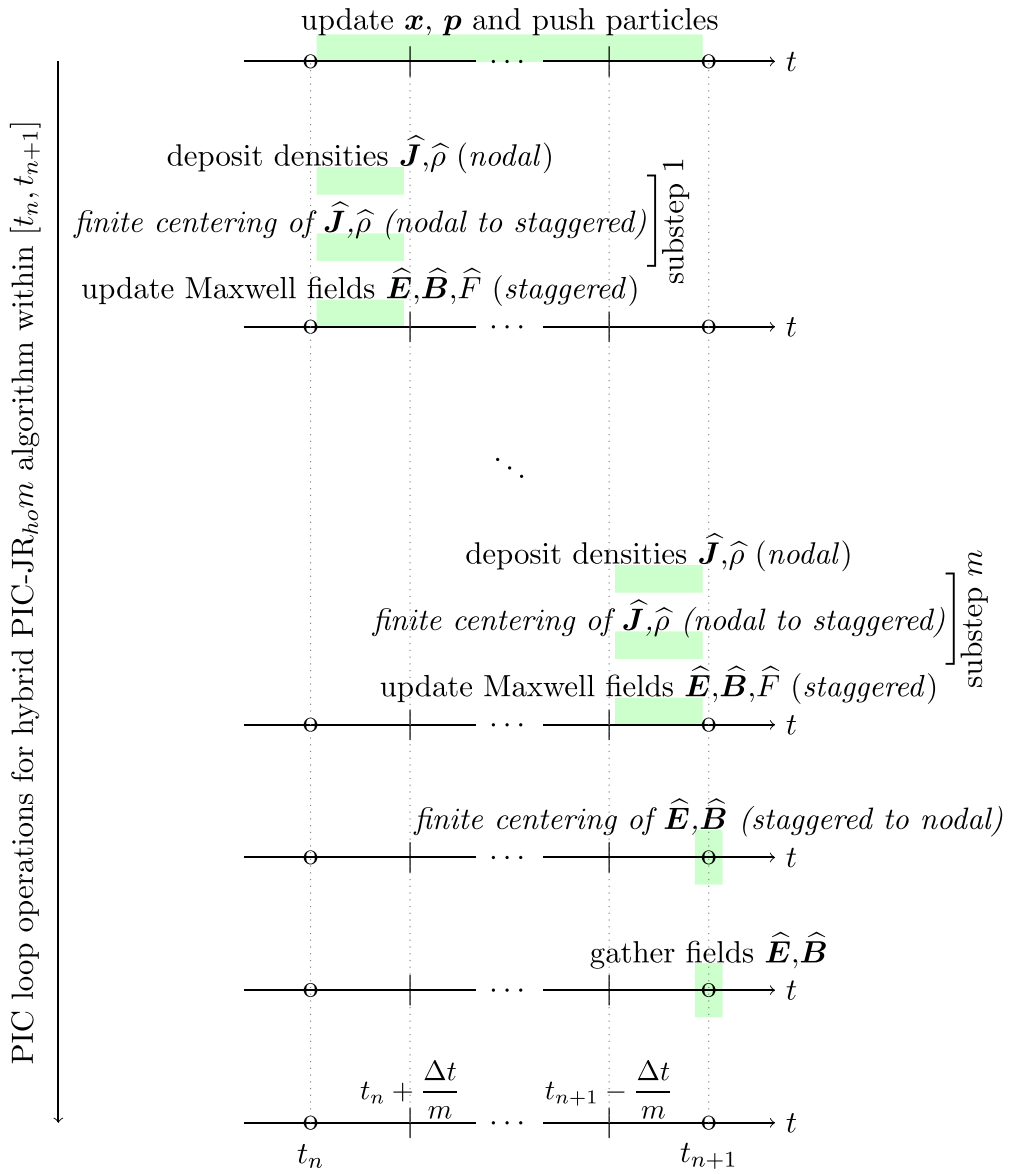

FIG. 3. Diagram of the alternating nodal-staggered PIC-JRho*m* algorithm.

where $\Omega := \boldsymbol{v}_{\text{gal}} \cdot \boldsymbol{k}$, $\omega := c\,k$, $C := \cos(\omega\,\Delta t)$, $S := \sin(\omega\,\Delta t)$, $\theta := e^{i\Omega\Delta t/2}$, and $\theta^* := e^{-i\Omega\Delta t/2}$.

When setting $\boldsymbol{v}_{\text{gal}} = 0$, the system (9a)–(11) converges to the standard PSATD algorithm, as expected.

Step (i), which corresponds to the multiplication of some of the terms by θ or θ^2, in red in Eqs. (9a) and (9b), is the easiest to interpret: noting that a multiplication by $\theta := e^{i\Omega\Delta t/2}$ in Fourier space corresponds to shifting the terms spatially by the distance $\boldsymbol{v}_{\text{gal}}\Delta t/2$ in real space, the terms known at time $n + 1/2$ are multiplied by θ, hence shifted by $\boldsymbol{v}_{\text{gal}}\Delta t/2$ while the terms known at time n are multiplied by θ^2, hence shifted by $\boldsymbol{v}_{\text{gal}}\Delta t$. These are exactly the shifts that are needed to bring the corresponding quantities to their new grid location after one time step, when assuming a Galilean frame of reference moving at $\boldsymbol{v}_{\text{gal}}$.

Understanding the terms associated with step (ii) requires a more detailed comparison between how the standard and the Galilean PIC equations are obtained. The standard PSATD algorithm is derived assuming that the current density (source term) is constant over one time step on a fixed grid. The Galilean algorithm makes the same assumption but in a Galilean frame, i.e., that the current density (source term) is constant over one time step on a Galilean grid. Following this comparison, it flows logically that step (ii) ought to correspond to an integration of the PSATD equations on a fixed grid assuming that the currents are constant along a segment of length $\boldsymbol{v}_{\text{gal}}\Delta t$. Indeed, it was verified that integrating the PSATD equations based on these assumptions leads to the system (9a)–(11) with the terms highlighted in red replaced by 1 in Eqs. (9a) and (9b).

From this, it follows that the algorithm PIC-JRhom is related to step (ii) of the Galilean PIC algorithm in the following way. While step (ii) of Galilean PIC provides a more accurate analytical integration of the PSATD equations over one time step for a flow that moves uniformly at v_{gal}, the PIC-JRhom, with its arbitrary time-dependence of $\boldsymbol{J}$ and ρ and its subintervals, provides a more accurate analytical integration

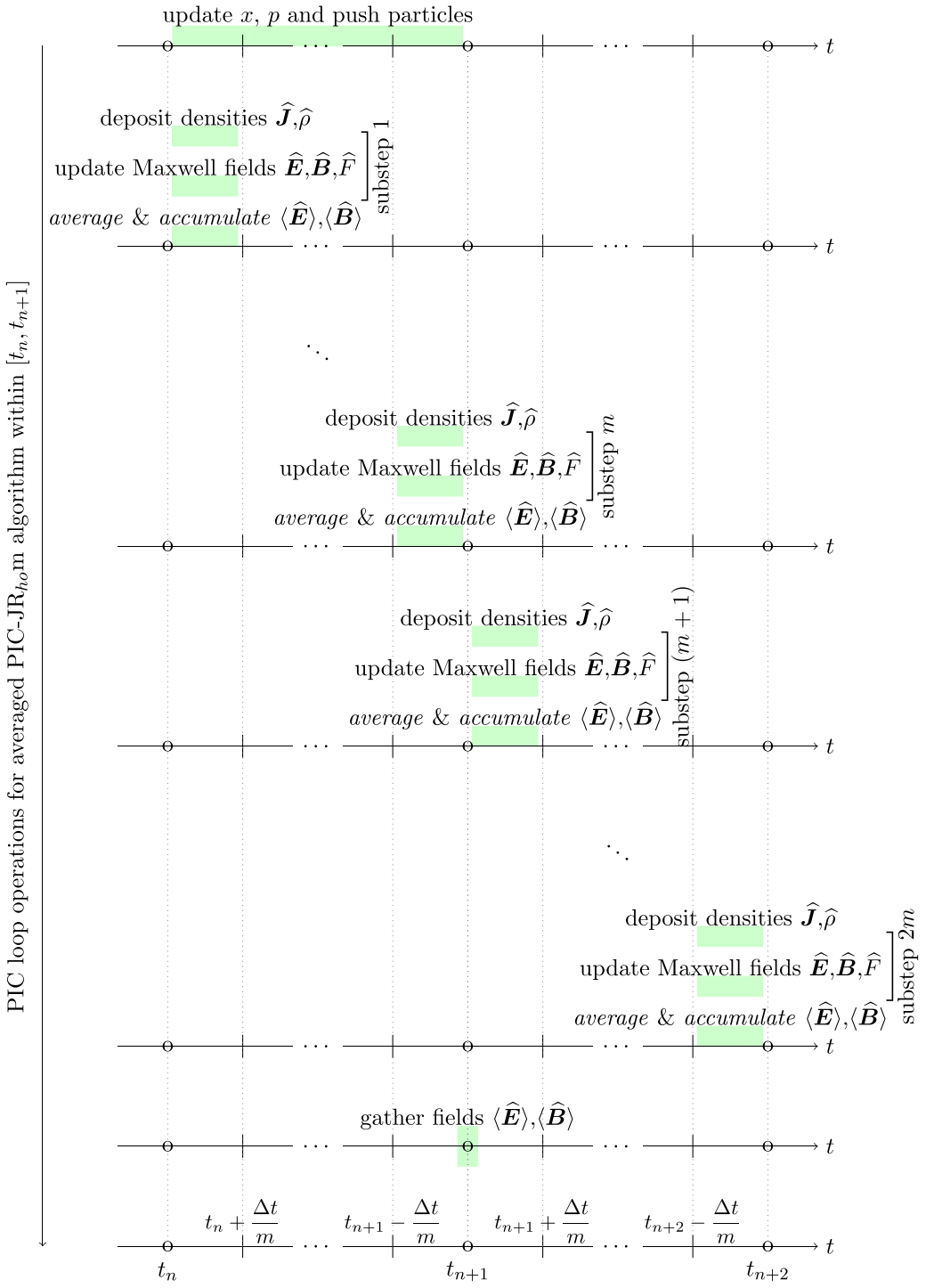

FIG. 4. Diagram of the time-averaged PIC-JRho*m* algorithm.

of the PSATD equations over one time step for an arbitrary local flow of particles. The PIC-JRhom algorithm can thus be viewed as a possible generalization of step (ii) of the Galilean PIC algorithm. Indeed, the numerical tests discussed below show that, like the Galilean PIC algorithm, PIC-JRhom can lead to simulations that are very stable with regard to the numerical Cherenkov instability, and that it can also remain accurate in cases where the Galilean assumption is becoming less appropriate.

III. NUMERICAL TESTS

This section presents various physics applications to test the PIC-JRhom algorithm. All simulations and results have been performed and obtained with the open-source electromagnetic PIC code WarpX [22–24]. The current implementation provides the flexibility to:

(1) Choose an arbitrary polynomial time dependency of $\boldsymbol{J}$ and ρ among the following combinations:

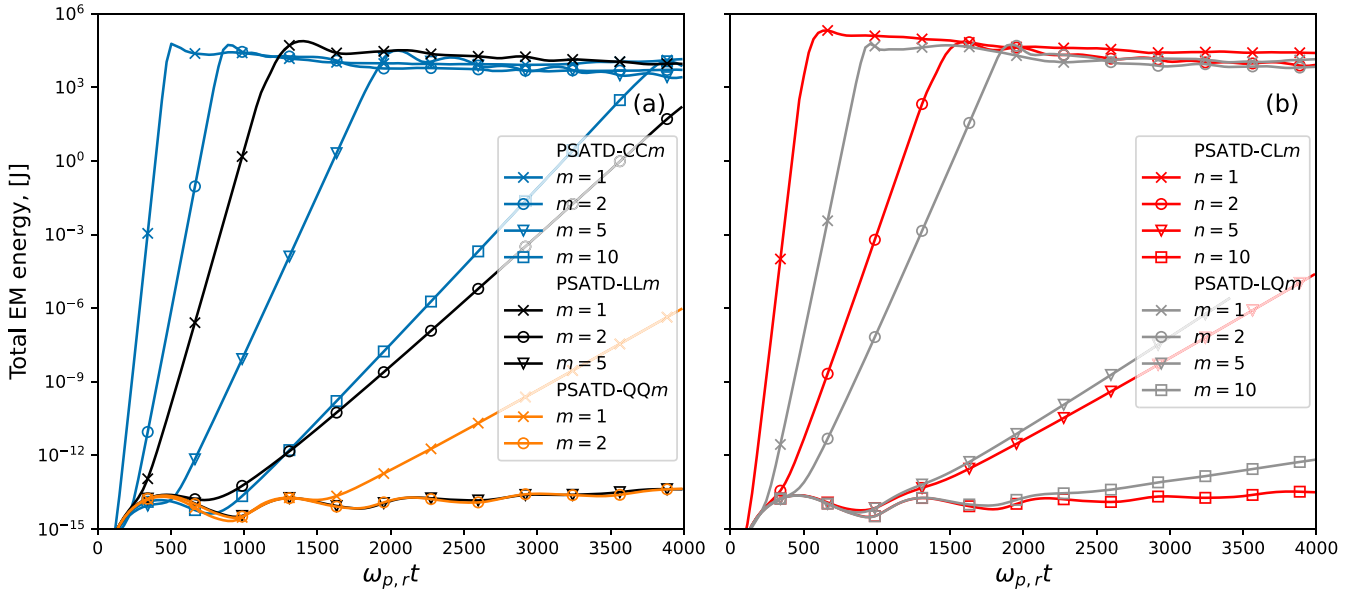

FIG. 5. WarpX simulations of a uniform plasma with a time step at the Courant condition limit $c\Delta t = \Delta x = \Delta z$ and a stencil at infinite order, without divergence cleaning. The total electromagnetic (EM) energy of a uniform plasma drifting at relativistic velocity $\boldsymbol{v}_0$ along the z axis is plotted versus the time of the simulation with (a) the same time-dependencies for $\boldsymbol{J}$ and ρ and (b) different time-dependencies for $\boldsymbol{J}$ and ρ, for various combinations of time-dependency and number of time-step subintervals. Here, $\omega_{p,r} = \omega_p/\sqrt{\gamma_0}$ is the relativistic plasma frequency, where time $\omega_{p,r}t = 4000$ corresponds to roughly 6.7×10^4 time steps.

(a) $\boldsymbol{J}$ and ρ constant in time (CCm);
(b) $\boldsymbol{J}$ constant in time and ρ linear in time (CLm);
(c) $\boldsymbol{J}$ and ρ linear in time (LLm);
(d) $\boldsymbol{J}$ and ρ quadratic in time (QQm);

(2) Choose the number of subintervals m within one time step;

(3) Turn on/off the divergence cleaning term, that is, solve Maxwell's equations (1) with or without the scalar field F;

(4) Turn on/off the time averaging of the $\boldsymbol{E}$ and $\boldsymbol{B}$ fields gathered on the macro-particles, as in Eq. (8).

To assess the stability of the PIC-JRhom method theoretically, the analytical dispersion equation was derived (see

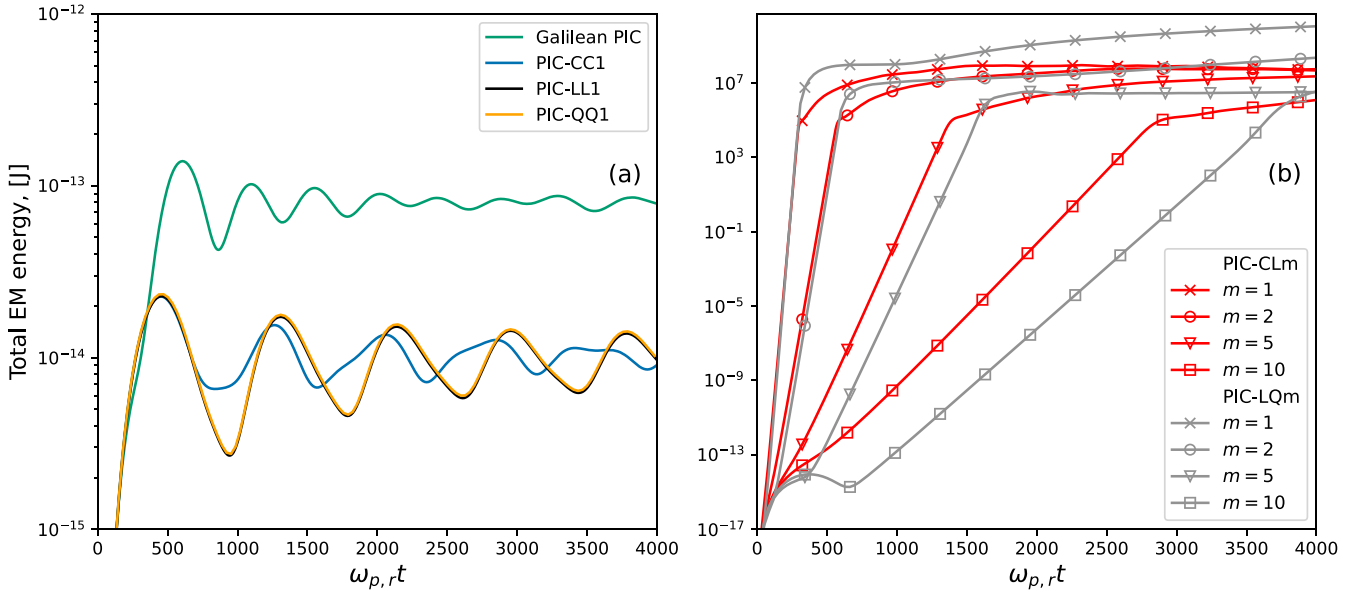

FIG. 6. WarpX simulations of a uniform plasma with a time step at the Courant limit $c\Delta t = \Delta x = \Delta z$ and a stencil at infinite order, with divergence cleaning. The total electromagnetic (EM) energy of a uniform plasma drifting at relativistic velocity $\boldsymbol{v}_0$ along the z axis is plotted versus the time of the simulation with (a) same time-dependencies for $\widehat{\boldsymbol{J}}$ and $\widehat{\rho}$ and (b) different time-dependencies for $\widehat{\boldsymbol{J}}$ and $\widehat{\rho}$, for various combinations of time-dependency and number of time-step subintervals. The energy history from a simulation using the Galilean PIC algorithm [13] is also plotted for comparison in plot (a).

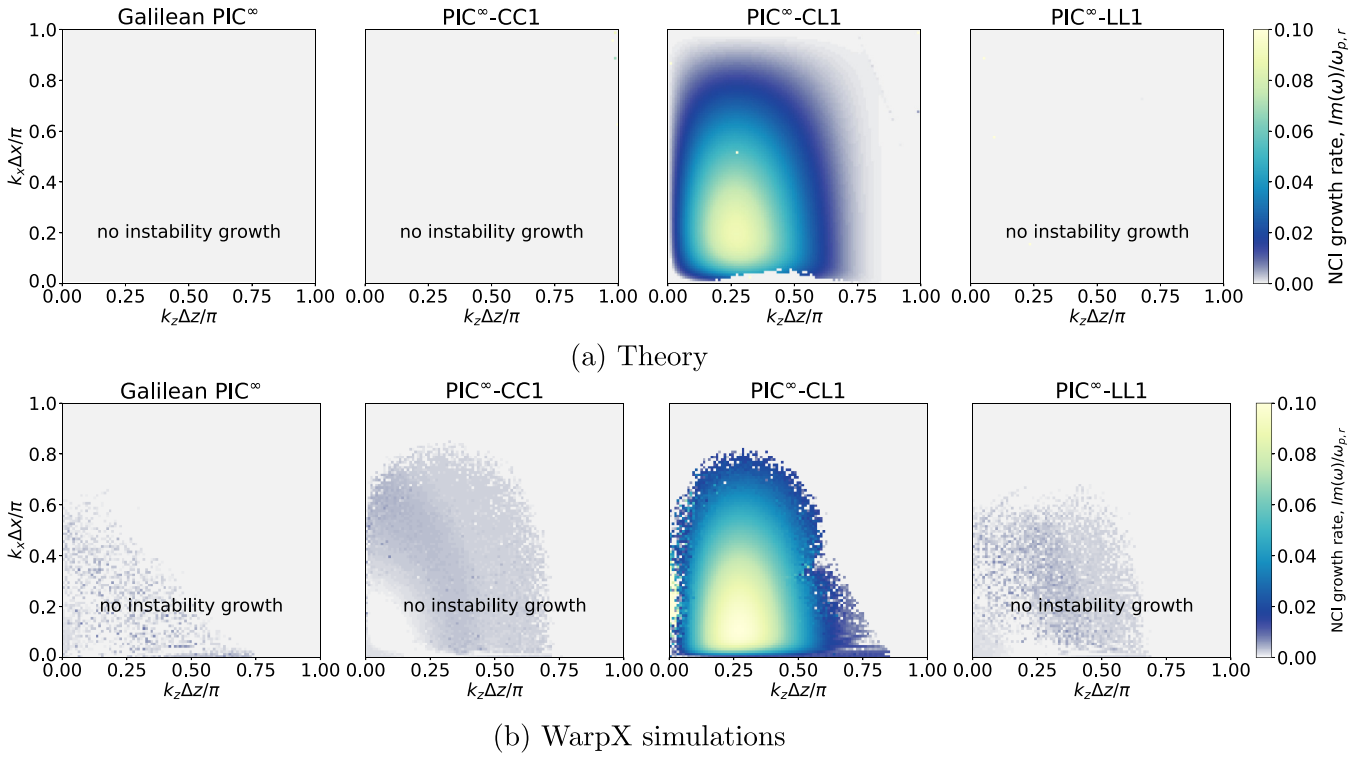

FIG. 7. NCI growth rates, with small time step $c\Delta t = \Delta x$. Normalized NCI growth rates $\mathrm{Im}(\omega)/\omega_{p,r}$ in spectral space (k_x, k_z), calculated from (a) the analytical dispersion equation and (b) WarpX simulations using four different solvers: Galilean PIC with matched velocity ($\boldsymbol{v}_{gal} = \boldsymbol{v}_0$), standard PIC-CL1, PIC-CC1, and PIC-LL1. All numerical and physical parameters are the same as in Fig. 6: divergence cleaning is used in all cases except for Galilean PIC. The simulation time step is $c\Delta t = \Delta x = \Delta z$ and the transverse and longitudinal cell sizes are $\Delta x = 6.45 \times 10^{-2}\, k_{p,r}^{-1}$, where $k_{p,r}^2 = n_0 e^2/(\epsilon_0 m_e c^2 \gamma_0)$.

Appendix E). This allows to predict the growth rates of the numerical Cherenkov instability in the case of a uniform drifting plasma. Moreover, a variety of WarpX simulation tests were run to further investigate the method's stability and accuracy. These tests include: 2D simulations of a uniform plasma drifting with a relativistic velocity $\boldsymbol{v}_0$ (with/without divergence cleaning, with/without subintervals, and with small/large time steps) and 3D simulations of laser wakefield acceleration (LWFA).

A. Stability of a uniform plasma drifting at relativistic velocity

This section presents WarpX simulations of a uniform electron-proton plasma with density $n_0 = \epsilon_0 m_e c^2 \gamma_0 / e^2$ (where ϵ_0 is the permittivity of free space, c is the speed of light in free space, and e and m_e are, respectively, the electron charge and mass), drifting along z with a relativistic velocity $\boldsymbol{v}_0 = (0, 0, v_0)$, with $v_0 = c\sqrt{1 - 1/\gamma_0^2}$ and Lorentz factor $\gamma_0 = 130$, through a two-dimensional computational domain with $x_{\min} = z_{\min} = -6.45\,\mu\mathrm{m}$ and $x_{\max} = z_{\max} = 6.45\,\mu\mathrm{m}$, periodic boundary conditions and 600×200 grid cells along x and z, respectively. The simulations were performed with 4 particles per cell, per species, 1 pass of bilinear filter in the transverse direction x and 4 passes in the longitudinal direction z (the direction along which the plasma is drifting). Four cases were considered:

(1) PIC-JRhom with $c\Delta t = \Delta x = \Delta z$ without divergence cleaning (Fig. 5);
(2) PIC-JRhom with $c\Delta t = \Delta x = \Delta z$ with divergence cleaning (Figs. 6 and 7);
(3) averaged PIC-JRhom with $c\Delta t = 6\Delta x = \Delta z$ with divergence cleaning (Fig. 8);
(4) PIC-JRhom with $c\Delta t = \Delta x = \Delta z$ and averaged PIC-JRhom with $c\Delta t = 6\Delta x = \Delta z$, with divergence cleaning and finite order stencils (Figs. 9 and 10);
and are discussed below in detail.

(1) PIC-JRhom with $c\Delta t = \Delta x = \Delta z$ without divergence cleaning.

Figure 5 shows the total electromagnetic energy as a function of $\omega_{p,r} t = \omega_p t/\sqrt{\gamma_0} = \sqrt{e^2 n_0/(m_e \epsilon_0)}$ obtained from WarpX simulations using PIC-JRhom with CCm, LLm, QQm, CLm and LQm, for $m = 1, 2, 5, 10$, without divergence cleaning. In this case, increasing the order of the polynomial dependency (from C, L, to Q), or time-step subintervals ($m > 1$), helps delaying the onset of the instability and lowering the growth rate. When using the same time dependency for $\boldsymbol{J}$ and ρ (CC, LL, and QQ), for a given number of depositions per step, it is more advantageous to increase the order of the polynomial than to increase the subintervals number m. Conversely, when using a different time dependency for $\boldsymbol{J}$ and ρ (CL, LQ), it is more advantageous to increase the number of subintervals m than to increase the order of the polynomial. Matching the time dependency

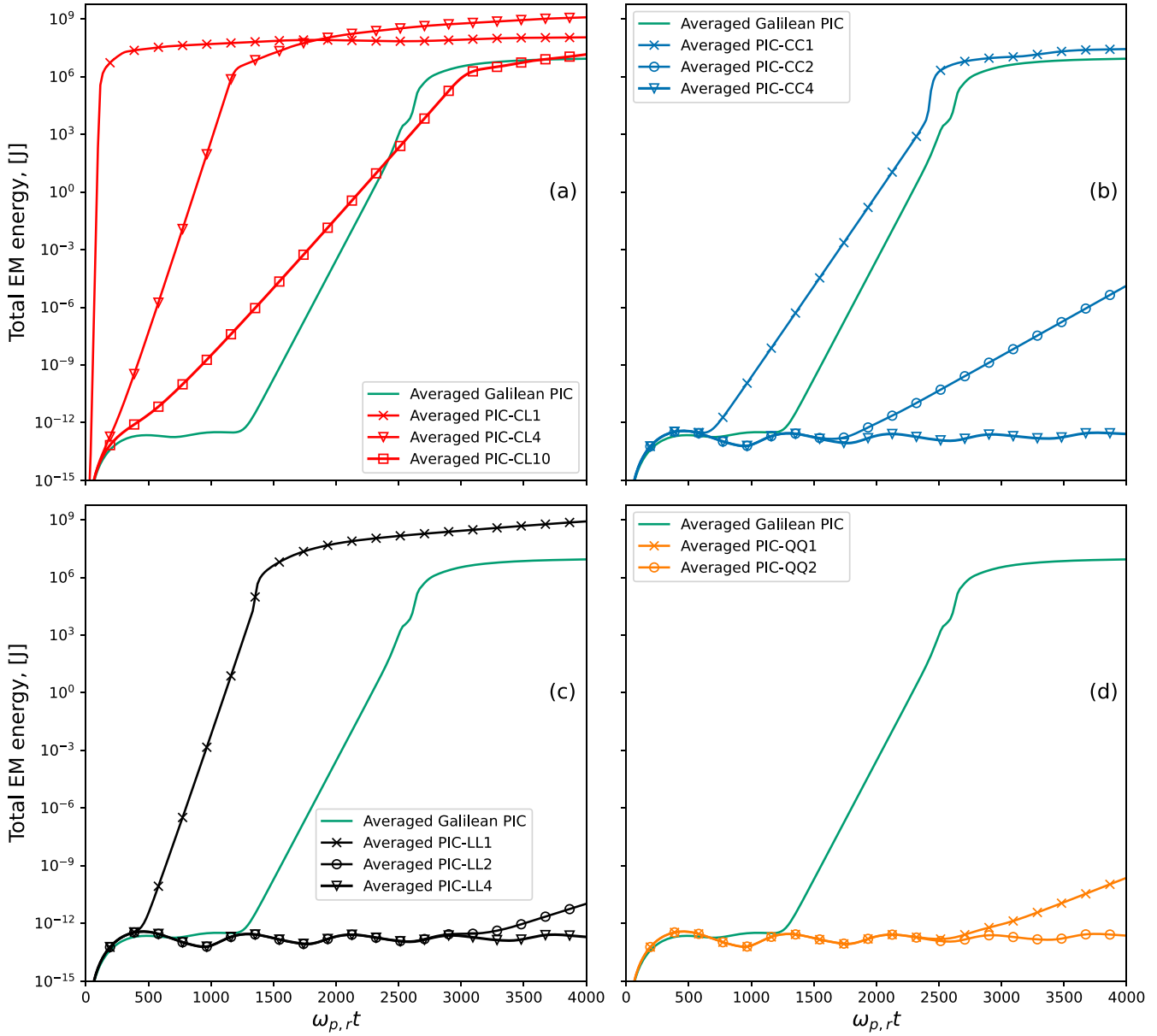

FIG. 8. WarpX simulation of a uniform plasma with $c\Delta t = \Delta z = 6\Delta x$. Total electromagnetic (EM) energy of a uniform plasma drifting at relativistic velocity $\boldsymbol{v}_0$ along the z axis. Simulations were performed with time steps of $c\Delta t = \Delta z = 6\Delta x$ and divergence cleaning, for (a) PIC-CLm, (b) PIC-CCm, (c) PIC-LLm, and (d) PIC-QQm, with $m = 1, 2, 4$. The results from a simulation using the average Galilean PIC solver is also plotted for comparison.

of $\widehat{\boldsymbol{J}}$ and $\widehat{\rho}$ (as in CC, LL, QQ) is also increasing stability, with PIC-LL5 and PIC-QQ2 being more stable than PIC-LQ10.

(2) PIC-JRhom with $c\Delta t = \Delta x = \Delta z$ with divergence cleaning.

Figure 6 shows the total electromagnetic energy as a function of $\omega_{p,r}t$ obtained from WarpX simulations using PIC-JRhom with CCm, LLm, QQm, CLm, and LQm for $m = 1, 2, 5, 10$, with divergence cleaning. The energy history from a simulation using the Galilean PIC algorithm [13] is also plotted for comparison.

In contrast to the previous case, when divergence cleaning is used, having the same time dependency for $\widehat{\boldsymbol{J}}$ and $\widehat{\rho}$ leads to an extraordinary level of stability that is comparable to the one of the Galilean PSATD method. Conversely, turning on the divergence cleaning degrades significantly the stability when using different time dependencies for $\widehat{\boldsymbol{J}}$ and $\widehat{\rho}$ (CL and LQ).

The remarkable stability reported in Fig. 6 when matching the time-dependencies is confirmed with a theoretical NCI analysis. Figure 7 shows the NCI growth rates, $\mathrm{Im}(\omega)/\omega_{p,r}$, obtained from theoretical calculations and WarpX simulations for the Galilean PIC, the standard PSATD PIC (CL1), PIC-CC1, and PIC-LL1, with an excellent agreement between theory and simulations.

A detailed derivation of the two-dimensional dispersion equation for the PIC-JRhom scheme, for time dependencies of $\widehat{\boldsymbol{J}}$ and $\widehat{\rho}$ up to quadratic, is presented in Appendix E, clarifying the origin of the remarkable stability that is observed

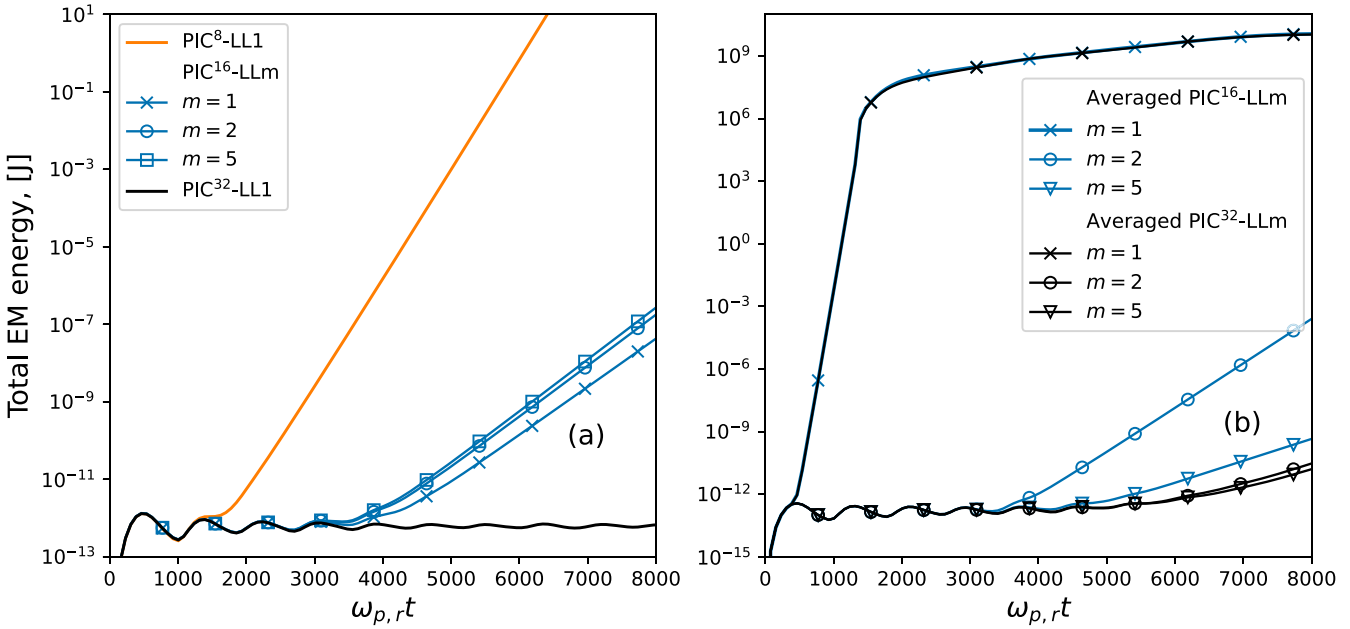

FIG. 9. WarpX simulation of a uniform plasma at finite spectral order p. Total electromagnetic (EM) energy of a uniform plasma drifting at relativistic velocity $\boldsymbol{v}_0$ along the z axis. Simulations were performed with (a) the standard PIC^p-JRhom algorithm with $c\Delta t = \Delta z = \Delta x$ and (b) the averaged PIC^p-JRhom algorithm with $c\Delta t = \Delta z = 6\Delta x$, using linear time dependency for $\widehat{\boldsymbol{J}}$ and $\widehat{\rho}$ in all cases and varying the spectral order $p = 8, 16, 32$.

with PIC-CC1, PIC-LL1 and PIC-QQ1. As explained in the Appendix, it can be shown that under some conditions that include having the same time dependency for $\widehat{\boldsymbol{J}}$ and $\widehat{\rho}$, key terms cancel out in the analysis matrix, leading to stable real solutions of the determinant.

(3) Averaged PIC-JRhom with $c\Delta t = 6\Delta x = \Delta z$ with divergence cleaning.

In this test, the transverse cell size is intentionally set to a much smaller value than the longitudinal cell size, as typical in plasma accelerator simulations in a Lorentz boosted frame of reference [16,25] with a high Lorentz factor γ_0 [12], while keeping the time step at the CFL limit of the longitudinal cell size: $c\Delta t = \Delta z = 6\Delta x$. The results from Fig. 8 show that this case is more challenging for all schemes, and even the averaged Galilean PIC scheme is not stable beyond 1000 plasma periods. Increasing the order of the polynomial and the number of subintervals m both help delaying the onset and lowering the growth rate of the instability, slowly for CLm but quite effectively for CCm, LLm, and QQm, with increasing the number of subintervals m being the most effective strategy for a given number of depositions per time step.

(4) PIC^p-JRhom with $c\Delta t = \Delta x = \Delta z$ and averaged PIC^p-JRhom with $c\Delta t = 6\Delta x = \Delta z$, with divergence cleaning and finite order stencils.

This test shows numerical (Fig. 9) and theoretical (Fig. 10) evidence that using a stencil at finite-order p with PIC^p-LLm leads to a degradation of the stability that increases as the

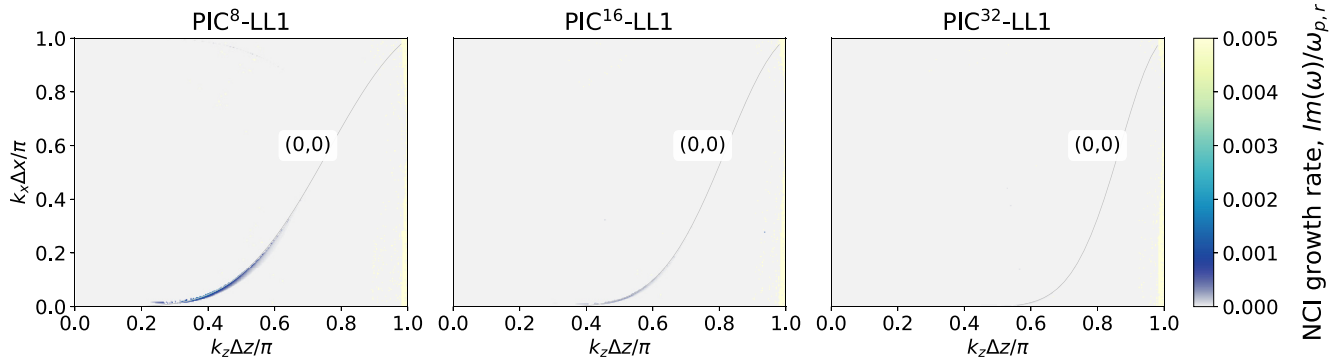

FIG. 10. NCI growth rates. Normalized NCI growth rates $\mathrm{Im}(\omega)/\omega_{p,r}$ in spectral space (k_x, k_z), calculated from the analytical dispersion equation of the PIC^p-LL1 algorithm with different stencil order $p = 8, 16, 32$. Solid grey lines correspond to $(m_t, m_z) = (0, 0)$ mode, which blue-shifts as the stencil order increases. All numerical and physical parameters are the same as the ones used for the results reported in Fig. 9.

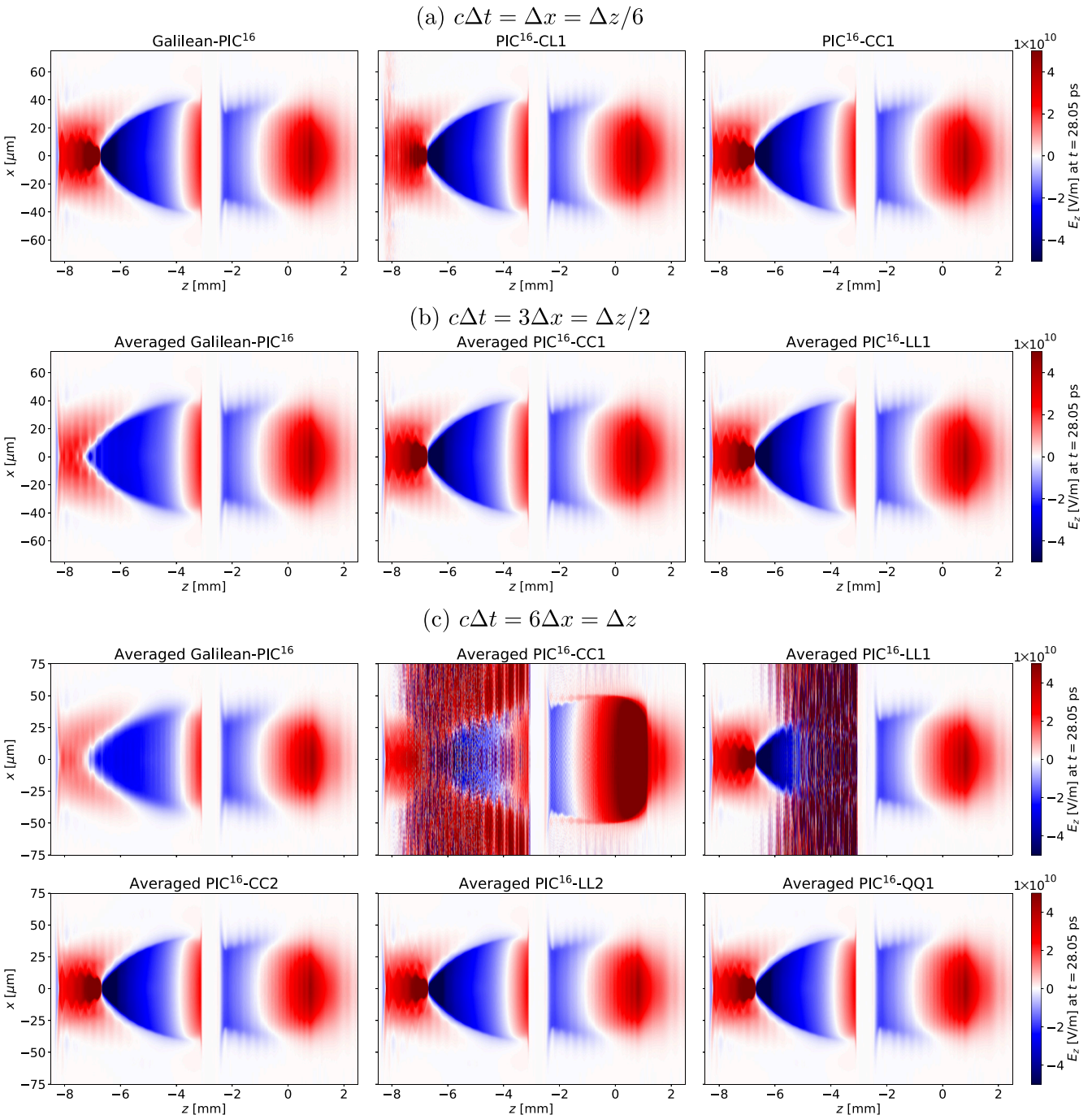

FIG. 11. WarpX simulation of LWFA with various time steps. Snapshots of the longitudinal electric field $E_z\,(x, z)$ slice at time $t = 28.05$ ps from the 3D simulation of two consecutive laser-driven plasma accelerator stages using the Galilean PIC[16] and PIC[16]-JRhom (with JRhom = CC1, CL1, LL1, CC2, LL2 or QQ1) algorithms with time step (a) $c\Delta t = \Delta x = \Delta z/6$, (b) $c\Delta t = 3\Delta x = \Delta z/2$ and (c) $c\Delta t = 6\Delta x = \Delta z$. The laser (not shown) that drives the wake propagates from left to right.

order p decreases. This is because the NCI resonant modes, caused by temporal and spatial aliasing, depends on the stencil order:

$$\left[k^p_{x,\mathrm{res}}\right] = \sqrt{\left(\left[k^p_z\right]\frac{v_0}{c} + m_z\frac{2\pi}{\Delta z}\frac{v_0}{c} - \frac{2\pi m_t}{c\Delta t}\right)^2 - \left[k^p_z\right]^2}, \quad (12)$$

for any $m_z, m_t \in \mathbb{Z}$, where m_z is the spatial alias index and m_t is the temporal alias index [26]. As the stencil order gets lower, such resonant modes relocate to lower wavenumbers where the resonance is stronger, as can be seen on Fig. 10 that shows the theoretical NCI growth rate at different spectral orders, $p = 8, 16, 32$. A nonzero growth rate is observed solely along the NCI resonant mode that is caused by aliasing between the temporal $m_t = 0$ and spatial $m_z = 0$ modes. The results from Figs. 9 and 10 indicate that the choice of stencil order will depend on the total duration of the simulations (as measured in plasma periods) for a given application.

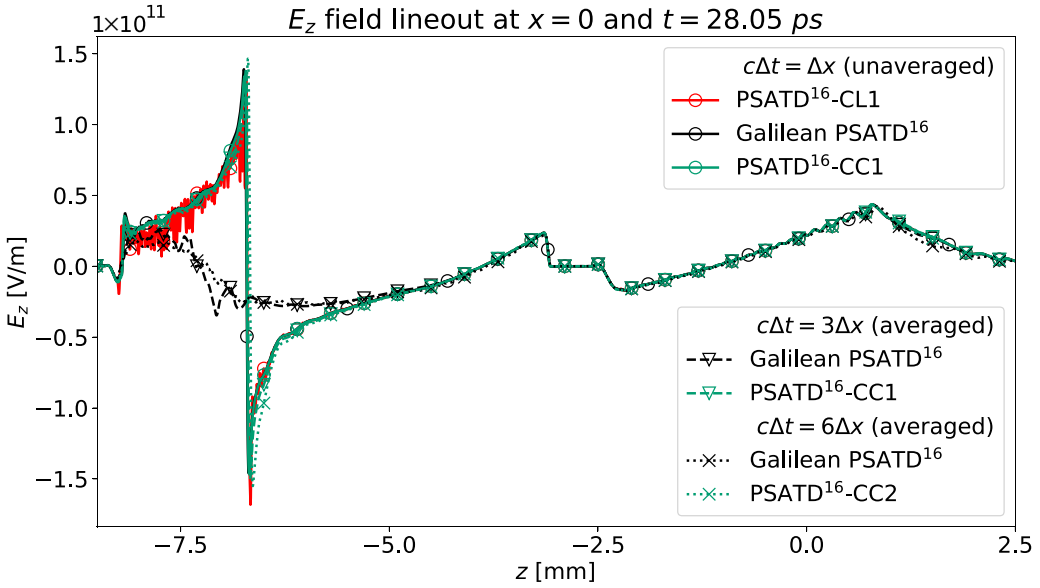

FIG. 12. WarpX simulation of LWFA. E_z-field lineouts at $x = 0$ for selected cases of the results reported in Fig. 11.

B. Laser-plasma acceleration

This section demonstrates the extension of the stability properties observed in the uniform plasma cases to realistic 3D simulations of laser wakefield acceleration (LWFA) [4]. In these runs, a Gaussian laser pulse with amplitude $a_0 = 1.7$, duration $\tau = 73.3$ fs, and waist $w_0 = 50$ μm is injected at the entrance of a parabolic plasma channel with a background density $n_0 = 10^{18}\,\mathrm{cm}^{-3}$ on axis. The simulations are run in a Lorentz boosted frame of reference [16] with $\gamma_0 = 60$ using the PIC[16]-JRhom scheme (stencil order $p = 16$ in all directions) with a hybrid alternating nodal-staggered grids (using field and current centering of order 16 in all directions) [11]. Similarly to the uniform plasma case, a bilinear filter was applied to the current and charge densities at each time step, with four passes in the z direction and one pass in the other direction. The simulations were run on the National Energy Research Scientific Computer Center (NERSC) supercomputer Perlmutter using 36 nodes (144 GPUs), with domain decomposition along both x and z, using 24 guard cells in each direction. The longitudinal resolution (in the boosted frame) was set to $\Delta z = (1+\beta_0)\gamma_0\lambda_{\mathrm{lab}}/24 = 4.08$ μm, where $\beta_0 = \sqrt{1-1/\gamma_0^2}$ and $\lambda_{\mathrm{lab}} = 0.8$ μm is the driving laser wavelength in the laboratory frame, while the transverse resolution was $\Delta x = 0.68$ μm, so that $\Delta z = 6\Delta x$. Simulations were also performed with the standard and averaged Galilean PIC[16] algorithm [13,14] for reference.

Figure 11 displays snapshots of the longitudinal electric field E_z from simulations running the Galilean PIC[16] and the PIC[16]-JRho algorithms at time $t = 28.05$ ps (which corresponds to $\omega_{p,r}t = 84.3$) with different simulation time steps: (a) $c\Delta t = \Delta x = \Delta z/6$, (b) $c\Delta t = 3\Delta x = \Delta z/2$ and (c) $c\Delta t = 6\Delta x = \Delta z$. Figure 12 shows the corresponding lineouts at $x = 0$ for a selection of runs. Table II compares the performance of the various runs in each case.

TABLE II. Performance comparison of runtimes for 3D LWFA simulations shown in Fig. 11 using different spectral PIC solvers, run on the Perlmutter supercomputer without I/O, using 36 nodes (144 GPUs), with domain decomposition in x and z and with 24 guard cells in each direction. Average time per step is from running up to time $t = 1.33$ ps which corresponds to the first 600 time steps with $c\Delta t = \Delta x$, or first 300 time steps with $c\Delta t = 3\Delta x$, or the first 100 time steps $c\Delta t = 6\Delta x$.

PSATD PIC solver	$c\Delta t/\Delta x$	Averaged in time	Stability	Average time per step [s]	Total time at $t = 1.33$ ps [s]
Galilean PIC	1	No	Stable	0.1441	86.4602
PIC-CL1	1	No	Unstable	0.1454	87.27
PIC-CC1	1	No	Stable	0.1405	84.35218
Galilean PIC	3	Yes	Stable, but inaccurate	0.1797	35.95
PIC-CC1	3	Yes	Stable	0.2349	46.995
PIC-LL1	3	Yes	Stable	0.2937	58.7453
Galilean PIC	6	Yes	Stable, but inaccurate	0.1862	18.62602
PIC-CC1	6	Yes	Unstable	0.2645	26.4574
PIC-CC2	6	Yes	Stable	0.3995	39.956
PIC-LL1	6	Yes	Unstable	0.3289	32.8913
PIC-LL2	6	Yes	Stable	0.4692	46.929
PIC-QQ1	6	Yes	Stable	0.465	45.52

When using the "small" time step $c\Delta t = \Delta x = \Delta z/6$, the PIC^{16}-CC1 algorithm is as effective as the standard Galilean PIC^{16} algorithm for mitigating the NCI instability (which is emerging at the end of the second stage in the simulations using PIC^{16}-CL1), with around 20% speedup. For larger time steps $c\Delta t = 3\Delta x = \Delta z/2$ and $c\Delta t = 6\Delta x = \Delta z$, although the averaged Galilean PIC^{16} method is stable, it does not produce accurate physics results, leading to a very diminished amplitude of the electric field in the second stage. Instead, the averaged PIC^{16}-JRhom method is stable and produces accurate results provided that the numbers of deposition and the number of time-step subintervals are high enough. For $c\Delta t = 3\Delta x = \Delta z/2$, both PIC^{16}-CC1 and PIC^{16}-LL1 are stable and accurate, with respective speedups of approximately $1.8\times$ and $1.5\times$ as compared to the Galilean reference case with small time steps. For $c\Delta t = 6\Delta x = \Delta z$, both PIC^{16}-CC1 and PIC^{16}-LL1 are unstable, while PIC^{16}-CC2, PIC^{16}-LL2, and PIC^{16}-QQ1 are stable and accurate, with respective speedups of approximately $2.2\times$, $1.8\times$, and $1.9\times$ as compared to the Galilean reference case with small time steps.

It may seem counterintuitive that the average time per step is slightly larger for both PIC-CC1 and PIC-LL1 when using $c\Delta t = 6\Delta x = \Delta z$ rather than $c\Delta t = 3\Delta x = \Delta z/2$. This is due to the fact that when using a larger time step, the number of plasma macroparticles that are exchanged between domain-decomposed regions grows with the size of the time step, leading to a fraction of extra time that grows with the time step. This is however a fraction of the total time and is thus not changing the general conclusions. These results show that the PIC^{p}-JRhom method is effective, efficient, and versatile for controlling the numerical Cherenkov instability in plasma accelerator simulations, both in cases for which other methods (e.g., averaged Galilean PIC) apply as well, and in other cases that happen to be more challenging for the other methods.

IV. CONCLUSION

A formulation of pseudospectral analytical time-domain particle-in-cell algorithm is proposed and analyzed. The formulation includes an additional term of "hyperbolic divergence cleaning" and a relaxation of the standard assumption of constant time dependency of the current density over one time step. Extensions of the algorithm to finite-order stencils, alternating nodal-staggered grids and time-averaging over a time step were also presented.

Tests and analyses revealed that assuming the same time dependency for the evolution of the charge and current densities over one time step leads to excellent stability with regard to the numerical Cherenkov instability. Detailed analysis of the dispersion relation of the algorithm (see Appendix E) provides a hint that explains the stability.

The algorithm is found to be effective, efficient, and versatile for controlling the numerical Cherenkov instability in plasma accelerator simulations, both in cases for which other methods (e.g., Galilean PIC) apply and, more importantly, in other cases that happen to be more challenging for the other methods. A possible extension of the algorithm for this particular application could be to incorporate the Galilean PIC algorithm in each subinterval, which should provide enhanced stability while preserving the versatility of the scheme.

While the application of the algorithm to the modeling of plasma acceleration has proven successful, the application to other domains must be explored with care. For example, initial testings of the application of the method to the modeling of relativistic plasma shocks [27] has led to the observation of unphysical effects that have been tentatively attributed to unphysical coupling between the unphysical longitudinal electric field waves associated with divergence cleaning [from the term F in Eq. (1c)] and the plasmas. Further studies are needed to fully understand the underlying mechanisms and propose possible remedies.

ACKNOWLEDGMENTS

This research used the open-source particle-in-cell code WarpX [28]. We acknowledge all WarpX contributors. This research was supported by the Exascale Computing Project (No. 17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration. This material is based upon work supported by the CAMPA collaboration, a project of the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research and Office of High Energy Physics, Scientific Discovery through Advanced Computing (SciDAC) program. This research was performed in part under the auspices of the U.S. Department of Energy by Lawrence Berkeley National Laboratory under Contract No. DE-AC02-05CH11231. This research used resources of the National Energy Research Scientific Computing Center (NERSC), a Department of Energy Office of Science User Facility using NERSC Award No. HEP-ERCAP0027030. This research used resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract No. DE-AC05-00OR22725. The data that support the findings of this study are available from the corresponding author upon reasonable request. O.S. derived and implemented the algorithms in their final forms, performed the numerical analyses and numerical tests. E.Z. contributed to the derivation, implementation and testing of the algorithms. R.L. wrote the initial implementation of the algorithm in WarpX and discussed the results. M.T. performed numerical tests of an early prototype implemented in the code Warp. J.L.V. proposed the concept and implemented an early prototype in the code Warp.

APPENDIX A: CONNECTION BETWEEN THE MODIFIED SYSTEM OF MAXWELL'S EQUATIONS AND A POTENTIAL FORMULATION

It is instructive to derive the modified system of Maxwell's equations (1) in its potential form, starting with

$$\frac{\partial \boldsymbol{E}}{\partial t} = c^2\nabla\times\boldsymbol{B} - \frac{\boldsymbol{J}}{\varepsilon_0} + c^2\nabla F, \tag{A1a}$$

$$\frac{\partial \boldsymbol{B}}{\partial t} = -\nabla\times\boldsymbol{E}, \tag{A1b}$$

$$\frac{\partial F}{\partial t} = \nabla\cdot\boldsymbol{E} - \frac{\rho}{\varepsilon_0}, \tag{A1c}$$

$$\nabla\cdot\boldsymbol{B} = 0. \tag{A1d}$$

Equation (A1d) implies that $\boldsymbol{B}$ can be derived from a potential $\boldsymbol{B} = \nabla \times \boldsymbol{A}$ which, when inserted into Eq. (A1b), gives $\nabla \times (\boldsymbol{E} + \frac{\partial \boldsymbol{A}}{\partial t}) = 0$. This means that $\boldsymbol{E} + \frac{\partial \boldsymbol{A}}{\partial t}$ can be written as the gradient of a potential Φ, giving

$$\boldsymbol{E} = -\nabla\Phi - \frac{\partial \boldsymbol{A}}{\partial t}. \tag{A2}$$

Plugging Eq. (A2) into Eqs. (A1a) and (A1c) leads to

$$\nabla^2\Phi + \frac{\partial}{\partial t}(\nabla \cdot \boldsymbol{A}) = -\frac{\rho}{\varepsilon_0} - \frac{\partial F}{\partial t}, \tag{A3a}$$

$$\nabla^2\boldsymbol{A} - \frac{\partial^2 \boldsymbol{A}}{c^2\partial t^2} - \nabla\left(\nabla \cdot \boldsymbol{A} + \frac{1}{c^2}\frac{\partial \Phi}{\partial t}\right) = -\mu_0 \boldsymbol{J} + \nabla F, \tag{A3b}$$

which, choosing Φ and $\boldsymbol{A}$ that verify the Lorentz gauge $\nabla\boldsymbol{A} + \frac{1}{c^2}\frac{\partial \Phi}{\partial t} = 0$, gives

$$\nabla^2\Phi - \frac{\partial^2\Phi}{c^2\partial t^2} = -\frac{\rho}{\varepsilon_0} - \frac{\partial F}{\partial t}, \tag{A4a}$$

$$\nabla^2\boldsymbol{A} - \frac{\partial^2 \boldsymbol{A}}{c^2\partial t^2} = -\mu_0 \boldsymbol{J} + \nabla F, \tag{A4b}$$

$$\nabla^2 F - \frac{\partial^2 F}{c^2\partial t^2} = \mu_0\left(\nabla \cdot \boldsymbol{J} + \frac{\partial \rho}{\partial t}\right). \tag{A4c}$$

A gauge transformation

$$\boldsymbol{A}' = \boldsymbol{A} - \nabla\Lambda, \tag{A5a}$$

$$\phi' = \phi + \frac{1}{c^2}\frac{\partial \Lambda}{\partial t}, \tag{A5b}$$

with

$$F = \left(-\nabla^2 + \frac{1}{c^2}\frac{\partial^2}{\partial t^2}\right)\Lambda = -\left(\nabla\boldsymbol{A}' + \frac{\partial \Phi'}{c^2\partial t}\right) \tag{A6a}$$

then leads to

$$\nabla^2\Phi' - \frac{\partial^2\Phi'}{c^2\partial t^2} = -\frac{\rho}{\varepsilon_0}, \tag{A7a}$$

$$\nabla^2\boldsymbol{A}' - \frac{\partial^2 \boldsymbol{A}'}{c^2\partial t^2} = -\mu_0 \boldsymbol{J}. \tag{A7b}$$

This is consistent with the derivation given in Ref. [17], where Eqs. (A1a)–(A1c) were derived from Maxwell's equations in the Lorentz gauge form [i.e., the form of Eqs. (A7a) and (A7b)] with the assumption that $\boldsymbol{J} = \boldsymbol{J}_0 + \delta\boldsymbol{J}$ where $\boldsymbol{J}_0$ is the portion of $\boldsymbol{J}$ that verifies the continuity equation $\frac{\partial \rho}{\partial t} + \nabla\boldsymbol{J}_0 = 0$, and defining $F = -\nabla\delta\boldsymbol{A}$ such that $\boldsymbol{A}' = \boldsymbol{A} + \delta\boldsymbol{A}$ with $\nabla\boldsymbol{A} + \frac{\partial \Phi'}{c^2\partial t} = 0$.

In addition to showing that the term F can arise from considerations other than a "divergence cleaning" term, this derivation also highlights how F relates more directly to the continuity equation via Eq. (A4c) and gauges via Eq. (A6).

APPENDIX B: ABSENCE OF SPURIOUS SELF-FORCE

The main reason behind the recommendation to use the same shape factor for charge and current deposition and field gather in PIC codes is to avoid a gravitational-like instability that occurs when using a shape factor for field gather that is at lower order than for charge and current deposition [29]. Also, as stated in Sec. 8.5 of Ref. [3], "If the difference equations relating the densities to the electric fields are symmetric in space, use of the same weight function eliminates the self-force and ensures conservation of momentum." The analyses of the self-force in these earlier work by Langdon, Birdsall, and others [30] indeed rely on the spatial symmetries of the difference equations used by the algorithm, which are preserved in the proposed scheme by the use of PSATD and the same splines for charge and current deposition and field gather. The subcycling in the deposition of the charge and current densities does not change the spatial symmetries of the difference equations, and hence do not lead to self-forces. We have indeed verified on simulations of a single particle at rest that the electric field gathered onto the particle was zero to machine precision (relative to the maximum electric field near the particle), independently of the position of the particle within a cell:

(1) Electric field (relative to maximum electric field) experienced by a single particle located at [0.25*dx, 0.75*dy, 0.5*dz] within a cell at the center of the simulation box (where dx, dy, and dz is the cell size along x, y, and z) after 20 times steps after initialization:

(a) CL1: $E_x \approx 1.85e-17$, $E_y \approx -1.38e-16$, $E_z \approx 1.48e-17$

(b) CC1: $E_x \approx -8.54e-18$, $E_y \approx -4.59e-16$, $E_z \approx 2.56e-17$

(c) LL2: $E_x \approx 7.69.3e-17$, $E_y \approx -2.15e-16$, $E_z \approx 1.01e-16$

The CL1 algorithm is the standard PIC algorithm, for which it is well known already that there is no self-force, and verified here. It was also verified for a number of different configurations of the algorithm (CC1 and LL2 are reported here) that the property is preserved, as evidenced by the relative values of self-field experienced by the particle reported above being zero to machine precision.

APPENDIX C: DERIVATION OF THE PIC-JRhoM EQUATIONS

We first rewrite Eqs. (2) in an equivalent second-order differential form,

$$\frac{\partial^2\widehat{\boldsymbol{E}}}{\partial t^2} + c^2k^2\widehat{\boldsymbol{E}} = -\frac{1}{\varepsilon_0}\left(\frac{\partial \widehat{\boldsymbol{J}}}{\partial t} + ic^2\boldsymbol{k}\widehat{\rho}\right), \tag{C1a}$$

$$\frac{\partial^2\widehat{\boldsymbol{B}}}{\partial t^2} + c^2k^2\widehat{\boldsymbol{B}} = \frac{1}{\varepsilon_0}i\boldsymbol{k} \times \widehat{\boldsymbol{J}}, \tag{C1b}$$

$$\frac{\partial^2\widehat{F}}{\partial t^2} + c^2k^2\widehat{F} = -\frac{1}{\varepsilon_0}\left(\frac{\partial \widehat{\rho}}{\partial t} + \boldsymbol{k}\widehat{\boldsymbol{J}}\right), \tag{C1c}$$

and then we sequentially integrate them analytically over each subinterval $[n\Delta t + \ell\delta t, n\Delta t + (\ell+1)\delta t]$, $\ell \in [0, m-1]$ with $\delta t = \Delta t/m$, assuming that the current and charge densities are piecewise functions of time, given by Eqs. (3a) and (3b). Each of those equations can be expressed in the following generalized form with a right part as time

TABLE III. Integration coefficients over ℓth time subinterval $[n\Delta t + \ell\delta t, n\Delta t + (\ell+1)\delta t]$.

$\widehat{f}$	$\partial_t \widehat{f}(t_n + \ell/m)$	C_3	C_4	C_5
$\widehat{\boldsymbol{E}}$	$ic^2\boldsymbol{k}\times\boldsymbol{B}^{n+\ell/m} - \frac{\boldsymbol{J}^{n+\ell/m}}{\varepsilon_0}$ $+ic^2\widehat{F}^{n+\ell/m}\boldsymbol{k}$	$-i\frac{2a_\rho^\tau c^2\boldsymbol{k}}{\varepsilon_0\delta t^2}$	$-\frac{4a_{\boldsymbol{J}}^\tau + ib_\rho^\tau c^2\delta t\boldsymbol{k}}{\varepsilon_0\delta t^2}$	$\frac{4ia_\rho^\tau c^2\boldsymbol{k} - b_{\boldsymbol{J}}^\tau\omega^2\delta t - ic^2 c_\rho\omega^2\delta t^2\boldsymbol{k}}{\varepsilon_0\delta t^2}$
$\widehat{\boldsymbol{B}}$	$-i\boldsymbol{k}\times\boldsymbol{E}^{n+\ell/m}$	$i\frac{2a_{\boldsymbol{J}}^\tau c^2\boldsymbol{k}\times}{\varepsilon_0\delta t^2}$	$i\frac{b_{\boldsymbol{J}}^\tau\boldsymbol{k}\times}{\varepsilon_0\delta t}$	$\frac{-4ia_{\boldsymbol{J}}^\tau\boldsymbol{k}\times + ic_{\boldsymbol{J}}\omega^2\delta t^2\boldsymbol{k}\times}{\varepsilon_0\delta t^2\omega^2}$
$\widehat{F}$	$i\boldsymbol{k}\cdot\widehat{\boldsymbol{E}}^{n+\ell/m} - \frac{\widehat{\rho}^{n+\ell/m}}{\varepsilon_0}$	$-i\frac{2a_{\boldsymbol{J}}^\tau\boldsymbol{k}}{\varepsilon_0\delta t^2}$	$-i\frac{4a_\rho^\tau\boldsymbol{k} + ib_{\boldsymbol{J}}\delta t\boldsymbol{k}}{\varepsilon_0\delta t^2}$	$\frac{4ia_{\boldsymbol{J}}^\tau\boldsymbol{k} - b_\rho\delta t\omega^2 - ic_{\boldsymbol{J}}\omega^2\delta t^2\boldsymbol{k}}{\varepsilon_0\delta t^2\omega^2}$

polynomial up to order two:

$$\left(\frac{\partial^2}{\partial t^2} + c^2k^2\right)\widehat{f} = \sum_{j=0}^{2} a_{0j}t^j, \tag{C2}$$

where $\{a_{0j}\}_{j=0}^2$ are known coefficients for any given $\widehat{f} = \widehat{\boldsymbol{E}}, \widehat{\boldsymbol{B}}, \widehat{F}$. The general solution of such a second-order PDE equation with constant coefficients is

$$\begin{aligned}\widehat{f}(t) = {} & C_1\cos(\omega(t - t_{n+\ell/m})) + C_2\sin(\omega(t - t_{n+\ell/m})) \\ & + \frac{1}{\omega^2}(C_3(t - t_{n+(\ell+1/2)m})^2 \\ & + C_4(t - t_{n+(\ell+1/2)m}) + C_5),\end{aligned} \tag{C3}$$

where $\{C_k\}_{k=1}^5$ are integration coefficients to be determined. The coefficients C_k with indexes $k = 3, 4, 5$ for any given $\widehat{f} = \widehat{\boldsymbol{E}}, \widehat{\boldsymbol{B}}, \widehat{F}$ can be determined by solving a system of linear equations, obtained from substitution of Eq. (C3) into the corresponding Eq. (C1) and calculated at time steps $t_{n+\ell/m}$, $t_{n+(\ell+1/2)/m}$ and $t_{n+(\ell+1)/m}$. While the remaining coefficients C_1 and C_2 can be determined from the initial conditions $\widehat{f}(t)|_{t_{n+\ell/m}}$ and $\partial_t\widehat{f}(t)|_{t_{n+\ell/m}}$, respectively,

$$\begin{aligned}C_1 &= f(t_n + \ell/m) - (C_3(\delta t/2)^2 + C_4(\delta t/2) + C_5)/\omega^2, \\ C_2 &= \partial_t f(t_n + \ell/m) - (2C_3(\delta t/2) + C_4)/\omega^2.\end{aligned} \tag{C4}$$

The expression of the field components $\widehat{f}(t_{n+(\ell+1)/m})$ at the next time subinterval are then given by

$$\begin{aligned}\widehat{f}(t_{n+(\ell+1)/m}) = {} & C_1\cos(\omega\delta t) + C_2\sin(\omega\delta t) \\ & + \frac{1}{\omega^2}(C_3(\delta t/2)^2 + C_4(\delta t/2) + C_5).\end{aligned} \tag{C5}$$

where C2, C3 and C4 are given in Table III.

APPENDIX D: DERIVATION OF THE AVERAGED PIC-JRhoM EQUATIONS

The notation $\langle\widehat{f}(t)\rangle^{n+1}$ is introduced to refer to the average of any given function $\widehat{f}(t)$ over the time interval $[n\Delta t, (n+1)\Delta t]$ as

$$\langle\widehat{f}\rangle^{n+1} = \frac{1}{2\Delta t}\int_{t_n}^{t_n+2\Delta t}\widehat{f}(t')dt', \quad \text{where } \widehat{f} = \widehat{\boldsymbol{E}}, \widehat{\boldsymbol{B}}. \tag{D1}$$

For any given number of subintervals m, the integral in Eq. (D1) can be split into a sum over $2m$ integrals over $[t_n + \ell\delta t, t_n + (\ell+1)\delta t]$, $\ell = 0, .., 2m-1$ as

$$\langle\widehat{f}\rangle^{n+1} = \frac{1}{2\Delta t}\sum_{\ell=0}^{2m-1}\int_{t_n+\ell\delta t}^{t_n+(\ell+1)\Delta t}\widehat{f}(t')dt', \text{ where } \widehat{f} = \widehat{\boldsymbol{E}}, \widehat{\boldsymbol{B}}. \tag{D2}$$

The averaged $\langle\widehat{\boldsymbol{E}}\rangle$ and $\langle\widehat{\boldsymbol{B}}\rangle$ fields are obtained through sequential integration of Eq. (C3) over each subinterval $[t_n + \ell\delta t, t_n + (\ell+1)\delta t]$, $\ell = 0, .., 2m-1$ and then substituted into Eq. (D2),

$$\begin{aligned}\int_{t_n+\ell\delta t}^{t_n+(\ell+1)\delta t}\widehat{\boldsymbol{E}}(t')dt' = {} & \frac{S}{ck}\widehat{\boldsymbol{E}}^{n+\ell/m} + \frac{ic^2Y_4}{c^2k^2}\boldsymbol{k}\times\widehat{\boldsymbol{B}}^{n+\ell/m} \\ & + \frac{i\boldsymbol{k}Y_4}{2ck\delta t}\widehat{F}^{n+\ell/m} \\ & + \frac{1}{\varepsilon_0c^2k^2}(Y_1\boldsymbol{a}_j - Y_5\boldsymbol{b}_J - Y_4\boldsymbol{c}_J) \\ & - ic^2\boldsymbol{k}(Y_6a_\rho + Y_7b_\rho + Y_8c_\rho),\end{aligned} \tag{D3a}$$

$$\begin{aligned}\int_{t_n+\ell\delta t}^{t_n+(\ell+1)\delta t}\widehat{\boldsymbol{B}}(t')dt' = {} & \frac{S}{ck}\widehat{\boldsymbol{B}}^{n+\ell/m} - \frac{iY_4}{c^2k^2}\boldsymbol{k}\times\widehat{\boldsymbol{E}}^{n+\ell/m} \\ & + i\boldsymbol{k}\times(Y_6\boldsymbol{a}_J + Y_7\boldsymbol{b}_J + Y_8\boldsymbol{c}_J),\end{aligned} \tag{D3b}$$

with

$$\begin{aligned}Y_6 = {} & \frac{1}{6\varepsilon_0c^5k^5\delta t^2}((ck\delta t)^2 - 3\delta(ck\delta t)^2S \\ & - 12ck\delta t(1+C) + 24S),\end{aligned} \tag{D4a}$$

$$Y_7 = \frac{1}{2\varepsilon_0c^4k^4\delta t}(ck\delta tS + 2(C-1)), \tag{D4b}$$

$$Y_8 = \frac{\delta t}{\varepsilon_0c^2k^2}\left(1 - \frac{S}{ck\delta t}\right). \tag{D4c}$$

APPENDIX E: DISPERSION RELATION FOR THE PIC-JRhom ALGORITHM

The 2D dispersion relation for Eqs. (4) is derived to analyze the algorithm's stability with respect to the numerical Cherenkov instability (NCI), for a uniform plasma flowing through a periodic grid along the z axis with a velocity $\boldsymbol{v}_0 = (0, 0, v_0)$, where $v_0 = c(1 - 1/\gamma_0^2)^{1/2}$. Following the analysis from Refs. [12,13], we consider the discretized perturbed

Vlasov equation, expressed in Fourier space:

$$\delta\hat{f}^{n+1/2}(\boldsymbol{k}_m,\boldsymbol{p})e^{i\boldsymbol{k}_m\cdot\boldsymbol{v}\Delta t/2}-\hat{f}^{n-1/2}(\boldsymbol{k}_m,\boldsymbol{p})e^{-i\boldsymbol{k}_m\cdot\boldsymbol{v}\Delta t/2}+q\Delta t\hat{S}(\boldsymbol{k}_m)[\widehat{\boldsymbol{E}}^n(\boldsymbol{k})+\boldsymbol{v}\times\widehat{\boldsymbol{B}}^n(\boldsymbol{k})]\cdot\frac{\partial f_0}{\partial\boldsymbol{p}}=0, \tag{E1}$$

where $f_0=n_0\delta(\boldsymbol{p}-m\gamma_0\boldsymbol{v}_0)$ is the distribution function of the uniform plasma in a state of equilibrium, and δf is a perturbation to f_0. The discretized formulas for the deposited current and charge in Fourier space at any time $\ell\Delta t$, $\ell\in[n,n+1]$ centered around $\delta\hat{f}^{n+1/2}$, are given by

$$\widehat{\boldsymbol{J}}^{\ell}(\boldsymbol{k})=\sum_m S(\boldsymbol{k}_m)\int d\boldsymbol{p}q\boldsymbol{v}\delta\hat{f}^{n+1/2}(\boldsymbol{k}_m,\boldsymbol{p})e^{-i\boldsymbol{k}_m\cdot\boldsymbol{v}(\ell-(n+1/2))\Delta t}, \tag{E2}$$

$$\widehat{\rho}^{\,\ell}(\boldsymbol{k})=\sum_m S(\boldsymbol{k}_m)\int d\boldsymbol{p}q\delta\hat{f}^{n+1/2}(\boldsymbol{k}_m,\boldsymbol{p})e^{-i\boldsymbol{k}_m\cdot\boldsymbol{v}(\ell-(n+1/2))\Delta t}. \tag{E3}$$

Then, assuming the same $e^{-i\omega t}$ time evolution for $\widehat{\boldsymbol{E}},\widehat{\boldsymbol{B}},\widehat{F},\widehat{\boldsymbol{J}},\widehat{\rho}$, and $\widehat{\delta f}$ with the following anzatz:

$$\widehat{\boldsymbol{E}}^n(\boldsymbol{k})=\widehat{\boldsymbol{E}}(\boldsymbol{k})e^{-i\omega n\Delta t}, \tag{E4a}$$

$$\delta\hat{f}^{n+1/2}(\boldsymbol{k}_m,\boldsymbol{p})=\delta\hat{f}(\boldsymbol{k}_m,\boldsymbol{p})e^{-i\omega(n+1/2)\Delta t}, \tag{E4b}$$

$$\widehat{\boldsymbol{J}}^n(\boldsymbol{k})=\widehat{\boldsymbol{J}}(\boldsymbol{k})e^{-i\omega n\Delta t}, \tag{E4c}$$

$$\widehat{\rho}^n(\boldsymbol{k})=\widehat{\rho}(\boldsymbol{k})e^{-i\omega n\Delta t}. \tag{E4d}$$

Equation (E1) yields

$$\delta\hat{f}(\boldsymbol{k}_m,\boldsymbol{p})=-i\frac{q\Delta t}{2}\widehat{S}(\boldsymbol{k}_m)\frac{\widehat{\boldsymbol{E}}(\boldsymbol{k})+\boldsymbol{v}\times\widehat{\boldsymbol{B}}(\boldsymbol{k})}{\sin((\omega-\boldsymbol{k}_m\cdot\boldsymbol{v})\Delta t/2)}. \tag{E5}$$

Substituting the Vlasov equation (E1) into Eqs. (E2) and (E3) gives the following expressions for the deposited current $\widehat{\boldsymbol{J}}(\boldsymbol{k})$ and the charge $\widehat{\rho}(\boldsymbol{k})$:

$$\widehat{\boldsymbol{J}}=i\frac{ck\varepsilon_0}{\widehat{T}}\Big(\xi_0+(\boldsymbol{\xi}\cdot\widehat{\boldsymbol{Q}})\frac{\boldsymbol{v}}{c}\Big), \tag{E6}$$

$$\widehat{\rho}=\frac{ik\varepsilon_0}{\widehat{T}}(\boldsymbol{\xi}\cdot\widehat{\boldsymbol{Q}}), \tag{E7}$$

$$\widehat{\boldsymbol{Q}}(\boldsymbol{k})=\widehat{\boldsymbol{E}}(\boldsymbol{k})+\boldsymbol{v}\times\widehat{\boldsymbol{B}}(\boldsymbol{k})-(\boldsymbol{v}\cdot\widehat{\boldsymbol{E}}(\boldsymbol{k}))\boldsymbol{v}/c^2, \tag{E8}$$

$$\xi_0=\frac{\widehat{T}\omega_p^2}{\gamma_0 ck}\sum_{m=-\infty}^{+\infty}S^2(\mathbf{k_m})\cdot\frac{1}{\frac{2}{\Delta t}s'_\omega}, \tag{E9}$$

$$\boldsymbol{\xi}=\frac{\widehat{T}\omega_p^2}{\gamma_0 k}\sum_{m=-\infty}^{+\infty}S^2(\mathbf{k_m})\cdot\frac{\mathbf{k_m}c'_\omega}{\big(\frac{2}{\Delta t}s'_\omega\big)^2}, \tag{E10}$$

where $\widehat{T}=\prod_i[1-\sin(k_i\Delta i/2)]$ is one pass of a binomial smoothing operator, and $\omega_p=(n_0q^2m_e^{-1}\varepsilon_0^{-1})^{1/2}$ is the plasma frequency, and $\widehat{S}(\boldsymbol{k}_m)$ is the particle shape factor. Still following Refs. [12,13], Eqs. (4) are then rewritten in the time-symmetrical form

$$(\widehat{\boldsymbol{E}}^{n+(\ell+1)/m}-\widehat{\boldsymbol{E}}^{n+\ell/m})=i\frac{S}{(1+C)ck}c^2\boldsymbol{k}\times(\widehat{\boldsymbol{B}}^{n+(\ell+1)/m)}+\widehat{\boldsymbol{B}}^{n+\ell/m})+i\frac{S}{(1+C)ck}c^2\boldsymbol{k}(\widehat{F}^{n+(\ell+1)/m}+\widehat{F}^{n+\ell/m})+\frac{1}{\varepsilon_0\omega}(Y_9\boldsymbol{a}_j-2S(1+C)^{-1}\boldsymbol{c}_j)-\frac{ic^2}{\varepsilon_0c^2k^2}Y_{10}\boldsymbol{k}b_\rho, \tag{E11a}$$

$$(\widehat{\boldsymbol{B}}^{n+(\ell+1)/m}-\widehat{\boldsymbol{B}}^{n+\ell/m})=-\frac{S}{(1+C)ck}i\boldsymbol{k}\times(\widehat{\boldsymbol{E}}^{n+(\ell+1)/m}+\widehat{\boldsymbol{E}}^{n+\ell/m})+\frac{i\boldsymbol{k}\times\boldsymbol{b}_j}{\varepsilon_0c^2k^2}Y_{10}, \tag{E11b}$$

$$(\widehat{F}^{n+(\ell+1)/m}-\widehat{F}^{n+\ell/m})=\frac{S}{(1+C)ck}i\boldsymbol{k}(\widehat{\boldsymbol{E}}^{n+(\ell+1)/m}+\widehat{\boldsymbol{E}}^{n+\ell/m})-\frac{1}{\varepsilon_0c^2k^2}i\boldsymbol{k}\boldsymbol{b}_jY_{10}+\frac{1}{\varepsilon_0ck}(Y_9a_\rho-2S(1+C)^{-1}c_\rho). \tag{E11c}$$

Substitution of Eqs. (E4) in Eqs. (E11a)–(E11c) gives

$$s_\omega\widehat{\boldsymbol{E}}=-t_{ck}c_\omega\boldsymbol{k}\times c\widehat{\boldsymbol{B}}-c_\omega t_{ck}\hat{\boldsymbol{k}}c\widehat{F}+i\big(Y_9\tilde{a}^\tau_\omega/2-t_{ck}\tilde{c}^\tau_\omega\big)\widehat{\boldsymbol{J}}+\big(Y_{10}\tilde{b}^\tau_\omega/2\big)\boldsymbol{k}\widehat{\rho}, \tag{E12a}$$

$$s_\omega\widehat{\boldsymbol{B}}=t_{ck}c_\omega\boldsymbol{k}\times\widehat{\boldsymbol{E}}-\big(Y_{10}\tilde{b}^\tau_\omega/2\big)\boldsymbol{k}\times\widehat{\boldsymbol{J}}, \tag{E12b}$$

$$s_\omega c\widehat{F}=-t_{ck}c_\omega\boldsymbol{k}\cdot\widehat{\boldsymbol{E}}+\boldsymbol{k}\cdot\widehat{\boldsymbol{J}}\big(Y_{10}\tilde{b}^\tau_\omega/2\big)+i\big(Y_9\tilde{a}^\tau_\omega/2-t_{ck}\tilde{c}_\omega\big)\widehat{\rho}. \tag{E12c}$$

Projecting Eqs. (E6) and (E12a) along x and z and Eq. (E12b) along y gives the following 2D dispersion equation in matrix form:

$$\boldsymbol{M}\boldsymbol{U}^T = 0, \tag{E13a}$$

$$\boldsymbol{M} = \begin{bmatrix} -s_\omega & 0 & c_\omega \widehat{k}_z t_{ck} & -c_\omega \widehat{k}_x t_{ck} & iT\chi_{\tau_J} & 0 & -iT\widehat{k}_x \psi_{\tau_\rho} \\ 0 & -s_\omega & -c_\omega \widehat{k}_x t_{ck} & -c_\omega \widehat{k}_z t_{ck} & 0 & iT\chi_{\tau_J} & -iT\widehat{k}_z \psi_{\tau_\rho} \\ c_\omega \widehat{k}_z t_{ck} & -c_\omega \widehat{k}_x t_{ck} & -s_\omega & 0 & iT\widehat{k}_z \psi_{\tau_J} & -iT\widehat{k}_x \psi_{\tau_J} & 0 \\ -c_\omega \widehat{k}_x t_{ck} & -c_\omega \widehat{k}_z t_{ck} & 0 & -s_\omega & -iT\widehat{k}_x \psi_{\tau_J} & -iT\widehat{k}_z \psi_{\tau_J} & iT\chi_{\tau_\rho} \\ \frac{i}{T}\xi_0 & 0 & -\frac{i}{T}\xi_0\beta_0 & 0 & -1 & 0 & 0 \\ \frac{i}{T}\xi_x\beta_0 & \frac{i}{T}\left(1-\beta_0^2\right)(\xi_0+\xi_z\beta_0) & -\frac{i}{T}\xi_x\beta_0^2 & 0 & 0 & -1 & 0 \\ \frac{i}{T}\xi_x & \frac{i}{T}\xi_z\left(1-\beta_0^2\right) & -\frac{i}{T}\xi_x\beta_0 & 0 & 0 & 0 & -1 \end{bmatrix}, \tag{E13b}$$

where $\boldsymbol{U} = (\widehat{E}_x, \widehat{E}_z, c\widehat{B}_y, c\widehat{F}, \widehat{J}_x/(ck\,\varepsilon_0), \widehat{J}_z/(ck\,\varepsilon_0), \widehat{\rho}/(k\,\varepsilon_0))$ and $\widehat{\boldsymbol{k}} = \boldsymbol{k}/k$ is the normalized wave vector. The matrix coefficients in $\boldsymbol{M}$ that depend on the time dependency of the current and charge densities $\widehat{\boldsymbol{J}}$ and $\widehat{\rho}$ are summarized in Table IV. For example, the upper index $\tau_{J/\rho}$ in the coefficients $\psi_{\tau_{J/\rho}}$ and $\chi_{\tau_{J/\rho}}$ indicates the time dependency of $\widehat{\boldsymbol{J}}$ and $\widehat{\rho}$ and can be *constant* (C), *linear* (L), or *quadratic* (Q).

The other coefficients are given by

$$c_\omega = \cos(\omega\delta t/2), \tag{E14a}$$
$$s_\omega = \sin(\omega\delta t/2), \tag{E14b}$$
$$t_\omega = s_\omega/c_\omega, \tag{E14c}$$
$$c'_\omega = \cos\left((\omega - \boldsymbol{k}_m\cdot\boldsymbol{v})\Delta t/2\right), \tag{E14d}$$
$$s'_\omega = \sin\left((\omega - \boldsymbol{k}_m\cdot\boldsymbol{v})\Delta t/2\right), \tag{E14e}$$
$$\boldsymbol{k}_m = \boldsymbol{k} + 2\pi m/\Delta\boldsymbol{r}, \quad m \in Z, \tag{E14f}$$
$$t_{ck} = \tan(ck\delta t/2), \tag{E14g}$$
$$Y_9 = \frac{t_{ck}(8 - c^2k^2\delta t^2) - 4ck\delta t}{(1+C)(ck\delta t)^2}, \tag{E14h}$$
$$Y_{10} = \left(1 - \frac{2t_{ck}}{ck\delta t}\right), \tag{E14i}$$
$$\chi_\tau = Y_9\tilde{a}^\tau_\omega - t_{ck}\tilde{c}^\tau_\omega, \tag{E14j}$$
$$\psi_\tau = Y_{10}\tilde{b}^\tau_\omega. \tag{E14k}$$

TABLE IV. Matrix coefficients of the dispersion equation (E13a), based on the time dependency of the current and charge densities $\widehat{\boldsymbol{J}}$ and $\widehat{\rho}$ over one time subinterval, $\delta t = \Delta t/m$.

	Time dependency of $\widehat{\boldsymbol{J}}$ or $\widehat{\rho}$		
Coefficients	Constant ($\tau = 0$)	Linear ($\tau = 1$)	Quadratic ($\tau = 2$)
$\tilde{a}^\tau_\omega$	0	0	$(c_\omega - 1)$
$\tilde{b}^\tau_\omega$	0	s_ω	s_ω
$\tilde{c}^\tau_\omega$	1	c_ω	1
χ_τ	$-t_{ck}$	$-c_\omega t_{ck}$	$Y_9(c_\omega - 1) - t_{ck}$
ψ_τ	0	$-is_\omega Y_{10}$	$-is_\omega Y_{10}$

The dispersion relation is given by computing the determinant of $(\boldsymbol{M})$ using the Sarrus rule. Interestingly, when the charge and current densities have the same temporal dependency, e.g., with CC, LL, or QQ, the determinant simplifies to the straightforward expression

$$\det(\boldsymbol{M}) = \alpha_1\alpha_2, \tag{E15}$$

where

$$\alpha_1 = \widehat{T}^3\big[\xi_0(\beta_0\widehat{k}_z(\chi_\tau c_\omega t_{ck} + \psi_\tau s_\omega) - (\chi_\tau s_\omega + \psi_\tau c_\omega t_{ck})) + \left(c_\omega^2 t_{ck}^2 - s_\omega^2\right)\big], \tag{E16a}$$

$$\alpha_2 = \left(c_\omega^2 t_{ck}^2 - s_\omega^2\right) + \left(1-\beta_0^2\right)[(\xi_x\widehat{k}_x + \xi_z\widehat{k}_z)(\chi_\tau c_\omega t_{ck} + \psi_\tau s_\omega) + \psi_\tau c_\omega t_{ck}(\xi_0 + \xi_z\beta_0) + \chi_\tau(\xi_z c_\omega\beta_0 + \xi_0 s_\omega)]. \tag{E16b}$$

Here, such simplification is possible due to the presence of similar terms of opposite sign that cancel each other when the charge and current densities have the same time dependency. For example, terms like $(\psi_{\tau_J})^2 k_x k_z c_\omega^2 - \psi_{\tau_J}\psi_{\tau_\rho} k_x k_z c_\omega^2 = 0$, since $\psi_{\tau_J} = \psi_{\tau_\rho} = \psi_\tau$ $(\tau_J = \tau_\rho = \tau)$. Moreover, at the asymptotic limit, assuming that (i) $\delta\omega = \omega - \boldsymbol{k}_m\boldsymbol{v}_0$ is small and (ii) considering an ultrarelativistic regime, e.g., $\beta_0 = v_0/c = 1$, the determinant equation reduces to

$$\xi_0\left(\widehat{k}_z(\chi_\tau c_{k_m v_0} t_{ck} + \psi_\tau s_{k_m v_0}) - (\chi_\tau s_{k_m v_0} + \psi_\tau c_{k_m v_0} t_{ck})\right) + \left(c^2_{k_m v_0} t^2_{ck} - s^2_{k_m v_0}\right) = 0, \tag{E17}$$

where $c_{k_m v_0} = \cos(\boldsymbol{k}_m\boldsymbol{v}\delta t/2)$, $s_{k_m v_0} = \sin(\boldsymbol{k}_m\boldsymbol{v}\delta t/2)$, and ξ_0^τ is proportional to $1/\delta\omega$ and reads

$$\xi_0^\tau = \frac{\widehat{T}\omega_p^2 S^2(\boldsymbol{k}_m)}{\gamma_0 ck}\frac{1}{\delta\omega} + \frac{\widehat{T}\omega_p^2}{\gamma_0 ck}\sum_{j=-\infty,\ m\neq j}^{+\infty} S^2(\mathbf{k_j})\cdot\frac{1}{\frac{2}{\Delta t}s'_{k_j v_0}} = \frac{\alpha_m}{\delta\omega} + \beta_m. \tag{E18}$$

Finally, we obtain a first-order equation for $\delta\omega$ with real coefficients,

$$\delta\omega = -\frac{\alpha_m\left(\widehat{k}_z(\chi_\tau c_{k_m v_0 t_{ck}} + \psi_\tau s_{k_m v_0}) - (\chi_\tau s_{k_m v_0} + \psi_\tau c_{k_m v_0 t_{ck}})\right)}{\beta_m\left(\widehat{k}_z(\chi_\tau c_{k_m v_0 t_{ck}} + \psi_\tau s_{k_m v_0}) - (\chi_\tau s_{k_m v_0} + \psi_\tau c_{k_m v_0 t_{ck}})\right) + (c^2_{k_m v_0} t^2_{ck} - s^2_{k_m v_0})}. \tag{E19}$$

It follows that, under assumptions (i)–(ii), the determinant has only real coefficients, $\delta\omega$ is real, and the algorithm is stable.

[1] O. Buneman, C. W. Barnes, J. C. Green, and D. E. Nielsen, Principles and capabilities of 3-D, E-M particle simulations, J. Comput. Phys. **38**, 1 (1980).

[2] J. M. Dawson, Particle simulation of plasmas, Rev. Mod. Phys. **55**, 403 (1983).

[3] C. K. Birdsall and A. B. Langdon, *Plasma Physics Via Computer Simulation* (Adam-Hilger, London, UK, 1991).

[4] T. Tajima and J. Dawson, Laser electron-accelerator, Phys. Rev. Lett. **43**, 267 (1979).

[5] P. Chen, J. M. Dawson, R. W. Huff, and T. Katsouleas, Acceleration of electrons by the interaction of a bunched electron beam with a plasma, Phys. Rev. Lett. **54**, 693 (1985).

[6] I. Haber, R. Lee, H. Klein, and J. Boris, Advances in electromagnetic simulation techniques, in *Proc. Sixth Conf. Num. Sim. Plasmas, Berkeley, CA* (1973), pp. 46–48.

[7] J.-L. Vay, I. Haber, and B. B. Godfrey, A domain decomposition method for pseudo-spectral electromagnetic simulations of plasmas, J. Comput. Phys. **243**, 260 (2013).

[8] K. Yee, Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media, IEEE Trans. Antennas Propag. Ap **14**, 302 (1966).

[9] H. Vincenti and J.-L. Vay, Detailed analysis of the effects of stencil spatial variations with arbitrary high-order finite-difference Maxwell solver, Comput. Phys. Commun. **200**, 147 (2016).

[10] H. Vincenti and J.-L. Vay, Ultrahigh-order Maxwell solver with extreme scalability for electromagnetic PIC simulations of plasmas, Comput. Phys. Commun. **228**, 22 (2018).

[11] E. Zoni, R. Lehe, O. Shapoval, D. Belkin, N. Zaïm, L. Fedeli, H. Vincenti, and J. L. Vay, A hybrid nodal-staggered pseudo-spectral electromagnetic particle-in-cell method with finite-order centering, Comput. Phys. Commun. **279**, 108457 (2022).

[12] O. Shapoval, R. Lehe, M. Thévenet, E. Zoni, Y. Zhao, and J.-L. Vay, Overcoming timestep limitations in boosted-frame particle-in-cell simulations of plasma-based acceleration, Phys. Rev. E **104**, 055311 (2021).

[13] R. Lehe, M. Kirchen, B. B. Godfrey, A. R. Maier, and J.-L. Vay, Elimination of numerical cherenkov instability in flowing-plasma particle-in-cell simulations by using Galilean coordinates, Phys. Rev. E **94**, 053305 (2016).

[14] M. Kirchen, R. Lehe, B. B. Godfrey, I. Dornmair, S. Jalas, K. Peters, J.-L. Vay, and A. R. Maier, Stable discrete representation of relativistically drifting plasmas, Phys. Plasmas **23**, 100704 (2016).

[15] B. Godfrey, Numerical Cherenkov instabilities in electromagnetic particle codes, J. Comput. Phys. **15**, 504 (1974).

[16] J.-L. Vay, Noninvariance of space- and time-scale ranges under a lorentz transformation and the implications for the study of relativistic interactions, Phys. Rev. Lett. **98**, 130405 (2007).

[17] J.-L. Vay and C. Deutsch, A three-dimensional electromagnetic particle-in-cell code to simulate heavy ion beam propagation in the reaction chamber, Fusion Eng. Des. **32**, 467 (1996).

[18] A. B. Langdon, On enforcing Gauss law in electromagnetic particle-in-cell codes, Comput. Phys. Commun. **70**, 447 (1992).

[19] B. Marder, A method for incorporating Gauss law into electromagnetic PIC codes, J. Comput. Phys. **68**, 48 (1987).

[20] C.-D. Munz, P. Omnes, R. Schneider, E. Sonnendrücker, and U. Voß, Divergence correction techniques for Maxwell solvers based on a hyperbolic model, J. Comput. Phys. **161**, 484 (2000).

[21] B. Fornberg, High-order finite differences and the pseudospectral method on staggered grids, SIAM J. Numer. Anal. **27**, 904 (1990).

[22] J.-L. Vay, A. Almgren, J. Bell, L. Ge, D. Grote, M. Hogan, O. Kononenko, R. Lehe, A. Myers, C. Ng, J. Park, R. Ryne, O. Shapoval, M. Thévenet, and W. Zhang, Warp-X: A new exascale computing platform for beam-plasma simulations, Nucl. Instrum. Methods Phys. Res. Sect. A **909**, 476 (2018).

[23] J.-L. Vay, A. Almgren, L. Amorim, J. Bell, L. Ge, K. Gott, D. Grote, M. Hogan, A. Huebl, R. Jambunathan, R. Lehe, A. Myers, C. Ng, J. Park, M. Rowan, O. Shapoval, M. Thévenet, W. Zhang, Y. Zhao, and E. Zoni, Toward the modeling of chains of plasma accelerator stages with WarpX, J. Phys.: Conf. Ser. **1596**, 012059 (2020).

[24] J. L. Vay, A. Huebl, A. Almgren, L. D. Amorim, J. Bell, L. Fedeli, L. Ge, K. Gott, D. P. Grote, M. Hogan, R. Jambunathan, R. Lehe, A. Myers, C. Ng, M. Rowan, O. Shapoval, M. Thévenet, H. Vincenti, E. Yang, N. Zaïm *et al.*, Modeling of a chain of three plasma accelerator stages with the WarpX electromagnetic PIC code on GPUs, Phys. Plasmas **28**, 023105 (2021).

[25] J. L. Vay, C. G. R. Geddes, E. Esarey, C. B. Schroeder, W. P. Leemans, E. Cormier-Michel, and D. P. Grote, Modeling of 10 Gev-1 TeV laser-plasma accelerators using Lorentz boosted simulations, Phys. Plasmas **18**, 123103 (2011).

[26] M. Kirchen, R. Lehe, S. Jalas, O. Shapoval, J. L. Vay, and A. R. Maier, Scalable spectral solver in Galilean coordinates for eliminating the numerical Cherenkov instability in particle-in-cell simulations of streaming plasmas, Phys. Rev. E **102**, 013202 (2020).

[27] A. Spitkovsky, Particle acceleration in relativistic collisionless shocks: Fermi process at last? Astrophys. J. **682**, L5 (2008).

[28] https://github.com/ECP-WarpX/WarpX.

[29] A. B. Langdon and C. K. Birdsall, Theory of Plasma Simulation Using Finite-Size Particles, Phys. Fluids **13**, 2115 (1970).

[30] R. W. Hockney and J. W. Eastwood, *Computer Simulation Using Particles* (Taylor & Francis, Bristol, UK, 1988).