Journal of Computational Physics (2026)

Contents lists available at ScienceDirect

Journal of Computational Physics

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

High-order geometry conforming immersed finite element particle-in-cell method for plasma–material simulations with curved interfaces and nonhomogeneous jumps

Pei Qiu[a], Chang Lu[b,*]

[a]College of Zhan Tianyou, Dalian Jiaotong University, Dalian, 116028, China
[b]State Key Laboratory of Industrial Equipment Structural Analysis and Optimization and CAE Software, Dalian University of Technology, Dalian, 116024, China

ARTICLE INFO

Article history:

High-order particle-in-cell method, Geometry-conforming immersed finite element, Curved plasma–material interface, Nonhomogeneous interface jump conditions, Boris–SDC time integration

ABSTRACT

This work develops a high-order geometry-conforming immersed finite-element particle-in-cell (GCIFE–PIC) method for electrostatic plasma simulations on Cartesian meshes. Its central innovation is the integration of pointwise geometry-exact interface basis construction with conservative particle–field transfer and high-order time integration, enabling curved-interface effects to be resolved without interface-fitted meshes. Nonhomogeneous jump conditions couple deposited charge and evolving surface charge to the field solution, while consistent volume and interface charge transfer preserves represented charge near material boundaries. This formulation combines high-order approximation with geometric fidelity, reducing the need for extensive mesh refinement to resolve interface effects. Numerical tests demonstrate high-order field convergence, roundoff-level charge conservation, and improved particle–field transfer near curved interfaces. Even the lowest-order GCIFE approximation captures interface effects on meshes coarser than its bilinear IFE counterpart. In the high-order coupled test, the coarse-mesh GCIFE configuration produces less noisy charge-density distributions at lower accumulated runtime than a refined bilinear IFE–PIC calculation. This method lays a foundation for accurate and efficient simulation of plasma–material interactions such as plasma sheath formation and interfacial charging.

© 2026 Elsevier Inc. All rights reserved.

1. Introduction

Particle-in-cell (PIC) methods are a central tool for kinetic plasma simulations, including fusion plasmas, charged-particle devices, and plasma–material interactions. By evolving particle ensembles in self-consistent mesh fields, PIC

*Corresponding author
e-mail: changlu@dlut.edu.cn (Chang Lu)

estimates statistical physical observables without resolving the full phase-space distribution [1]. More particles or longer averaging can reduce sampling fluctuations, but cannot remove systematic errors in the field solve, particle–mesh transfer, or particle advance. In conventional electrostatic PIC methods, such errors arise from the difficulty of resolving steep gradients and complex or nonconforming geometries [2], and the numerical dispersion and phase errors associated with the discretization of plasma oscillations[3]. In particular, charge accumulation at complex material interfaces yields steep near-surface fields, making sheath fields and related near-interface observables, like ion impact energies and plasma-wave damping rates, highly sensitive to low-order spatial discretization. High-order discretizations have the potential to reduce these errors and provide a numerical foundation for reliable statistical predictions.

Cartesian meshes are attractive for PIC because their regular structure simplifies particle location, tensor-product field operations, and repeated particle–mesh transfer. For curved boundaries, adaptive mesh refinement (AMR) is commonly employed to enhance interface resolution: it reduces this error by concentrating Cartesian cells where finer resolution is needed [4], and has been applied to plasma PIC simulations [5, 6]. Even so, curved boundaries and material interfaces remain approximated by staircasing. Refining such a staircased interface does not resolve the fundamental difficulty. Local normals and curvature are misrepresented, and jump conditions are enforced on an approximate surface offset from the true interface by $O(h)$. Even with high-order polynomials, overall accuracy remains geometry-limited. Such distortions may mask curvature-dependent physics in geometry-sensitive processes including secondary electron emission, plasma etching [7], and dust charging [8]. Long-time simulations further exacerbate these discrepancies via accumulated surface charge and particle-trajectory deviations.

Unstructured meshes provide another route to both geometric flexibility and high-order PIC discretization. Jacobs and Hesthaven [9] developed a high-order nodal discontinuous Galerkin PIC method on such meshes, combining a high-order field solver with smooth particle shapes and full-order field interpolation. This significantly improved the accuracy of unstructured-mesh PIC and reduced numerical noise. Brown et al. [10] represented smooth finite-support particle shapes by weighted virtual particles derived from Gaussian quadrature, thereby evaluating their weak-form coupling on unstructured elements and reducing the cost of charge deposition and field interpolation. Stindl et al. [11] developed a fully electromagnetic HDG-PIC solver for complex-shaped domains and systematically compared different particle-grid coupling approaches, evaluating their accuracy and computational demand. These developments show the value of combining geometric flexibility with high-order PIC. They also highlight the work required to couple a particle support to field tests when the support crosses multiple elements.

Immersed finite element (IFE) methods on Cartesian meshes provide a different route to geometric flexibility while retaining the regular background grid. They use specially constructed basis functions on elements crossed by an interface and standard finite-element basis functions on ordinary elements [12, 13, 14, 15, 16, 17, 18]. For PIC, this decouples the field mesh from interface conformity, so that Cartesian IFE meshes retain the efficiency of regular-grid PIC while still resolving complex interfaces. Unlike interface-fitted unstructured finite elements, they avoid interface-conforming mesh generation and preserve direct cell indexing for particle location. IFE-PIC therefore combines efficient Cartesian-grid particle–mesh operations with interface-aware field approximation for

complex geometries [19]. Numerous variants have been developed [20, 21, 22, 23, 24, 25, 26, 27], and IFE–PIC has been applied to ion propulsion [28, 29, 30, 31], plume dynamics [32, 33, 34], lunar surface charging [35, 36, 37, 8, 38, 39, 40, 41], and asteroid plasma environments [42, 43, 44, 45, 46, 47, 48, 49, 50]. However, commonly used linear and bilinear IFE–PIC formulations still rely on low-order field approximations, limiting their ability to exploit high-order particle coupling near curved interfaces.

A recently developed geometry-conforming immersed finite-element (GCIFE) method extends this approach to high-order approximation on curved interfaces. It uses a Frenet transformation to enable high-order enforcement of interface jump conditions with a universal $\mathbb{Q}_m$-type polynomial basis [51]. Featuring well-conditioned local formulations and optimal approximation properties—with approximation constants insensitive to coefficient contrast and interface position—GCIFE offers a promising pathway toward high-order PIC simulations on Cartesian meshes containing curved material interfaces. Naturally, realizing this potential still requires addressing several particle–field coupling challenges. First, nonhomogeneous flux and source jumps originating from surface charge and one-sided particle deposition need to be properly incorporated. Second, the disparate particle-shape and GCIFE field bases must be coupled self-consistently to maintain deposition-gathering reciprocity, particularly near curved interfaces. Failure to resolve these issues can lead to spurious self-forces, grid-crossing noise, and deteriorated conservation behavior.

This work develops a GCIFE–PIC formulation for self-consistent particle transport and surface charging at curved material interfaces on Cartesian meshes. Nonhomogeneous flux and source jumps are incorporated through local unit-jump functions, weighted by the prescribed jump moments and added to the homogeneous GCIFE approximation. The flux jump is supplied by surface charge; source jumps are obtained from the traces and normal derivatives of independent one-sided Frenet projections of deposited density. These contributions modify only the Poisson right-hand side, allowing the homogeneous stiffness matrix to be reused throughout the simulation. On interface elements, particle charge enters the Poisson load directly through particle–basis moments. Particle shapes are restricted to the material side containing their centers and normalized over their complete accessible supports; the same shapes gather the electric field from the full potential, including its nonhomogeneous part. Together with arclength-normalized deposition of absorbed charge on the true interface, this coupling preserves represented charge and maintains deposition–gathering consistency near curved surfaces. Boris–SDC supplies high-order particle advancement [52].

The resulting formulation retains Cartesian particle tracking while resolving interface geometry and charge-induced field jumps without a fitted mesh. Its geometric fidelity and consistent particle–field coupling allow high-order field accuracy to benefit the coupled dynamics on coarse meshes. The numerical tests demonstrate improved near-interface resolution and, for the high-order configuration, lower visible charge-density noise and reduced accumulated runtime relative to a refined bilinear control. These features support accurate plasma–material simulations with fewer cells, while the local DG structure provides a foundation for future hp refinement.

The remainder of this paper is organized as follows. Section 2 presents the governing equations and interface conditions. Section 3 details the homogeneous and nonhomogeneous GCIFE constructions. Section 4 elaborates the high-order PIC coupling and time step advance. Section 5 presents numerical illustrations. Section 6 concludes the present work.

2. Governing Equations and the Electrostatic PIC Framework

Under the collisionless approximation, the phase-space distribution function $f_p(\mathbf{x}, \mathbf{v}, t)$ of a particle species, satisfies the Vlasov equation

$$\frac{\partial f_p}{\partial t} + \mathbf{v} \cdot \nabla_{\mathbf{x}} f_p + \frac{q_p}{m_p} (\mathbf{E} + \mathbf{v} \times \mathbf{B}_0) \cdot \nabla_{\mathbf{v}} f_p = 0, \tag{1}$$

where q_p and m_p are the charge and mass of one charged particle, respectively, and $\mathbf{B}_0$ denotes a prescribed static magnetic field.

In the electrostatic regime considered in this work, the electric field is irrotational and can be expressed in terms of an electrostatic potential u as

$$\mathbf{E} = -\nabla u. \tag{2}$$

Let the computational domain $\Omega \subset \mathbb{R}^2$ be partitioned by a smooth curved interface Γ into two disjoint subdomains Ω^+ and Ω^-, such that $\Omega = \Omega^+ \cup \Gamma \cup \Omega^-$. The interface may represent, for example, a plasma–dielectric boundary or an interface between two materials with different permittivities. The electrostatic potential satisfies

$$-\nabla \cdot (\varepsilon(\mathbf{x}) \nabla u(\mathbf{x}, t)) = \rho(\mathbf{x}, t), \qquad \mathbf{x} \in \Omega^+ \cup \Omega^-, \tag{3}$$

where ρ is the total charge density of all charged particles, and the permittivity ε is piecewise constant,

$$\varepsilon(\mathbf{x}) = \begin{cases} \varepsilon^+, & \mathbf{x} \in \Omega^+, \\ \varepsilon^-, & \mathbf{x} \in \Omega^-. \end{cases} \tag{4}$$

Appropriate Dirichlet, Neumann, or mixed boundary conditions are prescribed on the outer boundary $\partial\Omega$.

For a general electrostatic interface problem, the interface data reflect the physical effects of surface and volume charges. In the present plasma–dielectric model, charged particles absorbed by the dielectric accumulate as surface charge on Γ. The resulting jump in the normal electric displacement is denoted by g_σ. Moreover, although the bulk plasma is approximately quasineutral, the unequal transport and absorption of electrons and ions produce a nonneutral sheath adjacent to the dielectric. The resulting net volume charge on the plasma side generally differs from that on the dielectric side, where free volume charge is absent in the present model. This charge-density mismatch induces jumps in the higher-order derivatives of the potential through the Poisson equation. We denote the corresponding nth source-jump datum by g_ρ^n. It is determined by the jump in the nth normal derivative of the volume charge density.

We define the potential and the normal electric displacement as

$$[\![u]\!]_\Gamma = g_0, \tag{5}$$

$$[\![\varepsilon \partial_{\mathbf{n}} u]\!]_\Gamma = g_\sigma, \tag{6}$$

where $[\![w]\!]_\Gamma = w^+\big|_\Gamma - w^-\big|_\Gamma$, and $\mathbf{n}$ is the unit normal directed from Ω^- to Ω^+. In this work, the electrostatic potential is assumed to be continuous, so that $g_0 = 0$. Since $\varepsilon^\pm$ are constant in their respective subdomains, taking the nth normal derivative of the Poisson equation on both sides of Γ gives

$$[\![\varepsilon \partial_{\mathbf{n}}^n \Delta u]\!]_\Gamma = -[\![\partial_{\mathbf{n}}^n \rho]\!]_\Gamma = g_\rho^n, \qquad n = 0, 1, \ldots, m-2. \tag{7}$$

Including these jump conditions removes the requirement that the deposited charge density be continuous across the interface, while independent one-sided reconstructions of the deposited source are required to approximate g_ρ^n. If the surface charge vanishes and the relevant normal derivatives of the source are continuous across Γ, then $g_\sigma = 0$ and $g_\rho^n = 0$, and the homogeneous GCIFE formulation is recovered. Otherwise, the nonzero data g_σ and g_ρ^n enter the nonhomogeneous GCIFE construction as separate interface contributions, and the discrete reconstructions of g_σ and g_ρ^n are defined in Section 4.3.

To discretize the Vlasov equation (1), the PIC method represents each distribution function by a finite collection of macroparticles:

$$f_p(\mathbf{x}, \mathbf{v}, t) \approx \sum_{p=1}^{N_p} \mathcal{N}_p S_p\Big(\mathbf{x} - \mathbf{x}_p(t)\Big)\,\delta\Big(\mathbf{v} - \mathbf{v}_p(t)\Big), \tag{8}$$

where N_p is the number of macroparticles , $\mathcal{N}_p$ is the corresponding statistical weight, and $\mathbf{x}_p$ and $\mathbf{v}_p$ denote the particle position and velocity. The spatial shape function S_p determines the finite extent and smoothness of the macroparticle.

The macroparticle trajectories satisfy the Newton–Lorentz equations

$$\frac{d\mathbf{x}_p}{dt} = \mathbf{v}_p, \tag{9}$$

$$\frac{d\mathbf{v}_p}{dt} = \frac{q_p}{m_p}\Big[\mathbf{E}_p\Big(\mathbf{x}_p\Big) + \mathbf{v}_p \times \mathbf{B}_0\Big], \tag{10}$$

where $\mathbf{E}_p$ is the electric field gathered to the particle position.

The volume charge density in Eq. (3) is contributed by particles:

$$\rho(\mathbf{x}, t) = \sum_s q_p^s \sum_{p=1}^{N_p^s} \mathcal{N}_p^s S_p^s\Big(\mathbf{x}^s - \mathbf{x}_p^s(t)\Big). \tag{11}$$

where s denotes the particle species, such as electronics or ions.

In the present work, the particle source (11) is coupled to Eq. (3) by direct mixed moments with the field test functions. In particularly, for interface elements, the one-sided accessible particle shape S_p is normalized over its complete accessible support, and its moments against homogeneous GCIFE test functions are assembled directly into the Poisson load. These moments are accumulated directly as Poisson load entries, but are not used to reconstruct the density on either side of an interface element. For source-jump data, the deposited source is projected independently on each material side into a raw Frenet polynomial basis, and the one-sided reconstructed traces and normal derivatives define the jumps.

For particles absorbed at the material interface Γ, charges are accumulated over the time step and processed together once the time step is accepted, at its endpoint. For each absorbed particle, a surface kernel is constructed by centering the volumetric particle kernel at first contact, restricting it to the interface, and normalizing it over its complete surface support using physical arclength. The resulting charge increment is projected onto each physical interface segment, and its local polynomial extension supplies the flux-jump moments for the GCIFE field solve, as specified in Subsection 4.3.

The self-consistent electrostatic PIC cycle over each time interval is therefore composed of the following coupled stages:

1. **Charge deposition and jump-data evaluation.** On ordinary elements, particle-shape moments are assembled against the standard finite-element basis. On interface elements, each particle is restricted to the accessible side of its center and normalized once over its complete accessible support. The resulting moments against homogeneous GCIFE field tests enter the Poisson right-hand side directly. Independently, one-sided density reconstructions are obtained by projecting the deposited density on each material side into a raw Frenet polynomial basis; these reconstructions are used to evaluate g_ρ^j. These reconstructions provide jump data, not an intermediate global density for the volume load. The flux-jump datum g_σ comes from the surface-charge deposition.
2. **Nonhomogeneous GCIFE field solve.** Using the density and jump data from the previous step, solve the Poisson problem (3) subject to the prescribed nonhomogeneous flux and source-induced jumps (6) and (7). To enforce these jumps, the solution is decomposed as [14, 15]:

$$u = u_0 + u_J, \tag{12}$$

   where u_J is a prescribed function satisfying the nonhomogeneous jump conditions (6) and (7), and u_0 is the unknown homogeneous part. The contribution of u_J is transferred to the right-hand side of the discrete system, while u_0 is expanded in the homogeneous GCIFE basis. The mesh electric field is then evaluated from the resulting potential as Eq. (2).
3. **Field gathering.** Gather the mesh electric field at each particle using the GCIFE field representation associated with the material containing the particle center. The same analytical particle shape used in charge deposition, with the same accessible-side restriction and normalization when required, is used in the particle–field coupling.
4. **Particle push.** Advance the particle positions and velocities according to Eqs. (9)–(10). In the high-order formulation developed below, this stage is performed using the Boris–spectral deferred correction integrator over a fixed time step. First-contact events are collected from the trial trajectories without subdividing the global step. At acceptance, absorbed particles are removed and their surface charges are updated together; surviving particles retain their full-step endpoint states. The endpoint field is then recomputed, as detailed in Subsection 4.4.

3. Geometry-Conforming Immersed Finite Element Method with Nonhomogeneous Flux and Source Jumps

In this section, we introduce the GCIFE method with **nonhomogeneous flux jumps** from interfacial surface charges and **high-order nonhomogeneous source jumps** from PIC charge deposition. This extension represents the core innovation of the high-order GCIFE–PIC framework, permitting accurate electrostatic field solutions for discontinuous interface conditions within self-consistent plasma simulations.

The GCIFE method possesses two prominent advantages over classical IFE methods: (1) it supports arbitrary high-order polynomial construction; (2) it strictly enforces physical interface jump conditions on the exact curved interface geometry. The fundamental building block of GCIFE is the Frenet coordinate transformation, which maps the curved physical interface in Cartesian coordinates to a straight line in a local curvilinear coordinate system. This

transformation allows direct and accurate imposition of both homogeneous and newly proposed nonhomogeneous jump conditions during local high-order basis construction.

3.1. Frenet Coordinate Mapping and Interface Straightening

Let the smooth material interface $\Gamma \subset \Omega$ with regular parametric representation $\mathbf{g}(\xi) : [\xi_s, \xi_e] \to \mathbb{R}^2$, oriented so the normal below points from Ω^- to Ω^+. The local unit tangent, unit normal, and signed curvature are

$$\boldsymbol{\tau}(\xi) = \frac{\mathbf{g}'(\xi)}{\|\mathbf{g}'(\xi)\|}, \quad \mathbf{n}(\xi) = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \boldsymbol{\tau}(\xi), \quad \kappa(\xi) = \frac{\det(\mathbf{g}'(\xi), \mathbf{g}''(\xi))}{\|\mathbf{g}'(\xi)\|^3}. \tag{13}$$

For any physical coordinate $\mathbf{x} = (x, y)$ in a tubular neighborhood of Γ, we define the Frenet forward transformation $P_\Gamma(\eta, \xi)$ and its inverse mapping $R_\Gamma(\mathbf{x}) = (\eta, \xi)$. Specifically, for some $\epsilon > 0$, let

$$N_\Gamma(\epsilon) = P_\Gamma([-\epsilon, \epsilon] \times [\xi_s, \xi_e]), \tag{14}$$

where the Frenet map P_Γ is assumed to be one-to-one and onto this neighborhood, with a nonsingular Jacobian. Then

$$\mathbf{x} = P_\Gamma(\eta, \xi) = \mathbf{g}(\xi) + \eta \mathbf{n}(\xi), \tag{15}$$

where η denotes the signed normal distance to Γ, and ξ parameterizes the tangential direction. Under R_Γ, a Cartesian interface element K is mapped to a curvilinear domain $\hat{K}$, enclosed within the rectangular Frenet fictitious element $\hat{K}_F = [-\eta_h, \eta_h] \times [\xi_1, \xi_2]$, where $\eta_h > 0$ is its normal half-width and $[\xi_1, \xi_2]$ its tangential interval. As illustrated in Fig. 1, the curved interface is straightened to $\eta = 0$ (denoted $\hat{\Gamma}_{K_F}$). For each element, write this same tangential interval as $I_K = [\xi_{0K}, \xi_{1K}] = [\xi_1, \xi_2]$, where ξ_{0K} and ξ_{1K} are the minimum and maximum tangential coordinates of its vertices under the local Frenet map. Set $\xi_{\mathrm{mid},K} = (\xi_{0K} + \xi_{1K})/2$ and $\xi_{h,K} = (\xi_{1K} - \xi_{0K})/2$. Thus $\hat{\Gamma}_{K_F} = \{0\} \times I_K$; this interval need not coincide with the parameter range of the physical segment $\Gamma \cap K$.

Remark 1. *For each interface element $K \in \mathcal{T}_h$, $\hat{K}$ is termed the Frenet interface element, $\hat{K}_F$ the Frenet fictitious element, K_F the physical fictitious element, and $\hat{\Gamma}_{K_F}$ the straightened local interface. Geometric illustrations are provided in Fig. 1.*

In the local Frenet coordinates on the fictitious element $\hat{K}_F$, define the pull-back quantities

$$\hat{\varepsilon}^K(\eta, \xi) = \varepsilon^K(P_\Gamma(\eta, \xi)), \qquad \hat{u}^K(\eta, \xi) = u^K(P_\Gamma(\eta, \xi)), \qquad \hat{\rho}^K(\eta, \xi) = \rho^K(P_\Gamma(\eta, \xi)).$$

Let $\mathcal{L}$ denote the Laplacian transformed into Frenet coordinates. Across $\hat{\Gamma}_{K_F}$, the potential remains continuous:

$$\left[\!\left[\hat{u}^K\right]\!\right]_{\hat{\Gamma}_{K_F}} = 0. \tag{16}$$

Unlike the original homogeneous GCIFE formulation, the present extension admits a prescribed nonzero flux jump

$$\left[\!\left[\hat{\varepsilon}^K \frac{\partial \hat{u}^K}{\partial \eta}\right]\!\right]_{\hat{\Gamma}_{K_F}} = \hat{g}_\sigma^K, \tag{17}$$

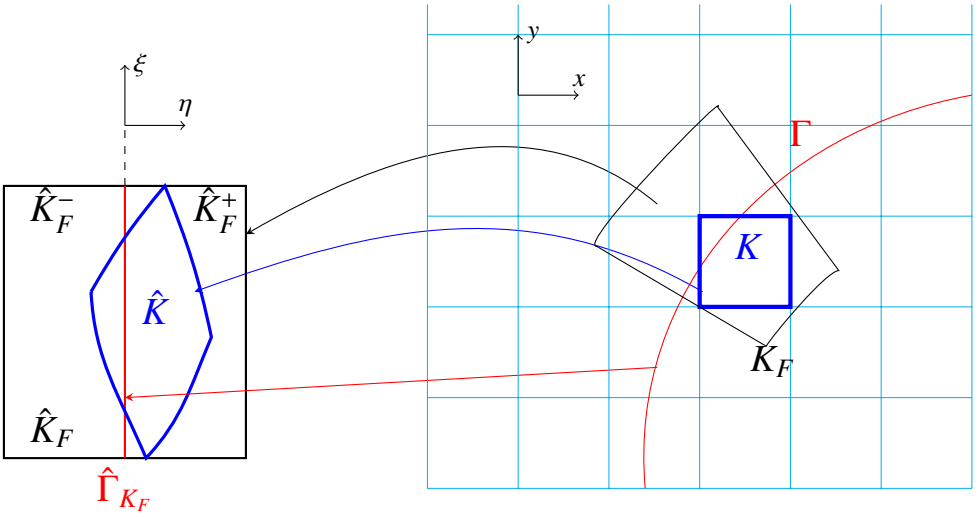


Fig. 1: Schematic of the Frenet mapping. **Right:** In the physical (x, y) domain, the Cartesian cell K is intersected by the curved interface Γ, enclosed by a fictitious Frenet box K_F. **Left:** The inverse mapping R_Γ transforms K_F into a standard rectangle $\hat{K}_F$, where the exact curved interface is straightened into a linear segment $\hat{\Gamma}_{K_F}$ along the ξ-axis, precisely embedding the curvature physics into the local metrics.

where $\hat{g}_\sigma^K$ is the local Frenet pull-back of the prescribed flux jump, with sign consistent with the normal convention. For polynomial degree $m \geq 2$, the charge source generally yields distinct traces on the two sides of the interface. The associated extended interface conditions are therefore nonhomogeneous:

$$\left[\!\!\left[\hat{\varepsilon}^K \frac{\partial^n}{\partial \eta^n} \mathcal{L}(\hat{u}^K) \right]\!\!\right]_{\hat{\Gamma}_{K_F}} = -\left[\!\!\left[\frac{\partial^n \hat{\rho}^K}{\partial \eta^n} \right]\!\!\right]_{\hat{\Gamma}_{K_F}} = \hat{g}_\rho^{K,n}, \qquad n = 0, 1, \ldots, m-2, \tag{18}$$

where $\mathcal{L}(\hat{u}(\eta, \xi)) = \Delta u(\mathbf{x})$. The local jump data are $\hat{g}_\sigma^K$ and $\hat{g}_\rho^{K,n}$, evaluated at the field-solve stage in Frenet coordinates.

The homogeneous GCIFE basis functions are constructed by enforcing homogeneous counterparts of Eqs. (16)–(18). As proved in [51], the local IFE space defined on the Frenet fictitious element $\hat{K}_F$ is well-posed:

$$\hat{V}_{\hat{\varepsilon},0}^m(\hat{K}_F) = \left\{ \hat{\phi} : \hat{K}_F \to \mathbb{R} \;\middle|\; \hat{\phi}\big|_{\hat{K}_F^\pm} \in \mathbb{Q}_m \text{ and } \hat{\phi} \text{ satisfies the homogeneous counterparts of Eqs. (16)–(18)} \right\}, \tag{19}$$

where $\mathbb{Q}_m$ denotes the tensor-product polynomial space of degree at most m in each coordinate. Given prescribed jump data $\mathcal{G} = (\hat{g}_\sigma^K, \hat{g}_\rho^{K,0}, \ldots, \hat{g}_\rho^{K,m-2})$, the corresponding local nonhomogeneous GCIFE space is the affine space

$$\hat{V}_{\hat{\varepsilon},\mathcal{G}}^m(\hat{K}_F) = \hat{u}_J^K + \hat{V}_{\hat{\varepsilon},0}^m(\hat{K}_F), \tag{20}$$

with $\hat{u}_J^K$ satisfying Eqs. (16) –(18). The homogeneous GCIFE space on the physical interface element K is defined via pull-back from the Frenet domain $N_\Gamma(\epsilon)$:

$$V_{\varepsilon,0}^m(K) = \left\{ \hat{\phi} \circ R_\Gamma|_K \;\middle|\; \hat{\phi} \in \hat{V}_{\hat{\varepsilon},0}^m(\hat{K}_F) \right\}, \tag{21}$$

and the associated nonhomogeneous local affine space reads

$$V_{\varepsilon,\mathcal{G}}^m(K) = u_J^K + V_{\varepsilon,0}^m(K). \tag{22}$$

Unlike classical IFE methods, functions in the affine GCIFE space satisfy the physical jump conditions exactly on the true curved interface, capturing both flux discontinuity from surface charge and high-order source jumps.

Given the computational domain Ω, let $\mathcal{T}_h$ be its uniform rectangular mesh with element diameter h. An element $K \in \mathcal{T}_h$ is called an interface element if its interior intersects the interface, $K \cap \Gamma \neq \emptyset$; otherwise it is a non-interface element. Denote by $\mathcal{T}_h^i$ and $\mathcal{T}_h^n$ the sets of interface and non-interface elements, respectively. The global homogeneous GCIFE space is defined by

$$V_{h,0} = \left\{ v_h \in L^2(\Omega) \;\middle|\; v_h|_K \in \mathbb{Q}_m(K), \quad K \in \mathcal{T}_h^n; \quad v_h|_K \in V_{\varepsilon,0}^m(K), \quad K \in \mathcal{T}_h^i \right\}. \tag{23}$$

The discrete solution belongs to the affine GCIFE space

$$V_{h,\mathcal{G}} = u_J + V_{h,0}, \tag{24}$$

where u_J is supported only on interface elements carrying nonzero jump data. Trial functions lie in $V_{h,\mathcal{G}}$, while test functions are selected from the homogeneous space $V_{h,0}$.

Because local GCIFE functions are constructed elementwise, they may be discontinuous across mesh edges. We therefore use the symmetric interior penalty discontinuous Galerkin (SIPDG) formulation. Let $\mathcal{E}_h$ represent all mesh edges, $\mathcal{E}_h^i$ represent all the interior edges and $\mathcal{E}_h^b$ represent all the boundary edges. For $z_h, v_h \in V_{h,0}$, define

$$\begin{aligned} a_h(z_h, v_h) = & \sum_{K \in \mathcal{T}_h} \int_K \varepsilon \nabla z_h \cdot \nabla v_h \, d\mathbf{x} \\ & - \sum_{e \in \mathcal{E}_h} \Big(\big\langle [\![\varepsilon \nabla z_h \cdot \mathbf{n}_e]\!]_e, \{\!\{ v_h \}\!\}_e \big\rangle_e + \big\langle \{\!\{ z_h \}\!\}_e, [\![\varepsilon \nabla v_h \cdot \mathbf{n}_e]\!]_e \big\rangle_e \\ & \qquad - \frac{\gamma_e \varepsilon_e}{h_e} \big\langle [\![z_h]\!]_e, [\![v_h]\!]_e \big\rangle_e \Big). \end{aligned} \tag{25}$$

Here the operators $\{\!\{\cdot\}\!\}$ and $[\![\cdot]\!]$ denote the standard average and jump operators on mesh edges, $\langle \cdot, \cdot \rangle_e$ denotes the L^2 inner product on e, $\mathbf{n}_e$ is the fixed unit normal on e, h_e is the characteristic edge length, ε_e is an edge permittivity, and $\gamma_e > 0$ is the penalty parameter.

The load functional reads

$$\ell_h(v_h) = \mathcal{D}_{\mathrm{PIC},h}(v_h) + \ell_h^{\partial\Omega}(v_h), \tag{26}$$

where $\mathcal{D}_{\mathrm{PIC},h}(v_h)$ is the direct PIC volume source term and $\ell_h^{\partial\Omega}(v_h)$ collects the Nitsche boundary contributions. The direct PIC volume source term is defined by

$$\mathcal{D}_{\mathrm{PIC},h}(v_h) = \sum_p \frac{q_p}{Q_p^{\alpha_p}} \sum_{K \in \mathcal{T}_h} \int_{K \cap D_p^{\alpha_p}} S_p v_h \, d\mathbf{x}, \tag{27}$$

where the outer sum ranges over all particles, α_p denotes the material side containing the particle center, $D_p^{\alpha_p}$ is the portion of its shape support lying on that side and inside the computational domain, and $Q_p^{\alpha_p} = \int_{D_p^{\alpha_p}} S_p \, d\mathbf{x} > 0$ is the corresponding normalization. This ensures that the total charge q_p is deposited only over the retained support on the side containing the particle center, even when the support is cut by the interface or the outer boundary of the computational domain. Their precise definitions are given in Section 4. Taking Dirichlet boundary conditions as an example, the boundary term is defined by

$$\ell_h^{\partial\Omega}(v_h) = \sum_{e \in \mathcal{E}_h^b} \Big\langle -\varepsilon_e \nabla v_h \cdot \mathbf{n}_e + \frac{\gamma_e \varepsilon_e}{h_e} v_h, \, g_D \Big\rangle_e, \tag{28}$$

where g_D is the prescribed Dirichlet datum.

The complete discrete equation for $u_h = u_{0,h} + u_{J,h}$ [14, 15] is

$$a_h(u_h, v_h) = \ell_h(v_h), \qquad \forall v_h \in V_{h,0}. \tag{29}$$

Equivalently, the homogeneous unknown component satisfies

$$a_h(u_{0,h}, v_h) = \ell_h(v_h) - a_h(u_{J,h}, v_h), \qquad \forall v_h \in V_{h,0}. \tag{30}$$

This shows that the proposed nonhomogeneous jump extension only modifies the right-hand side, while the stiffness matrix from the homogeneous GCIFE basis remains unchanged.

The above summarizes the GCIFE framework and its extension developed in this paper for high-order electrostatic PIC simulations. Compared to standard homogeneous GCIFE, the present formulation incorporates nonhomogeneous flux jumps from interface surface charge and high-order source jumps from PIC particle deposition. Jump constraints are realized by local unit-jump nonhomogeneous functions, which contribute only to the right-hand side of the discrete variational problem. The homogeneous stiffness matrix remains fixed as particle positions and surface charge evolve. Further details for the baseline homogeneous GCIFE construction can be found in [51, 53].

3.2. Construction of GCIFE Basis with Homogeneous and Nonhomogeneous Jump Conditions

3.2.1. Homogeneous Basis Functions

To construct the high-order homogeneous GCIFE space $\hat{V}^m_{\hat{\mathrm{e}},0}(\hat{K}_F)$ with dimension $d_m = (m+1)^2$ on the Frenet fictitious element $\hat{K}_F$, we employ tensor-product polynomials defined within the local Frenet coordinate system. For each interface element K, geometric scaling parameters $(\eta_h, \xi_h, \xi_{\mathrm{mid}})$ are extracted from element vertices following [53]. The scaled one-dimensional basis functions read

$$\hat{p}_i(\xi) = p_i\left(\frac{\xi - \xi_{\mathrm{mid}}}{\xi_h}\right), \quad \hat{q}_j(\eta) = q_j\left(\frac{\eta}{\eta_h}\right), \quad 0 \le i, j \le m, \tag{31}$$

where $\{p_i\}_{i=0}^m$ denote standard L^2-orthogonal Legendre polynomials on $[-1, 1]$. The element subscript on the geometric scaling parameters and the superscript K on the local basis are suppressed here.

To decouple jump constraints imposed on the straightened interface $\hat{\Gamma}_{K_F}$ $(\eta = 0)$, a family of custom polynomials $\{q_j(x)\}_{j=0}^m$ is defined recursively from Legendre polynomials [20]:

$$q_0(x) = 1, \quad q_1(x) = x, \quad q_j(x) = (2j-1)xp_{j-1}(x) - (2j-1)p_{j-1}(0)x, \quad 2 \le j \le m. \tag{32}$$

These polynomials satisfy the critical double-zero property at the interface:

$$q_j(0) = 0, \quad q_j'(0) = 0, \quad \forall\, 2 \le j \le m, \quad \text{and} \quad q_1(0) = 0. \tag{33}$$

This property enforces high-order modes and their normal derivatives to vanish on the interface, substantially simplifying the imposition of jump conditions.

We next construct the generalized homogeneous GCIFE basis over the Frenet fictitious element $\hat{K}_F$. Let

$$\mathcal{R} = \{R_\ell\}_{\ell=1}^{(m+1)^2}$$

be a basis for the tensor-product polynomial space $\mathbb{Q}_m(\hat{K}_F)$. Explicitly, define

$$R_{(m+1)t+s+1}(\eta,\xi) = \hat{q}_t(\eta)\hat{p}_s(\xi) = q_t\left(\frac{\eta}{\eta_h}\right)p_s\left(\frac{\xi-\xi_{\mathrm{mid}}}{\xi_h}\right), \qquad 0 \le s,t \le m, \tag{34}$$

where $\{q_t\}_{t=0}^m$ and $\{p_s\}_{s=0}^m$ represent the one-dimensional polynomial bases along the normal and tangential directions, respectively.

Then each homogeneous GCIFE basis function is built by extending one polynomial component from one side of the interface across to the opposite side. For $j = 1,\dots,(m+1)^2$, define

$$\hat{\lambda}_j(\eta,\xi) = \begin{cases} \hat{\lambda}_j^-(\eta,\xi) = \displaystyle\sum_{\ell=1}^{(m+1)^2} C_{\ell,j}^- R_\ell(\eta,\xi), & \eta < 0, \\ \hat{\lambda}_j^+(\eta,\xi) = \displaystyle\sum_{\ell=1}^{(m+1)^2} C_{\ell,j}^+ R_\ell(\eta,\xi), & \eta > 0. \end{cases} \tag{35}$$

The coefficient vectors $\mathbf{C}_j^{\pm}$ are determined such that $\hat{\lambda}_j$ satisfies the homogeneous interface conditions on $\hat{\Gamma}_{K_F}$.

The interface constraints for the j-th basis function are enforced in weak form against the tangential basis $\{\hat{p}_k\}_{k=0}^m$ defined in Eq. (31):

$$\sum_{\ell=1}^{(m+1)^2}\left(\int_{\hat{\Gamma}_{K_F}} R_\ell(0,\xi)\hat{p}_k(\xi)\,d\xi\right)C_{\ell,j}^+ = \sum_{\ell=1}^{(m+1)^2}\left(\int_{\hat{\Gamma}_{K_F}} R_\ell(0,\xi)\hat{p}_k(\xi)\,d\xi\right)C_{\ell,j}^-, \tag{36}$$

$$\hat{\varepsilon}^+\sum_{\ell=1}^{(m+1)^2}\left(\int_{\hat{\Gamma}_{K_F}} \partial_\eta R_\ell(0,\xi)\hat{p}_k(\xi)\,d\xi\right)C_{\ell,j}^+ = \hat{\varepsilon}^-\sum_{\ell=1}^{(m+1)^2}\left(\int_{\hat{\Gamma}_{K_F}} \partial_\eta R_\ell(0,\xi)\hat{p}_k(\xi)\,d\xi\right)C_{\ell,j}^-, \tag{37}$$

$$\hat{\varepsilon}^+\sum_{\ell=1}^{(m+1)^2}\left(\int_{\hat{\Gamma}_{K_F}} \partial_\eta^n\mathcal{L}(R_\ell(0,\xi))\hat{p}_k(\xi)\,d\xi\right)C_{\ell,j}^+ = \hat{\varepsilon}^-\sum_{\ell=1}^{(m+1)^2}\left(\int_{\hat{\Gamma}_{K_F}} \partial_\eta^n\mathcal{L}(R_\ell(0,\xi))\hat{p}_k(\xi)\,d\xi\right)C_{\ell,j}^-, \tag{38}$$

for $k = 0,\dots,m$ and $n = 0,\dots,m-2$.

Introduce the matrices

$$B_{k+1,\ell}^{(0)} = \int_{\hat{\Gamma}_{K_F}} R_\ell(0,\xi)\hat{p}_k(\xi)\,d\xi, \tag{39}$$

$$B_{k+1,\ell}^{(1)} = \int_{\hat{\Gamma}_{K_F}} \partial_\eta R_\ell(0,\xi)\hat{p}_k(\xi)\,d\xi, \tag{40}$$

$$\widetilde{A}_{k+1,\ell}^{(n)} = \int_{\hat{\Gamma}_{K_F}} \partial_\eta^n\mathcal{L}(R_\ell(0,\xi))\hat{p}_k(\xi)\,d\xi, \qquad 0 \le n \le m-2. \tag{41}$$

Stacking these blocks yields

$$\widetilde{A} = \begin{bmatrix} B^{(0)} \\ B^{(1)} \\ \widetilde{A}^{(0)} \\ \vdots \\ \widetilde{A}^{(m-2)} \end{bmatrix}, \qquad J = \mathrm{diag}\left(\underbrace{1,\dots,1}_{m+1}, \underbrace{\frac{\hat{\varepsilon}^-}{\hat{\varepsilon}^+},\dots,\frac{\hat{\varepsilon}^-}{\hat{\varepsilon}^+}}_{m(m+1)}\right). \tag{42}$$

The interface constraints may then be written compactly as

$$\widetilde{A}\mathbf{C}_j^+ = J\widetilde{A}\mathbf{C}_j^-. \tag{43}$$

By unisolvency of the homogeneous interface conditions, $\widetilde{A}$ is nonsingular. Accordingly, one side of a homogeneous GCIFE basis function can be freely chosen within $\mathbb{Q}_m(\hat{K}_F)$, and the component on the opposite side is uniquely determined via (43). All entries of the matrices defined in (42) are evaluated via high-order quadrature along the straightened interface. Integrals of reference polynomials can be precomputed, while geometry-dependent terms arising from the Frenet transformation are evaluated individually for each interface element.

3.2.2. Nonhomogeneous Basis Functions

This subsection constructs nonhomogeneous GCIFE basis functions to accommodate prescribed interface flux and source jumps. Consistent with the homogeneous formulation in §3.2.1, we adopt the same tensor-product basis $\mathcal{R}$ and two-sided piecewise polynomial framework on the Frenet fictitious element $\hat{K}_F$. We emphasize that the nonhomogeneous jump conditions are enforced weakly, not pointwise, since the source jump conditions (18) do not yield polynomial equations for determining the polynomial terms. Instead, the jump of each basis function, when tested against the tangential basis $\{\hat{p}_k\}_{k=0}^m$, matches the corresponding moment of the prescribed interface jump data, that is, $\hat{u}_{J,h}^K$ satisfies the following moment-matching conditions for $0 \le k \le m$ and $0 \le n \le m-2$:

$$\begin{aligned}
&\int_{\hat{\Gamma}_{K_F}} [\![\hat{u}_{J,h}^K]\!] \hat{p}_k \, d\xi = 0, \\
&\int_{\hat{\Gamma}_{K_F}} \left([\![\hat{\varepsilon}\partial_\eta \hat{u}_{J,h}^K]\!] - \hat{g}_\sigma^K \right) \hat{p}_k \, d\xi = 0, \\
&\int_{\hat{\Gamma}_{K_F}} \left([\![\hat{\varepsilon}\partial_\eta^n \mathcal{L}(\hat{u}_{J,h}^K)]\!] - \hat{g}_\rho^{K,n} \right) \hat{p}_k \, d\xi = 0.
\end{aligned} \tag{44}$$

As the continuity jump remains homogeneous, only the flux and higher-order source jumps need lifting, yielding $m(m+1)$ nonhomogeneous basis functions. For $j = 1, \ldots, m(m+1)$, a general nonhomogeneous basis function $\hat{\chi}_j(\eta, \xi)$ defined over $\hat{K}_F$ takes the piecewise form

$$\hat{\chi}_j(\eta,\xi) = \begin{cases} \hat{\chi}_j^-(\eta,\xi) = \displaystyle\sum_{\ell=1}^{(m+1)^2} C_{\ell,j}^- R_\ell(\eta,\xi), & (\eta,\xi) \in \hat{K}_F^-, \\ \hat{\chi}_j^+(\eta,\xi) = \displaystyle\sum_{\ell=1}^{(m+1)^2} C_{\ell,j}^+ R_\ell(\eta,\xi), & (\eta,\xi) \in \hat{K}_F^+. \end{cases} \tag{45}$$

The coefficient vectors $\mathbf{C}_j^{\pm}$ are determined such that $\hat{\chi}_j$ satisfies the nonhomogeneous interface conditions on $\hat{\Gamma}_{K_F}$.

All integral constraint blocks $B^{(0)}$, $B^{(1)}$, $\widetilde{A^{(n)}}$ and the assembled constraint matrix $\widetilde{A}$ are identical to those derived for homogeneous GCIFE basis functions. The key distinction of the nonhomogeneous system lies in an additional jump source term $\mathbf{s}_j$ on the right-hand side, yielding the unified extension equation

$$\widetilde{A}\mathbf{C}_j^+ = J\widetilde{A}\mathbf{C}_j^- + \mathbf{s}_j. \tag{46}$$

Thus the homogeneous and nonhomogeneous GCIFE constructions differ only in the jump-induced source vector $\mathbf{s}_j$. To ensure a unique extension, we set the coefficients on the smaller-volume subdomain of $\hat{K}_F$ to zero and solve for the remaining coefficients on the opposite subdomain. This one-sided zero-coefficient constraint is purely a numerical device for stable polynomial extension and imposes no physical restrictions on the interface jump behavior.

We define the canonical basis vectors $\mathbf{e}_k \in \mathbb{R}^{m+1}$ satisfying

$$(\mathbf{e}_k)_{r+1} = \delta_{kr}, \qquad 0 \le k, r \le m,$$

based on which two families of unit-jump vectors are constructed for basis generation: the *normal flux-jump vectors* $\mathbf{e}_k^\sigma$ and *order-n source-jump vectors* $\mathbf{e}_k^n$. For flux-jump modes with $0 \le k \le m$, the unit flux-jump vector is formulated as

$$\mathbf{e}_k^\sigma = \left[\underbrace{\mathbf{0}_{m+1}}_{1 \text{ block}},\ \mathbf{e}_k,\ \underbrace{\mathbf{0}_{m+1},\ \ldots,\ \mathbf{0}_{m+1}}_{m-1 \text{ blocks}}\right]^T \in \mathbb{R}^{(m+1)^2}. \tag{47}$$

For source-jump modes with $0 \le n \le m-2$ and $0 \le k \le m$, the unit source-jump vector reads

$$\mathbf{e}_k^n = \left[\underbrace{\mathbf{0}_{m+1},\ \ldots,\ \mathbf{0}_{m+1}}_{n+2 \text{ blocks}},\ \mathbf{e}_k,\ \underbrace{\mathbf{0}_{m+1},\ \ldots,\ \mathbf{0}_{m+1}}_{m-1-n \text{ blocks}}\right]^T \in \mathbb{R}^{(m+1)^2}, \tag{48}$$

where the non-zero unit vector $\mathbf{e}_k$ is placed in the block corresponding to the constraint matrix $\widetilde{A}^{(n)}$. Both flux-jump and source-jump basis functions follow the same two-sided polynomial extension strategy.

For the jump basis $\hat{\chi}_j^\sigma$ associated with normal flux jump modes, it corresponds to the global nonhomogeneous basis index $j = k+1$, $k = 0, \ldots, m$. The extension system is given by

$$\mathbf{s}_j^\sigma = \frac{1}{\hat{\varepsilon}^+}\mathbf{e}_k^\sigma, \tag{49}$$

where $\mathbf{s}_j^\sigma$ is the source vector for this mode, occupying the j-th position in the global nonhomogeneous basis enumeration. For the jump basis $\hat{\chi}_j^n$ associated with order-n source jump modes, $0 \le n \le m-2$, it corresponds to the global nonhomogeneous basis index $j = m+1+n(m+1)+k+1, k = 0, \ldots, m$. The extension system is given by

$$\mathbf{s}_j^n = \frac{1}{\hat{\varepsilon}^+}\mathbf{e}_k^n, \tag{50}$$

where $\mathbf{s}_j^n$ is the source vector for this mode, again occupying the j-th position in the global nonhomogeneous basis enumeration.

The constructed unit-jump basis functions $\hat{\chi}_j^\sigma$ and $\hat{\chi}_j^n$ serve as unit-jump templates. The prescribed interface data, i.e., the flux-jump datum $\hat{g}_\sigma^K$ and the higher-order source-jump data $\hat{g}_\rho^{K,n}$, are integrated against the tangential basis to produce the tangential jump moments defined below. These moments are then used directly as the lifting coefficients, rather than the coefficient vectors $\mathbf{C}_j^\pm$.

Over $\hat{K}_F$, define

$$(\mathbf{b}_\sigma^K)_{k+1} = \int_{\hat{\Gamma}_{K_F}} \hat{g}_\sigma^K(\xi)\hat{p}_k(\xi)\,d\xi, \tag{51}$$

$$(\mathbf{b}_{\rho,n}^K)_{k+1} = \int_{\hat{\Gamma}_{K_F}} \hat{g}_\rho^{K,n}(\xi)\hat{p}_k(\xi)\,d\xi, \qquad 0 \le k \le m. \tag{52}$$

The components of these moment vectors serve directly as the lifting coefficients, so that the nonhomogeneous lifting on $\hat{K}_F$ is

$$\hat{u}_{J,h}^K = \sum_{k=0}^{m}(\mathbf{b}_\sigma^K)_{k+1}\hat{\chi}_{k+1}^{K,\sigma} + \sum_{n=0}^{m-2}\sum_{k=0}^{m}(\mathbf{b}_{\rho,n}^K)_{k+1}\hat{\chi}_{k+1}^{K,n}. \tag{53}$$

By construction, $\hat{u}_{J,h}^K$ satisfies the moment-matching conditions (44).

3.2.3. Assembly of $a_h(u_{J,h}, v_h)$

In the final discrete system (30), the assembly of the boundary terms for the stiffness matrix and right-hand-side vector follows [53]; details of the assembly can be found therein. The direct PIC volume source term $\mathcal{D}_{\mathrm{PIC},h}(v_h)$ is presented in Subsection 4.3. This subsection mainly describes the assembly of the nonhomogeneous lifting contribution $a_h(u_{J,h}, v_h)$.

The local basis functions on an interface element $K \in \mathcal{T}_h^i$ are obtained by coupling the basis functions defined on the Frenet fictitious element with the Frenet inverse transformation $R_\Gamma(\mathbf{x})$. In particular, the homogeneous GCIFE basis function λ_j^K is defined by

$$\lambda_j^K(\mathbf{x}) = \hat{\lambda}_j^K(R_\Gamma(\mathbf{x})). \tag{54}$$

For a standard element $K \notin \mathcal{T}_h^i$, the corresponding basis functions are the standard tensor-product Legendre basis functions. Therefore, the global homogeneous basis is composed of standard Legendre basis functions on standard elements and homogeneous GCIFE basis functions λ_j^K on interface elements.

Similarly, the nonhomogeneous basis functions are defined by

$$\chi_{k+1}^{K,\sigma}(\mathbf{x}) = \hat{\chi}_{k+1}^{K,\sigma}(R_\Gamma(\mathbf{x})), \quad \chi_{k+1}^{K,n}(\mathbf{x}) = \hat{\chi}_{k+1}^{K,n}(R_\Gamma(\mathbf{x})). \tag{55}$$

For each interface element K, the physical nonhomogeneous lifting $u_{J,h}^K$ is then defined by coupling $\hat{u}_{J,h}^K$ with $R_\Gamma(\mathbf{x})$:

$$u_{J,h}^K(\mathbf{x}) = \hat{u}_{J,h}^K(R_\Gamma(\mathbf{x})). \tag{56}$$

Equivalently, using the nonhomogeneous basis functions defined above,

$$u_{J,h}^K = \sum_{k=0}^{m} (\mathbf{b}_\sigma^K)_{k+1} \chi_{k+1}^{K,\sigma} + \sum_{n=0}^{m-2} \sum_{k=0}^{m} (\mathbf{b}_{\rho,n}^K)_{k+1} \chi_{k+1}^{K,n}. \tag{57}$$

The global nonhomogeneous lifting is obtained by assembling the local liftings over all interface elements:

$$u_{J,h} = \sum_{K \in \mathcal{T}_h^i} u_{J,h}^K. \tag{58}$$

To assemble $a_h(u_{J,h}, v_h)$, it is sufficient to consider interface elements $K \in \mathcal{T}_h^i$. Since the nonhomogeneous lifting $u_{J,h}$ is supported only on interface elements, the corresponding test functions in this assembly are the homogeneous basis functions λ_j. Standard basis functions on non-interface elements are therefore not involved in this part of the assembly.

Consider first one nonhomogeneous basis function χ associated with an interface element K, extended by zero outside K. For a homogeneous basis function λ_j, its contribution is

$$\begin{aligned} a_h(\chi, \lambda_j) = & \int_K \varepsilon \nabla \chi \cdot \nabla \lambda_j \, \mathrm{d}\mathbf{x} \\ & - \sum_{e \in \mathcal{E}_h^i(K)} \Big(\langle [\![\varepsilon \nabla \chi \cdot \mathbf{n}_e]\!]_e, \{\!\{ \lambda_j \}\!\}_e \rangle_e + \langle \{\!\{ \chi \}\!\}_e, [\![\varepsilon \nabla \lambda_j \cdot \mathbf{n}_e]\!]_e \rangle_e - \frac{\gamma_e \varepsilon_e}{h_e} \langle [\![\chi]\!]_e, [\![\lambda_j]\!]_e \rangle_e \Big) \\ & - \sum_{e \in \mathcal{E}_h^b(K)} \Big(\langle \varepsilon \nabla \chi \cdot \mathbf{n}_e, \lambda_j \rangle_e + \langle \chi, \varepsilon \nabla \lambda_j \cdot \mathbf{n}_e \rangle_e - \frac{\gamma_e \varepsilon_e}{h_e} \langle \chi, \lambda_j \rangle_e \Big). \end{aligned} \tag{59}$$

The volume part of $a_h(\chi, \lambda_j)$ is evaluated separately on the two subdomains of an interface element:

$$\int_K \varepsilon \nabla\chi \cdot \nabla\lambda_j \, \mathrm{d}\mathbf{x} = \int_{K^-} \varepsilon^- \nabla\chi^- \cdot \nabla\lambda_j^- \, \mathrm{d}\mathbf{x} + \int_{K^+} \varepsilon^+ \nabla\chi^+ \cdot \nabla\lambda_j^+ \, \mathrm{d}\mathbf{x}. \tag{60}$$

The functions $\chi^\pm$ and $\lambda_j^\pm$ denote the restrictions of the corresponding interface-element basis functions to $K^\pm = K \cap \Omega^\pm$.

The face contributions are evaluated according to the classification of the faces. For an interface-affected interior face $e \in \mathcal{E}_h^{ii}$ or an interface-affected boundary face $e \in \mathcal{E}_h^{bi}$, two cases are distinguished.

If $e \cap \Gamma = \emptyset$, the face is contained entirely in one subdomain Ω^s, with $s \in \{-, +\}$. In this case, the complete face e is used in the corresponding face integrals. Although the face is not cut by the interface, the traces of the basis functions from an adjacent interface element are still evaluated using the corresponding active pieces of the interface basis functions.

If $e \cap \Gamma \neq \emptyset$, the face is partitioned into

$$e^- = e \cap \Omega^-, \qquad e^+ = e \cap \Omega^+.$$

Each SIPDG face contribution is then evaluated separately on e^- and e^+. For example, the contribution associated with χ and λ_j on a cut interior face is

$$\mathcal{A}_e(\chi, \lambda_j) = \sum_{s\in\{-,+\}} \Big[-\langle [\![\varepsilon^s \nabla\chi^s \cdot \mathbf{n}_e]\!]_{e^s}, \{\!\{ \lambda_j^s \}\!\}_{e^s} \rangle_{e^s} - \langle \{\!\{ \chi^s \}\!\}_{e^s}, [\![\varepsilon^s \nabla\lambda_j^s \cdot \mathbf{n}_e]\!]_{e^s} \rangle_{e^s} + \frac{\gamma_e \varepsilon_e}{h_e} \langle [\![\chi^s]\!]_{e^s}, [\![\lambda_j^s]\!]_{e^s} \rangle_{e^s} \Big]. \tag{61}$$

For an uncut face, the same expression is evaluated once on the complete face e, using the active subdomain s. Thus, an interface-affected face that is not cut is integrated over its full measure, whereas a cut face is integrated separately over its Ω^- and Ω^+ portions.

For every interface element K, the local matrix associated with the nonhomogeneous lifting is defined by

$$\mathbf{M}_J^K = \begin{bmatrix} \mathbf{M}_\sigma^K & \mathbf{M}_{\rho,0}^K & \mathbf{M}_{\rho,1}^K & \cdots & \mathbf{M}_{\rho,m-2}^K \end{bmatrix}. \tag{62}$$

Its entries are the values of the SIPDG bilinear form applied to each nonhomogeneous basis function and each homogeneous basis function:

$$\left(\mathbf{M}_\sigma^K\right)_{j,k+1} = a_h\left(\chi_{k+1}^{K,\sigma}, \lambda_j\right), \qquad k = 0, \ldots, m, \tag{63}$$

$$\left(\mathbf{M}_{\rho,n}^K\right)_{j,k+1} = a_h\left(\chi_{k+1}^{K,n}, \lambda_j\right), \qquad k = 0, \ldots, m, \quad n = 0, \ldots, m-2. \tag{64}$$

Each matrix entry consists of the volume contribution in (60) and the SIPDG face contributions on the faces associated with K.

Using the previously defined coefficients of the nonhomogeneous lifting, the contribution of the local lifting $u_{J,h}^K$ to the j-th interface test equation is

$$a_h(u_{J,h}^K, \lambda_j) = \sum_{k=0}^{m} \left(\mathbf{b}_\sigma^K\right)_{k+1} \left(\mathbf{M}_\sigma^K\right)_{j,k+1} + \sum_{n=0}^{m-2} \sum_{k=0}^{m} \left(\mathbf{b}_{\rho,n}^K\right)_{k+1} \left(\mathbf{M}_{\rho,n}^K\right)_{j,k+1}. \tag{65}$$

Finally, the global contribution is obtained by adding the local contributions from all interface elements:

$$a_h(u_{J,h}, \lambda_j) = \sum_{K \in \mathcal{T}_h^i} a_h(u_{J,h}^K, \lambda_j). \tag{66}$$

Equivalently, the action of the global nonhomogeneous lifting on the global homogeneous basis is assembled from the local matrices $\mathbf{M}_\sigma^K$ and $\mathbf{M}_{\rho,n}^K$ through the standard local-to-global degree-of-freedom mapping.

Remark 2. *For both homogeneous and nonhomogeneous GCIFE basis functions, this work employs the orthogonalization procedure of [53] to enhance orthogonality. Curved-cell integrals are evaluated using geometry-conforming mappings and high-order quadrature rules also consistent with [53].*

4. Consistent High-Order PIC framework

This section introduces a high–order PIC framework self–consistently coupled to the nonhomogeneous GCIFE field solver. Each particle adopts a fixed compactly supported tensor–product shape function specified analytically. Charge density and electrostatic potential are discretized using element–adapted finite-element bases. On interface elements, the Poisson load is assembled directly from particle-shape moments against homogeneous GCIFE tests, while source jumps come from independent one-sided raw Frenet density projections. Particle–field coupling integrates analytical shapes against finite-element bases over overlapping domains. Particle dynamics are advanced by the high–order Boris–SDC time integrator.

4.1. Volume and Surface Particle Shapes

This subsection constructs the d-dimensional volumetric particle shape from a one-dimensional reference-kernel family. Its restriction to the interface is normalized separately with respect to arclength.

Common one-dimensional reference kernel. For an integer $r \geq 1$, define the normalized one-dimensional reference kernel by

$$s_r(x_{ref}) = \begin{cases} C_r(1 - x_{ref}^2)^r, & |x_{ref}| \leq 1, \\ 0, & |x_{ref}| > 1, \end{cases} \tag{67}$$

where the normalization condition requires

$$C_r = \left[\int_{-1}^{1} (1 - x_{ref}^2)^r \, \mathrm{d}x_{ref} \right]^{-1} = \frac{\Gamma\left(r + \frac{3}{2}\right)}{\sqrt{\pi}\,\Gamma(r+1)}. \tag{68}$$

Fixed translated tensor-product volume shape. Let $K_{\mathrm{ref}} = [-1, 1]^d$ denote the dimensionless reference support centered at the origin. The d-dimensional reference particle shape is defined by the tensor product

$$S_{\mathrm{ref}}(\boldsymbol{x}_{ref}) = \prod_{\ell=1}^{d} s_r((\boldsymbol{x}_{ref})_\ell), \qquad \boldsymbol{x}_{ref} \in K_{\mathrm{ref}}, \tag{69}$$

and vanishes outside K_{ref}. The normalization condition is satisfied directly by this construction.

Let $a_\ell > 0$ denote the prescribed physical support half-width in the ℓ-th coordinate direction. These user-defined parameters are fixed at initialization and need not coincide with the Cartesian mesh widths h_ℓ. They may, for example, be selected according to

$$a_\ell = \kappa_\ell h_\ell, \qquad \kappa_\ell > 0, \tag{70}$$

where κ_ℓ is fixed at initialization. No subsequent rescaling of the volumetric particle support is performed during deposition, field gathering, or particle advection.

For a particle centered at

$$\mathbf{x}_p = (x_{p,1}, \ldots, x_{p,d}), \tag{71}$$

the corresponding physical volumetric particle shape is

$$S_p(\mathbf{x}; \mathbf{x}_p) = \prod_{\ell=1}^{d} \frac{1}{a_\ell} s_r\left(\frac{x_\ell - x_{p,\ell}}{a_\ell}\right). \tag{72}$$

Equivalently,

$$S_p(\mathbf{x}; \mathbf{x}_p) = \begin{cases} \displaystyle\prod_{\ell=1}^{d} \frac{C_r}{a_\ell}\left[1 - \left(\frac{x_\ell - x_{p,\ell}}{a_\ell}\right)^2\right]^r, & |x_\ell - x_{p,\ell}| \le a_\ell \quad \text{for every } \ell, \\ 0, & \text{otherwise.} \end{cases} \tag{73}$$

Its physical support is

$$K_p(\mathbf{x}_p) = \prod_{\ell=1}^{d} [x_{p,\ell} - a_\ell,\ x_{p,\ell} + a_\ell]. \tag{74}$$

Increasing a_ℓ broadens the profile without changing r; particle motion changes only its center $\mathbf{x}_p$.

Interface-restricted surface shape. For an absorption event, let t_p^* and $\mathbf{x}_p^* = P_\Gamma(\xi_p^*) \in \Gamma$ denote the first-contact time and position. The surface kernel is constructed directly from the normalized one-dimensional reference kernel s_r defined in Eq. (67), rather than from the restriction of a volumetric particle shape.

Let $a_\ell > 0$ denote the prescribed physical support half-width in the ℓ-th coordinate direction. The physical support half-width of the interface kernel is chosen as

$$h_p = \max_{1 \le \ell \le d} a_\ell. \tag{75}$$

If ξ is an arc-length parameter, so that $\mathrm{d}s = \mathrm{d}\xi$, the surface kernel is defined by

$$S_{\Gamma,p}(P_\Gamma(\xi)) = \frac{1}{h_p} s_r\left(\frac{\xi - \xi_p^*}{h_p}\right). \tag{76}$$

For a general parameterization, let

$$\frac{\mathrm{d}s}{\mathrm{d}\xi} = \left\|P_\Gamma'(\xi)\right\|, \qquad s(\xi) - s(\xi_p^*) = \int_{\xi_p^*}^{\xi} \left\|P_\Gamma'(\zeta)\right\| \,\mathrm{d}\zeta. \tag{77}$$

The corresponding surface kernel is

$$S_{\Gamma,p}(P_\Gamma(\xi)) = \frac{1}{h_p} s_r\left(\frac{s(\xi) - s(\xi_p^*)}{h_p}\right). \tag{78}$$

When evaluated in parameter coordinates, the surface integral uses the physical interface measure

$$\mathrm{d}s = \left\|P_\Gamma'(\xi)\right\| \,\mathrm{d}\xi. \tag{79}$$

4.2. Particle Supports Intersecting Only Standard Elements

Let $\mathcal{P}_{\text{std}}$ and $\mathcal{P}_{\text{int}}$ denote the disjoint sets of particles whose supports intersect only standard elements and at least one interface element, respectively. This subsection treats the contributions of particles in $\mathcal{P}_{\text{std}}$, considered separately from particles in $\mathcal{P}_{\text{int}}$. For particle p, let $\hat{\mathbf{x}}_p$ denote the signed Frenet normal coordinate of its center $\mathbf{x}_p$. More precisely, the side label introduced above is

$$\alpha_p = \text{sign}(\hat{\mathbf{x}}_p) \in \{-, +\}, \tag{80}$$

Then define the accessible portion of the particle support by

$$D_p^{\alpha_p} = \text{supp}(S_p) \cap \Omega^{\alpha_p}. \tag{81}$$

The particle-wise normalization factor is

$$Q_p^{\alpha_p} = \int_{D_p^{\alpha_p}} S_p(\mathbf{x}; \mathbf{x}_p)\, \mathrm{d}\mathbf{x}, \qquad 0 < Q_p^{\alpha_p} \le 1. \tag{82}$$

It is evaluated once over the complete accessible support, not once per element. If the particle support is truncated by the outer computational boundary, only the portion lying on the side containing the particle center and inside the computational domain contributes to the element integrals, and $Q_p^{\alpha_p}$ is evaluated over this retained portion to preserve normalization.

Particle Deposition. Let N_K be the number of local basis functions on K (in two dimensions, $N_K = (m+1)^2$), and let $\mathcal{P}_K$ denote the particles in $\mathcal{P}_{\text{std}}$ whose physical supports intersect K. The contribution of these particles to $\mathcal{D}_{\text{PIC},h}$ is evaluated through their particle–field moments. The set $\{\phi_j^K\}_{j=1}^{N_K}$ serves as the standard orthogonal Legendre basis for finite-element representations on K. The particle shape coupling to the finite-element space is defined directly through the particle–field moments

$$M_{p,j}^K(\mathbf{x}_p) = \int_K S_p(\mathbf{x}; \mathbf{x}_p) \phi_j^K(\mathbf{x})\, \mathrm{d}\mathbf{x}. \tag{83}$$

Since S_p has compact support $K_p(\mathbf{x}_p)$, this integral is equivalently evaluated over the overlap region:

$$M_{p,j}^K(\mathbf{x}_p) = \int_{K \cap K_p(\mathbf{x}_p)} S_p(\mathbf{x}; \mathbf{x}_p) \phi_j^K(\mathbf{x})\, \mathrm{d}\mathbf{x}. \tag{84}$$

For any particle p, let $\mathcal{T}_p^{\text{regular}} := \{K \in \mathcal{T}_h^n : K \cap D_p^{\alpha_p} \neq \emptyset\}$. The corresponding contribution to $\mathcal{D}_{\text{PIC},h}$ is

$$\mathcal{D}_{\text{PIC},h} = \sum_{K \in \mathcal{T}_h^{\text{regular}}} \sum_{p \in \mathcal{P}_K} \frac{q_p}{Q_p^{\alpha_p}} \sum_{j=1}^{N_K} M_{p,j}^K(\mathbf{x}_p). \tag{85}$$

The ordinary-element Legendre basis is orthogonal but not unit-normalized in $L^2(K)$. Define its local mass matrix M^K by

$$\mathsf{M}_{ij}^K := \int_K \phi_i^K(\mathbf{x}) \phi_j^K(\mathbf{x})\, \mathrm{d}\mathbf{x} = 0 \quad (i \neq j), \tag{86}$$

with generally nonunit diagonal entries $\mathsf{M}_{jj}^K > 0$. The corresponding density coefficients R_j^K therefore satisfy

$$\sum_{i=1}^{N_K} \mathsf{M}_{ji}^K R_i^K = \sum_{p \in \mathcal{P}_K} \frac{q_p}{Q_p^{\alpha_p}} M_{p,j}^K(\mathbf{x}_p). \tag{87}$$

The right-hand side is assembled directly from the same particle–field moments that define $\mathcal{D}_{\mathrm{PIC},h}(v_h)$, without an intermediate particle-shape projection. Recovering the density coefficients R_j^K, when needed, requires the local mass matrix, which is diagonal for these ordinary Cartesian elements. When reconstructed, their contribution to the deposited charge density is represented in this finite-element basis:

$$\rho_h^K = \sum_{j=1}^{N_K} R_j^K \phi_j^K. \tag{88}$$

Field Gathering. On a standard element, the discrete potential is represented as

$$u_h^K = \sum_{j=1}^{N_K} U_j^K \phi_j^K, \tag{89}$$

and the corresponding electric field is

$$\mathbf{E}_h^K = -\nabla u_h^K = -\sum_{j=1}^{N_K} U_j^K \nabla \phi_j^K. \tag{90}$$

For $p \in \mathcal{P}_{\mathrm{std}}$, using the same particle shape as in deposition, the field gathered from these standard elements is

$$\mathbf{E}_p^{\mathrm{regular}} = \sum_{K \in \mathcal{T}_p^{\mathrm{regular}}} \frac{1}{Q_p^{\alpha_p}} \int_K S_p(\mathbf{x}; \mathbf{x}_p) \mathbf{E}_h^K(\mathbf{x})\, \mathrm{d}\mathbf{x}. \tag{91}$$

Substituting Eq. (90) gives

$$\mathbf{E}_p^{\mathrm{regular}} = -\sum_{K \in \mathcal{T}_p^{\mathrm{regular}}} \sum_{j=1}^{N_K} U_j^K \mathbf{G}_{p,j}^K(\mathbf{x}_p), \tag{92}$$

where

$$\mathbf{G}_{p,j}^K(\mathbf{x}_p) = \frac{1}{Q_p^{\alpha_p}} \int_K S_p(\mathbf{x}; \mathbf{x}_p) \nabla \phi_j^K(\mathbf{x})\, \mathrm{d}\mathbf{x} \tag{93}$$

is the particle–field gradient moment.

4.3. Particle Supports Intersecting at Least One Interface Element

A particle in $\mathcal{P}_{\mathrm{int}}$ may contribute to both standard and interface elements. The same normalization factor $Q_p^{\alpha_p}$ is used for all of its contributions to deposition and gathering, including those on standard elements. The contributions on standard elements use the ordinary-element basis described in Subsection 4.2. On interface elements, the particle contribution is restricted to the material side containing the particle center.

For an interface element K, let $\mathcal{P}_K$ denote the particles in $\mathcal{P}_{\mathrm{int}}$ whose physical supports intersect K. The implementation evaluates the corresponding overlap integrals by two-dimensional Gaussian quadrature over the intersection of the particle support, the element, and the accessible material side.

Particle Deposition. Let N_K be the number of local homogeneous GCIFE basis functions on K; in two dimensions, $N_K = (m+1)^2$. The set $\{\lambda_j^K\}_{j=1}^{N_K}$ denotes the homogeneous GCIFE basis on the full physical cell K. For $p \in \mathcal{P}_K$, define the one-sided particle–field moment by

$$M_{p,j}^{K,\alpha_p}(\mathbf{x}_p) = \int_{K \cap D_p^{\alpha_p}} S_p(\mathbf{x}; \mathbf{x}_p) \lambda_j^K(\mathbf{x})\, \mathrm{d}\mathbf{x}, \qquad j = 1, \ldots, N_K. \tag{94}$$

Let $\mathcal{T}_p^{\text{interface}} := \{K \in \mathcal{T}_h^i : K \cap D_p^{\alpha_p} \neq \emptyset\}$. Consequently, the contribution of the interface particles to the Poisson load functional is assembled directly from the same particle–field moments:

$$\mathcal{D}_{\text{PIC},h} = \sum_{K\in(\mathcal{T}_h^{\text{interface}},\mathcal{T}_h^{\text{regular}})} \sum_{p\in\mathcal{P}_K} \frac{q_p}{Q_p^{\alpha_p}} \sum_{j=1}^{N_K} M_{p,j}^{K,\alpha_p}(\mathbf{x}_p). \tag{95}$$

Since the homogeneous GCIFE basis is orthonormal and normalized in the full physical-cell inner product,

$$\mathsf{M}_{ji}^K := \int_K \lambda_j^K(\mathbf{x})\lambda_i^K(\mathbf{x})\,\mathrm{d}\mathbf{x} = \delta_{ji}. \tag{96}$$

When density coefficients are needed on an interface element, the full-cell projection is represented in the homogeneous GCIFE basis:

$$\rho_h^K = \sum_{j=1}^{N_K} R_j^K \lambda_j^K. \tag{97}$$

where

$$R_j^K = \sum_{p\in\mathcal{P}_K} \frac{q_p}{Q_p^{\alpha_p}} M_{p,j}^{K,\alpha_p}(\mathbf{x}_p), \qquad j = 1,\dots,N_K. \tag{98}$$

The resulting projection may be nonzero on the material side containing no particle. It represents the full-cell finite-element projection used for the Poisson load and does not replace the separate one-sided reconstruction required for source-jump data.

One-sided density reconstruction for source-jump. The normalized one-sided particle source is defined on the accessible support by

$$\rho_p^{\alpha_p}(\mathbf{x}) = \frac{q_p}{Q_p^{\alpha_p}} S_p(\mathbf{x};\mathbf{x}_p), \qquad \mathbf{x} \in D_p^{\alpha_p}. \tag{99}$$

It produces no volumetric-source contribution on the opposite material side. This one-sided restriction is imposed directly through the integration domain $K \cap D_p^{\alpha_p}$ and therefore requires no separate side-indicator function.

For source jumps, we use the reference tensor-product Frenet basis $\{R_\ell\}$ defined in Eq. (34). Its raw physical pullback on the material portion K^α is

$$\mathcal{R}_{K,\ell}^\alpha(\mathbf{x}) := R_\ell(R_\Gamma(\mathbf{x})), \qquad \mathbf{x} \in K^\alpha. \tag{100}$$

Here "raw" means that these sidewise polynomial functions are used before imposing the coupled GCIFE constraints or applying the homogeneous-basis orthogonalization.

For each $\alpha \in \{-,+\}$, define

$$\mathcal{P}_K^\alpha := \{p \in \mathcal{P}_K : \alpha_p = \alpha\}.$$

The density is reconstructed independently on each material side. The side-specific coefficients satisfy

$$\sum_r \mathsf{M}_{K,\ell r}^\alpha c_{K,r}^\alpha = b_{K,\ell}^\alpha, \tag{101}$$

where

$$\mathsf{M}_{K,\ell r}^\alpha = \int_{K^\alpha} \mathcal{R}_{K,\ell}^\alpha(\mathbf{x})\mathcal{R}_{K,r}^\alpha(\mathbf{x})\,\mathrm{d}\mathbf{x}, \tag{102}$$

$$b^{\alpha}_{K,\ell} = \sum_{p\in\mathcal{P}^{\alpha}_{K}} \frac{q_p}{Q^{\alpha}_{p}} \int_{K\cap D^{\alpha}_{p}} S_p(\mathbf{x};\mathbf{x}_p)\mathcal{R}^{\alpha}_{K,\ell}(\mathbf{x})\,\mathrm{d}\mathbf{x}. \tag{103}$$

The independent one-sided reconstructions are

$$\rho^{K,\alpha}_{\mathrm{raw}} = \sum_{r} c^{\alpha}_{K,r}\mathcal{R}^{\alpha}_{K,r}, \qquad \alpha\in\{-,+\}. \tag{104}$$

These define the piecewise function ρ^{K}_{raw} on $K^{-}\cup K^{+}$. Its interface traces and normal derivatives provide the deposited source-jump data:

$$g^{K,n}_{\rho} = -[\partial^{n}_{\eta}\rho^{K}_{\mathrm{raw}}]_{\Gamma\cap K}, \qquad n = 0,\dots,m-2. \tag{105}$$

Thus, the homogeneous GCIFE projection supplies the Poisson load moments, whereas the independent raw-basis projections supply the one-sided source traces used to construct the jump data.

Field Gathering. The same normalized one-sided kernel is used for field gathering. For an interface element K, define the particle–GCIFE gradient moment by

$$\mathbf{G}^{K}_{p,j} = \frac{1}{Q^{\alpha_p}_{p}} \int_{K\cap D^{\alpha_p}_{p}} S_p(\mathbf{x};\mathbf{x}_p)\nabla\phi^{K}_{j}(\mathbf{x})\,\mathrm{d}\mathbf{x}. \tag{106}$$

The electric fields are represented as

$$\mathbf{E}^{K}_{h} = -\nabla\left(\sum_{j=1}^{(m+1)^2} U^{K}_{j}\phi^{K}_{j} + u^{K}_{J,h}\right). \tag{107}$$

The electric-field contribution gathered from interface element K is

$$\mathbf{E}^{K}_{p} = -\sum_{j=1}^{(m+1)^2} U^{K}_{j}\mathbf{G}^{K}_{p,j} - \frac{1}{Q^{\alpha_p}_{p}} \int_{K\cap D^{\alpha_p}_{p}} S_p(\mathbf{x};\mathbf{x}_p)\nabla(u^{K}_{J,h})(\mathbf{x})\,\mathrm{d}\mathbf{x}. \tag{108}$$

The total interface-element contribution is

$$\mathbf{E}^{\mathrm{interface}}_{p} = \sum_{K\in\mathcal{T}^{\mathrm{interface}}_{p}} \mathbf{E}^{K}_{p}. \tag{109}$$

For $p\in\mathcal{P}_{\mathrm{int}}$, the standard-element contribution $\mathbf{E}^{\mathrm{regular}}_{p}$ is given by Eq. (91) with the integral restricted to $K\cap D^{\alpha_p}_{p}$ and S_p divided by the same $Q^{\alpha_p}_{p}$. The full gathered field and force are

$$\mathbf{E}_p = \mathbf{E}^{\mathrm{interface}}_{p} + \mathbf{E}^{\mathrm{regular}}_{p}, \qquad \mathbf{F}_p = q_p\mathbf{E}_p. \tag{110}$$

Conservative surface-charge update and interface load. Let $\mathcal{P}^{n}$ be the set of first-contact absorption events on Γ during the accepted step $[t_n, t_{n+1}]$. Using the normalized surface kernels from Subsection 4.1. If the kernel support is truncated by the finite interface, define

$$Q_{\Gamma,p} = \int_{\hat{\Gamma}\cap\{|s(\xi)-s(\xi^{*}_{p})|\le h_p\}} \frac{1}{h_p} s_r\left(\frac{s(\xi)-s(\xi^{*}_{p})}{h_p}\right)\mathrm{d}s. \tag{111}$$

and replace the surface kernel by

$$S_{\Gamma,p}(\xi) = \frac{1}{Q_{\Gamma,p}h_p} s_r\left(\frac{s(\xi)-s(\xi^{*}_{p})}{h_p}\right). \tag{112}$$

Then define the unprojected charge increment $\Delta\sigma_\Gamma^{n+1}$ by

$$\Delta\sigma_\Gamma^{n+1}(\mathbf{x}) = \sum_{p\in\mathcal{P}^n} q_p S_{\Gamma,p}(R_\Gamma(\mathbf{x})), \qquad \mathbf{x}\in\Gamma, \tag{113}$$

For an interface element K, let $\Gamma_K = \Gamma \cap K$ be its physical interface segment. Its tangential basis is the Legendre polynomials of Eq. (31). The basis uses the fixed virtual interval $I_K = [\xi - \xi_{\mathrm{mid}}, \xi + \xi_{\mathrm{mid}}]$ even when integrated over Γ_K. Write the projected increment as $\Delta\sigma_\Gamma^{n+1}|_{\Gamma_K} = \sum_{j=0}^m \Delta c_j^{K,n+1}\hat{p}_j^K(R_\Gamma(\mathbf{x}))$ and $\Delta c_j^{K,n+1}$ are determined by

$$\begin{aligned}
\sum_{j=0}^m \mathsf{M}_{jk}^{\Gamma_K}\Delta c_j^{K,n+1} &= b_k^{K,n+1}, \qquad 0\le k\le m,\\
\mathsf{M}_{jk}^{\Gamma_K} &:= \int_{\Gamma_K} \hat{p}_j^K(\xi)\hat{p}_k^K(\xi)\,ds,\\
b_k^{K,n+1} &:= \sum_{p\in\mathcal{P}^n} q_p \int_{\Gamma_K\cap\mathrm{supp}(S_{\Gamma,p})} S_{\Gamma,p}(\xi)\hat{p}_k^K(\xi)\,ds.
\end{aligned} \tag{114}$$

The mass matrix $\mathsf{M}_{jk}^{\Gamma_K}$ is formed over the entire physical segment and can be factored once for a fixed mesh; only the particle integrals are restricted to their support intersections. Thus a particle may contribute to several elements with the same $Q_{\Gamma,p}$.

With absorption as the only surface-charge mechanism, the update is

$$\sigma_\Gamma^{n+1} = \sigma_\Gamma^n + \Delta\sigma_\Gamma^{n+1}. \tag{115}$$

With the Gauss-law surface-charge convention, the flux jump on the physical interface is

$$g_\sigma^{n+1} = -\sigma_\Gamma^{n+1}. \tag{116}$$

Let $c_j^{K,n}$ denote the accumulated surface coefficients, updated by $c_j^{K,n+1} = c_j^{K,n} + \Delta c_j^{K,n+1}$. Extend the corresponding local polynomial to the full virtual interval: $\hat{\sigma}_\Gamma^{K,n+1}(\xi) := \sum_{j=0}^m c_j^{K,n+1}\hat{p}_j^K(\xi)$, $\xi\in I_K$. The flux-jump moments supplied to Eq. (53) are

$$\begin{aligned}
(\mathbf{b}_\sigma^{K,n+1})_{k+1} &= -\int_{I_K} \hat{\sigma}_\Gamma^{K,n+1}(\xi)\hat{p}_k^K(\xi)\,d\xi = -\sum_{j=0}^m \mathsf{T}_{kj}^K c_j^{K,n+1},\\
\mathsf{T}_{kj}^K &:= \int_{I_K} \hat{p}_k^K(\xi)\hat{p}_j^K(\xi)\,d\xi = \frac{2\xi_{h,K}}{2k+1}\delta_{kj},
\end{aligned} \tag{117}$$

where δ_{kj} is the Kronecker delta. The fixed matrix T^K converts projection coefficients to the required $d\xi$ moments. Overlap between virtual intervals does not duplicate physical charge, which is counted only on the physical segments.

4.4. High-Order Particle Advancement with Boris–SDC

Particle trajectories are advanced by the Boris–SDC method following Ref. [52]. Boris–SDC combines the classical Boris integrator with spectral deferred corrections (SDC). The Boris scheme, a second-order velocity-Verlet-type integrator for the Lorentz force, serves as the low-order sweeper. SDC iteratively solves the collocation system

$$\mathbf{U} - \mathbf{Q}_{\mathrm{coll}}\mathbf{F}(\mathbf{U}) = \mathbf{C}_{\mathrm{coll}}\mathbf{U}_0, \tag{118}$$

which is the discrete form of the Picard integral equations

$$\mathbf{v}(t) = \mathbf{v}_0 + \int_{t_n}^{t} \mathbf{f}(\mathbf{x}(s), \mathbf{v}(s))\,\mathrm{d}s, \qquad \mathbf{x}(t) = \mathbf{x}_0 + \int_{t_n}^{t} \mathbf{v}(s)\,\mathrm{d}s. \tag{119}$$

Here t_n is the start of the time step, M is the number of collocation nodes, $\mathbf{U}$ collects the stage values of $\mathbf{x}$ and $\mathbf{v}$, and $\mathbf{U}_0$ collects their initial values. The vector $\mathbf{F}$ contains the corresponding right-hand sides $(\mathbf{v}, \mathbf{f})$, where $\mathbf{f}$ is the Lorentz acceleration. The matrices $\mathbf{Q}_{\mathrm{coll}}$ and $\mathbf{C}_{\mathrm{coll}}$ are the quadrature matrices that approximate the integrals and the initial-value contributions, respectively.

Each SDC sweep improves the stage values from iterate k to $k+1$ by applying the Boris update from node i to node $i+1$. For the velocity, we solve

$$\frac{\mathbf{v}_{i+1}^{k+1} - \mathbf{v}_i^{k+1}}{\Delta\tau_{i+1}} = \frac{q_s}{m_s}\left[\mathbf{E}_{i+1/2}^{k+1} + \frac{\mathbf{v}_{i+1}^{k+1} + \mathbf{v}_i^{k+1}}{2} \times \mathbf{B}_0\right] + \mathbf{e}_i^k, \tag{120}$$

where τ_i and τ_{i+1} are consecutive collocation times, $\Delta\tau_{i+1} = \tau_{i+1} - \tau_i$ is their separation, and q_s/m_s is the charge-to-mass ratio for species s. The field $\mathbf{B}_0$ is prescribed and static, while $\mathbf{E}_{i+1/2}^{k+1} := (\mathbf{E}_i^{k+1} + \mathbf{E}_{i+1}^{k+1})/2$ is the average of the gathered endpoint fields in the Boris sweep. The term $\mathbf{e}_i^k$ is a known correction assembled from the previous iterate; it measures the mismatch between the low-order quadrature used in the sweep and the higher-order collocation quadrature. Since $\mathbf{e}_i^k$ does not depend on the unknowns $\mathbf{v}_{i+1}^{k+1}$, the velocity update retains the same structure as the classical Boris step and can be solved explicitly by the Boris rotation.

The position at the next node is obtained from the explicit quadrature formula

$$\mathbf{x}_{i+1}^{k+1} = \mathbf{x}_i^{k+1} + \Delta\tau_{i+1}\mathbf{v}_0 + \sum_{l=1}^{i} \widetilde{s}_{i+1,l}^{x}\left(\mathbf{f}_l^{k+1} - \mathbf{f}_l^k\right) + \sum_{l=1}^{M} s_{i+1,l}^{x}\mathbf{f}_l^k, \tag{121}$$

where $\mathbf{x}_i^{k+1}$ is the position at node i and iterate $k+1$, $\mathbf{v}_0$ is the initial velocity, and $\mathbf{f}_l^k = \mathbf{f}(\mathbf{x}_l^k, \mathbf{v}_l^k)$ is the acceleration due to the Lorentz force at node l and iterate k. The coefficients $\widetilde{s}_{i+1,l}^{x}$ and $s_{i+1,l}^{x}$ are, respectively, the subinterval weights of the velocity-Verlet position preconditioner and the double-integrated collocation quadrature. Both include the squared time-step scaling and are precomputed from the nodes; they are independent of the particle state. All terms on the right-hand side are already known, so the position update is fully explicit. Each sweep costs roughly one Boris step per node, while the SDC iteration increases the temporal order by about two per sweep, up to the order of the underlying collocation method.

Remark 3 (Particle treatment at the interface and computational boundary)**.** *During each SDC sweep, the field is recomputed at the required stage nodes using the particles present at t_n and the fixed surface charge $\sigma_{\Gamma,h}^n$. Each particle is advanced through the full step $[t_n, t_{n+1}]$. If a trial particle position crosses the interface, its closest point on the interface is used for field deposition and gathering only. The particle's trajectory is otherwise unchanged. No particle is removed and no surface charge is updated during the sweeps.*

After all sweeps are complete, piecewise cubic Hermite trajectories are constructed from the positions and velocities at consecutive distinct nodes. These trajectories are used to identify particles that hit Γ during the step and to determine their first contact times t_p^* and positions $\mathbf{x}_p^*$. Each such particle is counted once. All hits are then

processed together: the hit particles are removed, their contributions are accumulated in Eq. (114)– (117)*, and the surface charge is updated once, using one local projection solve per affected element. Surviving particles retain their full-step SDC positions and velocities, subject to the prescribed particle boundary conditions at* $\partial\Omega$*. Since the computational boundary is aligned with the coordinate axes, these boundary conditions can be handled coordinate-wise using standard PIC procedures [1].*

Finally, the volume load, lifting, and field are recomputed at t_{n+1} *using the surviving particles and the updated surface charge. Thus, for a fixed number of nodes and SDC sweeps, the number of field solves per step does not depend on the number of particle hits.*

5. Numerical Experiments and Validations

The tests proceed from the nonhomogeneous GCIFE field solver (Section 5.1), through charge projection, and field gathering (Section 5.2), to a statistically steady sheath simulation (Section 5.3). The test results of Section 5.1 and Section 5.2 will verify the accuracy and convergence characteristics of each key module of the algorithm. The test results of Section 5.3 will compare the impact of traditional methods with existing methods and orders.

5.1. Field Accuracy and Conditioning of Nonhomogeneous GCIFE on a Curved Interface

The computational domain is $\Omega = (0,1)^2$. The interface is the ellipse

$$\Gamma = \{(x,y) : \phi(x,y) = 0\}, \qquad \phi(x,y) = \frac{(x-c_x)^2}{a^2} + \frac{(y-c_y)^2}{b^2} - 1, \quad a = 0.31, \quad b = 0.23, \tag{122}$$

For the $p = 3$ convergence series in Figure 2, $(c_x, c_y) = (13/24, 1/2)$; all other field results in this subsection use $(c_x, c_y) = (1/2, 1/2)$. The material subdomains are $\Omega^- = \{\phi < 0\}$ and $\Omega^+ = \{\phi > 0\}$. The unit normal $\mathbf{n} = \nabla\phi/|\nabla\phi|$ points from the interior to the exterior. We take $\kappa^- = 1$ and $\kappa^+ = 1000$, with the closed-form manufactured solution

$$q(x,y) = \sin(\pi x)\sin(\pi y), \qquad u_{\text{exact}}(x,y) = \begin{cases} q(x,y), & (x,y) \in \Omega^-, \\ q(x,y) + \phi(x,y), & (x,y) \in \Omega^+. \end{cases} \tag{123}$$

The solution is continuous across Γ, while its normal flux has a nonzero jump. With $S = 2(a^{-2} + b^{-2})$, the piecewise source is

$$f^- = 2\pi^2\kappa^- q, \qquad f^+ = \kappa^+(2\pi^2 q - S). \tag{124}$$

The prescribed flux jump is

$$g_\sigma = (\kappa^+ - \kappa^-)\partial_{\mathbf{n}} q + \kappa^+ \partial_{\mathbf{n}}\phi. \tag{125}$$

The source-jumps are

$$g_\rho^0 = -2\pi^2(\kappa^+ - \kappa^-)q + \kappa^+ S, \qquad g_\rho^n = -2\pi^2(\kappa^+ - \kappa^-)\partial_{\mathbf{n}}^n q, \quad n = 0, \ldots, m-2. \tag{126}$$

The mesh is a uniformly refined Cartesian mesh with $N = 6, 9, 12, 18, 24$ at every polynomial degree $p = 1, \ldots, 5$. We use weak Dirichlet enforcement and penalty $30p(p+1)/2$.

We report the global L^2 error and the broken H^1 seminorm error of $u_h + u_J$ against u_{exact} for every combination of $p = 1, \ldots, 5$ and $N = 6, 9, 12, 18, 24$. Figure 2 shows the computed errors as markers and the ordinary least-squares

fits as lines. Figure 3 also shows where the field errors occur on the common 24×24 mesh. All panels share one logarithmic color scale and use the physical solution $u_h + u_J$. For GCIFE with $p = 1, \ldots, 5$, the fitted L^2 slopes are $2.32, 2.80, 3.81, 5.17$, and 6.18, while the corresponding broken H^1 slopes are $1.28, 2.01, 2.72, 4.13$, and 5.10, respectively. These results are consistent with the optimal convergence rates of order $p+1$ in the L^2 norm and order p in the broken H^1 norm. Here we also compare against the nonhomogeneous bilinear IFE [14], a traditional IFE method that replaces the curved interface with straight-line segments; this control gives slopes of 1.67 and 0.87, compared with its expected asymptotic rates of 2 and 1, respectively. This slight reduction is attributed to the pre-asymptotic behavior on the relatively coarse meshes used in the fit. The curved interface and the associated interface-element contributions are not yet sufficiently resolved for the asymptotic error terms to dominate. Also, this indicates that, even at the lowest order, GCIFE produces markedly smaller errors than standard bilinear IFE, owing to its pointwise exact enforcement of jump conditions on the geometric interface.

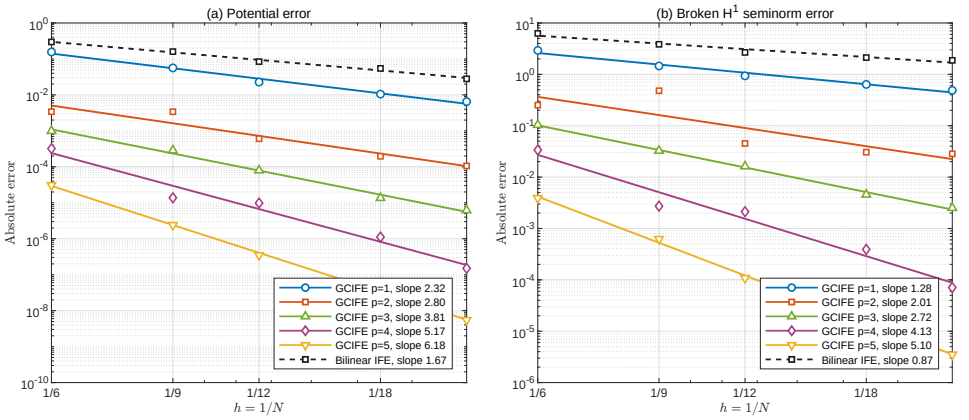


Fig. 2: Elliptical-interface convergence for bilinear IFE and GCIFE degrees $p = 1, \ldots, 5$ on five meshes: (a) L^2 potential error and (b) broken H^1 seminorm error.

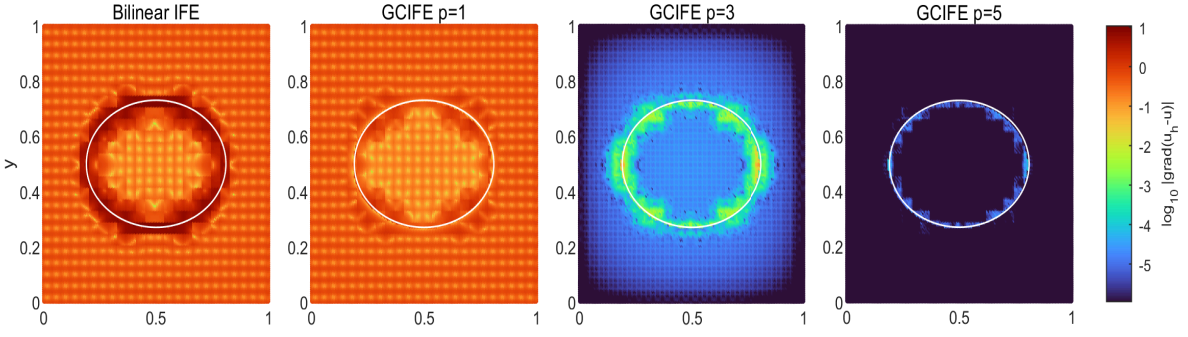


Fig. 3: Pointwise absolute gradient error for the elliptical-interface problem on the 24×24 mesh. Panels show bilinear IFE and GCIFE degrees $p = 1, 3, 5$ using a common logarithmic color scale. The white curve is the exact interface. Integrated potential and field errors and refinement fits appear in Figure 2.

5.2. Component Tests for Charge Representation and Field Gathering

This subsection validates three components of the GCIFE–PIC algorithm in separate tests: one-sided projection of volume and interface charge, particle–field gathering. The deposition and gathering tests use the elliptical interface and material contrast described in Section 5.1.

5.2.1. Volume and interface charge deposition

The Cartesian mesh is refined with $N = 6, 9, 12, 18, 24$ cells per coordinate direction as well. For the volume-density panel, particles form a uniform grid with spacing $h/2$, so this panel combines mesh refinement with particle-source refinement. For the interface-charge panel, we instead keep the same six physical events, charges, and kernel fixed on every mesh; this isolates the polynomial-space effect from event resampling. The six events lie at ellipse parameters $\theta_j = (-2.4, -1.2, -0.2, 0.8, 1.6, 2.4)$ with charges $q_j = 1/6$. We use the tensor-product kernel shape with half-width $(0.08, 0.08)$ and shape order 6. Polynomial degrees $p = 1, \dots, 5$ are considered.

The bilinear-IFE volume control uses Cartesian cloud-in-cell (CIC) deposition. The classical interface control is constant on each physical interface segment Γ_k. Thus it retains the deposited segment charge and distributes that charge uniformly in physical arclength. It uses the same normalized event kernels as GCIFE, without a higher-degree fit on the segment. Cartesian CIC, however, uses its grid-dependent linear particle weights rather than the fixed order-6 reference kernel and its one-sided normalization. Its error against the common kernel-sum target therefore includes particle-shape and interface-truncation differences as well as grid-transfer error.

To distinguish reconstruction error from particle sampling error, the target for the volume test is the normalized discrete particle kernel sum $\rho_{\mathcal{P}}$ on the material side of each point, and the target for the interface test is the corresponding discrete event-kernel sum $\sigma_{\mathcal{E}}$. For GCIFE, ρ_h denotes the diagnostic density formed by Legendre projection on ordinary elements and the independent one-sided raw Frenet reconstructions of Eq. (104) on cut elements. It is distinct from the full-cell homogeneous-GCIFE projection in Eq. (97), which may extend onto the opposite material side. We report the relative errors

$$e_\rho = \frac{\|\rho_h - \rho_{\mathcal{P}}\|_{L^2(\Omega)}}{\|\rho_{\mathcal{P}}\|_{L^2(\Omega)}}, \qquad e_\sigma = \frac{\|\sigma_h - \sigma_{\mathcal{E}}\|_{L^2(\Gamma)}}{\|\sigma_{\mathcal{E}}\|_{L^2(\Gamma)}}. \tag{127}$$

The discrete total-charge defect is normalized by the total absolute particle or event charge.

Figure 4 shows the relative L^2 density errors and five-mesh log–log fits. The fitted slopes for the volume-density test are 1.34 for CIC with bilinear IFE and $2.09, 3.77, 4.50, 5.61, 6.90$ for GCIFE $p = 1, \dots, 5$. For the fixed six-event interface-charge test, the corresponding slopes are 0.55 for the arc-length-mean control and $1.24, 1.79, 2.63, 3.50, 4.31$ for GCIFE $p = 1, \dots, 5$. Figure 5 shows the spatial errors at $N = 24$ for the same source-refined particle lattice and fixed six-event interface source used in Figure 4. These results show that GCIFE–PIC provides a systematically improvable high-order representation of both volume and interface charge. Increasing the GCIFE degree produces steeper mesh-refinement trends and substantially reduces the spatial errors, including those concentrated near the curved interface. In particular, the interface-charge oscillations visible for the low-order controls are progressively suppressed as p increases. At the same time, charge is conserved to roundoff accuracy for all GCIFE degrees. There-

fore, compared with the classical CIC and arc-length-mean controls, GCIFE–PIC achieves higher accuracy and faster observed convergence without compromising discrete charge conservation.

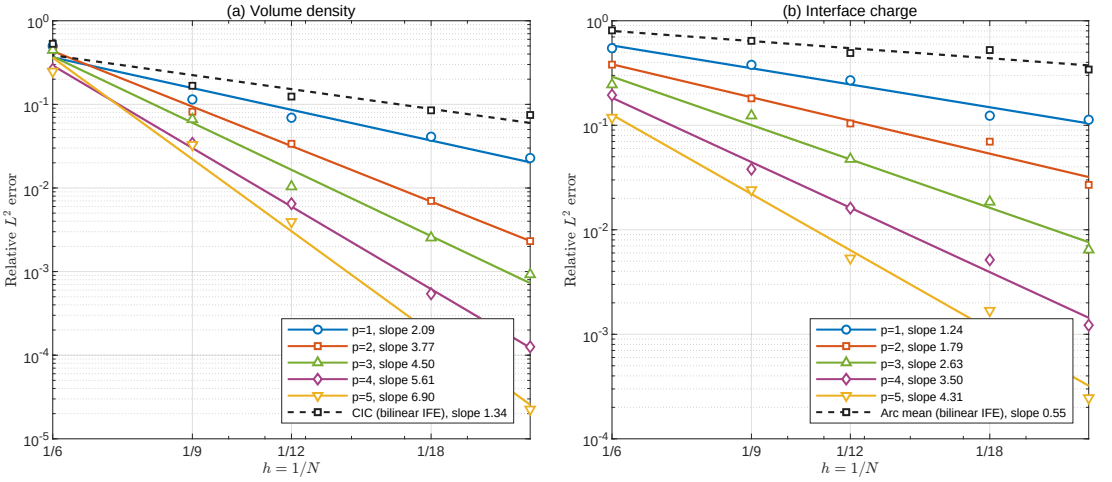


Fig. 4: Relative L^2 density errors for classical deposition controls and GCIFE degrees $p = 1, \ldots, 5$ on $N = 6, 9, 12, 18, 24$. (a) Volume density, using CIC for the bilinear-IFE control and a particle lattice refined alongside the mesh. (b) Interface charge, using segmentwise arc-length averages for the control, for one fixed group of six events, with unchanged physical positions, charges, and kernel on every mesh. Markers show computed errors and lines show five-mesh log–log fits. Charge conservation is stated in the text.

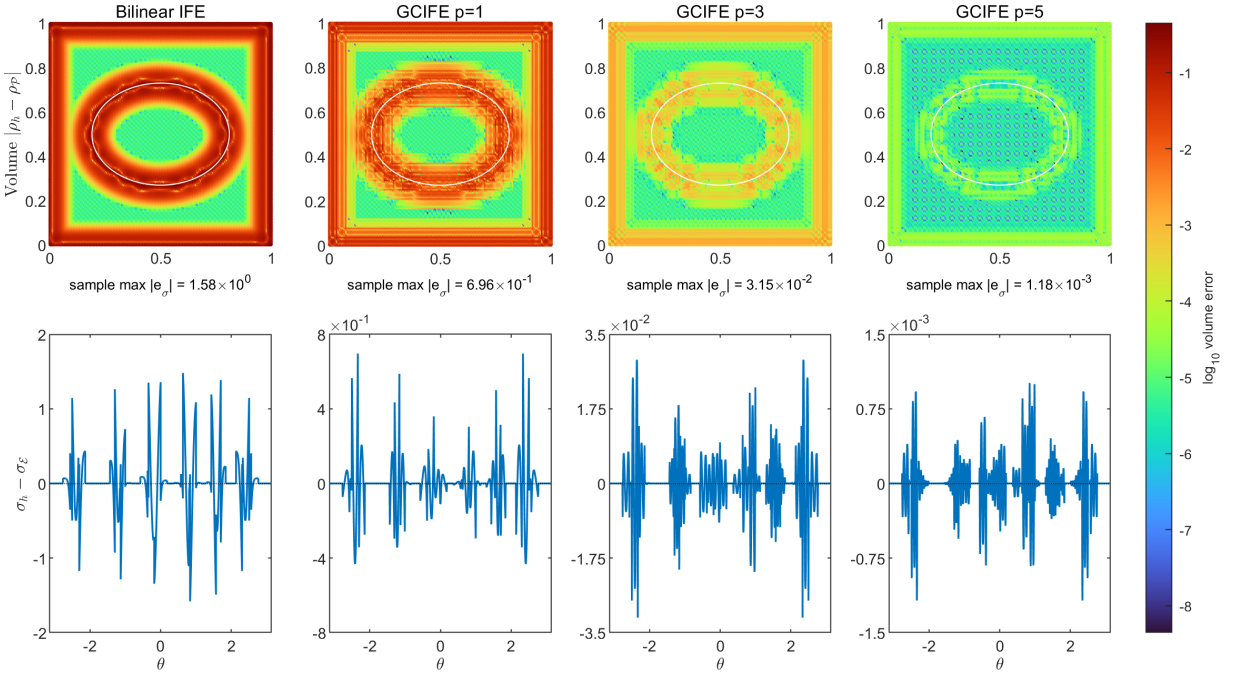


Fig. 5: Spatial density errors at $N = 24$. The bilinear-IFE column uses CIC for volume charge and segmentwise arclength averages for interface charge. The remaining columns show GCIFE degrees $p = 1, 3, 5$. The top row displays $|\rho_h - \rho_{\mathcal{P}}|$ for the source-refined particle lattice at fixed cell-interior points with a common logarithmic color scale. The white curve is the exact interface. The bottom row displays the signed difference $\sigma_h - \sigma_{\mathcal{E}}$ along the interface parameter θ, with an independent symmetric vertical scale in each panel. Axis multipliers and annotated sample maxima indicate the actual error magnitudes. The bottom row uses the same fixed six-event group as Fig. 4. Integrated errors appear in Fig. 4(b).

5.2.2. *Single-particle field gathering*

We assess field gathering using one unit-charge particle held at $(x_p, y_p) = (0.6449, 0.7162)$ on the exterior side of the ellipse on all five meshes. All fields are computed from the manufactured potential of Section 5.1, with ellipse center $(1/2, 1/2)$. The gathered field $\mathbf{E}_h$ (GCIFE kernel average or CIC for the bilinear-IFE control) is compared with the analytic reference field $\mathbf{E}_{\text{ana}}$ through the relative error

$$e_F = \frac{\|\mathbf{E}_h - \mathbf{E}_{\text{ana}}\|_2}{\|\mathbf{E}_{\text{ana}}\|_2}, \tag{128}$$

Figure 6 shows the comparison results. The CIC errors increase from 0.0811 at $N = 6$ to 0.4739 at $N = 24$; their five-mesh log–log fit has a slope of -1.15. The GCIFE slopes remain $0.58, 2.61, 2.62, 5.51, 5.81$ for $p = 1, \ldots, 5$. The negative CIC slope describes this finite mesh range, rather than an asymptotic order. Even on the finest mesh ($N = 24$), the particle-containing cell intersects the dielectric interface, so the CIC stencil samples both material sides and smears the electric-field jump, contributing substantially to the force error. By averaging the field over the particle shape while respecting the one-sided field representation, the GCIFE–PIC gathering operator avoids the cross-interface smearing inherent in the conventional CIC stencil and thus achieves high-order, systematically decreasing force errors near material interfaces, being several orders of magnitude more accurate than the bilinear-IFE–CIC result at $N = 24$.

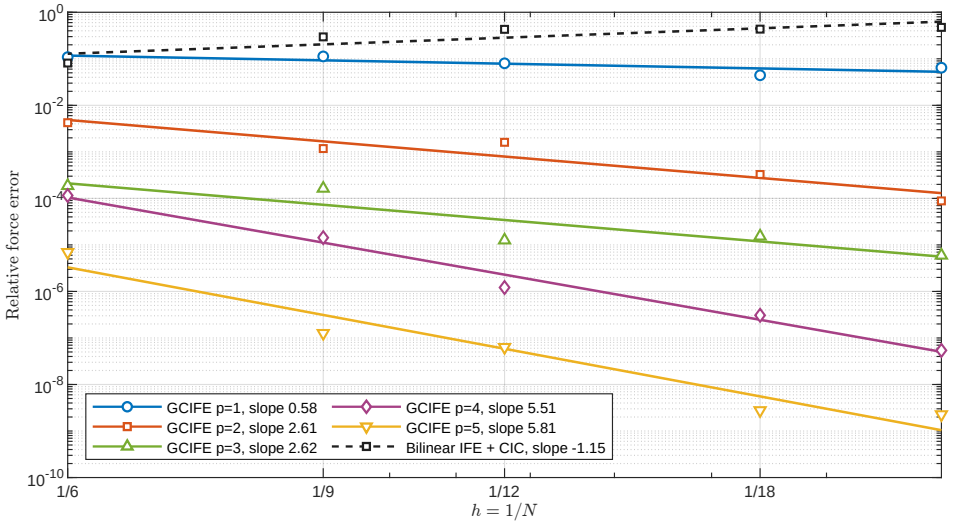


Fig. 6: Single-particle field-gather error against the common analytic kernel-average reference on $N = 6, 9, 12, 18, 24$. The bilinear-IFE control uses four-node CIC with a centered-difference nodal field; GCIFE $p = 1, \ldots, 5$ uses the fixed compact kernel. The particle position is fixed. Markers are computed errors and lines are five-mesh log–log fits. The CIC fit is negative in this mesh range, and the method-dependent gathering operators are specified in the text.

5.3. *Paired Comparisons of Particle Collection and Interface Charging*

Without an analytical solution, this test assesses physical plausibility through the key field and charging features, compares GCIFE–PIC with conventional bilinear IFE–PIC, and examines the advantages of GCIFE–PIC. The domain is $\Omega = (0, 1)^2$, with periodic lateral boundaries, an upper reservoir held at $u = 0$, and a lower electrode held at

$u = -\log 2$. The exposed plasma–dielectric interface is the part above $y = 0$ of

$$\left(\frac{x - 0.517}{0.281}\right)^2 + \left(\frac{y - 0.14}{0.20}\right)^2 = 1,$$

with dielectric and plasma permittivities $\epsilon^- = 10$ and $\epsilon^+ = 1$. Particles are absorbed at the interface and lower electrode and removed at the upper boundary; charge collected on the dielectric enters the Poisson flux jump. Every 0.16 time units, 10 ion and 10 electron computational particles are injected at $y = 0.97$, with uniformly spaced positions in x shifted between injection times. Their represented charges are $+0.1$ and -0.1, and their charge-to-mass ratios are $1/16$ and -1, respectively. Ions enter downward at speed 0.25; electron normal speeds are deterministic midpoint quantiles of the normalized incoming flux distribution $f(v) = v\exp(-v^2/2)$, $v > 0$. For GCIFE-PIC, the tensor-product kernel shape with half-width $(0.06, 0.06)$ and shape order 6 are used.

Following the accuracy trends in Sections 5.1 and 5.2, we compare bilinear $N = 24$ with GCIFE $p = 1, N = 12$, and bilinear $N = 96$ with GCIFE $p = 3, N = 6$. For $h = 1/N$, the benchmark uses the reference CFL step $\Delta t_{\mathrm{CFL}} = 0.48h$. The bilinear and GCIFE $p = 1$ cases use Boris advancement with $\Delta t = \frac{1}{2}\Delta t_{\mathrm{CFL}}$, giving 0.01, 0.0025, and 0.02 for bilinear $N = 24$, bilinear $N = 96$, and GCIFE $p = 1, N = 12$, respectively. GCIFE $p = 3$ uses the formally fourth-order, three-node Boris–SDC method with two sweeps [52]; motivated by its higher temporal accuracy, we adopt a fourfold enlargement of the half-CFL step, $\Delta t = 4 \times \frac{1}{2}\Delta t_{\mathrm{CFL}} = 0.16$ at $N = 6$.

All cases start from an empty domain and pass the blockwise interface-current stationarity test over a 40.96-time-unit window. The checkpoint of steady state is confirmed by a balance test on the projected interface-charge currents, requiring the mean current to lie within 2 standard errors of zero for the total current, and within 3.5 standard errors for every segment and mode. Figures 7 and 8 average the final 40.96 time units ending at each case's first passing checkpoint.

Figure 7 shows potential (top row) and net charge density (bottom row) at the steady state. GCIFE $p = 1$, GCIFE $p = 3$, and bilinear $N = 96$ all capture the potential drop toward the lower electrode, its deformation around the dielectric, and the predominantly positive space-charge region below the inlet. Their interface charge varies along the curved surface without the pronounced, localized alternating concentrations seen for bilinear $N = 24$ (See Figure 8(a)). These concentrations occur near the high-curvature ends of the ellipse's major axis and coincide with the potential depressions in Figure 7(a), indicating inadequate near-interface resolution. Bilinear refinement to $N = 96$ suppresses these distortions; GCIFE captures the principal field and charging structures already on coarse meshes. The three configurations therefore give qualitatively reasonable results, with GCIFE requiring substantially fewer cells. Even at $p = 1$, the GCIFE basis construction accounts for the exact interface geometry pointwise. This geometric fidelity enables GCIFE–PIC at $N = 12$ to capture the key interface effects on a mesh coarser than the bilinear $N = 24$ mesh, despite its low polynomial degree.

Table 1 compares single-thread MATLAB runtimes through each first steady-state pass. GCIFE $p = 1$ improves the interface representation but has a higher accumulated cost than bilinear $N = 24$. The stronger computational advantage appears for GCIFE $p = 3, N = 6$: its larger step reduces the accumulated runtime from 1464.64 s for bilinear $N = 96$ to 941.24 s, a 35.7% reduction, despite a higher per-step deposition cost. This cost comparison

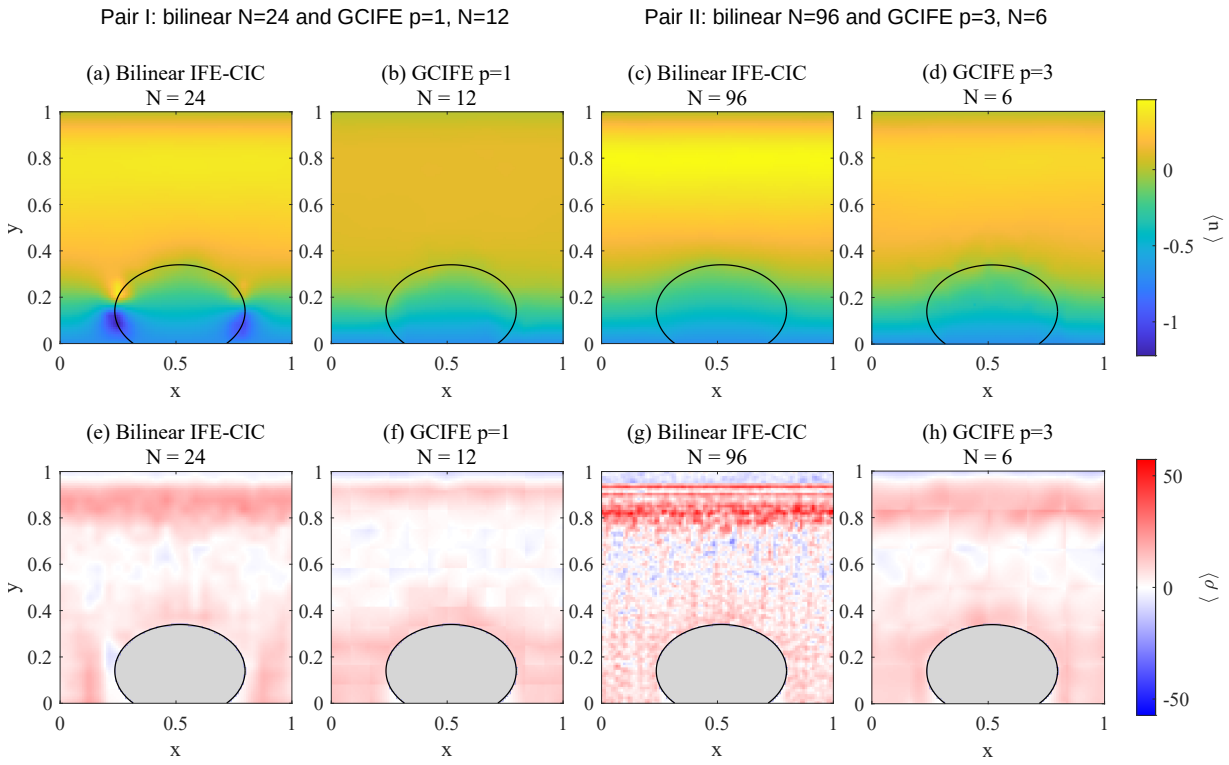


Fig. 7: Mean potential (a–d) and net volume charge density (e–h). Columns show bilinear $N = 24$, GCIFE $p = 1, N = 12$, bilinear $N = 96$, and GCIFE $p = 3, N = 6$, evaluated on a common 96×96 diagnostic grid. Each quantity shares one color scale. Black curves mark the interface; gray regions in (e–h) mark the dielectric.

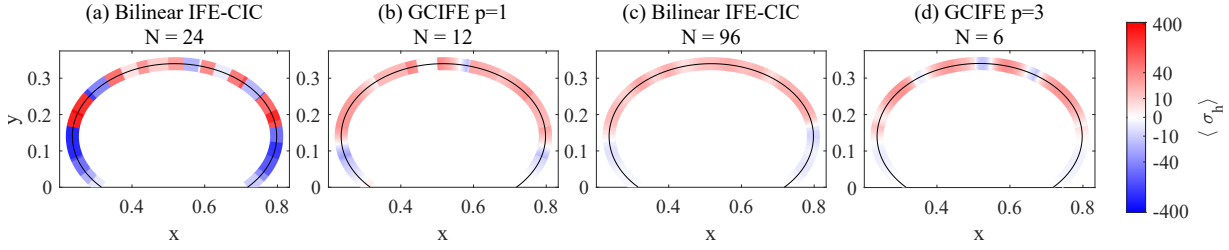


Fig. 8: Mean interface charge density, with columns ordered as in Figure 7. A common symmetric $\operatorname{asinh}(\sigma_h/10)$ color mapping reveals the weaker variations while retaining the full $N = 24$ extrema; colorbar ticks give actual charge-density values. The native segmentwise representation is extended into a normal display band without spatial smoothing.

should be read together with the resolution requirements. Using the prescribed inlet thermal scale and charge-mass ratio, the bulk reference electron Debye length estimated from the electron density is approximately 0.201, 0.193, 0.226, and 0.200 for bilinear $N = 24$, bilinear $N = 96$, GCIFE $p = 1, N = 12$, and GCIFE $p = 3, N = 6$, respectively; the corresponding mesh–Debye length ratios are 0.207, 0.054, 0.369, and 0.831. This indicates that bilinear IFE–PIC requires a mesh length much smaller than the Debye length to correctly resolve physical quantities near the interface. The resident particle populations are comparable (363 versus 370), yet the GCIFE charge-density field in Figure 7(h) is markedly less noisy than panel (g). Its coarse mesh alleviates low particle occupancy, while moment-based transfer gives a smoother charge representation. Reducing the bilinear $N = 96$ noise by increasing particle number would add deposition, gathering, and particle-advancement work, further increasing the cost of that strategy. Thus, the high-order coarse-mesh configuration combines lower measured accumulated runtime with a cleaner charge-density distribution. Relative to bilinear IFE–PIC, the GCIFE–PIC cost is concentrated more strongly in charge-density deposition and in field gathering within the gather-and-push stage; dedicated optimization of these stages is not pursued here and offers scope for further improvements in computational speed.

Table 1: Runtime and resident particle populations for the two paired comparisons. Component times average the final 128 PIC steps before the first steady-state pass. Accumulated runtime includes PIC steps only, excluding initialization, diagnostics, and output. Particle counts include ions and electrons remaining in the domain at that checkpoint.

	Pair I		Pair II	
Quantity	Bilinear IFE–CIC $N = 24$	GCIFE $p = 1$ $N = 12$	Bilinear IFE–CIC $N = 96$	GCIFE $p = 3$ $N = 6$
Δt	0.01	0.02	0.0025	0.16
Passed steady-state criterion	Yes	Yes	Yes	Yes
Field solve (ms/step)	1.757	0.797	72.464	2.608
Volume deposition (ms/step)	0.182	574.804	0.638	2269.734
Interface deposition (ms/step)	4.758	21.017	14.591	36.473
Gather and push (ms/step)	2.237	77.251	2.499	271.627
Total time per step (ms/step)	9.204	674.699	90.566	2581.594
Steps to steady state	55808	5120	24576	448
Resident particles at steady state	413	367	370	363
Accumulated step runtime (s)	531.34	3308.38	1464.64	941.24

6. Conclusion

In this paper, we developed a nonhomogeneous high-order GCIFE–PIC framework for electrostatic particle simulations with curved material interfaces on Cartesian meshes. We first proposes a nonhomogeneous GCIFE method, which can accurately calculate non-homogeneous curved boundary interface problems. Then we combine the nonhomogeneous GCIFE with the moment-based particle deposition and gathering, interface-aware charge transfer, high-order particle integration, and Boris-SDC advancement, constructing a consistent high-order GCIFE-PIC framework.

The numerical tests demonstrate high-order field convergence, roundoff-level charge conservation, and improved particle–field transfer and charge representation near curved interfaces. Even the lowest-order GCIFE–PIC con-

figuration outperforms the conventional bilinear IFE–PIC method based on line-segment interface approximation, demonstrating the benefit of accurate curved-interface representation on coarse Cartesian meshes. Furthermore, mesh refinement in the conventional PIC method mitigates the field-discretization error inherent to the low-order scheme; however, at a fixed particle count, it simultaneously aggravates the statistical fluctuations in particle sampling. The higher-order GCIFE–PIC method circumvents this trade-off by attaining a comparable field resolution on a coarse mesh, thereby avoiding the additional sampling noise. The coupled tests show a configuration-dependent computational cost: the coarse high-order GCIFE case reduces accumulated step runtime relative to its refined bilinear control.

Acknowledgments

This work is supported by Doctoral Start-up Foundation of Liaoning Province(2026-BS-0078, 2026-BS-0402).

References

[1] C. K. Birdsall, A. B. Langdon, Plasma Physics Via Computer Simulation, Plasma Physics Via Computer Simulation, 1991.
[2] A. Christlieb, R. Krasny, J. Verboncoeur, J. Emhoff, I. Boyd, Grid-free plasma simulation techniques, IEEE Transactions on Plasma Science 34 (2006) 149–165.
[3] A. D. Greenwood, K. L. Cartwright, J. W. Luginsland, E. A. Baca, On the elimination of numerical cerenkov radiation in pic simulations, Journal of Computational Physics 201 (2004) 665–684.
[4] M. J. Berger, J. Oliger, Adaptive mesh refinement for hyperbolic partial differential equations, Journal of Computational Physics 53 (1984) 484–512.
[5] J.-L. Vay, P. Colella, J. W. Kwan, P. McCorquodale, D. B. Serafini, A. Friedman, D. P. Grote, G. Westenskow, J.-C. Adam, A. Héron, I. Haber, Application of adaptive mesh refinement to particle-in-cell simulations of plasmas and beams, Physics of Plasmas 11 (2004) 2928–2934.
[6] J.-L. Vay, A. Almgren, J. Bell, L. Ge, D. Grote, M. Hogan, O. Kononenko, R. Lehe, A. Myers, C. Ng, et al., Warp-x: A new exascale computing platform for beam–plasma simulations, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment 909 (2018) 476–479.
[7] G. Memos, E. Lidorikis, G. Kokkoris, Roughness evolution and charging in plasma-based surface engineering of polymeric substrates: The effects of ion reflection and secondary electron emission, Micromachines 9 (2018) 415.
[8] D. Lund, X.-M. He, D. Han, Charging of irregularly-shaped dust grains near surfaces in space, in: AIAA SciTech 2023 Forum, AIAA 2023-2616, National Harbor, Maryland & Virtual Conference, 2023.
[9] G. B. Jacobs, J. S. Hesthaven, High-order nodal discontinuous galerkin particle-in-cell method on unstructured grids, Journal of Computational Physics 214 (2006) 96–121.
[10] D. A. S. Brown, M. T. Bettencourt, S. A. Wright, S. M. Maheswaran, J. P. Jones, S. A. Jarvis, Higher-order particle representation for particle-in-cell simulations, Journal of Computational Physics 435 (2021) 110255.
[11] T. Stindl, J. Neudorfer, A. Stock, M. Auweter-Kurtz, C.-D. Munz, S. Roller, R. Schneider, Comparison of coupling techniques in a high-order discontinuous galerkin-based particle-in-cell solver, Journal of Physics D: Applied Physics 44 (2011) 194004.
[12] Z. Li, T. Lin, X. Wu, New Cartesian grid methods for interface problems using the finite element formulation, Numerische Mathematik 96 (2003) 61–98.
[13] B. Camp, T. Lin, Y. Lin, W. Sun, Quadratic immersed finite element spaces and their approximation capabilities, Advances in Computational Mathematics 24 (2006) 81–112.
[14] X.-M. He, Bilinear immersed finite elements for interface problems, Ph.D. thesis, Virginia Polytechnic Institute and State University, 2009.
[15] X.-M. He, T. Lin, Y. Lin, Immersed finite element methods for elliptic interface problems with non-homogeneous jump conditions, International Journal of Numerical Analysis and Modeling 8 (2011) 284–301.
[16] T. Lin, D. Sheen, X. Zhang, A locking-free immersed finite element method for planar elasticity interface problems, Journal of Computational Physics 247 (2013) 228–247.
[17] R. Guo, T. Lin, X. Zhang, Nonconforming immersed finite element spaces for elliptic interface problems, Computers & Mathematics with Applications 75 (2018) 2002–2016.
[18] R. Guo, Design, analysis, and application of immersed finite element methods, Ph.D. thesis, Virginia Polytechnic Institute and State University, 2019.
[19] R. Kafafy, T. Lin, Y. Lin, J. Wang, Three-dimensional immersed finite element methods for electric field simulation in composite materials, International Journal for Numerical Methods in Engineering 64 (2005) 940–972.
[20] S. Adjerid, M. Ben-Romdhane, T. Lin, Higher degree immersed finite element methods for second-order elliptic interface problems, International Journal of Numerical Analysis and Modeling 11 (2014) 541–566.
[21] R. Guo, T. Lin, A higher degree immersed finite element method based on a Cauchy extension for elliptic interface problems, SIAM Journal on Numerical Analysis 57 (2019) 1545–1573.
[22] R. Guo, Solving parabolic moving interface problems with dynamical immersed spaces on unfitted meshes: fully discrete analysis, SIAM Journal on Numerical Analysis 59 (2021) 797–828.
[23] D. Kwak, S. Jin, D. Kyeong, A stabilized P_1-nonconforming immersed finite element method for the interface elasticity problems, ESAIM: Mathematical Modelling and Numerical Analysis 51 (2017) 187–207.

[24] C. Wang, P. Sun, Z. Li, An iterative approach for constructing immersed finite element spaces and applications to interface problems, International Journal of Numerical Analysis and Modeling 16 (2019) 167–191.
[25] J. Wang, X. Zhang, Q. Zhuang, An immersed Crouzeix-Raviart finite element method for Navier-Stokes equations with moving interfaces, International Journal of Numerical Analysis and Modeling 19 (2022) 563–586.
[26] C. Lu, Z. Yang, J. Bai, Y. Cao, X.-M. He, Three-dimensional immersed finite element method for anisotropic magnetostatic/electrostatic interface problems with non-homogeneous flux jump, International Journal for Numerical Methods in Engineering 121 (2020) 2107–2127.
[27] J. Bai, Y. Cao, X.-M. He, H. Liu, X. Yang, Modeling and an immersed finite element method for an interface wave equation, Computers & Mathematics with Applications 76 (2018) 1625–1638.
[28] R. Kafafy, Immersed finite element Particle-In-Cell simulations of ion propulsion, Ph.D. thesis, Virginia Polytechnic Institute and State University, 2005.
[29] R. Kafafy, J. Wang, T. Lin, A hybrid-grid immersed-finite-element particle-in-cell simulation model of ion optics plasma dynamics, Dynamics of Continuous, Discrete and Impulsive Systems, Series B: Applications and Algorithms 12 (2005) 1–16.
[30] Y. Han, G. Xia, C. Lu, X.-M. He, Trilinear immersed finite element method for 3D anisotropic interface problems with applications to plasma thrusters, AIAA Journal 61 (2023).
[31] C. Lu, J. Wan, Y. Cao, X.-M. He, A fully decoupled iterative method with three-dimensional anisotropic immersed finite elements for Kaufman-type discharge problems, Computer Methods in Applied Mechanics and Engineering 372 (2020) 113345.
[32] R. S. Roy, D. Hastings, N. Gastonis, Ion-thruster plume modeling for backflow contamination, Journal of Spacecraft and Rockets 33 (1996) 525–534.
[33] H. Jian, Y. Chu, H. Cao, Y. Cao, X.-M. He, G. Xia, Three-dimensional IFE-PIC numerical simulation of background pressure's effect on accelerator grid impingement current for ion optics, Vacuum 116 (2015) 130–138.
[34] R. Kafafy, J. Wang, A hybrid grid immersed finite element particle-in-cell algorithm for modeling spacecraft–plasma interactions, IEEE Transactions on Plasma Science 34 (2006) 2114–2124.
[35] D. Depew, D. Han, J. Wang, X.-M. He, T. Lin, Immersed-finite-element particle-in-cell simulations of lunar surface charging, in: Proceedings of the 13th Spacecraft Charging Technology Conference, Pasadena, California, 2014, p. 199.
[36] D. Han, J. Wang, X.-M. He, Immersed-finite-element particle-in-cell simulations of plasma charging at lunar terminator, Journal of Spacecraft and Rockets 55 (2018) 1490–1497.
[37] D. Han, P. Wang, X.-M. He, T. Lin, J. Wang, A 3D immersed finite element method with non-homogeneous interface flux jump for applications in particle-in-cell simulations of plasma-lunar surface interactions, Journal of Computational Physics 321 (2016) 965–980.
[38] D. Lund, X.-M. He, X. Zhang, D. Han, Weak scaling of the parallel immersed finite element particle-in-cell (PIFE-PIC) framework with lunar plasma charging simulations, Computational Particle Mechanics (2022).
[39] J. Wang, X.-M. He, Y. Cao, Modeling spacecraft charging and charged dust particle interactions on lunar surface, in: Proceedings of the 10th Spacecraft Charging Technology Conference, Biarritz, France, 2007.
[40] J. Wang, X.-M. He, Y. Cao, Modeling electrostatic levitation of dusts on lunar surface, IEEE Transactions on Plasma Science 36 (2008) 2459–2466.
[41] P. Wang, Immersed finite element particle-in-cell modeling of surface charging in rarefied plasmas, Ph.D. thesis, Virginia Polytechnic Institute and State University, 2010.
[42] D. Han, J. Wang, X.-M. He, A non-homogeneous immersed-finite-element particle-in-cell method for modeling dielectric surface charging in plasmas, IEEE Transactions on Plasma Science 44 (2016) 1326–1332.
[43] Y. Chu, D. Han, Y. Cao, X.-M. He, J. Wang, An immersed-finite-element particle-in-cell simulation tool for plasma surface interaction, International Journal of Numerical Analysis and Modeling 14 (2017) 175–200.
[44] D. Han, J. Wang, X.-M. He, PIFE-PIC: a 3-D parallel immersed finite element particle-in-cell framework for plasma simulations, in: Proceeding of 2018 AIAA Aerospace Sciences Meeting, AIAA-2018-2196, Kissimmee, Florida, 2018.
[45] D. Han, X.-M. He, D. Lund, X. Zhang, PIFE-PIC: parallel immersed-finite-element particle-in-cell for 3-D kinetic simulations of plasma-material interactions, SIAM Journal on Scientific Computing 43 (2021) C235–C257.
[46] H. Cao, Y. Cao, Y. Chu, X.-M. He, T. Lin, A Huygens immersed-finite-element particle-in-cell method for modeling plasma-surface interactions with moving interface, Communications in Nonlinear Science and Numerical Simulation 59 (2018) 132–148.
[47] J. Bai, Y. Cao, Y. Chu, X. Zhang, An improved immersed finite element particle-in-cell method for plasma simulation, Computers & Mathematics with Applications 75 (2018) 1887–1899.
[48] J. Bai, Y. Cao, X.-M. He, P. E, An implicit particle-in-cell model based on anisotropic immersed-finite-element method, Computer Physics Communications 261 (2021) 107655.
[49] Y. Cao, Y. Chu, X.-M. He, T. Lin, An iterative immersed finite element method for an electric potential interface problem based on given surface electric quantity, Journal of Computational Physics 281 (2015) 82–95.
[50] Y. Chu, Y. Cao, X.-M. He, M. Luo, Asymptotic boundary conditions with immersed finite elements for interface magnetostatic/electrostatic field problems with open boundary, Computer Physics Communications 182 (2011) 2331–2338.
[51] S. Adjerid, T. Lin, H. Meghaichi, A high order geometry conforming immersed finite element for elliptic interface problems, Computer Methods in Applied Mechanics and Engineering 420 (2024) 116703.
[52] M. Winkel, R. Speck, D. Ruprecht, A high-order boris integrator, Journal of Computational Physics 295 (2015) 456–474.
[53] S. Adjerid, T. Lin, H. Meghaichi, Construction of basis functions for the geometry conforming immersed finite element method (2025). Preprint.