A decoupling two-grid method for the time-dependent Poisson-Nernst-Planck equations

Ruigang Shen - Shi Shu - Ying Yang - Benzhuo Lu

Received: 30 July 2018 / Accepted: 29 May 2019 / Published online: 7 June 2019
© Springer Science+Business Media, LLC, part of Springer Nature 2019

Abstract

We study a two-grid strategy for decoupling the time-dependent Poisson-Nernst-Planck equations describing the mass concentration of ions and the electrostatic potential. The computational system is decoupled to smaller systems by using coarse space solutions at each time level, which can speed up the solution process compared with the finite element method combined with the Gummel iteration. We derive the optimal error estimates in $L^2$ norm for both semi- and fully discrete finite element approximations. Based on the a priori error estimates, the error estimates in $H^1$ norm are presented for the two-grid algorithm. The theoretical results indicate this decoupling method can retain the same accuracy as the finite element method. Numerical experiments including the Poisson-Nernst-Planck equations for an ion channel show the efficiency and effectiveness of the decoupling two-grid method.

Keywords  Poisson-Nernst-Planck equations · Decoupling method · Two-grid method · Semi-discretization · Full discretization · Optimal error estimate · Gummel iteration

Mathematics Subject Classification (2010)  65N15 · 65N30

1 Introduction

In this paper, we consider the following time-dependent Poisson-Nernst-Planck (PNP) equations

$$
\begin{align*}
\frac{\partial p^i}{\partial t} - \nabla \cdot (\nabla p^i + q^i p^i \nabla \phi) &= F_i, \quad i = 1, 2, \\
-\Delta \phi - \sum_{i=1}^{2} q^i p^i &= F_3,
\end{align*}
$$

(1.1)
for $x \in \Omega$ and $t \in [0, T]$, where $\Omega$ is a bounded Lipschitz domain in $\mathbb{R}^d$ ($d = 2, 3$) and $\partial_t = \partial/\partial t$. The index $i$ represents different ionic species, $p^i$ is the concentration of the $i$th ionic species with charge $q^i$, $\phi$ is the electrostatic potential and $F_i$ ($i = 1, 2, 3$) are the reaction source terms. Denote the initial concentrations and potential by $(p_i^0, \phi^0)$, $i = 1, 2$. For simplicity, we employ the following homogeneous Dirichlet boundary conditions:

$$p^1 = p^2 = \phi = 0, \quad \text{on } \partial \Omega \times (0, T]. \quad (1.2)$$

The classic PNP system was first proposed by W. Nernst [7] and M. Planck [8]. It mainly describes the mass concentration of ions $p^i : \Omega \times (0, T] \to \mathbb{R}_0^+$ and the electrostatic potential $\phi : \Omega \times (0, T] \to \mathbb{R}$. As a continuum electrodiffusion model, PNP equations play an important role in the electrodiffusion reaction process. PNP equations couple the ion concentration distributions with the electrostatic potential which provide an ideal mean-field for describing this process [14, 15]. They have been widely used to study the ion channels and nanopores etc. [16, 17, 37].

Since the strong nonlinearity and coupling of the PNP system, in general, it is difficult to find the analytic solution of PNP equations. Therefore, there appears many numerical methods for solving PNP equations, including finite difference method, finite volume method, and finite element method. Finite difference method has been widely used to solve the PNP equations [21–23], but the accuracy is not so good when it is applied to the biomolecular models with highly irregular surfaces. Finite volume method, which focuses on avoiding the disadvantage of finite difference method, was then applied to solve the PNP equations in irregular domains, but it is not easy to achieve the high accuracy owing to the difficulty of the design of high-order control volume [24, 25]. Finite element method (FEM) has more flexibility and adaptability in irregular regions, which has shown the efficiency and effectiveness of dealing with PNP equations [9, 10, 15].

In contrast to amount of work on the numerical computations of PNP equations, the work of mathematical analysis of PNP equations seems limited, especially for finite element method. The existence and uniqueness of the finite element approximation for the time-dependent PNP equations are shown in [11]. Recently, Yang and Lu [12] presented an error analysis of the finite element method for a type of steady-state PNP equations modeling the electrodiffusion of ions in a solvated biomolecular system, in which the error estimates for the potential and concentration in $H^1$ norm depend on the $L^2$ error of the concentration. Sun et al. [5] analyzed a fully implicit nonlinear Crank-Nicolson scheme of the finite element method for the PNP equations, where an optimal $H^1$ norm error estimate is obtained for both the ion concentration and electrostatic potential. They also presented a $L^2$ norm error estimate which is only suboptimal for linear finite element approximations. And the results hold under the condition $k \geq d - 1$ ($k$ is the degree of polynomials) for the suboptimal $L^2$ error estimates, which implies that a second-order finite element has to be used for three-dimensional problem ($d = 3$) in [5]. Soon afterwards, Gao and He [13] obtained an optimal $L^2$ error estimate with linear finite element approximations for a linearized backward Euler scheme. It is shown that this linearized scheme can preserve mass conservation and energy decay. In this paper, we shall present an optimal $L^2$ error estimate for the classic backward Euler scheme. Compared with the
scheme in [13], the backward Euler scheme analyzed in this paper (see (2.6)–(2.7)) is fully implicit nonlinear. It is considered that this implicit nonlinear scheme could preserve most of the properties of the PNP equations and has been commonly used in the computation of the PNP system [15, 20, 37]. Compared with [5], we use a different projection operator (see (2.9)–(2.10)), and the optimal error estimates in $L^2$ norm are obtained for both semi- and fully discrete finite element approximations without any constraint condition except the regularity assumption which improves the a priori error estimates for PNP equations. These results shall be used in the error analysis of the main algorithm of the paper.

The PNP equations are a type of strong coupled system. Since the system consists of more than two partial differential equations, generally speaking, it is more convenient to solve it by using a decoupling method than solving it directly in application for large-scale problems. Decoupling methods, by which the coupled problems can be separated into single subproblems, have some appealing features. For example, the existed computing resources are more flexibly applied to solving each subproblem separately, and the numerical implementation is more easy and efficient. The main decoupling method used currently for solving PNP equations is the Gummel iteration [18–20]. For example, consider the following system coupled by two equations

\[
\begin{align*}
F(u_1, u_2) &= 0, \\
G(u_1, u_2) &= 0.
\end{align*}
\] (1.3)

The Gummel iteration for the above system could be: given $u^0_2$, for $k \geq 0$, find $(u^{k+1}_1, u^{k+1}_2)$ such that

\[
\begin{align*}
F(u^{k+1}_1, u^k_2) &= 0, \\
G(u^{k+1}_1, u^{k+1}_2) &= 0,
\end{align*}
\] (1.4)

until the error between the $(k+1)$th solution and $k$th solution is less than the tolerance. However, it converges slowly even diverges if the discretized system of the PNP equations is a large scale problem.

We note that two-grid method is also one of the decoupling methods which have been applied successfully to some coupled systems such as the Schrödinger equation arising from quantum mechanics [26] and the mixed Stokes-Darcy model for coupling fluid flow with porous media flow [27, 28]. The two-grid method, proposed originally by Xu [29] in 1992, was designed for dealing with nonselfadjoint or indefinite problems and has a variety of application to solving many problems, such as the nonlinear parabolic equation [30], the nonlinear reaction-diffusion equation [31, 32], the time-dependent Navier-Stokes equations [33, 34], and the nonlinear coupled miscible displacement problems [35, 36]. In [33], He proposed the two-level method based on finite element and Crank-Nicolson extrapolation for solving the two-dimensional time-dependent Navier-Stokes equations, in which the optimal error estimates of the discrete solution are obtained for the two-level method by using the Crank-Nicolson extrapolation solution on a spatial-time coarse grid and a backward Euler solution on a space-time fine grid. In [34], a multilevel finite element method
in space-time for the nonstationary Navier-Stokes problem is provided by He and Liu. In contrast to the two-level method proposed in [33], the method is a multi-scale method in which the fully nonlinear Navier-Stokes problem is only solved on a single coarsest space-time mesh. Theoretical analysis shows that the multilevel method provides the same accuracy as the one-level method in space-time. As a decoupling method for the coupled equations, the procedure of the two-grid method may be different from that for a single partial differential equation mentioned above, but it has the similar idea that a coarse space solution is chosen as a reliable approximation to the fine space solution. In [35], the concentration equation is approximated by the Eulerian-Lagrangian localized adjoint method in [36]. The two-grid algorithms based on the Newton iteration are designed to linearize and decouple the mixed-method equations in both [35] and [36] for the discrete nonlinear coupled miscible displacement problems. The theoretical results show that the asymptotically optimal approximation can be obtained for the two-grid methods with mixed finite elements when the mesh size satisfies $H = O(h^{\frac{1}{2}})$. In the two-grid algorithms designed in this paper for decoupling the time-dependent PNP equations, since we can use an appropriate coarse space solution as a reliable approximation to the fine space solution, the iteration between the equations solving individually can be avoided on the fine space, while it may require lots of iterations for the Gummel method (1.4) if an inappropriate initial value is used. Moreover, since the two-grid method is based on the finite element method, the numerical implement of the decoupling process is easy if the finite element method is used to solve PNP equations. These are the main reasons that we consider the two-grid method to deal with PNP equations among many decoupling methods.

In this paper, we propose and analyze the two-grid algorithm for time-dependent PNP equations in a fully discrete scheme. Since PNP equations are different from the coupled models mentioned above, the design and analysis of the two-grid method cannot directly follow the existed work. The error estimates in $H^1$ norm are obtained for both the concentration and potential. The theoretical results show that if the mesh sizes $H$ and $h$ satisfy some requirement (for example $H = O(h^{\frac{1}{2}})$ with linear finite elements), then the two-grid method can retain the same accuracy as the conventional finite element method. In addition, some numerical examples including an ion channel problem are shown to verify the theoretic results. The CPU time cost shows the validity and efficiency of the two-grid method for PNP equations.

The rest of this paper is organized as follows. In Section 2, we introduce some notations and the weak formulations of the PNP system. The projection operators and some useful estimates are also given in this section. In Section 3, we show the optimal $L^2$ error estimates of the standard finite element method for both semi- and fully discrete schemes. The two-grid method and some error analysis are presented in Section 4. Numerical experiments are reported in Section 5 to show the effectiveness of the proposed method. An application of the two-grid method in ion channel problem is shown in Section 6. The conclusion is presented in Section 7.
2 Weak formulation and projection operators

In this section, we shall present the variational forms of PNP system (1.1)–(1.2) and some projection estimates which shall be used in our analysis.

First, we clarify the standard notations for Sobolev spaces $W^{s,p}(\Omega)$ and their associated norms and seminorms (see, e.g., [1, 3]). For $p = 2$, we denote $W^{s,2}(\Omega) = H^s(\Omega)$, $H^1_0(\Omega) = \{ v| v \in H^1(\Omega) : v|_{\partial\Omega} = 0 \}$, $\| \cdot \|_{s,p,\Omega} = \| \cdot \|_{W^{s,p}(\Omega)}$ with the expression that $\| \cdot \|$ and $(\cdot, \cdot)$ denote the norm and inner product in $L^2$, and $\| \cdot \|_{0,\infty} = \| \cdot \|_{L^\infty}$.

Let $T_h = \{ e \}$ be a quasi-uniform partition of $\Omega$, where $e$ is the element, and the mesh size $h = \max_{e \in T_h} \{ \text{diam } e \}$. Then, for a given partition $T_h$, we define $V_r^h$ as the $r$th-order finite element subspace of $H^1_0(\Omega)$ as follows:

$$V_r^h = \{ v \in H^1(\Omega) : v|_{\partial\Omega} = 0 \text{ and } v|_e \in P_r(e), \forall e \in T_h \}, \quad (2.1)$$

where $P_r(e)$ is the space of polynomial with degree $r$.

The weak formulations of (1.1)–(1.2) are that: find $p_i^t \in L^2(0, T; H^1_0(\Omega)) \cap L^\infty(0, T; L^\infty(\Omega))$, $i = 1, 2$, and $\phi(t) \in H^1_0(\Omega)$ such that

$$(\partial_t p_i^t, v) + (\nabla p_i^t, \nabla v) + (q_i^t p_i^t \nabla \phi, \nabla v) = (F_i, v), \quad \forall v \in H^1_0(\Omega), \quad (2.2)$$

$$(\nabla \phi, \nabla w) - \sum_{i=1}^2 q_i^t (p_i^t, w) = (F_3, w), \quad \forall w \in H^1_0(\Omega). \quad (2.3)$$

The corresponding semi-discretization to (2.2)–(2.3) is defined as follows: find $(p_i^h, \phi_h) \in [V_r^h]^3$, $i = 1, 2$, where $[V_r^h]^3 = V_r^h \times V_r^h \times V_r^h$, such that

$$(\partial_t p_i^h, v_h) + (\nabla p_i^h, \nabla v_h) + (q_i^h p_i^h \nabla \phi_h, \nabla v_h) = (F_i, v_h), \quad \forall v_h \in V_r^h, \quad (2.4)$$

$$(\nabla \phi_h, \nabla w_h) + \sum_{i=1}^2 q_i^h (p_i^h, w_h) = (F_3, w_h), \quad \forall w_h \in V_r^h. \quad (2.5)$$

with the initial condition $(p_i^{t^0}_h, \phi_h^{t^0})$ is an approximation of $(p_i^{t^0}, \phi^{t^0})$ and the Dirichlet boundary condition $p_i^h = \phi_h = 0$ on $\partial\Omega$.

In order to get the full discretization of the system (2.2)–(2.3), we first define a uniform partition $0 = t^0 < t^1 < \cdots < t^N = T$ with time step size $\tau = \frac{T}{N}$ and $t^n = n\tau$, $n \in \mathbb{Z}$. For any function $u$, denote by

$$u^n = u(x, t^n),$$

and

$$D_\tau u^{n+1} = \frac{u^{n+1} - u^n}{\tau}, \text{ for } n = 0, 1, 2, \ldots, N - 1.$$
Then, the backward Euler full discretization scheme of the system (2.2)–(2.3) is:
given $P_{i,n}^{h} \in [V_r^h]^2$, $i = 1, 2$, where $[V_r^h]^2 = V_r^h \times V_r^h$, find $(P_{i,n+1}^{h}, \Phi_{i,n+1}^{h}) \in [V_r^h]^3$, such that

$$
(D_{\tau} P_{i,n+1}^{h}, v_h) + (\nabla P_{i,n+1}^{h}, \nabla v_h) + (q^i P_{i,n+1}^{h} \nabla \Phi_{i,n+1}^{h}, \nabla v_h) = (F_{i,n+1}^{h}, v_h), \forall v_h \in V_r^h, (2.6)
$$

$$
(\nabla \Phi_{i,n+1}^{h}, \nabla w_h) - \sum_{i=1}^{2} q^i (P_{i,n+1}^{h}, w_h) = (F_{i,n+1}^{h}, w_h), \forall w_h \in V_r^h. (2.7)
$$

The well-posedness and stability of the solutions to the the schemes (2.6)–(2.7) have been presented in [11]. In the rest part of this paper, we assume that the exact solution of the PNP (1.1) exists and satisfies the following regularity assumptions

$$
\begin{align*}
\| \pi \|_{L^{\infty}(0,T;H^{r+1} \cap W^{1,\infty}())} + \| p_i \|_{L^{\infty}(0,T;H^{r+1} \cap W^{1,\infty}())} + \| p_i \|_{L^{\infty}(0,T;H^{r+1} \cap W^{1,\infty}())} \leq C, \\
\| \phi \|_{L^{\infty}(0,T;W^{r+1,\infty}())} + \| \phi_i \|_{L^{\infty}(0,T;W^{r+1,\infty}())} + \| \phi_i \|_{L^{\infty}(0,T;W^{r+1,\infty}())} \leq C.
\end{align*}
$$

(2.8)

To present the error estimates in this paper, for given $t \in [0,T]$, we define $R_h : H_0^1(\Omega) \to V_r^h$ to be a Ritz projection operator by

$$
\begin{align*}
(\nabla (R_h p^i - p^i), \nabla v_h) + (q^i p^i \nabla (R_h \phi - \phi), \nabla v_h) = 0, \forall v_h \in V_r^h, \\
(\nabla (R_h \phi - \phi), \nabla w_h) - \sum_{i=1}^{2} q^i (R_h p^i - p^i), w_h) = 0, \forall w_h \in V_r^h.
\end{align*}
$$

(2.9)

(2.10)

Particularly, the similar definition of the projection operator $R_h$ can be found in [13].

At the initial step in (2.6)–(2.7), we take the initial value $p_{i,0}^h = R_h p_{i,0}^i$.

We define the projection error by

$$
\theta_{p^i} = R_h p^i - p^i, \quad \theta_{\phi} = R_h \phi - \phi.
$$

Then, by standard finite element theory and the regularity assumption (2.8), we have

$$
\begin{align*}
\| \theta_{p^i} \| + h \| \theta_{p^i} \|_1 & \leq C h^{r+1}, \\
\| \theta_{\phi} \| + h \| \theta_{\phi} \|_1 & \leq C h^{r+1}, \\
\| \partial_t \theta_{p^i} \| + h \| \partial_t \theta_{p^i} \|_1 & \leq C h^{r+1}.
\end{align*}
$$

(2.11)

(2.12)

(2.13)

Finally, we introduce two lemmas which will be used in the error analysis.

**Lemma 2.1 (Gagliardo–Nirenberg inequality [39])** Let $u$ be a function defined on a bounded domain $\Omega \subset \mathbb{R}^d$ and its derivatives of order $m$ belongs to $L^{q^*}$ in $\Omega$. Then, for the derivatives $\partial^j u$, $0 \leq j < m$, the following inequalities hold (where constant $C$ depends only on $\Omega$, $m$, $j$, $q$, $q^*$):

$$
\| \partial^j u \|_{L^p} \leq C \left( \| \partial^m u \|_{L^{q^*}}^{\alpha} \| u \|_{L^q}^{1-\alpha} + \| u \|_{L^q} \right),
$$

$\square$ Springer
\[ \frac{1}{p} = \frac{j}{d} + a \left( \frac{1}{q^*} - \frac{m}{d} \right) + (1-a) \frac{1}{q}, \]

except \( 1 < q^* < \infty \) and \( m - j - \frac{d}{q^*} \) is a non-negative integer, in which case the above estimate holds only for \( \frac{j}{m} \leq a < 1 \).

**Lemma 2.2** [40] Suppose that \( \Omega \) is a smooth bounded domain and \( u \in H^2(\Omega) \) is a solution of

\[
\begin{aligned}
-\Delta u &= f, \quad x \in \Omega, \\
 u &= 0, \quad x \in \partial \Omega.
\end{aligned}
\]

Then, the following estimate holds for \( 1 < p < \infty \):

\[ \|u\|_{W^{2,p}} \leq C \|f\|_{L^p}. \]

### 3 L2 norm error analysis for finite element approximation

In this section, we give the a priori error estimates for both the semi-discretization finite element solution \((p_i^h, \phi_h)\) of (2.4)–(2.5) and the fully discrete finite element solution \((P_i^n, \Phi^n_h)\) of (2.6)–(2.7). For the sake of analysis, we assume the source term \( F_3 \in L^4(\Omega) \) and the size of the grid \( h \ll 1 \).

#### 3.1 Error analysis for the semi-discretization

We give the a priori error estimate for the semi-discretization finite element approximation \((p_i^h, \phi_h)\) as follows:

**Theorem 3.1** Let \((p^i, \phi)\) and \((p_i^h, \phi_h)\) be the solutions of (2.2)–(2.3) and (2.4)–(2.5), respectively. Then, for \( t \in [0, T] \), we have the following estimate:

\[ \|p^i - p_i^h\| + \|\phi - \phi_h\| \leq C h^{r+1}. \quad (3.1) \]

**Proof** From the projection error estimates (2.11)–(2.13), we only need to estimate the following error functions:

\[ e_{p^i} = p_i^h - R_h p^i, \quad e_{\phi} = \phi_h - R_h \phi. \]

It follows from (2.2)–(2.3) and (2.4)–(2.5) that, \( \forall v_h, w_h \in V_h^r \)

\[ \left( \partial_t (p_i^h - p^i), v_h \right) + \left( \nabla (p_i^h - p^i), \nabla v_h \right) + q^i \left( p_i^h \nabla \phi_h - p^i \nabla \phi, \nabla v_h \right) = 0, \quad (3.2) \]

\[ \left( \nabla (\phi_h - \phi), \nabla w_h \right) - \left( \sum_{i=1}^2 q^i (p_i^h - p^i), w_h \right) = 0. \quad (3.3) \]
Taking \( w_h = e_\phi \) in (3.3) and using (2.10), we have
\[
(\nabla e_\phi, \nabla e_\phi) = \left( \sum_{i=1}^{2} q^i e_{p_i^1}, e_\phi \right), \tag{3.4}
\]
which easily yields
\[
\|\nabla e_\phi\| \leq C \sum_{i=1}^{2} \|e_{p_i^1}\|. \tag{3.5}
\]
Taking \( v_h = e_{p_i^1} \) in (3.2) and using (2.9), we have
\[
(\partial_t e_{p_i^1}, e_{p_i^1}) + (\nabla e_{p_i^1}, \nabla e_{p_i^1}) = \sum_{j=1}^{3} I_j, \tag{3.6}
\]
where \( I_j, j = 1, 2, 3 \), are defined as
\[
I_1 := - (\partial_t \theta_{p_i^1}, e_{p_i^1}), \quad I_2 := - q^i \left( p^i \nabla e_\phi, \nabla e_{p_i^1} \right), \quad I_3 := - q^i \left( (e_{p_i^1} + \theta_{p_i^1}) \nabla \phi_h, \nabla e_{p_i^1} \right).
\]
In the following, we shall estimate \( I_1, I_2, \) and \( I_3 \), respectively. By the projection estimate (2.13), there holds
\[
I_1 \leq \|\partial_t \theta_{p_i^1}\| \|e_{p_i^1}\| \leq C h^{r+1} \|e_{p_i^1}\| \leq C (h^{2r+2} + \|e_{p_i^1}\|^2). \tag{3.7}
\]
Using (3.5) and the regularity assumption (2.8), we have
\[
I_2 \leq C \|p^i\|_{0, \infty} \|\nabla e_\phi\| \|\nabla e_{p_i^1}\| \leq C \sum_{i=1}^{2} \|e_{p_i^1}\| \|\nabla e_{p_i^1}\|
\leq C \sum_{i=1}^{2} \left( \|e_{p_i^1}\|^2 + \epsilon \|\nabla e_{p_i^1}\|^2 \right), \tag{3.8}
\]
where \( 0 < \epsilon < 1 \) is a constant. To estimate \( I_3 \), we shall prove the following result
\[
\|\nabla \phi_h\|_{0, \infty} \leq C \left( \sum_{i=1}^{2} \|\nabla e_{p_i^1}\| + C_1 \right), \tag{3.9}
\]
where \( C_1 \) is a positive constant satisfying \( \sum_{i=1}^{2} \|p^i\|_{1,2} + \|F_3\|_{0,4} \leq C_1 \).

It is easy to see that \( \phi_h \) can be viewed as the finite element approximation to the solution of the Poisson equation
\[
- \Delta \phi = \sum_{i=1}^{2} q^i p^i_{h} + F_3. \tag{3.10}
\]
with homogeneous Dirichlet boundary condition. Hence, by $W^{1,p}$-estimate of the finite element methods [3, 4], Lemma 2.1 and Lemma 2.2, we have

$$\| \nabla \phi_h \|_{0, \infty} \leq C \| \phi \|_{1, \infty} \leq C \| \phi \|_{2,4} \leq C \sum_{i=1}^{2} q^i p^i_h + F_3 \| \nabla \phi_h \|_{0,4}$$

$$\leq C \left( \sum_{i=1}^{2} \left( \| p^i_h - R_h p^i \|_{0,4} + \sum_{i=1}^{2} \| R_h p^i \|_{0,4} + \| F_3 \|_{0,4} \right) \right)$$

$$\leq C \left( \sum_{i=1}^{2} \| e_{pi} \|_{1,2} + \sum_{i=1}^{2} \| p^i \|_{1,2} + \| F_3 \|_{0,4} \right)$$

$$\leq C \left( \sum_{i=1}^{2} \| \nabla e_{pi} \| + C_1 \right), \quad (3.11)$$

which yields estimate (3.9).

Then by (3.9) and the projection error estimate (2.11), it yields

$$I_3 \leq C \| e_{pi} \| + \| p^i \| \| \nabla \phi_h \|_{0, \infty} \| \nabla e_{pi} \|

\leq C(\| e_{pi} \| + h^{r+1}) \left( \sum_{i=1}^{2} \| \nabla e_{pi} \| + C_1 \right) \| \nabla e_{pi} \|

\leq C \left( \sum_{i=1}^{2} \| e_{pi} \| \| \nabla e_{pi} \|^2 + \| e_{pi} \| \| \nabla e_{pi} \| + h^{r+1} \sum_{i=1}^{2} \| \nabla e_{pi} \|^2 + h^{r+1} \| \nabla e_{pi} \| \right)$$

$$\leq C \left( \sum_{i=1}^{2} \| e_{pi} \| \| \nabla e_{pi} \|^2 + \| e_{pi} \|^2 + h^{2r+2} \right) + \epsilon \| \nabla e_{pi} \|^2, \quad (3.12)$$

where we have used $Ch^{r+1} \leq \epsilon$ when $h \ll 1$. Substituting estimates (3.7)–(3.8) and (3.12) into (3.6), we get

$$\frac{1}{2} \partial_t \| e_{pi} \|^2 + \| \nabla e_{pi} \|^2 \leq C \left( \sum_{i=1}^{2} \| e_{pi} \| \| \nabla e_{pi} \|^2 + \| e_{pi} \|^2 + h^{2r+2} \right) + \epsilon \| \nabla e_{pi} \|^2. \quad (3.13)$$

Now, we conduct a mathematic induction process to prove the following inequality:

$$\| e_{pi} \| \leq C h^{r+1}, \forall t \in [0, T]. \quad (3.14)$$

Assume (3.14) holds for any $t \in [0, T^*], \ T^* < T$. Then by (3.13), we get

$$\frac{1}{2} \partial_t \| e_{pi} \|^2 + \| \nabla e_{pi} \|^2 \leq C(\| e_{pi} \|^2 + \| e_{pi} \|^2 + \| e_{pi} \|^2 + h^{2r+2} + \epsilon \| \nabla e_{pi} \|^2). \quad (3.15)$$

Take integral with respect to $t$,

$$\| e_{pi} \|^2 + \int_0^t \| \nabla e_{pi}(s) \|^2 ds \leq C \left( h^{2r+2} + \int_0^t \| e_{pi}(s) \|^2 ds \right).$$
where we have used the fact \( \| e_{p'}(0) \| = 0 \) by the initial condition \( R_h p^i(0) = p^i_h(0) \). By using Gronwall’s inequality, we have for \( 0 \leq t \leq T^* \),
\[
\| e_{p'} \| \leq C h^{r+1}. 
\] (3.16)
This implies that
\[
h^{-(r+1)} \| e_{p'} \| \leq C.
\]
On the other hand, since \( h^{-(r+1)} \| e_{p'} \| \) is a continuous function with respect to \( t \in [0, T] \), and due to the uniform continuity with time, then for any \( \epsilon > 0 \), there exists \( \delta \) such that for any \( t \in [T^*, T^* + \delta] \),
\[
\| e_{p'}(t) \| - \| e_{p'}(T^*) \| \leq \epsilon h^{r+1}.
\]
Then from (3.16), we get
\[
\| e_{p'}(t) \| \leq \| e_{p'}(t) \| - \| e_{p'}(T^*) \| + \| e_{p'}(T^*) \| \leq \epsilon h^{r+1} + C h^{r+1} \leq C_0 h^{r+1},
\]
where \( C_0 \) is a positive constant independent of \( h \). Because \([0, T]\) is a finite interval, so the induction hypothesis (3.14) holds for all \( t \in [0, T] \).
Therefore, for any \( t \in [0, T] \), by projection estimate (2.11) and (3.14), we can easily get
\[
\| p^i - p^i_h \| \leq \| e_{p'} \| + \| \theta_{p'} \| \leq C h^{r+1}.
\] (3.17)
Then, (3.1) is proved by combining (3.5), the projection estimate (2.12) and (3.17). This completes the proof of Theorem 3.1.

Now, we turn to the full discretization scheme.

### 3.2 Error analysis for the full discretization

In this subsection, we present the error estimate of the full discretization schemes (2.6)–(2.7). For given \( t = t^n \), define the error functions
\[
e_{p'}^n = P^{i,n}_h - R_h P^{i,n}_n, \quad e_{\Phi}^n = \Phi^n_h - R_h \Phi^n_n, \quad \text{for } n = 0, 1, 2, \cdots, N.
\] (3.18)

**Theorem 3.2** Let \( (p^{i,n}, \phi^n) \) and \( (P^{i,n}_h, \Phi^n_h) \) be the solutions of (2.2)–(2.3) and (2.6)–(2.7), respectively. Then, there exists two positive constants \( \tau_0 \) and \( h_0 \) such that for any \( n = 0, 1, \cdots, N \),
\[
\max_{0 \leq n \leq N} (\| P^{i,n}_h - P^{i,n} \| + \| \Phi^n_h - \phi^n \|) \leq C(\tau + h^{r+1}),
\] (3.19)
provided by \( \tau < \tau_0 \) and \( h \leq h_0 \).
Proof By the weak formulation (2.2)–(2.3) and the Ritz projection (2.9)–(2.10), \( \forall v_h, w_h \in V_h^r \), we have

\[
(D_\tau p^{i,n+1}, v_h) + (\nabla R_h p^{i,n+1}, \nabla v_h) + q^i (p^{i,n+1} \nabla R_h \phi^{n+1}, \nabla v_h) = (D_\tau p^{i,n+1}, v_h) - (\partial_t p^i |_{\tau^{n+1}}, v_h) + (F^{n+1}_i, v_h), \tag{3.20}
\]

\[
(\nabla R_h \phi^{n+1}, \nabla w_h) = \left( \sum_{i=1}^2 q^i R_h p^{i,n+1}, w_h \right) + (F^{n+1}_3, w_h). \tag{3.21}
\]

Then from (3.20)–(3.21) and the full discretization schemes (2.6)–(2.7), we have

\[
(D_\tau (P_i^{n+1} - p^{i,n+1}) + P_i^{n+1} - R_h p^{i,n+1}, v_h) + (\nabla (P_i^{n+1} - R_h p^{i,n+1}), \nabla v_h) = -q^i (P_i^{n+1} \nabla \Phi^{n+1}_h - p^{i,n+1} \nabla R_h \phi^{n+1}, \nabla v_h) - (D_\tau p^{i,n+1}, v_h) + (\partial_t p^i |_{\tau^{n+1}}, v_h), \tag{3.22}
\]

\[
(\nabla (\Phi^{n+1}_h - R_h \phi^{n+1}), \nabla w_h) = \sum_{i=1}^2 q^i \left( P_i^{n+1} - R_h p^{i,n+1}, w_h \right). \tag{3.23}
\]

Choosing \( v_h = e^{n+1}_p \) in (3.22) and \( w_h = e^{n+1}_\phi \) in (3.23), respectively, we get

\[
\left( D_\tau e^{n+1}_p, e^{n+1}_p \right) + \left( \nabla e^{n+1}_p, \nabla e^{n+1}_p \right) = H_1^n + H_2^n + H_3^n, \tag{3.24}
\]

\[
(\nabla e^{n+1}_\phi, \nabla e^{n+1}_\phi) = \left( \sum_{i=1}^2 q^i e^{n+1}_p, e^{n+1}_\phi \right), \tag{3.25}
\]

where

\[
H_1^n := -(D_\tau \theta^{n+1}_p + e^{n+1}_p),
\]

\[
H_2^n := -q^i (P_i^{n+1} \nabla \Phi^{n+1}_h - p^{i,n+1} \nabla R_h \phi^{n+1}, \nabla e^{n+1}_p),
\]

\[
H_3^n := -(D_\tau p^{i,n+1} - \partial_t p^i |_{\tau^{n+1}}, e^{n+1}_p).
\]

By (3.25), we can easily get

\[
\| \nabla e^{n+1}_\phi \| \leq C \sum_{i=1}^2 \| e^{n+1}_p \|. \tag{3.26}
\]

Now, we focus on deriving the estimates of \( H_1^n \), \( H_2^n \), and \( H_3^n \).

First, by the projection estimate (2.13), we have

\[
H_1^n \leq \frac{1}{\tau} \| \theta^{n+1}_p - \theta^n_p \| \| e^{n+1}_p \| \leq \frac{1}{\tau} \left( \int_{\tau^n}^{\tau^{n+1}} \| \partial_s (\theta^p_s) (s) \| ds \right) \| e^{n+1}_p \| \leq C h^{r+1} \| e^{n+1}_p \| \leq C (h^{2r+2} + \| e^{n+1}_p \|^2),
\]

(3.27)
Choosing a sufficiently small $\epsilon$ Combining (3.24), (3.27), (3.28), and (3.31), we have
\[
H_{3}^{n} \leq \| D_{t} p^{i,n+1} - \partial_{t} p^{i} |_{n+1} \| e^{n+1}_{p'} \| \leq \frac{1}{2} \tau \cdot \partial_{tt} p^{i} (x, \xi) \| e^{n+1}_{p'} \|
\]
\[
\leq C \tau \| e^{n+1}_{p'} \| \leq C (\tau^{2} + \| e^{n+1}_{p'} \|^{2}), \quad (t^{n} < \xi < t^{n+1}). \quad (3.28)
\]
For $H_{2}^{n}$, there holds
\[
H_{2}^{n} = -q i \left( (p_{h}^{i,n+1} - p^{i,n+1}) \nabla \Phi_{n+1}^{i} + p^{i,n+1} \nabla (\Phi^{n+1}_{h} - R_{h} \Phi^{n+1}_{1}), \nabla e^{n+1}_{p'} \right)
\]
\[
= -q i \left( (e^{n+1}_{p'} + \theta^{n+1}_{p'}) \nabla \Phi_{n+1}^{i}, \nabla e^{n+1}_{p'} \right) - q i \left( p^{i,n+1} \nabla e^{n+1}_{\phi}, \nabla e^{n+1}_{p'} \right)
\]
\[
\leq C \left( \| e^{n+1}_{p'} \| + h^{r+1} \| \nabla \Phi_{n+1}^{i} \|_{0,\infty} \| \nabla e^{n+1}_{p'} \| + 2 \sum_{i=1}^{2} \| e^{n+1}_{p'} \| \| \nabla e^{n+1}_{p'} \| \right),
\]
where we have used (2.8), (2.11), and (3.26).

On the other hand, by inverse inequality and (3.26),
\[
\| \nabla \Phi_{n+1}^{i} \|_{0,\infty} \leq \| \nabla R_{h} \Phi_{n+1}^{i} \|_{0,\infty} + \| \nabla (\Phi_{n+1}^{i} - R_{h} \Phi_{n+1}^{i}) \|_{0,\infty}
\]
\[
\leq \| \nabla \phi^{n+1} \|_{0,\infty} + \| \nabla (R_{h} \phi^{n+1} - \phi^{n+1}) \|_{0,\infty} + \| \nabla (\Phi_{n+1}^{i} - R_{h} \Phi_{n+1}^{i}) \|_{0,\infty}
\]
\[
\leq \| \phi \|_{L_{\infty}(W^{1,\infty}(Q_{2}))} + C \left( h^{r} \| \phi^{n+1} \|_{r+1,\infty} + h^{-\frac{d}{2}} \| \nabla (\Phi_{n+1}^{i} - R_{h} \Phi^{n+1}_{1}) \| \right)
\]
\[
\leq C (1 + h^{-\frac{d}{2}} \| \nabla e^{n+1}_{\phi} \|) \leq C (1 + h^{-\frac{d}{2}} \sum_{i=1}^{2} \| e^{n+1}_{p'} \|). \quad (3.29)
\]
In what follows, we shall prove by mathematical induction that the following inequality holds for $n = 0, 1, \ldots, N - 1$:
\[
\| e^{n+1}_{p'} \| \leq C (\tau + h^{r+1}). \quad (3.30)
\]
Assume (3.30) holds for any $n = 0, 1, \ldots, J$, $0 \leq J \leq N - 2$. Then by (3.29), we get $\| \nabla \Phi_{n+1}^{i} \|_{0,\infty} \leq C$. Hence,
\[
H_{2}^{n} \leq C \left( \| e^{n+1}_{p'} \|^{2} + h^{2r+2} + \| \nabla e^{n+1}_{p'} \|^{2} \right) + \| \nabla e^{n+1}_{p'} \|^{2}. \quad (3.31)
\]
Combining (3.24), (3.27), (3.28), and (3.31), we have
\[
\frac{1}{2} D_{t} \| e^{n+1}_{p'} \|^{2} + \| \nabla e^{n+1}_{p'} \|^{2} \leq C (\tau^{2} + h^{2r+2} + \| e^{n+1}_{p'} \|^{2} + \| \nabla e^{n+1}_{p'} \|^{2}). \quad (3.32)
\]
Choosing a sufficiently small $\epsilon$ and summing up for the index $n = 0, 1, \ldots, J$, $0 \leq J \leq N - 1$ on both sides of (3.32), then we can easily get the following inequality
\[
\| e^{J+1}_{p'} \|^{2} + \tau \sum_{n=0}^{J} \| \nabla e^{n+1}_{p'} \|^{2} \leq C (\tau^{2} + h^{2r+2}) + C \tau \sum_{n=0}^{J} \| e^{n+1}_{p'} \|^{2}.
\]
By the discrete Gronwall’s inequality, we get

\[ \|e_{p_i}^{J+1}\|^2 + \tau \sum_{n=0}^{J} \|\nabla e_{p_i}^{n+1}\|^2 \leq C(\tau^2 + h^{2r+2}). \quad (3.33) \]

This implies that

\[ \|e_{p_i}^{J+1}\| \leq C(\tau + h^{r+1}), \quad 0 \leq J \leq N - 1. \]

Thus, (3.30) holds for \( n = 0, 1, \cdots, N - 1 \). We complete the induction.

Finally, by projection error estimate (2.11) and (3.30), it yields

\[ \max_{0 \leq n \leq N} \|P_{h}^{i,n} - p^{i,n}\| \leq \max_{0 \leq n \leq N} \left( \|\theta_{p}^{n}\| + \|e_{p_i}^{n}\| \right) \leq C(\tau + h^{r+1}). \quad (3.34) \]

Theorem 3.2 is proved by combining (3.34), the projection error estimate (2.12) and (3.26).

**Remark 3.1** Theorem 3.2 shows that if we choose the time step \( \tau \) and mesh size \( h \) satisfy \( \tau = O(h^{r+1}) \), then the optimal \( L^2 \) norm error estimate is obtained when \( r \)-order finite element is used for both the concentration and electrostatic potential. In fact, choosing \( v_h = D_{\tau} e_{p_i}^{n+1} \) in (3.22) instead of \( e_{p_i} \) and following an analogous arguments in \( H^1 \) norm for the concentrations and electrostatic potential in [5], we can prove the error estimate in \( H^1 \) norm, i.e.,

\[ \|P_{h}^{i,n} - p^{i,n}\|_1 + \|\Phi_{h}^{n} - \phi^{n}\|_1 \leq C(\tau + h^r). \quad (3.35) \]

Next, a two-grid finite element method for PNP equation (1.1) will be presented in full discretization schemes. Some error estimates are derived which show our method can achieve the same error accuracy as the standard finite element method. However, the CPU time cost of the two-grid method is much less than that of standard finite element method, which is shown in Section 5.

### 4 The two-grid algorithm and error analysis

In this section, we shall present the main algorithms of the paper. Two quasi-uniform triangulations \( T_H \) and \( T_h \) of \( \Omega \) with two different mesh sizes \( H \) and \( h \) (\( H > h \)) are introduced. The corresponding finite element spaces \( V_H^r \) and \( V_h^r \), which satisfy \( V_H^r \subset V_h^r \), are called the coarse-grid and fine-grid spaces, respectively. Two algorithms are provided to decouple the strong coupled equations and some error estimates are also derived.

First, a semi-decoupled scheme is presented as follows:
We need the following error estimate in the later analysis.

**Lemma 4.1** Let \((p_{i,n}^{i+1}, \phi_{i,n}^{i+1})\) and \((p_{r*,i,n}^{i+1}, \Phi_{r*,i,n}^{i+1})\) be the solutions of (2.2)–(2.3) and (4.3)–(4.4), respectively. Then for any \(n = 0, 1, 2, \ldots, N - 1\), we have

\[
\|\rho_{i,n}^{i+1} - \rho_{i,n}^{i+1}\| \leq C(\tau + H^{r+1}).
\] (4.5)

**Proof** Denote

\[
\rho_{i,n}^{i+1} = p_{i,n}^{i+1} - p_{i,n}^{i}, \quad \Phi_{i,n}^{i+1} = \phi_{i,n}^{i+1} - \phi_{i,n}^{i},
\] (4.6)

where

\[
\rho_{p_{i}}^{i+1} = p_{i,n}^{i+1} - R_{h} p_{i,n}^{i+1}, \quad \theta_{p_{i}}^{i+1} = R_{h} p_{i,n}^{i+1} - p_{i,n}^{i+1},
\]

\[
\rho_{\phi_{i}}^{i+1} = \phi_{i,n}^{i+1} - R_{h} \phi_{i,n}^{i+1}, \quad \theta_{\phi_{i}}^{i+1} = R_{h} \phi_{i,n}^{i+1} - \phi_{i,n}^{i+1}.
\]

Following a similar proof of Theorem 3.2, subtracting (3.20) from (4.3), and taking \(v_{h} = \rho_{p_{i}}^{i+1}\), we have the error equation

\[
(D_{\tau} \rho_{p_{i}}^{n+1}, \rho_{p_{i}}^{n+1}) + (\nabla \rho_{p_{i}}^{n+1}, \nabla \rho_{p_{i}}^{n+1}) = T_{1}^{n} + T_{2}^{n} + T_{3}^{n},
\] (4.7)
where
\[
T_1^n = - \left( D_\tau \theta_{p_i}^{n+1}, \rho_{p_i}^{n+1} \right), \\
T_2^n = - (D_\tau p_{i,n}^{i,n+1} - \partial_t p_{i,n}^{i,n+1}, \rho_{p_i}^{n+1}), \\
T_3^n = - q_i \left( (\rho_{p_i}^{n+1} + \theta_{p_i}^{n+1}) \nabla \Phi_{h*}^{n+1}, \nabla \rho_{p_i}^{n+1} \right) - q_i \left( p_{i,n}^{i,n+1} \nabla \rho_{p_i}^{n+1}, \nabla \rho_{p_i}^{n+1} \right).
\]

We shall estimate \(T_1^n\), \(T_2^n\), and \(T_3^n\), respectively, below.

By the similar arguments as in (3.27)–(3.28), we get
\[
T_1^n \leq C (h^{2r+2} + \| \rho_{p_i}^{n+1} \|^2), \\
T_2^n \leq C (\tau^2 + \| \rho_{p_i}^{n+1} \|^2). 
\tag{4.8} \tag{4.9}
\]

To estimate the third term, \(T_3^n\), we need the fact \(\| \nabla \Phi_{h*}^{n+1} \| \infty \leq C\), and the estimate of \(\| \nabla \rho_{p_i}^{n+1} \|\).

In fact, by (3.21) and (4.4), \(\forall w_h \in V_h^r\), we have
\[
(\nabla \rho_{p_i}^{n+1}, \nabla w_h) = \sum_{i=1}^{2} q_i \left( p_{i,H}^{i,n+1} - R_h p_{i,n}^{i,n+1}, w_h \right). \tag{4.10}
\]

Taking \(w_h = \rho_{p_i}^{n+1}\) in (4.10), we can easily get
\[
\| \nabla \rho_{p_i}^{n+1} \| \leq \sum_{i=1}^{2} \| P_{i,H}^{i,n+1} - R_h p_{i,n}^{i,n+1} \| \leq C (\tau + H^{r+1}). \tag{4.11}
\]

Then, \(\| \nabla \Phi_{h*}^{n+1} \| \infty \leq C\) holds by using (4.11) and the same arguments as in (3.29).

By the regularity assumption (2.8), the projection estimate (2.11) and (4.11), \(T_3^n\) is estimated by
\[
T_3^n \leq C \left( \| \rho_{p_i}^{n+1} + \theta_{p_i}^{n+1} \| \| \nabla \Phi_{h*}^{n+1} \| \infty \| \nabla \rho_{p_i}^{n+1} \| + \| p_{i,n}^{i,n+1} \| \| \nabla \rho_{p_i}^{n+1} \| \| \nabla \rho_{p_i}^{n+1} \| \right) \\
\leq C \left( (\| \rho_{p_i}^{n+1} \| + H^{r+1}) \| \nabla \rho_{p_i}^{n+1} \| + (\tau + H^{r+1}) \| \nabla \rho_{p_i}^{n+1} \| \right) \\
\leq C \left( \tau^2 + H^{2r+2} + \| \rho_{p_i}^{n+1} \|^2 \right) + \epsilon \| \nabla \rho_{p_i}^{n+1} \|^2. \tag{4.12}
\]

Thus, by (4.8), (4.9), and (4.12), (4.7) becomes
\[
\frac{1}{2} D_\tau \| \rho_{p_i}^{n+1} \|^2 + \| \nabla \rho_{p_i}^{n+1} \|^2 \leq C (\tau^2 + H^{2r+2} + \| \rho_{p_i}^{n+1} \|^2) + \epsilon \| \nabla \rho_{p_i}^{n+1} \|^2. \tag{4.13}
\]

Applying a summation of time step \(n\) from 0 to \(J\) on both side of (4.13), where \(0 \leq J \leq N - 1\), we get the following inequality
\[
\frac{1}{2\tau} \| \rho_{p_i}^{J+1} \|^2 + \sum_{n=0}^{J} \| \nabla \rho_{p_i}^{n+1} \|^2 \leq C \sum_{n=0}^{J} \left( \tau^2 + H^{2r+2} + \| \rho_{p_i}^{n+1} \|^2 + \epsilon \| \nabla \rho_{p_i}^{n+1} \| \right).
\]
Then by discrete Gronwall’s inequality, it yields
\[
\|\rho_{p_i}^{J+1}\|^2 + \tau \sum_{n=0}^{J} \|\nabla \rho_{p_i}^{n+1}\|^2 \leq C(\tau^2 + H^{2r+2}).
\]

This implies that for \(0 \leq J \leq N - 1\),
\[
\|\rho_{p_i}^{J+1}\| \leq C(\tau + H^{r+1}).
\]

Finally, by triangle inequality and projection estimate (2.11), for \(n = 0, 1, \cdots, N - 1\), we can easily get
\[
\|P_{h^*}^{i,n+1} - P_{i,n+1}\| \leq \|\rho_{p_i}^{n+1}\| + \|\theta_{p_i}^{n+1}\| \leq C(\tau + H^{r+1}).
\]

This completes the proof. \(\Box\)

**Theorem 4.1** Suppose \((p^{i,n+1}, \phi^{n+1})\) and \((P_{h^*}^{i,n+1}, \Phi_{h^*}^{n+1})\) are the solutions of (2.2)–(2.3) and (4.3)–(4.4), respectively. Then for any \(n = 0, 1, \cdots, N - 1\), we have the following estimate:
\[
\|\Phi_{h^*}^{n+1} - \phi^{n+1}\|_1 + \|P_{h^*}^{i,n+1} - p^{i,n+1}\|_1 \leq C(\tau + h^r + H^{r+1}).
\] (4.14)

**Proof** First by (4.6), (4.11), and the projection estimate (2.12), it easily yields
\[
\|\Phi_{h^*}^{n+1} - \phi^{n+1}\|_1 \leq C(\tau + h^r + H^{r+1}).
\] (4.15)

Now, we turn to estimate \(\|P_{h^*}^{i,n+1} - p^{i,n+1}\|_1\).

Choosing \(v_h = D_\tau \rho_{p_i}^{n+1}\) in (4.7) instead of \(\rho_{p_i}^{n+1}\), we get
\[
\left(\nabla \rho_{p_i}^{n+1}, D_\tau \nabla \rho_{p_i}^{n+1}\right) + \left(D_\tau \rho_{p_i}^{n+1}, D_\tau \rho_{p_i}^{n+1}\right) = \sum_{i=1}^{3} \hat{H}_i^n,
\] (4.16)

where
\[
\hat{H}_1^n = -\left(D_\tau \theta_{p_i}^{n+1}, D_\tau \rho_{p_i}^{n+1}\right) \leq Ch^{2r+2} + \epsilon \|D_\tau \rho_{p_i}^{n+1}\|^2,
\]
\[
\hat{H}_2^n = -(D_\tau p_{i,n+1} - \partial_t p_{i,n+1}, D_\tau \rho_{p_i}^{n+1}) \leq C\tau^2 + \epsilon \|D_\tau \rho_{p_i}^{n+1}\|^2,
\]
\[
\hat{H}_3^n = -q_i \left(\left((P_{h^*}^{i,n+1} - p^{i,n+1})\nabla \Phi_{h^*}^{n+1} + p^{i,n+1}\nabla (\Phi_{h^*}^{n+1} - R_h \Phi^{n+1}), \nabla D_\tau \rho_{p_i}^{n+1}\right)\right)
\]
\[
\quad = -q_i \frac{1}{\tau} \left[\left((\rho_{p_i}^{n+1} + \theta_{p_i}^{n+1})\nabla \Phi_{h^*}^{n+1}, \nabla (\rho_{p_i}^{n+1} - \rho_{p_i}^{n})\right)\right]
\quad + \left(p^{i,n+1}\nabla \rho_{\phi}^{n+1}, \nabla (\rho_{p_i}^{n+1} - \rho_{p_i}^{n})\right).\]
On the other hand, by (4.5), (4.11), and \( \| \nabla \Phi_{h^*}^{n+1} \|_{0, \infty} \leq C \), there holds

\[
\begin{align*}
\left( \rho_{p_i}^{n+1} + \theta_{p_i}^{n+1} \right) \nabla \Phi_{h^*}^{n+1}, \nabla (\rho_{p_i}^{n+1} - \rho_{p_i}^n) \right) \\
= \left( \rho_{p_i}^{n+1} + \theta_{p_i}^{n+1} \right) \nabla \Phi_{h^*}^{n+1}, \nabla (\rho_{p_i}^{n+1} - \rho_{p_i}^n) \right) - \left( \rho_{p_i}^{n+1} + \theta_{p_i}^{n+1} \right) \nabla \Phi_{h^*}^{n+1}, \nabla (\rho_{p_i}^{n+1} - \rho_{p_i}^n) \right) \\
- \left( \rho_{p_i}^{n+1} + \theta_{p_i}^{n+1} \right) \nabla \Phi_{h^*}^{n+1}, \nabla (\rho_{p_i}^{n+1} - \rho_{p_i}^n) \right) \\
\leq \left( \rho_{p_i}^{n+1} + \theta_{p_i}^{n+1} \right) \nabla \Phi_{h^*}^{n+1}, \nabla (\rho_{p_i}^{n+1} - \rho_{p_i}^n) \right) \\
+ C(\tau^2 + H^{2r+2} + \| \nabla \rho_{p_i}^n \|^2),
\end{align*}
\]

and

\[
\begin{align*}
\left( p^{i,n+1} \nabla \rho_{p_i}^{n+1}, \nabla (\rho_{p_i}^{n+1} - \rho_{p_i}^n) \right) &= \left( p^{i,n+1} \nabla \rho_{p_i}^{n+1}, \nabla (\rho_{p_i}^{n+1} - \rho_{p_i}^n) \right) \\
&- \left( p^{i,n+1} \nabla \rho_{p_i}^{n+1} - p^{i,n} \nabla \rho_{p_i}^{n+1}, \nabla \rho_{p_i}^{n+1} \right) \\
&\leq \left( p^{i,n+1} \nabla \rho_{p_i}^{n+1}, \nabla (\rho_{p_i}^{n+1} - \rho_{p_i}^n) \right) \\
&+ C(\tau^2 + H^{2r+2} + \| \nabla \rho_{p_i}^n \|^2).
\end{align*}
\]

Then, the third term estimated by

\[
\hat{H}_3^n \leq C \frac{1}{\tau} \left[ \left( \rho_{p_i}^{n+1} + \theta_{p_i}^{n+1} \right) \nabla \Phi_{h^*}^{n+1}, \nabla (\rho_{p_i}^{n+1} - \rho_{p_i}^n) \right] \\
+ (p^{i,n+1} \nabla \rho_{p_i}^{n+1}, \nabla (\rho_{p_i}^{n+1} - \rho_{p_i}^n) \right) + \tau^2 + H^{2r+2} + \| \nabla \rho_{p_i}^n \|^2 \right].
\]

Inserting the error estimates of \( \hat{H}_1^n \), \( \hat{H}_2^n \), and \( \hat{H}_3^n \) into (4.16), it yields

\[
\begin{align*}
\frac{1}{2} D_\tau \| \nabla \rho_{p_i}^{n+1} \|^2 + \| D_\tau \rho_{p_i}^{n+1} \|^2 \leq C(\tau^2 + h^{2r+2}) + \epsilon \| D_\tau \rho_{p_i}^{n+1} \|^2 \\
+ C \frac{1}{\tau} \left[ \left( \rho_{p_i}^{n+1} + \theta_{p_i}^{n+1} \right) \nabla \Phi_{h^*}^{n+1}, \nabla (\rho_{p_i}^{n+1} - \rho_{p_i}^n) \right] \\
- \left( \rho_{p_i}^{n+1} + \theta_{p_i}^{n+1} \right) \nabla \Phi_{h^*}^{n+1}, \nabla (\rho_{p_i}^{n+1} - \rho_{p_i}^n) \right] \\
+ (p^{i,n+1} \nabla \rho_{p_i}^{n+1}, \nabla (\rho_{p_i}^{n+1} - \rho_{p_i}^n) \right) - (p^{i,n} \nabla \rho_{p_i}^{n+1}, \nabla (\rho_{p_i}^{n+1} - \rho_{p_i}^n) \right] \\
+ \tau^2 + H^{2r+2} + \| \nabla \rho_{p_i}^n \|^2 \right].
\]
Multiplying the time step size $\tau$ on both sides of (4.17), and applying a summation of time step $n$ from 0 to $J$, where $0 \leq J \leq N - 1$, by using (4.5), we get

$$\| \nabla \rho_{J+1}^{\pi} \|^2 + \tau \sum_{n=0}^{J} \| D_{\tau} \rho_{J+1}^{n+1} \|^2 \leq C \left( \sum_{n=0}^{J} \left( \tau^2 + H^{2r+2} \right) + \| \nabla \rho_{J+1}^{n} \|^2 + \epsilon \| D_{\tau} \rho_{J+1}^{n+1} \|^2 \right)$$

$$+ \left( (\rho_{J+1}^{n} + \rho_{J+1}^{n+1}) \nabla \phi_{h*}^{J+1} \cdot \nabla \rho_{J+1}^{n+1} \right) + \left( p_{J+1}^{i,n+1} \nabla \phi_{h*}^{J+1}, \nabla \rho_{J+1}^{n+1} \right)$$

$$\leq C \sum_{n=0}^{J} \left( \tau^2 + H^{2r+2} + \| \nabla \rho_{J+1}^{n} \|^2 + \epsilon \| D_{\tau} \rho_{J+1}^{n+1} \|^2 \right)$$

$$+ C (\tau^2 + H^{2r+2} + \| \nabla \rho_{J+1}^{n+1} \|^2)$$

$$\leq C \left( \tau^2 + H^{2r+2} + \sum_{n=0}^{J} \| \nabla \rho_{J+1}^{n+1} \|^2 \right) + \epsilon \| D_{\tau} \rho_{J+1}^{n+1} \|^2,$$

where we have used $\rho_{0}^{0} = 0$ by the initial condition $P_{h*}^{J,0} = \Phi_{h*}^{J+1}$.

Applying the discrete Gronwall’s inequality, it easily yields

$$\| \nabla \rho_{J+1}^{\pi} \| \leq C (\tau + H^{r+1}). \tag{4.18}$$

Thus, by (4.18) and the projection estimate (2.11), for any $n = 0, 1, \cdots, N - 1$, we can easily get

$$\| P_{h*}^{J,n+1} - p_{J,n+1}^{i,n+1} \|_1 \leq C (\tau + h^{r} + H^{r+1}). \tag{4.19}$$

Then, the desired result (4.14) is completed by (4.15) and (4.19). □

**Remark 4.1** Theorem 4.1 shows that if we choose the mesh size $h^{r} = \mathcal{O}(H^{r+1})$ for $r$th finite element, then the two-grid method can reach the same convergence order as the standard finite element method for both the electrostatic potential and concentration in $H^1$ norm. For example, if we choose the linear finite element to discrete PNP equation, i.e., $r = 1$ in this case, then our two-grid method can achieve the same convergence rate when $H = \mathcal{O}(h^{1/2})$.

In the following, we give another two-grid algorithm which is called the full decoupled scheme.
Algorithm 2 Full decoupled scheme.

Step 1. Given $P_{H}^{i,n} \in [V_{H}^{r}]^{3}$, $i = 1, 2$, find $(P_{H}^{i,n+1}, \Phi_{H}^{n+1}) \in [V_{H}^{r}]^{3}$, such that

$$(D_{\tau} P_{H}^{i,n+1}, v_{H}) + (\nabla P_{H}^{i,n+1}, \nabla v_{H}) + (q^{i} P_{H}^{i,n+1}, \nabla \Phi_{H}^{n+1}, \nabla v_{H}) = (F_{i}^{n+1}, v_{H}), \ \forall v_{H} \in V_{H}^{r}, \ (4.20)$$

$$(\nabla \Phi_{H}^{n+1}, \nabla w_{H}) - \sum_{i=1}^{2} q^{i} (P_{H}^{i,n+1}, w_{H}) = (P_{3}^{n+1}, w_{H}), \ \forall w_{H} \in V_{H}^{r}, \ (4.21)$$

Step 2. Given $(P_{H}^{i,n+1}, \Phi_{H}^{n+1}) \in [V_{H}^{r}]^{3}$ and $P_{h*}^{i,n} \in [V_{h}^{r}]^{2}$, find $(P_{h*}^{i,n+1}, \Phi_{h*}^{n+1}) \in [V_{h}^{r}]^{3}$, such that

$$(D_{\tau} P_{h*}^{i,n+1}, v_{h}) + (\nabla P_{h*}^{i,n+1}, \nabla v_{h}) + (q^{i} P_{h*}^{i,n+1}, \nabla \Phi_{h*}^{n+1}, \nabla v_{h}) = (F_{i}^{n+1}, v_{h}), \ \forall v_{h} \in V_{h}^{r}, \ (4.22)$$

$$(\nabla \Phi_{h*}^{n+1}, \nabla w_{h}) - \sum_{i=1}^{2} q^{i} (P_{h*}^{i,n+1}, w_{h}) = (F_{3}^{n+1}, w_{h}), \ \forall w_{h} \in V_{h}^{r}, \ (4.23)$$

where

$$D_{\tau} P_{H}^{i,n+1} = \frac{P_{H}^{i,n+1} - P_{H}^{i,n}}{\tau}, \text{ for } n = 0, 1, 2, \ldots, N - 1,$$

and

$$D_{\tau} P_{h*}^{i,n+1} = \frac{P_{h*}^{i,n+1} - P_{h*}^{i,n}}{\tau}, \text{ for } n = 0, 1, 2, \ldots, N - 1.$$
Proof By (2.3), (4.21), and (4.24), for any $w_H \in V^r_H$, we get
\[
(\nabla (\Phi^{n+1}_H - \Phi^I_n), \nabla w_H) = (\nabla (\Phi^{n+1}_H - \Phi^I_n), \nabla w_H)
\]
\[
= \sum_{i=1}^{2} q^i (P^{i,n+1}_H - p^{i,n+1}, w_H)
\]
\[
= \sum_{i=1}^{2} \|P^{i,n+1}_H - p^{i,n+1}\| \|w_H\|
\]
\[
\leq C \sum_{i=1}^{2} \|P^{i,n+1}_H - p^{i,n+1}\| \|w_H\|
\]
\[
\leq C \sum_{i=1}^{2} \|P^{i,n+1}_H - p^{i,n+1}\| \|w_H\|
\]
\[
\leq C \sum_{i=1}^{2} \|P^{i,n+1}_H - p^{i,n+1}\| \|w_H\|
\]
\[
\leq C (\tau + H^{r+1}) \|w_H\|
\]
\[
\leq C (\tau + H^{r+1}) \|w_H\|
\]
where we have used (3.19) in the last inequality. Then, (4.26) follows by taking $w_H = \Phi^{n+1}_H - \Phi^I_n$.

Lemma 4.4 Let $(p^{i,n+1}, \Phi^{n+1})$ and $(P^{i,n+1}_{h^*}, \Phi^{n+1}_{h^*})$ be the solutions of (2.2)–(2.3) and (4.22)–(4.23), respectively. Then for any $n = 0, 1, \cdots, N - 1$, we have
\[
\|P^{i,n+1}_{h^*} - p^{i,n+1}\| \leq C (\tau + H^{r+1}).
\]

Proof From (2.2) and (4.22), we get
\[
\begin{align*}
(D_{\tau} (P^{i,n+1}_{h^*} - p^{i,n+1}), v_h) &+ (\nabla (P^{i,n+1}_{h^*} - p^{i,n+1}), \nabla v_h) \\
= &- (D_{\tau} p^{i,n+1} - \partial_t p^{i}_{|t^{n+1}}, v_h) - q^i \left( P^{i,n+1}_{h^*} \nabla \Phi^{n+1}_{h^*} - p^{i,n+1} \nabla \phi^{n+1}, \nabla v_h \right) \\
= & I_1 + I_2.
\end{align*}
\]
In the following, we estimate $I_1$ and $I_2$, respectively.
First, by Taylor’s expansion, it yields
\[
I_1 \leq \|D_{\tau} p^{i,n+1} - \partial_t p^{i}_{|t^{n+1}}\| \|v_h\| \leq \| \frac{1}{2} \tau \cdot \partial_t p^{i}(x, \xi) \| \|v_h\|
\]
\[
\leq C \tau \|v_h\| \leq C (\tau^2 + \|v_h\|^2), \quad (t^n < \xi < t^{n+1}).
\]
Now, we recast $I_2$ as follows:
\[
I_2 = -q^i \left( P^{i,n+1}_{h^*} \nabla \Phi^{n+1}_{h^*} - p^{i,n+1} \nabla \phi^{n+1}, \nabla v_h \right)
\]
\[
= -q^i \left( (P^{i,n+1}_{h^*} - p^{i,n+1}) \nabla \Phi^{n+1}_{h^*}, \nabla v_h \right) - q^i \left( p^{i,n+1} \nabla (\Phi^{n+1}_{h^*} - \phi^{n+1}), \nabla v_h \right)
\]
\[
\leq C \left( \| P^{i,n+1}_{h^*} - p^{i,n+1} \| \| \nabla \Phi^{n+1}_{h^*} \|, \| \nabla v_h \| + \| p^{i,n+1} \| \| \nabla (\Phi^{n+1}_{h^*} - \phi^{n+1}) \| \right)
\]
\[
\leq C \left( \| P^{i,n+1}_{h^*} - p^{i,n+1} \| \| \nabla \Phi^{n+1}_{h^*} \|, \| \nabla v_h \| + \| p^{i,n+1} \| \| \nabla (\Phi^{n+1}_{h^*} - \phi^{n+1}) \| \right).
\]
We turn to estimate the second term in the right-hand side of (4.31).

First, for any $u \in W^{1,\infty}(e)$, $\forall e \in T_h$, denote the average of $u$ on the element $e$ by $ar{u} = \frac{1}{|e|} \int_e u \, dX$. We know that

$$\|u - \bar{u}\|_{0,\infty,e} \leq Ch_e^r |u|_{1,\infty,e}. \quad (4.32)$$

Let $\phi_I$ be the nodal interpolation of $\phi$ and by using (4.25) and (4.32), it yields

$$\left| \left( p^{i,n+1} \nabla (\Phi^{n+1}_H - \phi^{n+1}) - \nabla u \right) \right| \leq \left| \left( p^{i,n+1} \nabla (\Phi_H^{n+1} - \nabla \phi^{n+1}) - \nabla u \right) \right|$$

$$\leq \left| \left( p^{i,n+1} \nabla (\Phi^{n+1}_H - \phi^{n+1}) - \nabla u \right) \right| + \sum_e \left| \left( p^{i,n+1} - p^{i,n+1}_I \right) \nabla (\phi^{n+1}_I - \nabla \phi^{n+1}) - \nabla u \right|$$

$$\leq \|p^{i,n+1}\|_{0,\infty} \|\nabla (\Phi^{n+1}_H - \phi^{n+1})\| \|\nabla u\|$$

$$+ \sum_e \|p^{i,n+1} - p^{i,n+1}_I\|_{0,\infty,e} \|\nabla (\phi^{n+1}_I - \nabla \phi^{n+1})\|_{0,e} \|\nabla u\|_{0,e}$$

$$+ Ch_r^{i+1} \sum_e \|\phi^{n+1}_I\|_{3,e} \|\nabla u\|_{0,e}$$

$$\leq C \left( (\tau + H_r^{i+1}) \|\nabla u\| + H^2 \sum_e \|p^{i,n+1}\|_{1,\infty,e} \|\nabla u\|_{0,e} + H^{i+1} \sum_e \|\phi^{n+1}_I\|_{3,e} \|\nabla u\|_{0,e} \right)$$

$$\leq C(\tau + H_r^{i+1}) \|\nabla u\|, \quad (4.33)$$

where we have used (4.24) and (4.26). Inserting (4.33) into (4.31) and by using (2.11), we get

$$I_2 \leq C \left( \|P^{i,n+1}_h - p^{i,n+1}\| + \tau + H_r^{i+1} \right) \|\nabla u\|$$

$$\leq C \left( \|P^{i,n+1}_h - R_h p^{i,n+1}\| + \|R_h p^{i,n+1} - p^{i,n+1}\| + \tau + H_r^{i+1} \right) \|\nabla u\|$$

$$\leq C \left( \|P^{i,n+1}_h - R_h p^{i,n+1}\| + \tau + H_r^{i+1} \right) \|\nabla u\|$$

$$\leq C \left( \|P^{i,n+1}_h - R_h p^{i,n+1}\|^2 + \tau^2 + H^{2r+2} \right) + \epsilon \|\nabla u\|^2. \quad (4.34)$$

Combining (4.29), (4.30), and (4.34) and using (2.9)–(2.10), it yields

$$\left( D_\tau (P^{i,n+1}_h - R_h p^{i,n+1}), v_h \right) + \left( \nabla (P^{i,n+1}_h - R_h p^{i,n+1}), \nabla v_h \right)$$

$$\leq - \left( D_\tau (R_h p^{i,n+1} - p^{i,n+1}), v_h \right) + \left( q^{i,n+1} \nabla (R_h \phi^{n+1} - \phi^{n+1}), \nabla v_h \right)$$
Then by (2.2) and (4.22), and using (2.9), we have the proof of (4.38).

Since the only difference between Algorithms 1 and 2 is (4.22), which is not used in the proof of (4.28), we can easily get

$$\| P^{i,n+1}_{h*} - R_h P^{i,n+1} \| \leq C (\tau + H^r + H^{r+1}).$$

(4.36)

Then, the desired result (4.28) established by (4.36) and (2.11).

By using the above lemmas, similar to Theorem 4.1, we have the following result:

**Theorem 4.2** Let \((p^{i,n+1}, \phi^{n+1})\) be the solution of (2.2)–(2.3). Assume \((P^{i,n+1}_{h*}, \Phi^{n+1}_{h*})\) is the solution obtained by Algorithm 2. Then for any \(n = 0, 1, \cdots, N - 1\), we have the following estimate:

$$\| \Phi^{n+1}_{h*} - \phi^{n+1} \|_1 + \| P^{i,n+1}_{h*} - p^{i,n+1} \|_1 \leq C (\tau + h^r + H^{r+1}).$$

(4.37)

**Proof** First, by using the same arguments as for (4.15), we can easily get

$$\| \Phi^{n+1}_{h*} - \phi^{n+1} \|_1 \leq C (\tau + h^r + H^{r+1}),$$

(4.38)

since the only difference between Algorithms 1 and 2 is (4.22), which is not used in the proof of (4.38).

Now, we only need to estimate \(\| P^{i,n+1}_{h*} - p^{i,n+1} \|_1\). Let \(e^{n+1}_h = P^{i,n+1}_{h*} - R_h p^{i,n+1}\). Then by (2.2) and (4.22), and using (2.9), we have

$$\left( D_t e^{n+1}_h, v_h \right) + (\nabla e^{n+1}_h, \nabla v_h) = - (D_t \theta^{n+1}, v_h) - (D_t p^{i,n+1} - \partial_t p^{i,n+1}, v_h)$$

$$- q_i \left( (P^{i,n+1}_{h*} - p^{i,n+1}) \nabla \Phi^{n+1}_{h*}, \nabla v_h \right) - q_i \left( (P^{i,n+1}_{h*} - p^{i,n+1}) \nabla \Phi^{n+1}_{h*}, \nabla v_h \right)$$

$$\leq C \left( h^{2r+2} + \tau^2 + \epsilon \| v_h \|_2 \right) - q_i \left( (P^{i,n+1}_{h*} - p^{i,n+1}) \nabla \Phi^{n+1}_{h*}, \nabla v_h \right)$$

(4.39)

where \(\theta^{n+1} = R_h p^{i,n+1} - p^{i,n+1}\). Taking \(v_h = D_t e^{n+1}_h\) in (4.39), we get

$$\frac{1}{2} D_t \| \nabla e^{n+1}_h \|_2^2 + \| D_t e^{n+1}_h \|_2^2 \leq C \left( \tau^2 + h^{2r+2} + \epsilon \| D_t e^{n+1}_h \|_2^2 \right)$$

$$- q_i \left( (P^{i,n+1}_{h*} - p^{i,n+1}) \nabla \Phi^{n+1}_{h*}, \nabla D_t e^{n+1}_h \right)$$

$$- q_i \left( (P^{i,n+1}_{h*} - p^{i,n+1}) \nabla \Phi^{n+1}_{h*}, \nabla D_t e^{n+1}_h \right)$$

$$\leq C \left( \tau^2 + h^{2r+2} + \epsilon \| D_t e^{n+1}_h \|_2^2 \right) + I_3 + I_4.$$  

(4.40)
On the other hand, by (4.28) and (4.33), we rewrite $I_3$ and $I_4$ as follows:

$$I_3 = -q^i \left( (p_{h,1}^{i,n+1} - p_{h,2}^{i,n+1}) \nabla \Phi_H^{n+1} + \nabla \tau \mathbf{e}_h^{n+1} \right)$$

$$= -q^i \frac{1}{\tau} \left[ \left( (e_h^{n+1} + \theta^{n+1}) \nabla \Phi_H^{n+1} + \nabla \tau \mathbf{e}_h^{n+1} \right) - \left( (e_h^n + \theta^n) \nabla \Phi_H^n + \nabla \tau \mathbf{e}_h^n \right) - \left( (e_h^{n+1} + \theta^{n+1}) \nabla \Phi_H^{n+1} - (e_h^n + \theta^n) \nabla \Phi_H^n - \nabla \mathbf{e}_h^n \right) \right]$$

$$\leq C \frac{1}{\tau} \left[ \left( (e_h^{n+1} + \theta^{n+1}) \nabla \Phi_H^{n+1} + \nabla \mathbf{e}_h^{n+1} \right) - \left( (e_h^n + \theta^n) \nabla \Phi_H^n + \nabla \mathbf{e}_h^n \right) + C (\tau^2 + H^{2r+2} + \| \nabla \mathbf{e}_h^n \|^2) \right]. \quad (4.41)$$

Similarly, $I_4$ is bounded by

$$I_4 = -q^i \left( (p_{h,1}^{i,n+1} \nabla (\Phi_H^{n+1} - R_h \phi^{n+1}) + \nabla \tau \mathbf{e}_h^{n+1} \right)$$

$$= -q^i \frac{1}{\tau} \left[ \left( (p_{i}^{i,n+1} \nabla (\Phi_H^{n+1} - R_h \phi^{n+1}) + \nabla \tau \mathbf{e}_h^{n+1} \right) - \left( (p_{h}^{i,n} \nabla (\Phi_H^n - R_h \phi^n) + \nabla \mathbf{e}_h^n \right) - \left( (p_{h}^{i,n+1} \nabla (\Phi_H^{n+1} - R_h \phi^{n+1}) - p_{h}^{i,n} \nabla (\Phi_H^n - R_h \phi^n) - \nabla \mathbf{e}_h^n \right) \right]$$

$$\leq C \frac{1}{\tau} \left[ \left( (p_{h}^{i,n+1} \nabla (\Phi_H^{n+1} - R_h \phi^{n+1}) + \nabla \mathbf{e}_h^{n+1} \right) - \left( (p_{h}^{i,n} \nabla (\Phi_H^n - R_h \phi^n) + \nabla \mathbf{e}_h^n \right) + C (\tau^2 + H^{2r+2} + \| \nabla \mathbf{e}_h^n \|^2) \right]. \quad (4.42)$$

Inserting (4.41) and (4.42) into (4.40), it yields

$$\frac{1}{2} \tau \| \nabla \mathbf{e}_h^{n+1} \|^2 + H \| \nabla \mathbf{e}_h^{n+1} \|^2 \leq C \frac{1}{\tau} \left( \tau^2 + H^{2r+2} + \| \nabla \mathbf{e}_h^n \|^2 \right)$$

$$+ C (\tau^2 + H^{2r+2} + \| \nabla \mathbf{e}_h^{n+1} \|^2)$$

$$+ \frac{1}{\tau} \left[ \left( (e_h^{n+1} + \theta^{n+1}) \nabla \Phi_H^{n+1} + \nabla \mathbf{e}_h^{n+1} \right) - \left( (e_h^n + \theta^n) \nabla \Phi_H^n + \nabla \mathbf{e}_h^n \right) \right]$$

$$+ \frac{1}{\tau} \left[ \left( p_{i}^{i,n+1} \nabla (\Phi_H^{n+1} - R_h \phi^{n+1}) + \nabla \mathbf{e}_h^{n+1} \right) - \left( p_{h}^{i,n} \nabla (\Phi_H^n - R_h \phi^n) + \nabla \mathbf{e}_h^n \right) \right]. \quad (4.43)$$

Multiplying the time step size $\tau$ on both sides of (4.43), applying a summation of time step $n$ from 0 to $J$, where $0 \leq J \leq N - 1$, and using (28.2) and (28.3), we get

$$\| \nabla \mathbf{e}_h^{i+1} \|^2 + \tau \sum_{n=0}^{J} \| \nabla \mathbf{e}_h^{n+1} \|^2 \leq C \left( \sum_{n=0}^{J} \left( \tau^2 + H^{2r+2} + \| \nabla \mathbf{e}_h^n \|^2 + \| \nabla \mathbf{e}_h^{n+1} \|^2 \right) \right)$$

$$+ \left( (e_h^{n+1} + \theta^{n+1}) \nabla \Phi_H^{n+1} + \nabla \mathbf{e}_h^{n+1} \right) + \left( p_{i}^{i,n+1} \nabla (\Phi_H^{n+1} - R_h \phi^{n+1}) + \nabla \mathbf{e}_h^{n+1} \right)$$
\[
\leq C \sum_{n=0}^{J} \left( \tau^2 + H^{2r+2} + \| \nabla e_h^n \|^2 + \epsilon \| D_t e_h^{n+1} \|^2 \right) + C \left( \tau^2 + H^{2r+2} + \| \nabla e_h^{J+1} \|^2 \right)
\]
\[
\leq C \left( \tau^2 + H^{2r+2} + \sum_{n=0}^{J} \| \nabla e_h^{n+1} \|^2 \right) + \sum_{n=0}^{J} \epsilon \| D_t e_h^{n+1} \|^2,
\]

where we have used \( e_h^0 = 0 \) from the initial condition \( P_{h^*}^{i,0} = R_h p^{i,0} \).

Applying the discrete Gronwall’s inequality in the above inequality, it easily yields

\[
\| \nabla e_h^{J+1} \| \leq C (\tau + H^{r+1}). \tag{4.44}
\]

Thus, from (4.44) and the projection estimate (2.11), for any \( n = 0, 1, \ldots, N - 1 \), we get

\[
\| P_{h^*}^{i,n+1} - p^{i,n+1} \|_1 \leq C (\tau + h^r + H^{r+1}). \tag{4.45}
\]

Then, we complete the proof of Theorem 4.2 from (4.38) and (4.45).

\[ \square \]

**Remark 4.2** Theorem 4.2 shows that the optimal convergence rate for both the electrostatic potential and concentration in \( H^1 \) norm could be reached when \( h^r = O(H^{r+1}) \), which indicates our two-grid method retains the same order of accuracy as the standard finite element method under the assumption \( H \leq C h^{r+1} \). Moreover, since Algorithm 2 is fully decoupled in step 2, it can be solved in parallel on the fine grid level at each time step, the efficiency of which could be much better than the standard finite element method.

### 5 Numerical experiments

We now present numerical experiments to demonstrate the effectiveness and efficiency of the two-grid approach. To implement the algorithms, the code is written in Fortran 90 and all the computations are carried out on the computer with Dual core 96 GB RAM HPZ280.

**Example 5.1** Let the computational domain be the unit square \( \Omega_1 = [0, 1] \times [0, 1] \), and a uniform triangular partition with \( M + 1 \) nodes in each direction is used. For the coarse-grid space and the fine-grid space, the domain \( \Omega \) is uniformly divided by the triangulation with mesh sizes \( H \) and \( h \), respectively. Particularly, for the odd grid, the uniform triangular partition with \( M = 3 \) on the coarse-grid space and \( M = 9 \) on the fine-grid space is shown in Fig. 1. In the computation of the two-grid method, the solutions \( p_H^i \) and \( \phi_H \) on the coarse-grid space are interpolated into the fine-grid space by finite element basis functions. For example, in Fig. 1, the values of the nodes “34,35,36,37,44,45,46,54,55,64” on the fine-grid space (right) are generated
Fig. 1 A uniform triangular partition for the odd grid. The uniform partition with mesh size $H = 1/3$ on the coarse-grid space (left) and the uniform partition with the mesh size $h = 1/9$ on the fine-grid space (right).

by nodes “6,7,10” in element ① on the coarse-grid space (left). The values of the nodes “37,46,55,64” on the red line can be generated by nodes “6,7,10” in element ① or nodes “7,10,11” in element ② on coarse-grid space, the effect is the same.

We choose $q^1 = 1, q^2 = -1$ and consider the following PNP equations (cf. [5]):

$$\begin{align*}
\frac{\partial p^1}{\partial t} - \nabla \cdot (\nabla p^1 + p^1 \nabla \phi) &= F_1, \\
\frac{\partial p^2}{\partial t} - \nabla \cdot (\nabla p^2 - p^2 \nabla \phi) &= F_2, \\
-\Delta \phi - (p^1 - p^2) &= F_3.
\end{align*}$$  \hspace{1cm} (5.1)

The initial-boundary condition and the right-hand side functions $F_i, ~ i = 1, 2, 3$ are chosen such that the exact solutions of (5.1) are given by

$$\begin{align*}
p^1(t, x, y) &= \sin(t) \sin(2\pi x) \sin(2\pi y), \\
p^2(t, x, y) &= \sin(2t) \sin(3\pi x) \sin(3\pi y), \\
\phi(t, x, y) &= (1 - e^{-t}) \sin(\pi x) \sin(\pi y).
\end{align*}$$

In the following, we first present the numerical results of standard finite element method (2.6)–(2.7), and then show the results of Algorithms 1 and 2.

To solve the nonlinear coupled system (2.6)–(2.7), we use the following algorithm which is introduced in [5, 11] to get the finite element solution.
In our computation, the piecewise linear finite elements on a uniform triangular mesh are used to discretize the PNP equations. The Gummel iteration \((1.4)\) is used during the finite element computation on each time level in step 3. We choose the time step \(\tau = h^2\) and set the final time \(T = 1.0\). The tolerance \(\epsilon = 1.0 \times 10^{-6}\) is chosen for the nonlinear iteration in Algorithm 3. Particularly, we adopt the AMG-PCG and AMG-PGMRES solver to solve the algebraic system \(Ax = b\) for the Poisson equation and Nernst-Planck equations, respectively, and the inneriteration stopped if the Euclidean norm of the residual vector is less than \(10^{-8}\). The numerical results in Tables 1 and 2 at \(t = 0.5\) show that the errors for \(\Phi_h^1\) and \(P_h^i\) \((i = 1, 2)\) in \(L^2\) norm and \(H^1\) norm are second-order and first-order reduction, respectively, which coincides with the convergence theory shown in (3.19) and (3.35).

The exact solution and the two-grid solution in Algorithm 1 when \(h = 1/64, t = 0.5\) are shown in Figs. 2, 3, and 4. Comparing the exact solution (a) and the two-grid solution (b), we can easily find that the two-grid finite element solution and the exact one are similar, which indicates the validity of the numerical test.

Tables 3 and 4 show the errors at \(t = 0.5\) between the exact solution and the two-grid solution of Algorithm 1 with varying mesh size \(H = \sqrt{h}\), where the order

\[\text{Algorithm 3 Standard finite element method.}\]

Step 1. Initialization for the time marching: Set time step \(n = 0\), and get the initial value \((P_h^{1,0}, P_h^{2,0}, \Phi_h^0) \in [V_h^r]^3\).

Step 2. Initialization for nonlinear iteration: Let \((P_h^{1,n+1,0}, P_h^{2,n+1,0}, \Phi_h^{n+1,0}) = (P_h^{1,n}, P_h^{2,n}, \Phi_h^n)\) when \(n \geq 0\) and \(l = 0\).

Step 3. Finite element computation on each time level: For \(l \geq 0\), compute \((P_h^{1,n+1,l+1}, P_h^{2,n+1,l+1}, \Phi_h^{n+1,l+1}) \in [V_h^r]^3\), such that for all \((v_{1h}, v_{2h}, w_h) \in [V_h^r]^3\),

\[
\begin{align*}
\frac{1}{\tau} (P_h^{1,n+1,l+1}, v_{1h}) + (\nabla P_h^{1,n+1,l+1}, \nabla v_{1h}) + (P_h^{1,n+1,l+1}, \nabla \Phi_h^{n+1,l}, \nabla v_{1h}) &= (F_1^{n+1}, v_{1h}) + \frac{1}{\tau} (P_h^{1,n}, v_{1h}), \\
\frac{1}{\tau} (P_h^{2,n+1,l+1}, v_{2h}) + (\nabla P_h^{2,n+1,l+1}, \nabla v_{2h}) + (P_h^{2,n+1,l+1}, \nabla \Phi_h^{n+1,l}, \nabla v_{2h}) &= (F_2^{n+1}, v_{2h}) + \frac{1}{\tau} (P_h^{2,n}, v_{2h}), \\
(\nabla \Phi_h^{n+1,l+1}, \nabla w_h) - (P_h^{1,n+1,l+1}, P_h^{2,n+1,l+1}, w_h) &= (F_3^{n+1}, w_h).
\end{align*}
\]

Step 4. Checking the stopping criteria for nonlinear iteration: For a given tolerance \(\epsilon\), stop the iteration when

\[
\|P_h^{1,n+1,l+1} - P_h^{1,n+1,l}\| + \|P_h^{2,n+1,l+1} - P_h^{2,n+1,l}\| + \|\Phi_h^{n+1,l+1} - \Phi_h^{n+1,l}\| \leq \epsilon,
\]

and set \((P_h^{1,n+1,0}, P_h^{2,n+1,0}, \Phi_h^{n+1,0}) = (P_h^{1,n+1,l+1}, P_h^{2,n+1,l+1}, \Phi_h^{n+1,l+1})\). Otherwise, set \(l \leftarrow l + 1\) and go to Step 3 to continue the nonlinear iteration.

Step 5. Time marching: Stop if \(n + 1 = N\). Otherwise, set \(n \leftarrow n + 1\), and go to Step 2.
Table 1 $L^2$ error of the standard finite element method at $t = 0.5$ (Example 5.1)

| $h$  | $\|\Phi_h - \phi\|$ | Order | $\|P^1_h - p^1\|$ | Order | $\|P^2_h - p^2\|$ | Order |
|------|---------------------|-------|---------------------|-------|---------------------|-------|
| 1/9  | 7.3983E-03          |       | 3.2614E-02          |       | 1.2117E-01          |       |
| 1/16 | 2.4124E-03          | 1.95  | 1.0904E-02          | 1.90  | 4.2949E-02          | 1.80  |
| 1/25 | 9.9267E-04          | 1.99  | 4.5135E-03          | 1.98  | 1.8098E-02          | 1.94  |
| 1/36 | 4.8039E-04          | 1.99  | 2.1894E-03          | 1.98  | 8.8305E-03          | 1.97  |
| 1/49 | 2.5946E-04          | 2.00  | 1.1835E-03          | 2.00  | 4.7870E-03          | 1.99  |
| 1/64 | 1.5221E-04          | 2.00  | 6.9466E-04          | 2.00  | 2.8131E-03          | 1.99  |

Table 2 $H^1$ error of the standard finite element method at $t = 0.5$ (Example 5.1)

| $h$  | $\|\Phi_h - \phi\|_1$ | Order | $\|P^1_h - p^1\|_1$ | Order | $\|P^2_h - p^2\|_1$ | Order |
|------|----------------------|-------|---------------------|-------|---------------------|-------|
| 1/9  | 1.5014E-01           |       | 7.1128E-01          |       | 2.6894E+00          |       |
| 1/16 | 8.5653E-02           | 0.98  | 4.1627E-01          | 0.93  | 1.6096E+00          | 0.89  |
| 1/25 | 5.4812E-02           | 1.00  | 2.7032E-01          | 0.97  | 1.0454E+00          | 0.97  |
| 1/36 | 3.8128E-02           | 1.00  | 1.9159E-01          | 0.94  | 7.3134E-01          | 0.98  |
| 1/49 | 2.8011E-02           | 1.00  | 1.4457E-01          | 0.91  | 5.3916E-01          | 0.99  |
| 1/64 | 2.1458E-02           | 1.00  | 1.1497E-01          | 0.86  | 4.1421E-01          | 0.99  |

Fig. 2 The exact solution a and two-grid solution b of $\phi$: $h = 1/64$, $\tau = h^2$, $t = 0.5$
Fig. 3 The exact solution (a) and two-grid solution (b) of $p^1$: $h = 1/64$, $\tau = h^2$, $t = 0.5$

Fig. 4 The exact solution (a) and two-grid solution (b) of $p^2$: $h = 1/64$, $\tau = h^2$, $t = 0.5$

Table 3 $L^2$ error of the two-grid solutions of Algorithm 1 at $t = 0.5$ (Example 5.1)

| $H$ | $h = H^2$ | $\| \Phi_h^* - \phi \|$ | Order | $\| P_{h^*}^1 - p^1 \|$ | Order | $\| P_{h^*}^2 - p^2 \|$ | Order |
|-----|-----------|-----------------|-------|-----------------|-------|-----------------|-------|
| 1/3 | 1/9       | 7.4133E−03      | –     | 3.2547E−02      | –     | 1.2124E−01      | –     |
| 1/4 | 1/16      | 2.4206E−03      | 1.95  | 1.0885E−02      | 1.90  | 4.2978E−02      | 1.80  |
| 1/5 | 1/25      | 9.9652E−04      | 1.99  | 4.5062E−03      | 1.98  | 1.8111E−02      | 1.94  |
| 1/6 | 1/36      | 4.8235E−04      | 1.99  | 2.1860E−03      | 1.98  | 8.8367E−03      | 1.97  |
| 1/7 | 1/49      | 2.6054E−04      | 2.00  | 1.1817E−03      | 2.00  | 4.7904E−03      | 1.99  |
| 1/8 | 1/64      | 1.5285E−04      | 2.00  | 6.9361E−04      | 2.00  | 2.8151E−03      | 1.99  |
represents the convergence order relating to the fine-grid size $h$ in $L^2$ or $H^1$ norm. The errors indicate that the numerical results coincide with the theoretical result in Theorem 4.1 when $r = 1$. For Algorithm 2, the errors at $t = 0.5$ between the exact solution and the two-grid solution with varying mesh size $H = \sqrt{h}$ are shown in Table 5, where the order denotes the convergence order relating to the fine-grid size $h$ in $H^1$ norm.

Comparing Tables 3 and 4 with Tables 1 and 2, respectively, we can find that when $H = O(h^{1/2})$, the errors in $H^1$ norm and $L^2$ norm approximate the first-order and second-order, respectively, which indicates the solution of Algorithm 1 remains the same convergence order as the standard finite element method. Similarly, by comparing the results in Table 5 with those in Table 2, the errors show that the full decoupled two-grid Algorithm 2 can also achieve the same order of accuracy as the standard finite element method. To show the efficiency of the two-grid method with the change of time, we also present the results in $H^1$ norm error at $t = 1.0$ in Tables 6, 7, and 8 computed by standard finite element method, Algorithm 1 and Algorithm 2, respectively. By comparing Tables 7 and 8 with Table 6, the results show that the two-grid method remains the same convergence order as the standard finite element method for a long time behavior.

The CPU time cost of Algorithm 3 (the finite element method combined with the Gummel iteration), Algorithm 1, and Algorithm 2 at $t = 0.5, 1.0$ is given in Tables 9 and 10, where the letter $h$ represents the size of grid in Algorithm 3 and also the size of the fine grid in Algorithms 1 and 2. As shown in Tables 9 and 10, the CPU

### Table 4 $H^1$ error of the two-grid solutions of Algorithm 1 at $t = 0.5$ (Example 5.1)

| $H$ | $h = H^2$ | $\|\Phi_h^* - \phi\|_1$ | Order | $\|P_{h^*}^1 - p^1\|_1$ | Order | $\|P_{h^*}^2 - p^2\|_1$ | Order |
|-----|-----------|----------------|-------|----------------|-------|----------------|-------|
| 1/3 | 1/9       | 1.5014E−01    |       | 7.0997E−01    |       | 2.6896E+00    |       |
| 1/4 | 1/16      | 8.5654E−02    | 0.98  | 4.1360E−01    | 0.94  | 1.6093E+00    | 0.89  |
| 1/5 | 1/25      | 5.4812E−02    | 1.00  | 2.6605E−01    | 0.99  | 1.0447E+00    | 0.97  |
| 1/6 | 1/36      | 3.8125E−02    | 1.00  | 1.8547E−01    | 0.99  | 7.3026E−01    | 0.98  |
| 1/7 | 1/49      | 2.8006E−02    | 1.00  | 1.3635E−01    | 1.00  | 5.3766E−01    | 0.99  |
| 1/8 | 1/64      | 2.1450E−02    | 1.00  | 1.0448E−01    | 1.00  | 4.1223E−01    | 0.99  |

### Table 5 $H^1$ error of the two-grid solutions of Algorithm 2 at $t = 0.5$ (Example 5.1)

| $H$ | $h = H^2$ | $\|\Phi_h^* - \phi\|_1$ | Order | $\|P_{h^*}^1 - p^1\|_1$ | Order | $\|P_{h^*}^2 - p^2\|_1$ | Order |
|-----|-----------|----------------|-------|----------------|-------|----------------|-------|
| 1/3 | 1/9       | 1.5014E−01    |       | 7.0999E−01    |       | 2.6896E+00    |       |
| 1/4 | 1/16      | 8.5657E−02    | 0.98  | 4.1360E−01    | 0.94  | 1.6093E+00    | 0.89  |
| 1/5 | 1/25      | 5.4814E−02    | 1.00  | 2.6613E−01    | 0.99  | 1.0447E+00    | 0.97  |
| 1/6 | 1/36      | 3.8127E−02    | 1.00  | 1.8554E−01    | 0.99  | 7.3032E−01    | 0.98  |
| 1/7 | 1/49      | 2.8007E−02    | 1.00  | 1.3641E−01    | 1.00  | 5.3776E−01    | 0.99  |
| 1/8 | 1/64      | 2.1451E−02    | 1.00  | 1.0454E−01    | 1.00  | 4.1227E−01    | 0.99  |
time cost by Algorithms 1 or 2 is much less than that by Algorithm 3 as $h$ becomes small, which reveals that the two-grid method is more efficient than the finite element method combined with the Gummel iteration. Moreover, Algorithm 2 could achieve a better effect for large-scale problems if a parallel program is applied at each time level.

In order to make more observation about the efficiency and effectiveness of the two-grid method, we consider another example whose the electrostatic potential is a Gaussian function as follows:

**Example 5.2** Set the computational domain $\Omega = [-0.25, 1.25] \times [-0.25, 1.25]$ and $t \in [0, 1]$. We consider problem (5.1) and the source terms $F_i, i = 1, 2, 3$ are obtained by the exact solutions

\[
\begin{align*}
\phi(t, x, y) &= \beta(t)e^{-50r^2(t, x, y)}, \\
p^1(t, x, y) &= x(x - 1) \sin(2\pi y) \sin(2t), \\
p^2(t, x, y) &= x^3(1 - x)y(1 - y)^3 \sin(t),
\end{align*}
\]

with

\[
\begin{align*}
\beta(t) &= 1 - e^{-5(0.98t+0.01)}, \\
r^2(t, x, y) &= (x - 0.4t - 0.3)^2 + (y - 0.4t - 0.3)^2.
\end{align*}
\]

Note that $\phi$ is a Gaussian function, whose center moves from $(0.3, 0.3)$ at $t = 0$ to $(0.7, 0.7)$ at $t = 1$ (see [41]). In this example, we choose the time step size $\tau = 0.01$.

### Table 7 $H^1$ error of the two-grid solutions of Algorithm 1 at $t = 1.0$ (Example 5.1)

| $H$ | $h = H^2$ | $\|\Phi^*_h - \phi\|_1$ | Order | $\|P^1_* - p^1\|_1$ | Order | $\|P^2_* - p^2\|_1$ | Order |
|-----|-----------|------------------|-------|------------------|-------|------------------|-------|
| 1/3 | 1/9       | 2.4347E−01       |       | 1.2606E+00       |       | 2.9313E+00       |       |
| 1/4 | 1/16      | 1.3759E−01       | 0.99  | 7.2654E−01       | 0.96  | 1.7397E+00       | 0.91  |
| 1/5 | 1/25      | 8.8161E−02       | 1.00  | 4.6857E−01       | 0.98  | 1.1307E+00       | 0.97  |
| 1/6 | 1/36      | 6.1249E−02       | 1.00  | 3.2683E−01       | 0.99  | 7.8983E−01       | 0.98  |
| 1/7 | 1/49      | 4.5008E−02       | 1.00  | 2.4116E−01       | 0.99  | 5.8205E−01       | 0.99  |
| 1/8 | 1/64      | 3.4463E−02       | 1.00  | 1.8566E−01       | 0.98  | 4.4659E−01       | 0.99  |
Table 8 \( H^1 \) error of the two-grid solutions of Algorithm 2 at \( t = 1.0 \) (Example 5.1)

| \( H \) | \( h = H^2 \) | \( \| \Phi^*_h - \phi \|_1 \) | Order | \( \| P_h^{1,*} - p^1 \|_1 \) | Order | \( \| P_h^{2,*} - p^2 \|_1 \) | Order |
|---|---|---|---|---|---|---|---|
| 1/3 | 1/9 | 2.4347E−01 | – | 1.2606E+00 | – | 2.9310E+00 | – |
| 1/4 | 1/16 | 1.3759E−01 | 0.99 | 7.2618E−01 | 0.96 | 1.7394E+00 | 0.91 |
| 1/5 | 1/25 | 8.8164E−02 | 1.00 | 4.6788E−01 | 0.98 | 1.1303E+00 | 0.97 |
| 1/6 | 1/36 | 6.1250E−02 | 1.00 | 3.2571E−01 | 0.99 | 7.8927E−01 | 0.98 |
| 1/7 | 1/49 | 4.5008E−02 | 1.00 | 2.3957E−01 | 1.00 | 5.8125E−01 | 0.99 |
| 1/8 | 1/64 | 3.4462E−02 | 1.00 | 1.8353E−01 | 1.00 | 4.4554E−01 | 1.00 |

The errors of the standard finite element method and the two-grid method are listed in Tables 11, 12, and 13. Comparing Tables 12 and 13 with Table 11, the results show that the two-grid method remains the same convergence order as the standard finite element method for the fixed time step size, where the order denotes the convergence order varying with \( h \) in \( H^1 \) norm.

6 Application in ion channel

In this section, we shall apply our two-grid method to solve the PNP model in ion channel. The computation is carried out by MATLAB R2012a on a microcomputer and the program is under the frame work of iFEM toolbox (https://bitbucket.org/ifem/ifem). It is shown by numerical results that the two-grid method still works in practical ion channel problem.

Example 6.1 We consider the following PNP model for simulating asymmetrical conductance changes in Gramicidin A (gA) with two ion species in a 1:1 CsCl

Table 9 The total CPU time (s) at \( t = 0.5 \) (Example 5.1)

| \( h \) | Algorithm 3 | Algorithm 1 | Algorithm 2 |
|---|---|---|---|
| CPU time | CPU time | CPU time |
| 1/9 | 4.35 | 1.19 | 0.62 |
| 1/16 | 37.62 | 7.83 | 6.19 |
| 1/25 | 351.76 | 29.52 | 22.89 |
| 1/36 | 3423.40 | 99.10 | 84.28 |
| 1/49 | 8954.83 | 253.18 | 225.79 |
| 1/64 | 28221.89 | 666.44 | 603.31 |
Table 10  The total CPU time (s) at $t = 1.0$ (Example 5.1)

| $h$   | Algorithm 3 CPU time | Algorithm 1 CPU time | Algorithm 2 CPU time |
|-------|-----------------------|-----------------------|-----------------------|
| 1/9   | 8.44                  | 2.13                  | 1.68                  |
| 1/16  | 73.55                 | 14.45                 | 11.36                 |
| 1/25  | 659.38                | 52.88                 | 45.83                 |
| 1/36  | 5032.74               | 186.54                | 157.75                |
| 1/49  | 15753.81              | 476.88                | 435.98                |
| 1/64  | 49090.98              | 1282.95               | 1166.21               |

solution with valence $+1$ and $-1$, respectively,

$$
\begin{align*}
\frac{\partial p}{\partial t} &= \nabla \cdot D_p \left( \nabla p + \frac{e}{k_B T} p \nabla \phi \right), \quad \text{in } \Omega_s, \\
\frac{\partial n}{\partial t} &= \nabla \cdot D_n \left( \nabla n - \frac{e}{k_B T} n \nabla \phi \right), \quad \text{in } \Omega_s, \\
-\nabla \cdot (\varepsilon \nabla \phi) &= (p - n)e, \quad \text{in } \Omega,
\end{align*}
$$

(6.1)

where $\Omega = \Omega_s \cup \Omega_m$, $\Omega_s$ is the solvent region, $\Omega_m$ is the solute region, $\phi$ is the electrostatic potential, and $p(x)$ and $n(x)$ are the concentrations of the positive ions and the negative ions in the bulk solvent respectively. The constant coefficients $D_p$ and $D_n$ are the diffusion coefficients of the positive ions and the negative ions respectively, $k_B T$ is the Boltzmann energy constant, $e$ is the charge for one electron, and $\varepsilon = \begin{cases} 2\varepsilon_0, & \text{in } \Omega_m, \\ 80\varepsilon_0, & \text{in } \Omega_s, \end{cases}$ is the dielectric permittivity coefficient, where $\varepsilon_0$ is the dielectric constant of vacuum.

Suppose $\Gamma_1$ and $\Gamma_2$ are the interfaces, where $\Gamma_1$ is the boundaries of membranes, $\Gamma_2$ is the boundaries of protein exposed to solvent, $\Gamma_3$ is the outside boundaries of

Table 11  $H^1$ error of the standard finite element method at $t = 1.0$ (Example 5.2)

| $h$   | $\| \Phi_h - \phi \|_1$ | Order | $\| P^1_h - P^1 \|_1$ | Order | $\| P^2_h - P^2 \|_1$ | Order |
|-------|--------------------------|-------|------------------------|-------|------------------------|-------|
| 1/9   | 1.0253E+00               | –      | 4.6645E+01             | –      | 1.1336E+01             | –      |
| 1/16  | 6.5794E+00               | 0.77   | 2.5931E+01             | 1.02   | 6.5054E+02             | 0.97   |
| 1/25  | 4.5454E+00               | 0.83   | 1.6293E+01             | 1.04   | 4.1802E+02             | 0.99   |
| 1/36  | 3.2584E+00               | 0.91   | 1.1237E+01             | 1.02   | 2.9101E+02             | 0.99   |
| 1/49  | 2.4465E+00               | 0.93   | 8.2313E+01             | 1.01   | 2.1386E+02             | 1.00   |
| 1/64  | 1.9131E+00               | 0.92   | 6.3268E+02             | 0.99   | 1.6389E+02             | 1.00   |
Table 12  \( H^1 \) error of the two-grid solutions of Algorithm 1 at \( t = 1.0 \) (Example 5.2)

| \( H \) | \( h = H^2 \) | \( \| \Phi_h^* - \phi \|_1 \) | Order | \( \| P_h^{1,*} - p^1 \|_1 \) | Order | \( \| P_h^{2,*} - p^2 \|_1 \) | Order |
|---|---|---|---|---|---|---|---|
| 1/3 | 1/9 | 1.0255E+00 | – | 4.6693E−01 | – | 1.1336E−01 | – |
| 1/4 | 1/16 | 6.5794E−01 | 0.77 | 2.5909E−01 | 1.02 | 6.5053E−02 | 0.97 |
| 1/5 | 1/25 | 4.5453E−01 | 0.83 | 1.6255E−01 | 1.04 | 4.1801E−02 | 0.99 |
| 1/6 | 1/36 | 3.2583E−01 | 0.91 | 1.1184E−01 | 1.03 | 2.9099E−02 | 0.99 |
| 1/7 | 1/49 | 2.4464E−01 | 0.93 | 8.1600E−02 | 1.02 | 2.1383E−02 | 1.00 |
| 1/8 | 1/64 | 1.9131E−01 | 0.92 | 6.2354E−02 | 1.01 | 1.6384E−02 | 1.00 |

\( \Omega_s \), and the boundaries of the whole domain are denoted by \( \partial \Omega \). The meshes of the simulation box and boundaries are shown in Fig. 5.

Then, the boundary and initial conditions are described as follows:

\[
\begin{align*}
(D_p \nabla p + \frac{D_p e}{K_B T} p \nabla \phi) \cdot \nu &= 0, \quad \text{on } \Gamma_1 \cup \Gamma_2, \\
(D_n \nabla n - \frac{D_n e}{K_B T} n \nabla \phi) \cdot \nu &= 0, \quad \text{on } \Gamma_1 \cup \Gamma_2, \\
p &= p_\infty, \quad n = n_\infty, \quad \text{on } \Gamma_3, \\
p(\cdot, 0) = p_\infty, \quad n(\cdot, 0) = n_\infty, \\
[\varepsilon \nabla \phi] &= \rho_1, \quad \text{on } \Gamma_1, \\
[\varepsilon \nabla \phi] &= \rho_2, \quad \text{on } \Gamma_2, \\
\phi(x, t) &= -\delta V x / L. \quad \text{on } \partial \Omega,
\end{align*}
\]

where \( \nu \) is the exterior unit normal with direction from solvent region to macromolecule part on the boundary, \( \rho_1 \) and \( \rho_2 \) are the charge densities on the surface of membranes and protein respectively, \( \delta V \) is the voltage difference between the left and right edges of the box along \( x \) direction, \( L \) is the length of the simulation box, and \( p_\infty \) and \( n_\infty \) are the initial-boundary charge densities.

This example uses the similar setup as the model presented in [37]. Suppose \( \Omega = \Omega_s \cup \Omega_m = [x, y] = [-6, 6] \times [-2, 2], \Omega_I = [-2, 2] \times [-2, -0.2] \cup [-2, 2] \times [0.2, 2] \) denotes the solute region, \( \Omega_I = [-2, 2] \times [-2, 2] \times [2, 6] \times [-2, 2] \) represents the solvent region excluding \( \Omega_I \). In our computations, the values of the parameters mentioned above are reported in Table 14.

Table 13  \( H^1 \) error of the two-grid solutions of Algorithm 2 at \( t = 1.0 \) (Example 5.2)

| \( H \) | \( h = H^2 \) | \( \| \Phi_h^* - \phi \|_1 \) | Order | \( \| P_h^{1,*} - p^1 \|_1 \) | Order | \( \| P_h^{2,*} - p^2 \|_1 \) | Order |
|---|---|---|---|---|---|---|---|
| 1/3 | 1/9 | 1.0251E+00 | – | 4.6681E−01 | – | 1.1336E−01 | – |
| 1/4 | 1/16 | 6.5856E−01 | 0.77 | 2.5898E−01 | 1.02 | 6.5053E−02 | 0.97 |
| 1/5 | 1/25 | 4.5518E−01 | 0.83 | 1.6234E−01 | 1.05 | 4.1799E−02 | 0.99 |
| 1/6 | 1/36 | 3.2633E−01 | 0.91 | 1.1155E−01 | 1.03 | 2.9099E−02 | 0.99 |
| 1/7 | 1/49 | 2.4502E−01 | 0.93 | 8.1203E−02 | 1.03 | 2.1379E−02 | 1.00 |
| 1/8 | 1/64 | 1.9210E−01 | 0.91 | 6.1843E−02 | 1.02 | 1.6379E−02 | 1.00 |
In order to compute the finite element solution of (6.1), we first give the weak formulation as follows: Let $V_0 = \{ v | v \in H^1(\Omega_s), v|_{\partial \Omega} = 0 \}$ and $V_{0, \Gamma_3} = \{ v | v \in H^1(\Omega_s), v|_{\Gamma_3} = 0 \}$. Find $\phi \in L^2(0, T; V)$, $p, n \in L^2(0, T; V_{\Gamma_3}) \cap L^\infty(\Omega_T)$ such that

$$\left( \frac{\partial p}{\partial t}, v \right)_{\Omega_s} + \left( D_p \nabla p + D_p \frac{e}{K_BT} p \nabla \phi, \nabla v \right)_{\Omega_s} = 0, \quad \forall v \in V_{0, \Gamma_3}, \quad (6.3)$$

$$\left( \frac{\partial n}{\partial t}, v \right)_{\Omega_s} + \left( D_n \nabla n - D_n \frac{e}{K_BT} n \nabla \phi, \nabla v \right)_{\Omega_s} = 0, \quad \forall v \in V_{0, \Gamma_3}, \quad (6.4)$$

$$(\varepsilon \nabla \phi, \nabla w)_{\Omega_s} = ((p - n)e, w)_{\Omega_s} + ([\varepsilon \nabla \phi], w)_{\Gamma_1 \cup \Gamma_2}, \quad \forall w \in V_0. \quad (6.5)$$

In this computation, the implicit Euler scheme is used for the time discretization with time step $\tau$. We set the final time $T = 1.0$ and choose the time step $\tau = 0.1$. The bulk densities of CsCl solution is 0.1 M and the voltage difference $\delta V = 8$ V. The edge average finite element method (EAFEM) [38] is used in our calculation to solve the density equations. For the charge distributions $p$ and $n$, the piecewise linear element is used on the triangulation of domain $\Omega_s$ and the second-order isoparametric

| Variables | Values |
|-----------|--------|
| Diffusion coefficient: $D_p$ | $2.0561 \times 10^{-9} m^2/s$ |
| Diffusion coefficient: $D_n$ | $2.0321 \times 10^{-9} m^2/s$ |
| Boltzmann energy: $K_BT$ | $4.14 \times 10^{-21} J$ |
| Elementary charge: $e$ | $1.6 \times 10^{-19} C$ |
| Initial density: $p_\infty(n_\infty)$ | $6.02 \times 10^{25}/m^3$ |
| Length of the box: $L$ | $1.0 \times 10^{-10} m$ |
| Permittivity of vacuum: $\varepsilon_0$ | $8.85 \times 10^{-12} C^2/(N \cdot m^2)$ |
Table 15  $L^2$ error of the EAFEM for $P_h, N_h,$ and $\Phi_h$

| $K_h$ | $\| P_h - p \|_{L^2}$ Order | $\| N_h - n \|_{L^2}$ Order | $K_h$ | $\| \Phi_h - \phi \|_{L^2}$ Order |
|------|----------------------------|----------------------------|------|----------------------------|
| 45   | 0.0288 –                  | 10.6052 –                  | 49   | 5.7778 –                  |
| 145  | 0.0443 –0.736            | 8.1898 0.442              | 169  | 4.3420 0.462              |
| 513  | 0.0407 0.134            | 7.2571 0.191              | 625  | 3.2902 0.424              |
| 1921 | 0.0410 –0.011          | 5.3555 0.460              | 2401 | 2.1120 0.659              |
| 7425 | 0.0262 0.662            | 3.3710 0.191              | 9409 | 1.2527 0.765              |
| 29185| 0.0098 1.437           | 1.2608 0.191              | 37249| 0.4599 1.457              |

finite element (cf. [2]) is used for the potential. The finite element approximation $(P_n, N_n, \Phi_n)$ satisfies

$$\left( \frac{P^{n+1}_h - P^n_h}{\tau}, \upsilon_h \right)_{\Omega_s} + \left( D_P \nabla P^{n+1}_h + D_P \frac{e}{K_BT} P^{n+1}_h \nabla \Phi^{n+1}_h, \nabla \upsilon_h \right)_{\Omega_s} = 0,$$  \hspace{1cm} (6.6)

$$\left( \frac{N^{n+1}_h - N^n_h}{\tau}, \upsilon_h \right)_{\Omega_s} + \left( D_n \nabla N^{n+1}_h + D_n \frac{e}{K_BT} N^{n+1}_h \nabla \Phi^{n+1}_h, \nabla \upsilon_h \right)_{\Omega_s} = 0,$$  \hspace{1cm} (6.7)

$$\left( \epsilon \nabla \Phi^{n+1}_h, \nabla \psi_h \right)_{\Omega} = \left( (P^{n+1}_h - N^{n+1}_h)e, \psi_h \right)_{\Omega} + \left( [\epsilon \nabla \Phi^{n+1}_h], \psi_h \right)_{\Gamma_1 \cup \Gamma_2},$$  \hspace{1cm} (6.8)

To illustrate the efficiency and effectiveness of the two-grid method for the ion channel problem, we first obtain the finite element solution of (6.6)–(6.8) by using EAFEM combined with the Gummel iteration. Then, Algorithm 1 is used to solve (6.6)–(6.8) to get the two-grid solution $(P^*_h, N^*_h, \Phi^*_h)$. Both the accuracy of these two solutions and the CPU time cost of the two methods are compared.

All the computations are implemented on quasi-uniform triangular meshes (see, e.g., Fig. 5). To obtain the convergence rate, we refine the initial mesh step by step uniformly in the solvent region $\Omega_s$ and the solute region $\Omega_m$, respectively. Since Example 5.2 is a problem without an analytic solution, we choose the finite element solution with the degrees of freedom $K = 115,713$ as “the exact solution” for the charge distributions $p, n$, and the finite element solution with the degrees of freedom

Table 16  $L^2$ error of the two-grid method for $P^*_h, N^*_h,$ and $\Phi^*_h$

| $K_H$ | $K_h$ | $\| P^*_h - p \|_{L^2}$ Order | $\| N^*_h - n \|_{L^2}$ Order | $K_H$ | $K_h$ | $\| \Phi^*_h - \phi \|_{L^2}$ Order |
|------|------|----------------------------|----------------------------|------|------|----------------------------|
| 16   | 45   | 0.0252 –                  | 10.6769 –                  | 16   | 49   | 5.8649 –                  |
| 45   | 145  | 0.0437 –0.941           | 8.2322 0.444              | 49   | 169  | 4.4391 0.450              |
| 145  | 513  | 0.0402 0.132           | 7.2697 0.197              | 625  | 625  | 3.3548 0.428              |
| 513  | 1921 | 0.0409 –0.026          | 5.3668 0.460              | 2401 | 2401 | 2.1697 0.648              |
| 1921 | 7425 | 0.0266 0.636           | 3.3843 0.682              | 9409 | 9409 | 1.2930 0.757              |
| 7425 | 29185| 0.0103 1.386          | 1.2672 1.435              | 37249| 37249| 0.4930 1.403              |
Table 17 The CPU time (s) of the EAFEM and two-grid method (Example 6.1)

| $K_h$ | EAFEM | Two-grid method (Algorithm 1) |
|-------|-------|-------------------------------|
| 49    | 2.499 | 2.812                         |
| 169   | 4.298 | 4.313                         |
| 625   | 11.830| 9.173                         |
| 2401  | 47.584| 31.581                        |
| 9409  | 211.146| 133.373                      |
| 37249 | 1060.000| 631.299                      |

$K = 148,225$ as “the exact solution” for the potential $\phi$, since they are defined in different domains.

Here, we first define the discrete $L^2$ norm as follows:

$$||e||_{L^2} = \sqrt{\frac{1}{K} \sum_{i=1}^{K} |e_i|^2},$$

where $e = (e_1, e_2, \ldots, e_K)^T$. Denote $K_H$ and $K_h$ are the degrees of freedom on the coarse grid and the fine grid, respectively. The numerical results for the finite element solutions and the two-grid solutions are shown in Tables 15, 16, and 17. First, comparing Table 15 with Table 16, the results show that the two-grid solutions have the similar order of accuracy as the finite element solutions for both the charge distributions $p, n$ and the electrostatic potential $\phi$, which indicates that the two-grid method is efficient for the PNP system in the ion channel. Second, as shown in Table 17, the CPU time cost by Algorithm 1 is much less than that by EAFEM as the degree of freedom becomes large, which indicates the efficiency of Algorithm 1. We also note that the accuracy of order in Table 15 or 16 is not so good as that in Example 5.1, since there are many charges on the interface of membranes which leads to the singularity of the solution for the PNP system in this example. The results can be improved if a better mesh could be used. We shall study the two-grid method on the nonuniform meshes such as the adaptive mesh in our further work.

7 Conclusion

In this paper, we first give the optimal error estimate in $L^2$ norm with linear element for both semi- and fully discrete finite element approximation for the time-dependent Poisson-Nernst-Planck equations. Then, the decoupling two-grid finite element algorithms are proposed for the time-dependent Poisson-Nernst-Planck equations. The optimal error estimates are obtained for the electrostatic potential and the concentrations in $H^1$ norm. The numerical experiments show that the two-grid algorithms remain the same order of accuracy but cost much less computational time compared with the finite element method combined with the Gummel iteration. It is promising to extend this method to more complex PNP models, such as PNP equations for...
three-dimensional ion channel and semiconductor devices, as well as modified PNP equations with size effects.

Acknowledgments The authors would like to thank Dr. Chunshen Feng and Dr. Shixin Xu for their valuable discussions on numerical experiments.

Funding information S. Shu was supported by the China NSF (NSFC 11571293). Y. Yang was supported by the China NSF (NSFC 11561016, NSFC 11661027, NSFC 11561015), Guangxi Colleges and Universities Key Laboratory of Data Analysis and Computation open fund, and the Hunan Key Laboratory for Computation and Simulation in Science and Engineering, Xiangtan University. B. Z. Lu was supported by Science Challenge Program under grant number TZ2016003, National Key Research and Development Program of China (Grant No. 2016YFB0201304), and China NSF (NSFC 21573274, 11771435). R. G. Shen was supported by Postgraduate Scientific Research and Innovation Fund of the Hunan Provincial Education Department (CX2017B268).

References

1. Adams, R.A.: Sobolev Spaces. Academic Press, New York (1975)
2. Ciarlet, P.G.: The Finite Element Method for Elliptic Problems. Elsevier, Amsterdam (1978)
3. Brenner, S.C., Scott, L.R. The Mathematical Theory of Finite Element Methods, 2nd edn. Springer, Berlin (2002)
4. Rannacher, R., Scott, R.: Some optimal error estimates for piecewise linear finite element approximations. Math. Comp. 38, 437–445 (1982)
5. Sun, Y.Z., Sun, P.T., Zheng, B., Lin, G.: Error analysis of finite element method for Poisson-Nernst-Planck equations. J. Comput. Appl. Math. 301, 28–43 (2016)
6. Brands, J., Křížek, M.: Gradient superconvergence on uniform simplicial partitions of polytopes. IMA J. Numer. Anal. 23, 489–505 (2003)
7. Nernst, W.: Die elektromotorische wirksamkeit der io-nen. Z. Phys. Chem. 4, 129–181 (1889)
8. Planck, M.: ÜBer die erregung von electricitüit und wärme in electrolyten. Annalen Der Physik. 275, 161–186 (1890)
9. Song, Y.H., Zhang, Y.J., Bajaj, C.L., Baker, N.A.: Continuum diffusion reaction rate calculations of wild-type and mutant mouse acetylcholinesterase: adaptive finite element analysis. Biophys. J. 87, 1558–1566 (2004)
10. Zhou, Y.C., Lu, B.Z., Huber, G.A., Holst, M.J., McAmmon, J.A.: Continuum simulations of acetylcholine consumption by acetylcholinesterase: a Poisson-Nernst-Planck approach. J. Phys. Chem. 112, 270–275 (2008)
11. Prohl, A., Schmuck, M.: Convergent discretizations for the Nernst-Planck-Poisson system. Numer. Math. 111, 591–630 (2009)
12. Yang, Y., Lu, B.Z.: An error analysis for the finite element approximation to the steady-state Poisson-Nernst-Planck equations. Adv. Appl. Math. Mech. 5, 113–130 (2013)
13. Gao, H.D., He, D.D.: Linearized conservative finite element methods for the Nernst-Planck-Poisson equations. J. Sci. Comput. 72, 1269–1289 (2017)
14. Eisenberg, R., Chen, D.P.: Poisson-Nernst-Planck (PNP) theory of an open ionic channel. Biophys. J. 64, A22 (1993)
15. Lu, B.Z., Holst, M.J., McCammond, J.A., Zhou, Y.C.: Poisson-Nernst-Planck equations for simulating biomolecular diffusion-reaction processes I: finite element solutions. J. Comput. Phys. 229, 6979–6994 (2010)
16. Coalson, R.D., Kurnikova, M.G.: Poisson-Nernst-Planck theory approach to the calculation of current through biological ion channels. IEEE Trans. Nanobiosci. 4, 81–93 (2005)
17. Singer, A., Norbury, J.: A Poisson-Nernst-Planck model for biological ion channels—an asymptotic analysis in a three-dimensional narrow funnel. SIAM J. Appl. Math. 70, 949–968 (2009)
18. Burger, M.: A globally convergent Gummel map for optimal dopant profiling. Math. Models Methods Appl. Sci. 19, 769–786 (2009)
19. Jerome, J.W., Brosowski, B.: Evolution systems in semiconductor device modeling: a cyclic uncoupled line analysis for the Gummel map. Math. Method. Appl. Sci. 9, 455–492 (1987)
20. Lu, B.Z., Zhou, Y.C.: Poisson-Nernst-Planck equations for simulating biomolecular diffusion-reaction processes II: size effects on ionic distributions and diffusion-reaction rates. Biophys J. 100, 2475–2485 (2011)
21. Flavell, A., Machen, M., Eisenberg, R., Kabre, J., Liu, C., Li, X.: A conservative finite difference scheme for Poisson-Nernst-Planck equations. J. Comput. Electron. 13, 235–249 (2014)
22. He, D., Pan, K.: An energy preserving finite difference scheme for the Poisson-Nernst-Planck system. Appl. Math. Comput. 287–288, 214–223 (2016)
23. Liu, H., Wang, Z.: A free energy satisfying finite difference method for Poisson-Nernst-Planck equations. J. Comput. Phys. 268, 363–376 (2014)
24. Mathur, S.R., Murthy, J.Y.: A multigrid method for the Poisson-Nernst-Planck equations. Int. J. Heat Mass Transfer. 52, 4031–4039 (2009)
25. Wu, J., Srinivasan, V., Xu, J., Wang, C.: Newton-Krylov-multigrid algorithms for battery simulation. J. Electrochem. Soc. 149, A1342–A1348 (2002)
26. Jin, J.C., Shu, S., Xu, J.C.: A two-grid discretization method for decoupling systems of partial differential equations. Math. Comput. 75, 1617–1626 (2006)
27. Mu, M., Xu, J.C.: A two-grid method of a mixed Stokes-Darcy model for coupling fluid flow with porous media flow. SIAM J. Numer. Anal. 45, 1801–1813 (2007)
28. Cai, M.C., Mu, M., Xu, J.C.: Numerical solution to a mixed Navier-Stokes/Darcy model by the two-grid approach. SIAM. J. Numer. Anal. 47, 3325–3338 (2009)
29. Xu, J.C.: A new class of iterative methods for nonselfadjoint or indefinite problems. SIAM J. Numer. Anal. 29, 303–319 (1992)
30. Chen, Y., Chen, L., Zhang, X.: Two-grid method for nonlinear parabolic equations by expanded mixed finite element methods. Numer. Meth. Part. D. E 29, 1238–1256 (2013)
31. Myron, L., Iii, B.A.: A two-grid method for mixed finite-element solution of reaction-diffusion equations. Numer. Meth. Part. D. E. 15, 317–332 (1999)
32. Chen, Y., Huang, Y., Yu, D.: A two-grid method for expanded mixed finite-element solution of semilinear reaction-diffusion equations. Int. J. Numer. Meth. Eng. 57, 193–209 (2003)
33. He, Y.N.: Two-level method based on finite element and Crank-Nicolson extrapolation for the time-dependent Navier-Stokes equations. SIAM J. Numer. Anal. 41, 1263–1285 (2003)
34. He, Y.N., Liu, K.M.: A multilevel finite element method in spacetime for the Navier-Stokes problem. Numer. Methods Partial Differential Equations 21, 1052–1078 (2005)
35. Chen, Y., Hu, H.: Two-grid method for miscible displacement problem by mixed finite element methods and mixed finite element method of characteristics. Commun. Comput. Phys. 19, 1503–1528 (2016)
36. Wang, Y., Chen, Y.: A two-grid method for incompressible miscible displacement problems by mixed finite element and Eulerian-Lagrangian localized adjoint methods. J. Math. Anal. Appl. 468, 406–422 (2018)
37. Xu, S., Chen, M., Majd, S., Yue, X., Liu, C.: Modeling and simulating asymmetrical conductance changes in gramicidin pores. Mol. Based Math. Biol. 2, 34–55 (2014)
38. Xu, J., Zikatanov, L.: A monotone finite element scheme for convection-diffusion equations. Math. Comp. 68, 1429–1446 (1999)
39. Nirenberg, L.: An extended interpolation inequality. Ann. Scuola Norm. Sup. Pisa 20, 733–737 (1966)
40. Chen, Y., Wu, L.: Second-order elliptic equations and elliptic systems. Translations of Mathematical Monographs 174 AMS (1998)
41. Araya, R., Venegas, P.: An a posteriori error estimator for an unsteady advection-diffusion-reaction problem. Comput. Math. Appl. 66, 2456–2476 (2014)

Publisher’s note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.
Affiliations

Ruigang Shen¹ · Shi Shu² · Ying Yang³ · Benzhuo Lu⁴

Ruigang Shen
shenuigang@163.com

Shi Shu
shushi@xtu.edu.cn

Benzhuo Lu
bzlu@lsec.cc.ac.cn

¹ School of Mathematics and Computational Science, Xiangtan University, Xiangtan, 411105, People’s Republic of China

² Hunan Key Laboratory for Computation and Simulation in Science and Engineering, Key Laboratory of Intelligent Computing and Information Processing of Ministry of Education, Xiangtan University, Xiangtan, 411105, People’s Republic of China

³ School of Mathematics and Computational Science, Guangxi Colleges and Universities Key Laboratory of Data Analysis and Computation, Guilin University of Electronic Technology, Guilin, 541004, People’s Republic of China

⁴ LSEC, Institute of Computational Mathematics and Scientific/Engineering Computing, the National Center for Mathematics and Interdisciplinary Sciences, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, 100190, People’s Republic of China