Journal of Computational Physics 546 (2026) 114508

Contents lists available at ScienceDirect

Journal of Computational Physics

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

A positivity-preserving hybrid DDG method for Poisson–Nernst–Planck systems

Hailiang Liu [a], Zhongming Wang [b], Peimeng Yin [c,*]

[a] Department of Mathematics, Iowa State University, Ames, IA 50011, USA
[b] Department of Mathematics and Statistics, Florida International University, Miami, FL 33199, USA
[c] Department of Mathematical Sciences, The University of Texas at El Paso, El Paso, TX 79968, USA

ARTICLE INFO

2000 MSC:
35K40
65M60
65M12
82C31

Keywords:
Poisson-Nernst-Planck equation
Positivity preserving
Discontinuous galerkin methods

ABSTRACT

In earlier work [H. Liu and Z. Wang, J. Comput. Phys., 328(2017)], an arbitrary high-order conservative and energy-dissipative direct discontinuous Galerkin (DDG) scheme was developed. Although this scheme enforced solution positivity using cell averages as reference values, it lacked a theoretical guarantee for the positivity of those cell averages. In this study, we develop a novel arbitrary high-order DDG method with rigorously proven positivity-preserving properties. Specifically, the positivity of the cell averages is ensured through a modified numerical flux in combination with forward Euler time discretization. To achieve point-wise positivity of ion concentrations, we introduce a hybrid algorithm that integrates a positivity-preserving limiter. The proposed method is further extended to higher-dimensional problems with rectangular meshes. Numerical results confirm the scheme's high-order accuracy, guaranteed positivity preservation, and consistent discrete energy dissipation.

1. Introduction

This work continues our project, initiated in [1], aimed at developing a rigorous structure-preserving discontinuous Galerkin (DG) method for the Poisson–Nernst–Planck (PNP) equations. In this paper, we focus on the PNP system given by

$$\partial_t c_i = \nabla \cdot (\nabla c_i + q_i c_i \nabla \psi), \quad x \in \Omega,\ t > 0, \quad i = 1, \cdots, m, \tag{1a}$$

$$-\Delta \psi = \sum_{i=1}^{m} q_i c_i + \rho_0, \quad x \in \Omega,\ t > 0, \tag{1b}$$

$$c_i(0, x) = c_i^{\text{in}}(x), \quad x \in \Omega; \quad \frac{\partial c_i}{\partial \mathbf{n}} + q_i c_i \frac{\partial \psi}{\partial \mathbf{n}} = 0 \ \text{ on } \partial\Omega,\ t > 0, \tag{1c}$$

$$\psi = \psi_D \ \text{ on } \partial\Omega_D, \ \text{ and } \frac{\partial \psi}{\partial \mathbf{n}} = \sigma \ \text{ on } \partial\Omega_N, \quad t > 0, \tag{1d}$$

where $c_i = c_i(t, x)$ represents the concentration of the i^{th} ionic species with charge q_i, and is subject to the given initial and boundary conditions. The electrostatic potential $\psi = \psi(t, x)$ is governed by the Poisson equation, with Neumann boundary conditions prescribed on $\partial\Omega_N$ and Dirichlet boundary conditions on $\partial\Omega_D$. Here, $\Omega \subset \mathbb{R}^d$ denotes a connected closed domain with a smooth boundary $\partial\Omega$, $\mathbf{n}$ is the unit outward normal vector on $\partial\Omega$.

* Corresponding author.
E-mail addresses: hliu@iastate.edu (H. Liu), zwang6@fiu.edu (Z. Wang), pyin@utep.edu (P. Yin).

https://doi.org/10.1016/j.jcp.2025.114508
Received 29 May 2025; Received in revised form 27 October 2025; Accepted 2 November 2025
Available online 7 November 2025

The PNP system (1) is widely used across many disciplines [2–10]. Among the key qualitative properties of its solution – such as mass conservation, energy dissipation, and positivity of concentrations– the preservation of positivity under high-order numerical approximations remains particularly challenging.

In this article, we advance our previous work by focusing on the challenge of positivity preservation within the same high-order DG approximation framework introduced in [1]. A general discussion of this issue and background references are given in the introduction to [1]. Recall that the direct DG method for (1) proposed in [1] is based on the following reformulation:

$$\partial_t c_i = \nabla \cdot (c_i \nabla p_i), \quad p_i = q_i \psi + \log c_i, \quad i = 1, \cdots, m,$$

where the electrostatic potential ψ is determined from the coupled Poisson equation. This reformulation, when discretized using DG methods, was shown to naturally preserve both mass conservation and energy dissipation. A critical ingredient of the method is the use of the direct DG (DDG) numerical fluxes on p_i, where the DDG fluxes are originally introduced in [11,12]), of the form for $w = p_i$:

$$\widehat{\partial_n w} := \beta_0 \frac{[w]}{h} + \{\partial_n w\} + \beta_1 h [\partial_n^2 w]. \tag{2}$$

Here n denotes the normal unit vector across cell interfaces, and $[q] = q^+ - q^-$, and $\{q\} = (q^+ + q^-)/2$, where q^+ and q^- representing values of q on either side of the interface. A broad range of parameter choices for β_0 and β_1 allows for the scheme to retain high-order accuracy while ensuring energy dissipation. However, this flexibility does not generally guarantee the preservation of solution positivity, which is not unexpected–achieving point-wise positivity in high-order schemes is well known to be unrealistic.

A commonly accepted strategy, following Shu and Zhang [13], is to ensure the positivity of cell averages and then apply a positivity-preserving limiter to enforce solution positivity. Unfortunately, the numerical method proposed in [1] may produce negative cell averages after a finite number of time steps, thereby causing the positivity-preserving limiter to fail.

In this work, we aim to overcome this critical limitation identified in [1], by developing a scheme that theoretically guarantees positivity of cell averages, even in general settings. We have two main objectives in this work:

i) To present a novel modified DDG numerical flux, and

ii) To analyze the propagation of positive cell averages.

The goal of (i) is to restore the positivity propagation of cell averages through a carefully designed flux modification. Our modified flux is based on a local correction constructed similarly to those given for one-dimensional nonlinear Fokker-Planck equations [14]. However, in our case, the correction is applied to the logarithmic flux formulation $p_i = q_i \psi + \log c_i$. Specifically, we define the modified normal derivative as:

$$\widetilde{\partial_n p_i} = \widehat{\partial_n p_i} + \frac{\widetilde{\beta}}{2}[c_i]. \tag{3}$$

Here $\tilde{\beta}$ is dimension-dependent and chosen to appropriately scale the correction term. The additional term $\widetilde{\beta}[c_i]/2$ in (3) is crucial in establishing the positivity of our DDG scheme. However, its inclusion transforms an originally linear scheme into a nonlinear one, introducing extra challenges for the error analysis. Despite this added complexity, the modification yields important practical benefits: the nonlinear formulation ensures positivity preservation and yields more physically consistent results in regimes where linear methods may produce nonphysical negative values. We view this as a meaningful trade-off between analytical simplicity and physical reliability.

For (ii), we analyze the evolution of cell averages under Euler time discretization. In the one dimensional case, the update for the cell average $\bar{c}_{ij}$ can be expressed as

$$\bar{c}_{ij}^{n+1} = \bar{c}_{ij}^{n} + \frac{\Delta t}{\Delta x}\{c_{ih}^n\}\left(\widehat{\partial_x p_{ih}^n} + \frac{\widetilde{\beta}_i}{2}[c_{ih}^n]\right)\Big|_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}},$$

where the flux terms are evaluated at cell interfaces. When using the modified flux (3) and applying the Gauss-Labotto decomposition of $\bar{c}_{ij}^n$, we show that all the interface trace terms stay non-negative, provided the time step is sufficiently small. This leads to a key result: the propagation of positive cell averages – that is, $\bar{c}_{ij}^{n+1} > 0$ if the current DG polynomials $c_{ih}^n > 0$ and the time step satisfies a suitable constraint.

As a result, we develop a hybrid DDG numerical algorithm that incorporates a positivity-preserving limiter, similar to the one proposed in [15], applied at each time step. This limiter enforces the positivity of the numerical solution without degrading the scheme's high-order accuracy. Our analysis and numerical results demonstrate that the resulting positivity-preserving DDG scheme maintains arbitrary high-order accuracy in both one-dimensional and multi-dimensional settings.

Related work.

The development of high-order, positivity-preserving numerical methods for the PNP system remains relatively limited. A notable contribution is found in [16], where the authors proposed a direct DG method that rigorously preserves the positivity of cell averages. Their method relies on the use of the non-logarithmic reformulation and carefully constructed numerical fluxes to enforce positivity; however, the use of global fluxes limits the achievable order of accuracy, making extensions to higher-order DG methods challenging. In contrast, the local flux strategy developed in this work, as well as in [14], overcomes this limitation and achieves spatial accuracy higher than third order. Note that the modified numerical fluxes render the scheme nonlinear – this is a deliberate design choice motivated by the need to enforce positivity of the solution. We acknowledge that this idea is not new; similar nonlinear modifications

to the advective fluxes have been proposed in earlier work on positivity-preserving high-order DG schemes [17–19]. Our approach follows the same general philosophy, modifying the diffusive fluxes within the DDG framework for the PNP system, which presents different challenges.

After our earlier work [1], which introduced a novel framework for achieving arbitrary high-order spatial accuracy while preserving key structural properties of the solution, there has been a growing body of research focused on designing numerical schemes that maintain essential solution features at the discrete level. These efforts typically involve reformulating the original PDE and adopting implicit-explicit time discretizations to balance stability and computational efficiency. Most of such schemes use central differencing for spatial discretization, typically resulting in second-order spatial accuracy.

Common reformulation strategies include:

(1) Non-logarithmic transformations, referred to as the Slotboom transformation in semiconductor modeling – convert the drift-diffusion operator into a self-adjoint elliptic operator. This is particularly effective for ensuring positivity; see e.g., [16,20–26].

(2) Gradient flow structures of the form $\rho_t = \nabla \cdot (\rho \nabla \delta_\rho E)$, where E denotes an energy functional associated with the physical system, are often discretized using explicit-implicit schemes; examples include the energetic variational approach (EnVarA) [27,28] and an independent development in [29]. In [30], the authors propose a fully-implicit time discretization combined with a log-density formulation to ensure positivity.

(3) Localized gradient flow formulations, such as $\rho_t = \nabla \cdot (\rho \nabla p)$ with $p = \delta_\rho E$ (see [1]) or $\rho_t = \nabla \cdot (\rho \xi)$ with $\xi = \nabla \delta_\rho E$ (see [18,31]), provide enhanced control over the solution structure over computational cells.

(4) Fully implicit gradient flows, often framed as JKO-type schemes [32,33], pose significant computational challenges. These are effectively addressed by recasting them as dynamic optimization problems [34], following the Benamou-Brenier formulation [35]. Notably, such dynamic optimal transport formulations can also be naturally derived using the Onsager variational principle [36]. These perspectives provide promising avenues for developing robust, structure-preserving numerical schemes.

Recent years have seen the development of several alternative positivity-preserving strategies beyond traditional flux limiters and reconstruction-based schemes. Notably, optimization-based post-processing techniques have been proposed to enforce discrete maximum principles and maintain non-negativity while preserving mass conservation and high-order accuracy [37]. In addition, the geometric quasilinearization method [38] offers a framework for transforming nonlinear PDEs into positivity-preserving forms through geometric reformulation. These approaches have demonstrated effectiveness in related diffusion and transport systems and may be adaptable to the PNP equations. Incorporating such ideas could further broaden the applicability of positivity-preserving algorithms for coupled electrostatic models.

We conclude this section by outlining the rest of the paper: in Section 2, we review the DDG method introduced in [1] and summarize its key solution properties, including mass conservation and the energy dissipation law. Section 3 focuses on the positivity-preserving property of the fully discretized scheme, incorporating the proposed novel local flux correction and the hybrid positivity-preserving algorithm in one dimension. In Section 4, we extend the method to higher dimensions and prove the positivity propagation results. Section 5 presents numerical examples that illustrate the effectiveness of the proposed approach. Finally, Section 6 offers concluding remarks.

2. Review of direct DG scheme in [1]

In this section, we briefly revisit the DG scheme for the PNP system (1) in 1D as presented in [1]. Consider a domain $\Omega = [a, b]$ with a possibly non-uniform mesh $\{I_j\}_{j=1}^N$, where the cell $I_j = (x_{j-1/2}, x_{j+1/2})$ with cell center $x_j = (x_{j-1/2} + x_{j+1/2})/2$, and

$$a = x_{1/2} < x_1 < \cdots < x_{N-1/2} < x_N < x_{N+1/2} = b.$$

We define the discontinuous finite element space as

$$V_h = \left\{ v \in L^2(\Omega), \quad v|_{I_j} \in P^k(I_j), j = 1, \cdots, N \right\},$$

where P^k denotes polynomials of degree at most k, and h the characteristic length of the mesh. At cell interfaces $x = x_{j+1/2}$, we introduce the notation

$$v^{\pm} = \lim_{\epsilon \to 0} v(x \pm \epsilon), \quad \{v\} = \frac{v^- + v^+}{2}, \quad [v] = v^+ - v^-.$$

The DDG scheme introduced in [1] is essentially based on the following localized gradient flow reformulation,

$$\partial_t c_i = \partial_x (c_i \partial_x p_i), \ i = 1, \cdots, m, \tag{4a}$$

$$p_i = q_i \psi + \log c_i, \tag{4b}$$

$$-\partial_x^2 \psi = \sum_{i=1}^m q_i c_i + \rho_0(x). \tag{4c}$$

More specifically, such a scheme when coupled with the first-order Euler time discretization with uniform step size Δt admits the following form: We seek $c_{ih}^{n+1}, \psi_h^{n+1} \in V_h$ such that for any $v_i, r_i, \eta \in V_h$,

$$\int_{I_j} \frac{c_{ih}^{n+1} - c_{ih}^n}{\Delta t} v_i dx = - \int_{I_j} c_{ih}^n \partial_x p_{ih}^n \partial_x v_i dx + \{c_{ih}^n\} \left(\widehat{\partial_x p_{ih}^n} v_i + (p_{ih}^n - \{p_{ih}^n\}) \partial_x v_i \right) \Big|_{\partial I_j}, \tag{5a}$$

$$\int_{I_j} p_{ih}^n r_i dx = \int_{I_j} (q_i \psi_h^n + \log c_{ih}^n) r_i dx, \tag{5b}$$

$$\int_{I_j} \partial_x \psi_h^n \partial_x \eta dx - \left(\widehat{\partial_x \psi_h^n} \eta + (\psi_h^n - \{\psi_h^n\}) \partial_x \eta \right)\Big|_{\partial I_j} = \int_{I_j} \left[\sum_{i=1}^m q_i c_{ih}^n + \rho_0 \right] \eta dx, \tag{5c}$$

where the numerical fluxes are $\widehat{\partial_x p_{ih}} = Fl(p_{ih})$ and $\widehat{\partial_x \psi_h} = Fl(\psi_h)$. The flux operator Fl at the cell interfaces $x_{j+1/2}$ for $0 < j < N$ is defined as

$$Fl(w) := \beta_0 \frac{[w]}{h_{j+\frac{1}{2}}} + \{\partial_x w\} + \beta_1 h_{j+\frac{1}{2}} [\partial_x^2 w], \tag{6}$$

where

$$h_{j+\frac{1}{2}} = x_{j+1} - x_j.$$

The boundary conditions are a critical component of the PNP model and determine important qualitative behavior of the solution. It is important to incorporate the boundary conditions properly. For the zero-flux condition $\partial_x c_i + q_i c_i \partial_x \psi = 0$ at both $x = a$ and $x = b$, we set at both $x_{1/2}$ and $x_{N+1/2}$,

$$Fl(p_{ih}) = 0, \{p_{ih}\} = p_{ih}, \{c_{ih}\} = c_{ih}. \tag{7}$$

For ψ with a mixed boundary condition of the form $\psi(t, a) = \psi_l, \partial_x \psi(t, b) = \sigma$, we set

$$Fl(\psi_h) = \beta_0 (\psi_h^+ - \psi_l)/h_{1/2} + \partial_x \psi_h^+, \{\psi_h\} = \psi_l, \quad x_{1/2} = a, \tag{8}$$

$$Fl(\psi_h) = \sigma, \{\psi_h\} = \psi_h^-, \quad x_{N+1/2} = b, \tag{9}$$

where

$$h_{1/2} = x_1 - x_{1/2}.$$

Such a choice is motivated by the results in [39]; otherwise, one would have to take $2\beta_0$ to replace the usual coefficient β_0. For other types of boundary conditions, numerical fluxes can be defined accordingly.

As usual, we initialize c_{ih}^0 by projecting c_i^{in} onto V_h so that

$$\int_{I_j} c_{ih}^0(x) v(x) dx = \int_{I_j} c_i^{\text{in}}(x) v(x) dx, \quad \forall v \in P^k(I_j), \quad i = 1, \cdots, m. \tag{10}$$

The following statement holds for the DG scheme (5) when subjected to a Neumann boundary condition.

Theorem 1. *cf. [1, Theorem 3.3]*

1. *The fully discrete scheme (5) conserves each total concentration $c_{ih}^n(x) (i = 1, \cdots, m)$ over time:*

$$\sum_{j=1}^N \int_{I_j} c_{ih}^n dx = \sum_{j=1}^N \int_{I_j} c_{ih}^{n+1} dx, \quad i = 1, \cdots, m, \qquad t > 0. \tag{11}$$

2. *Assuming $c_{ih}^n(x) > 0$ in each I_j, there exists $\mu^* > 0$ such that if the mesh ratio $\mu = \frac{\Delta t}{\Delta x^2} \in (0, \mu^*)$, then the fully discrete free energy*

$$F^n = \sum_{j=1}^N \int_{I_j} \left[\sum_{i=1}^m c_{ih}^n \log c_{ih}^n + \frac{1}{2} \left(\sum_{i=1}^m q_i c_{ih}^n + \rho_0 \right) \psi_h^n \right] dx + \frac{1}{2} \int_{\partial\Omega} \sigma \psi_h^n ds \tag{12}$$

satisfies

$$D_t F^n \le -\frac{1}{2} \sum_{i=1}^m A_{c_{ih}^n}(p_{ih}^n, p_{ih}^n), \tag{13}$$

where the bilinear form

$$A_{c_{ih}^n}(p_{ih}^n, v_i) = \sum_{j=1}^N \int_{I_j} c_{ih}^n \partial_x p_{ih}^n \partial_x v_i dx + \sum_{j=0}^N \{c_{ih}^n\} \left(\widehat{\partial_x p_{ih}^n} [v_i] + [p_{ih}^n] \{\partial_x v_i\} \right)_{j+\frac{1}{2}}.$$

Moreover,

$$F^{n+1} \le F^n, \tag{14}$$

provided that β_0 is suitably large, and $\beta_1 = 0$ in $Fl(\psi_h)$ defined in (6).

A significant limitation in the above results is the assumption $c_{ih}^n > 0$. It is well known that maintaining point-wise positivity directly through high order numerical approximations is unrealistic. A widely accepted approach, following Shu and Zhang [13], is to ensure positivity of cell averages, and restore solution positivity by further applying a positivity-preserving limiter. However, the numerical scheme (5) presented in [1] can produce negative cell averages in a finite number of steps, ultimately causing the positivity-preserving limiter to fail. This is confirmed in Sections 5.2 and 5.3 of this paper for both 1D and 2D cases. Our main objective here is to introduce a remedy to ensure the positivity of numerical averages.

3. A positivity-preserving flux

The key ingredient is a locally modified flux $\widetilde{\partial_x p_{ih}}$ which is used to replace the original flux $\widehat{\partial_x p_{ih}^n}$ in (5a) at each edge ∂I_j. The resulting DDG scheme becomes

$$\int_{I_j} \frac{c_{ih}^{n+1}-c_{ih}^n}{\Delta t} v_i dx = -\int_{I_j} c_{ih}^n \partial_x p_{ih}^n \partial_x v_i dx + \{c_{ih}^n\}\left(\widetilde{\partial_x p_{ih}^n} v_i + (p_{ih}^n - \{p_{ih}^n\})\partial_x v_i\right)\Big|_{\partial I_j}, \tag{15a}$$

$$\int_{I_j} p_{ih}^n r_i dx = \int_{I_j} (q_i \psi_h^n + \log c_{ih}^n) r_i dx, \tag{15b}$$

$$\int_{I_j} \partial_x \psi_h^n \partial_x \eta dx - \left(\widehat{\partial_x \psi_h^n}\eta + (\psi_h^n - \{\psi_h^n\})\partial_x \eta\right)\Big|_{\partial I_j} = \int_{I_j}\left[\sum_{i=1}^m q_i c_{ih}^n + \rho_0\right]\eta dx, \tag{15c}$$

where the newly introduced flux at the cell interface $x_{j+\frac{1}{2}}$ for $0 < j < N$ is given by

$$\widetilde{\partial_x p_{ih}} = \widehat{\partial_x p_{ih}} + \frac{\widetilde{\beta}_i}{2}[c_{ih}], \tag{16}$$

and

$$\widetilde{\beta}_i = \begin{cases} \frac{|\widehat{\partial_x p_{ih}^n}|}{\{c_{ih}^n\}}\Big|_{x_{j+\frac{1}{2}}}, & \text{if } \{c_{ih}^n\} > 0; \\ 0, & \text{if } \{c_{ih}^n\} = 0. \end{cases} \tag{17}$$

For $j = 0$ or $j = N$, no modification is imposed so that

$$\widetilde{\partial_x p_{ih}} = \widehat{\partial_x p_{ih}} = 0, \quad \tilde{\beta}_i = 0.$$

3.1. Positive cell averages

We define the cell average on cell I_j as

$$\bar{c}_{ij}^n := \frac{1}{\Delta x_j}\int_{I_j} c_{ih}^n dx, \quad \Delta x_j = x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}},$$

and establish the following result:

Theorem 2. *For the fully discrete scheme (15), the cell average $\bar{c}_{ij}^{n+1} > 0$ provided $c_{ih}^n(x) > 0$ and*

$$\Delta t \le \omega_1 \min_{1\le j\le N}\left\{\Delta x_j\, |\widehat{\partial_x p_{ih}^n}|^{-1}\Big|_{x_{j\pm\frac{1}{2}}}\right\} \tag{18}$$

is satisfied. Here, ω_1 represents the first weight of the Gauss-Lobatto quadrature rules with $M \ge \frac{k+3}{2}$ points. For notational convenience, we include the terms for $j = 0, N$ with $|\widehat{\partial_x p_{ih}^n}|^{-1}\Big|_{x_{j+1/2}} = \infty$.

Proof. We define

$$\lambda_j := \frac{\Delta t}{(\Delta x_j)^2}.$$

Taking $v_i = \frac{\Delta t}{\Delta x_j}$ in (15a), the time evolution of $\bar{c}_{ij}^{n+1}$ on I_j is given by

$$\begin{aligned}
\bar{c}_{ij}^{n+1} =& \bar{c}_{ij}^n + \lambda_j\{c_{ih}^n\}\Delta x_j\left(\widehat{\partial_x p_{ih}^n} + \frac{\widetilde{\beta}_i}{2}[c_{ih}^n]\right)\Big|_{x_{j+\frac{1}{2}}} - \lambda_j \Delta x_j\{c_{ih}^n\}\left(\widehat{\partial_x p_{ih}^n} + \frac{\widetilde{\beta}_i}{2}[c_{ih}^n]\right)\Big|_{x_{j-\frac{1}{2}}} \\
=& \sum_{m=1}^M \omega_m c_{ih}^n(x_m^*) + \frac{\lambda_j}{2}\Delta x_j\left(\widehat{\partial_x p_{ih}^n} + \widetilde{\beta}_i\{c_{ih}^n\}\right)\Big|_{x_{j+\frac{1}{2}}} c_{ih}^n(x_{j+\frac{1}{2}}^+) \\
&+ \frac{\lambda_j}{2}\Delta x_j\left(\widehat{\partial_x p_{ih}^n} - \widetilde{\beta}_i\{c_{ih}^n\}\right)\Big|_{x_{j+\frac{1}{2}}} c_{ih}^n(x_{j+\frac{1}{2}}^-) \\
&- \frac{\lambda_j}{2}\Delta x_j\left(\widehat{\partial_x p_{ih}^n} + \widetilde{\beta}_i\{c_{ih}^n\}\right)\Big|_{x_{j-\frac{1}{2}}} c_{ih}^n(x_{j-\frac{1}{2}}^+) \\
&+ \frac{\lambda_j}{2}\Delta x_j\left(-\widehat{\partial_x p_{ih}^n} + \widetilde{\beta}_i\{c_{ih}^n\}\right)\Big|_{x_{j-\frac{1}{2}}} c_{ih}^n(x_{j-\frac{1}{2}}^-),
\end{aligned}$$

where the M pairs (ω_m, x_m^*) are given by the Gauss-Lobatto quadrature rules with $M \geq \frac{k+3}{2}$ that are exact for k-th polynomials and

$$x_1^* = x_{j-\frac{1}{2}}, \quad x_M^* = x_{j+\frac{1}{2}}.$$

Thanks to the choice of $\widetilde{\beta}_i$ in (17), we have

$$\left(\widehat{\partial_x p_{ih}^n} + \widetilde{\beta}_i\{c_{ih}^n\}\right)\Big|_{x_{j+\frac{1}{2}}} \geq 0, \quad \left(-\widehat{\partial_x p_{ih}^n} + \widetilde{\beta}_i\{c_{ih}^n\}\right)\Big|_{x_{j-\frac{1}{2}}} \geq 0,$$

for $1 \leq j \leq N$.

It follows that

$$\begin{aligned}\bar{c}_{ij}^{n+1} = \sum_{m=2}^{M-1} \omega_m c_{ih}^n(x_m^*) &+ \frac{\lambda_j}{2}\Delta x_j\left(\widehat{\partial_x p_{ih}^n} + \widetilde{\beta}_i\{c_{ih}^n\}\right)c_{ih}^n(x_{j+\frac{1}{2}}^+) \\ &+ \frac{\lambda_j}{2}\Delta x_j\left(-\widehat{\partial_x p_{ih}^n} + \widetilde{\beta}_i\{c_{ih}^n\}\right)c_{ih}^n(x_{j-\frac{1}{2}}^-) \\ &+ \left[\omega_1 - \frac{\lambda_j}{2}\Delta x_j(\widehat{\partial_x p_{ih}^n} + \widetilde{\beta}_i\{c_{ih}^n\})\right]c_{ih}^n(x_{j-\frac{1}{2}}^+) \\ &+ \left[\omega_M - \frac{\lambda_j}{2}\Delta x_j(-\widehat{\partial_x p_{ih}^n} + \widetilde{\beta}_i\{c_{ih}^n\})\right]c_{ih}^n(x_{j+\frac{1}{2}}^-) > 0,\end{aligned} \tag{19}$$

provided

$$\lambda_j \leq \min\left\{\frac{\omega_1}{\Delta x_j}|\widehat{\partial_x p_{ih}^n}|^{-1}\Big|_{x_{j-\frac{1}{2}}}, \frac{\omega_1}{\Delta x_j}|\widehat{\partial_x p_{ih}^n}|^{-1}\Big|_{x_{j+\frac{1}{2}}}\right\}$$

for $2 \leq j \leq N-1$, and

$$\lambda_1 \leq \frac{\omega_1}{\Delta x_1}|\widehat{\partial_x p_{ih}^n}|^{-1}\Big|_{x_{\frac{3}{2}}} = \min\left\{\frac{\omega_1}{\Delta x_1}|\widehat{\partial_x p_{ih}^n}|^{-1}\Big|_{x_{\frac{1}{2}}}, \frac{\omega_1}{\Delta x_1}|\widehat{\partial_x p_{ih}^n}|^{-1}\Big|_{x_{\frac{3}{2}}}\right\},$$

$$\lambda_N \leq \frac{\omega_1}{\Delta x_N}|\widehat{\partial_x p_{ih}^n}|^{-1}\Big|_{x_{N-\frac{1}{2}}} = \min\left\{\frac{\omega_1}{\Delta x_N}|\widehat{\partial_x p_{ih}^n}|^{-1}\Big|_{x_{N-\frac{1}{2}}}, \frac{\omega_1}{\Delta x_N}|\widehat{\partial_x p_{ih}^n}|^{-1}\Big|_{x_{N+\frac{1}{2}}}\right\},$$

where the equalities follow from the fact that

$$|\widehat{\partial_x p_{ih}^n}|^{-1}\Big|_{x_{1/2}} = |\widehat{\partial_x p_{ih}^n}|^{-1}\Big|_{x_{N+1/2}} = \infty.$$

Note that $\omega_1 = \omega_M$ due to the symmetry of the Gauss-Lobatto quadrature rules. □

3.2. Positivity-preserving reconstruction

The equation (15b) involves the term $\log c_{ih}^n$, which requires the concentrations c_{ih}^n to remain positive at each time step. However, Theorem 2 only guarantees positive cell averages each step, provided the previous profile is positive. To ensure the propagation of positivity over time, we apply an accuracy-preserving positive limiter based on positive cell averages.

Let $w_h \in P^k(I_j)$ approximate a smooth function $w(x) \geq 0$, with cell averages $\bar{w}_j > \delta$, where δ is a small positive parameter or zero. We define the modified polynomial $w_h^\delta(x) \in P^k(I_j)$ as follows:

$$w_h^\delta(x) = \begin{cases}\bar{w}_j + \frac{\bar{w}_j - \delta}{\bar{w}_j - \min_{I_j} w_h(x)}(w_h(x) - \bar{w}_j), & \text{if } \min_{I_j} w_h(x) < \delta, \\ w_h(x), & \text{otherwise.}\end{cases} \tag{20}$$

This reconstruction preserves the original cell averages and ensures that

$$\min_{I_j} w_h^\delta(x) \geq \delta. \tag{21}$$

More importantly, the following estimate demonstrates how the choice of δ affects the reconstruction accuracy.

Lemma 1. *cf. [1, Lemma 4.2] If $\bar{w}_j > \delta$, the reconstructed polynomial w_h^δ satisfies the following estimate*

$$|w_h^\delta(x) - w_h(x)| \leq C(k)\big(\|w_h(x) - w(x)\|_\infty + \delta\big), \quad \forall x \in I_j,$$

where $C(k)$ is a constant depending on k.

This result guarantees that $w_h^\delta(x)$ maintains accuracy when $\delta \leq h^{k+1}$. Such reconstruction limiting techniques are inspired by the limiter introduced in [13] for hyperbolic conservation laws.

Remark 1. Let S_j denote the set of Gauss–Lobatto quadrature points on I_j. In the fully discrete DG scheme (15), the value of $w_h(x)$ is only needed at the points in S_j. Therefore, in practice, we replace $\min_{x\in I_j} w_h(x)$ in (20) with $\min_{x\in S_j} w_h(x)$, following the work of Zhang and Shu [13]. The estimate in Lemma 1 still holds under this modification. A similar strategy can be applied to the higher-dimensional case discussed in the next section. This approach avoids the computational cost of finding the true minimum of a high-degree polynomial while providing sufficient accuracy for enforcing positivity in practice.

3.3. The hybrid positivity-preserving algorithm

Remark 2. We provide some remarks concerning the numerical implementation.

1. We assume the mesh is regular, satisfying $h_{\max} \le C h_{\min}$ for some constant C, where $h_{\max}$ and $h_{\min}$ denote the maximum and minimum element diameters, respectively. Let h be the characteristic length of the mesh size, then we have $\widehat{\partial_x p_{ih}^n} \approx \mathcal{O}(h^{-1})$. From the CFL condition (18), we observe that
$$\Delta t \le \mathcal{O}(h^2), \tag{22}$$
which is consistent with the expected condition on the explicit time step for diffusion problems.
2. Due to the presence of the logarithmic term in the DG scheme (15b), we additionally require that the cell average satisfy
$$\bar{c}_{ij}^{n+1} \ge \varepsilon,$$
for some tolerance $\varepsilon \in (0, C_\eta h^{k+1})$, where C_η is a positive constant. Let
$$\lambda = \frac{\omega_1}{h} \min_{0\le j\le N} \left\{ |\widehat{\partial_x p_{ih}^n}|^{-1}\Big|_{x_{j+\frac{1}{2}}} \right\}, \quad \text{with } h = \min_j\{\Delta x_j\}.$$
It then suffices to replace condition (18) with the following time-step restriction:
$$\Delta t \le \lambda h^2 = \omega_1\, h \min_{1\le j\le N} \left\{ |\widehat{\partial_x p_{ih}^n}|^{-1}\Big|_{x_{j\pm\frac{1}{2}}} \right\} \le \omega_1 \min_{1\le j\le N} \left\{ \Delta x_j\, |\widehat{\partial_x p_{ih}^n}|^{-1}\Big|_{x_{j\pm\frac{1}{2}}} \right\}. \tag{23}$$
In our numerical tests, the time step is chosen to satisfy
$$\Delta t = \gamma \lambda h^2 \le \lambda h^2, \tag{24}$$
where $\gamma \in (0, 1]$ is a empirical parameter. The inclusion of parameter γ in the empirical condition (18) is motivated by (19) in the proof of Theorem 2, and it serves to provide a practical safety margin in computations where solutions may challenge the limiter near the positivity threshold. The effect of this choice can be observed in Section 5.3 (Test Case 3).
3. The time discretization in the scheme (15) can be implemented using high order strong stability preserving Runge-Kutta schemes [40], as they are positive linear combinations of the forward Euler method used in (15), hence the positive-preserving property remains valid when taking suitably small time steps.
4. If $c_i(0, x) = 0$ only on a set of measure zero, the cell averages typically stay positive, and the limiter ensures that (21) is satisfied. However, when $c_i(0, x) = 0$ over a non-empty interior region, some cell averages may vanish. To address this, we first regularize the initial condition by setting $c_i(0, x) = \max\{c_i(0, x), \delta\}$ for some constant $\delta > 0$ of order h^{k+1}. The regularized initial data are then projected onto V_h using equation (10) to obtain c_{ih}^0.

The corresponding algorithm is summarized in Algorithm 1.

Algorithm 1: The hybrid positivity-preserving algorithm for the PNP system.

Input: Initial data $c_i^{\text{in}}(x)$, boundary condition for ψ, initial time step Δt_0
Output: Updated concentration c_{ih}^n

Initialization: Project $c_i^{\text{in}}(x)$ onto V_h, as defined in (10), to obtain $c_{ih}^0(x)$.
Set $t = 0$, and $\Delta t = \Delta t_0$.
while $t < T$ **do**
  Positivity reconstruction: If necessary, reconstruct c_{ih}^n according to (20) to ensure positivity: $c_{ih}^n > \delta$.
  Solve for ψ_h^n: Using c_{ih}^n, solve (15c) to compute ψ_h^n.
  Compute p_{ih}^n: Using c_{ih}^n and ψ_h^n, evaluate p_{ih}^n from (15b).
  Update c_{ih}^{n+1}: Solve (5a) or (15a) to obtain c_{ih}^{n+1}.
    if $\bar{c}_{ih}^{n+1} < \delta$ **then**
      Update Δt using equation (18) or (24).
      Solve (15a) to recompute c_{ih}^{n+1}.
    Advance time: $t = t + \Delta t$.

Remark 3. Although not formally proven, our extensive 1D and 2D numerical experiments in Section 5 show that the numerical solution produced from Algorithm 1 exhibits the property of discrete mass conservation and free energy dissipation.

4. High-dimensional positivity-preserving DG schemes

In this section, we extend our result to DG schemes of $(k + 1)$-th order accuracy on rectangular meshes solving multidimensional (multi-D) PNP equations.

4.1. Scheme formulation

Assume that (1) is posed on $x = (x^1, \dots, x^d) \in \Omega = \Pi_{l=1}^{d}[L_{x^l}, R_{x^l}] \subset \mathbb{R}^d$, where L_{x^l}, R_{x^l} are endpoints in x^l direction satisfying $L_{x^l} < R_{x^l}$. Consider a rectangular partition $\mathcal{T}_h$ of

$$\Omega = \bigcup_{\alpha}^{N} K_\alpha, \tag{25}$$

where $\alpha = (\alpha_1, \dots, \alpha_d)$, $N = (N_1, \dots, N_d)$. Here, $K_\alpha = I^1_{\alpha_1} \times \dots \times I^d_{\alpha_d}$ with $I^l_{\alpha_l} = [x^l_{\alpha_l - 1/2}, x^l_{\alpha_l + 1/2}]$ and its center $x^l_{\alpha_l} = (x^l_{\alpha_l - 1/2} + x^l_{\alpha_l + 1/2})/2$, for $\alpha_l = 1, \dots, N_l$.

We define the discontinuous finite element space as the space of the tensor product of piecewise polynomials of degree at most k in each variable on every element,

$$V_h = \{v \in L^2(\Omega),\ v|_{K_\alpha} \in Q^k(K_\alpha),\ \forall \alpha = 1, \dots, N\},$$

where Q^k denotes the tensor producto of P^k polynomials in each direction. Similar to the one-dimensional case, we introduce the following notation at cell interfaces $x^l = x^l_{\alpha_l + 1/2}$,

$$v^{\pm} = \lim_{\epsilon \to 0} v(x^1, \dots, x^l \pm \epsilon, \dots, x^d), \quad \{v\} = \frac{v^- + v^+}{2}, \quad [v] = v^+ - v^-.$$

The fully discrete DG scheme on each computational cell is to find $c^{n+1}_{ih}, \psi^{n+1}_h \in V_h$ such that for any $v_i, r_i, \eta \in V_h$,

$$\begin{aligned}
&\int_{K_\alpha} D_t c^n_{ih} v_i \, dx = -\int_{K_\alpha} c^n_{ih} \nabla p^n_{ih} \cdot \nabla v_i \, dx \\
&\quad + \sum_{l=1}^{d} \int_{K_\alpha \setminus I^l_{\alpha_l}} \{c^n_{ih}\} \left(\widehat{\partial_{x^l} p^n_{ih}} v_i + (p^n_{ih} - \{p^n_{ih}\}) \partial_{x^l} v_i \right) \Big|_{x^{l,+}_{\alpha_l - 1/2}}^{x^{l,-}_{\alpha_l + 1/2}} d(x \backslash x^l),
\end{aligned} \tag{26a}$$

$$\int_{K_\alpha} p_{ih} r_i \, dx = \int_{K_\alpha} \left(q_i \phi^n_h + \log c^n_{ih} \right) r_i \, dx, \tag{26b}$$

$$\begin{aligned}
&\int_{K_\alpha} \nabla \psi^n_h \cdot \nabla \eta \, dx - \sum_{l=1}^{d} \int_{K_\alpha \setminus I^l_{\alpha_l}} \left(\widehat{\partial_{x^l} \psi^n_h} \eta + (\psi^n_h - \{\psi^n_h\}) \partial_{x^l} \eta \right) \Big|_{x^{l,+}_{\alpha_l - 1/2}}^{x^{l,-}_{\alpha_l + 1/2}} d(x \backslash x^l) \\
&\quad = \int_{K_\alpha} \left(\sum_{i=1}^{m} q_i c^n_{ih} + \rho_0 \right) \eta \, dx,
\end{aligned} \tag{26c}$$

where at interior interface $(x^1, \dots, x^l_{\alpha_l + \frac{1}{2}}, \dots, x^d)$, the DDG numerical fluxes are given by

$$\widehat{\partial_{x^l} w} = \beta_0 \frac{[w]}{h^l_{\alpha_l + \frac{1}{2}}} + \{\partial_{x^l} w\} + \beta_1 h^l_{\alpha^l + \frac{1}{2}} [\partial^2_{x^l} w], \tag{27}$$

for

$$h^l_{\alpha_l + \frac{1}{2}} = x^l_{\alpha_l + 1} - x^l_{\alpha_l};$$

the modified numerical fluxes are defined as

$$\widetilde{\partial_{x^l} p^n_{ih}} = \widehat{\partial_{x^l} p^n_{ih}} + \frac{\widetilde{\beta}^l_i}{2} [c^n_{ih}], \tag{28}$$

with the functions

$$\widetilde{\beta}^l_i = \begin{cases} \frac{|\widehat{\partial_{x^l} p^n_{ih}}|}{\{c^n_{ih}\}}, & \text{if } \{c^n_{ih}\} > 0, \\ 0, & \text{if } \{c^n_{ih}\} = 0. \end{cases} \tag{29}$$

At the boundary $x^l = x^l_{\frac{1}{2}}$ or $x^l = x^l_{N_l + \frac{1}{2}}$, our guiding principle for defining numerical fluxes and averages is as follows: whenever exact boundary data is prescribed, it is used directly; otherwise, the interior numerical approximation is employed. To enforce the zero-flux boundary condition in (1c), we set

$$\widehat{\partial_{x^l} p^n_{ih}} = 0,\ \{p^n_{ih}\} = p^n_{ih},\ \{c^n_{ih}\} = c^n_{ih}.$$

The modified numerical flux takes the given data, namely,

$$\widetilde{\partial_{x^l} p^n_{ih}} = \widehat{\partial_{x^l} p^n_{ih}} = 0, \quad \widetilde{\beta}^l_i = 0.$$

For the potential ψ subject to mixed boundary condition as in (1d), we specify the numerical fluxes and the averages at the interface $x^l = x^l_{\frac{1}{2}}$ or $x^l = x^l_{N_l+\frac{1}{2}}$ on a case-by-case basis, depending on whether Dirichlet or Neumann conditions are imposed. If a Dirichlet boundary condition is prescribed at the boundary, then:

$$\widehat{\partial_{x^l}\psi_h^n} = \beta_0 \frac{\psi_h^{n,+} - \psi_D}{h^l_{\frac{1}{2}}} + \partial_{x^l}\psi_h^{n,+}, \ \{\psi_h^n\} = \psi_D, \quad \text{if } (x^1, \dots, x^l_{\frac{1}{2}}, \dots, x^d) \in \partial\Omega_D,$$

$$\widehat{\partial_{x^l}\psi_h^n} = \beta_0 \frac{\psi_D - \psi_h^{n,-}}{h^l_{N_l+\frac{1}{2}}} + \partial_{x^l}\psi_h^{n,-}, \ \{\psi_h^n\} = \psi_D, \quad \text{if } (x^1, \dots, x^l_{N_l+\frac{1}{2}}, \dots, x^d) \in \partial\Omega_D,$$

where

$$h^l_{1/2} = x^l_1 - x^l_{1/2}, \quad h^l_{N_l+1/2} = x^l_{N_l+1/2} - x^l_{N_l}.$$

If a Neumann boundary condition is prescribed, then:

$$\widehat{\partial_{x^l}\psi_h^n} = \sigma, \ \{\psi_h^n\} = \psi_h^{n,+}, \quad \text{if } (x^1, \dots, x^l_{\frac{1}{2}}, \dots, x^d) \in \partial\Omega_N,$$

$$\widehat{\partial_{x^l}\psi_h^n} = \sigma, \ \{\psi_h^n\} = \psi_h^{n,-}, \quad \text{if } (x^1, \dots, x^l_{N_l+\frac{1}{2}}, \dots, x^d) \in \partial\Omega_N.$$

As in the 1D case, we initialize c^0_{ih} by projecting c_i^{in} onto V_h so that

$$\int_{K_\alpha} c^0_{ih}(x)v(x)dx = \int_{K_\alpha} c_i^{\text{in}}(x)v(x)dx, \quad \forall v \in V_h, \quad i = 1, \dots, m. \tag{30}$$

With these initial and boundary conditions, the scheme is now fully defined.

4.2. Positivity propagation

We define the cell averages of c^n_{ih} on K_α as

$$\bar{c}^n_{i\alpha} := \frac{1}{|K_\alpha|}\int_{K_\alpha} c^n_{ih}dx,$$

where $|K_\alpha| = \Pi_{l=1}^d \Delta x^l_{\alpha_l}$ with $\Delta x^l_{\alpha_l} = x^l_{\alpha_l+1/2} - x^l_{\alpha_l-1/2}$. On the interval $I^l_{\alpha_l}$, $l = 1, \dots, d$, we introduce the M-point Gauss-Lobatto quadrature rule using pairs $(\omega_{m_l}, x^{l,*}_{m_l})$, $1 \le m_l \le M$, with $M \ge \frac{k+3}{2}$. This quadrature is exact for polynomials of degree up to k, and the nodes satisfy

$$x^{l,*}_1 = x^l_{\alpha_l-\frac{1}{2}}, \quad x^{l,*}_M = x^l_{\alpha_l+\frac{1}{2}}.$$

Similar to the 1D scheme (15), the following results hold for the multidimensional scheme (26a)-(26c).

Theorem 3. *For the fully discrete DG schemes (26a)-(26c), the cell average $\bar{c}^{n+1}_{i\alpha} > 0$ provided $c^n_{ih}(x) > 0$ and the CFL condition for $\forall 1 \le l \le d$ and $\forall \ell \ne l$,*

$$\Delta t \le \frac{\omega_1}{d} \min\left\{ \Delta x^l_{\alpha_l} |\widehat{\partial_{x^l} p^n_{ih}}|^{-1}\Big|_{(x^{1,*}_{m_1}, \dots, x^l_{\alpha_l\pm1/2}, \dots, x^{d,*}_{m_d})} \right\}, \quad 1 \le \alpha_l \le N_l, 1 \le m_\ell \le M, \tag{31}$$

is satisfied. Here, ω_1 represents the first weight of the Gauss-Lobatto quadrature rules with $M \ge \frac{k+3}{2}$ points.

Proof. Taking $v_i = \frac{\Delta t}{|K_\alpha|}$ in (26a), the time evolution of $\bar{c}^{n+1}_{i\alpha}$ on K_α is given by

$$\begin{aligned}
\bar{c}^{n+1}_{i\alpha} =& \bar{c}^n_{i\alpha} + \frac{\Delta t}{|K_\alpha|}\sum_{l=1}^d \int_{K_\alpha \setminus I^l_{\alpha_l}} \{c^n_{ih}\}\widehat{\partial_{x^l} p^n_{ih}}\Big|_{x^l_{\alpha_l+1/2}} - \{c^n_{ih}\}\widehat{\partial_{x^l} p^n_{ih}}\Big|_{x^l_{\alpha_l-1/2}} d(x\backslash x^l) \\
=& \sum_{l=1}^d \left(\frac{1}{d}\bar{c}^n_{ih} + \frac{\Delta t}{|K_\alpha|}\int_{K_\alpha \setminus I^l_{\alpha_l}} \{c^n_{ih}\}\widehat{\partial_{x^l} p^n_{ih}}\Big|_{x^l_{\alpha_l+1/2}} - \{c^n_{ih}\}\widehat{\partial_{x^l} p^n_{ih}}\Big|_{x^l_{\alpha_l-1/2}} d(x\backslash x^l) \right) \\
=& \frac{1}{d|K_\alpha|}\sum_{l=1}^d \Delta x^l_{\alpha_l} \int_{K_\alpha \setminus I^l_{\alpha_l}} \left(\frac{1}{\Delta x^l_{\alpha_l}}\int_{I^l_{\alpha_l}} c^n_{ih}dx^l + d\lambda^l_{\alpha_l}\Delta x^l_{\alpha_l}\{c^n_{ih}\}\widehat{\partial_{x^l} p^n_{ih}}\Big|_{x^l_{\alpha_l+1/2}} \right. \\
& \left. - d\lambda^l_{\alpha_l}\Delta x^l_{\alpha_l}\{c^n_{ih}\}\widehat{\partial_{x^l} p^n_{ih}}\Big|_{x^l_{\alpha_l-1/2}} \right) d(x\backslash x^l),
\end{aligned}$$

where

$$\lambda^l_{\alpha_l} := \frac{\Delta t}{(\Delta x^l_{\alpha_l})^2}.$$

We first apply the Gauss-Lobatto quadrature rule to approximate $\bar{c}^n_{i\alpha}$ on $I^l_{\alpha_l}$ and obtain

$$
\begin{aligned}
\bar{c}^{n+1}_{i\alpha} =& \frac{1}{d|K_\alpha|}\sum_{l=1}^{d}\Delta x^l_{\alpha_l}\int_{K_\alpha\backslash I^l_{\alpha_l}}\Bigg[\sum_{m_l=1}^{M}\omega_{m_l}c^n_{ih}(x^1,\dots,x^{l,*}_{m_l},\dots,x^d)\\
&+\frac{d\lambda^l_{\alpha_l}\Delta x^l_{\alpha_l}}{2}\left(\widehat{\partial_{x^l}p^n_{ih}}+\widetilde{\beta}^l_i\{c^n_{ih}\}\right)\Big|_{x^l_{\alpha_l+\frac12}}c^n_{ih}(x^1,\dots,x^{l,+}_{\alpha_l+\frac12},\dots,x^d)\\
&+\frac{d\lambda^l_{\alpha_l}\Delta x^l_{\alpha_l}}{2}\left(\widehat{\partial_{x^l}p^n_{ih}}-\widetilde{\beta}^l_i\{c^n_{ih}\}\right)\Big|_{x^l_{\alpha_l+\frac12}}c^n_{ih}(x^1,\dots,x^{l,-}_{\alpha_l+\frac12},\dots,x^d)\\
&-\frac{d\lambda^l_{\alpha_l}\Delta x^l_{\alpha_l}}{2}\left(\widehat{\partial_{x^l}p^n_{ih}}+\widetilde{\beta}^l_i\{c^n_{ih}\}\right)\Big|_{x^l_{\alpha_l-\frac12}}c^n_{ih}(x^1,\dots,x^{l,+}_{\alpha_l-\frac12},\dots,x^d)\\
&+\frac{d\lambda^l_{\alpha_l}\Delta x^l_{\alpha_l}}{2}\left(-\widehat{\partial_{x^l}p^n_{ih}}+\widetilde{\beta}^l_i\{c^n_{ih}\}\right)\Big|_{x^l_{\alpha_l-\frac12}}c^n_{ih}(x^1,\dots,x^{l,-}_{\alpha_l-\frac12},\dots,x^d)\Bigg]d(x\backslash x^l)\\
=&\frac{1}{d|K_\alpha|}\sum_{l=1}^{d}\Delta x^l_{\alpha_l}\int_{K_\alpha\backslash I^l_{\alpha_l}}\Bigg[\sum_{m_l=2}^{M-1}\omega_{m_l}c^n_{ih}(x^1,\dots,x^{l,*}_{m_l},\dots,x^d)\\
&+\frac{d\lambda^l_{\alpha_l}\Delta x^l_{\alpha_l}}{2}\left(\widehat{\partial_{x^l}p^n_{ih}}+\widetilde{\beta}^l_i\{c^n_{ih}\}\right)\Big|_{x^l_{\alpha_l+\frac12}}c^n_{ih}(x^1,\dots,x^{l,+}_{\alpha_l+\frac12},\dots,x^d)\\
&+\frac{d\lambda^l_{\alpha_l}\Delta x^l_{\alpha_l}}{2}\left(-\widehat{\partial_{x^l}p^n_{ih}}+\widetilde{\beta}^l_i\{c^n_{ih}\}\right)\Big|_{x^l_{\alpha_l-\frac12}}c^n_{ih}(x^1,\dots,x^{l,-}_{\alpha_l-\frac12},\dots,x^d)\\
&+\left(\omega_1-\frac{d\lambda^l_{\alpha_l}\Delta x^l_{\alpha_l}}{2}\left(\widehat{\partial_{x^l}p^n_{ih}}+\widetilde{\beta}^l_i\{c^n_{ih}\}\right)\Big|_{x^l_{\alpha_l-\frac12}}\right)c^n_{ih}(x^1,\dots,x^{l,+}_{\alpha_l-\frac12},\dots,x^d)\\
&+\left(\omega_M-\frac{d\lambda^l_{\alpha_l}\Delta x^l_{\alpha_l}}{2}\left(-\widehat{\partial_{x^l}p^n_{ih}}+\widetilde{\beta}^l_i\{c^n_{ih}\}\right)\Big|_{x^l_{\alpha_l+\frac12}}\right)c^n_{ih}(x^1,\dots,x^{l,-}_{\alpha_l+\frac12},\dots,x^d)\Bigg]d(x\backslash x^l).
\end{aligned}
$$

Based on the choice of $\widetilde{\beta}^l_i$ in (29), it follows

$$
\left(\widehat{\partial_{x^l}p^n_{ih}}+\widetilde{\beta}^l_i\{c^n_{ih}\}\right)\Big|_{(x^1,\dots,x^l_{\alpha_l+\frac12},\dots,x^d)}\geq 0,\quad \left(-\widehat{\partial_{x^l}p^n_{ih}}+\widetilde{\beta}^l_i\{c^n_{ih}\}\right)\Big|_{(x^1,\dots,x^l_{\alpha_l-\frac12},\dots,x^d)}\geq 0.
$$

Then applying the Gauss-Lobatto quadrature rule in all directions except x^l gives

$$
\begin{aligned}
\bar{c}^{n+1}_{i\alpha} =& \frac{1}{d|K_\alpha|}\sum_{l=1}^{d}\Delta x^l_{\alpha_l}\sum_{m_1=1}^{M}\omega_{m_1}\cdots\sum_{m_{l-1}=1}^{M}\omega_{m_{l-1}}\sum_{m_{l+1}=1}^{M}\omega_{m_{l+1}}\cdots\sum_{m_d=1}^{M}\omega_{m_d}\\
&\cdot\Bigg[\sum_{m_l=2}^{M-1}\omega_{m_l}c^n_{ih}(x^{1,*}_{m_1},\dots,x^{l,*}_{m_l},\dots,x^{d,*}_{m_d})\\
&+\frac{d\lambda^l_{\alpha_l}\Delta x^l_{\alpha_l}}{2}\left(\widehat{\partial_{x^l}p^n_{ih}}+\widetilde{\beta}^l_i\{c^n_{ih}\}\right)\Big|_{(x^{1,*}_{m_1},\dots,x^l_{\alpha_l+\frac12},\dots,x^{d,*}_{m_d})}c^n_{ih}(x^{1,*}_{m_1},\dots,x^{l,+}_{\alpha_l+\frac12},\dots,x^{d,*}_{m_d})\\
&+\frac{d\lambda^l_{\alpha_l}\Delta x^l_{\alpha_l}}{2}\left(-\widehat{\partial_{x^l}p^n_{ih}}+\widetilde{\beta}^l_i\{c^n_{ih}\}\right)\Big|_{(x^{1,*}_{m_1},\dots,x^l_{\alpha_l-\frac12},\dots,x^{d,*}_{m_d})}c^n_{ih}(x^{1,*}_{m_1},\dots,x^{l,-}_{\alpha_l-\frac12},\dots,x^{d,*}_{m_d})\\
&+\left(\omega_1-\frac{d\lambda^l_{\alpha_l}\Delta x^l_{\alpha_l}}{2}\left(\widehat{\partial_{x^l}p^n_{ih}}+\widetilde{\beta}^l_i\{c^n_{ih}\}\right)\Big|_{(x^{1,*}_{m_1},\dots,x^l_{\alpha_l-\frac12},\dots,x^{d,*}_{m_d})}\right)c^n_{ih}(x^{1,*}_{m_1},\dots,x^{l,+}_{\alpha_l-\frac12},\dots,x^{d,*}_{m_d})\\
&+\left(\omega_M-\frac{d\lambda^l_{\alpha_l}\Delta x^l_{\alpha_l}}{2}\left(-\widehat{\partial_{x^l}p^n_{ih}}+\widetilde{\beta}^l_i\{c^n_{ih}\}\right)\Big|_{(x^{1,*}_{m_1},\dots,x^l_{\alpha_l+\frac12},\dots,x^{d,*}_{m_d})}\right)c^n_{ih}(x^{1,*}_{m_1},\dots,x^{l,-}_{\alpha_l+\frac12},\dots,x^{d,*}_{m_d}))\Bigg]\\
>&\ 0,
\end{aligned}
$$

provided

$$
\lambda^l_{\alpha_l}\leq\frac{1}{d}\frac{\omega_1}{\Delta x^l_{\alpha_l}}\min\left\{|\widehat{\partial_{x^l}p^n_{ih}}|^{-1}\Big|_{(x^{1,*}_{m_1},\dots,x^l_{\alpha_l\pm1/2},\dots,x^{d,*}_{m_d})}\right\},\quad 1\leq m_\ell\leq M,\quad\forall\ell\neq l.
$$

Table 1
Errors for Example 1 at $T = 0.05$.

(k, β_0, β_1)	h	c_1 error	order	c_2 error	order	ψ error	order
$(1, 2, -)$	0.2	0.00074325	–	0.001946	–	0.00044667	–
	0.1	0.00015581	2.0164	0.00022054	2.6233	4.2569e-05	3.3250
	0.05	4.3138e-05	1.8976	3.8745e-05	2.3642	4.1411e-06	3.2918
	0.025	1.1223e-05	1.9425	8.3198e-06	2.2194	4.4382e-07	3.2220
$(2, 4, 1/20)$	0.2	0.0035405	–	0.0013336	–	0.00026474	–
	0.1	0.00075498	2.5660	0.00022787	2.6867	4.1164e-05	2.7921
	0.05	0.00011782	2.7343	3.4364e-05	2.7555	5.7354e-06	2.8456
	0.025	1.7049e-05	2.7889	4.9971e-06	2.7817	7.9669e-07	2.8478

Again, we used $\omega_1 = \omega_N$ due to the symmetry of Gauss-Lobatto quadrature rules.

□

Remark 4. If $\widetilde{\beta}_i^l \equiv 0$ in (28), then the DG scheme (26a)-(26c) reduces to the high-dimensional form of the fully discrete DG scheme described in (5).

5. Numerical examples

In this section, we present a set of selected examples to validate our positivity-preserving DDG scheme. For accuracy, we assess the order of accuracy by performing numerical convergence tests using discrete L^1 errors.

5.1. 1D Accuracy tests

We begin to study the impact of the positivity preserving flux $\widetilde{\partial_x p_{ih}}$ on the scheme accuracy. We revisit the example from §5.1 of [1], where the DDG scheme (5) was applied. Consider the domain $\Omega = [0, 1]$ and the PNP problem with source terms defined by the following system of equations:

$$\begin{aligned}
&\partial_t c_1 = \partial_x(\partial_x c_1 + q_1 c_1 \partial_x \psi) + f_1,\\
&\partial_t c_2 = \partial_x(\partial_x c_2 + q_2 c_2 \partial_x \psi) + f_2,\\
&-\partial_x^2 \psi = q_1 c_1 + q_2 c_2,\\
&\partial_x \psi(t, 0) = 0, \quad \partial_x \psi(t, 1) = -e^{-t}/60,\\
&\partial_x c_i + q_i c_i \partial_x \psi = 0, \quad x = 0, 1,
\end{aligned}$$

where the source terms are given by

$$\begin{aligned}
f_1 &= \frac{(50x^9 - 198x^8 + 292x^7 - 189x^6 + 45x^5)}{30e^{2t}} + \frac{(-x^4 + 2x^3 - 13x^2 + 12x - 2)}{e^t},\\
f_2 &= \frac{(x-1)(110x^9 - 430x^8 + 623x^7 - 393x^6 + 90x^5)}{60e^{2t}} + \frac{(x-1)(x^4 - 2x^3 + 21x^2 - 16x + 2)}{e^t}.
\end{aligned}$$

This system, with $q_1 = 1$ and $q_2 = -1$, admits exact solutions:

$$\begin{aligned}
c_1 &= x^2(1-x)^2 e^{-t},\\
c_2 &= x^2(1-x)^3 e^{-t},\\
\psi &= -(10x^7 - 28x^6 + 21x^5)e^{-t}/420.
\end{aligned}$$

Note that we impose $\psi(t, 0) = 0$ to select a specific solution, as ψ is unique up to an additive constant.

We solve this problem using the positivity-preserving DG scheme (15). Table 1 presents both L^1 errors and orders of convergence when using P^k elements at $T = 0.05$, incorporating the modified local flux $\widetilde{\partial_x p_{ih}}$. Our results indicate that the method preserves an order of convergence of $k + 1$, consistent with the findings in [1]. This suggests that the modified flux $\widetilde{\partial_x p_{ih}}$ maintains optimal $k + 1$ order of accuracy. We adhere to the CFL conditions outlined in Theorem 1 and Remark 2. Fig. 1 provides a visual comparison between numerical and exact solutions at $t = 0.05$.

5.2. 1D Tests on solution properties

We test solution properties such as solution positivity, mass conservation and free energy dissipation. We apply the proposed scheme to the one-dimensional PNP system (1) with $m = 2$ in $\Omega = [0, 1]$, defined by the following equations:

$$\begin{aligned}
&\partial_t c_1 = \partial_x(\partial_x c_1 + q_1 c_1 \partial_x \psi),\\
&\partial_t c_2 = \partial_x(\partial_x c_2 + q_2 c_2 \partial_x \psi),
\end{aligned}$$

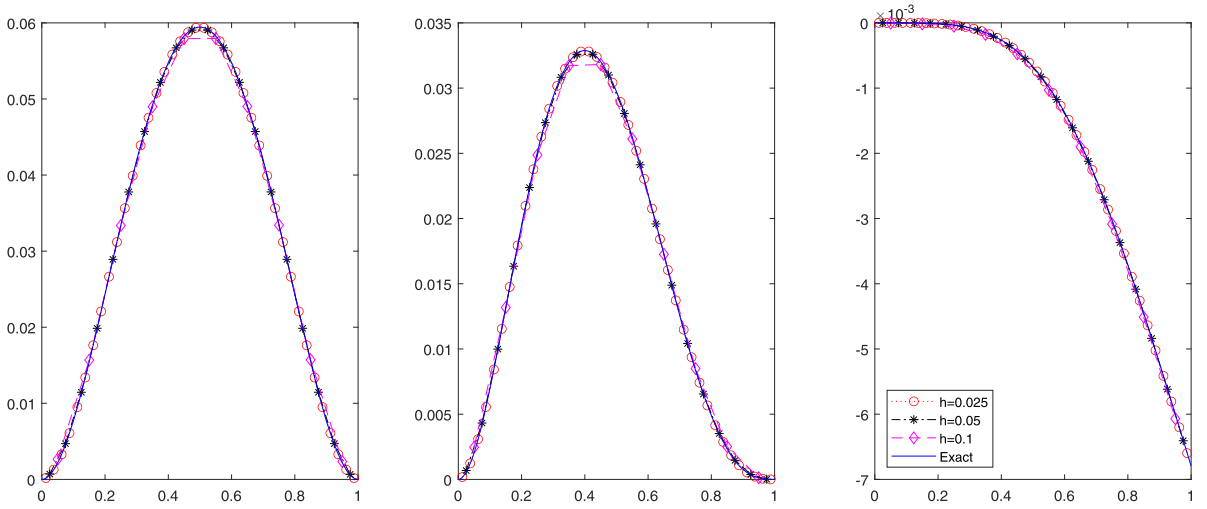
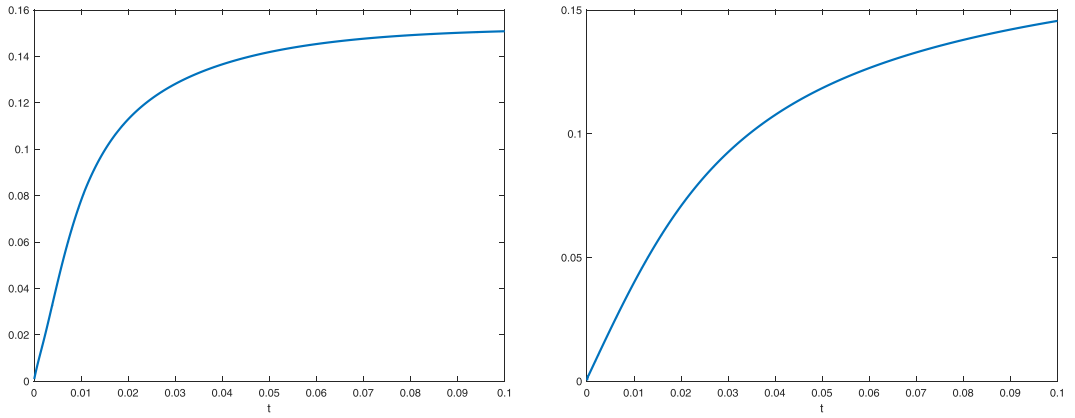


Fig. 1. Numerical solution versus exact solution at $T = 0.05$.



(a) Smallest cell average of c_1 (b) Smallest cell average of c_2

Fig. 2. Smallest cell averages of c_1, c_2.

$$
\begin{aligned}
&-\partial_x^2\psi = q_1c_1 + q_2c_2,\\
&c_1(0,x) = c_1^{\text{in}}(x), \quad c_2(0,x) = c_2^{\text{in}}(x),\\
&\psi(t,0) = 0, \quad \partial_x\psi(t,0) = 0, \quad \partial_x\psi(t,1) = 0,\\
&\partial_x c_i + q_i c_i \partial_x\psi = 0, \quad x = 0,\ 1.
\end{aligned}
$$

In this system, the ion charges are $q_1 = 1$ and $q_2 = -1$, and the initial concentrations are given by

$$
c_1^{\text{in}}(x) = \begin{cases} 0.1, & \text{if } x \in (0.4, 0.6),\\ 0.288, & \text{if } x \in [0.2, 0.4] \cup [0.6, 0.8],\\ 5x^2(1-x)^2, & \text{otherwise,} \end{cases}
$$

$$
c_2^{\text{in}}(x) = \frac{\pi}{10}\left|\sin(2\pi x^2)\right|.
$$

For the DG schemes applied to this example, we use P^1 polynomials with numerical flux parameters $\beta_0 = 4$. The initial time step is set to $\Delta t_0 = 3.5 \times 10^{-5}$, and the mesh size is $h = 0.025$.

Test Case 1. We solve this problem using the DG scheme (15) with a fixed time step $\Delta t = \Delta t_0$ and employ the modified flux $\widetilde{\partial_x p_{ih}}$ in (16) at each step. Fig. 2 shows the evolution of the smallest cell averages of c_1 and c_2, demonstrating that the DG scheme (15) conserves the positivity of the solutions at quadrature points.

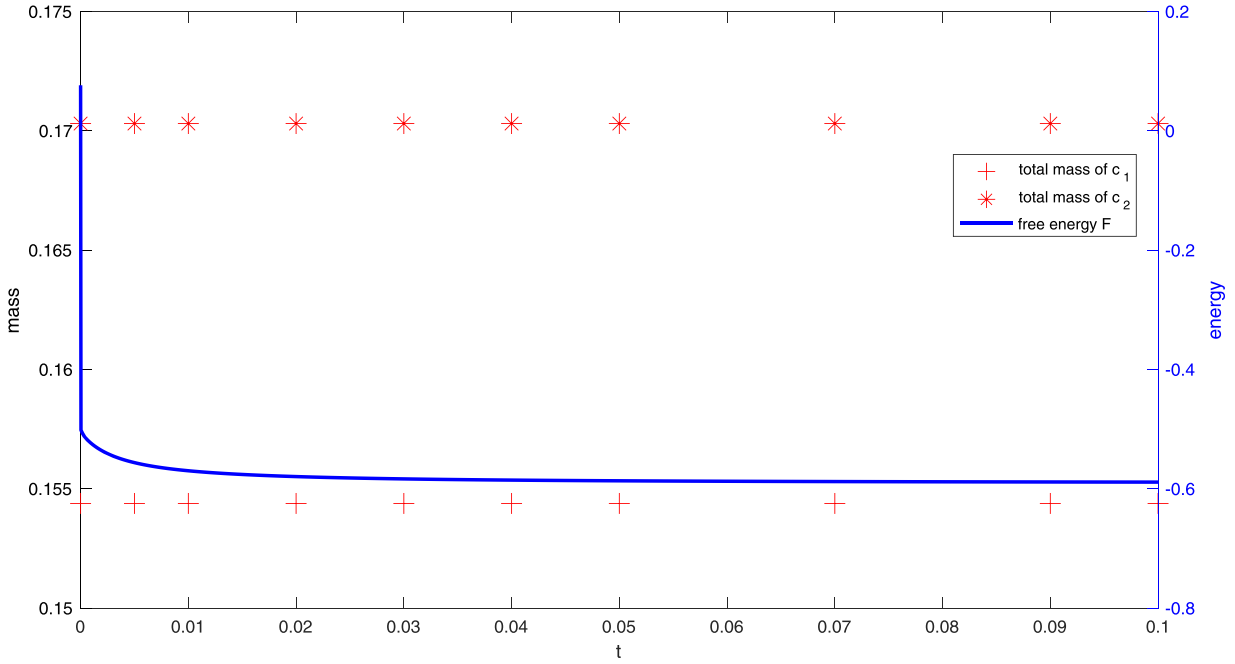


Fig. 3. Conservation of mass and decay of free energy.

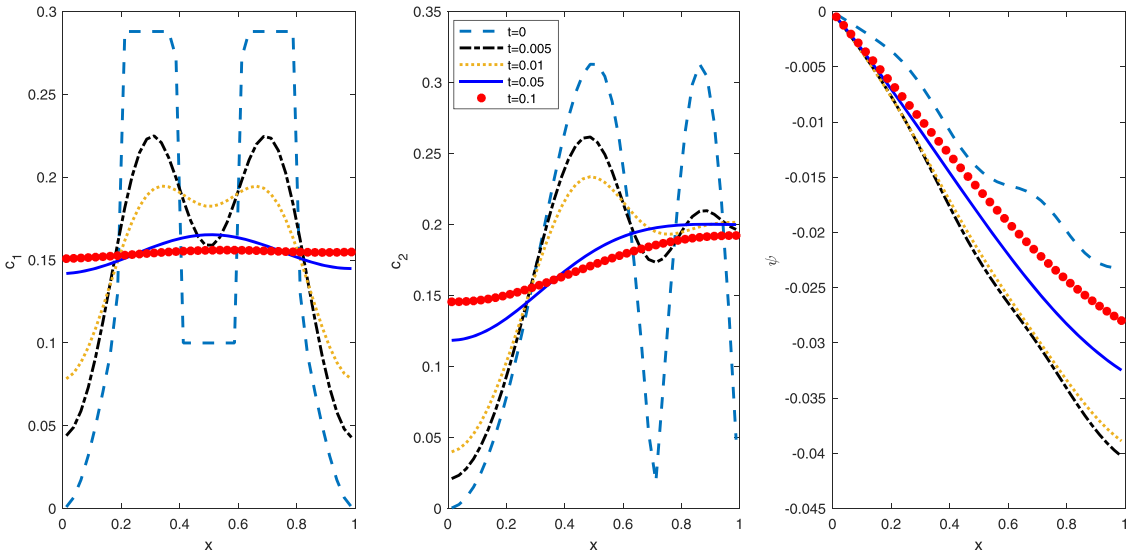
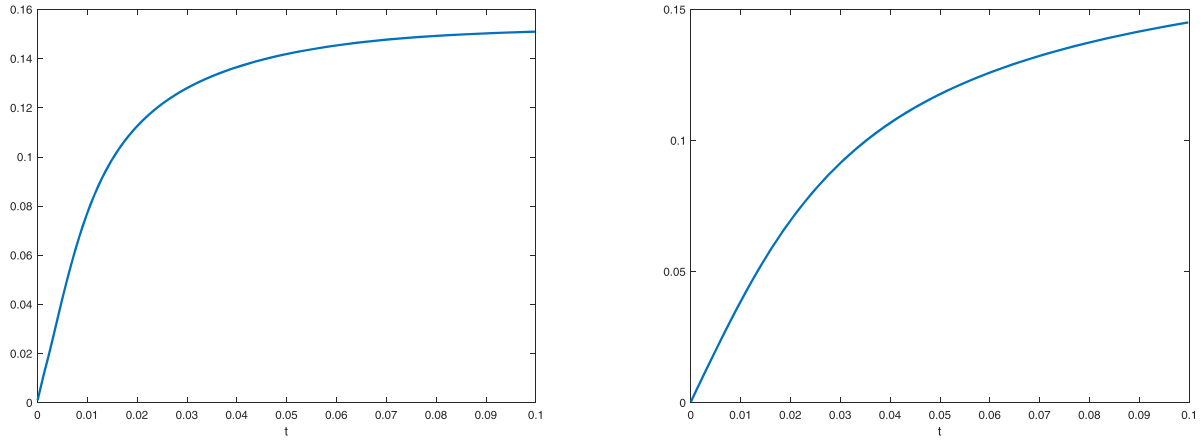


Fig. 4. Numerical solution evolution.

Fig. 3 displays the evolution of the total masses of c_1 and c_2, which are conserved at approximately 0.15439 for c_1 and 0.17031 for c_2. Additionally, Fig. 3 confirms the dissipation of free energy. Fig. 4 shows the evolution of the solutions, indicating that c_1 and c_2 approach equilibrium at $T = 0.1$.

Test Case 2. Next, we solve this problem using the DG scheme (5) from our previous work [1] with the same fixed time step $\Delta t = \Delta t_0$ as used in Test Case 1. However, the numerical results indicate that at $t = 0.000245$, the smallest cell average of the concentration c_2 on the quadrature points is -0.029306, indicating a failure of the DG scheme (5).

To address this issue, we modify the algorithm for the DG scheme (5). Specifically, if the smallest cell averages of c_1 and c_2 drop below 0 (or small threshold ε), we switch to using the DG scheme (15) at that particular time step. Additionally, we adjust the time step according to the CFL condition specified in (18) or Remark 2 with $\gamma = 1$.



(a) Smallest cell average of c_1 (b) Smallest cell average of c_2

Fig. 5. Smallest cell averages of c_1, c_2.

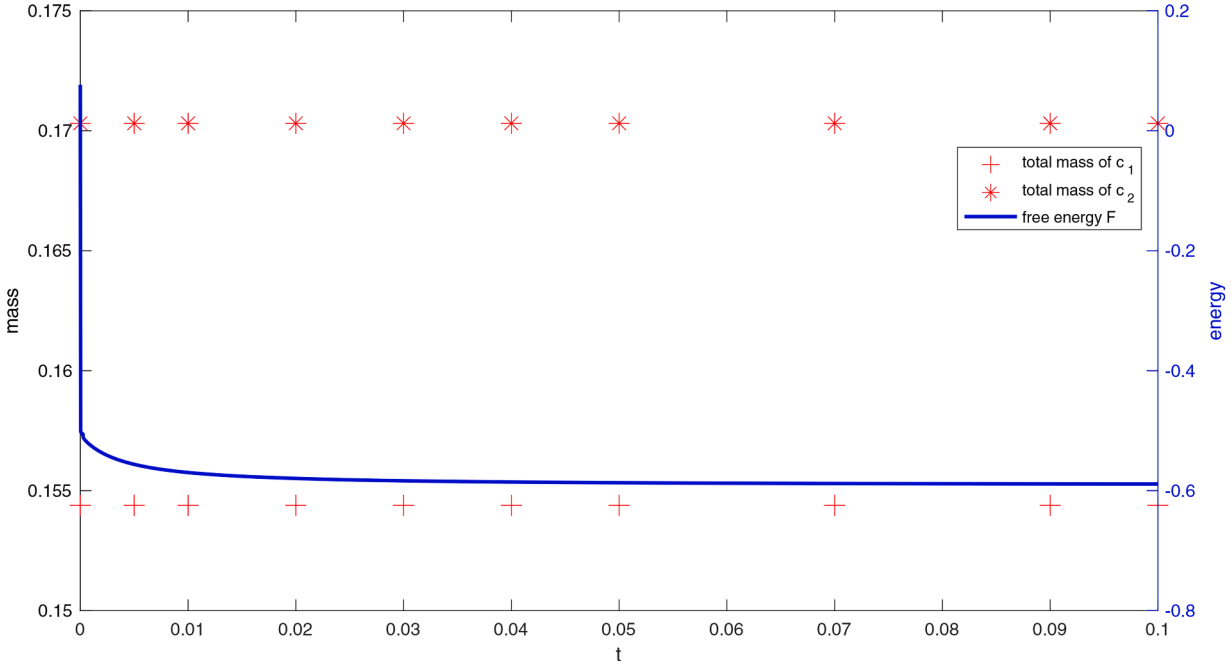


Fig. 6. Conservation of mass and decay of free energy.

Numerically, Fig. 5 shows that the modified algorithm effectively preserves the positivity of c_1 and c_2. Fig. 6 illustrates the decay of the free energy and confirms the conservation of total masses 0.15439 for c_1 and 0.17031 for c_2. Furthermore, Fig. 7 shows a similar evolution of the solutions as observed in Test Case 1.

From **Test Case 1** and **Test Case 2**, we see that our positivity preserving flux $\widetilde{\partial_x p_{ih}}$ will prevent negative cell average when applied at each time step following Algorithm 1, or only when needed (i.e., when (5) from [1] fails), while maintaining order of accuracy.

Test Case 3. We further consider an example using the same settings as in **Test Case 1**, with the following initial concentrations:

$$c_1^{\text{in}}(x) = \begin{cases} 0, & \text{if } x \in [0.425, 0.575], \\ (x-0.5)^2, & \text{otherwise}, \end{cases}$$

$$c_2^{\text{in}}(x) = \frac{\pi}{10}\left|\sin(2\pi x^2)\right|.$$

Since $c_1^{\text{in}}(x) = 0$ on an interval, the regularization of the initial condition described in Remark 2 sets $c_1^{\text{in}} = h^2$ in $[0.425, 0.575]$ when P^1 polynomials are employed. Fig. 8 shows the evolution of the smallest cell averages of c_1 and c_2, demonstrating that our hybrid scheme with flux (16) remains robust even when the initial concentrations vanish on a set with a non-empty interior. Fig. 9 (left)

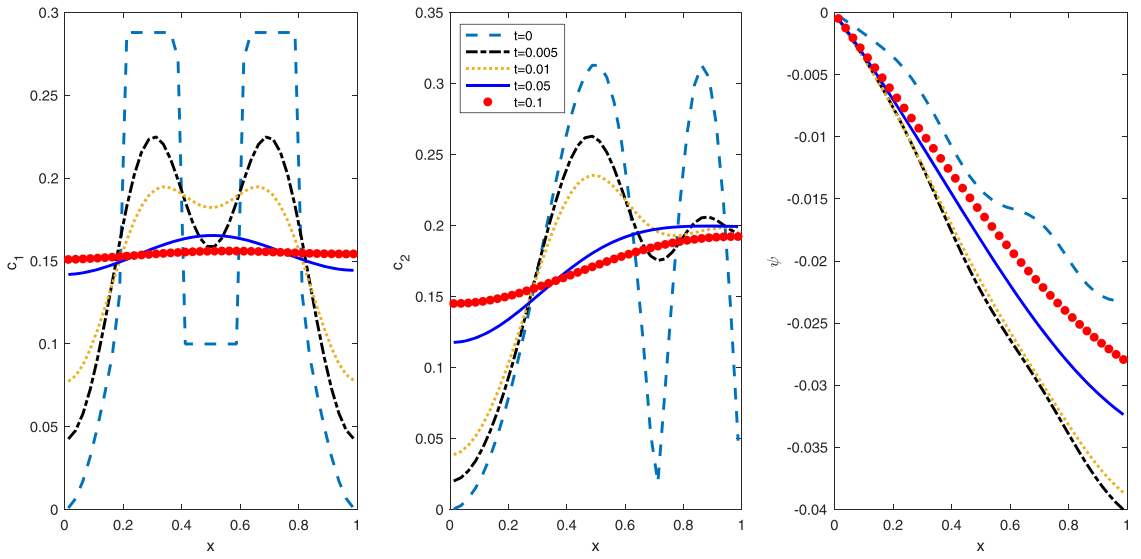
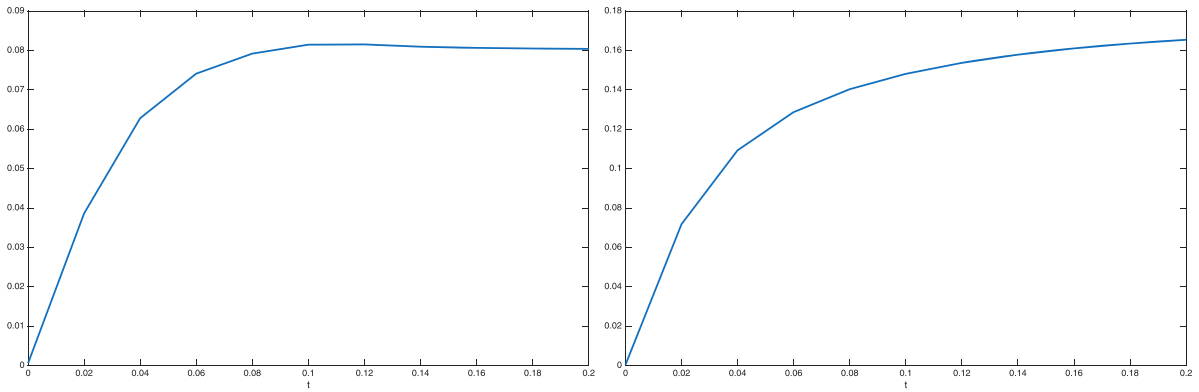


Fig. 7. Numerical solution evolution.



(a) Smallest cell average of c_1 (b) Smallest cell average of c_2

Fig. 8. Smallest cell averages of c_1, c_2.

confirms the conservation of the modified total mass for c_1 and c_2, while Fig. 9(right) shows the dissipation of free energy. Fig. 10 shows the evolution of the solutions, indicating that c_1 and c_2 approach equilibrium at $T = 0.2$.

5.3. 2D Accuracy tests

We evaluate the spatial accuracy of our scheme in a 2D setting using the PNP problem (1) defined on $\Omega = [0, \pi]^2$, with prescribed source terms:

$$\partial_t c_1 = \nabla \cdot (\nabla c_1 + c_1 \nabla \psi) + f_1,$$
$$\partial_t c_2 = \nabla \cdot (\nabla c_2 - c_2 \nabla \psi) + f_2,$$
$$-\Delta \psi = c_1 - c_2 + f_3.$$

Here, the functions $f_i(t, x, y)$ are determined by the exact solutions $c_1(t, x, y)$, $c_2(t, x, y)$, and $\psi(t, x, y)$, which are specified for each test case. The initial conditions in (1c) are derived by evaluating the exact solution at $t = 0$, while the boundary conditions in (1c) are set to zero flux. We denote the boundary sets as $\partial\Omega_D = \{(x, y) \in \bar{\Omega} : x = 0, x = \pi\}$ and $\partial\Omega_N = \partial\Omega \backslash \partial\Omega_D$. The boundary data in (1d) is derived from evaluating the exact solution $\psi(t, x, y)$ on $\partial\Omega_D$, and its normal derivative $\frac{\partial \psi}{\partial \mathbf{n}}$ on $\partial\Omega_N$. The problem is tackled using the fully discrete DG scheme (5) in a 2D setting (for more details, refer to Remark 4), along with the DG scheme (26a)-(26c), which has been adjusted to incorporate the specific source terms.

Test Case 1. We consider a system, with particular source terms f_1 and f_2, that admits the following exact solutions

$$c_1(t, x, y) = \alpha_1 \left(e^{-\alpha t} \cos(x) \cos(y) + 2 \right),$$

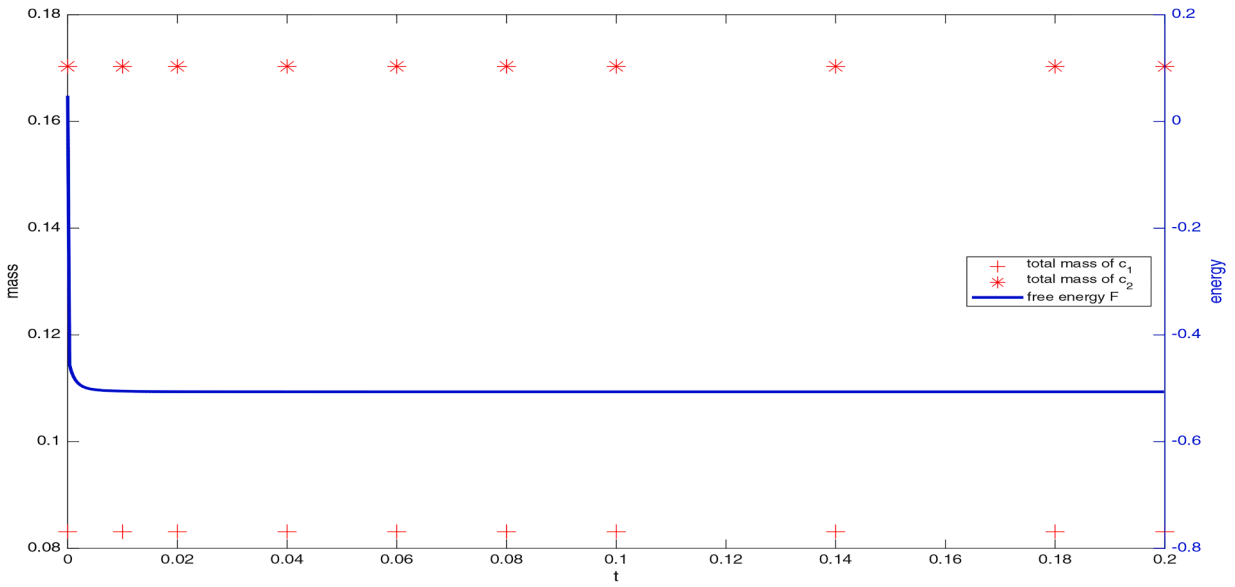


Fig. 9. Conservation of mass and decay of free energy.

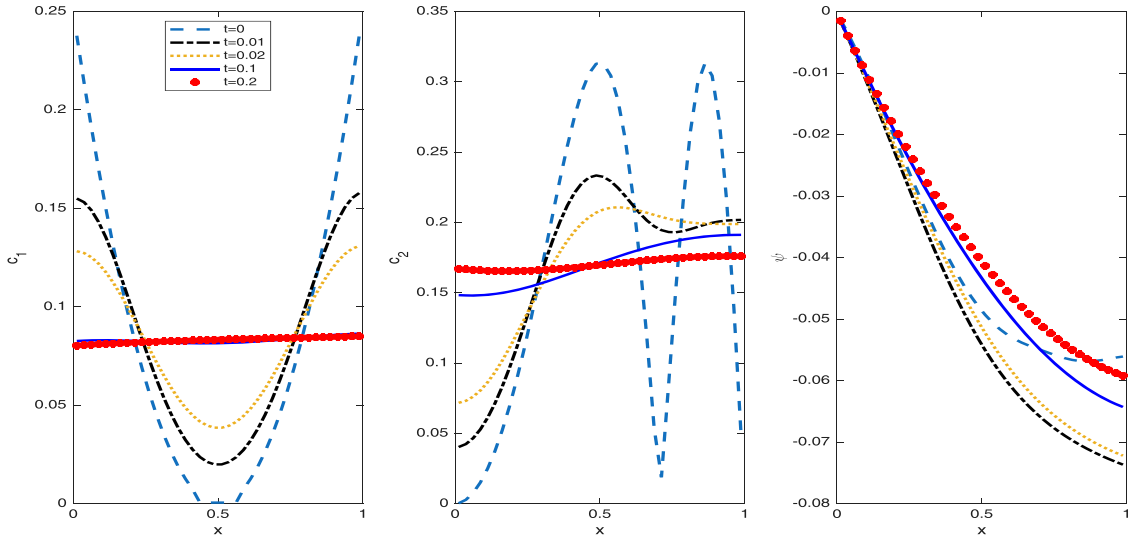


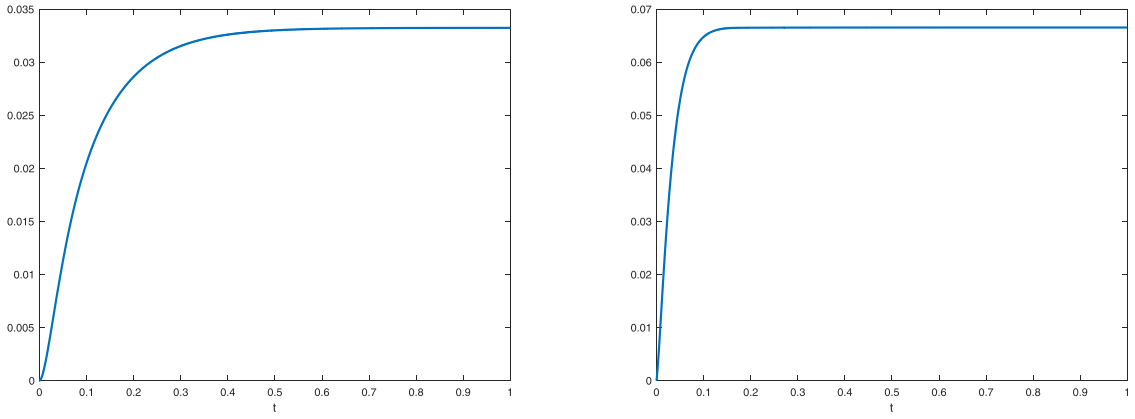
Fig. 10. Numerical solution evolution.

$$c_2(t,x,y) = \alpha_2\left(e^{-\alpha t}\cos(x)\cos(y) + 2\right),$$
$$\psi(t,x,y) = \alpha_3 e^{-\alpha t}\cos(x)\cos(y),$$

where the parameters are set as $\alpha = \alpha_1 = 2\alpha_2 = \alpha_3 = 10^{-2}$. The numerical flux parameters are $\beta_0 = 2$ and $\beta_1 = 1/12$, which are taken from [11,12]. To examine the spatial convergence rate, we choose the time step such that the spatial discretization error dominates while satisfying the CFL conditions stated in Theorem 3. Clearly, such a choice should depend on both k and h. For the numerical tests with results presented in Table 2-Table 4, we find it sufficient to select a fixed time step with $\Delta t = 10^{-(k+1)}h^2$ for the DG scheme (26a)-(26c). Table 2-Table 4 present the L^1, L^2, and L^∞ errors of the DG solutions based on P^k polynomials with $k = 1, 2, 3$. The results show that all errors converge to zero at an order of $k+1$. Therefore, in the subsequent numerical tests, we report only the L^1 error.

Test Case 2. We consider the exact solutions:

$$c_1(t,x,y) = \alpha_1\left(e^{-\alpha t}\cos(x)\cos(y) + 1\right),$$
$$c_2(t,x,y) = \alpha_2\left(e^{-\alpha t}\cos(x)\cos(y) + 1\right),$$
$$\psi(t,x,y) = \alpha_3 e^{-\alpha t}\cos(x)\cos(y),$$



(a) Smallest cell average of c_1. (b) Smallest cell average of c_2.

Fig. 11. Smallest cell averages of c_1, c_2.

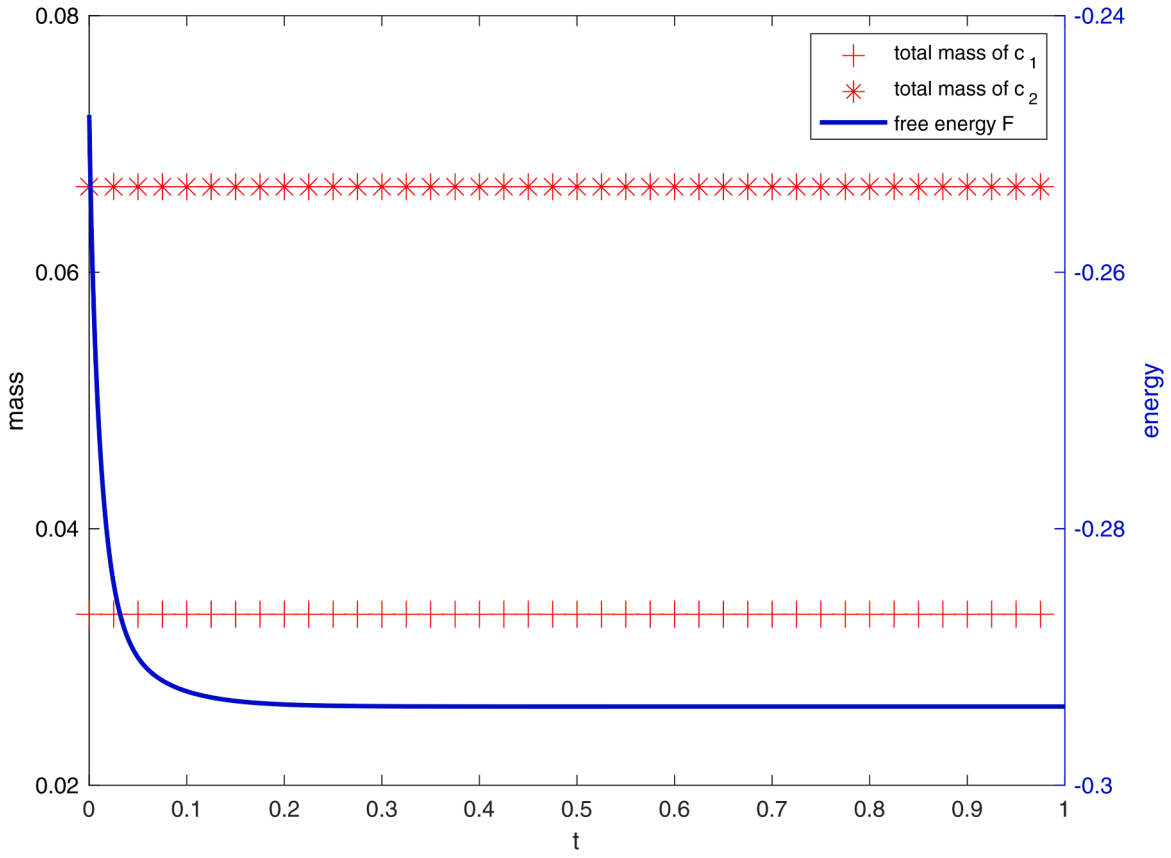


Fig. 12. Conservation of mass and decay of free energy.

where the parameters are set as $\alpha = \alpha_1 = 2\alpha_2 = \alpha_3 = 10^{-2}$. It is observed that $c_1 \geq 0$ and $c_2 \geq 0$ for $t \geq 0$, with $\min_{(x,y)} c_1 = 0$ and $\min_{(x,y)} c_2 = 0$ achievable only at $t = 0$.

The numerical flux parameters β_0, β_1 in the DDG schemes follow the guidelines from [41,42], and the specific values of β_0, β_1 can be found in Table 5 and Table 6. A time step of $\Delta t = 10^{-(k+1)}h^2$ is chosen for both DG schemes.

Table 5 presents the L^1 errors and orders of convergence at $t = 0.1$ for the DG scheme (5) in the 2D setting with $N \times N$ meshes. Additionally, results for the DG scheme (26a)-(26c), using the numerical fluxes (28) at each time step, are reported in Table 6. These

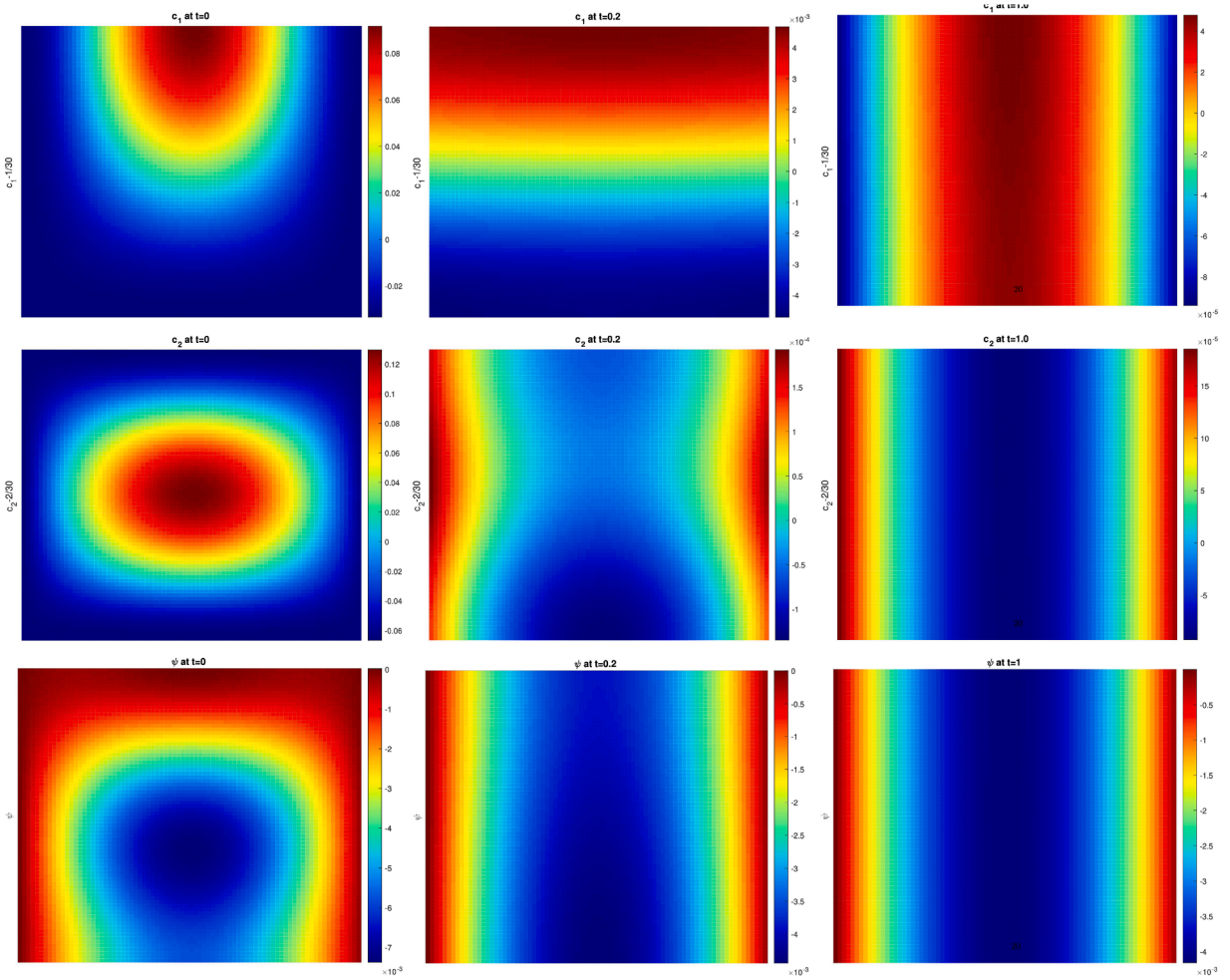


Fig. 13. The contours evolution of $c_1 - 1/30$, $c_2 - 2/30$ and ψ.

Table 2
Errors and orders of convergence for Test Case 1 in Section 5.3 for the DG scheme (26a)-(26c) with P^1 polynomials, meshes $N \times N$ at $T = 0.1$.

error type	N	c_1 error	order	c_2 error	order	ψ error	order
L^1 errors	10	3.85935e-04	–	1.92517e-04	–	6.57306e-04	–
	20	9.45627e-05	2.03	4.72446e-05	2.03	1.60531e-04	2.03
	30	4.18052e-05	2.01	2.08857e-05	2.01	7.00864e-05	2.04
	40	2.34677e-05	2.01	1.17247e-05	2.01	3.90013e-05	2.04
L^2 errors	10	1.63591e-04	–	8.15444e-05	–	2.98864e-04	–
	20	4.05095e-05	2.01	2.02155e-05	2.01	6.92868e-05	2.11
	30	1.79575e-05	2.01	8.96478e-06	2.01	2.93974e-05	2.11
	40	1.00903e-05	2.00	5.03826e-06	2.00	1.60776e-05	2.10
L^∞ errors	10	2.44245e-04	–	1.22173e-04	–	6.16937e-04	–
	20	6.22895e-05	1.97	3.11681e-05	1.97	1.65259e-04	1.90
	30	2.77985e-05	1.99	1.39107e-05	1.99	7.44373e-05	1.97
	40	1.56598e-05	1.99	7.83651e-06	1.99	4.20696e-05	1.98

results demonstrate that both schemes achieve $(k+1)$-th order of accuracy in space, confirming that the modified numerical fluxes do not compromise the accuracy of the numerical solutions.

Test Case 3. We modify the exact solution from Test Case 2 as follows:

$$c_1(t,x,y) = \alpha_1 e^{-\alpha t}(\cos(x)\cos(y)+1),$$
$$c_2(t,x,y) = \alpha_2 e^{-\alpha t}(\cos(x)\cos(y)+1),$$

Table 3
Errors and orders of convergence for Test Case 1 in Section 5.3 for the DG scheme (26a)-(26c) with P^2 polynomials, meshes $N \times N$ at $T = 0.1$.

error type	N	c_1 error	order	c_2 error	order	ψ error	order
L^1 errors	10	1.93889e-05	–	9.68440e-06	–	2.13254e-05	–
	20	2.30225e-06	3.07	1.15065e-06	3.07	2.44165e-06	3.13
	30	6.73857e-07	3.03	3.36838e-07	3.03	7.03237e-07	3.07
	40	2.82902e-07	3.02	1.41423e-07	3.02	2.92728e-07	3.05
L^2 errors	10	8.80869e-06	–	4.39905e-06	–	9.71548e-06	–
	20	1.07322e-06	3.04	5.36315e-07	3.04	1.13147e-06	3.10
	30	3.16278e-07	3.01	1.58082e-07	3.01	3.26827e-07	3.06
	40	1.33175e-07	3.01	6.65695e-08	3.01	1.36133e-07	3.04
L^∞ errors	10	1.52274e-05	–	7.59902e-06	–	2.97439e-05	–
	20	1.87084e-06	3.02	9.35758e-07	3.02	3.83554e-06	2.96
	30	5.51976e-07	3.01	2.76065e-07	3.01	1.14294e-06	2.99
	40	2.32209e-07	3.01	1.16136e-07	3.01	4.83137e-07	2.99

Table 4
Errors and orders of convergence for Test Case 1 in Section 5.3 for the DG scheme (26a)-(26c) with P^3 polynomials, meshes $N \times N$ at $T = 0.1$.

error type	N	c_1 error	order	c_2 error	order	ψ error	order
L^1 errors	10	1.01624e-06	–	5.08885e-07	–	9.63443e-07	–
	20	5.63171e-08	4.17	2.81710e-08	4.18	5.75301e-08	4.07
	30	1.08817e-08	4.05	5.44201e-09	4.05	1.13094e-08	4.01
	40	3.42318e-09	4.02	1.71190e-09	4.02	3.57488e-09	4.00
L^2 errors	10	5.02332e-07	–	2.51498e-07	–	4.28336e-07	–
	20	2.72926e-08	4.20	1.36511e-08	4.20	2.57910e-08	4.05
	30	5.28341e-09	4.05	2.64225e-09	4.05	5.08222e-09	4.01
	40	1.66107e-09	4.02	8.30683e-10	4.02	1.60776e-09	4.00
L^∞ errors	10	1.86301e-06	–	9.32333e-07	–	9.47550e-07	–
	20	7.54018e-08	4.63	3.76428e-08	4.63	5.36137e-08	4.14
	30	1.39411e-08	4.16	6.96154e-09	4.16	1.07042e-08	3.97
	40	4.20394e-09	4.17	2.09997e-09	4.17	3.40215e-09	3.98

Table 5
Errors and orders of convergence for Test Case 2 in Section 5.3 for the DG scheme (5) in the 2D setting with meshes $N \times N$ at $T = 0.1$.

(k, β_0, β_1)	N	c_1 error	order	c_2 error	order	ψ error	order
$(1, 3, -)$	10	4.35298e-04	–	2.17789e-04	–	6.23731e-04	–
	20	1.09438e-04	1.99	5.47851e-05	1.99	1.67117e-04	1.90
	30	4.86339e-05	2.00	2.43496e-05	2.00	7.55204e-05	1.96
	40	2.73409e-05	2.00	1.36889e-05	2.00	4.27726e-05	1.98
$(2, 9, \frac{1}{12})$	10	2.62719e-05	–	1.31419e-05	–	2.05689e-05	–
	20	3.17503e-06	3.05	1.58789e-06	3.05	2.38879e-06	3.11
	30	9.09636e-07	3.08	4.54895e-07	3.08	6.92827e-07	3.05
	40	3.75304e-07	3.08	1.87676e-07	3.08	2.89435e-07	3.03
$(3, 19, \frac{1}{12})$	10	5.34967e-06	–	2.67516e-06	–	7.94054e-07	–
	20	3.60228e-07	3.89	1.80121e-07	3.89	4.82770e-08	4.04
	30	6.93756e-08	4.06	3.46887e-08	4.06	9.46406e-09	4.02
	40	2.07955e-08	4.19	1.03979e-08	4.19	2.98476e-09	4.01

$$\psi(t, x, y) = \alpha_3 e^{-\alpha t} \cos(x) \cos(y),$$

where the parameters are set as $\alpha = \alpha_1 = 2\alpha_2 = \alpha_3 = 10^{-2}$. It can be verified that the densities $c_1 \geq 0$ and $c_2 \geq 0$ for $t \geq 0$, and $\min_{(x,y)} c_1 = 0$ and $\min_{(x,y)} c_2 = 0$ are achieved for any $t \geq 0$.

For this test case, we consider the DG scheme (26a)-(26c) based on P^2 polynomials. Following the guidelines in [16] for the third-order positivity-preserving DDG method for PNP equations, we select the numerical flux parameters $\beta_0 = 16$ and $\beta_1 = \frac{1}{6}$. The initial time step is set according to the DDG method in [16] to ensure positive average values of densities c_1 and c_2 in each computational cell, specifically $\Delta t = 5 \times 10^{-3} h^2$ for both DG schemes in this paper.

In the DG scheme (26a)-(26c), we use the numerical fluxes (28) selectively when (5) produces negative cell averages. To ensure positive cell averages, we adjust the time step to satisfy (24) in the 2D setting, where the parameter γ depends on the total number of the quadrature points in each direction.

Table 6
Errors and orders of convergence for Test Case 2 in Section 5.3 for the DG scheme (26a)-(26c) with meshes $N \times N$ at $T = 0.1$.

(k, β_0, β_1)	N	c_1 error	order	c_2 error	order	ψ error	order
$(1, 3, -)$	10	4.36263e-04	–	2.18339e-04	–	6.23863e-04	–
	20	1.09719e-04	1.99	5.49267e-05	1.99	1.67132e-04	1.90
	30	4.87287e-05	2.00	2.43965e-05	2.00	7.55255e-05	1.96
	40	2.73888e-05	2.00	1.37128e-05	2.00	4.27750e-05	1.98
$(2, 9, 1/12)$	10	2.61752e-05	–	1.30945e-05	–	2.05701e-05	–
	20	3.16680e-06	3.05	1.58381e-06	3.05	2.38883e-06	3.11
	30	9.07046e-07	3.08	4.53606e-07	3.08	6.92832e-07	3.05
	40	3.74168e-07	3.08	1.87111e-07	3.08	2.89436e-07	3.03
$(3, 19, 1/12)$	10	5.29795e-06	–	2.64941e-06	–	7.94046e-07	–
	20	3.57902e-07	3.89	1.78959e-07	3.89	4.82773e-08	4.04
	30	6.90252e-08	4.06	3.45136e-08	4.06	9.46415e-09	4.02
	40	2.07029e-08	4.19	1.03517e-08	4.19	2.98479e-09	4.01

Table 7
L^1 errors and orders at $t = 0.01$ with meshes $N \times N$, m Gauss-Lobatto quadrature points in each direction, and the parameter γ in (24).

(m, γ)	N	c_1 error	order	c_2 error	order	ψ error	order
$(3, 1/4)$	10	3.65605e-03	–	1.73140e-03	–	2.48497e-04	–
	20	6.15627e-05	5.89	1.23012e-05	7.14	7.67527e-06	5.02
	30	4.70311e-06	6.34	2.34902e-06	4.08	1.03025e-06	4.95
	40	2.51946e-06	–	1.29628e-06	–	4.09384e-07	3.21
$(4, 1/2)$	10	7.73601e-05	–	3.49995e-05	–	2.69952e-05	–
	20	4.71377e-06	4.04	2.26200e-06	3.95	3.15719e-06	3.10
	30	1.06658e-06	3.67	5.29898e-07	3.58	9.23417e-07	3.03
	40	4.29086e-07	3.17	2.13824e-07	3.15	3.87547e-07	3.02
$(5, 1)$	10	1.26670e-04	–	4.12020e-05	–	2.48704e-05	–
	20	4.35146e-06	4.86	2.09879e-06	4.30	2.68506e-06	3.21
	30	1.08331e-06	3.43	5.37169e-07	3.36	7.71977e-07	3.07
	40	4.37280e-07	3.15	2.18748e-07	3.12	3.21122e-07	3.05

We calculate the L^1 errors and orders of convergence at $t = 0.01$ using various numbers of Gauss-Lobatto quadrature points and different values of γ. The results are summarized in Table 7. Notably, we observe third-order accuracy and find that as the total number of quadrature points increases, γ can be set to a larger value. This adjustment improves the approximation of the parameter γ in (24), thereby enhancing numerical stability.

Finally, we analyze the evolution of c_1, c_2, and ψ for $t \in (0, 1]$. Fig. 13 shows the contours of $c_1 - 1/30$ (first column), $c_2 - 2/30$ (second column) and ψ (third column) at $t = 0, 0.2, 0.5$, and $t = 1$. Notably, the contours of c_2 and ψ at $t = 0.5$ and $t = 1.0$ are nearly identical, suggesting that the solution approaches a steady state by $t = 1$.

5.4. 2D Tests on solution properties

In this 2D example, we test solution properties, including solution positivity, mass conservation, and free energy dissipation. Following a methodology similar to [16, Example 4], we assess the effectiveness of the proposed scheme in solving the two-dimensional PNP system (1) with $m = 2$ in $\Omega = [0, 1]^2$, given by

$$
\begin{aligned}
&\partial_t c_1 = \nabla \cdot (\nabla c_1 + c_1 \nabla \psi),\\
&\partial_t c_2 = \nabla \cdot (\nabla c_2 - c_2 \nabla \psi),\\
&-\Delta \psi = c_1 - c_2,\\
&c_1^{\text{in}}(x, y) = \frac{1}{2} x^2 (1 - x)^2 (1 - \cos(\pi y)),\\
&c_2^{\text{in}}(x, y) = \pi \sin(\pi x) y^2 (1 - y)^2,\\
&\frac{\partial c_i}{\partial \mathbf{n}} + q_i c_i \frac{\partial \psi}{\partial \mathbf{n}} = 0, \quad (x, y) \in \partial\Omega,\\
&\psi = 0 \ \text{ on } \partial\Omega_D, \ \text{ and } \frac{\partial \psi}{\partial \mathbf{n}} = 0 \ \text{ on } \partial\Omega_N, \quad t > 0,
\end{aligned}
$$

where $\partial\Omega_D = \{(x, y) \in \bar{\Omega} : x = 0, x = 1\}$ and $\partial\Omega_N = \partial\Omega \backslash \partial\Omega_D$, with ion charges $q_1 = 1$ and $q_2 = -1$.

For this example, the DG schemes use P^2 polynomials with numerical flux parameters $\beta_0 = 16$ and $\beta_1 = \frac{1}{6}$. A 20×20 mesh is used with five-point Gauss-Lobatto quadrature. A fixed time step $\Delta t = 10^{-5}$ is applied for the original DG scheme (5) in **Test Cast 1**, while the adaptive time stepping strategy in Algorithm 1, based on the CFL condition (24) with an initial time step $\Delta t_0 = 10^{-5}$ and $\gamma = 1$, is used for the hybrid DG scheme (26) in **Test Case 2 and 3**.

Test Case 1. We first solve this problem using the DG scheme (5) in the 2D setting. At $t = 4 \times 10^{-5}$, the minimum cell average of c_1 reaches -1.8946×10^{-6}, indicating a loss of positivity and thus failure of the scheme.

Test Case 2. We then apply the DG scheme (26a)-(26c), where the numerical fluxes (28) are selectively employed at time steps for which scheme (5) produces negative cell averages. To examine positivity preservation, Fig. 11(a) and 11(b) show the evolution of the smallest cell averages of c_1, c_2 for $t \in [0, 1]$. As shown in Fig. 11, both c_1 and c_2 maintain strictly positive cell averages through the entire simulation period.

Test Case 3. Regarding mass conservation and free energy dissipation, Fig. 12 illustrates the temporal evolution of total mass for both c_1 and c_2. The results confirm that the DG scheme conserves total mass, $\frac{1}{30}$ for c_1 and $\frac{2}{30}$ for c_2. Furthermore, Fig. 12 shows the energy evolution, demonstrating that the scheme correctly captures the dissipation of free energy over time.

6. Concluding remarks

In this paper, we develop a novel positivity-preserving numerical flux to replace the DDG flux introduced in [1] for the PNP equations. By employing Gauss-Lobatto quadrature in the new flux formulation, we establish that the scheme preserves positive cell-averaged concentrations under a forward Euler time discretization, provided a proper CFL condition is satisfied. To ensure point-wise positivity, which is crucial for the validity of our logarithmic reformulations, we apply a subsequent positivity-preserving reconstruction. This hybrid algorithm fulfills the goal of constructing arbitrarily high-order (in space) positivity-preserving DDG schemes for the PNP equations.

Our extensive numerical experiments in both one and two-dimensional settings also demonstrate mass conservation and energy dissipation. However, due to the coupling between the positivity-preserving numerical flux and the reconstruction limiter, rigorously proving discrete energy dissipation remains a challenging task and is left for future work, (Table 3).

CRediT authorship contribution statement

Hailiang Liu: Writing – review & editing, Writing – original draft, Supervision, Methodology, Investigation, Formal analysis, Conceptualization; **Zhongming Wang:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization; **Peimeng Yin:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization.

Data availability

No data was used for the research described in the article.

Declaration of competing interest

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

References

[1] H. Liu, Z. Wang, A free energy satisfying discontinuous galerkin method for one-dimensional poisson-nernst-planck systems, J. Comput. Phys. 328 (2017) 413–437.
[2] M.S. Mock, Analysis of Mathematical Models of Semiconductor Devices, 3, Boole Press, 1983.
[3] P. Markowich, The Stationary Semiconductor Device Equations, Springer-Verlag, New York, New York, 1986.
[4] P.A. Markowich, C.A. Ringhofer, C. Schmeiser, Semiconductor Equations, Springer-Verlag Inc, New York, New York, 1990.
[5] J. Jerome, Analysis of Charge Transport: A Mathematical Study of Semiconductor Devices, Springer, Berlin, Berlin, 1996.
[6] S. Datta, Electronic Transport in Mesoscopic Systems, Cambridge University Press, 1997.
[7] D. Li, Electrokinetics in Microfluidics, Academic Press, 2004.
[8] S. Glasstone, An introduction to Electrochemstry, Van Nostrand Company, Inc., Princeton, NJ, Inc., Princeton, NJ, 1942.
[9] B. Hille, Ion channels of excitable membranes, 507, Sinauer Sunderland, MA, 2001.
[10] B. Eisenberg, W. Liu, Poisson-nernst-planck systems for ion channels with permanent charges, SIAM J. Math. Anal. 38 (2007) 1932–1966.
[11] H. Liu, J. Yan, The direct discontinuous galerkin (DDG) methods for diffusion problems, SIAM J. Numer. Anal 47 (2009) 675–698.
[12] H. Liu, J. Yan, The direct discontinuous galerkin (DDG) method for diffusion with interface corrections, Commun. Comput. Phys. 8 (3) (2010) 541–564.
[13] X. Zhang, C.W. Shu, On maximum-principle-satisfying high order schemes for scalar conservation laws, J. Comput. Phys. 229 (2010) 3091–3120.
[14] J.A. Carrillo, H. Liu, H. Yu, Positivity-preserving and energy-dissipating discontinuous galerkin methods for nonlinear nonlocal fokker-planck equations, Commun. Appl. Ind. Math. (2025).
[15] H. Liu, H. Yu, The entropy satisfying discontinuous galerkin method for fokker–planck equations, J. Sci. Comput. 62 (2015) 803–830.
[16] H. Liu, Z. Wang, P. Yin, H. Yu, Positivity-preserving third order DG schemes for poisson-nernst-planck equations, J. Comput. Phys. 452 (2022) 110777. https://doi.org/10.1016/j.jcp.2021.110777
[17] X. Zhang, On positivity-preserving high order discontinuous galerkin schemes for compressible navier-stokes equations, J. Comput. Phys. 328 (2017) 301–343. https://doi.org/10.1016/j.jcp.2016.10.002

[18] S. Srinivasan, J. Poggie, X. Zhang, A positivity-preserving high order discontinuous galerkin scheme for convection-diffusion equations, J. Comput. Phys. 366 (2018) 120–143. https://doi.org/10.1016/j.jcp.2018.04.002
[19] Z. Sun, J.A. Carrillo, C.W. Shu, A discontinuous galerkin method for nonlinear parabolic equations and gradient flow problems with interaction potentials, J. Comp. Phys. 352 (2018) 76–104.
[20] J. Ding, Z. Wang, S. Zhou, Positivity preserving finite difference methods for poisson-nernst-planck equations with steric interactions: application to slit-shaped nanopore conductance, J. Comput. Phys. 397 (2019) 108864.
[21] D. He, K. Pan, X. Yue, A positivity preserving and free energy dissipative difference scheme for the poisson-nernst-planck system, J. Sci. Comput. 81 (2019) 436–458.
[22] J. Ding, Z. Wang, S. Zhou, Structure-preserving and efficient numerical methods for ion transport, J. Comput. Phys. 418 (2020) 109597.
[23] J. Hu, X. Huang, A fully discrete positivity-preserving and energy-dissipative finite difference scheme for poisson-nernst-planck equations, Numer. Math. 145 (2020) 77–115.
[24] H. Liu, W. Maimaitiyiming, Unconditional positivity-preserving and energy stable schemes for a reduced poisson-nernst-planck system, Commun. Comput. Phys. 27 (2020) 1505–1529.
[25] H. Liu, W. Maimaitiyiming, Efficient, positive, and energy stable schemes for multi-d poisson-nernst-planck systems, J. Sci. Comput. 87 (2021) 92.
[26] S. Su, H. Tang, A positivity-preserving and free energy dissipative hybrid scheme for the poisson-nernst-planck equations on polygonal and polyhedral meshes, Comput. Math. Appl. 108 (2022) 33–48. https://doi.org/10.1016/j.camwa.2021.12.019
[27] C. Liu, C. Wang, S.M. Wise, X. Yue, S. Zhou, A positivity-preserving, energy stable and convergent numerical scheme for the poisson-Nernst-Planck system, Math. Comp. 90 (2021) 2071–2106.
[28] C. Liu, C. Wang, S.M. Wise, X. Yue, S. Zhou, A second order accurate, positivity preserving numerical method for the poisson-nernst-planck system and its convergence analysis, J. Sci. Comput. 97 (23) (2023).
[29] J. Shen, J. Xu, Unconditionally positivity preserving and energy dissipative schemes for poisson–Nernst–Planck equations, Numer. Math. 148 (2021) 671–697.
[30] M. Metti, J. Xu, C. Liu, Energetically stable discretizations for charge transport and electrokinetic models, J. Comput. Phys. 306 (2016) 1–18.
[31] J.A. Carrillo, A. Chertock, Y.H. Huang, A finite-volume method for nonlinear nonlocal equations with a gradient flow structure, Commun. Comput. Phys. 17 (1) (2015) 233–258.
[32] R. Jordan, D. Kinderlehrer, F. Otto, The variational formulation of the fokker–planck equation, SIAM J. Math. Anal. 29 (1) (1998) 1–17. https://doi.org/10.1137/S0036141096303359
[33] D. Kinderlehrer, L. Monsaingeon, X. Xu, A wasserstein gradient flow approach to poisson-nernst-planck equations, ESAIM Control Optim. Calc. Var. 23 (1) (2017) 137–164.
[34] H. Liu, W. Maimaitiyiming, A dynamic mass transport method for poisson-nernst-planck equations, J. Comput. Phys 473 (2023) 111699. https://doi.org/10.1016/j.jcp.2022.111699
[35] J.-D. Benamou, Y. Brenier, A computational fluid mechanics solution to the monge-kantorovich mass transfer problem, Numer. Math. 84 (3) (2000) 375–393.
[36] H. Chen, H. Liu, X. Xu, The onsager principle and structure preserving numerical schemes, J. Comput. Phys. 523 (2025) 113679.
[37] C. Liu, J. Hu, W.T. Taitano, X. Zhang, An optimization-based positivity-preserving limiter in semi-implicit discontinuous galerkin schemes solving fokker-planck equations, Comput. Math. Appl. 192 (2025) 54–71. https://doi.org/10.1016/j.camwa.2025.05.008
[38] K. Wu, C.-W. Shu, Geometric quasilinearization framework for analysis and design of bound-Preserving schemes, SIAM Rev. 65 (4) (2023) 1031–1073. https://doi.org/10.1137/21M1458247
[39] H. Liu, Analysis of direct discontinuous galerkin methods for multi-dimensional convection–diffusion equations, Numer. Math. 147 (4) (2021) 839–867.
[40] S. Gottlieb, C. Shu, E. Tadmor, Strong stability-preserving high-order time discretization methods, SIAM Rev. 43 (1) (2001) 89–112.
[41] P. Yin, Y. Huang, H. Liu, An iterative discontinuous galerkin method for solving the nonlinear poisson-boltzmann equation, Commun. Comput. Phys. 16 (2) (2014) 491–515.
[42] P. Yin, Y. Huang, H. Liu, Error estimates for the iterative discontinuous galerkin method to the nonlinear poisson-boltzmann equation, Commun. Comput. Phys. 23 (1) (2018) 168–197.