

**Note on implementation of a semi-implicit, energy-conserving
PIC**

Patrick Kilian

Space Science Institute

(Dated: September 23, 2023)

I. INTRODUCTION

This PiC code follows closely the reference in [1]. The basic idea is to keep particle updates in position and velocity full explicit, but to include changes to the current from future fields. This leads to the requirements for deposition of mass matrices and for solving a matrix equation for the future fields, but no subcycling is necessary. The method conserves energy and can be used on cell sizes much larger than a Debye length since the Finite-Grid-Instability is removed.

II. VALIDATION TESTS

A. Two-Stream Instability

This test follows the benchmark test described in [1, section 4.1]. Pick a reference density n_0 for normalization. Compute plasma frequency $\omega_{pe} = \sqrt{4\pi n_o e^2/m_e}$ and skin depth $d_e = c/\omega_{pe}$. The simulation domain is only resolved in one spatial dimension and is $L_x = 2\pi d_e$ long and resolved by $N_x = 64$ cells. This implies $\Delta x = L_x/N_x = \pi/32d_e \approx .098d_e$, or in terms of Debye length $\Delta x \approx 9.8\lambda_D$. This resolution is borderline, but feasible with regular PiC codes. We use a timestep of $\Delta t = 1/8\omega_{pe}^{-1}$ and follow the system for $T = 125\omega_{pe}^{-1}$, which implies we perform $N_t = 1000$ time steps. The CFL condition number is given by $\mathcal{C} = c\Delta t/\Delta x = 1/8c\omega_{pe}^{-1}/(\pi/32c\omega_{pe}^{-1}) = 4/\pi \approx 1.27$. This is again borderline, it's possible in electrostatic explicit PiC codes ($\mathcal{C} < \pi/2$), but not in (most) electromagnetic PiC codes. Two electron populations of equal density $n_0/2$ are initialized with beam speeds $v_{beam,x} = \pm 0.2c$, thermal spread $v_{th,e} = 0.01c$ and a small perturbation $\delta v = 0.001c$.

We use exactly 154 particles per cell (following [1], even though 156 ppc would get us closer to 10^4 particles in total) positioned at offsets that are distributed uniformly randomly in the cell for each particle in that cell. Linear theory predicts (according to [1]), that the fastest growing mode has $m = 3$ (corresponding to $k = 2\pi m/L_x$) and its growth rate is $\gamma = 0.35\omega_{pe}$. And this is also the m that we pick for the small perturbations of particle velocities. We do not set any background magnetic fields.

Since this is an one dimensional electrostatic problem we can run this test case with all three different simulation codes:

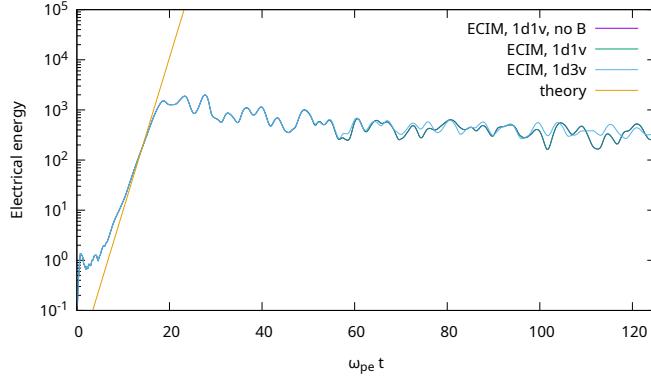


FIG. 1: Growth of the electric energy in the three simulation, compared against the analytic prediction of $\exp(2\gamma t)$. Compare with [1, Figure 1].

1. 1d1v phase space, only solving for E_x . B_x is required to be homogeneous due to the divergence constraint. It will remain unchanged in time since there is no currents $j_{y,z}$ and it would only modify $v_{y,z}$ and can hence be completely removed from the code.
2. 1d1v phase space, but retaining E_x and B_x . B_x of course still needs to remain homogeneous and static, but we can check if the Maxwell solver (the most complicated part of the algorithm) reproduces it, instead of enforcing it be construction.
3. 1d3v phase space, retaining all six field components. Thermal fluctuations in $v_{y,z}$ will produce transverse current and hence transverse fields, but none of these fluctuations should have a strong impact on the electrostatic instability. Late term behaviour might be changed due to particles scattering of magnetic field fluctuations.

To check the validity of the code we look at three diagnostics:

In figure 1 we reproduce the first result figure of this plot in the reference paper and look at the growth of the electric field energy $\mathcal{E}_E = \sum_g (E_x^2 + E_y^2 + E_z^2) V_d / (8\pi)$ as a function of time and compare against the analytic growth rate. Both 1d1v runs produce identical results to within machine precision and are not distinguishable in the plot. The 1d3v run agrees on growth rate and saturation but shows somewhat different late time behaviour.

In figure 2 we reproduce the second result figure and check conservation of total energy. With carefully selected tolerances in the matrix solver we achieve $|E(t) - E_0| < 10^{-15} E_0$ for the duration of the simulation run. This is using a setting of `tol = 10^-6/c` and `atol = 10^-15` for nearly all timesteps, automatically falling back to `tol = 10^-5/c` or `tol = 10^-4/c`

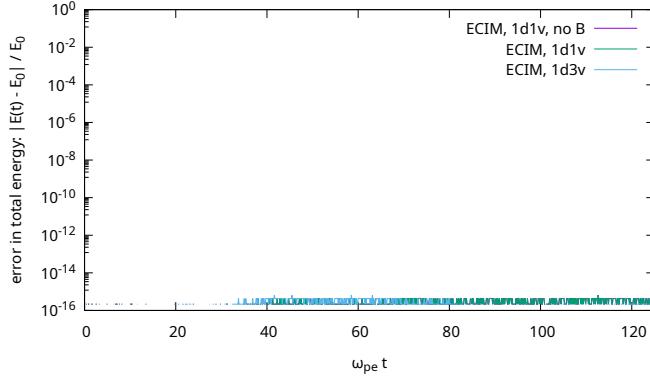


FIG. 2: Change in total energy over the full simulation duration for all three simulations.

Compare with [1, Figure 2].

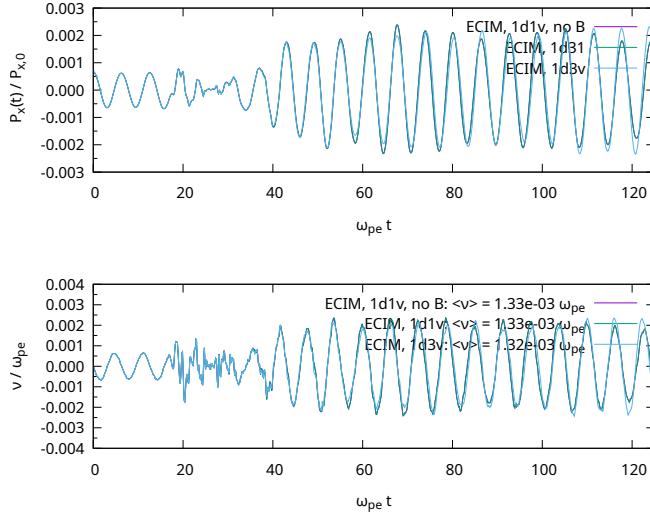


FIG. 3: Fluctuations in total particle momentum (upper panel) and numerical collision frequency (lower panel).

where GMRES fails to converge with this setting. Larger values for `tol` are possible, but relaxing `atol` leads to less stringent energy conservation.

We can also check the evolution of the total particle momentum, analogous to [1, Fig. 3]. Computation of $P_0 = \sum_i m_i v_{beam,x}$ and $P_x(t) = \sum_i m_i v_{i,x}$ is straight forward. Setting up an initial state with zero momentum is quite a bit harder. The particle number is so low, that generating N_p independent normally distributed numbers will NOT sum to zero (or close) to it. Hence the generation of pair particles with anti-correlated thermal velocities described above. Using that we get the results in Fig. 3 that compare favourably with the expected

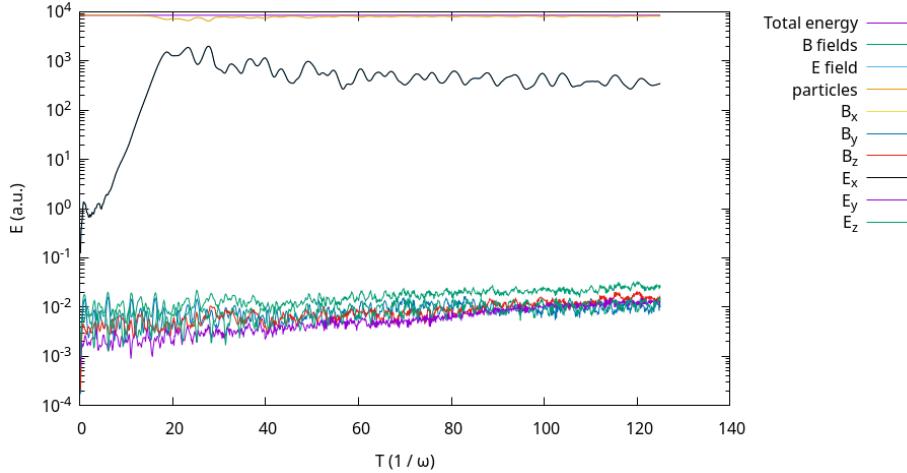


FIG. 4: Energy breakdown of the 1d3v simulation. All of the energy is initially in kinetic energy of the streaming electrons. Some of it is transferred to E_x by the instability. The other fields are transverse and only populated by fluctuation in the thermal velocities v_y, z of the particles. They contain two (initially) to five (at the peak of the instability) order of magnitude less energy. B_x stays uniformly and statically at $B_x = 0$ up to machine precision, as required by the divergence constraint.

performance of the method.

Computation of the effective numerical collision frequency $\nu(t) = \frac{dP_x(t)}{dt} \frac{1}{P_0}$ is also straightforward. Taking the RMS value of $\nu(t)$ over the entire duration of the simulation yields $\langle \nu \rangle = 1.2 \cdot 10^{-3} \omega_{pe}$. Restricting the time to $20\omega_{pe}^{-1} < t < 125\omega_{pe}^{-1}$ increases the value slightly to $\langle \nu \rangle = 1.3 \cdot 10^{-3} \omega_{pe}$.

More careful particle initialization by creating pairs of particles with opposite v_x to ensure $P_x(0) = 0$ would reduce the oscillations of momentum between particles and field and bring down the numerical collisionality.

The last result that we consider is the breakdown of energy into components in the 1d3v run, as shown in 4. Thermal fluctuations drive transverse fields that quickly reach a relatively steady level. These fields will be analyzed in more detail in II D.

B. Weibel Instability

This is a test of the transverse two-stream instability, which is closely related to the Weibel instability driven by temperature anisotropy.

Pick a reference density n_0 for normalization. Compute the plasma frequency $\omega_{pe} = \sqrt{4\pi n_o e^2/m_e}$ and skin depth $d_e = c/\omega_{pe}$. The simulation domain is only resolved in one spatial dimension and is $L_x = 2\pi d_e$ long and resolved by $N_x = 64$ cells. This implies $\Delta x = L_x/N_x = \pi/32d_e \approx .098d_e$, or in terms of Debye length $\Delta x \approx 9.8\lambda_D$. This resolution is borderline, but feasible with regular PiC codes. We use a timestep of $\Delta t = 1/8\omega_{pe}^{-1}$ and follow the system for $T = 125\omega_{pe}^{-1}$, which implies we perform $N_t = 1000$ time steps. The CFL condition number is given by $\mathcal{C} = c\Delta t/\Delta x = 1/8c\omega_{pe}^{-1}/(\pi/32c\omega_{pe}^{-1}) = 4/\pi \approx 1.27$. This is the same CFL condition number before, but this time in an electromagnetic instability. Explicit PiC codes would have a hard time with this. Two electron populations of equal density $n_0/2$ are initialized with beam speeds $v_{beam,y} = \pm 0.8c$ and thermal spread $v_{th,e} = 0.01c$. Velocity of particle i is given by

$$\begin{aligned} v_x[i] &= v_{th,e} * \mathcal{N}(0, 1) \\ v_y[i] &= -1^i * v_{beam,y} + v_{th,e} * \mathcal{N}(0, 1) \\ v_z[i] &= v_{th,e} * \mathcal{N}(0, 1) \end{aligned}$$

We use 154 particles per cell (following [1], even though 156 ppc would get us closer to 10^4 particles in total). The growth rate for this instability is given (in the fluid limit, see e.g. https://en.wikipedia.org/wiki/Weibel_instability) by

$$\frac{\gamma}{\omega_{pe}} = \frac{v_0}{c} \sqrt{\frac{k^2 d_e^2}{1 + k^2 d_e^2}}$$

For Fourier mode m in a domain of length $L_x = 2\pi d_e$ that has $k_m = \frac{2\pi m}{L_x} = \frac{2\pi m}{2\pi d_e} = \frac{m}{d_e}$ we expect a growth rate of

$$\frac{\gamma}{\omega_{pe}} = \frac{v_0}{c} \sqrt{\frac{m^2}{1 + m^2}}$$

Energy growth of the different m Fourier modes of B_z are shown in 5 and match expected growth rates rather well. But for some reason $m = 2$ is the most unstable mode and even seeding $m = 3$ doesn't help.

However, conservation of total energy is great as displayed in Fig. 6.

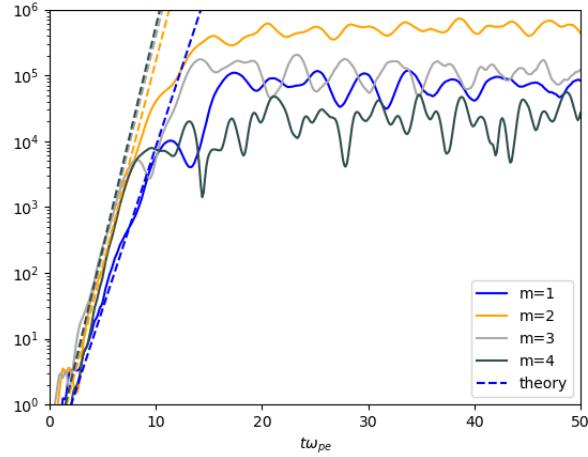


FIG. 5: Compare with [1, figure 4]. For us $m = 2$ dominates the dynamics, not $m = 3$.

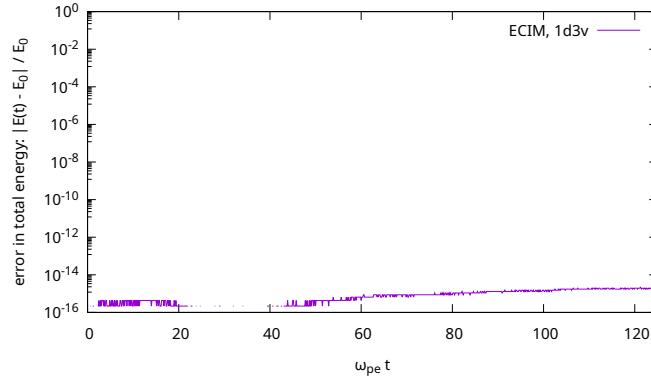


FIG. 6: Energy is conserved up to machine precision.

C. Finite-Grid-Instability

This is repeating the tests in [1, Section 5]. The paper states the domain length as $L = 2\pi \Xi$ which I take to mean $L_x = 2\pi \Xi d_e$. This domain length is devided into $N_x = 64$ cells. The cell size is therefore $\Delta x = \frac{L_x}{N_x} = \frac{\pi \Xi}{32} d_e$. Given the thermal speed $v_{th} = 0.01c$ we see that the Debye length is $\lambda_D = \frac{v_t h}{\omega_{pe}} = \frac{1}{100} \frac{c}{\omega_{pe}} = \frac{1}{100} d_e$. So if we calculate the cell sizes as

a multiple of the Debye length we find

$$\frac{\Delta x}{\lambda_D} = \frac{L_x \omega_{pe}}{N_x v_{th}} \quad (1)$$

$$= \frac{\omega_{pe} 2\pi \Xi d_e}{64 v_{th}} \quad (2)$$

$$= \frac{200\pi \Xi}{64} \quad (3)$$

$$= \frac{25\pi \Xi}{8} \quad (4)$$

$$\approx 9.81\Xi \quad (5)$$

Note that [1] has an extra factor L in Eq. (2), probably a copy-and-paste mistake. Additionally d_e is omitted. The paper then states that values of $\Xi \in [1, 10^{15}]$ are considered, making $\Delta x/\lambda_D \in [10, 10^{16}]$ which seems close enough to what we get.

For the timestep the paper states $\omega_{pe}\Delta t = .125\Xi$ which obviously implies $\Delta t = \frac{1}{8}\Xi\omega_{pe}^{-1}$.

If I use that to compute the CFL number

$$\mathcal{C} = \frac{c \Delta t}{\Delta x} = \frac{c \frac{1}{8}\Xi\omega_{pe}^{-1}}{\frac{\pi\Xi}{32}d_e} \quad (6)$$

$$= \frac{4}{\pi} \quad (7)$$

$$\approx 1.27 \quad (8)$$

I get a value that is much larger than the quoted “so that the CFL number is constant in all runs and equal to 0.0127”. So potentially he is defining the CFL number with respect to the thermal speed (if we treat it as an electrostatic problem, there is no lightwaves). That would get us

$$\mathcal{C}' = \frac{v_{th} \Delta t}{\Delta x} = \frac{\frac{1}{100}c \frac{1}{8}\Xi\omega_{pe}^{-1}}{\frac{\pi\Xi}{32}d_e} \quad (9)$$

$$= \frac{4}{100\pi} \quad (10)$$

$$\approx 0.0127 \quad (11)$$

which would match the quoted text, but the paper is not clear in the definition of the CFL number and there is going to be a few percent of particles that go faster than v_{th} .

The duration of the simulation runs is not reported, so I am picking the number of time steps to be $N_t = 1000$, which makes $T = 125\Xi\omega_{pe}^{-1}$.

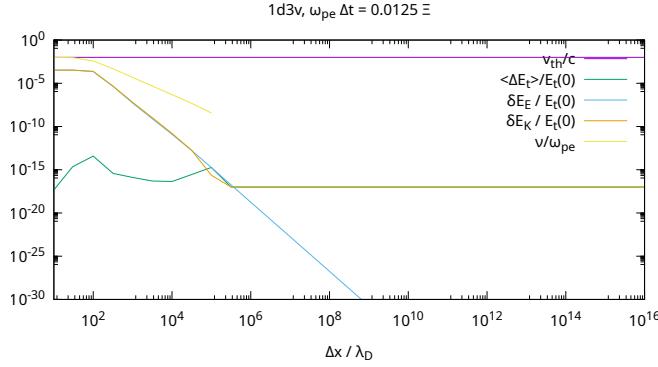


FIG. 7: Performance for simulations using $\omega_{pe}\Delta t = 0.0125$ and various cell sizes $\Delta x/\lambda_D$.

D. Wave Modes

These tests follow the setup described in [2], with one exception: The resolved direction is x instead of z .

1. Electromagnetic Mode

First we test the dispersion relation of (relatively) high-frequency modes in an unmagnetized plasma. The first one is the electromagnetic mode, analogous to [2, Figure 3]. This is shown in Fig. 8. We see that at large k the numerical dispersion is even large than in an explicit code with a regular Yee solver. This is not unexpected for an algorithm that colocated E fields at vertices and B fields at cell centers. But high k is not the focus of the algorithm. This poor numerical dispersion would likely lead to problems with relativistic particles through the production of NCI, but the code is currently non-relativistic anyway.

If we zoom in to lower k and ω (as shown in Fig. 9) we can identify the plasma frequency ω_{pe} at the correct value. This indicates that the normalization of current deposition, particle pusher and Maxwell solver is consistent. This figure is analogous to [2, Figure 2]

Electrostatic modes such as the Langmuir mode show the same plasma frequency and show an evolution with k that shows that the particle thermal speed is correct in the code. This can be seen in Fig. 10 which is analogous to [2, Figure 8].

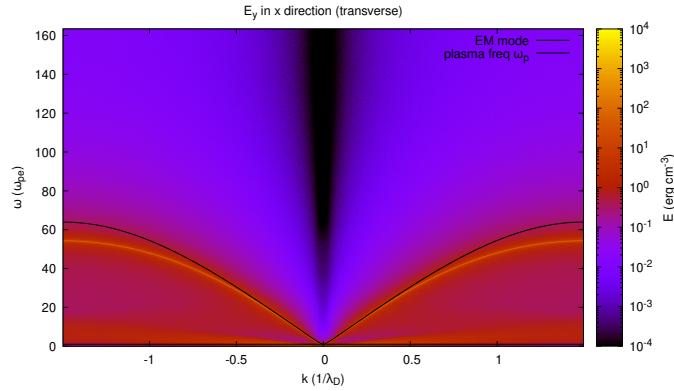


FIG. 8: Electromagnetic mode at large k and ω .

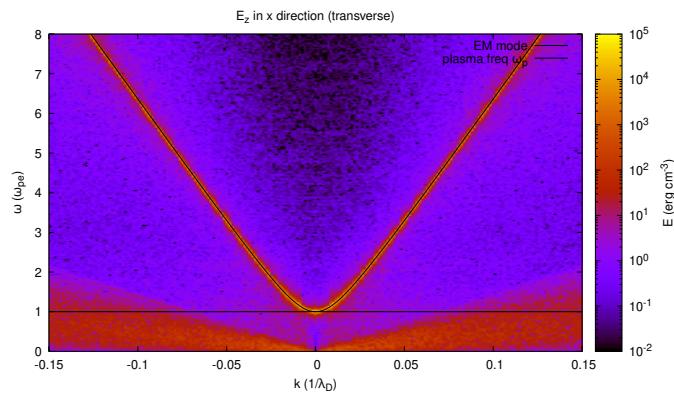


FIG. 9: Electromagnetic mode at smaller k and ω .

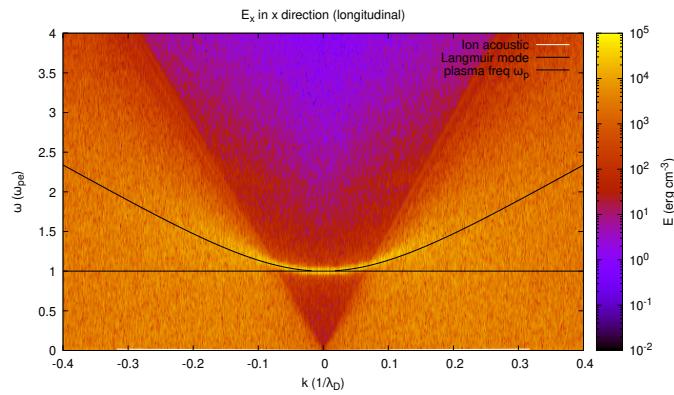


FIG. 10: Electrostatic mode at smaller k and ω .

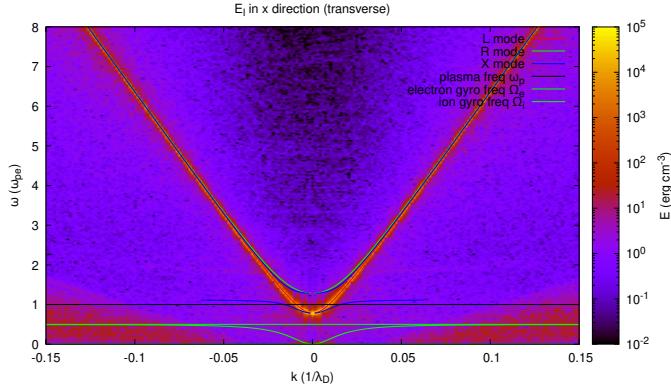


FIG. 11: Left-hand circular polarized electromagnetic modes.

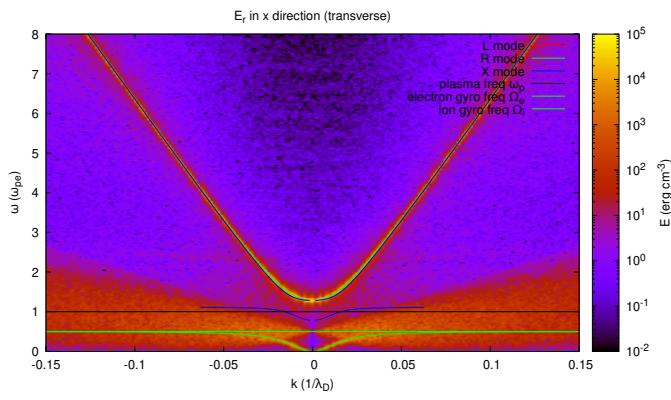


FIG. 12: Right-hand circular polarized electromagnetic modes.

2. High-frequency L- and R- mode

Adding a magnetic field along x such that $\omega_{pe}/\Omega_{ce} = 2$ the electromagnetic mode splits into a left and right handed mode. This is shown in Fig. 11, the equivalent of [2, Figure 5] and Fig. 12, the equivalent of [2, Figure 6]. This test is important because it shows that the signs in α is correct and we are not accidentally using the transpose α^T .

3. Extraordinary mode

Rotating the magnetic field to point along B_y allows us to benchmark the extraordinary (X) mode. Fig. 13 is the equivalent of [2, Figure 7].

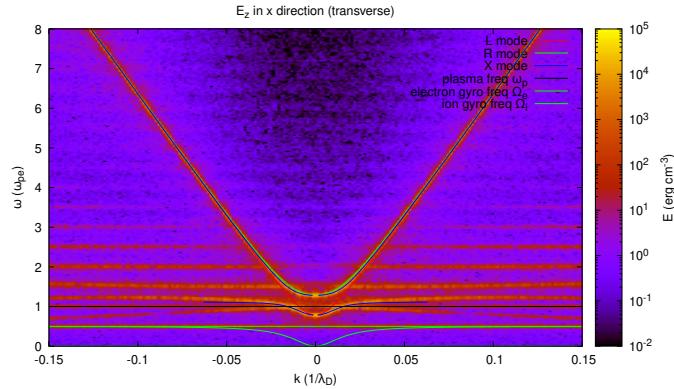


FIG. 13: Extraordinary electromagnetic mode at smaller k and ω .

4. Electron Bernstein modes

Analyzing the longitudinal field in the same test clearly shows the electron Bernstein modes. These are shown in Fig. 14 is the equivalent of [2, Figure 11].

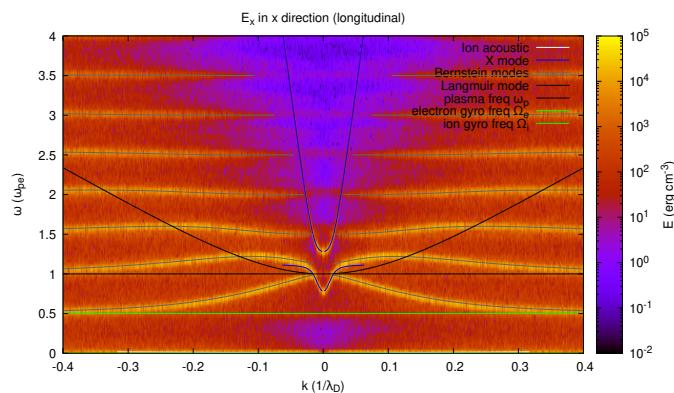


FIG. 14: Electron Bernstein modes.

5. Low-frequency R mode

In this test we start testing low-frequency modes that are expensive to simulate in explicit particle-in-cell codes. The domain length here is $L_x = 8192\text{cm} = 273d_e$, but using only $N_x = 512$ cells instead of 8192 cells as in the reference paper. The simulation is run for $T = 966\omega_{pe}^{-1}$ but using only $N_t = 2560$. This makes this test actually one of the cheaper steps in the wave modes validation test.

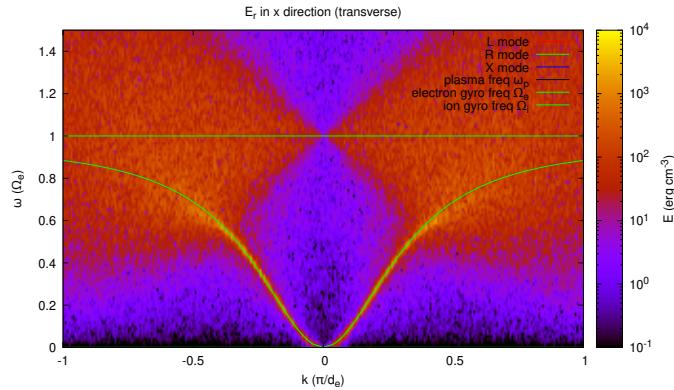


FIG. 15: Right-hand circular polarized electromagnetic modes at small k and ω . This in particular includes Whistler waves.

6. Low-frequency L mode

This test is using a larger domain of $L_x = 546 d_e$ and using $N_x = 3415$ (much less than the 16384 cells in the reference paper). The simulation covers $T = 4000\omega_{pe}^{-1}$ but needs only $N_t = 3534$. To have a nice low-frequency mode that is left hand circular, this test requires the presence of a proton species. We use a reduced mass ratio $m_p/m_e = 18.36$ here, following the reference paper, but with this code much larger mass ratios would be feasible too.

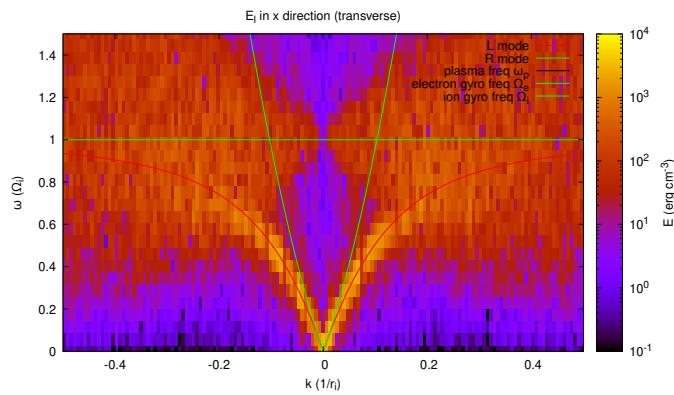


FIG. 16: Left-hand circular polarized electromagnetic modes at small k and ω .

-
- [1] G. Lapenta, Exactly energy conserving semi-implicit particle in cell formulation, *Journal of Computational Physics* **334**, 349 (2017).
 - [2] P. Kilian, P. A. Muñoz, C. Schreiner, and F. Spanier, Plasma waves as a benchmark problem, *Journal of Plasma Physics* **83**, 707830101 (2017).