Lawrence Berkeley National Laboratory
LBL Publications

Title
Overcoming timestep limitations in boosted-frame Particle-In-Cell simulations of plasma-based acceleration

Permalink
https://escholarship.org/uc/item/2zb490jb

Authors
Shapoval, Olga
Lehe, Remi
Thévenet, Maxence
et al.

Publication Date
2021-04-28

Peer reviewed

Overcoming timestep limitations in boosted-frame Particle-In-Cell simulations of plasma-based acceleration

Olga Shapoval[a],[*] Remi Lehe[a],[†] Maxence Thévenet[a],[‡]
Edoardo Zoni[a], Yinjian Zhao[a], Jean-Luc Vay[a]
[a]Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA

April 30, 2021

Abstract

Explicit electromagnetic Particle-In-Cell (PIC) codes are typically limited by the Courant-Friedrichs-Lewy (CFL) condition, which implies that the timestep multiplied by the speed of light must be smaller than the smallest cell size. In the case of boosted-frame PIC simulations of plasma-based acceleration, this limitation can be a major hinderance as the cells are often very elongated along the longitudinal direction and the timestep is thus limited by the small, transverse cell size. This entails many small-timestep PIC iterations, and can limit the potential speed-up of the boosted-frame technique. Here, by using a CFL-free analytical spectral solver, and by mitigating additional numerical instabilities that arise at large timestep, we show that it is possible to overcome traditional limitations on the timestep and thereby realize the full potential of the boosted-frame technique over a much wider range of parameters.

Particle-In-Cell (PIC) simulations [1, 2] are key to the development of plasma-based accelerators and of their potential future applications [3]. However, these simulations can typically be very computationally expensive. One way to reduce their computational cost is to use the boosted-frame technique [4], whereby the simulation is performed in a Lorentz frame moving relativistically in the same direction as the beam or laser driver. The boosted-frame technique is nowadays routinely used in simulations of plasma-based accelerators, and can speed up simulations by several orders of magnitude. To a large extent, this was made possible by the development of a number of algorithms that mitigate the Numerical Cherenkov Instability (NCI) [5–20] – a numerical instability that would otherwise rapidly grow in the boosted frame and irremediably corrupt the simulated physics.

The remarkable speedup afforded by the boosted-frame technique is due largely to the possibility of increasing the timestep in the boosted frame, and thereby reducing the number of PIC iterations to be performed compared to a corresponding laboratory-frame simulation. For example, in a typical laboratory-frame simulation of laser-wakefield acceleration, both the longitudinal cell size

---

[*]CONTACT O. Shapoval. Email: oshapoval@lbl.gov
[†]CONTACT R. Lehe. Email: rlehe@lbl.gov
[‡]Now at DESY, 22607 Hamburg, Germany

Δz_{lab} and timestep Δt_{lab} are constrained to resolve the small laser oscillations at wavelength λ_{lab}: $\Delta z_{lab} \ll \lambda_{lab}$ for a laser propagating along z, with $c\Delta t_{lab} \leq \Delta z_{lab} \ll \lambda_{lab}$ (while the transverse cell size is usually much larger: $\Delta x, \Delta y \gg \Delta z_{lab}$). By contrast, in a Lorentz boosted frame drifting along z at relativistic velocity with a Lorentz factor $\gamma_b \gg 1$, the laser oscillations are dilated by a factor of approximately $2\gamma_b$ ($\lambda \approx 2\gamma_b\lambda_{lab}$), which greatly relaxes the constraints on the longitudinal cell size and timestep: $\Delta z \ll 2\gamma_b\lambda_{lab}$, with $c\Delta t \leq \Delta z \ll 2\gamma_b\lambda_{lab}$ (where the quantities Δz and Δt denote the longitudinal cell size and timestep *in the boosted frame*).

However, for large γ_b, as the constraints imposed by the laser are relaxed, the timestep often becomes constrained instead by the transverse cell size: $c\Delta t \leq \Delta x, \Delta y$. (Note that the transverse cell size is left unchanged in the boosted-frame simulation as compared to the corresponding laboratory-frame simulation, since transverse physical length scales are unchanged by the Lorentz transform.) In the case of Finite-Difference Time-Domain (FDTD) PIC algorithms, this constraint on the timestep is due to the Courant-Friedrichs-Lewy (CFL) condition [21, 22]. Similarly, the Pseudo-Spectral Time-Domain (PSTD) PIC algorithm [23] also has a CFL condition. As a consequence of the CFL condition, the timestep of the boosted-frame simulation is relatively small and limits the potential computational speedup, even though the physics at stake does not necessarily require such a high temporal resolution.

On the other hand, unlike FDTD and PSTD PIC algorithms, Pseudo-Spectral Analytical Time-Domain (PSATD) PIC algorithms [24, 25], which integrate analytically Maxwell's equations over one time step in Fourier space, do not have a similar CFL condition. It follows that boosted-frame PIC simulations that use the PSATD Maxwell solver could use a larger timestep, as it is thus not explicitly constrained by the transverse resolution. However, it turns out that PSATD boosted-frame simulations are empirically unstable for $c\Delta t > \Delta x, \Delta y$. More specifically, the Galilean PSATD algorithm [16–18], which does efficiently mitigate the NCI for $c\Delta t < \Delta x, \Delta y$, does not seem to suppress the NCI anymore for $c\Delta t > \Delta x, \Delta y$.

This paper examines the nature of this resurgent NCI and shows that this instability can be strongly mitigated with a new algorithm, referred to as the *averaged Galilean PSATD*, whereby a key feature of the PSATD algorithm is exploited to analytically average the electromagnetic fields in time before gathering them onto the macroparticles. Hence, with this new algorithm, simulations can run with a large timestep ($c\Delta t \leq \Delta z \gg \Delta x, \Delta y$) and exhibit the corresponding computational speedup, while preserving the integrity of the simulated physics. While this development was motivated here with the example of laser-wakefield acceleration, it is generally applicable to any simulation where the physics imposes a high transverse spatial resolution but does not impose such strong constraints on the timestep, so that it would be advantageous to use a large timestep compared to the cell size. For instance, this also includes the simulations of low-emittance pencil-like beams [26], in which the space charge requires a high transverse resolution, but has a relatively slow time evolution.

The remainder of the paper is structured as follows. We first examine in more detail the NCI that occurs for large timesteps in the case of the standard Galilean PSATD algorithm. Based on this analysis, we introduce the averaged Galilean PSATD algorithm and describe the corresponding modified PIC loop. We then demonstrate the stability of this new algorithm with large timesteps, first for a uniform plasma, and then for 2D simulations of laser-wakefield acceleration (LWFA) and 3D simulations of plasma wakefield acceleration (PWFA).

Results

Limitations of the standard Galilean PSATD algorithm for large timesteps

As mentioned in the introduction, boosted-frame simulations with the Galilean PSATD algorithm are typically unstable when using a large timestep $c\Delta t > \Delta x, \Delta y$. Here, we illustrate this by examining the theoretical NCI growth rate of the Galilean PSATD algorithm for two-dimensional simulations of a uniform plasma drifting at a relativistic velocity $\boldsymbol{v_0} = v_0 \boldsymbol{u}_z$ (where $\boldsymbol{u}_z$ is the unit vector along the z axis). As a reminder, the Galilean PSATD algorithm solves the Maxwell equations on a moving grid, which drifts at a velocity $\boldsymbol{v}_{gal} = v_{gal} \boldsymbol{u}_z$. This algorithm was shown to suppress the NCI when $v_{gal} = v_0$ [16].

In this section, we in fact consider two cases: that of a matching velocity $v_{gal} = v_0$ and that of a slightly detuned velocity $v_{gal} = 0.99 \, v_0$. Conceptually, these two cases represent – at a simplified level – different areas of the simulation box, in the case of a realistic LWFA simulation. More specifically, the matched case ($v_{gal} = v_0$) represents the background, quiescent plasma, far from the driver and the wakefield, since the Galilean velocity is typically chosen to match its velocity (i.e., $v_{gal} = -\sqrt{1 - 1/\gamma_b^2}\, c$). On the other hand, the case of the detuned velocity represents the perturbed plasma around the laser driver and in the wakefield, where the local velocity is different than that of the background plasma, and thus different than the Galilean velocity.

In both of these cases, we choose $\Delta z \gg \Delta x$. This is typical for boosted-frame simulations with a large γ_b, since the longitudinally Lorentz-dilated driver and wakefield relax the requirement on the longitudinal resolution. We then further consider two cases: that of a small timestep $c\Delta t = \Delta x$ and that of a large timestep $c\Delta t = \Delta z$. Note that the latter case would not be allowed by the CFL condition of an FDTD algorithm.

Fig. 1 displays the theoretical NCI growth rate for the four possible combinations (i.e., small/large timestep and matched/detuned Galilean velocity). The growth rates are obtained by solving the theoretical dispersion relation, namely equation (19) in [16]. In order to guide the interpretation of this figure, we also plot the position of well-known NCI resonant modes [8], which are caused by temporal and spatial aliasing. For the Galilean PSATD algorithm, the equation of these aliased resonant modes is given by:

$$k_{x,res} = \sqrt{\left(k_z \frac{v_0}{c} + m_z \frac{2\pi}{\Delta z} \frac{(v_0 - v_{gal})}{c} - \frac{2\pi n}{c\Delta t}\right)^2 - k_z^2}, \quad m_z, n \in \mathbb{Z} \tag{1}$$

where m_z is the spatial alias index and n is the temporal alias index [18]. As one can observe, if $v_{gal} \approx v_0$, the term proportional to m_z almost cancels and the position of these lines mainly depends on the time aliasing n.

As can be seen in Fig. 1 (a)-(b), in the matched-velocity case ($v_{gal} = v_0$), the Galilean PSATD algorithm suppresses the NCI, both for $c\Delta t = \Delta x$ (upper left panel) and $c\Delta t \gg \Delta x$ (upper right panel). By contrast, in the detuned case ($v_{gal} \neq v_0$, lower panels in Fig. 1), the NCI has a more noticeable growth rate. This growth rate is relatively small for $c\Delta t = \Delta x$, but is much larger for $c\Delta t \gg \Delta x$. In practice, this implies that the Galilean algorithm is relatively robust to velocity perturbations (e.g. in the wakefield) in the case of a small timestep ($c\Delta t \leq \Delta x$), but it is much less robust to those perturbations in the case of a large timestep ($c\Delta t \gg \Delta x$). This explains the empirical observation, mentioned in the introduction, that boosted-frame simulations of LWFA are typically unstable with large timestep.
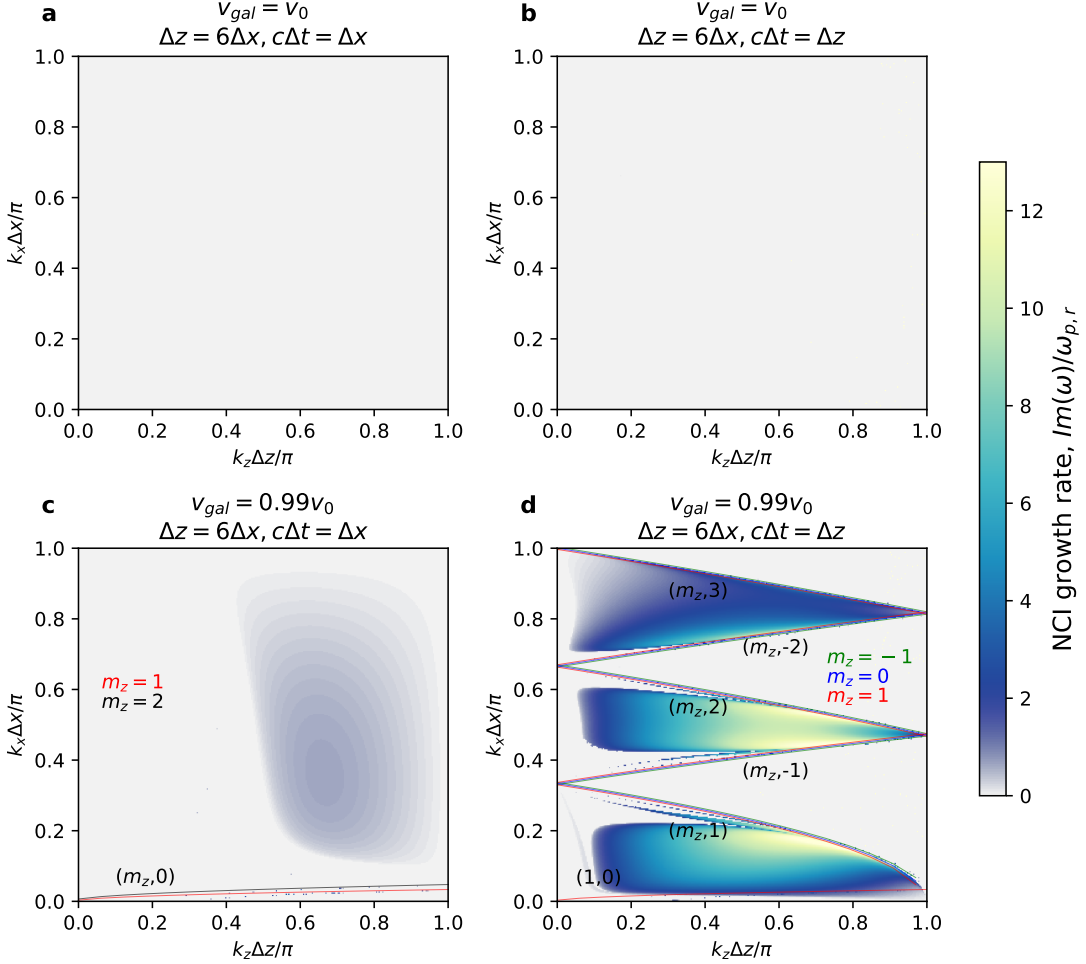
3

Figure 1: **NCI growth rate of the Galilean PSATD scheme.** Normalized NCI growth rate $Im(\omega)/\omega_{p,r}$ in spectral (k_x,k_z) space, calculated from the analytical stability analysis for different Galilean velocities $v_{gal} = v_0$ (a)-(b) and $v_{gal} = 0.99\,v_0$ (c)-(d), and for different timesteps $c\Delta t = \Delta x$ (left) and $c\Delta t = 6\Delta x$ (right). The solid colored lines correspond to well-known aliased NCI resonant modes, with alias number (m_z, n), as given by equation (1). In this simulation, a uniform plasma drifts at a velocity $v_0 = c(1 - 1/\gamma_b^2)^{1/2}$ with $\gamma_b = 130$, and the transverse and longitudinal cell sizes are $\Delta x = 6.4 \times 10^{-2}\, k_{p,r}^{-1}$ and $\Delta z = 6\Delta x$, respectively (where $k_{p,r}^2 = n_0 e^2/(m_e \epsilon_0 c^2 \gamma_0)$, and where n_0 is the plasma density).

Furthermore, in the case $v_{gal} \neq v_0$ and $c\Delta t \gg \Delta x$ (panels (c)-(d) in Fig. 1), we see that the large NCI growth rate is concentrated near time-aliased resonances. Thus, the NCI arises here primarily from the resonant interaction of particles with electromagnetic modes that are not resolved in time. More specifically, the electromagnetic modes are in principle oscillating in time (as predicted by the analytical formulas used in the derivation of the PSATD algorithm), and in most cases their

net effect on particles averages to zero. Yet because these fields are sampled with a (large) discrete timestep – which can be comparable to the period of their oscillations – particles may in certain cases see an almost constant value (because of aliasing) instead of an oscillating one, and thus experience a lasting, resonant effect.

Averaged Galilean PSATD algorithm

The above analysis suggests a natural remedy: when pushing the particles, instead of using the value of the fields sampled at a specific time $t = n\Delta t$ for some integer n, the particles should instead be pushed with the fields *averaged in time* between $t = (n-1/2)\Delta t$ and $t = (n+1/2)\Delta t$. Averaging the fields over one timestep will barely affect the physics, provided that it is well-resolved in time with the chosen timestep. On the other hand, this average will damp the under-resolved modes that are spuriously resonant in Fig. 1. Since this is a *temporal* average, not a *spatial* one, it will not affect the above-mentioned fine spatial details that typically impose a high transverse resolution Δx, Δy (for example, the space-charge field of a low-emittance beam), as long as they vary slowly in time (in comparison to the time step used in the simulation).

We note that with most Maxwell solvers (for example, the FDTD and PSTD algorithms), the time evolution of the electromagnetic fields *within* one time step is in general not known. However, this evolution is indeed known in the case of the PSATD algorithm. More specifically, as part of the derivation of the PSATD algorithm [16, 24, 25], the time evolution of the $\boldsymbol{E}$ and $\boldsymbol{B}$ fields in Fourier space is calculated analytically. Here, we propose to average this analytical expression over one timestep in Fourier space (see equations (10) and (11) in the Methods section), and then to transform these averaged fields $\langle \boldsymbol{E} \rangle$ and $\langle \boldsymbol{B} \rangle$ to real space, where they are gathered onto the macroparticles and then discarded. (However, the *unaveraged* $\boldsymbol{E}$ and $\boldsymbol{B}$ fields are still kept in memory, and are updated by the standard Galilean PSATD equations [16,17] at each PIC iteration.) The corresponding modified PIC loop is illustrated in Fig. 2 and described in more detail in the Methods section.

In the rest of this article, we refer to this new scheme as the *averaged Galilean PSATD algorithm*, since it combines the Galilean PSATD scheme [16–18] and the temporal average of the fields over one timestep. By construction, the averaged Galilean PSATD algorithm inherits the main advantages of the Galilean PSATD scheme: it has a low amount of spurious numerical dispersion (for high-order spatial derivatives [27,28]) and does not have a CFL limit. In addition, as shown in the next sections, the averaged Galilean PSATD algorithm efficiently mitigates the NCI for large timesteps.

Stability analysis for a uniform plasma drifting at relativistic velocity

In order to analyze the stability of the new averaged Galilean PSATD algorithm, we consider again the case of a two-dimensional uniform plasma drifting with a relativistic velocity. We derived the theoretical dispersion equation for this system, by using a similar method as for the standard Galilean PSATD algorithm [16] – while taking into account the additional average in time. The full derivation of this theoretical dispersion equation is given in the Supplementary Information. By solving this dispersion equation numerically, we can extract the NCI growth rate $Im(\omega)/\omega_{p,r}$ as a function of $\boldsymbol{k}$. In addition, we also performed actual PIC simulations for the same system. We used the PIC code WarpX, in which we implemented the averaged Galilean PSATD algorithm, and we then extracted the NCI growth rate of the NCI in post-processing.
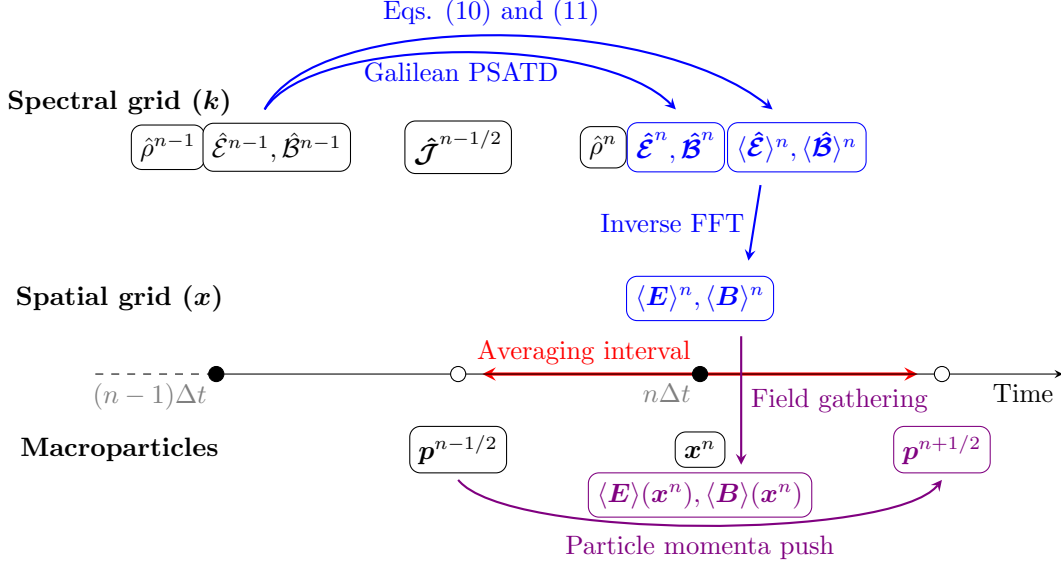
5

Figure 2: **Illustration of the field push and particle momenta push, in the averaged Galilean PSATD algorithm.** The quantities represented in black are the ones that are known just before the field and particle push. (Note that this includes the deposited charge and current $\hat{\rho}$ and $\hat{\mathcal{J}}$.) The quantities in blue and purple are the ones that are being computed during the field and particle push. As part of the field push (blue arrows), the regular fields $\hat{\mathcal{E}}$ and $\hat{\mathcal{B}}$ are updated, and the averaged fields $\langle\hat{\mathcal{E}}\rangle$ and $\langle\hat{\mathcal{B}}\rangle$ are calculated and transformed to the spatial grid. As part of the particle push, the averaged fields $\langle E \rangle$ and $\langle B \rangle$ are gathered onto the macroparticles, in order to update the macroparticles' momenta. The rest of the PIC cycle (e.g. charge and current deposition, particle position push) is not shown here but is identical to the standard Galilean PSATD [16, 17]. See the Methods section for more details on the PIC loop and exact definitions of the notation.

The growth rates extracted from both the WarpX simulations and the theoretical dispersion equation are shown in Fig. 3 - both for standard Galilean PSATD (left panels) and for the averaged Galilean PSATD (right panels). For this case, we used the same parameters as for the lower right panel of Fig. 1, $i.e.$, $\gamma_b = 130$, $\Delta x = 6.4 \times 10^{-2} \, k_{p,r}^{-1}$, a large timestep $c\Delta t = \Delta z = 6\Delta x$, and a detuned velocity $v_{gal} = 0.99 \, v_0$. (Recall from the previous sections that the case of a detuned velocity is the one for which using a large timestep presents a major issue.)

As can be seen in Fig. 3, the theoretical predictions (upper panels) and simulation results (lower panels) are in good agreement, which confirms that the theoretical dispersion equation correctly captures the nature of the instability. (Note that growth rate measured from simulations is typically noisy, which limits the comparison.) More importantly, both the theoretical predictions and simulation results show that averaged Galilean PSATD (right panels) strongly reduces the growth of the instability compared to the standard Galilean PSATD (left panels). This confirms that averaging the fields in time inhibits spurious resonances with under-resolved electromagnetic modes, and thereby enables stable simulations with large timesteps.
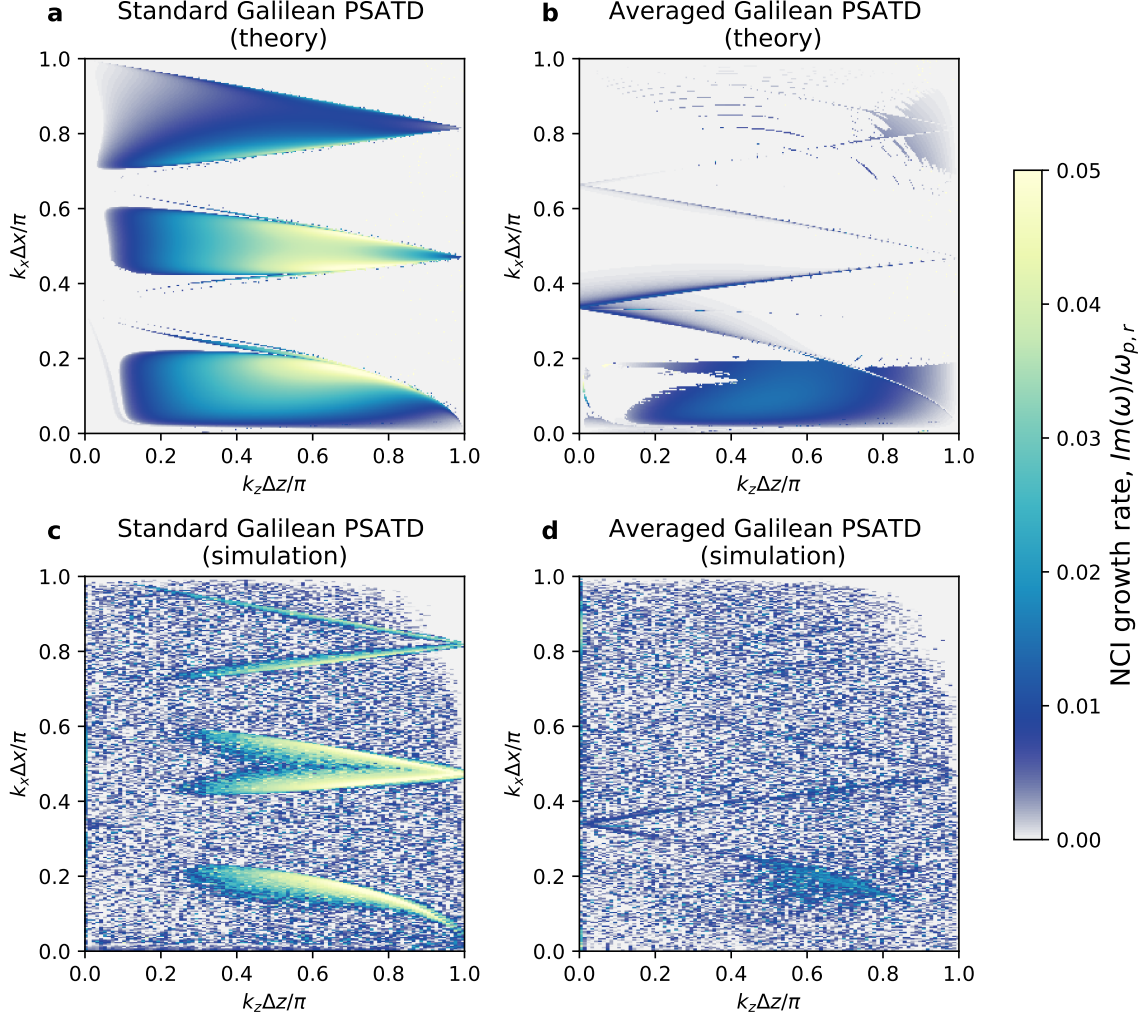
Figure 3: **NCI growth rate: Galilean PSATD vs. averaged Galilean PSATD schemes.**
Normalized NCI growth rate $\text{Im}(\omega)/\omega_{p,r}$ in spectral (k_x,k_z) space, calculated from the analytical
stability analysis (a)-(b) and from WarpX simulation results (c)-(d), obtained using the Galilean
PSATD and averaged Galilean PSATD schemes, at infinite spectral order, with large time step
$c\Delta t = \Delta z = 6\Delta x$ and slightly detuned Galilean velocity $v_{gal} = 0.99\,v_0$.

Application to simulations of plasma wakefield in a Lorentz-boosted frame

This section illustrates that the stability properties observed in the case of a uniform plasma also
apply to realistic simulations of plasma-based acceleration. To this end, we first perform two-
dimensional (2D) simulations of a laser-wakefield accelerator. In these simulations, an x-polarized
Gaussian laser pulse with amplitude $a_0 = 1$, duration $\tau = 20\,\text{fs}$ and waist $w_0 = 15\,\mu\text{m}$ propagates in
a matched parabolic plasma channel with a background density of $1.0 \times 10^{18}\,\text{cm}^{-3}$. The simulation

runs in a Lorentz-boosted frame ($\gamma_b = 30$) with a nodal PSATD solver with finite order 16 along the x direction and 32 along the z direction [18, 27, 28]. The longitudinal resolution (in the boosted frame) is set to $\Delta z = 2\gamma_b\lambda_{lab}/32 = 1.52\,\mu$m, while the transverse resolution is $\Delta x = 0.15\,\mu$m, so that $\Delta z = 10\Delta x$. We run the simulation with the standard and averaged Galilean PSATD, and with a small timestep ($c\Delta t = \Delta x$) as well as large timesteps ($c\Delta t = 5\Delta x$ and $c\Delta t = 10\Delta z$). Fig. 4) displays snapshots of the longitudinal electric field E_z and of the longitudinal current density J_z *in the boosted frame*, for these different cases. In these colormaps, the rapid oscillations of E_z for $z > 0\,$mm correspond to the longitudinal component of the laser field, which undergoes significant non-linear evolution and red-shifting, while the slow oscillations of E_z and J_z for $z < 0\,$mm correspond to the plasma wakefield.

As expected, the standard Galilean PSATD is stable for a small timestep (panel (a)), but unstable for large timesteps (panels (c) and (e)). More specifically, in panels (c) and (e), spurious oscillations rapidly grow in the wakefield and severely disrupt its structure. We also note that the results of the averaged Galilean PSATD with a small timestep (panel (b)) are almost indistinguishable from those of the standard Galilean PSATD (panel (a)) - thereby confirming that averaging the fields in time preserves the essential physics. More importantly, for large timestep (panel (d) and panel (f)), the averaged Galilean PSATD achieves stability while preserving the overall structure of the wakefield. Indeed for $c\Delta t = 5\Delta x$ (panel (d)) the E_z field is still almost indistinguishable from that of panel (a). For $c\Delta t = 10\Delta x$ (panel (f)), small differences become noticeable, especially in the red-shifted laser oscillations - although they hardly affect the structure of the accelerating wakefield. This may indicate that this value of Δt starts to reach the limit for which the simulation is not well-resolved in time anymore.

We also note that, both for $c\Delta t = 5\Delta x$ (panel (d)) and $c\Delta t = 10\Delta x$ (panel (f)), small transverse oscillations become noticeable in J_z for $z < -1.0\,$mm. These oscillations may be due to the remaining non-zero growth rate of the averaged Galilean algorithm (see the growth rates represented in panel (b) and (d) of Fig. 3). However, their magnitude is small enough that they do not lead to a modulation of the electric field, hence they do not affect the dynamics. Again, this represents a clear improvement compared to the standard Galilean algorithm (panel (b) and (e)).

We now go further and show that this algorithm is applicable to a full-scale three-dimensional (3D) setting, by considering the 3D simulation of charged particle beam-driven wakefield. In this simulation, a 1 nC Gaussian electron beam propagates in a plasma with a background density of $1.0 \times 10^{17}\,\text{cm}^{-3}$, and experiences a typical evolution whereby the head of the beam erodes while the tail of the beam performs betatron oscillations in the generated wakefield. The electron beam initially has a mean Lorentz factor $\gamma = 2000$, with a relative RMS spread $\Delta\gamma/\gamma = 0.01$, and a transverse and longitudinal RMS size of $5\,\mu$m and $20\,\mu$m respectively. The simulation is run in a Lorentz-boosted frame ($\gamma_b = 5.6$), with 512^3 cells of size $\Delta x = \Delta y = 0.78\,\mu$m, $\Delta z = 5\,\mu$m $= 6.4\,\Delta x$ (in the boosted frame), and a nodal PSATD solver with finite order 16 [18, 27, 28]. In order to verify again that the averaged Galilean PSATD algorithm preserves the physics of interest, we run the simulation both with the standard Galilean PSATD algorithm and a small timestep $c\Delta t = \Delta x$ (fiducial case) and with the averaged Galilean PSATD algorithm and a large timestep $c\Delta t = \Delta z$. In both cases, we ran the WarpX code on the Summit supercomputer, using 24 GPUs with domain decomposition along z.

The top panels of Figure 5 display colormaps of the wakefield in the laboratory frame, which were reconstructed on-the-fly during the boosted-frame simulation. Again, the simulation with the large
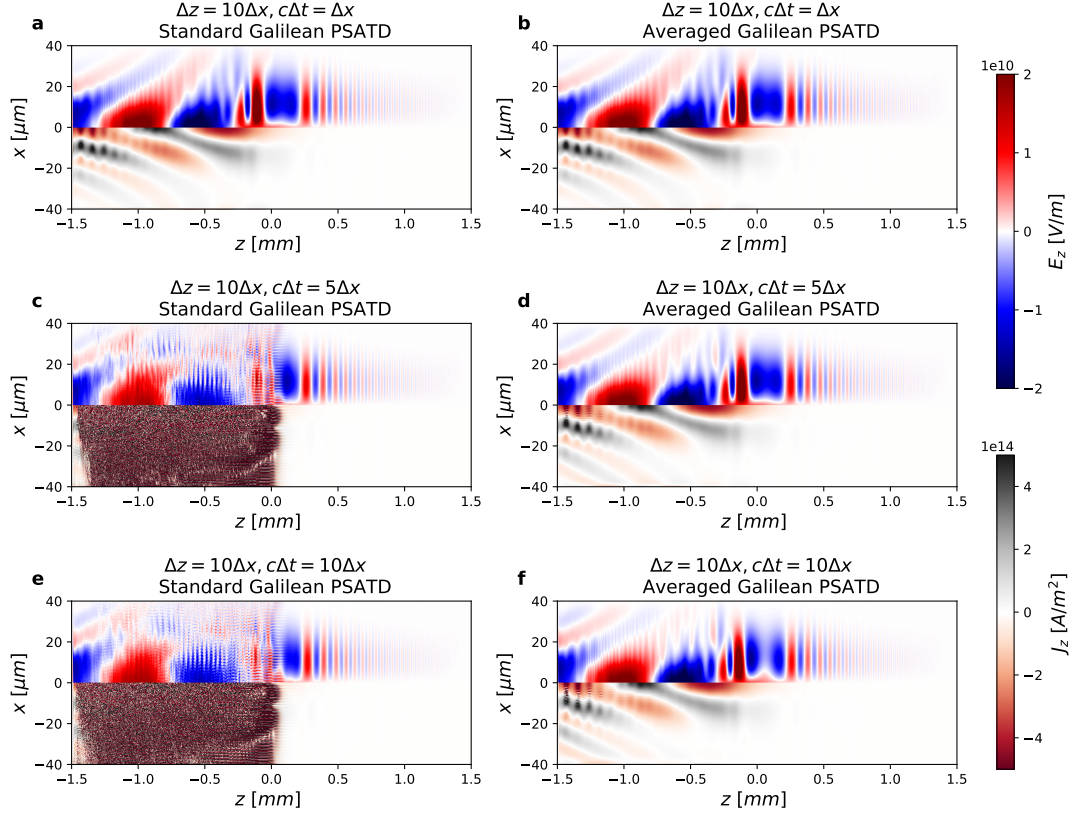
Figure 4: **Instability mitigation in a 2D laser-wakefield simulation with large timestep**. The upper half of each subplot shows the E_z field seen by the macro-particles (*i.e.*, the regular E_z field for the standard Galilean PSATD, and the averaged $\langle E_z \rangle$ field for the averaged Galilean PSATD). The lower half of each subplot shows the longitudinal current density J_z of the plasma. All the fields are shown in the boosted frame. The different subplots correspond to the standard Galilean PSATD (a,c,e) and averaged Galilean PSATD (b,d,f), with $c\Delta t = \Delta x$ (a-b), $c\Delta t = 0.5\Delta z = 5\Delta x$ (c-d) and $c\Delta t = \Delta z = 10\Delta x$ (e-f).

timestep and the averaged Galilean PSATD is stable (panel (b)) and the simulated wakefield is almost indistinguishable from that produced from the fiducial small-timestep simulation (panel (a)). In addition, panels (c) and (d) in Figure 5 show the evolution of the emittance and relative energy spread of the driver beam in the laboratory frame, as it undergoes head erosion and betatron oscillation. This is obtained from laboratory-frame particle data that is reconstructed on-the-fly during the boosted-frame simulation. As can be seen, the evolution of these beam quantities show excellent agreement between the fiducial small-timestep simulation and the large-timestep with the
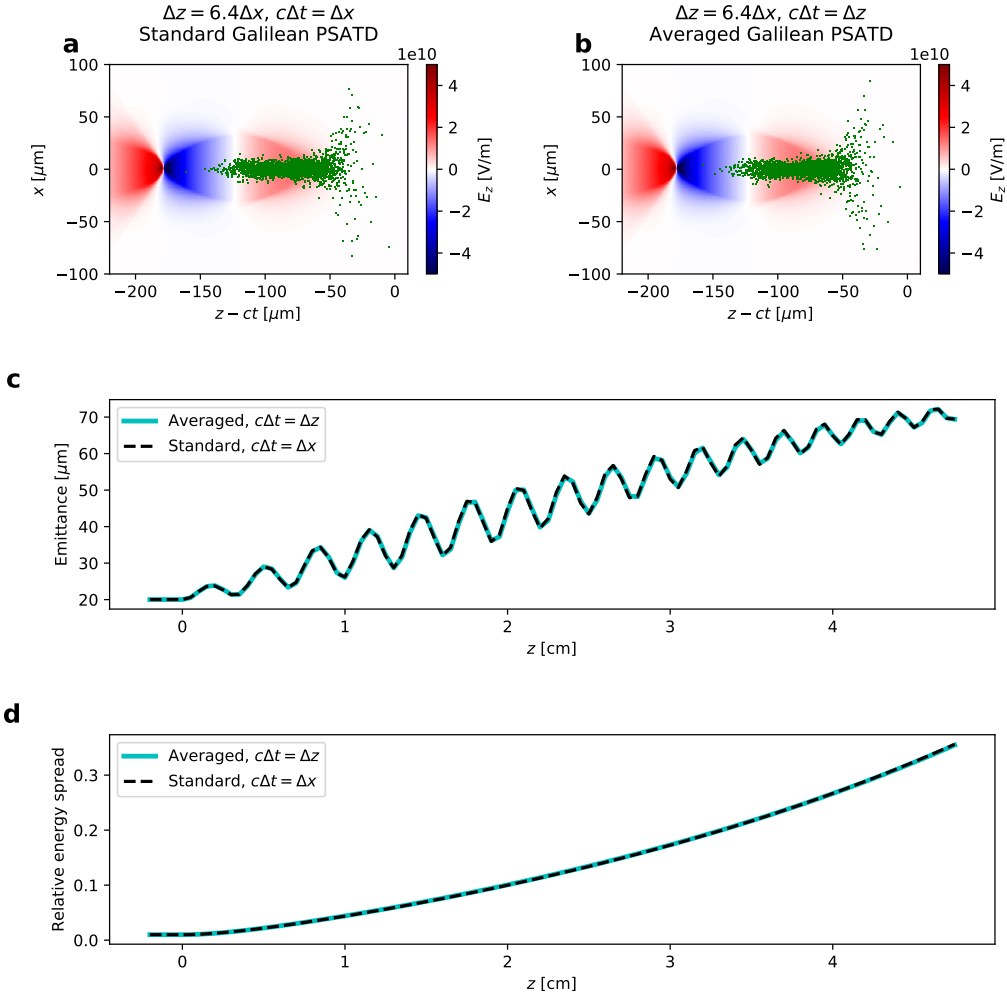
Figure 5: **Algorithm comparison for a 3D plasma-wakefield simulation.** (a,b) Snapshot of the E_z field ($\langle E_z \rangle$ in the case of the averaged Galilean PSATD) in the laboratory frame, for the small-timestep standard Galilean PSATD simulation (a) and large-timestep averaged Galilean PSATD simulation (b). The green dots are representative random samples of the macroparticles in the beam driver. (c,d) Evolution of the emittance and relative energy spread of the beam, in the laboratory frame.

averaged Galilean PSATD.

Thus, in the above example, the averaged Galilean PSATD allowed stable simulations to be run with a large timestep while preserving the integrity of the physics at stake. We note that, as a consequence of the large timestep, the simulation using the averaged Galilean PSATD exhibited a $5\times$ overall speed-up compared to the small-timestep standard Galilean PSATD simulation, on the Summit supercomputer.

Discussion

In this paper, we proposed a modified PIC algorithm that enables stable boosted-frame simulations of plasma-based acceleration with a large timestep $c\Delta t \gg \Delta x$, where Δx represents the smallest cell size. This was achieved by using a CFL-free Galilean PSATD solver and by averaging the $\boldsymbol{E}$ and $\boldsymbol{B}$ fields in time, in order to inhibit spurious resonances with under-resolved, aliased electromagnetic modes. We demonstrated this novel scheme in realistic 2D and 3D plasma-wakefield simulations.

We note that the proposed algorithm could certainly be further refined and improved upon in the future. For instance, although the proposed algorithm strongly reduces the NCI growth rate for large timesteps, it does not completely eliminate it. As a consequence, the NCI at large timesteps could still be an issue for certain sets of parameters. Similarly, we observed in separate tests that realistic 3D simulations of laser-driven wakefield could remain unstable in certain cases for large timesteps, even with the proposed algorithm.

In conclusion, this work demonstrates that it is possible to run boosted-frame simulations with a much larger timestep than the traditional CFL limit, while still accurately capturing the physics. This new development enables potential speedups of an order of magnitude or more, opening up a new area of investigation within the field of first-principles, Particle-In-Cell modeling of plasma-wakefield particle accelerators, whereby the simulation timestep is chosen much more freely than before.

Although the present work is focused on simulations of plasma accelerators and on a particular method (the averaged Galilean PSATD), it could have a wider impact. For instance, even though the algorithm proposed here builds specifically upon the PSATD framework, the central idea (namely averaging the fields in time) is fairly general and could thus also guide the future development of similar solutions for FDTD-based methods. In addition, beyond the plasma accelerator community, this work may be of interest to the modeling of advanced light sources concepts, coherent synchrotron radiation in particle accelerators, astrophysical shocks or beam-plasma instabilities of astrophysical relativistic jets, which can also utilize the boosted-frame PIC method for accurate modeling from first principles. We also envision that the method that is used in this paper can be employed to overcome similar timestep limitations in PIC simulations that do not employ the boosted-frame technique, with impact to a much wider range of applications.

Methods

While the main text gave a brief overview and discussed the main advantages of the averaged Galilean PSATD algorithm, in this section we present the mathematical details of the proposed scheme, including the derivation of the discretized averaged fields.

Derivation of the averaged fields

Throughout the paper, we use the notation $\langle \hat{\mathcal{F}} \rangle^n$ to refer to the averaged field within the time interval $[(n - \frac{1}{2})\Delta t, (n + \frac{1}{2})\Delta t]$, defined as

$$\langle \hat{\mathcal{F}} \rangle^n = \frac{1}{\Delta t} \int_{(n-1/2)\Delta t}^{(n+1/2)\Delta t} \hat{\mathcal{F}}(\boldsymbol{k}, \tau) d\tau \,, \tag{2}$$

where $\hat{\mathcal{F}} = \hat{\mathcal{F}}(\boldsymbol{k}, t) = \int \boldsymbol{F}(\boldsymbol{x}, t) e^{-i\boldsymbol{k}\cdot\boldsymbol{x}} d^3\boldsymbol{x}$ refers to the Fourier transform of the $\boldsymbol{F}(\boldsymbol{x}, t)$ field.

In the Galilean coordinates drifting at $\boldsymbol{v}_{gal}$, the p-order discretized Maxwell equations transformed to Fourier space read [18]:

$$\left(\frac{\partial}{\partial t} - i[\boldsymbol{k}] \cdot \boldsymbol{v}_{gal} \right)^2 \hat{\boldsymbol{B}} + c^2[k]^2 \hat{\boldsymbol{B}} = \frac{1}{\varepsilon_0} i[\boldsymbol{k}] \times \hat{\boldsymbol{J}}, \tag{3}$$

$$\left(\frac{\partial}{\partial t} - i[\boldsymbol{k}] \cdot \boldsymbol{v}_{gal} \right)^2 \hat{\boldsymbol{\mathcal{E}}} + c^2[k]^2 \hat{\boldsymbol{\mathcal{E}}} = -\frac{c^2}{\varepsilon_0} \hat{\rho} i[\boldsymbol{k}] - \frac{1}{\varepsilon_0} \left(\frac{\partial}{\partial t} - i[\boldsymbol{k}] \cdot \boldsymbol{v_{gal}} \right) \hat{\boldsymbol{J}}, \tag{4}$$

where $[k] = \sqrt{[\boldsymbol{k}]^2} = \sqrt{[k_x]^2 + [k_y]^2 + [k_z]^2}$ and where $[k_u]$ with $u = \{x, y, z\}$ is the Fourier transform of the p-order discretized stencil $\hat{\nabla}_u$ (i.e., such that a p-order Taylor expansion yields $\hat{\nabla}_u F = \partial_u F + O(\Delta u^p)$.) [27, 28].

As explained in [16], these equations can be integrated analytically under the assumption that the current $\hat{\boldsymbol{J}}$ is constant over one timestep, and that the fields $\hat{\boldsymbol{\mathcal{E}}}, \hat{\boldsymbol{B}}, \hat{\boldsymbol{J}}$ and $\hat{\rho}$ satisfy the conservation equations. More specifically, assuming that the $\boldsymbol{E}$ and $\boldsymbol{B}$ fields are known at $t = (n-1)\Delta t$, and under the assumption that $\hat{\boldsymbol{J}}(\boldsymbol{k}, t)$ is constant and equal to $\hat{\boldsymbol{J}}^{n-1/2}(\boldsymbol{k})$ over the time interval $[(n-1)\Delta t, n\Delta t]$, we can obtain the expressions of $\hat{\boldsymbol{\mathcal{E}}}(\boldsymbol{k}, t), \hat{\boldsymbol{B}}(\boldsymbol{k}, t)$ as a function of $\hat{\boldsymbol{\mathcal{E}}}^{n-1}, \hat{\boldsymbol{B}}^{n-1}, \hat{\boldsymbol{J}}^{n-1/2}, \hat{\rho}^{n-1}, \hat{\rho}^n$:

$$\hat{\boldsymbol{\mathcal{E}}}(\boldsymbol{k}, t) = \left[\hat{\boldsymbol{\mathcal{E}}}^{n-1} - \frac{\boldsymbol{\alpha}_1}{c^2[k]^2(1-\nu^2)} - \frac{\boldsymbol{\beta}_1}{c^2[k]^2} \right] \cos\left[c[k](t - (n-1)\Delta t)\right] e^{i\nu c[k](t-(n-1)\Delta t)}$$

$$+ \frac{\boldsymbol{\alpha}_1}{c^2[k]^2(1-\nu^2)} + \frac{\boldsymbol{\beta}_1}{c^2[k]^2} e^{i\nu c[k](t-(n-1)\Delta t)}$$

$$+ \frac{1}{c[k]} \left[c^2 i[\boldsymbol{k}] \times \hat{\boldsymbol{B}}^{n-1} - \frac{1}{\varepsilon_0} \hat{\boldsymbol{J}}^{n-1/2} + i\nu \frac{\boldsymbol{\alpha}_1}{c[k](1-\nu^2)} \right] \sin\left[c[k](t - (n-1)\Delta t)\right] e^{i\nu c[k](t-(n-1)\Delta t)}$$

$$\tag{5}$$

$$\hat{\boldsymbol{B}}(\boldsymbol{k}, t) = \left[\hat{\boldsymbol{B}}^{n-1} - \frac{\boldsymbol{\alpha}_2}{c^2[k]^2(1-\nu^2)} \right] \cos\left[c[k](t - (n-1)\Delta t)\right] e^{i\nu c[k](t-(n-1)\Delta t)} + \frac{\boldsymbol{\alpha}_2}{c^2[k]^2(1-\nu^2)}$$

$$+ \frac{1}{c[k]} \left[-i[\boldsymbol{k}] \times \hat{\boldsymbol{\mathcal{E}}}^{n-1} + i\nu \frac{\boldsymbol{\alpha}_2}{c[k](1-\nu^2)} \right] \sin\left[c[k](t - (n-1)\Delta t)\right] e^{i\nu c[k](t-(n-1)\Delta t)}$$

$$\tag{6}$$

where $\nu = [\boldsymbol{k}] \cdot \boldsymbol{v_{gal}}/c[k]$ and

$$\boldsymbol{\alpha_1} = \frac{i\nu c[k]}{\varepsilon_0} \boldsymbol{\hat{\mathcal{J}}}^{n-1/2} - \frac{c^2}{\varepsilon_0} \frac{\hat{\rho}^n - \hat{\rho}^{n-1} e^{i\nu c[k]\Delta t}}{1 - e^{i\nu c[k]\Delta t}} i[\boldsymbol{k}] \tag{7}$$

$$\boldsymbol{\beta_1} = \frac{c^2}{\varepsilon_0} \frac{\hat{\rho}^n - \hat{\rho}^{n-1}}{1 - e^{i\nu c[k]\Delta t}} i[\boldsymbol{k}] \tag{8}$$

$$\boldsymbol{\alpha_2} = \frac{1}{\varepsilon_0} i[\boldsymbol{k}] \times \boldsymbol{\hat{\mathcal{J}}}^{n-1/2} \tag{9}$$

Strictly speaking, these expressions of $\boldsymbol{\hat{\mathcal{E}}}(\boldsymbol{k},t)$, $\boldsymbol{\hat{\mathcal{B}}}(\boldsymbol{k},t)$ in Eqns. (5) and (6) are only valid for t in the interval $[(n-1)\Delta t, n\Delta t]$ (because of the assumption $\boldsymbol{\hat{\mathcal{J}}}(\boldsymbol{k},t) = \boldsymbol{\hat{\mathcal{J}}}^{n-1/2}(\boldsymbol{k})$). However, we assume that there are also approximately valid on the interval $[(n-1/2)\Delta t, (n+1/2)\Delta t]$ over which the fields are averaged (see Fig. 2). This is valid if $\boldsymbol{\hat{\mathcal{J}}}$ varies slowly from one timestep to the next - *i.e.*, if the plasma response is well-resolved in time.

Under these assumptions, we average Eqns. (5) and (6) in time as defined in Eq. (2), and obtain:

$$\langle \boldsymbol{\hat{\mathcal{E}}} \rangle^n = \Psi_1 \boldsymbol{\hat{\mathcal{E}}}^{n-1} - ic\Psi_2 \frac{[\boldsymbol{k}]}{[k]} \times \boldsymbol{\hat{\mathcal{B}}}^{n-1} + \left(\frac{i\nu c[k]}{\varepsilon_0} A_1 + \frac{\Psi_2}{c[k]\varepsilon_0} \right) \boldsymbol{\hat{\mathcal{J}}}^{n-1/2}$$
$$+ C_\rho(A_2 - A_1)\hat{\rho}^n \frac{[\boldsymbol{k}]}{[k]} + C_\rho(\theta^2 A_1 - A_2)\hat{\rho}^{n-1} \frac{[\boldsymbol{k}]}{[k]} , \tag{10}$$

$$\langle \boldsymbol{\hat{\mathcal{B}}} \rangle^n = \Psi_1 \boldsymbol{\hat{\mathcal{B}}}^{n-1} + \frac{i}{c}\Psi_2 \frac{[\boldsymbol{k}]}{[k]} \times \boldsymbol{\hat{\mathcal{E}}}^{n-1} + \frac{i[k]}{\varepsilon_0} A_1 \frac{[\boldsymbol{k}]}{[k]} \times \boldsymbol{\hat{\mathcal{J}}}^{n-1/2} . \tag{11}$$

Here again, $\nu = [\boldsymbol{k}] \cdot \boldsymbol{v_{gal}}/c[k]$, $\theta = e^{i[\boldsymbol{k}]\cdot\boldsymbol{v_{gal}}\Delta t/2}$, and the other coefficients are given by:

$$C_\rho = \frac{ic^2[k]}{\varepsilon_0(1-\theta^2)} , \tag{12a}$$

$$A_1 = \frac{\Psi_1 - 1 + i\nu\Psi_2}{c^2[k]^2(\nu^2 - 1)} , \tag{12b}$$

$$A_2 = \frac{\Psi_3 - \Psi_1}{c^2[k]^2} , \tag{12c}$$

$$\Psi_1 = \theta \frac{(S_1 + i\nu C_1) - \theta^2(S_3 + i\nu C_3)}{c[k]\Delta t(\nu^2 - 1)} , \tag{12d}$$

$$\Psi_2 = \theta \frac{(C_1 - i\nu S_1) - \theta^2(C_3 - i\nu S_3)}{c[k]\Delta t(\nu^2 - 1)} , \tag{12e}$$

$$\Psi_3 = \frac{i\theta(1-\theta^2)}{c[k]\Delta t\nu} , \tag{12f}$$

with $C_m = \cos(m\,c[k]\,\Delta t/2)$ and $S_m = \sin(m\,c[k]\,\Delta t/2)$ for $m = 1,2,3$.

PIC cycle overview

Fig. 2 gives an overview of a key part of the PIC loop for the averaged Galilean PSATD algorithm. Here we describe the exact PIC loop in more detail. Assuming that we originally know the particle

positions and momenta $\boldsymbol{x}^n$ at $\boldsymbol{p}^{n-1/2}$ and the fields $\boldsymbol{E}^{n-1}$ and $\boldsymbol{B}^{n-1}$, the loop consists of the following steps:

1. Deposit the charge and current densities of the particles onto the spatial grid. In particular, we deposit the charge density ρ^n at time $t = n\Delta t$ from the particle positions $\boldsymbol{x}^n$ and the current density $\boldsymbol{J}^{n-1/2}$ at time $t = (n - \frac{1}{2})\Delta t$ from the particle positions $\boldsymbol{x}^{n-1}$ and $\boldsymbol{x}^n$ and the particle velocities $\boldsymbol{v}^{n-1/2}$;

2. Transform all relevant physical quantities from physical space to Fourier space;

3. Compute the new electromagnetic fields in Fourier space $\hat{\boldsymbol{\mathcal{E}}}^n$ and $\hat{\boldsymbol{\mathcal{B}}}^n$, from the charge and current densities $\hat{\rho}^{n-1}$ (available from the previous PIC iteration), $\hat{\rho}^n$ and $\hat{\boldsymbol{\mathcal{J}}}^{n-1/2}$;

4. Compute the averaged electromagnetic fields, $\langle\hat{\boldsymbol{\mathcal{E}}}\rangle^n$ and $\langle\hat{\boldsymbol{\mathcal{B}}}\rangle^n$;

5. Transform all relevant physical quantities from Fourier space back to physical space;

6. Gather the averaged electromagnetic fields, in physical space, from the spatial grid onto the particles;

7. Push the particles by updating their positions and momenta based on the current values of the averaged electromagnetic fields, $\langle\boldsymbol{E}\rangle^n$ and $\langle\boldsymbol{B}\rangle^n$, whose precise definition is given in the next section. In particular, the momenta are updated from $\boldsymbol{p}^{n-1/2}$ to $\boldsymbol{p}^{n+1/2}$ and the positions are then updated from $\boldsymbol{x}^n$ to $\boldsymbol{x}^{n+1}$

WarpX code

We implemented the averaged Galilean PSATD algorithm of arbitrary spectral order in the code WarpX [29] in Cartesian coordinates. WarpX is an open-source PIC code that combines advanced algorithms with adaptive mesh refinement to allow challenging simulations of a multi-stage plasma-based TeV acceleration relevant for future high-energy physics discoveries. WarpX relies on the ECP AMReX co-design center for mesh refinement and runs on CPU and GPU-accelerated platforms, and production simulations have run on Cori KNL at NERSC and on Summit at OLCF.

Code availability

WarpX code is available at `https://github.com/ECP-WarpX/WarpX`.

References

[1] R.W. Hockney and J.W. Eastwood. *Computer Simulation Using Particles*. Taylor & Francis, 1988.

[2] C.K. Birdsall and A.B. Langdon. *Plasma Physics via Computer Simulation, Appendix E*. Series in Plasma Physics. Taylor & Francis, 2004.

[3] Felicie Albert, Marie-Emmanuelle Couprie, Alexander D Debus, Mike Downer, Jerome Faure, Alessandro Flacco, Leonida A. Gizzi, Thomas E Grismayer, Axel Huebl, Chandrashekhar Joshi, Marie Labat, Wim P Leemans, Andreas Maier, Stuart Mangles, Paul Mason, Francois Mathieu,

Patric Muggli, Mamiko Nishiuchi, Jens Osterhoff, Pattahil P Rajeev, Ulrich Schramm, Joerg Schreiber, Alexander G R Thomas, Jean-Luc Vay, Marija Vranic, and Karl Zeil. 2020 roadmap on plasma accelerators. *New Journal of Physics*, 2020.

[4] 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, Mar 2007.

[5] Brendan B Godfrey. *Journal of Computational Physics*, 15(4):504 – 521, 1974.

[6] Brendan B. Godfrey. Canonical momenta and numerical instabilities in particle codes. *Journal of Computational Physics*, 19(1):58 – 76, 1975.

[7] J.-L. Vay, C.G.R. Geddes, E. Cormier-Michel, and D.P. Grote. *Journal of Computational Physics*, 230(15):5908 – 5929, 2011.

[8] Brendan B. Godfrey and Jean-Luc Vay. *Journal of Computational Physics*, 248:33 – 46, 2013.

[9] Xinlu Xu, Peicheng Yu, Samual F. Martins, Frank S. Tsung, Viktor K. Decyk, Jorge Vieira, Ricardo A. Fonseca, Wei Lu, Luis O. Silva, and Warren B. Mori. *Computer Physics Communications*, 184(11):2503 – 2514, 2013.

[10] Brendan B. Godfrey, Jean-Luc Vay, and Irving Haber. *Journal of Computational Physics*, 258:689 – 704, 2014.

[11] Brendan B. Godfrey and Jean-Luc Vay. Suppressing the numerical cherenkov instability in {FDTD} {PIC} codes. *Journal of Computational Physics*, 267:1 – 6, 2014.

[12] B.B. Godfrey, J.-L. Vay, and I. Haber. *Plasma Science, IEEE Transactions on*, 42(5):1339–1344, May 2014.

[13] Brendan B. Godfrey and Jean-Luc Vay. *Computer Physics Communications*, pages –, 2015.

[14] Peicheng Yu, Xinlu Xu, Viktor K. Decyk, Frederico Fiuza, Jorge Vieira, Frank S. Tsung, Ricardo A. Fonseca, Wei Lu, Luis O. Silva, and Warren B. Mori. Elimination of the numerical cerenkov instability for spectral em-pic codes. *Computer Physics Communications*, 192:32 – 47, 2015.

[15] Peicheng Yu, Xinlu Xu, Adam Tableman, Viktor K. Decyk, Frank S. Tsung, Frederico Fiuza, Asher Davidson, Jorge Vieira, Ricardo A. Fonseca, Wei Lu, Luis O. Silva, and Warren B. Mori. Mitigation of numerical cerenkov radiation and instability using a hybrid finite difference-fft maxwell solver and a local charge conserving current deposit. *Computer Physics Communications*, 197:144 – 152, 2015.

[16] Remi Lehe, Manuel Kirchen, Brendan B. Godfrey, Andreas R. Maier, and Jean-Luc Vay. Elimination of numerical cherenkov instability in flowing-plasma particle-in-cell simulations by using galilean coordinates. *Phys. Rev. E*, 94:053305, Nov 2016.

[17] 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. *Physics of Plasmas*, 23(10):100704, 2016.

[18] Manuel Kirchen, Remi Lehe, Soeren Jalas, Olga Shapoval, Jean-Luc Vay, and Andreas 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, Jul 2020.

[19] Fei Li, Peicheng Yu, Xinlu Xu, Frederico Fiuza, Viktor K. Decyk, Thamine Dalichaouch, Asher Davidson, Adam Tableman, Weiming An, Frank S. Tsung, Ricardo A. Fonseca, Wei Lu, and Warren B. Mori. Controlling the numerical cerenkov instability in pic simulations using a customized finite difference maxwell solver and a local fft based current correction. *Computer Physics Communications*, 214:6–17, 2017.

[20] Alexander Pukhov. X-dispersionless maxwell solver for plasma-based particle acceleration. *Journal of Computational Physics*, 418:109622, 2020.

[21] R. Courant, K. Friedrichs, and H. Lewy. über die partiellen differenzengleichungen der mathematischen physik. *Mathematische Annalen*, 100:32–74, 1928.

[22] Allen Taflove and Susan C. Hagness. *Computational electrodynamics: the finite-difference time-domain method*. Artech House, Norwood, 3rd edition, 2005.

[23] Q. H. Liu. The pstd algorithm: A time-domain method requiring only two cells per wavelength. *Microwave and Optical Technology Letters*, 15(3):158–165, 1997.

[24] I. Haber, R. Lee, H. Klein, and J. Boris. 1973.

[25] Jean-Luc Vay, Irving Haber, and Brendan B. Godfrey. *Journal of Computational Physics*, 243:260 – 268, 2013.

[26] T. J. Mehrling, C. Benedetti, C. B. Schroeder, and E. Esarey. A subgrid algorithm for the efficient modeling of plasma-based accelerators with ion motion using quasi-static particle-in-cell codes. In *2018 IEEE Advanced Accelerator Concepts Workshop (AAC)*, pages 1–5, 2018.

[27] H. Vincenti and J.-L. Vay. Detailed analysis of the effects of stencil spatial variations with arbitrary high-order finite-difference maxwell solver. *Computer Physics Communications*, 200:147–167, 2016.

[28] S. Jalas, I. Dornmair, R. Lehe, H. Vincenti, J.-L. Vay, M. Kirchen, and A. R. Maier. Accurate modeling of plasma acceleration with arbitrary order pseudo-spectral particle-in-cell methods. *Physics of Plasmas*, 24(3):033115, 2017.

[29] J.-L. Vay, A. Almgren, J. Bell, L. Ge, D.P. 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. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 909:476–479, 2018. 3rd European Advanced Accelerator Concepts workshop (EAAC2017).

Acknowledgments

Author contributions

O. S. derived the equations of the averaged Galilean PSATD algorithm, implemented them in the WarpX code, and extensively tested the algorithm. R. L. proposed the initial idea of the averaged Galilean PSATD algorithm and provided guidance for the derivation and implementation of the algorithm. J. V. initiated the research into an algorithm that enables larger time steps, helped the implementation and testing of an early prototype of the new algorithm in the Warp code and provided guidance to the overall project and paper writing. M. T. and Y. Z. performed additional simulations, and E. Z. and M. T. made major contributions to the implementation in WarpX. O. S. and R. L. wrote the paper, with key inputs from M. T., E. Z., Y. Z. and J. V.

Funding

Competing interests

The authors declare no competing interests.

Additional information

The derivation of the dispersion relation for the averaged Galilean PSATD algorithm is given in Supplementary Information.

Supplementary information: Derivation of the Dispersion Relation for the Averaged Galilean PSATD Algorithm

Here we derive the 2D dispersion relation to study the NCI induced by a relativistic plasma flowing through a periodic grid along the z-axis with velocity $\boldsymbol{v_0} = v_0 \boldsymbol{u_z}$, where $v_0 = c(1 - 1/\gamma_b^2)^{1/2}$. This is done by combining the discretized Maxwell equations in the Galilean frame and the discretized Vlasov equation, and assuming small perturbations for the electromagnetic fields $\boldsymbol{E}, \boldsymbol{B}$ and the distribution function δf. Because the Vlasov equation involves the averaged fields $\langle \boldsymbol{E} \rangle$, $\langle \boldsymbol{B} \rangle$, we include their expression as a function of the regular fields $\boldsymbol{E}$, $\boldsymbol{B}$ in the system of equations. Hence, when expressed in spectral space, the different equations of the system are:

- the discretized Maxwell equations at n-th time step in time-symmetrical form [16]:

$$\theta^* c\hat{\boldsymbol{\mathcal{B}}}^n - \theta c\hat{\boldsymbol{\mathcal{B}}}^{n-1} = -t_{ck}\frac{i\boldsymbol{k}\times(\theta^*\hat{\boldsymbol{\mathcal{E}}}^n + \theta\hat{\boldsymbol{\mathcal{E}}}^{n-1})}{k} + 2\chi_4'\frac{\hat{T}}{\varepsilon_0 ck}\frac{\boldsymbol{k}\times\hat{\boldsymbol{\mathcal{J}}}_d^{n-1/2}}{k}\,, \tag{13}$$

$$\theta^*\hat{\boldsymbol{\mathcal{E}}}^n - \theta\hat{\boldsymbol{\mathcal{E}}}^{n-1} = t_{ck}\frac{i\boldsymbol{k}\times(\theta^* c\hat{\boldsymbol{\mathcal{B}}}^n + \theta c\hat{\boldsymbol{\mathcal{B}}}^{n-1})}{k} - \frac{\hat{T}i\boldsymbol{k}}{\varepsilon_0 k^2}(\theta^*\hat{\rho}^n - \theta\hat{\rho}^{n-1})$$
$$- 2\chi_4\frac{\hat{T}}{\varepsilon_0 ck}\left[\hat{\boldsymbol{\mathcal{J}}}_d^{n-1/2} - \frac{(\boldsymbol{k}\cdot\hat{\boldsymbol{\mathcal{J}}}_d^{n-1/2})\boldsymbol{k}}{k^2}\right]; \tag{14}$$

- the perturbed Vlasov equation [16]. Note that, here, we replaced the regular fields $\hat{\boldsymbol{\mathcal{E}}},\hat{\boldsymbol{\mathcal{B}}}$ by the averaged fields $\langle\hat{\boldsymbol{\mathcal{E}}}\rangle,\langle\hat{\boldsymbol{\mathcal{B}}}\rangle$ in order to take into account the changes associated with the averaged Galilean PSATD.

$$\delta\hat{f}^{n+1/2}(\boldsymbol{k_m},\boldsymbol{p})e^{i\boldsymbol{k_m}(\boldsymbol{v}-\boldsymbol{v_{gal}})\Delta t/2} - \hat{f}^{n-1/2}(\boldsymbol{k_m},\boldsymbol{p})e^{-i\boldsymbol{k_m}(\boldsymbol{v}-\boldsymbol{v_{gal}})\Delta t/2}$$
$$+ q\Delta t\hat{S}(\boldsymbol{k_m})\left[\langle\hat{\boldsymbol{\mathcal{E}}}^n(\boldsymbol{k})\rangle + \boldsymbol{v}\times\langle\hat{\boldsymbol{\mathcal{B}}}^n(\boldsymbol{k})\rangle\right]\cdot\frac{\partial f_0}{\partial\boldsymbol{p}} = 0\,; \tag{15}$$

- the expression of the averaged field for $t\in[(n-1/2)\Delta t,(n+1/2)\Delta t]$:

$$\langle\hat{\boldsymbol{\mathcal{B}}}^n(\boldsymbol{k},t)\rangle = \Psi_1\hat{\boldsymbol{\mathcal{B}}}^{n-1} + \frac{i}{c}\Psi_2\frac{\boldsymbol{k}}{k}\times\hat{\boldsymbol{\mathcal{E}}}^{n-1} + \frac{ik}{\varepsilon_0}A_1\frac{\boldsymbol{k}}{k}\times\hat{\boldsymbol{\mathcal{J}}}^{n-1/2}\,, \tag{16}$$

$$\langle\hat{\boldsymbol{\mathcal{E}}}^n(\boldsymbol{k},t)\rangle = \Psi_1\hat{\boldsymbol{\mathcal{E}}}^{n-1} - ic\Psi_2\frac{\boldsymbol{k}}{k}\times\hat{\boldsymbol{\mathcal{B}}}^{n-1} + \left(\frac{i\nu ck}{\varepsilon_0}A_1 + \frac{\Psi_2}{ck\varepsilon_0}\right)\hat{\boldsymbol{\mathcal{J}}}^{n-1/2}$$
$$+ C_\rho(A_2 - A_1)\hat{\rho}^n\frac{\boldsymbol{k}}{k} + C_\rho(\theta^2 A_1 - A_2)\hat{\rho}^{n-1}\frac{\boldsymbol{k}}{k}\,. \tag{17}$$

Here, $\hat{T} = \prod_i\left[1 - \sin(k_i\Delta i/2)\right]$ represents a one-pass binomial smoother, and $\hat{S}(\boldsymbol{k_m})$ is the particle shape factor, with $\boldsymbol{k_m} = \boldsymbol{k} + \boldsymbol{K_m}$ ($\boldsymbol{K_m} = 2\pi\sum_i u_i m_i/\Delta i$) for $i = \{x, y, z\}$.

As follows from the discrete continuity equation, the corrected current $\hat{\boldsymbol{\mathcal{J}}}^{n-1/2}$ satisfies

$$\hat{\boldsymbol{\mathcal{J}}}^{n-1/2} = \hat{\boldsymbol{\mathcal{J}}}_d^{n-1/2} - \frac{(\boldsymbol{k}\cdot\hat{\boldsymbol{\mathcal{J}}}_d^{n-1/2})\boldsymbol{k}}{k^2} + \frac{(\boldsymbol{k}\cdot\boldsymbol{v_{gal}})\boldsymbol{k}}{k^2}\frac{\hat{\rho}^n - \hat{\rho}^{n-1}\theta^2}{1 - \theta^2}\,. \tag{18}$$

As in [16], we use the following Ansatz for the electromagnetic modes:

$$\hat{\boldsymbol{\mathcal{E}}}^n(\boldsymbol{k}) = \hat{\boldsymbol{\mathcal{E}}}(\boldsymbol{k})e^{-i(\omega-\boldsymbol{k}\cdot\boldsymbol{v_{gal}})n\Delta t}\,, \tag{19a}$$

$$\langle\hat{\boldsymbol{\mathcal{E}}}^n(\boldsymbol{k})\rangle = \langle\hat{\boldsymbol{\mathcal{E}}}(\boldsymbol{k})\rangle e^{-i(\omega-\boldsymbol{k}\cdot\boldsymbol{v_{gal}})n\Delta t}\,, \tag{19b}$$

$$\delta\hat{f}^{n-1/2}(\boldsymbol{k_m},\boldsymbol{p}) = \delta\hat{f}(\boldsymbol{k_m},\boldsymbol{p})e^{-i(\omega-\boldsymbol{k}\cdot\boldsymbol{v_{gal}})(n-1/2)\Delta t}\,, \tag{19c}$$

$$\hat{\boldsymbol{\mathcal{J}}}_d^{n-1/2}(\boldsymbol{k}) = \hat{\boldsymbol{\mathcal{J}}}_d(\boldsymbol{k})e^{-i(\omega-\boldsymbol{k}\cdot\boldsymbol{v_{gal}})(n-1/2)\Delta t}\,, \tag{19d}$$

$$\hat{\rho}^n(\boldsymbol{k}) = \hat{\rho}(\boldsymbol{k})e^{-i(\omega-\boldsymbol{k}\cdot\boldsymbol{v_{gal}})n\Delta t}\,, \tag{19e}$$

and after some amount of algebra, we derive the following equations for $\hat{\boldsymbol{\mathcal{J}}}_d(\boldsymbol{k})$ and $\hat{\rho}(\boldsymbol{k})$ from the Vlasov equation (see a similar derivation in [16]):

$$\hat{\boldsymbol{\mathcal{J}}}_d = i\frac{ck\varepsilon_0}{\hat{T}}\left(\xi_1\langle\hat{\boldsymbol{\mathcal{F}}}\rangle + (\boldsymbol{\xi_2}\cdot\langle\hat{\boldsymbol{\mathcal{F}}}\rangle)\frac{\boldsymbol{v_0}}{c}\right),\tag{20}$$

$$\hat{\rho} = \frac{ik\varepsilon_0}{\hat{T}}(\boldsymbol{\xi_3}\cdot\langle\hat{\boldsymbol{\mathcal{F}}}\rangle),\tag{21}$$

where $\langle\hat{\boldsymbol{\mathcal{F}}}(\boldsymbol{k})\rangle = \langle\hat{\boldsymbol{\mathcal{E}}}(\boldsymbol{k})\rangle + \boldsymbol{v_0}\times\langle\hat{\boldsymbol{\mathcal{B}}}(\boldsymbol{k})\rangle - (\boldsymbol{v_0}\cdot\langle\hat{\boldsymbol{\mathcal{E}}}(\boldsymbol{k})\rangle)\boldsymbol{v_0}/c^2$.

In addition, by substituting the expressions (18) and (19) into (13), (14), (16), the problem is reduced to the following set of equations to be solved:

$$s_\omega c\hat{\boldsymbol{\mathcal{B}}} = t_{ck}c_\omega\frac{\boldsymbol{k}\times\hat{\boldsymbol{\mathcal{E}}}}{k} + \chi_4'\frac{\hat{T}}{\varepsilon_0 ck}\frac{i\boldsymbol{k}\times\hat{\boldsymbol{\mathcal{J}}}_d}{k},\tag{22}$$

$$s_\omega\hat{\boldsymbol{\mathcal{E}}} = -c_\omega t_{ck}\frac{\boldsymbol{k}c\hat{\boldsymbol{\mathcal{B}}}}{k} - is_\omega\frac{\hat{T}\boldsymbol{k}}{\varepsilon_0 k^2}\hat{\rho} - i\chi_4\frac{\hat{T}}{\varepsilon_0 ck}\left[\hat{\boldsymbol{\mathcal{J}}}_d - \frac{(\boldsymbol{k}\cdot\hat{\boldsymbol{\mathcal{J}}}_d)\boldsymbol{k}}{k^2}\right],\tag{23}$$

$$\theta^2 c\langle\hat{\boldsymbol{\mathcal{B}}}\rangle = \Psi_1 c\hat{\boldsymbol{\mathcal{B}}}e^{i\omega\Delta t} + i\Psi_2\frac{\boldsymbol{k}\times\hat{\boldsymbol{\mathcal{E}}}}{k}e^{i\omega\Delta t} + \frac{ickA_1\hat{T}\theta}{\varepsilon_0}\frac{\boldsymbol{k}\times\hat{\boldsymbol{\mathcal{J}}}_d}{k}e^{\frac{i\omega\Delta t}{2}},\tag{24}$$

$$\theta^2\langle\hat{\boldsymbol{\mathcal{E}}}\rangle = \Psi_1\hat{\boldsymbol{\mathcal{E}}}e^{i\omega\Delta t} - ic\Psi_2\frac{\boldsymbol{k}}{k}\times\hat{\boldsymbol{\mathcal{B}}}e^{i\omega\Delta t} + \frac{iA_\nu\hat{T}}{ck\varepsilon_0}\theta e^{\frac{i\omega\Delta t}{2}}\left[\hat{\boldsymbol{\mathcal{J}}}_d - \frac{(\boldsymbol{k}\cdot\hat{\boldsymbol{\mathcal{J}}}_d)\boldsymbol{k}}{k^2}\right]$$
$$+\frac{i\boldsymbol{k}\hat{\rho}\hat{T}}{\varepsilon_0 k^2}\frac{c^2 k^2 A_2(\theta^2 - e^{i\omega\Delta t}) + \theta^2(c^2 k^2 A_1 - \nu A_\nu)(e^{i\omega\Delta t} - 1)}{(1-\theta^2)}.\tag{25}$$

Here, $A_\nu = \nu c^2 k^2 A_1 - i\Psi_2$, and the $\xi_1, \boldsymbol{\xi}_{2,3}$ coefficients represent the plasma response (for more details see Appendix A in [16]):

$$\xi_1 = \frac{\hat{T}\omega_p^2}{\gamma_0 ck}\sum_m\frac{\hat{S}^2(\boldsymbol{k_m})}{s_\omega'}, \quad \boldsymbol{\xi_2} = \frac{\hat{T}\omega_p^2}{\gamma_0 k}\sum_m\frac{c_{\omega'}\hat{S}^2(\boldsymbol{k_m})}{s_{\omega'}^2}\boldsymbol{k_m}, \quad \boldsymbol{\xi_3} = \frac{\hat{T}\omega_p^2}{\gamma_0 k}\sum_m\frac{\hat{S}^2(\boldsymbol{k_m})}{s_{\omega'}^2}\boldsymbol{k_m},\tag{26}$$

where

$$c_{\omega'} = \cos\left(\frac{\omega - \boldsymbol{k}\cdot\boldsymbol{v}_0 - \boldsymbol{K_m}(\boldsymbol{v_0} - \boldsymbol{v}_{gal})}{2\Delta t^{-1}}\right), \quad s_{\omega'} = \frac{2}{\Delta t}\sin\left(\frac{\omega - \boldsymbol{k}\cdot\boldsymbol{v}_0 - \boldsymbol{K_m}(\boldsymbol{v_0} - \boldsymbol{v}_{gal})}{2\Delta t^{-1}}\right).\tag{27}$$

By projecting equations (23) and (25) along y and equations (20), (22) and (24) along x and z, the final system of equations can be written in the matrix form

$$\boldsymbol{M_{av}U} = 0,\tag{28}$$

where $\boldsymbol{M_{av}}$ is the block matrix

$$
\boldsymbol{M_{av}} = \begin{bmatrix}
 & & & -\theta^2 & 0 & 0 & & & \\
 & \boldsymbol{M} & & 0 & -\theta^2 & 0 & & \boldsymbol{N} & \\
 & & & 0 & 0 & -\theta^2 & & & \\
0 & 0 & 0 & & & & 1 & 0 & 0 \\
0 & 0 & 0 & & \boldsymbol{P} & & 0 & 1 & 0 \\
0 & 0 & 0 & & & & 0 & 0 & 1 \\
 & & & 0 & 0 & 0 & & & \\
 & \boldsymbol{R} & & 0 & 0 & 0 & & \boldsymbol{Q} & \\
 & & & 0 & 0 & 0 & & &
\end{bmatrix}
\tag{29}
$$

and $\boldsymbol{U}$ is the vector

$$
\boldsymbol{U} = \left(c\hat{\mathcal{B}}_y, \hat{\mathcal{E}}_z, \hat{\mathcal{E}}_x, c\langle\hat{\mathcal{B}}_y\rangle, \langle\hat{\mathcal{E}}_z\rangle, \langle\hat{\mathcal{E}}_x\rangle, \frac{\hat{\mathcal{J}}_z}{ck\varepsilon_0}, \frac{\hat{\mathcal{J}}_x}{ck\varepsilon_0}, \frac{\hat{\rho}}{k\varepsilon_0} \right)^T .
\tag{30}
$$

The resulting dispersion relation is given by the determinant equation

$$
\det \boldsymbol{M_{av}} = 0.
\tag{31}
$$

Here, the individual matrices defining $\boldsymbol{M_{av}}$ read

$$
\boldsymbol{M} = e_\omega \begin{bmatrix}
\Psi_1 & -i\Psi_2 k_{xn} & i\Psi_2 k_{zn} \\
-i\Psi_2 k_{xn} & \Psi_1 & 0 \\
i\Psi_2 k_{zn} & 0 & \Psi_1
\end{bmatrix} ,
\tag{32a}
$$

$$
\boldsymbol{N} = \theta \sqrt{e_\omega} \, \hat{T} \begin{bmatrix}
-iA_1 k_{xn} & iA_1 k_{zn} & 0 \\
ik_{xn}^2 A_\nu & -ik_{xn}k_{zn}A_\nu & ik_{zn}r_{\omega\nu} \\
-ik_{xn}k_{zn}A_\nu & ik_{zn}^2 A_\nu & ik_{xn}r_{\omega\nu}
\end{bmatrix} ,
\tag{32b}
$$

$$
\boldsymbol{P} = \frac{1}{\hat{T}} \begin{bmatrix}
i\beta_0^2 \xi_{2x} & -i(1-\beta_0^2)(\beta_0\xi_{2z}+\xi_1) & -i\beta_0\xi_{2x} \\
i\beta_0\xi_1 & 0 & -i\xi_1 \\
i\beta_0\xi_{3x} & -i(1-\beta_0^2)\xi_{3z} & -i\xi_{3x}
\end{bmatrix} ,
\tag{32c}
$$

$$
\boldsymbol{R} = \begin{bmatrix}
s_\omega & c_\omega k_{xn} t_{ck} & -c_\omega k_{zn} t_{ck} \\
c_\omega k_{xn} t_{ck} & s_\omega & 0 \\
-c_\omega k_{zn} t_{ck} & 0 & s_\omega
\end{bmatrix} ,
\tag{32d}
$$

$$
\boldsymbol{Q} = \hat{T} \begin{bmatrix}
ik_{xn}\chi_4' & -ik_{zn}\chi_4' & 0 \\
ik_{xn}^2 \chi_4 & -ik_{xn}k_{zn}\chi_4 & ik_{zn}s_\omega \\
-ik_{xn}k_{zn}\chi_4 & ik_{zn}^2 \chi_4 & ik_{xn}s_\omega
\end{bmatrix} ,
\tag{32e}
$$

with $k_{xn} = k_x/k$, $k_{zn} = k_z/k$, $\beta_0 = v_0/c$, $c_\omega = \cos(\omega\Delta t/2)$, $s_\omega = \sin(\omega\Delta t/2)$, and

$$
r_{\omega\nu} = \frac{\theta^*}{\sqrt{e_\omega}} \frac{A_2(\theta^2 - e^{i\omega\Delta t}) + \theta^2(A_1 - \nu A_\nu)(e^{i\omega\Delta t} - 1)}{(1-\theta^2)} .
\tag{33}
$$

20

Even though the matrix $\boldsymbol{M_{av}}$ has multiple zeros entries, it is difficult to find an analytical solution of equation (31) for any pair (k_x, k_z). To solve it numerically, we used the secant method as a root-finding algorithm, which allowed us to calculate the NCI growth rates across a wide range of frequencies.

We remark that in the case of the standard Galilean PSATD scheme, $\boldsymbol{M_{av}}$ reduces to

$$\boldsymbol{M_{av}} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{Q} \\ \boldsymbol{P} & \boldsymbol{I} \end{bmatrix}, \tag{34}$$

which is equivalent to equation (19) of [16].