

# Esirkepov current deposit summary

(Dated: June 18, 2018)

## I. INTRODUCTION

Numerical noise in particle in cell (PIC) simulations can limit their effectiveness in modeling plasma phenomena. A number of strategies have been proposed to improve control of numerical heating in PIC, for example filtering the charge and current densities.<sup>1,2</sup> Recently, the use of higher-order particle form-factors has been proposed as a different method to improve energy conservation in PIC simulations, and its effectiveness has been demonstrated with the 1D electrostatic code SHARP.<sup>3</sup> The utility of higher-order shape functions in a 2D and 3D electromagnetic code remains under-explored.

In this document, we investigate the use of higher-order shape functions in improving energy conservation, via the Esirkepov density decomposition algorithm for current deposition. The outline is as follows: In Section II, we review the Esirkepov scheme. In Section III, we present several tests of our code, and check that it is a correct implementation of the Esirkepov algorithm. In Section IV, we compare the performance of the Esirkepov density decomposition with the zigzag scheme, for the current deposition and particle pusher. We conclude in Section V.

## II. ESIRKEPOV SCHEME

In this section, we review briefly the Esirkepov density decomposition scheme for current deposition.<sup>4</sup>

Consider the motion of a quasi-particle on a Cartesian grid from initial position  $(x, y, z)$  to  $(x + \Delta x, y + \Delta y, z + \Delta z)$ . The particle motion produces current that is deposited on the grid, as determined by the continuity equation,

$$\frac{d\rho}{dt} + \nabla \cdot \mathbf{j} = 0, \quad (1)$$

or written as a finite difference equation,

$$\frac{\rho_{n+1}^{i,j,k} - \rho_n^{i,j,k}}{\Delta t} + \frac{j_x^{i+1,j,k} - j_x^{i,j,k}}{\Delta x} + \frac{j_y^{i,j+1,k} - j_y^{i,j,k}}{\Delta y} + \frac{j_z^{i,j,k+1} - j_z^{i,j,k+1}}{\Delta z} = 0, \quad (2)$$

where the subscript  $n$  denotes the timestep, superscripts  $i, j, k$  refer to grid coordinates in the directions  $x, y$ , and  $z$ ;  $\Delta_x, \Delta_y$ , and  $\Delta_z$  are grid spacings in the directions  $x, y$ , and  $z$ , and  $\Delta_t$  is

the timestep. Since the particle motion is independent in the  $x$ ,  $y$ , and  $z$  directions, Eq. 2 may be recast as three separate equations governing current deposition along each direction:

$$j_x^{i+1,j,k} = j_x^{i,j,k} - q \frac{\Delta_x}{\Delta_t} W_x^{i,j,k} \quad (3)$$

$$j_y^{i,j+1,k} = j_y^{i,j,k} - q \frac{\Delta_y}{\Delta_t} W_y^{i,j,k} \quad (4)$$

$$j_z^{i,j,k+1} = j_z^{i,j,k} - q \frac{\Delta_z}{\Delta_t} W_z^{i,j,k}, \quad (5)$$

where  $q$  is the charge of the quasi-particle. Eqs. 3, 4, and 5 define a vector of weights  $\mathbf{W}^{i,j,k} = [W_x^{i,j,k}, W_y^{i,j,k}, W_z^{i,j,k}]$ , which are the densities  $(\rho_{x,n+1}^{i,j,k} - \rho_{x,n}^{i,j,k})/q$ ,  $(\rho_{y,n+1}^{i,j,k} - \rho_{y,n}^{i,j,k})/q$ , and  $(\rho_{z,n+1}^{i,j,k} - \rho_{z,n}^{i,j,k})/q$  associated to current deposition along each direction. If the total charge density  $\rho^{i,j,k}$  is expressed in terms of an arbitrary particle form-factor (or density)  $S^{i,j,k}$ ,

$$\rho^{i,j,k}(x, y, z) = q S^{i,j,k}(x, y, z), \quad (6)$$

then by the continuity equation 2, the densities  $W_x^{i,j,k}, W_y^{i,j,k}, W_z^{i,j,k}$  satisfy

$$W_x^{i,j,k} + W_y^{i,j,k} + W_z^{i,j,k} = S^{i,j,k}(x + \Delta x, y + \Delta y, z + \Delta z) - S^{i,j,k}(x, y, z). \quad (7)$$

To employ the independence of Eqs. 3, 4, and 5, the individual components of  $\mathbf{W}^{i,j,k}$  (rather than the sum, as in Eq. 7) must be expressed as differences of form-factors  $S^{i,j,k}$ . The motion of the quasi-particle can generate 8 possible form-factors,

$$\begin{aligned} &S(x + \Delta x, y + \Delta y, z + \Delta z), \\ &S(x, y + \Delta y, z + \Delta z), S(x + \Delta x, y, z + \Delta z), S(x + \Delta x, y + \Delta y, z), \\ &S(x, y, z + \Delta z), S(x, y + \Delta y, z), S(x + \Delta x, y, z), \\ &S(x, y, z), \end{aligned} \quad (8)$$

so in general, the components of  $\mathbf{W}$  may be written as linear sums of these factors.  $\mathbf{W}$  may in fact be expressed as a unique linear combination of the functions above, if the following properties of  $\mathbf{W}$  are enforced:<sup>4</sup>

1. The sum of weights  $W_x, W_y, W_z$  is equal to the difference of form-factors (Eq. 7).
2. If the shift in any direction is zero, the corresponding weight is zero, i.e.

$$\Delta x = 0 \iff W_x = 0 \quad (9)$$

$$\Delta y = 0 \iff W_y = 0 \quad (10)$$

$$\Delta z = 0 \iff W_z = 0. \quad (11)$$

3. Symmetry with respect to permutation of coordinate:

$$S(x, y, z) = S(y, x, z) \text{ and } \Delta x = \Delta y \iff W_x = W_y \quad (12)$$

$$S(x, y, z) = S(x, z, y) \text{ and } \Delta y = \Delta z \iff W_y = W_z \quad (13)$$

$$S(x, y, z) = S(z, y, x) \text{ and } \Delta x = \Delta z \iff W_x = W_z. \quad (14)$$

Esirkepov was able to formulate and solve this problem, i.e. the expression of  $W_x, W_y, W_z$  subject to the above constraints. The unique solution is:<sup>4</sup>

$$\begin{aligned} W_x &= \frac{1}{3}S(x + \Delta x, y + \Delta y, z + \Delta z) - \frac{1}{3}S(x, y + \Delta y, z + \Delta z) \\ &\quad + \frac{1}{6}S(x + \Delta x, y, z + \Delta z) - \frac{1}{6}S(x, y, z + \Delta z) \\ &\quad + \frac{1}{6}S(x + \Delta x, y + \Delta y, z) - \frac{1}{6}S(x, y + \Delta y, z) \end{aligned} \quad (15)$$

$$\begin{aligned} W_y &= \frac{1}{3}S(x + \Delta x, y + \Delta y, z + \Delta z) - \frac{1}{3}S(x + \Delta x, y, z + \Delta z) \\ &\quad + \frac{1}{6}S(x, y + \Delta y, z + \Delta z) - \frac{1}{6}S(x, y, z + \Delta z) \\ &\quad + \frac{1}{6}S(x + \Delta x, y + \Delta y, z) - \frac{1}{6}S(x + \Delta x, y, z) \\ &\quad + \frac{1}{3}S(x + \Delta x, y, z) - \frac{1}{3}S(x, y, z) \end{aligned} \quad (16)$$

$$\begin{aligned} W_z &= \frac{1}{3}S(x + \Delta x, y + \Delta y, z + \Delta z) - \frac{1}{3}S(x + \Delta x, y + \Delta y, z) \\ &\quad + \frac{1}{6}S(x, y + \Delta y, z + \Delta z) - \frac{1}{6}S(x, y + \Delta y, z) \\ &\quad + \frac{1}{6}S(x + \Delta x, y, z + \Delta z) - \frac{1}{6}S(x + \Delta x, y, z) \\ &\quad + \frac{1}{3}S(x, y, z + \Delta z) - \frac{1}{3}S(x, y, z). \end{aligned} \quad (17)$$

In the two dimensional problem, Eqs. 15, 16, and 17 admit a simplified form,<sup>4</sup>

$$\begin{aligned} W_x^{2D} &= \frac{1}{2}S^{2D}(x + \Delta x, y + \Delta y) - \frac{1}{2}S^{2D}(x, y + \Delta y) \\ &\quad + \frac{1}{2}S^{2D}(x + \Delta x, y) - \frac{1}{2}S^{2D}(x, y) \end{aligned} \quad (18)$$

$$\begin{aligned} W_y^{2D} &= \frac{1}{2}S^{2D}(x + \Delta x, y + \Delta y) - \frac{1}{2}S^{2D}(x + \Delta x, y, z + \Delta z) \\ &\quad + \frac{1}{2}S^{2D}(x, y + \Delta y) - \frac{1}{2}S^{2D}(x, y) \end{aligned} \quad (19)$$

$$\begin{aligned} W_z^{2D} &= \frac{1}{3}S^{2D}(x + \Delta x, y + \Delta y) + \frac{1}{6}S^{2D}(x, y + \Delta y) \\ &\quad + \frac{1}{6}S^{2D}(x + \Delta x, y) + \frac{1}{3}S^{2D}(x, y). \end{aligned} \quad (20)$$

Note that in the 2D case, translational invariance in the  $z$  direction is assumed, so that  $j_z^{i,j,k+1} - j_z^{i,j,k}$  vanishes. In this case, Eqs. 3, 4, and 5 reduce to

$$j_x^{i+1,j} = j_x^{i,j} - q \frac{\Delta_x}{\Delta_t} W_x^{(2D)i,j} \quad (21)$$

$$j_y^{i,j+1} = j_y^{i,j} - q \frac{\Delta_y}{\Delta_t} W_y^{(2D)i,j} \quad (22)$$

$$j_z^{i,j} = q v_z W_z^{(2D)i,j}, \quad (23)$$

where  $v_z$  is the velocity of the quasi-particle in the  $z$  direction. In the 2D case, current deposited in the  $z$  direction is computed as a weighted average of the charge flux, over the initial position  $(x, y)$ , two intermediate positions  $(x + \Delta x, y)$  and  $(x, y + \Delta y)$ , and the final position  $(x + \Delta x, y + \Delta y)$ ; see Eq. 20. Additional details of the derivation are provided in Appendix A.

One advantage of the Esirkepov scheme is that it allows for the straightforward implementation of different particle form-factors, which enter through the definition of weights, Eqs. 15, 16, and 17 (or in 2D, Eqs. 18, 19, and 20). In particular, higher-order shape functions may be implemented with ease, which is suggested as a method of controlling numerical heating in PIC simulations.<sup>3</sup>

In the remaining sections, we investigate the effect of higher-order shape functions in a few test problems, and compare to another commonly used charge-conserving method of current deposition, the zigzag scheme.<sup>5</sup> To study the higher-order form-factors and compare with the zigzag algorithm, we use an implementation of the Esirkepov density decomposition in the electromagnetic PIC code TRISTAN-MP.<sup>1,2</sup>

We focus on shape functions of order 1, 2, and 3, which are defined as follows.<sup>12</sup> Starting with the zeroth-order shape function,

$$S^0(x) = \begin{cases} 1, & \text{if } |x| < \frac{1}{2} \\ 0, & \text{otherwise} \end{cases} \quad (24)$$

shape functions of higher order are defined via recursion:

$$S^{n+1}(x) = \int S^0(x - y) S^n(y) dy. \quad (25)$$

The first few shape functions have the explicit forms

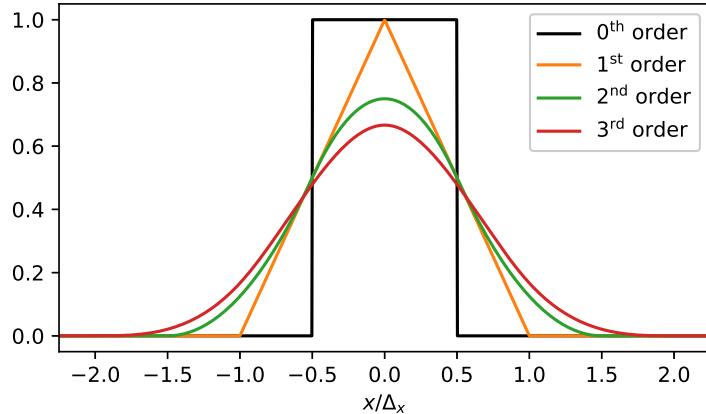
$$S^1(x) = \begin{cases} x + 1, & \text{if } -1 < x \leq 0 \\ 1 - x, & \text{if } 0 < x < 1 \\ 0, & \text{otherwise} \end{cases} \quad (26)$$

$$S^2(x) = \begin{cases} \frac{1}{8}(2x+3)^2, & \text{if } -\frac{3}{2} < x \leq -\frac{1}{2} \\ \frac{3}{4} - x^2, & \text{if } -\frac{1}{2} < x \leq \frac{1}{2} \\ \frac{1}{8}(3-2x)^2, & \text{if } \frac{1}{2} < x < \frac{3}{2} \\ 0, & \text{otherwise} \end{cases} \quad (27)$$

$$S^3(x) = \begin{cases} \frac{1}{6}(x+2)^3, & \text{if } -2 < x \leq -1 \\ \frac{1}{6}(4-3x^2(x+2)), & \text{if } -1 < x \leq 0 \\ \frac{1}{6}(3(x-2)x^2 + 4), & \text{if } 0 < x < 1 \\ -\frac{1}{6}(x-2)^3, & \text{if } 1 \leq x < 2 \\ 0, & \text{otherwise} \end{cases} \quad (28)$$

The shape functions Eqs. 24, 26, 27, and 28 are plotted in Fig. 1.

FIG. 1. Particle form-factors of order 0, 1, 2, and 3.



### III. TESTS

In this section, we present a few test cases (heating and energy non-conservation in a single-temperature plasma; Filamentation instability; numerical Cherenkov instability) to demonstrate the effect of higher-order shape functions in an electromagnetic PIC code. In each test problem, we compare the zigzag current deposit scheme, as well as the Esirkepov density decomposition scheme for 1<sup>st</sup>, 2<sup>nd</sup>, and 3<sup>rd</sup> order shape functions. We note that the zigzag scheme of Umeda et. al, which assumes a 1<sup>st</sup> order particle shape function, still differs from the 1<sup>st</sup> order density decomposition scheme in the assumed motion of the particle.<sup>5</sup> In the zigzag method, particles

may travel along trajectories that resemble a zigzag pattern, whereas in the Esirkepov density decomposition, particle trajectories are treated as straight lines.

For all of the test problems presented here, the initial magnetic field is zero.

### A. Numerical heating in a uniform plasma

Here, we investigate the effect of different current deposit schemes on numerical heating in a uniform plasma initialized at rest. The ion and electron masses are equal,  $m = m_i = m_e$ , and the plasma is single-temperature,  $T_i = T_e = T$ . Each simulation is initialized with a different choice of dimensionless temperature,

$$\theta = \frac{k_B T}{mc^2}. \quad (29)$$

To avoid severe heating associated with the Finite Grid instability, we study plasmas with temperature less than the numerical Debye temperature,  $\theta < \theta_D$ ;<sup>6,7</sup> this is the temperature at which the Debye length

$$\lambda_D = \sqrt{\frac{k_B T}{4\pi n_0 e^2}} \quad (30)$$

is equal to the grid spacing,  $\Delta_x = \Delta_y = \Delta_z = \Delta$ ; here,  $n_0$  is number density and  $e$  is electric charge  $e$ . With  $\lambda_D = \Delta$ , the dimensionless Debye temperature can be written

$$\theta_D = \left( \frac{\Delta}{c/\omega_p} \right)^2, \quad (31)$$

where  $c/\omega_p$  is the plasma skin depth, and the plasma frequency is

$$\omega_p = \sqrt{\frac{4\pi n_0 e^2}{m}}. \quad (32)$$

The violent numerical heating associated with the Finite Grid Instability is strongest at  $\theta = \theta_D$ , however its effect is mitigated at temperatures above or below  $\theta_D$ .<sup>6</sup> In the simulations presented here, we use 10 cells per electron skin depth, then the numerical Debye temperature is  $\theta_D = 10^{-2}$ . We study numerical heating in plasmas with initial dimensionless temperatures  $\theta_0 = 10^{-5}, 10^{-4}$ , and  $10^{-3}$ . Plasma with temperature  $\theta$  is initialized according to a Maxwell-Jüttner distribution,

$$f_{MJ}(\gamma, \theta) \propto \gamma (\gamma^2 - 1)^{\frac{d}{2}-1} \exp(-\gamma/\theta), \quad (33)$$

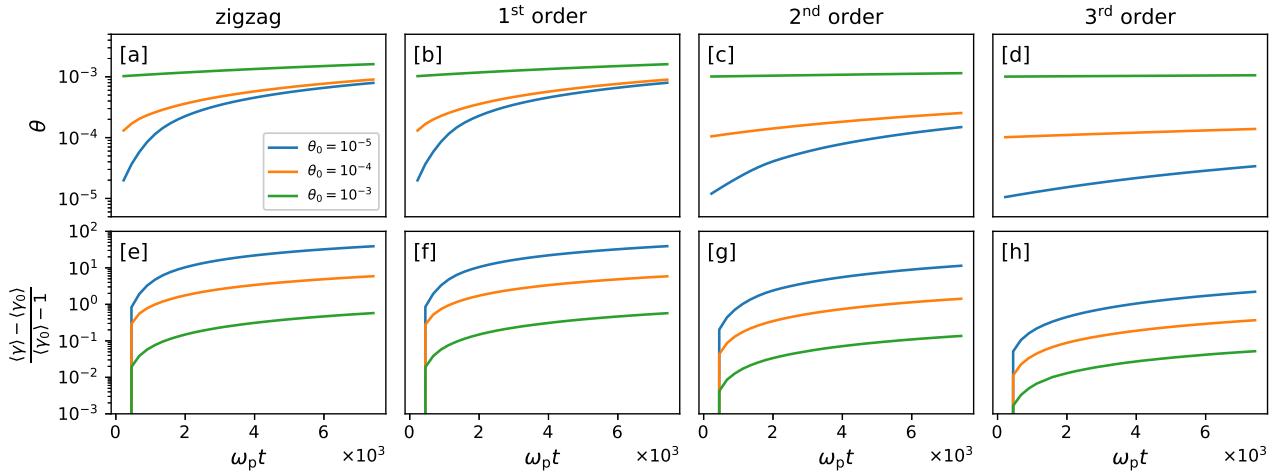
where  $\gamma$  is the particle Lorentz factor and  $d$  is the number of dimensions; for the simulations in this section,  $d = 3$ . The plasma is initialized in a box of dimension  $L_x = 15 c/\omega_p, L_y = 30 c/\omega_p, L_z =$

$15 c/\omega_p$ . The simulation domain is periodic in  $x, y$ , and  $z$ . The number of particles per cell is  $N_{\text{ppc}} = 64$ .

In Fig. 2, we demonstrate the effect different current deposit schemes on numerical heating in a uniform plasma. Columns 1, 2, 3, and 4 correspond to zigzag, 1<sup>st</sup> order, 2<sup>nd</sup> order, and 3<sup>rd</sup> order density decomposition, respectively. Panels (a) - (d) show the evolution of dimensionless temperature  $\theta$  as a function of dimensionless time,  $t\omega_p$ . Panels (e) - (h) show the fractional error in mean particle Lorentz factor relative to the initial mean value  $\langle \gamma_0 \rangle - 1$ , i.e.  $(\langle \gamma \rangle - \langle \gamma_0 \rangle)/(\langle \gamma_0 \rangle - 1)$ ; here,  $\gamma_0 \equiv 1 + \theta_0/(\Gamma_{\text{ad}}(\theta_0) - 1)$ , where the adiabatic index  $\Gamma_{\text{ad}}$  is a function of dimensionless temperature  $\theta$ .

In an ideal simulation, the fractional error should remain zero, but due to numerical heating, the fractional error increases. The zigzag scheme and 1<sup>st</sup> order density decomposition have similar values for the fractional error over the runtime of the simulations (which is roughly  $t\omega_p = 8 \times 10^3$ ). By using a higher-order particle shape, as in panels (g) and (h), the fractional error induced by numerical heating is decreased; for example, with a 3<sup>rd</sup> order shape function, the fractional error is reduced by about 1.5 orders of magnitude, relative to zigzag or 1<sup>st</sup> order density decomposition.

FIG. 2. ((a) - (d)) Evolution of dimensionless temperature and ((e) - (h)) fractional error in a uniform plasma with initial dimensionless temperatures  $\theta_0 = 10^{-5}, 10^{-4}$ , and  $10^{-3}$ , for zigzag, 1<sup>st</sup>, 2<sup>nd</sup>, and 3<sup>rd</sup> order Esirkepov (columns 1 - 4, respectively). The simulations presented here are in full 3D.



Numerical heating in the 2D case is similar to that of the 3D case discussed here; see Appendix B.

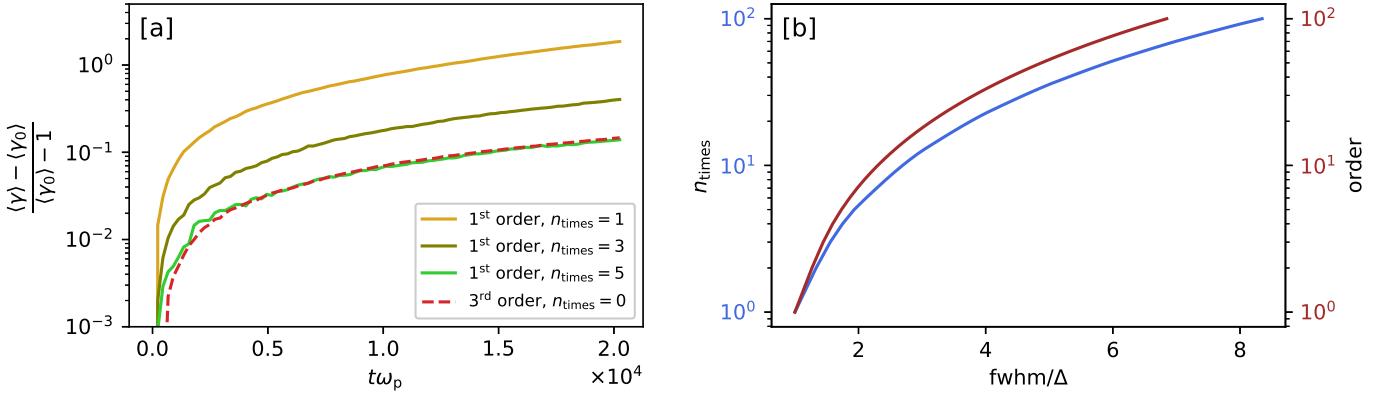
An alternate method that allows for improved control of numerical heating is the application

of a filter to smooth the charge and current densities.<sup>1,2</sup> One such implementation is to, at each timestep and in each direction, redistribute the value of the current density at a grid point into neighboring cells, according to prescribed weights; a commonly used choice of weights is 1-2-1. By repeated application of such a filter, the current density is smoothed, which improves energy conservation in the PIC simulation.

How does the effect of such a filter on numerical heating compare with that of a higher-order shape function? We can get a rough idea by repeated convolution of a 1-2-1 filter on a square-shaped quasi-particle of width  $\Delta$ , equal to the grid spacing; we consider applying the filter a total number of times  $n_{\text{times}}$ . After each application of the filter, the width of the particle at half maximum can be measured, yielding a correspondence between  $n_{\text{times}}$  and the particle full width at half maximum (fwhm). Alternatively, based on the definition of particle shape function Eqs. 24, 25, the fwhm can be computed as a function of the order of the particle shape. To demonstrate the rough correspondence between  $n_{\text{times}}$  and order of shape function, with respect to the effective fwhm of the quasi-particle, we show in Fig. 3 (b) a plot of  $n_{\text{times}}$  vs. fwhm and order vs. fwhm. The fwhm scales roughly as the square root of  $n_{\text{times}}$ ,  $\text{fwhm}/\Delta \propto n_{\text{times}}^{0.49}$ , and the scaling with order of the particle shape is slightly weaker,  $\text{fwhm}/\Delta \propto \text{order}^{0.47}$ . This comparison shows that repeated convolution of a 1-2-1 filter and higher-order shape functions both increase the effective fwhm of the quasi-particle, so both methods should result in improved control of numerical heating.

While intuitive, this comparison is a simplified picture of the two methods, and cannot capture the full details of a PIC simulation. To demonstrate a more robust comparison of the filter smoothing and higher-order shape function, we show in Fig. 3 (a) the time evolution of fractional error in a plasma with initial temperature  $\theta_0 = 10^{-4}$ , for simulations with 1<sup>st</sup> order shape function and  $n_{\text{times}} = 1, 3, 5$ , and a simulation with a 3<sup>rd</sup> order shape function, but  $n_{\text{times}} = 0$ . In this case, we find a rough equivalence, in terms of evolution of fractional error, for the simulation with a 1<sup>st</sup> order shape function,  $n_{\text{times}} = 5$ , and 3<sup>rd</sup> order shape function,  $n_{\text{times}} = 0$ . With respect to fractional error, the scalings shown in Fig. 3 (b) predict an equivalence between a 3<sup>rd</sup> order shape function, and  $n_{\text{times}} \approx 2$ , which differs from the empirically measured equivalence between 3<sup>rd</sup> order shape, and  $n_{\text{times}} \approx 5$ , shown in panel (a); the simplified picture which leads to the scalings  $\text{fwhm}/\Delta \propto n_{\text{times}}^{0.49}$  and  $\text{fwhm}/\Delta \propto \text{order}^{0.47}$  is probably insufficient to capture the full details of the PIC implementation.

FIG. 3. (a) For 2D simulations with  $\theta_0 = 10^{-4}$ , comparison of fractional error for 1<sup>st</sup> order form-factor and  $n_{\text{times}} = 1, 3, 5$  (solid lines) vs. 3<sup>rd</sup> order shape and  $n_{\text{times}} = 0$  (dashed); (b)  $n_{\text{times}}$  vs. effective fwhm (blue) and order of particle shape vs. effective fwhm (dark red).



## B. Filamentation instability

For the next test, we consider a counterstreaming beam. The beam is aligned with the  $x$  direction. We choose equal densities for the beam and background plasma, and consider two cases, one where the beam Lorentz factor  $\gamma_b$  is nonrelativistic ( $v_b/c = 0.5 \rightarrow \gamma_b \approx 1.15$ ), and a second with a marginally relativistic beam ( $v_b/c = 0.95 \rightarrow \gamma_b \approx 3.20$ ). Given the initial anisotropy of this setup, plasma instabilities will grow to drive the particle distribution function towards isotropy. Here, we consider the growth of the perpendicular mode, known as the Filamentation instability, which is the fastest growing mode for symmetric beams. It has a maximum growth rate<sup>8</sup>

$$\delta_{\max, \text{th}} = \frac{v_b}{c} \omega_p, \quad (34)$$

where  $\omega_p$  is the plasma frequency, with mass rescaled by  $\gamma_b$ ,  $\omega_p = \sqrt{4\pi n_0 e^2 / (\gamma_b m)}$ ,  $n_0$  is the total number density of positrons and electrons.

For the 3D simulations presented here, we use a periodic box whose dimensions are  $L_x = L_y = L_z = 12.6 c/\omega_p$ , with  $c/\omega_p = 10$  cells. In the nonrelativistic case, the box length is about equal to the wavelength of the unstable mode, however we have checked that results our agree with those obtained by use of a box that is three times as large in the perpendicular ( $z$ ) direction. The initial dimensionless temperature of the plasma is  $\theta_0 = 2 \times 10^{-5}$ , and we use  $N_{\text{ppc}} = 64$  particles per cell. For 2D simulations, we use the same choice of numerical parameters, except for the box size which is  $L_x = L_y = 12.8 c/\omega_p$ .

In Fig. 4, we show evolution of (2×)perpendicular magnetic energy  $\langle |B_z|^2 \rangle / (4\pi nmc^2)$ , where angle brackets denote a spatial average over the simulation domain, which is 3D. Panels (a) and (b) show the growth rates for the nonrelativistic and relativistic cases, respectively. The measured growth rate is presented for zigzag, as well as Esirkepov's scheme with 1<sup>st</sup>, 2<sup>nd</sup>, and 3<sup>rd</sup> order shape functions. The measured growth rates are compared with the analytical growth rate, Eq. 34. In the relativistic case, the different current deposit schemes show acceptable agreement with the predicted maximum growth rate of the unstable mode. In the nonrelativistic case, the density decomposition with 2<sup>nd</sup> and 3<sup>rd</sup> order shape functions show reasonable agreement with the analytical prediction, however, the zigzag and density decomposition with 1<sup>st</sup> order shape function do not show a close agreement to the prediction. In these cases, however, the agreement improves with increased number of particles per cell  $N_{ppc}$ , which is another way to control numerical heating, apart from the filtering and higher-order shape functions discussed in Section III A. For the zigzag scheme, we find improved agreement with the analytical prediction when the number of particles per cell is increased; we have tested with  $N_{ppc} = 512$ . The disagreement between the analytical prediction, as compared to zigzag and 1<sup>st</sup> order density decomposition, is likely linked to numerical heating at early times which distorts the measured growth rate, rather than an intrinsic limitation of 1<sup>st</sup> order shape functions in capturing the physics of the Filamentation instability.

In Figs. 5 and 6, we show spatial profiles of (2×)magnetic energy,  $B_z$ , and density normalized to initial density  $n_0$ , for 2D simulations with nonrelativistic and relativistic beams, respectively. The first, second, third, and fourth columns correspond to different schemes for current deposition: zigzag, 1<sup>st</sup>, 2<sup>nd</sup>, and 3<sup>rd</sup> order density decomposition. In each figure, the spatial profiles are shown close to the saturation of the instability; for the nonrelativistic case, this is around  $t\omega_p \approx 22.73$ , and for the relativistic case,  $t\omega_p \approx 16.88$ . In each case, nonrelativistic and relativistic, the four simulations start with the same initial conditions, so any differences in the spatial profiles is caused by the different numerical choices in each of the four current deposition schemes.

FIG. 4. Time evolution of perpendicular magnetic energy for (a) nonrelativistic and (b) relativistic counter-streaming beam, for zigzag, 1<sup>st</sup>, 2<sup>nd</sup>, and 3<sup>rd</sup> order Esirkepov. Solid lines show the measured growth rates, and dashed lines show the analytical prediction (see Eq. 34). The simulations here are 3D.

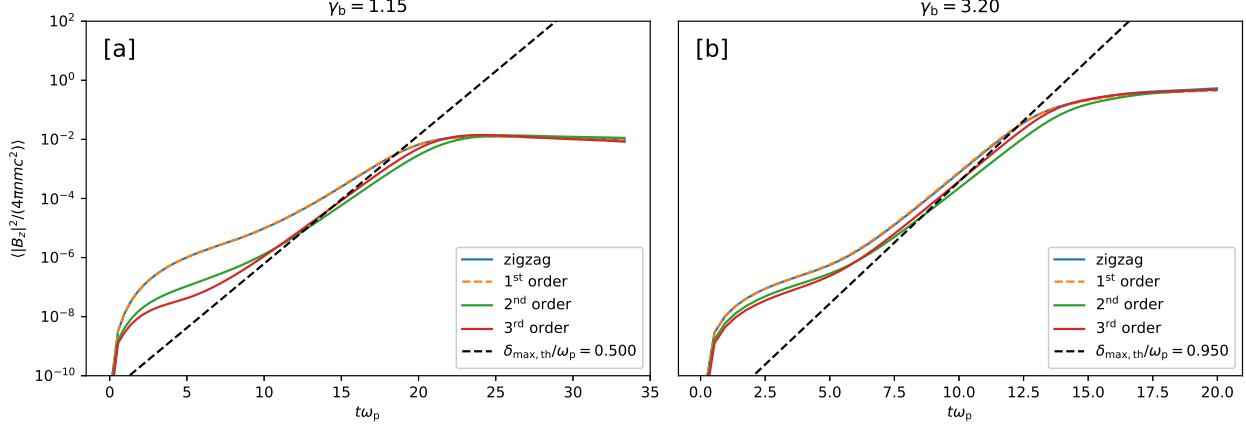


FIG. 5. For the nonrelativistic counterstreaming setup ( $\gamma_b \approx 1.15$ ): 2D spatial profiles of magnetic energy (first row),  $z$  component of magnetic field (second row), and density in units of initial density (third row), for 1<sup>st</sup>, 2<sup>nd</sup>, and 3<sup>rd</sup> order Esirkepov (columns 1 - 4). The spatial profiles are from 2D simulations at  $t\omega_p \approx 22.73$ , which is near saturation of the growth of the unstable mode. The spatial profiles shown here are from 2D simulations.

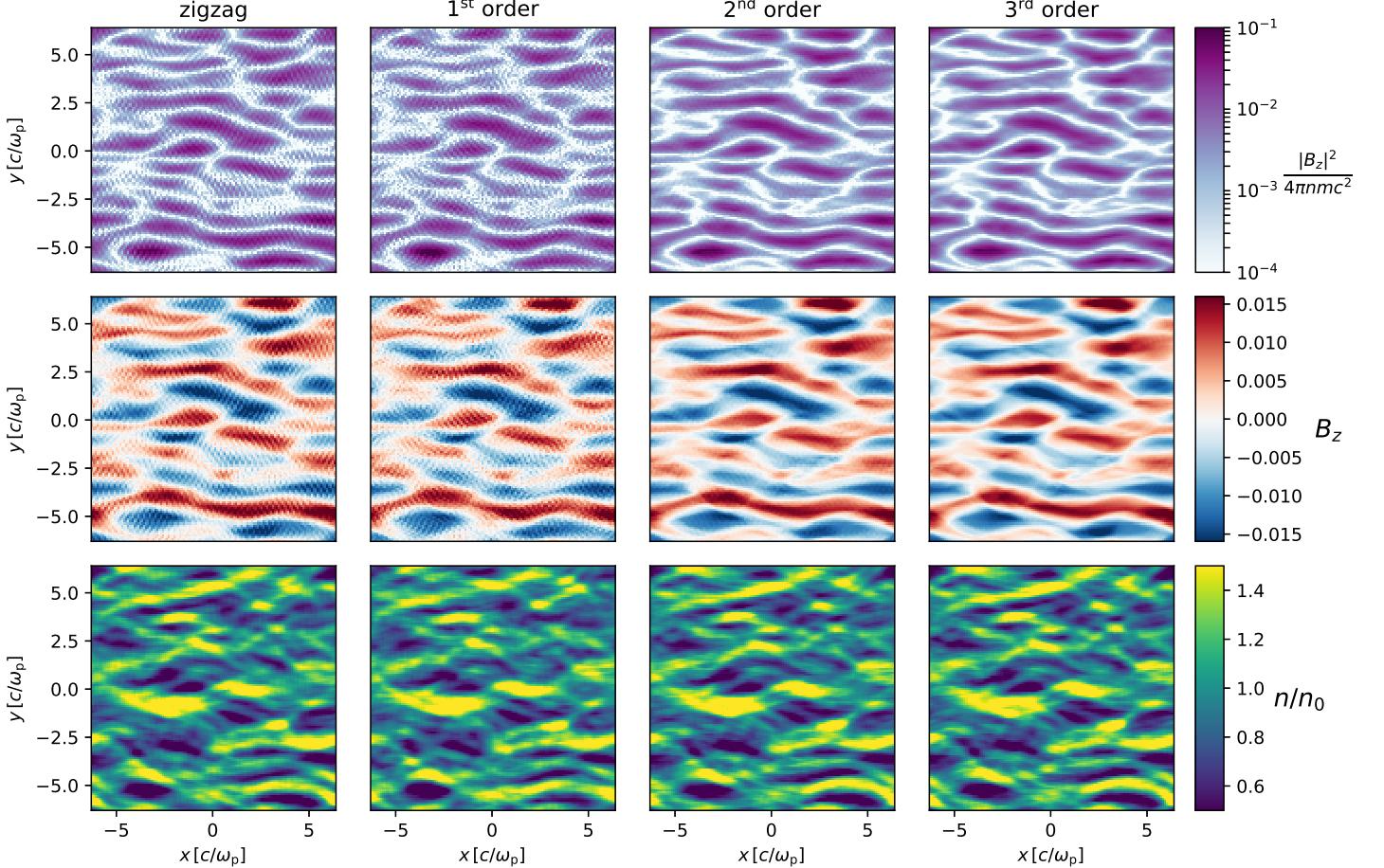
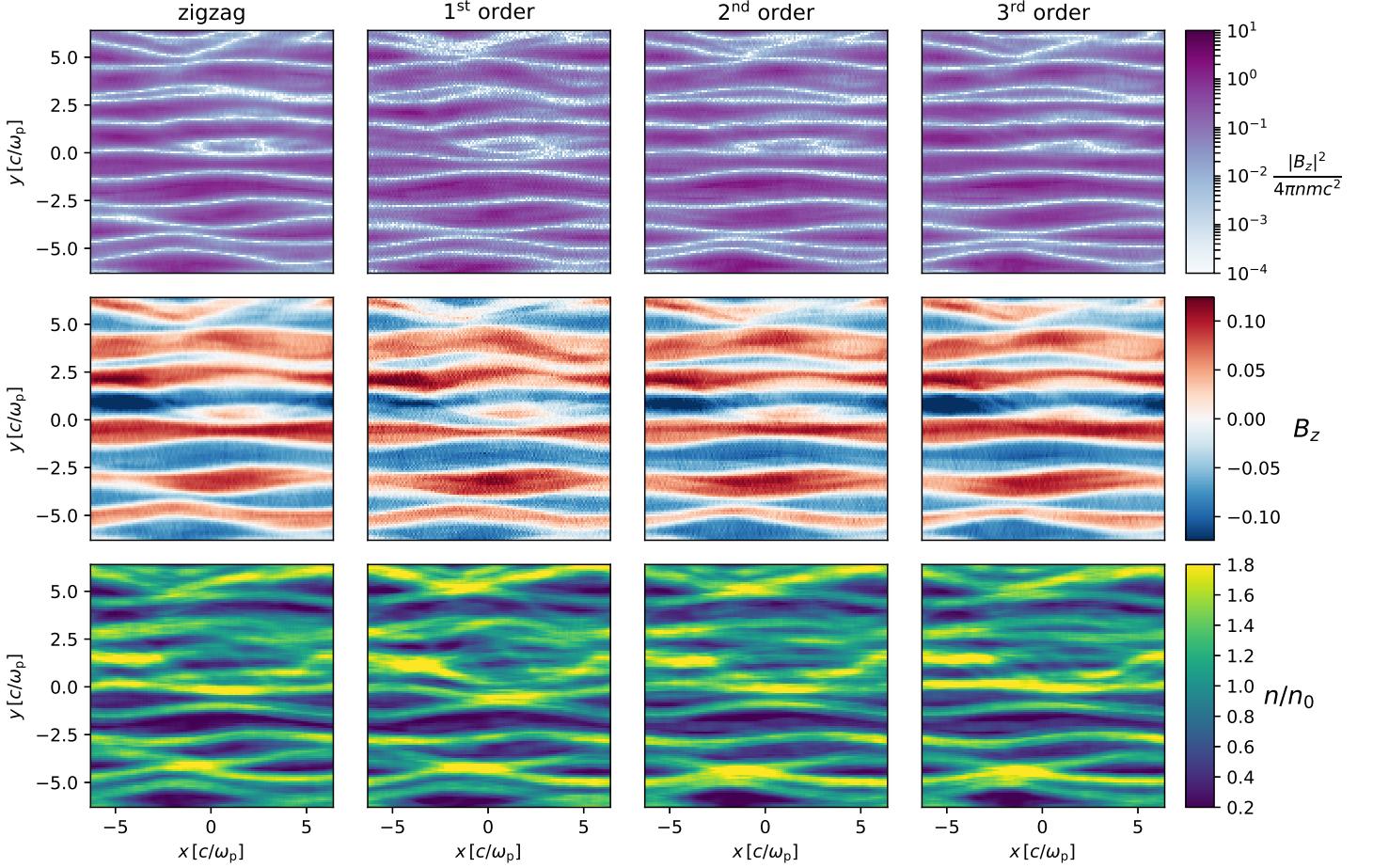


FIG. 6. Similar to Fig. 5, but for the relativistic counterstreaming setup ( $\gamma_b \approx 3.20$ ): 2D spatial profiles of magnetic energy (first row),  $z$  component of magnetic field (second row), and density in units of initial density (third row), for 1<sup>st</sup>, 2<sup>nd</sup>, and 3<sup>rd</sup> order Esirkepov (columns 1 - 4). The spatial profiles are from 2D simulations at  $t\omega_p \approx 16.88$ , which is near saturation of the growth of the unstable mode. The spatial profiles shown here are again from 2D simulations.

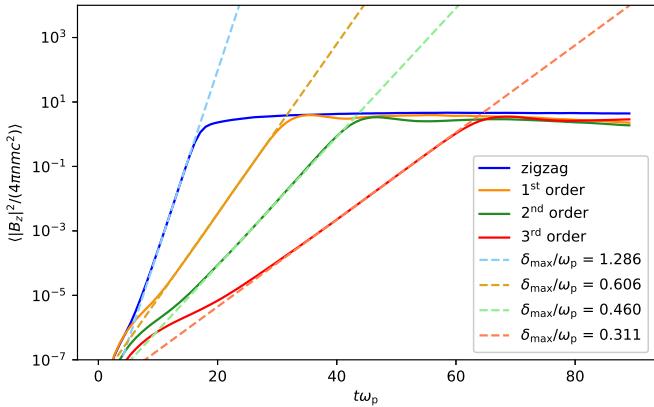


### C. Numerical Cherenkov instability

Numerical dispersion in relativistic plasma flows generates a nonphysical numerical instability, known as the numerical Cherenkov instability.<sup>9</sup> Here, we investigate the effect of different current deposition schemes on the growth of the numerical Cherenkov instability. The simulations presented here are similar to the 2D simulations described in Section III B, however we use a single beam streaming in the  $+x$  direction, with Lorentz factor  $\gamma_b \approx 22.37$ , and  $N_{ppc} = 16$ . Again, the plasma is electron-positron with initial dimensionless temperature  $\theta_0 = 2 \times 10^{-5}$ .

In Fig. 7, we show the growth of  $(2\times)$ perpendicular magnetic energy  $\langle |B_z|^2 \rangle / (4\pi nmc^2)$ , for the current deposit schemes zigzag, 1<sup>st</sup>, 2<sup>nd</sup>, and 3<sup>rd</sup> order Esirkepov. The growth of the instability is shown with respect to dimensionless time  $t\omega_p$ , with the plasma frequency as defined as in Section III B. The solid lines show the measured growth of the instability, whereas dashed lines show fits proportional to  $\exp(\delta_{\max}t)$ , used to extract the maximum growth rate in each case. The growth rate of the instability is reduced by use of higher-order shape functions; the maximum growth rate for the simulation which uses a 3<sup>rd</sup> order form-factor is a factor of two smaller than in the 1<sup>st</sup> order Esirkepov simulation. Compared to the simulation with zigzag current deposit, the simulation which uses a 3<sup>rd</sup> order form-factor has a maximum growth rate that is a factor of four smaller.

FIG. 7. Time evolution of perpendicular magnetic energy for zigzag, 1<sup>st</sup>, 2<sup>nd</sup>, and 3<sup>rd</sup> order Esirkepov (solid lines), and measured maximum growth rates (dashed). These simulations are 2D.



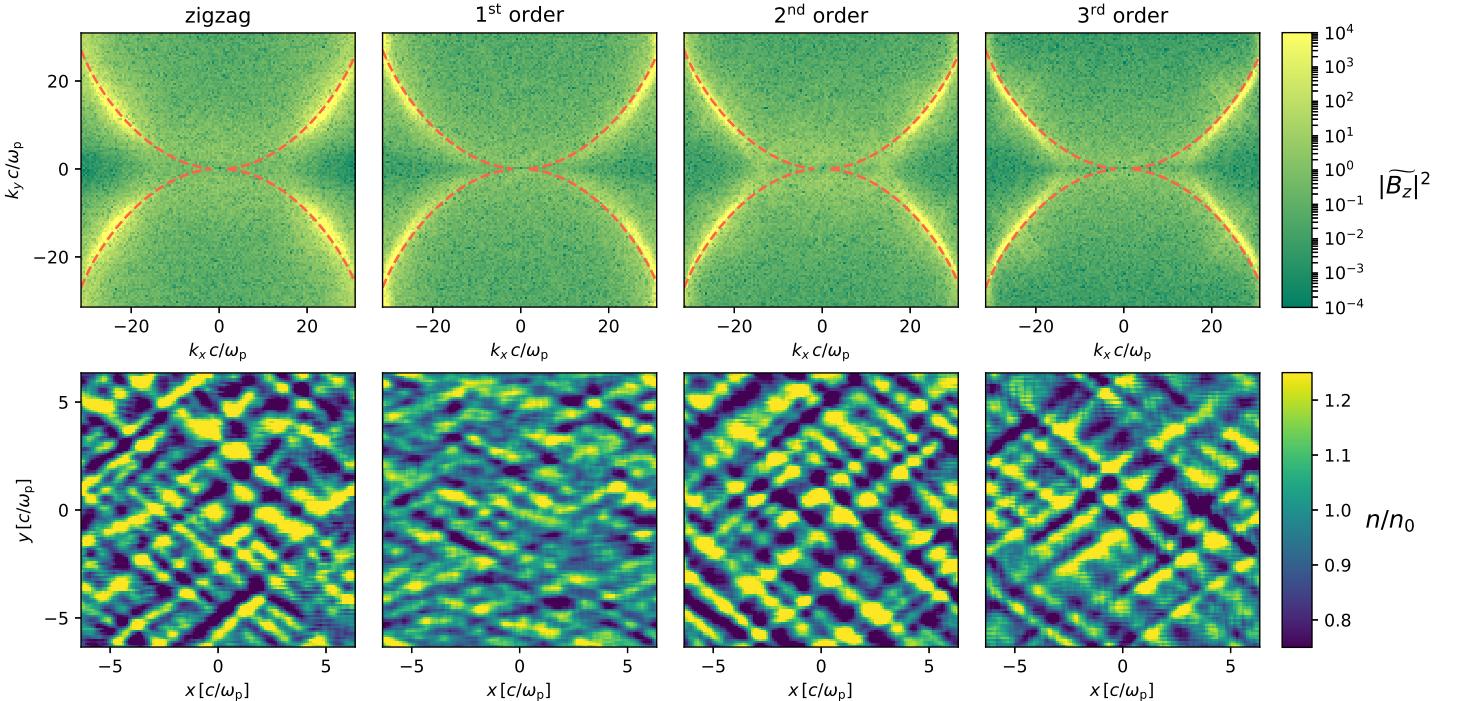
In Fig. 8, we show the  $\mathbf{k}$ -space profiles of the Fourier power spectrum  $|\tilde{B}_z(k_x c/\omega_p, k_y c/\omega_p)|^2$ , in the first row, and the spatial profile of density, in units of initial density, in the second row. Columns 1-4 correspond to zigzag, 1<sup>st</sup>, 2<sup>nd</sup>, and 3<sup>rd</sup> order Esirkepov schemes. For each current deposit scheme, the spatial profiles are shown at a snapshot in time near the end of the exponential growth phase of the numerical Cherenkov instability ( $t\omega_p \approx 20, 40, 50$ , and 70 for zigzag, 1<sup>st</sup>, 2<sup>nd</sup>, and 3<sup>rd</sup> order Esirkepov, respectively; see Fig. 7). The density plots show a streak pattern that is characteristic of the numerical Cherenkov instability. Profiles of the Fourier power spectrum also show a signature of the excited Cherenkov modes. The unstable wave numbers are regulated by the equation<sup>10</sup>

$$k_y \Delta_y = 2 \arcsin \left[ \pm \frac{\Delta_y}{c \Delta_t} \sqrt{\tan^2 \left( \frac{\beta_b k_x c \Delta_t}{2} \right) - \left( c \frac{\Delta_t}{\Delta_x} \sin \left( \frac{k_y \Delta_y}{2} \right) \right)^2} \right], \quad (35)$$

where  $\Delta_t$  is the simulation timestep and  $\beta_b = v_b/c$ , the dimensionless velocity of the beam. In our

simulations,  $\Delta_x = \Delta_y$ . The locus of wavenumbers  $k_x, k_y$  satisfying the condition Eq. 35 is shown by dashed red curves in the first row of Fig. 8.

FIG. 8. For a 2D, relativistic single beam simulation ( $\gamma_b \approx 22.7$ ):  $\mathbf{k}$ -space profiles of the Fourier power spectrum  $|\tilde{B}_z(k_x c/\omega_p, k_y c/\omega_p)|^2$  (first row), and spatial profiles of density in units of initial density (second row), for zigzag, 1<sup>st</sup>, 2<sup>nd</sup>, and 3<sup>rd</sup> order Esirkepov (columns 1 - 4). Dashed red lines in the first row show  $k_x, k_y$  satisfying Eq. 35. For each case, the  $\mathbf{k}$ -space and spatial profiles are shown near saturation of the Numerical Cherenkov instability; the simulations here are the same as those shown in Fig. 7.



#### IV. PERFORMANCE

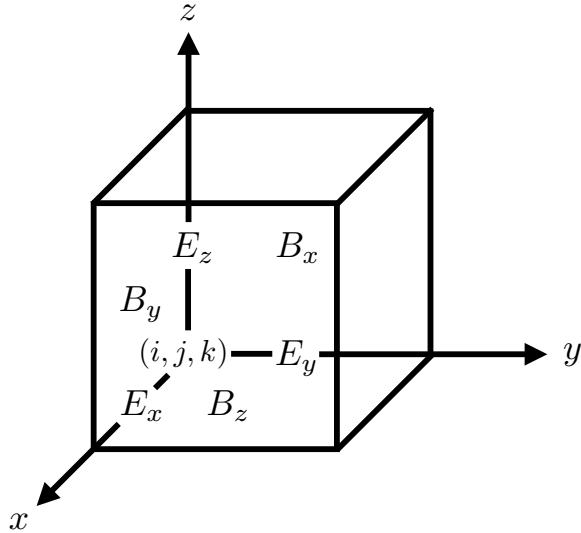
In this section, we compare the performance of our implementation of zigzag, 1<sup>st</sup>, 2<sup>nd</sup>, and 3<sup>rd</sup> order density decomposition for the current deposit step of the PIC loop.

In Fig. 10, we show the average CPU time per step for the current deposit as a function of  $N_{ppc}$ , for 2D and 3D (panels (a) and (b), respectively). The zigzag and 1<sup>st</sup> order Esirkepov schemes show similar performance in 2D, but the zigzag implementation is fastest in 3D, by roughly a factor of 6.5 for the current deposit, and 3.2 for the particle push. For 3D, the average time of the current deposit step, when using a 3<sup>rd</sup> order shape function, is nearly an order of magnitude larger than the time required by zigzag. The number of operations needed during the current deposit step,

when using the Esirkepov method, scales with both the number of particles and the number of array elements over which the 1D particle shape has support (we can call the latter  $N_{\text{shape}}^{\text{1D}}$ ). The average CPU time should scale roughly as  $(N_{\text{ppc}} \times N_{\text{shape}}^{\text{1D}})^2$  in 2D, and  $(N_{\text{ppc}} \times N_{\text{shape}}^{\text{1D}})^3$  in 3D. The dependence on the dimension of the simulation should at least partly explain why, for fixed  $N_{\text{ppc}}$ , the difference in CPU time between the 3<sup>rd</sup> and 1<sup>st</sup> order schemes is greater in 3D than in 2D (see by comparing panels (a) and (b)).

Fig. 11 is similar to Fig. 10, but here we compare the performance of the particle pusher for the different implementations. The performance of each scheme can differ, depending on how the electromagnetic fields are interpolated to the location of a particle. For zigzag, we compute field values via a linear interpolation from Yee lattice points (see Fig. 9) to gridpoints, then gridpoints to particle location; the procedure is described in detail in Cai et. al.<sup>11</sup>

FIG. 9. Diagram of Yee lattice.  $E_x, E_y$ , and  $E_z$  are defined at edge midpoints of the cube;  $B_x, B_y$ , and  $B_z$  are defined at face centers of the cube.



For the 1<sup>st</sup>, 2<sup>nd</sup>, and 3<sup>rd</sup> order schemes, the field value at a particle location  $F(x, y, z)$  is computed as a sum over the product of the particle shape and field, over cells where the form-factor is nonzero:<sup>13</sup>

$$F(x, y, z) = \sum_{n|S^{\text{3D}}(i_n, j_n, k_n) \neq 0} F(i_n, j_n, k_n) S_x^{\text{1D}}(i_n) S_y^{\text{1D}}(j_n) S_z^{\text{1D}}(k_n). \quad (36)$$

Here,  $i_n, j_n$ , and  $k_n$  denote the cell location. The fields at location  $F(i_n, j_n, k_n)$  can be computed via linear interpolation of the fields defined on the Yee lattice. As an example, consider the case

$F = E_x$ ; on the Yee lattice, the  $x$  component of the electric field is defined at points staggered (with respect to the primal grid) by half a cell,  $(i_n + \Delta/2, j_n, k_n)$ . To interpolate from Yee lattice gridpoints to midpoints (thereby obtaining the first term on the righthand side of Eq. 36),  $E_x(i_n, j_n, k_n)$  is computed from two surrounding points as

$$E_x(i_n, j_n, k_n) = \frac{1}{2} [E_x(i_n + \Delta/2, j_n, k_n) + E_x(i_n - \Delta/2, j_n, k_n)], \quad (37)$$

and the other EM field components are computed similarly. With the EM fields computed at the same grid points over which the particle shape function is defined, Eq. 36 yields the field components at the particle location, and allows for calculation of the force on a particle.

From Eq. 36, the particle push step depends on  $(N^{1D})^3$  in 3D, and is also proportional to  $(N_{\text{ppc}})^3$ , so the required time should scale as  $(N_{\text{ppc}} \times N_{\text{shape}}^{1D})^3$  (or in 2D,  $(N_{\text{ppc}} \times N_{\text{shape}}^{1D})^2$ ), as in the current deposit step. Comparing panels (a) and panels (b) in Figs. 10, 11, the current deposit and particle push steps show similar scalings with  $N_{\text{ppc}}$ , for each scheme. The current deposit and particle push steps in 3D show some differences for zigzag and 1st order density decomposition.

FIG. 10. For current deposit, the mean CPU time per step, as a function of particles per cell for (a) 2D and (b) 3D.

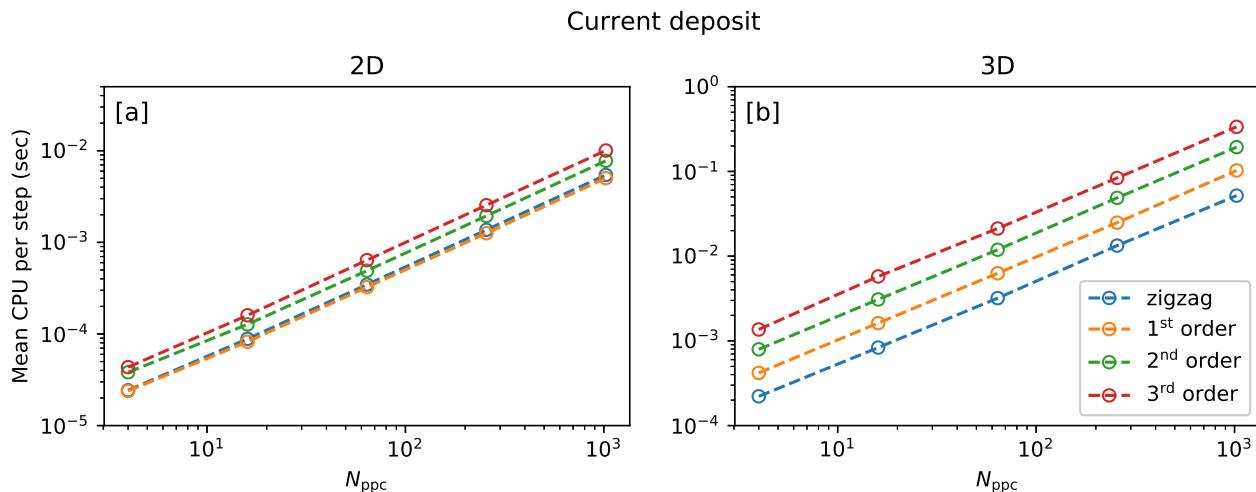
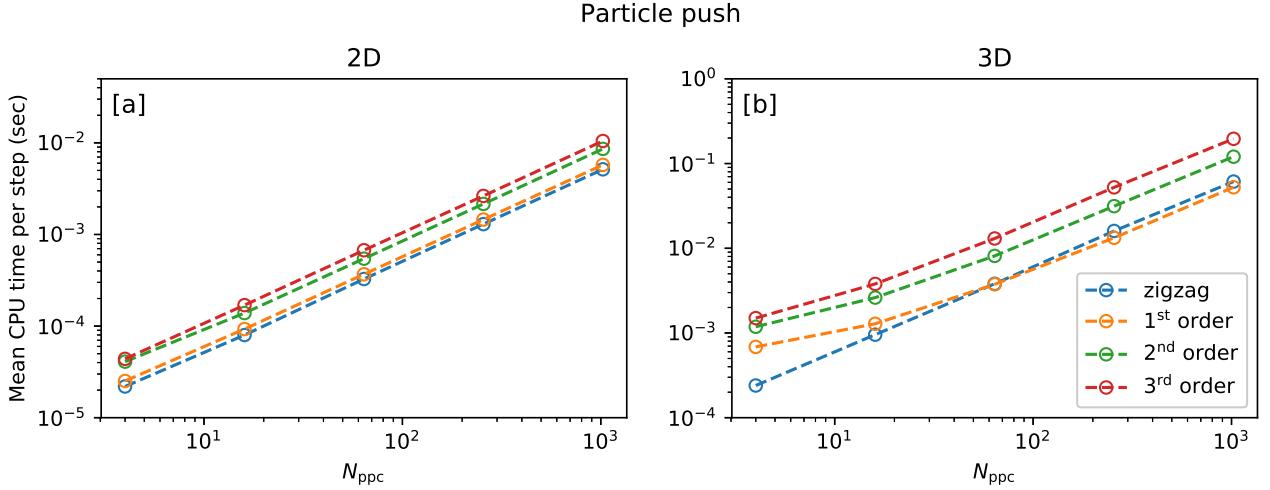


FIG. 11. For particle push, the mean CPU time per step, as a function of particles per cell for (a) 2D and (b) 3D. This is similar to Fig. 10.



## V. SUMMARY

In this chapter, we investigated the use of higher-order shape functions in PIC simulations, by use of the Esirkepov density decomposition scheme for current deposition, and compared with the zigzag scheme. We show that higher-order shape functions offer increased control with respect to numerical heating, and that a smoothing filter can offer similar improvements. We demonstrate the correctness of our implementation by comparing the analytical growth rate of the Filamentation instability to the measured growth rate in simulations. For a relativistic plasma, we demonstrate that the use of higher-order shape functions offers some reduction in the growth rate of the numerical Cherenkov instability. Last, we assess the performance of our implementation of the Esirkepov density decomposition scheme, and compare with zigzag, for the current deposit and particle push steps of the PIC loop, in both 2D and 3D; the 1<sup>st</sup> order density decomposition shows similar performance to zigzag for the current deposit and particle push steps in 2D; in 3D, zigzag current deposit is fastest, and at low  $N_{\text{ppc}}$ , its associated particle push implementation is faster than that of the 1<sup>st</sup> order Esirkepov scheme.

## Appendix A: Esirkepov coefficients

Here, we elaborate on the derivation of numerical coefficients which relate densities  $W_x$ ,  $W_y$ , and  $W_z$  to form-factors (i.e., the coefficients in Eqs. 15, 16, and 17). To shorten notation, define a

vector of the possible form-factors generated by particle motion in 3D on a Cartesian grid, Eq. 8,

$$\begin{aligned} \mathbf{v} = [S(x + \Delta x, y + \Delta y, z + \Delta z), \\ S(x, y + \Delta y, z + \Delta z), S(x + \Delta x, y, z + \Delta z), S(x + \Delta x, y + \Delta y, z), \\ S(x, y, z + \Delta z), S(x, y + \Delta y, z)S(x + \Delta x, y, z), \\ S(x, y, z)]. \end{aligned} \quad (\text{A1})$$

We assume that the densities  $W_x$ ,  $W_y$ , and  $W_z$  are linear combinations of the functions Eq. 8,

$$W_x(x, y, z, \Delta x, \Delta y, \Delta z) = \sum_{i=1}^8 a^i v^i \quad (\text{A2})$$

$$W_y(x, y, z, \Delta x, \Delta y, \Delta z) = \sum_{i=1}^8 b^i v^i \quad (\text{A3})$$

$$W_z(x, y, z, \Delta x, \Delta y, \Delta z) = \sum_{i=1}^8 c^i v^i. \quad (\text{A4})$$

Here, we consider the weights to have six arguments, corresponding to the particle position  $(x, y, z)$  for the first three, and the shift  $(\Delta x, \Delta y, \Delta z)$  for the last three; with this notation,  $W_x(x + \Delta x, y, z) \equiv W_x(x, y, z, \Delta x, 0, 0)$ , for example. Our task is to determine the coefficients  $a^i, b^i, c^i$ , subject to the constraints (1, 2, and 3) on  $\mathbf{W}$  listed in Section II. We translate the three constraints into a system of equations which uniquely determines the 24 coefficients  $a^i, b^i$ , and  $c^i$ .

The first constraint we consider, item 1, can be understood as enforcing that the weights  $W_x$ ,  $W_y$ , and  $W_z$  satisfy the continuity equation (Eq. 2). Inserting the definitions A2, A3, and A4 into Eq. 7, we obtain

$$\sum_{i=1}^8 (a^i + b^i + c^i) v^i = S(x + \Delta x, y + \Delta y, z + \Delta z) - S(x, y, z). \quad (\text{A5})$$

The next property of  $\mathbf{W}$ , item 2, ensures that if the particle shift in a direction is zero, then no current is deposited along that direction. This allows us to set  $\Delta x, \Delta y$ , and  $\Delta z$  each to zero in Eq. A5 to get three sets of constraints,

$$W_x(x, y, z, \Delta x, \Delta y, 0) + W_y(x, y, z, \Delta x, \Delta y, 0) = S(x + \Delta x, y + \Delta y, z) - S(x, y, z) \quad (\text{A6})$$

$$W_x(x, y, z, \Delta x, 0, \Delta z) + W_z(x, y, z, \Delta x, 0, \Delta z) = S(x + \Delta x, y, z + \Delta z) - S(x, y, z) \quad (\text{A7})$$

$$W_y(x, y, z, 0, \Delta y, \Delta z) + W_z(x, y, z, 0, \Delta x, \Delta z) = S(x, y + \Delta y, z + \Delta z) - S(x, y, z) \quad (\text{A8})$$

or

$$\sum_{i=1}^8 a^i v^i|_{\Delta z=0} + \sum_{i=1}^8 b^i v^i|_{\Delta z=0} = S(x + \Delta x, y + \Delta y, z) - S(x, y, z) \quad (\text{A9})$$

$$\sum_{i=1}^8 a^i v^i|_{\Delta y=0} + \sum_{i=1}^8 c^i v^i|_{\Delta y=0} = S(x + \Delta x, y, z + \Delta z) - S(x, y, z) \quad (\text{A10})$$

$$\sum_{i=1}^8 b^i v^i|_{\Delta x=0} + \sum_{i=1}^8 c^i v^i|_{\Delta x=0} = S(x, y + \Delta y, z + \Delta z) - S(x, y, z). \quad (\text{A11})$$

Additionally, we have

$$W_x(x, y, z, \Delta x, \Delta y, 0) = 0 \quad (\text{A12})$$

$$W_x(x, y, z, \Delta x, 0, \Delta z) = 0 \quad (\text{A13})$$

$$W_y(x, y, z, 0, \Delta y, \Delta z) = 0, \quad (\text{A14})$$

which yields 12 constraints,

$$a^1 = -a^2, a^3 = -a^4, a^5 = -a^6, a^7 = -a^8, \quad (\text{A15})$$

$$b^1 = -b^2, b^3 = -b^4, b^5 = -b^6, b^7 = -b^8, \quad (\text{A16})$$

$$c^1 = -c^2, c^3 = -c^4, c^5 = -c^6, c^7 = -c^8 \quad (\text{A17})$$

Lastly, we consider the symmetry constraint, item 3. Suppose that the form-factors are invariant with respect to permutation of coordinate, and shifts are identical:  $\Delta x = \Delta y = \Delta z = \Delta$ . Then, for example,  $S(x, y + \Delta, z) = S(y + \Delta, x, z)$ . Moreover, invariance with respect to permutation of  $(x, y)$ ,  $(y, z)$ , and  $(x, z)$  yield, respectively, the equations

$$W_x(x, y, z, \Delta, \Delta, \Delta) - W_y(x, y, z, \Delta, \Delta, \Delta) = 0 \quad (\text{A18})$$

$$W_y(x, y, z, \Delta, \Delta, \Delta) - W_z(x, y, z, \Delta, \Delta, \Delta) = 0 \quad (\text{A19})$$

$$W_x(x, y, z, \Delta, \Delta, \Delta) - W_z(x, y, z, \Delta, \Delta, \Delta) = 0, \quad (\text{A20})$$

or, inserting Eqs. A2, A3, and A4,

$$\begin{aligned} & (a^1 - b^1)S(x + \Delta, y + \Delta, z + \Delta) + (a^2 - b^3)S(x, y + \Delta, z + \Delta) \\ & + (a^3 - b^2)S(x + \Delta, y, z + \Delta) + (a^4 - b^4)S(x, y, z + \Delta) \\ & + (a^5 - b^5)S(x + \Delta, y + \Delta, z) + (a^6 - b^7)S(x, y + \Delta, z) \\ & + (a^7 - b^6)S(x + \Delta, y, z) + (a^8 - b^8)S(x, y, z) = 0 \end{aligned} \quad (\text{A21})$$

$$\begin{aligned}
& (b^1 - c^1)S(x + \Delta, y + \Delta, z + \Delta) + (b^2 - c^5)S(x + \Delta, y, z + \Delta) \\
& + (b^3 - c^3)S(x, y + \Delta, z + \Delta) + (b^4 - c^7)S(x, y, z + \Delta) \\
& + (b^5 - c^2)S(x + \Delta, y + \Delta, z) + (b^6 - c^6)S(x + \Delta, y, z) \\
& + (b^7 - c^4)S(x, y + \Delta, z) + (b^8 - c^8)S(x, y, z) = 0
\end{aligned} \tag{A22}$$

$$\begin{aligned}
& (a^1 - c^1)S(x + \Delta, y + \Delta, z + \Delta) + (a^2 - c^3)S(x, y + \Delta, z + \Delta) \\
& + (a^3 - c^5)S(x + \Delta, y, z + \Delta) + (a^4 - c^7)S(x, y, z + \Delta) \\
& + (a^5 - c^2)S(x + \Delta, y + \Delta, z) + (a^6 - c^4)S(x, y + \Delta, z) \\
& + (a^7 - c^6)S(x + \Delta, y, z) + (a^8 - c^8)S(x, y, z) = 0.
\end{aligned} \tag{A23}$$

Enforcing the symmetry property in each of Eqs. A21, A22, and A23, we can group together equivalent form-factors. The linearly independent terms in each equation must still vanish, individually, and we obtain the constraints

$$a^8 = b^8, a^4 = b^4, a^5 = b^5, a^1 = b^1, a^7 - b^6 = b^7 - a^6, a^3 - b^2 = b^3 - a^2 \tag{A24}$$

$$b^8 = c^8, b^3 = c^3, b^6 = c^6, b^1 = c^1, b^4 - c^7 = c^4 - b^7, b^2 - c^5 = c^2 - b^5 \tag{A25}$$

$$a^8 = c^8, a^1 = c^1, a^6 = c^4, a^3 = c^5, a^4 - c^7 = c^6 - a^7, a^2 - a^3 = c^2 - a^5. \tag{A26}$$

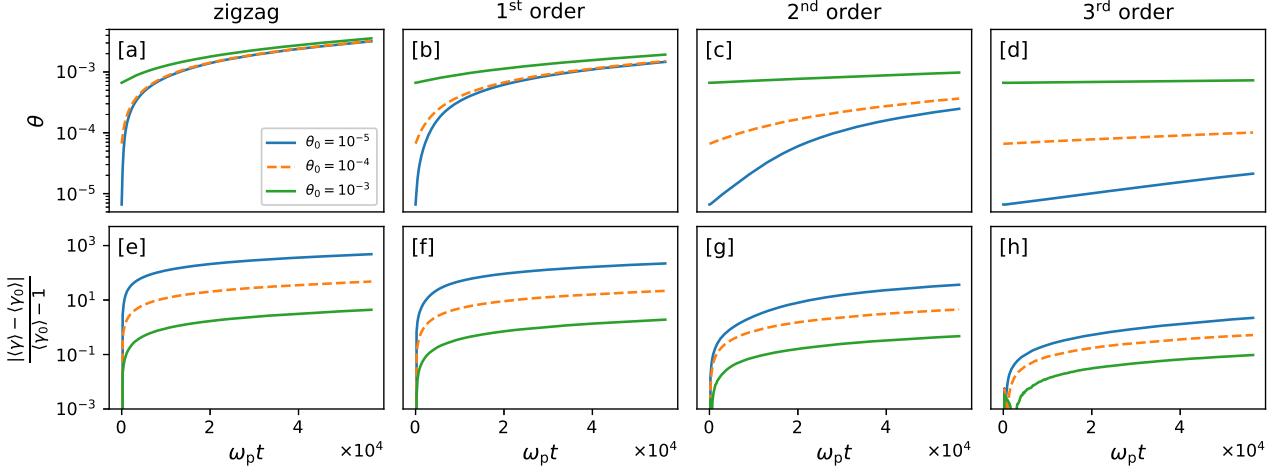
Solving the (overconstrained, but still consistent) set of Eqs. A5, A9, A10, A11, A15, A16, A17, A24, A25, and A26, we obtain the coefficients,

$$\begin{aligned}
a^1 &= \frac{1}{3}, a^2 = -\frac{1}{3}, a^3 = \frac{1}{6}, a^4 = -\frac{1}{6}, a^5 = \frac{1}{6}, a^6 = -\frac{1}{6}, a^7 = \frac{1}{3}, a^8 = -\frac{1}{3}, \\
b^1 &= \frac{1}{3}, b^2 = -\frac{1}{3}, b^3 = \frac{1}{6}, b^4 = -\frac{1}{6}, b^5 = \frac{1}{6}, b^6 = -\frac{1}{6}, b^7 = \frac{1}{3}, b^8 = -\frac{1}{3}, \\
c^1 &= \frac{1}{3}, c^2 = -\frac{1}{3}, c^3 = \frac{1}{6}, c^4 = -\frac{1}{6}, c^5 = \frac{1}{6}, c^6 = -\frac{1}{6}, c^7 = \frac{1}{3}, c^8 = -\frac{1}{3}.
\end{aligned} \tag{A27}$$

## Appendix B: Numerical heating in 2D

Fig. 12 shows the evolution of dimensionless temperature and fractional error for a set of simulations with initial dimensionless temperatures  $\theta_0 = 10^{-5}, 10^{-4}$ , and  $10^{-3}$ , in 2D. Apart from box dimensions and the distribution used for initialization of the plasma (see Eq. 33), the numerical parameters of these simulations are the same as in the 3D cases presented in Section III A.

FIG. 12. Similar to Fig. 2, but for 2D rather than 3D simulations; ((a) - (d)) Evolution of dimensionless temperature and ((e) - (h)) fractional error in a uniform plasma with initial dimensionless temperatures  $\theta_0 = 10^{-5}, 10^{-4}$ , and  $10^{-3}$ , for zigzag, 1<sup>st</sup>, 2<sup>nd</sup>, and 3<sup>rd</sup> order Esirkepov (columns 1 - 4, respectively).



- <sup>1</sup> Buneman, O., Nishikawa, K.-I., and Neubert, T. (1993). *Plasma Physics and Controlled Nuclear Fusion (ITC-4)*, ed. H. T. D. Guyenne, and J. J. Hunt, (ESA), 285.
- <sup>2</sup> Spitkovsky, A. (2005). *AIP Conf. Ser.* 801, Astrophysical Sources of High Energy Particles and Radiation, ed. T. Bulik, B. Rudak, and G. Madejski. (Melville, NY: AIP), 345.
- <sup>3</sup> Shalaby M., Broderick A. E., Chang P., Pfrommer C., Lamberts A., and Puchwein E. (2017). *ApJ* 841 (1), 52.
- <sup>4</sup> Esirkepov, T. Z. (2001). *CoPhC*, 135, 144.
- <sup>5</sup> Umeda, T., Omura, Y., Tominaga, T., and Matsumoto, H. 2003, *CoPhC*, 156, 73.
- <sup>6</sup> Birdsall, C. K., and Maron, N. (1980). *J. Comp. Phys.*, 36,1.
- <sup>7</sup> Birdsall, C. K., and Langdon, A. B. (1991). *Plasma Physics via Computer Simulation*.
- <sup>8</sup> Bret, A. (2012). CfA Plasma Talks. ArXiv:1205.6259.
- <sup>9</sup> Godfrey, B. (1974). *J. Comp. Phys.*, 15, 504.
- <sup>10</sup> Ikeya N., and Matsumoto Y. (2015). *PASJ*, 67, 64.
- <sup>11</sup> Cai, D., Li, Y., Nishikawa, K.-I., Xiao, C., Yan, X., and Pu, Z. (2003). *Space Plasma Simulation*, ed. J. Bchnner, C. Dum, and M. Scholer., Lecture Notes in Physics, 615, 25-53.
- <sup>12</sup> These definitions are for 1D shape functions. Note that the form-factors in Eqs. 18, 19, 20, 15, 16, and 17 are 2D and 3D shape functions. The relation between 1D and 2D is:  $S^{2D}(x, y) = S^{1D}(x)S^{1D}(y)$ , for example (and similarly for 3D).

<sup>13</sup> The field interpolation in 2D is analogous; homogeneity is assumed in the  $z$  direction, so  $F(x, y, z) = F(x, y) = \sum_{n|S^{2D}(i_n, j_n) \neq 0} F(i_n, j_n) S_x^{1D}(i_n) S_y^{1D}(j_n)$ .