Journal of Computational Physics 519 (2024) 113427

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

An implicit particle code with *exact* energy and charge conservation for studies of dense plasmas in axisymmetric geometries

Justin Ray Angus *, William Farmer, Alex Friedman, Vasily Geyko, Debojyoti Ghosh, Dave Grote, David Larson, Anthony Link

Lawrence Livermore National Laboratory, Livermore, CA 94551, USA

A B S T R A C T

A collisional particle code based on implicit energy- and charge-conserving methods in axisymmetric geometries is presented. A new particle pusher for axisymmetric systems is introduced that is compatible with exact energy and charge conservation and yields improved accuracy compared to other methods. How to appropriately treat all aspects of the algorithm near the $r = 0$ axis of symmetry is described in detail. The axisymmetric model is verified by simulating the free expansion of a plasma sphere in 2D cylindrical and 1D spherical geometries. The algorithm's ability to study the dynamic compression of a dense plasma is illustrated by simulating the dynamic Z-pinch in 1D cylindrical geometry.

1. Introduction

The particle-in-cell Monte-Carlo collision (PIC-MCC) method is a widely used numerical technique for studying ideal plasmas as governed by the Maxwell-Boltzmann set of equations [1–4]. Conventional approaches, such as the explicit leap-frog method and the semi-implicit direct-implicit method [5,6], have well-known problems associated with the inability to conserve energy. The energy in the system is known to artificially increase over time (i.e., heating) when using conventional explicit methods, while it typically decreases over time (i.e., cooling) when using conventional semi-implicit methods. The rate of artificial heating or cooling for these methods can drastically increase for collisional plasmas [7–9]. Energy-conservation issues, along with other time-step and grid-size restraints, limit the applicability of conventional PIC-MCC methods for high-fidelity simulations of dense plasma systems, such as those found in inertial-confinement-fusion (ICF) and dense Z-pinch (DZP) experiments.

Plasmas found in ICF and DZP systems can be characterized as dense because the dynamic timescales of interest are often long compared to the timescales for plasma oscillations and collisions. While fluid models can be efficiently applied to study dense plasmas, they fail to capture kinetic effects such as non-local heat flux at shock fronts [10] and transport/deposition of high-energy beam-like particles [11,12]. In a previous work by some of the Authors, a collisional particle code suitable for high fidelity simulations of dense plasmas was introduced [13]. The algorithm in Ref. [13] is based on the relatively new fully implicit PIC method [14,15], which has the attractive property of being *exactly* energy conserving, even when the plasma is strongly collisional [9]. Using the planar analog of the dynamic pinch in 1D [16], it is shown in Ref. [13] that the algorithm is capable of accurately and efficiently simulating the dynamic compression of a dense plasma using time steps much longer than the plasma period, grid cell sizes much larger than the Debye length, and a moderate number of simulation particles.

* Corresponding author.
E-mail address: angus1@llnl.gov (J.R. Angus).

https://doi.org/10.1016/j.jcp.2024.113427
Received 9 April 2024; Received in revised form 22 August 2024; Accepted 9 September 2024

Available online 11 September 2024
0021-9991/Published by Elsevier Inc.

Fig. 1. Cylindrical coordinates (left) and spherical coordinates (right) used in this work. The azimuthal angle θ is the same for both coordinate systems and ranges from 0 to 2π. The polar angle ϕ in the spherical coordinate system ranges from $-\pi/2$ to $\pi/2$.

A cylindrical implementation of the energy-conserving semi-implicit PIC method [17] is presented in Ref. [18]. A general curvilinear formulation of the fully implicit Vlasov-Darwin PIC method in 2D is given in Ref. [19]. Curvilinear implementations of structure-preserving PIC methods can be found in Refs. [20,21]. The focus of the present paper is to extend the fully implicit PIC-MCC algorithm from Ref. [13] from planar to axisymmetric geometries. The geometries considered are 1D/2D cylindrical, and 1D spherical. The algorithm employs a new Cartesian-based particle orbit integrator. Steps taken to maintain the exact energy and exact charge conserving properties of the algorithm are described in detail. Particular attention is paid to the treatment at the $r = 0$ axis of symmetry.

The axisymmetric coordinates systems are given in the following section, Sec. 2, followed by a presentation of the algorithm in Sec. 3. The Cartesian particle orbit integrator is outlined in Sec. 4. The implementation of the implicit solver in axisymmetric geometries is discussed in Sec. 5. Results from free expansion of a plasma sphere simulations in 2D cylindrical and 1D spherical geometry are given in Sec. 6. In Sec. 7, the algorithm is used to simulate the dynamic Z-pinch in 1D cylindrical geometry. Further discussion and a summary are given in Sec. 8.

2. Axisymmetric coordinate systems

Axisymmetric cylindrical and spherical geometries are considered in this work. The logical coordinates are $\boldsymbol{\xi} = (r, \theta, z)$ for cylindrical and $\boldsymbol{\xi} = (r, \theta, \phi)$ for spherical. In both coordinate systems (see Fig. 1), θ is the azimuthal angle that ranges from 0 to 2π. The angle ϕ in the spherical coordinate system is the polar angle that ranges from $-\pi/2$ to $\pi/2$. The relationship between the logical and physical coordinates in cylindrical geometry is given by

$$x = r\cos\theta,\ y = r\sin\theta \ \Rightarrow r = \sqrt{x^2 + y^2},\ \tan\theta = \frac{y}{x}. \tag{1}$$

For spherical geometry, the relations are

$$x = r\cos\phi\cos\theta,\ y = r\cos\phi\sin\theta,\ z = r\sin\phi \ \Rightarrow r = \sqrt{x^2 + y^2 + z^2},\ \tan\theta = \frac{y}{x},\ \sin\phi = \frac{z}{r}. \tag{2}$$

The unit vectors for the cylindrical and spherical coordinate system are related to the physical unit vectors, respectively, by

$$\begin{bmatrix} \hat{\mathbf{r}} \\ \hat{\boldsymbol{\theta}} \\ \hat{\mathbf{z}} \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}} \\ \hat{\mathbf{y}} \\ \hat{\mathbf{z}} \end{bmatrix}, \text{ and } \begin{bmatrix} \hat{\mathbf{r}} \\ \hat{\boldsymbol{\theta}} \\ \hat{\boldsymbol{\phi}} \end{bmatrix} = \begin{bmatrix} \cos\phi\cos\theta & \cos\phi\sin\theta & \sin\phi \\ -\sin\theta & \cos\theta & 0 \\ -\sin\phi\cos\theta & -\sin\phi\sin\theta & \cos\phi \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}} \\ \hat{\mathbf{y}} \\ \hat{\mathbf{z}} \end{bmatrix}. \tag{3}$$

$\mathbb{M}$ is used below to denote the mapping matrices given in Eq. (3). The inverse mapping matrices, used to transform a vector from mapped to Cartesian are the transpose of those given in Eq. (3): $\mathbb{M}^{-1} = \mathbb{M}^T$.

3. The Maxwell-Boltzmann system and the implicit θ-PIC-MCC algorithm

The fully implicit PIC-MCC algorithm is described in this section. Starting with a presentation of the Maxwell-Boltzmann set of equations, the discrete numerical formulation follows, and a corresponding discrete energy conservation law is presented. Finally, the interpolation function used to achieve local charge conservation is discussed.

3.1. The Maxwell-Boltzmann set of equations: axisymmetric geometries

The Boltzmann equation governing the evolution of the distribution function of species α, f_α, in physical phase-space coordinates $(\mathbf{x}, \mathbf{u})$ can be written as

$$\frac{\partial f_\alpha}{\partial t} + \frac{d\mathbf{x}}{dt} \cdot \frac{\partial f_\alpha}{\partial \mathbf{x}} + \frac{d\mathbf{u}}{dt} \cdot \frac{\partial f_\alpha}{\partial \mathbf{u}} = C_\alpha. \tag{4}$$

The left-hand side is the phase-space continuity law governing long-range (i.e., collisionless) phenomena, and C_α on the right-hand side is the operator representing discrete discontinuities in phase space owing to short-range collisional processes. For a fully ionized ideal plasma, C_α is that corresponding to screened Coulomb collisions, same as that considered in Ref. [13].

The time derivatives of the physical phase space coordinates are obtained from the equations of motion, which in the non-relativistic limit are

$$\frac{d\mathbf{x}}{dt} = \mathbf{u}, \quad \frac{d\mathbf{u}}{dt} = \frac{q_\alpha}{m_\alpha}\left[\mathbf{E}(\mathbf{x}) + \mathbf{u}\times\mathbf{B}(\mathbf{x})\right], \tag{5}$$

where m_α and q_α are the mass and charge, respectively, of species α. $\mathbf{E}$ and $\mathbf{B}$ here are the physical (i.e., Cartesian) electric and magnetic fields, respectively. These quantities are obtained from the logical space fields using the inverse of the mapping matrices defined in Eq. (3): $\mathbf{E} = \mathbb{M}^{-1}\mathbf{E}^\xi$, $\mathbf{B} = \mathbb{M}^{-1}\mathbf{B}^\xi$. Here, the superscript ξ is used to denote logical space for vector quantities. For example, $\mathbf{E}^\xi = E_r\hat{\mathbf{r}} + E_\theta\hat{\boldsymbol{\theta}} + E_z\hat{\mathbf{z}}$ for cylindrical geometry. The electric and magnetic fields are governed by the laws of Ampère and Faraday. These laws, along with the divergence constraints on the fields, are expressed in logical space as

$$\frac{1}{c^2}\frac{\partial\mathbf{E}^\xi}{\partial t} = \nabla_\xi\times\mathbf{B}^\xi - \mu_0\mathbf{J}^\xi, \quad \nabla_\xi\cdot\epsilon_0\mathbf{E}^\xi = \rho, \tag{6}$$

$$\frac{\partial\mathbf{B}^\xi}{\partial t} = -\nabla_\xi\times\mathbf{E}^\xi, \qquad \nabla_\xi\cdot\mathbf{B}^\xi = 0. \tag{7}$$

Here, $c = 1/\sqrt{\epsilon_0\mu_0}$ is the speed of light, ρ is the charge density, and $\mathbf{J}^\xi$ is the logical current density. These latter quantities are obtained from the species distribution functions:

$$\rho = \sum_{\alpha\in s} q_\alpha \int f_\alpha d\mathbf{u}^3, \quad \mathbf{J}^\xi = \sum_{\alpha\in s} q_\alpha \int \mathbb{M}\mathbf{u} f_\alpha d\mathbf{u}^3, \tag{8}$$

where $d\mathbf{u}^3$ is the velocity space volume element and the sum is over all species α. For the axisymmetric geometries considered here, the logical space curl and divergence operators in Eqs. (6)-(7) can be written as

$$\nabla_\xi\times\mathbf{F}^\xi\cdot\hat{\mathbf{r}} = -\frac{\partial F_\theta}{\partial z}, \quad \nabla_\xi\times\mathbf{F}^\xi\cdot\hat{\boldsymbol{\theta}} = \frac{\partial F_r}{\partial z} - \frac{\partial F_z}{\partial r}, \quad \nabla_\xi\times\mathbf{F}^\xi\cdot\hat{\mathbf{z}} = \frac{1}{r}\frac{\partial\left(rF_\theta\right)}{\partial r}, \tag{9}$$

$$\nabla_\xi\cdot\mathbf{F}^\xi = \frac{1}{r}\frac{\partial\left(rF_r\right)}{\partial r} + \frac{\partial F_z}{\partial z}, \tag{10}$$

for 2D cylindrical geometry (set $\partial/\partial z$ terms to zero for 1D cylindrical) and as

$$\nabla_\xi\times\mathbf{F}^\xi\cdot\hat{\mathbf{r}} = 0, \quad \nabla_\xi\times\mathbf{F}^\xi\cdot\hat{\boldsymbol{\theta}} = -\frac{1}{r}\frac{\partial\left(rF_\phi\right)}{\partial r}, \quad \nabla_\xi\times\mathbf{F}^\xi\cdot\hat{\boldsymbol{\phi}} = \frac{1}{r}\frac{\partial\left(rF_\theta\right)}{\partial r}, \tag{11}$$

$$\nabla_\xi\cdot\mathbf{F}^\xi = \frac{1}{r^2}\frac{\partial\left(r^2F_r\right)}{\partial r}, \tag{12}$$

for 1D spherical geometry. See Appendix A for a formal derivation of the curl and divergence operators for the spherical coordinates used in this work.

3.2. The implicit PIC-MCC algorithm

The numerical stencil considered here is essentially the same as that presented previously in Refs. [9,13], which is an implicit θ-PIC method [14,22,23] coupled with a binary MCC model for Coulomb collisions [24]. However, only a value of 1/2 is considered for the time-biasing parameter θ, which is the value needed for exact energy conservation.

In a PIC scheme, the distribution function is approximated as a collection of discrete (i.e., macro) particles with weight w_p. The mass and charge associated with particle $p\in\alpha$ are $m_p = w_p m_\alpha$ and $q_p = w_p q_\alpha$, respectively. The discrete version of the equations governing the phase-space trajectories of the macro particles (Eqs. (5)) are

$$\frac{\mathbf{x}_p^{n+1} - \mathbf{x}_p^n}{\Delta t} = \bar{\mathbf{u}}_p, \quad \frac{\mathbf{u}_p^{n+1} - \mathbf{u}_p^n}{\Delta t} = \frac{q_p}{m_p}\left(\bar{\mathbf{E}}_p + \bar{\mathbf{u}}_p\times\bar{\mathbf{B}}_p\right), \tag{13}$$

where $\bar{\mathbf{E}}_p = \mathbb{M}^{-1}\sum_g S^{\mathbf{E}}_{gp}\bar{\mathbf{E}}^\xi_g$ and $\bar{\mathbf{B}}_p = \mathbb{M}^{-1}\sum_g S^{\mathbf{B}}_{gp}\bar{\mathbf{B}}^\xi_g$ are the Cartesian fields at the particle obtained by first interpolating the mapped field values on the logical space grid to the particle positions using the shape functions, $S^{\mathbf{E}}_{gp}$ and $S^{\mathbf{B}}_{gp}$, and then transforming to Cartesian using the inverse mapping matrix $\mathbb{M}^{-1}$. The bar notation is used to represent time-centered quantities (e.g., $\bar{\mathbf{u}}_p \equiv \left(\mathbf{u}_p^{n+1} + \mathbf{u}_p^n\right)/2$). Like the model described in Ref. [19], the fields are discretized with a 2nd-order centered finite difference approximation on a uniform logical-space Yee grid [25] where $\mathbf{J}^\xi$ and $\mathbf{E}^\xi$ are defined along cell edges and $\mathbf{B}^\xi$ is defined on cell faces (see Fig. B.12 in Appendix B). This formalism ensures that the physical condition $\nabla_\xi\cdot\mathbf{B}^\xi = 0$ is satisfied discretely. The discrete versions of the laws of Ampère and Faraday, Eqs. (6)-(7), used to advance $\mathbf{E}^\xi$ and $\mathbf{B}^\xi$ at logical grid location $\boldsymbol{\xi}_g$ are given as

$$\frac{\mathbf{E}_g^{\xi,n+1}-\mathbf{E}_g^{\xi,n}}{c^2\Delta t}=\nabla_\xi\times\bar{\mathbf{B}}_g^{\xi}-\mu_0\bar{\mathbf{J}}_g^{\xi},\qquad \frac{\mathbf{B}_g^{\xi,n+1}-\mathbf{B}_g^{\xi,n}}{\Delta t}=-\nabla_\xi\times\bar{\mathbf{E}}_g^{\xi}, \tag{14}$$

where $\bar{\mathbf{J}}_g^{\xi}$ is the time-centered logical current density at grid location $\mathbf{x}_g$ defined as

$$\bar{\mathbf{J}}_g^{\xi}=\sum_s\sum_{p\in s}\frac{q_p\mathbb{M}\bar{\mathbf{u}}_p}{\Delta V_g}S_{gp}^{\mathbf{J}}. \tag{15}$$

Here, $S_{gp}^{\mathbf{J}}$ is the shape function used to interpolate the mapped particle velocity, $\mathbb{M}\bar{\mathbf{u}}_p$, associated with particle p to logical grid position $\boldsymbol{\xi}_g$. A necessary condition for exact energy conservation is that $S_{gp}^{\mathbf{E}}=S_{gp}^{\mathbf{J}}$. The discrete volume element in Eq. (15) is defined as $\Delta V_g=2\pi r_g\Delta r\Delta z\alpha_g$ for 2D cylindrical geometry and $\Delta V_g=4\pi r_g^2\Delta r\alpha_g$ for 1D spherical geometry. The coefficient α_g equals one if the grid location $\boldsymbol{\xi}_g$ is in the interior of the domain, is set to 1/2 if $\boldsymbol{\xi}_g$ lives on a non-periodic domain boundary other than the axis and has a special definition on axis to be discussed in the following subsection where the discrete energy law is derived.

3.3. Discrete energy law

A derivation of the discrete energy conservation law for the implicit method considered here, including finite fluxes of electromagnetic energy at domain boundaries, is given in Refs. [13] considering a planar geometry. That derivation is revisited here considering axisymmetric geometries that include the axis. It is assumed that the domain boundaries are aligned with the tangential components of the electric field. The energy in the fields at time step n is defined as

$$W_{fields}^n=W_E^n+W_B^n=\frac{\epsilon_0}{2}\sum_g\mathbf{E}_g^{\xi,n}\cdot\mathbf{E}_g^{\xi,n}\Delta V_g+\frac{1}{2\mu_0}\sum_g\mathbf{B}_g^{\xi,n}\cdot\mathbf{B}_g^{\xi,n}\Delta V_g, \tag{16}$$

where ΔV_g is defined below Eq. (15). With some algebriac manipulation of Eqs. (14) [9], the discrete energy law for the fields can be expressed as

$$\frac{W_{fields}^{n+1}-W_{fields}^n}{\Delta t}+\sum_g\nabla_\xi\cdot\bar{\boldsymbol{\mathcal{S}}}_g^{\xi}\Delta V_g=-\sum_g\bar{\mathbf{J}}_g^{\xi}\cdot\bar{\mathbf{E}}_g^{\xi}\Delta V_g. \tag{17}$$

The second term on the left-hand-side is the discrete version of the volume integral of the divergence of the Poynting flux, $\boldsymbol{\mathcal{S}}=\mathbf{E}\times\mathbf{B}/\mu_0$. The explicit definition of this quantity is

$$\sum_g\nabla_\xi\cdot\bar{\boldsymbol{\mathcal{S}}}_g^{\xi}\Delta V_g\equiv\frac{1}{\mu_0}\sum_g\left[\bar{\mathbf{B}}_g^{\xi}\cdot\nabla_\xi\times\bar{\mathbf{E}}_g^{\xi}-\bar{\mathbf{E}}_g^{\xi}\cdot\nabla_\xi\times\bar{\mathbf{B}}_g^{\xi}\right]\Delta V_g. \tag{18}$$

This term can be written as a surface integral of an appropriately defined discrete definition of the Poynting flux [13]. For simplicity, consider a 1D radial grid and let that the magnetic field in cylindrical coordinates be polarized such that $\mathbf{B}^{\xi}=B_\theta\hat{\boldsymbol{\theta}}$. With these simplifications, the Poynting flux term in cylindrical coordinates simplifies to the following:

$$\sum_g\nabla_\xi\cdot\bar{\boldsymbol{\mathcal{S}}}_g^{\xi}\Delta V_g=\frac{2\pi R}{\mu_0}\frac{\left(r_{N_r+\frac{1}{2}}\bar{B}_{\theta,N_r+\frac{1}{2}}+r_{N_r-\frac{1}{2}}\bar{B}_{\theta,N_r-\frac{1}{2}}\right)}{2R}\bar{E}_{z,N_r}-\frac{2\pi}{\mu_0}r_{\frac{1}{2}}\bar{B}_{\theta,\frac{1}{2}}\left(\frac{4\Delta V_0}{\pi\Delta r^2}-1\right)\bar{E}_{z,0}, \tag{19}$$

where N_r is the index for the upper boundary of the radial domain where $r_i=R$. The first term on the right hand side of Eq. (19) is the surface integral of the discrete Poynting flux at the upper radial boundary. The last term is the discrete value at $r=0$. Physically, this term is identically zero. However, since $\bar{B}_{\theta,1/2}$ and $\bar{E}_{z,0}$ are both non-zero in general, achieving this numerically requires a specific definition for the volume element on axis, ΔV_0. Setting the term inside the parentheses to zero, the definition of the on-axis volume element is

$$\Delta V_0=\frac{\pi\Delta r^2}{4}=\int_0^{\Delta r/2}2\pi r dr, \tag{20}$$

which is the total volume per unit length of a cylinder with radius $\Delta r/2$. Note that this value differs by a factor of 3/4 from that recommended by Verboncoeur in Ref. [26] for the on axis volume element in cylindrical coordinates, which is formulated such that a uniform particle distribution is preserved. For an electromagnetic algorithm, using the volume element from Ref. [26] in cylindrical coordinates results in a non-vanishing on-axis Poynting flux.

Similarly, the Poynting flux term in Eq. (18) for 1D spherical coordinates becomes

$$\sum_g\nabla_\xi\cdot\bar{\boldsymbol{\mathcal{S}}}_g^{\xi}\Delta V_g=\frac{4\pi R^2}{\mu_0}\left[\frac{\left(r_{N_r+\frac{1}{2}}\bar{B}_{\theta,N_r+\frac{1}{2}}+r_{N_r-\frac{1}{2}}\bar{B}_{\theta,N_r-\frac{1}{2}}\right)}{2R}\bar{E}_{\phi,N_r}-\frac{\left(r_{N_r+\frac{1}{2}}\bar{B}_{\phi,N_r+\frac{1}{2}}+r_{N_r-\frac{1}{2}}\bar{B}_{\phi,N_r-\frac{1}{2}}\right)}{2R}\bar{E}_{\theta,N_r}\right]. \tag{21}$$

In contrast to cylindrical coordinates, the Poynting flux term is identically zero on axis independent of the on-axis volume element. This is because the θ and ϕ components of the electric field are identically zero on axis. For consistency, the on-axis volume element used in spherical coordinates here to compute the charge and current density on axis is obtained in the same way as it is for cylindrical coordinates:

$$\Delta V_0 = \int_0^{\Delta r/2} 4\pi r^2 dr = \frac{4\pi}{3}\left(\frac{\Delta r}{2}\right)^3. \tag{22}$$

The energy in the particles at time step n is given as

$$W^n_{parts} = \sum_s \sum_{p\in s} \frac{m_p}{2}\left|\mathbf{u}^n_p\right|^2. \tag{23}$$

The change in energy in the particles after a single PIC advance (assuming no particles leave or enter the domain) is [9,14]

$$W^{n+1}_{parts} - W^n_{parts} = \sum_s \sum_{p\in s} m_p \bar{\mathbf{u}}_p \cdot \left(\mathbf{u}^{n+1}_p - \mathbf{u}^n_p\right) = \sum_g \bar{\mathbf{J}}^{\xi}_g \cdot \bar{\mathbf{E}}^{\xi}_g \Delta V_g \Delta t. \tag{24}$$

The derivation of Eq. (24) requires that $S^{\mathbf{E}}_{gp} = S^{\mathbf{J}}_{gp}$, as mentioned previously. The change in total energy in the system, $W_{tot} = W_{fields} + W_{parts}$, when going from time step n to time step $n+1$ is obtained by combining Eqs. (17) and (24):

$$W^{n+1}_{tot} - W^n_{tot} + \sum_g \nabla_\xi \cdot \bar{\boldsymbol{\mathcal{S}}}^{\xi}_g \Delta V_g \Delta t = 0. \tag{25}$$

3.4. Shape factors and charge conservation

The shape factors used in this work are the same as those discussed in Ref. [13]. The in-plane current density is deposited to the grid using the first-order charge-conserving stencil by Villasenor and Buneman [27], which uses a combination of first-order and second-order splines and the deposit is done in segments that are determined by cell crossings locations [28]. This in-plane current deposition scheme preserves local charge density at the nodes defined using the second order spline. The out-of-plane components of the particle current are deposited to the grid using a first-order spline (e.g., cloud in cell) computed using the time-centered particle position. The magnetic field is interpolated to the particle using the same stencil. For energy conservation, the electric field used to advance the particle velocities is interpolated to the particle using the same component-wise shape factor as for the current deposit: $S^{\mathbf{E}}_{gp} = S^{\mathbf{J}}_{gp}$.

Charge-conserving shape factors are formulated such that the change in charge deposited to the nodes of the grid after time Δt is equal to the total charge that fluxed across the neighboring cell edges surrounding the node [27,29]. This can be expressed mathematically in 2D cylindrical as

$$\frac{Q^{n+1}_{i,j} - Q^n_{i,j}}{\Delta t} + \frac{\bar{I}_{r,i+1/2,j} - \bar{I}_{r,i-1/2,j}}{\Delta r} + \frac{\bar{I}_{z,i,j+1/2} - \bar{I}_{z,i,j-1/2}}{\Delta z} = 0, \tag{26}$$

where $Q^n_{i,j} \equiv \sum_p q_p S^{\rho}_r\left(r_i - r^n_p\right) S^{\rho}_z\left(z_j - z^n_p\right)$ is the charge, $\bar{I}_{r,i+1/2,j} \equiv \sum_p q_p \bar{u}_{pr} S^{J_r}_{i+1/2,j,p}$ is the logical space flux of charge in the radial direction, and $\bar{I}_{z,i,j+1/2} \equiv \sum_p q_p \bar{u}_{pz} S^{J_z}_{i,j+1/2,p}$ is the logical space flux of charge in the axial direction. It is essential that the logical particle velocity that appears in the flux definition be consistent with the change in logical particle position. That is, $\bar{u}_{pr} = \left(r^{n+1}_p - r^n_p\right)/\Delta t$ and $\bar{u}_{pz} = \left(z^{n+1}_p - z^n_p\right)/\Delta t$. The conservation of charge law given in Eq. (26) can be recast as a density law by dividing through the volume element at the nodes, which is $\Delta V_{i,j} = 2\pi r_i \Delta r \Delta z$ for 2D cylindrical. This gives

$$\frac{\rho^{n+1}_{i,j} - \rho^n_{i,j}}{\Delta t} + \frac{r_{i+1/2}\bar{J}_{r,i+1/2,j} - r_{i-1/2}\bar{J}_{r,i-1/2,j}}{r_i \Delta r} + \frac{\bar{J}_{z,i,j+1/2} - \bar{J}_{z,i,j-1/2}}{\Delta z} = 0. \tag{27}$$

This equation shows the discrete form of the divergence operator in 2D cylindrical geometry that is consistent with the change in charge density after time step Δt when using a charge-conserving shape factor. Likewise, via Gauss's law, the charge density on the grid, ρ^{n+1}_g, is consistent with $\nabla_\xi \cdot \epsilon_0 \mathbf{E}^{\xi,n+1}_g$ on the grid using this same discrete form for the divergence operator as used in Eq. (27).

Following the same procedure above for 2D cylindrical, one arrives at the following discrete form of the charge continuity law in 1D spherical geometry that is satisfied when using a charge-conserving current deposition scheme

$$\frac{\rho^{n+1}_i - \rho^n_i}{\Delta t} + \frac{r^2_{i+1/2}\bar{J}_{r,i+1/2} - r^2_{i-1/2}\bar{J}_{r,i-1/2}}{r^2_i \Delta r} = 0. \tag{28}$$

One may note that a singularity exists in Eqs. (27) and (28) right on axis where $r_{i=0} = 0$. The appropriate definition for the divergence operators on axis consistent with local charge conservation is given in Appendix B.

4. Orbit integrator

A Cartesian orbit integrator, where the particle velocities and positions are updated in Cartesian coordinates, is used in this work. This gives improved accuracy (especially near $r = 0$) over direct and hybrid orbit integrators. A direct method is one where the particle velocity and position are both integrated in logical coordinates. A hybrid method is where the particle velocity is advanced in Cartesian, but the position is advanced in logical coordinates. For energy-conserving PIC methods, orbit integrators that update the particle position in logical coordinates are naturally compatible with charge-conserving current depositions [19]. A standard implementation of a Cartesian update for the particle position, on the other hand, is not. It is shown here that updating the particle position in Cartesian coordinates can be made compatible with charge-conserving current depositions by using particular definitions for cosine and sine used to convert between logical and Cartesian coordinates.

Algorithm 1 Cartesian particle orbit integrator for cylindrical geometry.

1: Set Cartesian values at t_n using $\theta_p^n = 0$: $x_p^n = r_p^n$, $y_p^n = 0$, $u_{px}^n = u_{pr}^n$, and $u_{py}^n = u_{p\theta}^n$.
2: Initial guess for time-centered Cartesian and logical-space positions: $\bar{x}_p = \bar{r}_p = r_p^n$, $\bar{y}_p = y_p^n$, and $\bar{z}_p = z_p^n$.
Iterate the following steps to obtain a time-centered solution:
3: Interpolate the logical $\bar{\mathbf{E}}_g^{\xi}$ and $\bar{\mathbf{B}}_g^{\xi}$ fields on the logical-space grid to the particle using $\bar{r}_p$ and $\bar{z}_p$.
4: Transform $\bar{\mathbf{E}}_p^{\xi}$ and $\bar{\mathbf{B}}_p^{\xi}$ from logical to Cartesian using $\cos\bar{\theta}_p = \bar{x}_p/\bar{r}_p$ and $\sin\bar{\theta}_p = \bar{y}_p/\bar{r}_p$.
5: Compute the time-centered particle velocities in Cartesian using Eq. (13).
6: Update the particle positions in Cartesian and logical: $\mathbf{x}_p^{n+1} = \mathbf{x}_p^n + \bar{\mathbf{u}}_p \Delta t$, $r_p^{n+1} = \sqrt{\left(x_p^{n+1}\right)^2 + \left(y_p^{n+1}\right)^2}$.
7: Compute the time-centered radial positions as $\bar{r}_p = \left(r_p^{n+1} + r_p^n\right)/2$.
8: Compute the time-centered Cartesian positions: $\bar{\mathbf{x}}_p = \left(\mathbf{x}_p^{n+1} + \mathbf{x}_p^n\right)/2$.
Solution converged:
9: Transform Cartesian velocity at t_{n+1} to logical using $\cos\theta_p^{n+1} = x_p^{n+1}/r_p^{n+1}$ and $\sin\theta_p^{n+1} = y_p^{n+1}/r_p^{n+1}$.
10: Rebase θ_p^{n+1} to zero.

4.1. Cartesian particle orbit integrator

The Cartesian particle orbit integrator used in this work is presented in Algorithm 1. An important thing to note about Algorithm 1 is that there is no direct usage of the time-centered angle $\bar{\theta}_p$ during the iterative particle update to obtain the time-centered solution. Instead, the cosine and sine used to do the transformations are defined in terms of the time-centered Cartesian positions $\bar{x}_p$ and $\bar{y}_p$ and the time-centered radial position $\bar{r}_p \equiv \left(r_p^{n+1} + r_p^n\right)/2$. This is a subtle but important point because these definitions are needed to make the Cartesian position advance compatible with charge-conserving current deposition schemes. However, these definitions are atypical because $\cos^2\bar{\theta}_p + \sin^2\bar{\theta}_p \neq 1$. This method is described in more detail below considering cylindrical geometry. Spherical geometry is considered in a following subsection.

The electric and magnetic fields on the grid are advanced in logical space. After interpolating the logical fields on the grid to the particle position, the fields acting on the particle are transformed to Cartesian prior to advancing the particle in Cartesian coordinates (e.g., $\bar{E}_{px} = \cos\bar{\theta}_p \bar{E}_{pr} - \sin\bar{\theta}_p \bar{E}_{p\theta}$ and $\bar{E}_{py} = \sin\bar{\theta}_p \bar{E}_{pr} + \cos\bar{\theta}_p \bar{E}_{p\theta}$). The discrete equations governing a particle's position and velocity in physical space are given in Eq. (13). The particle's position and velocity, while advanced in Cartesian coordinates, are stored by the particle in logical space. Each particle is re-based back to $\theta_p = 0$ at the end of a time step, in which case the particle position and velocity at the beginning of the next step are such that $x_p^n = r_p^n$, $y_p^n = 0$, $u_{px}^n = u_{pr}^n$, and $u_{py}^n = u_{p\theta}^n$.

The energy conservation law for a single particle using this second-order time-centered advance is obtained by taking the scalar product of the discrete equation of motion and the time-centered velocity. For cylindrical geometry, this gives

$$\frac{\mathcal{E}_p^{n+1} - \mathcal{E}_p^n}{\Delta t} = q_p\left[\bar{u}_{px}\bar{E}_{px} + \bar{u}_{py}\bar{E}_{py} + \bar{u}_{pz}\bar{E}_{pz}\right] = q_p\left[\bar{u}_{pr}\bar{E}_{pr} + \bar{u}_{p\theta}\bar{E}_{p\theta} + \bar{u}_{pz}\bar{E}_{pz}\right], \tag{29}$$

where $\mathcal{E}_p \equiv m_p|\mathbf{u}_p|^2/2$ is the kinetic energy of the particle and the logical components of the time-centered velocity in the last expression are defined as

$$\bar{u}_{pr} \equiv \cos\bar{\theta}_p\bar{u}_{px} + \sin\bar{\theta}_p\bar{u}_{py} \text{ and } \bar{u}_{p\theta} \equiv -\sin\bar{\theta}_p\bar{u}_{px} + \cos\bar{\theta}_p\bar{u}_{py}. \tag{30}$$

To achieve exact energy conservation, the logical velocities defined in Eq. (30) must be used when depositing the logical current to the grid at the half time so that the discrete $\mathbf{J}\cdot\mathbf{E}$ term in the energy law for the fields is equal and opposite to that for the particles. Furthermore, if a charge-conserving current deposition scheme is being used, then the change in the particle's logical position must be consistent with the logical velocity used to deposit the current to the grid. For axisymmetric geometries, this means that $\left(r_p^{n+1} - r_p^n\right)/\Delta t$ must be equal to $\bar{u}_{pr}$ given in Eq. (30).

The change in logical position over the change in time for the Cartesian advance (see Algorithm 1) is

$$\frac{r_p^{n+1} - r_p^n}{\Delta t} = \frac{\left(r_p^{n+1}\right)^2 - \left(r_p^n\right)^2}{\Delta t\left(r_p^{n+1} + r_p^n\right)} = \frac{\left(x_p^n + \Delta t\bar{u}_{px}\right)^2 + \left(y_p^n + \Delta t\bar{u}_{py}\right)^2 - \left(r_p^n\right)^2}{2\Delta t\bar{r}_p},$$

$$= \frac{\left(x_p^n + \frac{1}{2}\Delta t \bar{u}_{px}\right)\bar{u}_{px} + \left(y_p^n + \frac{1}{2}\Delta t \bar{u}_{py}\right)\bar{u}_{py}}{\bar{r}_p},$$
$$= \frac{\bar{x}_p \bar{u}_{px} + \bar{y}_p \bar{u}_{py}}{\bar{r}_p}. \tag{31}$$

Thus, in order for Eq. (31) to be the same as $\bar{u}_{pr}$ given in Eq. (30), $\cos\bar{\theta}_p$ and $\sin\bar{\theta}_p$ should be defined as

$$\cos\bar{\theta}_p = \frac{\bar{x}_p}{\bar{r}_p}, \quad \sin\bar{\theta}_p = \frac{\bar{y}_p}{\bar{r}_p}. \tag{32}$$

The definitions given here in Eq. (32) are needed to make a Cartesian advance method compatible with both exact energy and charge-conserving methods. However, these definitions are atypical as the sum of the squares is

$$C_p \equiv \cos^2\bar{\theta}_p + \sin^2\bar{\theta}_p = \frac{\bar{x}_p^2 + \bar{y}_p^2}{\bar{r}_p^2} = 1 - \frac{2 r_p^n r_p^{n+1}}{\left(r_p^n + r_p^{n+1}\right)^2}(1 - \cos\Phi) \le 1, \tag{33}$$

where Φ is the angle between the physical particle position vector at step n and step $n+1$: $\cos\Phi = \mathbf{x}_p^n \cdot \mathbf{x}_p^{n+1} / \left(r_p^n r_p^{n+1}\right)$. The effects of using the expressions in Eq. (32) for transforming between Cartesian and logical coordinates in the particle advance can be analyzed by looking at the equation governing the time-centered logical velocity used in the current deposit. These equations are obtained by writing the discrete equations of motion given in Eq. (13) in terms of the time-centered Cartesian velocity and then transforming to the logical velocity using the expressions in Eq. (32). After some algebra, this gives

$$\bar{u}_{pr} = +\cos\bar{\theta}_p u_{px}^n + \sin\bar{\theta}_p u_{py}^n + \alpha_p\left[C_p \bar{E}_{pr} + \bar{u}_{p\theta}\bar{B}_{pz} - \bar{u}_{pz} C_p \bar{B}_{p\theta}\right], \tag{34}$$
$$\bar{u}_{p\theta} = -\sin\bar{\theta}_p u_{px}^n + \cos\bar{\theta}_p u_{py}^n + \alpha_p\left[C_p \bar{E}_{p\theta} + \bar{u}_{pz} C_p \bar{B}_{pr} - \bar{u}_{pr}\bar{B}_{pz}\right], \tag{35}$$
$$\bar{u}_{pz} = u_{pz}^n + \alpha_p\left[\bar{E}_{pz} + \bar{u}_{pr}\bar{B}_{p\theta} - \bar{u}_{p\theta}\bar{B}_{pr}\right], \tag{36}$$

where $\alpha_p \equiv \Delta t q_p/(2m_p)$ and the positive-definite coefficient $C_p \le 1$ is defined in Eq. (33). C_p is equal to unity to second order in Δt for particles not too close to axis such that $|\Delta t \bar{u}_{px}|$ and $|\Delta t \bar{u}_{py}|$ are small compared to r_p^n, but C_p can be less than unity for particles close to axis where the angle Φ in Eq. (33) can be relatively large. From Eqs. (34)-(36), it is seen that an effect of defining $\cos\bar{\theta}_p$ and $\sin\bar{\theta}_p$ as given in Eq. (32) is that the radial and theta components of the electric and magnetic fields in the radial and theta components of the force are damped by C_p for particles near axis. Each of these field quantities scale with $\bar{r}_p$ close to axis in axisymmetric geometries [30] (viz., these field quantities are small here anyway). The accuracy of the Cartesian particle integrator used in this work is quantified below using single particle orbit tests in static magnetic and electric fields.

4.2. Hybrid particle orbit integrator

Results from single particle orbit tests given in following subsections are compared with those obtained using a hybrid integrator [19]. The hybrid integrator is briefly described here. Using $x_p = r_p\cos\theta_p$ and $y_p = r_p\sin\theta_p$, the equations governing the logical coordinates r_p and θ_p are obtained:

$$\frac{dr_p}{dt} = u_{pr}, \quad \frac{d\theta_p}{dt} = \frac{u_{p\theta}}{r_p}, \tag{37}$$

where the corresponding logical components of the velocity vector are defined as

$$u_{pr} \equiv \frac{d\mathbf{x}_p}{dt}\cdot\hat{\mathbf{r}} = +\cos\theta_p u_{px} + \sin\theta_p u_{py}, \quad u_{p\theta} \equiv \frac{d\mathbf{x}_p}{dt}\cdot\hat{\boldsymbol{\theta}} = -\sin\theta_p u_{px} + \cos\theta_p u_{py}. \tag{38}$$

For hybrid integrators, the velocity is advanced in Cartesian coordinates, bypassing errors associated with under-resolving inertial forces for particles near axis, but the position advance is done in logical space using Eq. (37). Since the logical particle velocity is consistent with the change in logical particle position over a time step, energy-conserving PIC methods employing hybrid orbit integrators are naturally compatible with charge-conserving current depositions.

The time-centered angle from the discrete version of Eq. (37) used to transform between logical and physical coordinates is $\bar{\theta}_p = 0.5\Delta t \bar{u}_{p\theta}/\bar{r}_p$. Because there is no direct usage of $\bar{\theta}_p$ (only $\cos\bar{\theta}_p$ and $\sin\bar{\theta}_p$ are used) there are no numerical issues associated with large values of $\Delta t|\bar{u}_{p\theta}|/\bar{r}_p$ that can occur for particles near axis. However, large values of $|\bar{\theta}_p|$ are still a source of numerical error in the hybrid approach that doesn't exist in the Cartesian orbit integrator.

4.3. Orbit integrator accuracy tests in cylindrical geometry

The first orbit integrator test is for cylindrical geometry and considers a particle interacting with a stationary azimuthal magnetic field given by the Bennett equilibrium Z-pinch profile [30]: $B_\theta(r) = 2arB_0/\left(a^2 + r^2\right)$. $B_0 = 3.3356\times10^{-4}$ T and $a = 1.0$ cm are used for the tests. Two electron test particles (call them P1 and P2) are considered with different initial conditions. For P1, the initial

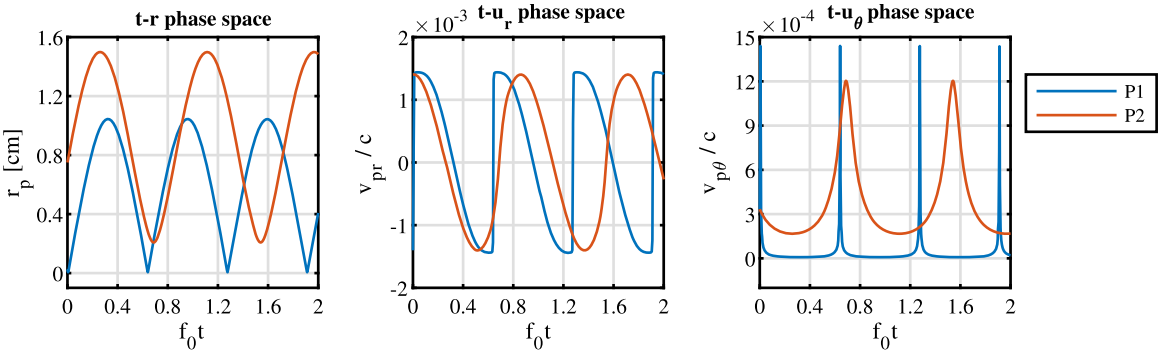

Fig. 2. Phase space plots for particles P1 and P2 in a stationary azimuthal magnetic field in cylindrical geometry. The magnetic field profile and the initial particle conditions are given in the text. $f_0 = eB_0/(2\pi m_e)$ is a characteristic angular frequency of the motion. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

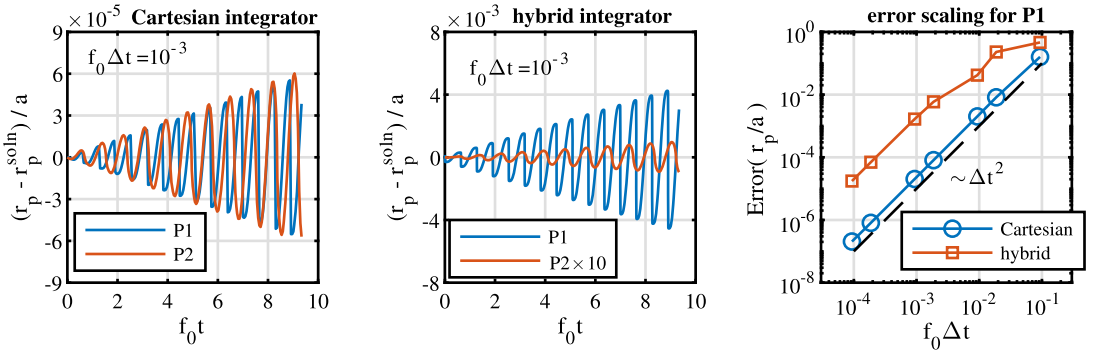

Fig. 3. Error analysis results for particles P1 and P2 in a stationary azimuthal magnetic field in cylindrical geometry using the Cartesian integrator (left panel) and the hybrid integrator (middle panel). Time profiles of the difference between the particle's radial position obtained from the numerical integration with respect to the semi-analytic result presented in Fig. 2 are shown. Note that the curve for P2 in the middle panel is enhanced by a factor of 10. The variation of an integrated measure of the error with time step as defined in Eq. (39) using $t_{\max} = 10/f_0$ is shown in the right panel for particle P1.

conditions are $r_p = 0.025$ cm, $u_{pr} = -4.2\times10^7$ cm/s, $u_{p\theta} = 1.0\times10^7$ cm/s, and $u_{pz} = 0$. The initial conditions for P2 are $r_p = 0.75$ cm, $u_{pr} = 4.2\times10^7$ cm/s, $u_{p\theta} = 1.0\times10^7$ cm/s, and $u_{pz} = 0$.

The particle energy and angular momentum are both constants of motion in this test. Semi-analytic solutions for the particle orbits, shown in Fig. 2, are obtained using an RK4 time-integrator with a sufficiently small time step such that energy and angular momentum are conserved to machine precision. Both particles undergo cyclotron motion with a characteristic angular frequency $f_0 = eB_0/(2\pi m_e)$, but the minimum radial position of P1 is much closer to axis than that for P2. Due to conservation of angular momentum and energy, one sees a sharp spike in the magnitude of $u_{p\theta}$ as the particle approaches the axis. This is accompanied by a sharp transition in the radial velocity u_{pr} from a negative to a positive value.

Error analysis results using the Cartesian and hybrid particle integrators for this test are shown in Fig. 3. Time profiles of the difference between the particles radial position with the semi-analytic solution obtained using the Cartesian integrator are shown in the left panel of this figure. Analogous results obtained using the hybrid integrator are shown in the middle panel. These results are obtained using $f_0\Delta t = 10^{-3}$. For the Cartesian integrator, the radial position error profiles for P1 and P2 are quantitatively similar. For the hybrid integrator, the radial position error profile for P1 is larger than that for P2 by several orders of magnitude. The magnitude of the error for particle P2 for the hybrid integrator is within a factor of order unity of that for the Cartesian integrator. However, the magnitude of the error for particle P1, which gets much closer to axis than P2, for the hybrid integrator is much larger than that for the Cartesian integrator.

The results in the left and middle panels of Fig. 3 illustrate the benefit of using a Cartesian integrator when the axis is included in the simulation domain. The error in particle orbits that get close to axis can be relatively large when using the hybrid integrator, whereas the error in particle orbits is relatively unaffected by its proximity to the axis when using a Cartesian integrator. An integrated measure of the error for the orbit integrators can be defined as

$$\text{Error}\left(r_p\right) \equiv \frac{1}{N_t}\sum_{n=1}^{N_t}\sqrt{\left(r_p^n - r_p^{\text{soln}}(t_n)\right)^2}, \tag{39}$$

where $N_t = t_{\max}/\Delta t$ is the total number of time steps taken to reach time $t_{\max}$. The scaling of this global measure of the error with Δt for the Cartesian and hybrid integrators is shown in the right panel of Fig. 3 for particle P1. Both integrators show second order

accuracy for sufficiently small time steps, but the magnitude of the error for the Cartesian integrator is about a factor of 100 smaller than that for the hybrid integrator.

4.4. Orbit integrator accuracy tests in spherical geometry

The Cartesian orbit integrator described above was for cylindrical geometry. It is straightforward to do the same for spherical geometry. Here we present the relevant equations for spherical geometry and show results from a single particle orbit test. The energy conservation law for a single particle in spherical geometry with the Cartesian orbit integrator, found by dotting the discrete equation of motion by the time-centered velocity, is

$$\frac{\mathcal{E}_p^{n+1} - \mathcal{E}_p^n}{\Delta t} = q_p \left[\bar{u}_{px}\bar{E}_{px} + \bar{u}_{py}\bar{E}_{py} + \bar{u}_{pz}\bar{E}_{pz}\right] = q_p \left[\bar{u}_{pr}\bar{E}_{pr} + \bar{u}_{p\theta}\bar{E}_{p\theta} + \bar{u}_{\phi}\bar{E}_{p\phi}\right], \tag{40}$$

where the components of the time-centered logical velocity used in the last expression are

$$\bar{u}_{pr} = +\cos\bar{\phi}_p \cos\bar{\theta}_p \bar{u}_{px} + \cos\bar{\phi}_p \sin\bar{\theta}_p \bar{u}_{py} + \sin\bar{\phi}_p \bar{u}_{pz}, \tag{41}$$

$$\bar{u}_{p\theta} = -\sin\bar{\theta}_p \bar{u}_{px} + \cos\bar{\theta}_p \bar{u}_{py}, \tag{42}$$

$$\bar{u}_{p\phi} = -\sin\bar{\phi}_p \cos\bar{\theta}_p \bar{u}_{px} - \sin\bar{\phi}_p \sin\bar{\theta}_p \bar{u}_{py} + \cos\bar{\phi}_p \bar{u}_{pz}. \tag{43}$$

Exact energy conservation demands that the logical velocities as defined in Eqs. (41)-(43) are used when depositing the current to the grid. For exact charge conservation in 1D spherical geometry, the change in radial position over Δt must be equal to $\bar{u}_{pr}$. The change in radial position when using the Cartesian integrator in spherical coordinates is

$$\begin{aligned}
\frac{r_p^{n+1} - r_p^n}{\Delta t} &= \frac{\left(r_p^{n+1}\right)^2 - \left(r_p^n\right)^2}{\Delta t\left(r_p^{n+1} + r_p^n\right)} = \frac{\left(x_p^n + \Delta t\bar{u}_{px}\right)^2 + \left(y_p^n + \Delta t\bar{u}_{py}\right)^2 + \left(z_p^n + \Delta t\bar{u}_{pz}\right)^2 - \left(r_p^n\right)^2}{2\Delta t\bar{r}_p}, \\
&= \frac{\left(x_p^n + \frac{1}{2}\Delta t\bar{u}_{px}\right)\bar{u}_{px} + \left(y_p^n + \frac{1}{2}\Delta t\bar{u}_{py}\right)\bar{u}_{py} + \left(z_p^n + \Delta t\bar{u}_{pz}\right)^2}{\bar{r}_p}, \\
&= \frac{\bar{x}_p\bar{u}_{px} + \bar{y}_p\bar{u}_{py} + +\bar{z}_p\bar{u}_{pz}}{\bar{r}_p}.
\end{aligned} \tag{44}$$

Thus, in order for Eq. (44) to be the same as $\bar{u}_{pr}$ given in Eq. (41), the cosines and sines should be defined as

$$\cos\bar{\theta}_p = \frac{\bar{x}_p}{\sqrt{\bar{x}_p^2 + \bar{y}_p^2}}, \quad \sin\bar{\theta}_p = \frac{\bar{y}_p}{\sqrt{\bar{x}_p^2 + \bar{y}_p^2}}, \quad \cos\bar{\phi}_p = \frac{\sqrt{\bar{x}_p^2 + \bar{y}_p^2}}{\bar{r}_p}, \quad \sin\bar{\phi}_p = \frac{\bar{z}_p}{\bar{r}_p}. \tag{45}$$

Here, the azimuthal angle is well defined ($\cos^2\bar{\theta} + \sin^2\bar{\theta} = 1$) but the cosine and sine of the polar angle are such that

$$\cos^2\bar{\phi}_p + \sin^2\bar{\phi}_p = \frac{\bar{x}_p^2 + \bar{y}_p^2 + \bar{z}_p^2}{\bar{r}_p^2} \leq 1. \tag{46}$$

The accuracy of the Cartesian orbit integrator in 1D spherical geometry is examined by considering an electron in a static radial electric field give as $E_r = E_0\left[1 - \exp\left(-r^3/a^3\right)\right] / \left(r^2/a^2\right)$ with $E_0 = 1$ kV/cm and $a = 0.01$ cm. This electric field profile corresponds to a charge density with radial profile $\rho(r) \sim \exp\left(-r^3/a^3\right)$. The initial conditions of the electron are: $r_p = 0.025$ cm, $u_{pr} = -4.2\times10^7$ cm/s, $u_{p\theta} = 1.0\times10^7$ cm/s, and $u_{p\phi} = 2.0\times10^7$ cm/s. A semi-analytic solution to the phase space orbit is obtained using an RK4 time integration method with a small time step. These results are shown in Fig. 4, where the phase space plots illustrate a particle undergoing harmonic motion in the radial electric field.

The numerical error of the Cartesian and hybrid integrators in spherical geometry for this test is quantified in Fig. 5 by looking at time profiles of the error in the radial position of the particle, computed using the results from the RK4 integrator as the solution. Both integrators show second order accuracy, but the amplitude of the error from the Cartesian integrator is about a factor of 6 smaller.

5. Solver implementation

The fully implicit field-particle system is advanced in time using the particle-suppressed JFNK method [14,15] as described in [13]. In this implementation, the particle positions and the magnetic field in the Lorentz force are held fixed during the linear iterations. This permits using the mass matrices, a central feature in energy-conserving semi-implicit PIC algorithms [17,18], to compute the current density during the linear iterations in place of a direct calculation from the particles. The particles are self-consistently updated at each nonlinear iteration using a Picard method, as described in Algorithm 1, with the current values of the fields on the grid.

The algorithm is implemented in PICNIC, an MPI-based code for particle methods that utilizes the Chombo library for data containers and MPI handling [31]. The preconditioned JFNK method is implemented using the SNES/KSP/PC modules in PETSc

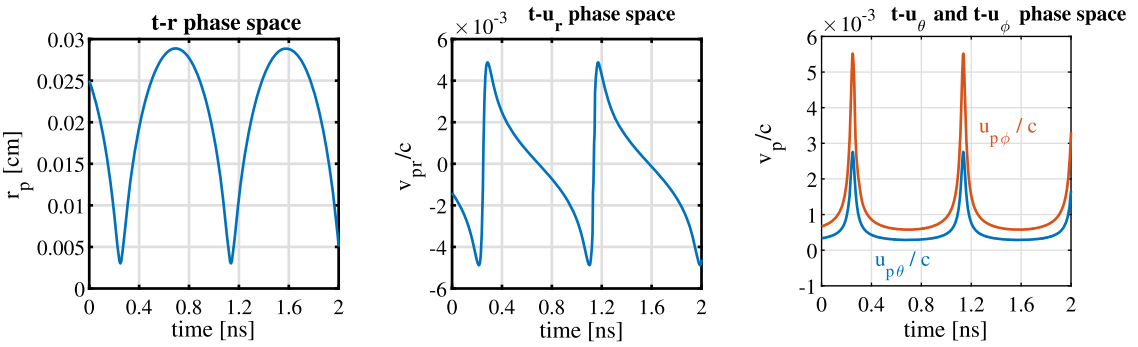

Fig. 4. Phase space profiles for an electron interacting with a static radial electric field in a 1D spherical geometry. These results are obtained using an RK4 time integration method with $\Delta t = 10^{-6}$ ns. The electric field profile and the initial conditions of the electron are given in the text.

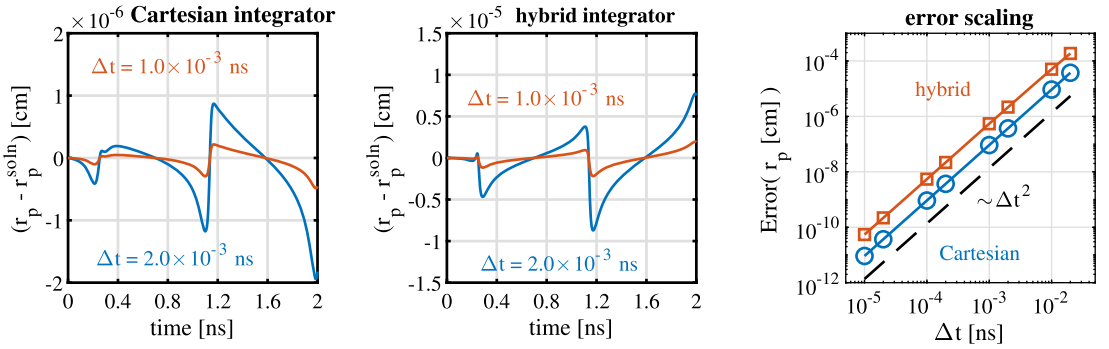

Fig. 5. Error analysis results for an electron in a uniform radial electric field in spherical geometry using the Cartesian integrator (left panel) and the hybrid integrator (middle panel). Time profiles of the difference between the particle's radial position obtained from the numerical integration with respect to the semi-analytic result presented in Fig. 4 are shown. The variation of an integrated measure of the error with time step as defined in Eq. (39) using $t_{\max} = 2$ ns is shown in the right panel.

[32]. The preconditioner matrix consists of full Maxwell's equations and the diagonal elements of the mass matrices [13]. The preconditioned matrices are solved using the Additive Schwarz method (ASM) [33] with local incomplete LU (ILU) [34] on each block. Each MPI rank constitutes one block.

The fields are advanced in logical space, while the particles are advanced in Cartesian. The mass matrices relate the logical space particle velocity vector to the logical space fields. Thus, the appropriate form of the mass matrices must be computed considering these transformations as used in the particle integrator. The non-trivial definitions for the cosines and sines used in this work to transform between logical and physical space result in the coefficient C_p (Eq. (33)) appearing in some of the matrix kernels. For cylindrical coordinates, the relationships between the mapped velocities used in the current deposition and the mapped fields are given in Eqs. (34)-(36). The corresponding relationships in spherical coordinates are

$$\bar{u}_{pr} = \tilde{u}^n_{pr} + \alpha_p \left[C_p \bar{E}_{pr} + \bar{u}_{p\theta} C_p \bar{B}_{p\phi} - \bar{u}_{p\phi} \bar{B}_{p\theta} \right], \tag{47}$$

$$\bar{u}_{p\theta} = \tilde{u}^n_{p\theta} + \alpha_p \left[\bar{E}_{p\theta} + \bar{u}_{pz} \bar{B}_{pr} - \bar{u}_{pr} \bar{B}_{p\phi} \right], \tag{48}$$

$$\bar{u}_{p\phi} = \tilde{u}^n_{p\phi} + \alpha_p \left[C_p \bar{E}_{p\phi} + \bar{u}_{pr} \bar{B}_{p\theta} - \bar{u}_{p\theta} C_p \bar{B}_{pr} \right], \tag{49}$$

where here $C_p \equiv \left(\bar{x}_p^2 + \bar{y}_p^2 + \bar{z}_p^2 \right) / \bar{r}_p^2$, and $\tilde{\mathbf{u}}^q_p = \mathbb{M}\mathbf{u}^n_p$ is the velocity vector at step n mapped to logical space using the time-centered cosines and sines. For spherical geometry, the coefficient C_p acts to damp the radial and polar components of the fields in the radial and polar components of the force for particles near $r = 0$. Like in cylindrical geometry, each of these fields being damped are small near axis anyway.

6. Free expansion of a sphere

The free diffusion of a collisionless sphere of plasma is a common test problem used to verify a PIC algorithm in cylindrical geometry [18]. This test problem is considered here for 2D cylindrical and 1D spherical geometries. The physical parameters for the simulations are the same as those considered in Ref. [18]. The plasma consists of electrons with mass m_e and singly ionized light ions with mass $m_i = 25 m_e$. The plasma sphere has an initial homogenous density for each species of $n_0 = 1.0{\times}10^{22}$ /m^3. The particle velocities are initialized by sampling from a non-drifting, isotropic Maxwellian distribution with $T_0 = 5.11$ keV. The ion plasma period for these initial conditions is $1/\omega_{pi} = 8.863{\times}10^{-13}$ s. The corresponding ion skin depth is $\delta_i \equiv c/\omega_{pi} = 2.657{\times}10^{-4}$ m. The corresponding electron Debye length is $\lambda_D \equiv V_{T,e}/\omega_{pe} = \delta_i/50$. The initial radius of the plasma sphere is $R = 1\delta_i = 50\lambda_{De}$.

Table 1
Simulation parameters for the free expansion of a sphere simulations.

	# cells	$\Delta r/\delta_i$	$\Delta z/\delta_i$	$\omega_{pi}\Delta t$	N_p
Case A (SPH)	25	0.2	–	0.4	6.600×10^4
Case D (SPH)	250	0.02	–	0.04	6.666×10^5
Case A (CYL)	25×50	0.2	0.2	0.4	6.726×10^4
Case D (CYL)	250×500	0.02	0.02	0.04	6.001×10^6

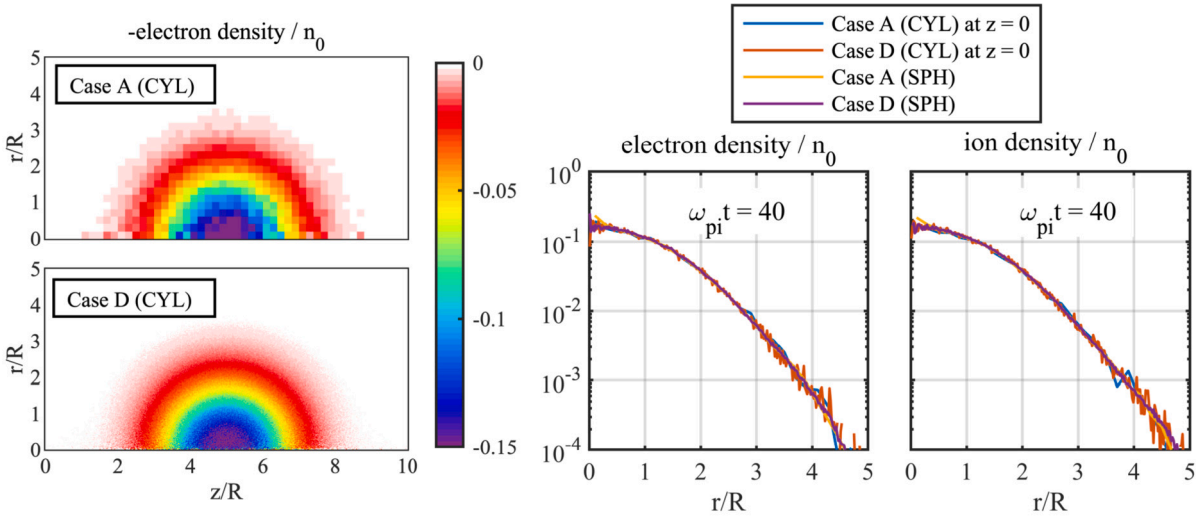

Fig. 6. Density profiles from free expansion of a collisionless plasma sphere simulations in 2D cylindrical geometry and 1D spherical geometry. The results shown are at $\omega_{pi}t = 40$.

The radial domain of the simulation grid ranges from the axis at $r = 0$ to $r = 5R$, which is treated as a perfect electrical conductor. Particles crossing this boundary are absorbed by the boundary. The z-direction for the simulations in cylindrical geometry ranges from $z = 0$ to $z = 10R$. Periodic boundary conditions are assumed for the z-direction. Results will be shown from simulations using a low-resolution setup (Case A) and a hi-resolution setup (Case D). The parameters used correspond to Case A and Case D from Ref. [18]. The grid parameters, time step, and total number of simulation particles per species (N_p) used for these simulations are given in Table. 1. Uniform particle weights are used in these simulations. The relative tolerances for the Newton and GMRES solvers are set to 2×10^{-14} and 10^{-6}, respectively. A value of 10^{-12} is used for the relative tolerance of the Picard solver for the particles.

Density profiles at $\omega_{pi}t = 40$ from the free expansion of a sphere simulations are shown in Fig. 6. Electron charge density contours are shown in the left Column. These results can be compared with those shown in Fig. 9 and Fig. 11 of Ref. [18]. Lineouts of the electron and ion density profiles at $z = 0$ from the 2D cylindrical simulations are shown in the right figures and observed to agree with the corresponding density profiles from the 1D spherical simulations.

The more mobile electrons in this problem initially expand radially outward faster than the heavier ions leading to charge separation. This sets up an ambipolar electric field that acts to decrease the electron flux and increase the ion flux until the net flux of charge is zero. This electric field transfers energy from the electrons to the ions, causing the speed distribution of electrons to shift towards lower values and that of the ions to shift towards larger values. The electron and ion speed distributions from the simulations are shown in Fig. 7. The distributions shown here are computed at $\omega_{pi}t = 40$ and include all simulation particles. These results can be compared against those in Fig. 16 of Ref. [18]. Again, the results from the 2D cylindrical simulations agree with those from the 1D spherical simulations.

The agreement between 2D cylindrical geometry and 1D spherical geometry simulations of the free expansion of a sphere shown in Figs. 6-7 serve as a verification of both axisymmetric models. A further verification of these models is shown in Fig. 8, where time profiles of the violation in energy conservation and a measure for the violation in Gauss's law are shown. The conservation of local charge density is measured by taking the L2 norm of the violation in Gauss's law in the grid, defined as

$$\mathrm{Err}\,(\delta\rho) \equiv \sqrt{\frac{\sum_g |\nabla_\xi \cdot \epsilon_0 \mathbf{E}_g^\xi - \rho_g|^2}{N_g}}, \tag{50}$$

where the sum is over all N_g grid cells. Global energy and local charge density are both conserved within expectations based on the relative tolerances used for the nonlinear solvers for these simulations.

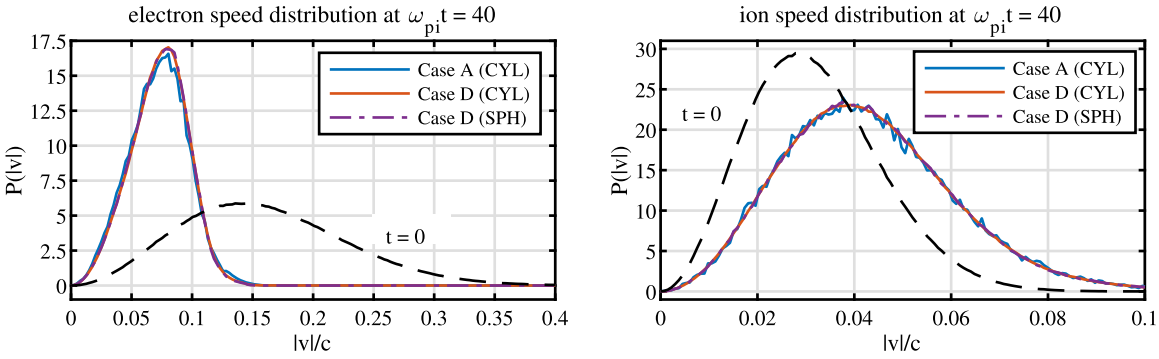

Fig. 7. Electron (left) and ion (right) speed distributions from free expansion of a collisionless plasma sphere simulations in 2D cylindrical geometry and 1D spherical geometry. The results shown are at $\omega_{pi}t = 40$.

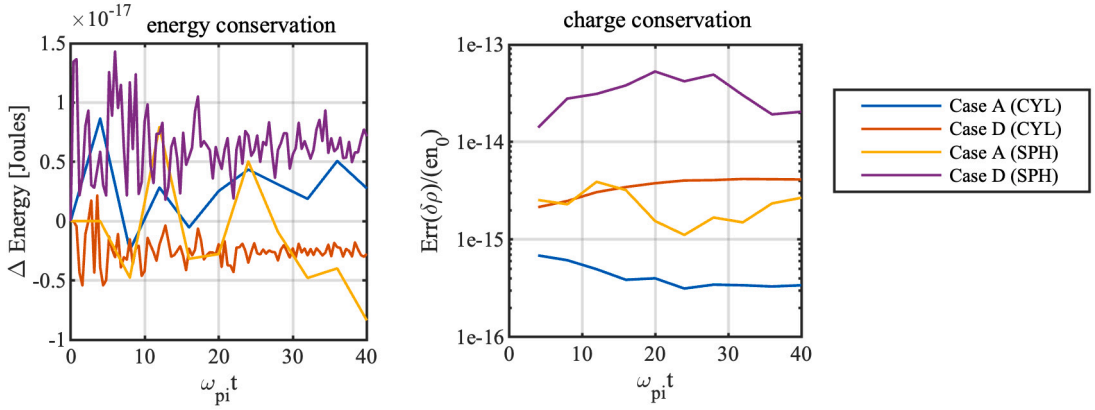

Fig. 8. Measures of global energy conservation (left) and local charge conservation (right) from the free expansion of a sphere simulations as given by Eq. (50). The violation in energy conservation is the change in total energy in the system with respect to the initial energy (≈ 0.002 J) and accounts for the energy of particles that are absorbed at the upper r-boundary of the domain.

7. Dynamic pinch simulations in 1D cylindrical geometry

The dynamic compression of a plasma to a high energy density state is a phenomenon that is typical of both DZP and ICF experiments. The planar analog of the dynamic pinch [35,16] is used in Ref. [13] as a surrogate problem to quantify the algorithm's ability to study the dynamic compression of a plasma. Here, we consider the same problem but in 1D cylindrical geometry. The initial conditions of the atomic deuterium plasma and the magnetic field at the upper boundary driving the compression are precisely the same as used in Ref. [13]: $n_0 = 1.0\times10^{17}/\text{cm}^3$, $T_0 = 1$ eV, $R_{p0} = 1.5$ cm. The magnetic field at the upper boundary of the simulation domain ($r = 1.54$ cm) rises linearly from 0 to $B_\theta = 2.667$ T over 8.154 ns and remains constant for later times. For these simulations, the relative tolerances are set to 10^{-10}, 10^{-4}, and 10^{-10}, for the Newton, GMRES, and Picard solvers, respectively.

Density and temperature profiles from simulations in a 1D cylindrical geometry using $N_r = 432$ and $\Delta t = 10^{-3}$ ns are shown in Fig. 9. The time slices shown in the left and middle panels of this figure are representative of the compression and stagnation stages of the dynamic Z-pinch [16]. Note that the magnitude of the plasma density and temperature at stagnation are about an order of magnitude larger in cylindrical geometry compared to that for a planar pinch [13].

The conditions used for the dynamic pinch simulation here are the same as the most collisional case considered for parameter set 4 from Ref. [8], where 1D dynamic pinch simulations were performed using the direct implicit PIC method [36] as implemented in the Chicago code [37]. An interesting comparison to make between the results of that work and those presented here is the compression ratio of the plasma at stagnation. Here, the compression ratio is seen from the profiles at stagnation in Fig. 9 to be $1.5/0.3 = 5$. The compression ratio for these conditions reported in Ref. [8] is 6. The fact that the compression ratio found in Ref. [8] is larger than that found here is consistent with what can be expected for an algorithm that suffers from cumulative numerical energy loss, which is the case for the algorithm used in Ref. [8]. In that work, it is stated that the numerical energy loss at stagnation is around 5% of the total energy in the plasma at that time. Artificial numerical cooling acts as an effective radiative loss of energy, limiting the ability of the plasma to push back against the magnetic piston resulting in an artificially large compression ratio.

Representative plots used to illustrate the performance of the dynamic pinch simulation in cylindrical geometry are shown in Fig. 10. The maximum value of $\omega_{pe}\Delta t \sim \sqrt{n_e}$ on the grid is shown in the left panel. This quantity peaks at the time of stagnation ($t = 128$ ns). The plasma period is a fundamental time scale that an implicit PIC algorithm should be capable of efficiently stepping over. The efficiency of the solver is illustrated in the middle panel of Fig. 10 where the running average number of Newton iterations

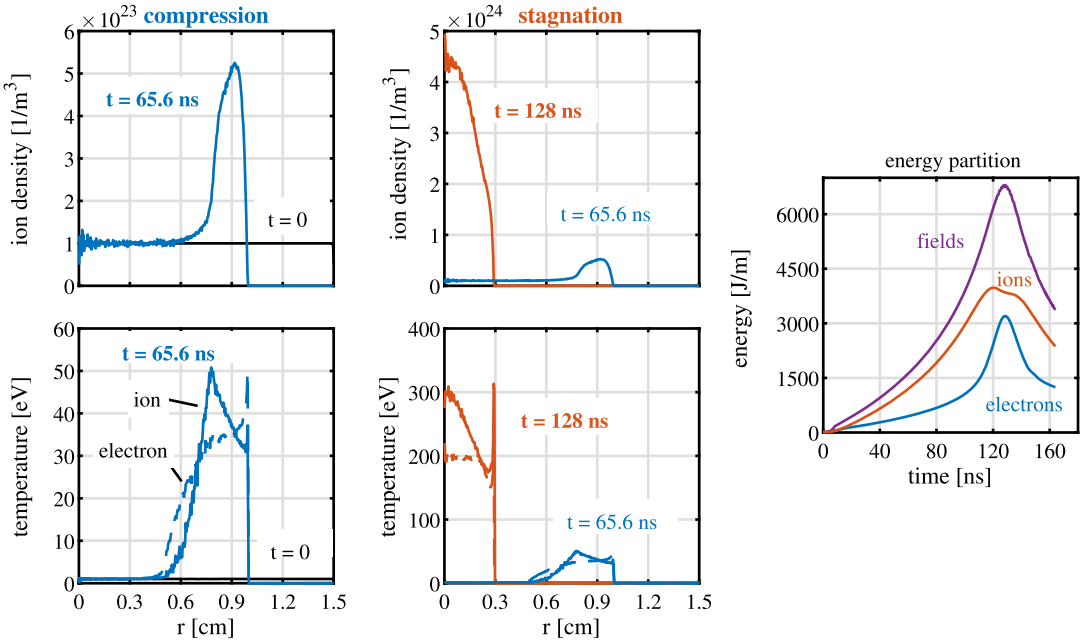

Fig. 9. Ion density (top) and species temperature (bottom) profiles from a dynamic pinch simulation in 1D cylindrical geometry. The figures on the left show the profiles at a time during the compression. Profiles at the time of stagnation are shown in the middle panels. Time profiles for energy in the fields, ions, and electrons are shown in the right panel.

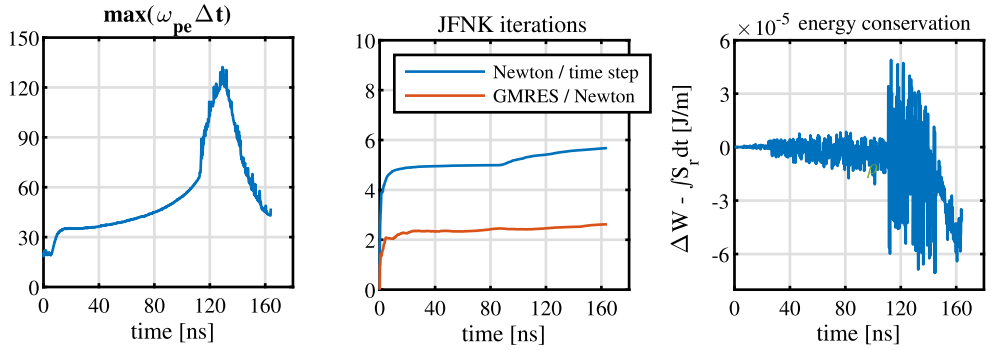

Fig. 10. Select performance metrics for the dynamic pinch simulation in 1D cylindrical geometry. These results correspond to the results shown in Fig. 9.

per time step and GMRES iterations per Newton iteration are shown. The Newton iterations range from 5 to 6 and the GMRES iterations are between 2 and 3. The low number of GMRES iterations shows the effectiveness of the preconditioner matrix and solver used in this work. The violation in energy conservation shown in the right panel of Fig. 10 is seen to remain eight orders of magnitude smaller than the peak energy in the system shown in the right panel of Fig. 9.

7.1. Dynamic pinch simulations with weighted particles

Equally weighted particles are used for the simulation corresponding to the results shown in Figs. 9-10. Since the initial density profile is uniform in the r-coordinate, the initial number of particles per cell increases linearly from the axis. In axisymmetric geometries, using equally weighted particles can be inefficient. Using the uniform density case as an example, the total number of particles in the simulation scales quadratic with the number of grid points N_r for equally weighted particles. However, it is well-known that standard Monte Carlo collision algorithms for Coulomb collisions [24,38] are only exactly energy and momentum conserving when the particles have equal weights. For weighted particles, these methods only conserve energy and momentum on average after many collisions [39,40].

Some of the authors have recently proposed a new particle method for doing weighted-particle Coulomb collisions in a plasma that 1) extends the standard methods by Takizuka and Abe [24] and Nanbu [38] to weighted particles such that the scattering physics is correct on average and 2) makes use of a new method for adjusting the particle velocities post scatter to restore exact momentum and energy conservation. The energy-correction method is based on inelastic scattering physics and works by rescaling the weighted-

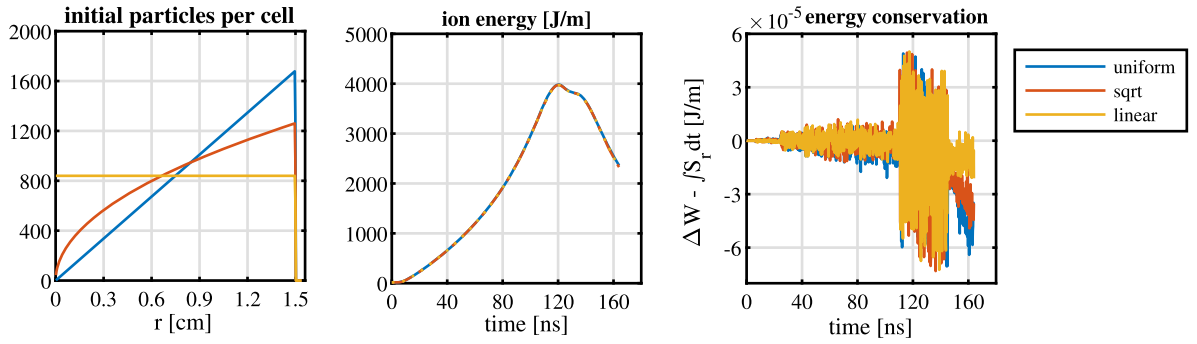

Fig. 11. Results from dynamic pinch simulations in 1D cylindrical geometry for various particle weighting strategies at initialization. The blue curves are from simulations with uniform particle weights. The red curves are from simulations where the particle weights scale with $\sqrt{r}$. The yellow curves are from simulations where the particle weights scale linearly with r. The radial profiles of the initial particles per cell shown in the left panel are per species.

center-of-mass energy of binary particle pairs by some small, specified fraction f_E. This value should be small to not disrupt the scattering physics, but a smaller value necessitates more binary pairs to correct for the violation in energy conservation. For example, assume that the particle energy in a cell is 2.01 Joules before scattering and is 2.0 Joules after the weighted-scattering algorithm has been applied and momentum conservation has been restored. Also, for simplicity, assume that the center-of-mass energy for each binary pair is 0.01 Joules and that there are 200 such pairs. Allowing the center-of-mass energy to change by at most 2%, then 50 binary pairs are need to restore energy conservation and the center-of-mass energy for each of these 50 pairs increases from 0.01 Joules to 0.0102 Joules.

Results from dynamic pinch simulations in 1D cylindrical geometry using weighted-particles are shown in Fig. 11. For these simulations, the weighted-particle Coulomb collision method from Ref. [41] is used and $f_E = 0.02$ is used for the energy correction factor. The simulation setup and physical parameters used for these simulations are the same as those corresponding to the results shown in Figs. 9-10. The only change is that the particles are weighted, which also effects the initial number of particles per cell. Two different weighted particle simulations are performed. The weights are initialized based on a square root scaling with the radial coordinate r for one simulation (red curves) and a linear scaling with r for the other (yellow curves). The left panel of Fig. 11 shows spatial profile of the initial number of particles per cell for each species for these simulations. The blue curve is for uniform weighted particles, in which case the number of particles per cell increases linear with r. At the other end of the spectrum is the linear weighting scheme, where the number of particles per cell at $t = 0$ is uniform in r. For each of these three simulations, the total number of particles in the simulation are approximately the same with N_p between 353 k and 358 k for each species.

Time profiles of the energy in the ions and the violation in energy conservation are shown in the middle and right panels, respectively, of Fig. 11. The ion energy curves are on top of each other, and the violation in energy conservation is 8 orders of magnitude lower for each of these simulations. These results illustrate that ability to use weighted-particles and maintain exact energy conservation, while also capturing the correct physics.

8. Discussion and summary

8.1. Further discussion on the Cartesian particle orbit integrator

The physical-space particle orbit integrator presented in this work uses non-trivial definition of cosine and sine to transform the fields from logical to physical space. The condition used to define cosine and sine is that the change in radial position over the change in time is equal to the time-centered radial velocity that appears in the particle energy law. Mathematically speaking, there are two unknowns, but use is only made of one constraint and thus the obtained solution is not unique. This is permissible because of the axisymmetric nature of the geometries considered in this work. However, if we consider non-axisymmetric geometries, such as 3D cylindrical or 2D polar, then one would additionally need to ensure that the change in angular position of the particle over the change in time is consistent with the expression for $u_{p\theta}$ given in Eq. (30) to maintain charge conservation. This second constraint produces the following unique solutions for cosine and sine:

$$\cos\bar{\theta}_p = \left(\bar{u}_{px}\frac{\Delta r_p}{\Delta t} + \bar{u}_{py}\bar{r}_p\frac{\Delta\theta_p}{\Delta t}\right) \Big/ \left(\bar{u}_{px}^2 + \bar{u}_{py}^2\right), \quad \sin\bar{\theta}_p = \left(\bar{u}_{py}\frac{\Delta r_p}{\Delta t} - \bar{u}_{px}\bar{r}_p\frac{\Delta\theta_p}{\Delta t}\right) \Big/ \left(\bar{u}_{px}^2 + \bar{u}_{py}^2\right). \tag{51}$$

The definitions for cosine and sine given previously in Eq. (32) are *sufficient* for exact energy and charge conservation for axisymmetric cylindrical geometries. Maintaining rigorous energy and charge conservation for non-axisymmetric cylindrical geometries *requires* using the definitions given in Eq. (51). One could potentially use the expressions in Eq. (51) for axisymmetric cylindrical geometries as well. One difference is that the sum of the squares for the expressions in Eq. (32) is equal to the square of the ratio of two different discrete definitions of the time-centered radial position; whereas the sum of the squares for the expressions in Eq. (51) equals the square of the ratio of two different definitions for the time-centered velocity magnitude. In either case, the sum of the squares is not identically one. For the axisymmetric method, it is shown that a consequence of this is to damp some components of the logical fields in the particle equations of motion (see Eqs. (34)-(36) and Eqs. (47)-(49)) for particles close to axis. However, each

of these fields being damped scale linearly with particle position near axis, and thus they are small anyway. The orbit integrator tests presented in Secs. 4 show that 2nd-order accuracy is maintained for particles near axis. Further mathematical analysis of these Cartesian particle orbit integrators, and potential application to non-axisymmetric geometries, is deferred to future work.

8.2. Summary

An implicit, collisional particle code with *exact* energy and charge conservation in axisymmetric cylindrical and spherical geometries is presented. The algorithm is an extension of the planar geometry algorithm presented in Ref. [13]. The volume element used on axis is derived based on that needed for exact energy and charge-conservation. A new Cartesian method for advancing particles is presented that is compatible with both exact energy conservation and charge-conserving current deposition schemes. The implementations of the 2D cylindrical and 1D spherical version of the algorithm are verified using simulations of the free expansion of a collisionless plasma sphere.

It is shown in Sec. 4 that the Cartesian-space particle orbit integrator introduced in this work yields improved accuracy compared to a standard hybrid integrator for particles near axis. The improved accuracy mainly results from not needing to resolve the angular motion of the particle. In principle, the accuracy of hybrid and direct integration methods can be improved for particles near axis by appropriately subcycling the orbit, but this requires infrastructure for determining when and how to subcycle (e.g., see Ref. [42]). Furthermore, the particle current must be deposited at each subcycle to maintain both energy and charge conservation, which requires modifications to the construction of the mass matrices [43].

The ability to use the algorithm presented in this manuscript to study the dynamic compression of a plasma in axisymmetric geometries is illustrated by simulating the dynamic Z-pinch in the collisional limit in 1D cylindrical geometry. This is similar to the planar analog of this problem that is considered in Ref. [13]. The algorithm can efficiently use time steps much larger than the plasma period and can use weighted particles while maintaining exact energy conservation [41]. The dynamic Z-pinch simulations presented here are in 1D cylindrical geometry using a specific set of physical parameters. In a future paper, the algorithm presented here will be used to study the dynamic Z-pinch for a wider range of physical parameters [8] in both 1D and 2D geometries.

CRediT authorship contribution statement

Justin Ray Angus: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **William Farmer:** Investigation. **Alex Friedman:** Investigation. **Vasily Geyko:** Investigation. **Debojyoti Ghosh:** Investigation. **Dave Grote:** Investigation. **David Larson:** Investigation. **Anthony Link:** Investigation.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Acknowledgements

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 and was supported by the LLNL-LDRD Program under Project No. 23-ERD-007.

Appendix A. Vector operators in spherical coordinates

The spherical coordinate system used in this work differs from that given in the NRL formulary [44]. For that reason, the divergence and curl operators for the general non-axisymmetric case are presented here. The mapping matrix is given as

$$\frac{\partial \mathbf{X}}{\partial \boldsymbol{\xi}} = \begin{bmatrix} \partial x/\partial r & \partial y/\partial r & \partial z/\partial r \\ \partial x/\partial \theta & \partial y/\partial \theta & \partial z/\partial \theta \\ \partial x/\partial \phi & \partial y/\partial \phi & \partial z/\partial \phi \end{bmatrix} = \begin{bmatrix} \cos\phi\cos\theta & \cos\phi\sin\theta & \sin\phi \\ -r\cos\phi\sin\theta & r\cos\phi\cos\theta & 0 \\ -r\sin\phi\cos\theta & -r\sin\phi\sin\theta & r\cos\phi \end{bmatrix}. \tag{A.1}$$

The rows of $\partial\mathbf{X}/\partial\boldsymbol{\xi}$ form the covariant basis vectors: $\mathbf{e}_0 = \hat{\mathbf{r}},\ \mathbf{e}_1 = r\cos\phi\hat{\boldsymbol{\theta}},\ \mathbf{e}_2 = r\hat{\boldsymbol{\phi}}$. The Jacobian of the transformation is $J \equiv \det(\partial\mathbf{X}/\partial\boldsymbol{\xi}) = r^2\cos\phi$. The inverse mapping matrix is given as

$$\frac{\partial \boldsymbol{\xi}}{\partial \mathbf{X}} = \begin{bmatrix} \partial r/\partial x & \partial\theta/\partial x & \partial\phi/\partial x \\ \partial r/\partial y & \partial\theta/\partial y & \partial\phi/\partial y \\ \partial r/\partial z & \partial\theta/\partial z & \partial\phi/\partial z \end{bmatrix} = \begin{bmatrix} \cos\phi\cos\theta & -\sin\theta/(r\cos\phi) & -\sin\phi\cos\theta/r \\ \cos\phi\sin\theta & \cos\theta/(r\cos\phi) & -\sin\phi\sin\theta/r \\ \sin\phi & 0 & \cos\phi/r \end{bmatrix}. \tag{A.2}$$

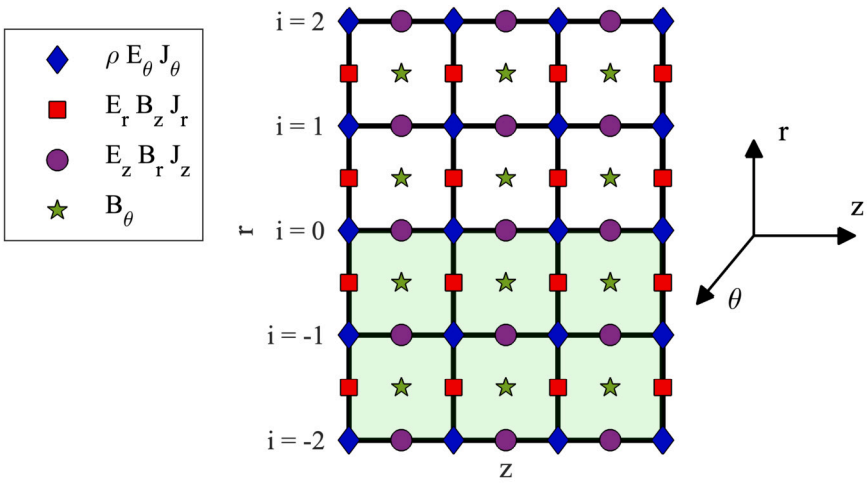

Fig. B.12. Yee grid diagram for 2D cylindrical geometry. The light green shaded region represents the ghost cell region below the axis. The varying markers show where different variables live on the grid as given in the legend.

The columns of $\partial\boldsymbol{\xi}/\partial\mathbf{X}$ form the contravariant basis vectors: $\mathbf{e}^0 = \hat{\mathbf{r}}$, $\mathbf{e}^1 = \hat{\boldsymbol{\theta}}/(r\cos\phi)$, $\mathbf{e}^2 = \hat{\boldsymbol{\phi}}/r$. The general expressions for the divergence and curl of some physical vector $\mathbf{F}$ are defined, respectively, as

$$\nabla\cdot\mathbf{F} \equiv \frac{1}{J}\frac{\partial}{\partial\boldsymbol{\xi}}\cdot\left(J\mathbf{F}\cdot\mathbf{e}^i\right), \quad \nabla\times\mathbf{F}\cdot\mathbf{e}^i \equiv \frac{1}{J}\left[\frac{\partial}{\partial\xi_j}\left(\mathbf{F}\cdot\mathbf{e}_k\right) - \frac{\partial}{\partial\xi_k}\left(\mathbf{F}\cdot\mathbf{e}_j\right)\right], \tag{A.3}$$

where the indices i, j, k in the expression for the curl are cyclic. That is, if $i = 0$, then $j = 1$ and $k = 2$. If $i = 1$, then $j = 2$ and $k = 0$. Using the mapping matrices defined in Eqs. (A.1)-(A.2) with the general definitions given in Eq. (A.3), the divergence and curl expressions for this spherical coordinate system are

$$\nabla\cdot\mathbf{F} = \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2F_r\right) + \frac{1}{r\cos\phi}\frac{\partial F_\theta}{\partial\theta} + \frac{1}{r\cos\phi}\frac{\partial}{\partial\phi}\left(\cos\phi F_\phi\right), \tag{A.4}$$

$$\nabla\times\mathbf{F} = \frac{1}{r\cos\phi}\left[\frac{\partial F_\phi}{\partial\theta} - \frac{\partial}{\partial\phi}\left(\cos\phi F_\theta\right)\right]\hat{\mathbf{r}} + \frac{1}{r}\left[\frac{\partial F_r}{\partial\phi} - \frac{\partial}{\partial r}\left(rF_\phi\right)\right]\hat{\boldsymbol{\theta}} + \frac{1}{r}\left[\frac{\partial}{\partial r}\left(rF_\theta\right) - \frac{1}{\cos\phi}\frac{\partial F_r}{\partial\theta}\right]\hat{\boldsymbol{\phi}}. \tag{A.5}$$

Appendix B. Treatments at the $r = 0$ axis of symmetry

The precise treatment of the axis for all aspects of the algorithm is described in this Appendix. Since a Cartesian method is used to advance the particles, the particles themselves never actually cross the axis and there are no issues for particles being infinitesimally close to the axis. The only possible way for a particle to live right at $r = 0$ is if $x_p = y_p = 0$ for cylindrical geometry and $x_p = y_p = z_p = 0$ for spherical geometry. If this rare situation is to occur, it is assumed that $\theta = \phi = 0$, in which case $\cos\theta = \cos\phi = 1$ and $\sin\theta = \sin\phi = 0$. A diagram of the Yee grid used in this work for 2D cylindrical geometry that include the axis is shown in Fig. B.12. This figure serves as a visual aid for the discussion below.

The interpolation of the fields from the grid to the particles will reach out to the ghost region below $r = 0$ for particles close to the axis. Likewise, particles close to the axis will deposit some of their current/charge to the ghost region below $r = 0$ when computing the current/charge density. The interpolation and deposition schemes themselves are not modified at the boundaries. Rather, the appropriate treatment of the fields acting on the particle near a boundary is handled by properly setting the fields in the ghost region used for interpolation to the particles. Similarly, the appropriate treatment of the current and charge density near a boundary is handled by appropriately folding the particle current and charge deposited to the ghost region back into the interior of the domain. Note that this operation can be performed on a grid-based container, which is more efficient than doing it on a particle-by-particle basis.

The fields in the ghost region beyond the axis of symmetry used for interpolation to the particles are set by treating each field quantity as either even or odd with respect to r. All field quantities that are identically zero on axis, which are E_r, E_θ, B_r, and B_θ for cylindrical geometry and all three vector components of $\mathbf{E}$ and $\mathbf{B}$ in spherical geometry, are treated as odd. For example,

$$E_r(i-1/2, j) = -E_r(-i+1/2, j), \text{ for } i \le 0, \tag{B.1}$$

$$B_r(i, j+1/2) = -B_r(-i, j+1/2), \text{ for } i \le 0. \tag{B.2}$$

Note that treating these quantities as odd about the axis is consistent with the analytical continuation of these fields for axisymmetric modes in cylindrical geometry where a power law expansion of these quantities near axis has only odd powers of r [30]. For non-axisymmetric modes in cylindrical geometry (not considered in this work), and generally for spherical geometry, each of the field values that are zero right on axis may have even powers of r in their power law expansions near axis. While treating these field

quantities as odd at $r = 0$ to fill the boundary values may not be consistent with their analytic continuation into this region, we argue that it should still be done as doing so ensures that these field values interpolated to the particle converge identically to zero as the particle approaches $r = 0$. The axial fields, B_z and E_z in cylindrical geometry, are treated as even with respect to $r = 0$, consistent with the boundary conditions $\partial B_z/\partial r = \partial E_z/\partial r = 0$ for axisymmetric cylindrical geometry.

The proper way to fold the current deposited to the region below $r = 0$ back into the domain is derived based on that needed to maintain exact energy conservation. It can be shown from the total energy law derived in Sec. 3.3 that in order to achieve exact energy conservation the particle must deposit current to each location on the grid where it gets electric field used to advance its velocity. It is straightforward to show that this is maintained by subtracting (adding) the current deposited to the ghost region from its mirror image location above $r = 0$ for the components of the current that correspond to the components of the electric field that are odd (even) about the axis. For example,

$$I_r(-i+1/2, j) \rightarrow I_r(-i+1/2, j) - I_r(i-1/2, j), \text{ for } i \leq 0, \tag{B.3}$$

$$I_z(-i, j+1/2) \rightarrow I_z(-i, j+1/2) + I_z(i, j+1/2), \text{ for } i < 0. \tag{B.4}$$

Folding the current deposited in the ghost region back into the domain as described in Eqs. (B.3)-(B.4) is precisely what is done at a standard symmetry plane in a planar geometry where one can argue that these contributions come from the mirror charge across the symmetry plane with equal and opposite current vector. If using a charge conserving current deposition scheme, then local charge conservation is maintained by adding the charge deposited into the ghost region back to its mirror image location across the symmetry axis (viz. Eq. (B.4)). All that remains is to prove local charge conservation right on axis at $r = 0$.

Here, without loss of generality, local charge conservation at $r = 0$ is proved for 1D cylindrical geometry. Extension to 2D cylindrical and 1D spherical is trivial. The divergence of the current density has a singularity at $r = 0$, and one cannot use the discrete form given in Eq. (27). The numerical treatment $\partial\left(rJ_r\right)/\partial r/r$ near axis can be obtained by expanding J_r in odd powers of r and only retaining the lowest order term. This gives

$$J_r(r \leq \Delta r/2) = J_{r,1/2}\frac{r}{\Delta r/2} \;\Rightarrow\; \frac{1}{r}\frac{\partial\left(rJ_r\right)}{\partial r}\bigg|_{r\rightarrow 0} = \frac{4J_{r,1/2}}{\Delta r} = \frac{4}{\pi(\Delta r)^2}\left(\frac{I_{r,1/2} - I_{r,-1/2}}{\Delta r}\right). \tag{B.5}$$

The last expression here takes into account the folding back in of the current deposited in the ghost region as given in Eq. (B.3) and makes use of the volume element at $i = 1/2$: $\Delta V_{1/2} = 2\pi r_{1/2}\Delta r = \pi(\Delta r)^2$. Using the on-axis volume element, $\Delta V_0 = \pi(\Delta r)^2/4$, the discrete rate of change in charge density at $i = 0$ can be written as

$$\frac{\rho_0^{n+1} - \rho_0^n}{\Delta t} = \frac{4}{\pi(\Delta r)^2}\left(\frac{Q_0^{n+1} - Q_0^n}{\Delta t}\right). \tag{B.6}$$

The term inside the parenthesis in Eq. (B.5) is equal and opposite to the term inside parenthesis in Eq. (B.6) when using a charge-conserving current deposition scheme (see Eq. (26)). Thus, local charge density is conserved at $r = 0$.

The same power-law expansion used for J_r in Eq. (B.5) is used to deal with the singularity on axis present in the z components of $\nabla \times \mathbf{B}$ for cylindrical geometry:

$$\frac{1}{r}\frac{\partial\left(rB_\theta\right)}{\partial r}\bigg|_{r\rightarrow 0} = \frac{4B_{\theta,1/2}}{\Delta r}. \tag{B.7}$$

No such treatment is needed for the similarly expressed θ and ϕ components of $\nabla \times \mathbf{B}$ in 1D spherical geometry as these terms are set identically to zero on axis consistent with the fact that E_θ and E_ϕ are identically zero on axis. Similarly, $\partial B_z/\partial r$ is set to zero at $r = 0$ for cylindrical geometry, consistent with $E_\theta = B_r = 0$ at $r = 0$. When doing second order finite differencing of Maxwell's Equations on a Yee grid, boundary conditions for the remaining variables are not required for the field solve.

References

[1] J.M. Dawson, Particle simulation of plasmas, Rev. Mod. Phys. 55 (1983) 403–447.
[2] C.K. Birdsall, A.B. Langdon, Plasma Physics via Computer Simulation, McGraw-Hill, New York, 1985.
[3] C.K. Birdsall, Particle-in-cell charged-particle simulations, plus Monte Carlo collisions with neutral atoms, pic-mcc, IEEE Trans. Plasma Sci. 19 (1991) 65–85.
[4] K. Nanbu, Probability theory of electron-molecule, ion-molecule, molecule-molecule, and Coulomb collisions for particle modeling of materials processing plasmas and cases, IEEE Trans. Plasma Sci. 28 (2000) 971–990.
[5] A. Friedman, A. Langdon, B. Cohen, A direct method for implicit particle-in-cell simulation, Comments Plasma Phys. Control. Fusion 6 (1981) 225–236.
[6] B.I. Cohen, A. Langdon, D.W. Hewett, R.J. Procassini, Performance and optimization of direct implicit particle simulation, J. Comput. Phys. 81 (1989) 151–168.
[7] E.P. Alves, W.B. Mori, F. Fiuza, Numerical heating in particle-in-cell simulations with Monte Carlo binary collisions, Phys. Rev. E 103 (2021) 013306.
[8] J.R. Angus, A.J. Link, A.E. Schmidt, 1d kinetic study of pinch formation in a dense plasma focus: transition from collisional to collisionless regimes, Phys. Plasmas 28 (2021) 010701.
[9] J.R. Angus, A. Link, A. Friedman, D. Ghosh, J.D. Johnson, On numerical energy conservation for an implicit particle-in-cell method coupled with a binary Monte-Carlo algorithm for Coulomb collisions, J. Comput. Phys. 456 (2022) 111030.
[10] S.E. Anderson, L. Chacón, W.T. Taitano, A.N. Simakov, B.D. Keenan, Fully kinetic simulations of strong steady-state collisional planar plasma shocks, Phys. Rev. E 104 (2021) 055205.
[11] D. Klir, A.V. Shishlov, V.A. Kokshenev, P. Kubes, A.Y. Labetsky, K. Rezac, R.K. Cherdizov, J. Cikhardt, B. Cikhardtova, G.N. Dudkin, F.I. Fursov, A.A. Garapatsky, B.M. Kovalchuk, J. Krasa, J. Kravarik, N.E. Kurmaev, H. Orcikova, V.N. Padalko, N.A. Ratakhin, O. Sila, K. Turek, V.A. Varlachev, A. Velyhan, R. Wagner, Deuterium z-pinch as a powerful source of multi-mev ions and neutrons for advanced applications, Phys. Plasmas 23 (2016) 032702.

[12] E.P. Hartouni, A.S. Moore, A.J. Crilly, B.D. Appelbe, P.A. Amendt, K.L. Baker, D.T. Casey, D.S. Clark, T. Döppner, M.J. Eckart, J.E. Field, M. Gatu-Johnson, G.P. Grim, R. Hatarik, J. Jeet, S.M. Kerr, J. Kilkenny, A.L. Kritcher, K.D. Meaney, J.L. Milovich, D.H. Munro, R.C. Nora, A.E. Pak, J.E. Ralph, H.F. Robey, J.S. Ross, D.J. Schlossberg, S.M. Sepke, B.K. Spears, C.V. Young, A.B. Zylstra, Evidence for suprathermal ion distribution in burning plasmas, Nat. Phys. 19 (2023) 72–77.
[13] J.R. Angus, W. Farmer, A. Friedman, D. Ghosh, D. Grote, D. Larson, A. Link, An implicit particle code with exact energy and charge conservation for electromagnetic studies of dense plasmas, J. Comput. Phys. 491 (2023) 112383.
[14] S. Markidis, G. Lapenta, The energy conserving particle-in-cell method, J. Comput. Phys. 230 (2011) 7037–7052.
[15] G. Chen, L. Chacón, D. Barnes, An energy- and charge-conserving, implicit, electrostatic particle-in-cell algorithm, J. Comput. Phys. 230 (2011) 7018–7036.
[16] J.R. Angus, A.J. Link, A.E.W. Schmidt, One-dimensional theory and simulations of the dynamic z-pinch, Phys. Plasmas 27 (2020) 012108.
[17] G. Lapenta, Exactly energy conserving semi-implicit particle in cell formulation, J. Comput. Phys. 334 (2017) 349–366.
[18] D. Gonzalez-Herrero, A. Micera, E. Boella, J. Park, G. Lapenta, Ecsim-cyl: energy conserving semi-implicit particle in cell simulation in axially symmetric cylindrical coordinates, Comput. Phys. Commun. 236 (2019) 153–163.
[19] L. Chacón, G. Chen, A curvilinear, fully implicit, conservative electromagnetic pic algorithm in multiple dimensions, J. Comput. Phys. 316 (2016) 578–597.
[20] J. Xiao, H. Qin, Explicit structure-preserving geometric particle-in-cell algorithm in curvilinear orthogonal coordinate systems and its applications to whole-device 6d kinetic simulations of tokamak physics, Plasma Sci. Technol. 23 (2021) 055102.
[21] B. Perse, K. Kormann, E. Sonnendrücker, Geometric particle-in-cell simulations of the Vlasov–Maxwell system in curvilinear coordinates, SIAM J. Sci. Comput. 43 (2021) B194–B218.
[22] J. Brackbill, D. Forslund, An implicit method for electromagnetic plasma simulation in two dimensions, J. Comput. Phys. 46 (1982) 271–308.
[23] S. Mattei, K. Nishida, M. Onai, J. Lettry, M. Tran, A. Hatayama, A fully-implicit particle-in-cell Monte Carlo collision code for the simulation of inductively coupled plasmas, J. Comput. Phys. 350 (2017) 891–906.
[24] T. Takizuka, H. Abe, A binary collision model for plasma simulation with a particle code, J. Comput. Phys. 25 (1977) 205–219.
[25] K. Yee, Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media, IEEE Trans. Antennas Propag. 14 (1966) 302–307.
[26] J. Verboncoeur, Symmetric spline weighting for charge and current density in particle simulation, J. Comput. Phys. 174 (2001) 421–427.
[27] J. Villasenor, O. Buneman, Rigorous charge conservation for local electromagnetic field solvers, Comput. Phys. Commun. 69 (1992) 306–316.
[28] G. Chen, L. Chacón, A multi-dimensional, energy- and charge-conserving, nonlinearly implicit, electromagnetic Vlasov–Darwin particle-in-cell algorithm, Comput. Phys. Commun. 197 (2015) 73–87.
[29] T. Esirkepov, Exact charge conservation scheme for particle-in-cell simulation with an arbitrary form-factor, Comput. Phys. Commun. 135 (2001) 144–153.
[30] J.R. Angus, J.J. Van De Wetering, M. Dorf, V.I. Geyko, Eigenmode analysis of the sheared-flow z-pinch, Phys. Plasmas 27 (2020) 122108.
[31] M. Adams, P. Colella, D.T. Graves, J. Johnson, N. Keen, T.J. Ligocki, D.F. Martin, P. McCorquodale, D. Modiano, P. Schwartz, T. Sternberg, B.V. Straalen, Chombo Software Package for AMR Applications - Design Document, technical report lbnl-6616e, Lawrence Berkeley National Laboratory, 2019.
[32] S. Balay, S. Abhyankar, M.F. Adams, S. Benson, J. Brown, P. Brune, K. Buschelman, E. Constantinescu, L. Dalcin, A. Dener, V. Eijkhout, J. Faibussowitsch, W.D. Gropp, V. Hapla, T. Isaac, P. Jolivet, D. Karpeev, D. Kaushik, M.G. Knepley, F. Kong, S. Kruger, D.A. May, L.C. McInnes, R.T. Mills, L. Mitchell, T. Munson, J.E. Roman, K. Rupp, P. Sanan, J. Sarich, B.F. Smith, S. Zampini, H. Zhang, H. Zhang, J. Zhang, PETSc/TAO Users Manual, Technical Report ANL-21/39 - Revision 3.18, Argonne National Laboratory, 2022.
[33] B. Smith, P.E. Bjørstad, W. Gropp, Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations, 1998.
[34] T.F. Chan, H.A. Van Der Vorst, Approximate and incomplete factorizations, in: Parallel Numerical Algorithms, Springer, 1997, pp. 167–202.
[35] J.E. Allen, An elementary theory of the transient pinched discharge, Proc. Phys. Soc. B 70 (1957) 24.
[36] A. Friedman, A second-order implicit particle mover with adjustable damping, J. Comput. Phys. 90 (1990) 292–312.
[37] C. Thoma, D.R. Welch, R.E. Clark, D.V. Rose, I.E. Golovkin, Hybrid-pic modeling of laser-plasma interactions and hot electron generation in gold Hohlraum walls, Phys. Plasmas 24 (2017) 062707.
[38] K. Nanbu, Theory of cumulative small-angle collisions in plasmas, Phys. Rev. E 55 (1997) 4642–4652.
[39] K. Nanbu, S. Yonemura, Weighted particles in Coulomb collision simulations based on the theory of a cumulative scattering angle, J. Comput. Phys. 145 (1998) 639–654.
[40] D.P. Higginson, I. Holod, A. Link, A corrected method for Coulomb scattering in arbitrarily weighted particle-in-cell plasma simulations, J. Comput. Phys. 413 (2020) 109450.
[41] J.R. Angus, Y. Fu, V. Geyko, D. Grote, D. Larson, Moment-preserving Monte-Carlo Coulomb collision method for particle codes, https://arxiv.org/abs/2407.19151, arXiv:2407.19151, 2024.
[42] P. Jiménez, L. Chacón, M. Merino, An implicit, conservative electrostatic particle-in-cell algorithm for paraxial magnetic nozzles, J. Comput. Phys. 502 (2024) 112826.
[43] G. Lapenta, Advances in the implementation of the exactly energy conserving semi-implicit (ecsim) particle-in-cell method, Physics 5 (2023) 72–89.
[44] A. Richardson, 2019 NRL Plasma Formulary, Naval Research Laboratory, Washington, DC, 2019.