

1 **AN EFFECTIVE FINITE ELEMENT ITERATIVE SOLVER FOR A**
2 **POISSON-NERNST-PLANCK ION CHANNEL MODEL WITH**
3 **PERIODIC BOUNDARY CONDITIONS***

4 DEXUAN XIE[†] AND BENZHUO LU[‡]

5 **Abstract.** A system of Poisson-Nernst-Planck equations (PNP) is an important dielectric con-
6 tinuum model for simulating ion transport across biological membrane. In this paper, a PNP ion
7 channel model with periodic boundary value conditions, denoted by PNPic, is presented and solved
8 numerically with an effective finite element iterative method. In particular, the periodic boundary
9 value conditions are used to mimic an infinitely large ion channel membrane, and the PNPic finite
10 element solver includes (1) a PNPic solution decomposition scheme for overcoming the singularity
11 difficulty caused by atomic charges, (2) Slotboom variables for transforming each related Nernst-
12 Planck equation to avoid gradient calculation for any electrostatic potential function, (3) an efficient
13 modified Newton iterative algorithm for solving each related nonlinear finite element equation, and
14 (4) communication operators for carrying out functions operations between different finite element
15 function spaces. This effective PNPic solver is implemented as a software package based on the
16 state-of-the-art finite element library from the FEniCS project and an ion channel mesh generation
17 package developed in Lu's group. Numerical results demonstrate the convergence of the PNPic finite
18 element iterative solver and the performance of the PNPic software package. Moreover, the PNPic
19 model is validated by the cation selectivity property and electric current experimental data of an ion
20 channel protein.

21 **Key words.** Poisson-Nernst-Planck model, finite element method, ion channel protein, periodic
22 boundary conditions

23 **AMS subject classifications.** 92-08, 65N30, 35J66, 65K10

24 **1. Introduction.** Electrodifffusion describes a diffusion process of charged par-
25 ticles in a self-induced electric field (sometimes together with an external electric
26 field), which widely exists in electrochemistry, biology, nanofluidics, and semiconduc-
27 tor physics, etc. A dielectric continuum implicit solvent model defined by Poisson-
28 Nernst-Planck (PNP) equations has been recognized to have significant advantages
29 in computational efficiency and in the calculation of macroscopic properties (e.g.,
30 electric current) for a diffusion process at the mean field level compared to the cor-
31 responding explicit solvent model [45, 13, 8, 26]. In the last two decades, many
32 PNP ion channel models were developed through considering volume-exclusion en-
33 tropy effects [37, 28, 44], hard sphere interactions [4, 17, 18, 32, 44, 43], van der
34 Waals interactions [22], ionic solvation effects [33], electric charge correlations [29],
35 variable dielectric properties [34], and surface energies [51], etc. They were solved
36 numerically by using finite difference methods [14, 15, 26, 27, 54], finite element
37 methods [16, 30, 36, 38, 41, 49], finite volume methods [40], and spectral element
38 methods [21] in either a simplified one-dimensional or a complex three-dimensional
39 setting. Special numerical techniques and implementation strategies were developed
40 to improve the performance of PNP numerical solvers, including a second-order fi-
41 nite difference method [54], a parallel finite element method [49], a potential decom-
42 position technique [36], stabilized techniques [7, 50], energy and mass preservation
43 schemes [14, 15, 20, 27, 41], and mixed finite element methods [16]. Slotboom vari-

*Submitted to SIAM Journal on Scientific Computing, Oct. 31, 2019. Revised on June 2, 2020

[†]Department of Mathematical Sciences, University of Wisconsin-Milwaukee, Milwaukee, WI, 53201-0413, USA (dxie@uwm.edu, <http://www.uwm.edu/~dxie/>). Corresponding author.

[‡]LSEC, National Center for Mathematics and Interdisciplinary Sciences, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China (bzhu@lsec.cc.ac.cn).

44 able transformation [47] and Gummel’s iteration technique [19], developed in the early
45 semiconductor device system simulations, were also used to solve PNP ion channel
46 models [26, 36, 49].

47 Compared with finite difference and finite volume methods, one major advantage
48 of a finite element method is to be able to approximate a complex geometrical shape
49 of an ion channel protein in a high degree of accuracy due to using an irregular
50 tetrahedral mesh. Indeed, well retaining the geometry of a three-dimensional X-ray
51 crystallographic ion channel molecular structure can significantly raise the quality
52 of a PNP ion channel model. But the generation of an irregular tetrahedral mesh
53 that can fit well a complex ion channel molecular surface is highly technical. In the
54 last ten years, Lu’s research team developed an ion channel mesh software package
55 based on the molecular surface triangular mesh package TMSmesh [9, 30, 31]. This
56 mesh package has been released to the public through the cloud computing website
57 <https://www.xyzgate.com>. As a unique ion channel tetrahedral mesh package, it will
58 be applied to the development of a new PNP ion channel finite element solver in this
59 paper.

60 Typically, a PNP ion channel model is based on a box domain that is separated
61 into two solvent compartments by a membrane. A single ion channel protein is then
62 embedded centrally in the membrane and acts as the conduct for transporting ions
63 from one solvent compartment to the other. The membrane normal direction and
64 the ion channel pore are set to coincide with the z -axis direction for the simplicity
65 of implementation. To account for the influence of other ion channel proteins on this
66 single ion channel model, it is natural to set periodic boundary value conditions on
67 the four side surfaces of the box. In fact, periodic boundary techniques have been
68 routinely applied to molecular dynamics for a protein simulation in a box of water
69 molecules. They were also applied to the construction of Poisson-Boltzmann (PB) ion
70 channel models [5, 24] and a finite difference PNP solver [23]. Even so, they have not
71 been considered in any PNP finite element solver yet since it is very difficult to develop
72 a PNP ion channel finite element solver even in the case that does not consider any
73 periodic boundary. In this paper, we attempt to develop an improved PNP ion channel
74 model using the periodic boundary value conditions that are different from those used
75 in [5, 24]. In fact, the periodic boundary conditions in [5] are set on the boundary
76 of a box domain as if one side surface is adjacent to the opposing side surface, while
77 the periodic boundary value conditions in [24] are constructed by setting the mesh
78 nodes of two opposite side surfaces to have the same labeling numbers on the four side
79 surfaces of the box. In our periodic boundary value conditions, each PNP unknown
80 function is set to have the same values on the two opposite side surfaces as done
81 commonly in a periodic boundary value problem.

82 Another major difficulty in solving a PNP ion channel model comes from the
83 solution singularity caused by atomic charges. As shown in [53, Figure 3], such a
84 difficulty cannot be overcome unless all the singularity points can be isolated by a
85 solution decomposition scheme. Two different solution decomposition schemes were
86 reported in [11, 52], respectively, to overcome this difficulty in the numerical solution
87 of a PB model for a protein surrounded by an ionic solvent. We recall that in [11], a PB
88 unknown function, u , which gives an electrostatic potential density of the electric field,
89 is split into three component functions, u^s , u^h , and u^r , within a protein region D_p only,
90 resulting in a Laplace boundary value problem of u^h in D_p and a nonlinear interface
91 boundary value problem of u^r in the box domain Ω . Since D_p is a strongly non-
92 convex domain with a complicated nonsmooth boundary (i.e., a molecular surface),
93 especially for an ion channel protein, solving such a Laplace boundary value problem

94 may cause problems in solution accuracy and solution regularity. The equation of
 95 u^r is also difficult to solve due to involving a jumpily discontinuous flux interface
 96 condition on the interface between D_p and a solvent region D_s . In contrast, in [52],
 97 u is split into three component functions, G , Ψ , and $\tilde{\Phi}$, over the box domain Ω such
 98 that G , Ψ , and $\tilde{\Phi}$ represent the electrostatic potentials induced by the atomic charges,
 99 the potentials from the interfaces and boundary, and the ionic charges from a solvent
 100 region, D_s , respectively. Since G contains all the singularity points of u , both Ψ and
 101 $\tilde{\Phi}$ become smooth within the solvent and solute regions. Note that $u^r = u$ within D_s ,
 102 and $u = G + \Psi + \tilde{\Phi}$. Hence, $\tilde{\Phi} = u^r - G - \Psi$. This shows that $\tilde{\Phi}$ does not involve any
 103 tough part of u^r from G and Ψ so that it is much smoother than u^r . As a result, the
 104 interface boundary value problem of $\tilde{\Phi}$ does not involve any jumpily discontinuous
 105 flux interface condition and can be much easier to solve numerically than that of u^r .
 106 It is this splitting scheme that leads to an efficient PB finite element solver in [52].
 107 The splitting scheme from [11] has been adapted to construct a PNP finite difference
 108 solver in [54] and a PNP finite element solver in [49]. In this paper, we will adapt the
 109 splitting scheme from [52] to construct a new finite element PNP ion channel solver
 110 subject to periodic boundary constraints.

111 In order to reduce numerical complexity and computer memory requirement
 112 sharply, a PNP iterative scheme is often constructed by classic successive relaxation
 113 iterative techniques [42] (or related Gummel's iterative technique [19]). In such a
 114 scheme, however, each Nernst-Planck equation of a PNP system is modified as an
 115 equation that requires calculating the gradient of a given potential function. From
 116 the finite element theory it is known that a gradient calculation may decay one degree
 117 of a finite element solution accuracy [6]. To avoid such a potential numerical prob-
 118 lem, the Slotboom variables, introduced in [47], can be used to transform each related
 119 Nernst-Planck equation as the one that does not involve any gradient of a potential
 120 function, but on the other hand, the related linear Poisson dielectric equation is trans-
 121 formed as a strongly nonlinear equation. Consequently, how to solve such a nonlinear
 122 equation becomes a key step in the development of an effective PNP numerical solver.
 123 Hence, one important task of this paper is to develop new numerical techniques for
 124 solving each related nonlinear equation efficiently.

125 A system of PNP finite element equations involves ionic concentration functions
 126 c_i and an electrostatic potential function u that belong to two different finite element
 127 function spaces, respectively. A communication operator is thus required to carry out
 128 function operations between these two spaces. Currently, such a function operation
 129 issue was simply addressed by extending each c_i from D_s to Ω through setting the
 130 values of c_i to be zero at the mesh nodes outside the solvent region D_s so that both
 131 c_i and u are defined on the same finite element function space based on a mesh
 132 of Ω . But this simple treatment may decay the accuracy of a PNP finite element
 133 system significantly since it actually causes c_i to be nonzero outside D_s on a layer
 134 of tetrahedra along the interface between D_s and a protein-membrane region. Under
 135 periodic boundary constraints, each of these two spaces is modified as a space with
 136 a reduced dimensionality, further increasing the difficulty of dealing with this issue.
 137 In this paper, we will directly construct a finite element function space for each ionic
 138 concentration function c_i based on an irregular tetrahedral mesh of D_s . We then derive
 139 all the required communication operators so that we can well retain the accuracy of
 140 a PNP finite element system in the implementation of function operations between
 141 different function spaces.

142 The rest of the paper is organized as follows. In Section 2, we present a PNP ion

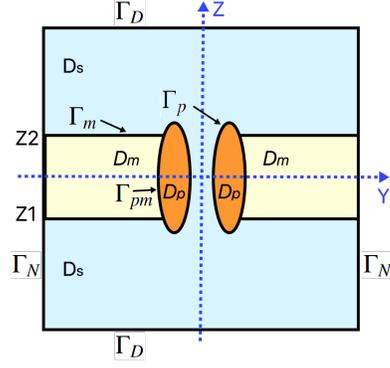


Fig. 1: An illustration of the region partition (2.2) of a rectangular box domain Ω .

143 channel model using periodic boundary value conditions (denoted by PNPic). In Sec-
 144 tion 3, we present a PNPic solution decomposition. In Section 4, we reformulate each
 145 equation of the PNPic solution decomposition into a variational problem. In Section
 146 5, we describe the construction of our PNPic finite element solver. In Section 6, we
 147 report our PNPic software package and numerical results to demonstrate the conver-
 148 gence and performance of our PNPic finite element iterative solver and to validate our
 149 PNPic software package, along with two new formulas for estimating the distribution
 150 of ions and electric current within an ion channel pore. Finally, conclusions are made
 151 in Section 7.

152 2. A PNP ion channel model with periodic boundary value conditions.

153 We construct a sufficiently large open box domain, Ω , by

$$154 \quad (2.1) \quad \Omega = \{(x, y, z) \mid L_{x1} < x < L_{x2}, L_{y1} < y < L_{y2}, L_{z1} < z < L_{z2}\},$$

155 and partition it and its boundary $\partial\Omega$, as illustrated in Figure 1, as follows:

$$156 \quad (2.2) \quad \Omega = D_p \cup D_m \cup D_s \cup \Gamma_m \cup \Gamma_p \cup \Gamma_{pm}, \quad \partial\Omega = \Gamma_D \cup \Gamma_N,$$

157 where $L_{x1}, L_{x2}, L_{y1}, L_{y2}, L_{z1}$, and L_{z2} are real numbers; D_p , D_m , and D_s denote an
 158 ion channel protein region, a membrane region, and a solvent region, respectively; Γ_m
 159 denotes the interface between D_m and D_s , Γ_p the interface between D_p and D_s ; Γ_{pm}
 160 the interface between D_p and D_m ; and Γ_D consists of the bottom and top surfaces
 161 of the box domain Ω and Γ_N the four side surfaces of Ω . In Figure 1, $Z1$ and $Z2$ set
 162 the location of the membrane, D_s contains an ionic solvent with n ionic species, and
 163 D_p hosts an ion channel protein with n_p atoms. We have set the normal direction
 164 of the membrane in the z -axis direction and the z -axis to pass the channel pore.
 165 Moreover, the position vector \mathbf{r}_j and charge number z_j of atom j are given from
 166 a three-dimensional X-ray crystallographic molecular structure of the ion channel
 167 protein. The bulk concentration c_i^b and charge number of species i are also given for
 168 the ionic solvent.

169 Based on the dielectric continuum approach, the three regions D_p , D_m , and D_s
 170 are treated as dielectric media with permittivity constants ϵ_p , ϵ_m , and ϵ_s , respec-
 171 tively. Since D_m consists of a double layer of phospholipid, cholesterol, and glycolipid
 172 molecules whereas D_p is composed of amino acids, ϵ_m may be greater than ϵ_p [48, 24].

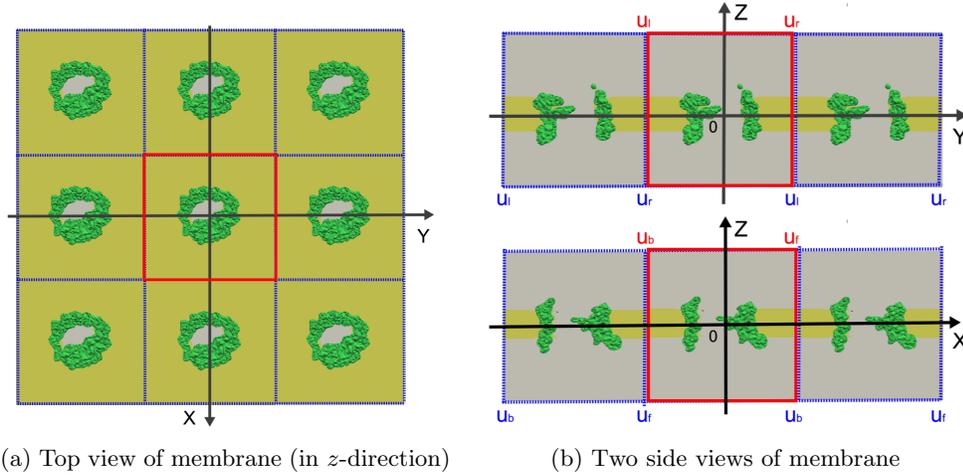


Fig. 2: (a) A membrane embedded with many ion channel proteins of the same type. (b) An illustration of the periodic boundary value conditions of a function u . Here the box domain for simulation is colored in red; u_l, u_r, u_f , and u_b denote the boundary values of u on the left, right, front, and back surfaces of each box domain, respectively; ion channel proteins are colored in green; and the membrane is colored in yellow.

173 We can duplicate the box domain Ω in the four side surface directions, as illus-
 174 trated in Figure 2(a), to produce an infinitely large membrane that is embedded with
 175 ion channel proteins of the same type. Since a dimensionless electrostatic potential
 176 function, u , on each box is identical to each other, it satisfies the periodic boundary
 177 value conditions, $u_l = u_r$ and $u_b = u_f$, as illustrated in Figure 2(b). Here u_l, u_r, u_b ,
 178 and u_f , respectively, denote the values of u on the left, right, back, and front side
 179 surfaces of the simulation box Ω , which is marked in red to differ from its neighboring
 180 boxes (in blue color). Hence, for a function, $u(t, \mathbf{r})$, of time t and spatial variable \mathbf{r}
 181 with $\mathbf{r} = (x, y, z) \in \Omega$, we obtain periodic boundary value conditions as follows:

$$182 \quad (2.3) \quad \begin{aligned} u(t, L_{x1}, y, z) &= u(t, L_{x2}, y, z), & (y, z) \in D_1, \\ u(t, x, L_{y1}, z) &= u(t, x, L_{y2}, z), & (x, z) \in D_2, \end{aligned}$$

184 where $D_1 = \{(y, z) \mid L_{y1} < y < L_{y2}, L_{z1} < z < L_{z2}\}$, $D_2 = \{(x, z) \mid L_{x1} <$
 185 $x < L_{x2}, L_{z1} < z < L_{z2}\}$. Similarly, we can obtain the periodic boundary value
 186 conditions for an ionic concentration function, $c_i(t, \mathbf{r})$ for $\mathbf{r} \in D_s$ and $t \geq 0$, of species
 187 i on the four side surface $\Gamma_N \cap \partial D_s$ of D_s . Here ∂D_s denotes the boundary of D_s .

188 Our PNP ion channel model using the above periodic boundary value conditions,
 189 which is denoted as PNPic, consists of the Poisson equations

$$190 \quad (2.4) \quad \begin{aligned} -\epsilon_p \Delta u(t, \mathbf{r}) &= \alpha \sum_{j=1}^{n_p} z_j \delta_{\mathbf{r}_j}, & \mathbf{r} \in D_p, \\ -\epsilon_m \Delta u(t, \mathbf{r}) &= 0, & \mathbf{r} \in D_m, & -\epsilon_s \Delta u(t, \mathbf{r}) = \beta \sum_{i=1}^n Z_i c_i(t, \mathbf{r}), & \mathbf{r} \in D_s, \end{aligned}$$

192 and the Nernst-Planck equations

$$193 \quad (2.5) \quad \frac{\partial c_i(t, \mathbf{r})}{\partial t} = \nabla \cdot \mathcal{D}_i [\nabla c_i(t, \mathbf{r}) + Z_i c_i(t, \mathbf{r}) \nabla u(t, \mathbf{r})], \quad \mathbf{r} \in D_s, \quad t > 0,$$

194 for $i = 1, 2, \dots, n$, subject to the following interface conditions, initial value conditions,
195 and boundary value conditions:

196 • Interface conditions:

$$197 \quad (2.6) \quad \begin{aligned} u(t, \mathbf{s}^-) &= u(t, \mathbf{s}^+), & \epsilon_p \frac{\partial u(t, \mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} &= \epsilon_s \frac{\partial u(t, \mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})}, & \mathbf{s} \in \Gamma_p, \\ u(t, \mathbf{s}^-) &= u(t, \mathbf{s}^+), & \epsilon_m \frac{\partial u(t, \mathbf{s}^-)}{\partial \mathbf{n}_m(\mathbf{s})} &= \epsilon_s \frac{\partial u(t, \mathbf{s}^+)}{\partial \mathbf{n}_m(\mathbf{s})}, & \mathbf{s} \in \Gamma_m, \\ u(t, \mathbf{s}^-) &= u(t, \mathbf{s}^+), & \epsilon_p \frac{\partial u(t, \mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} &= \epsilon_m \frac{\partial u(t, \mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})}, & \mathbf{s} \in \Gamma_{pm}. \end{aligned}$$

198 • Initial value conditions:

$$199 \quad (2.7) \quad c_i(0, \mathbf{r}) = c_i^0(\mathbf{r}), \quad \mathbf{r} \in D_s, \quad i = 1, 2, \dots, n.$$

200 • Dirichlet boundary value conditions on the bottom and top surfaces:

$$201 \quad (2.8) \quad u(t, \mathbf{s}) = g(\mathbf{s}), \quad \mathbf{s} \in \Gamma_D, \quad c_i(t, \mathbf{s}) = g_i(\mathbf{s}), \quad \mathbf{s} \in \Gamma_D.$$

202 • Periodic boundary value conditions on the four side surfaces:

$$203 \quad (2.9) \quad u(t, \mathbf{s}) \text{ is periodic for } \mathbf{s} \in \Gamma_N, \quad c_i(t, \mathbf{s}) \text{ is periodic for } \mathbf{s} \in \Gamma_N \cap \partial D_s.$$

204 • Robin boundary value conditions on the interface $\Gamma_p \cup \Gamma_m$:

$$205 \quad (2.10) \quad \frac{\partial c_i(t, \mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} + Z_i c_i(t, \mathbf{s}) \frac{\partial u(t, \mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} = 0, \quad \mathbf{s} \in \Gamma_p \cup \Gamma_m.$$

206 Here $\delta_{\mathbf{r}_j}$ is the Dirac delta distribution at \mathbf{r}_j ; α and β are defined by

$$207 \quad (2.11) \quad \alpha = \frac{10^{10} e_c^2}{\epsilon_0 k_B T}, \quad \beta = \frac{N_A e_c^2}{10^{17} \epsilon_0 k_B T};$$

208 \mathbf{n}_p , \mathbf{n}_m , and \mathbf{n}_s are the unit outward normal directions of D_p , D_m , and D_s , respec-
209 tively; g and g_i are boundary value functions; c_i^0 is an initial value function; and \mathcal{D}_i
210 denote a diffusion coefficient function of the i -th ionic species. Here ϵ_0 is the permit-
211 tivity of the vacuum, e_c is the elementary charge, k_B is the Boltzmann constant, T
212 is the absolute temperature, and N_A is the Avogadro number, which estimates the
213 number of ions per mole. Note that we have measured ionic concentration function c_i
214 in moles per liter (mol/L), time t in picoseconds (ps), spatial length in angstroms (\AA),
215 and diffusion function \mathcal{D}_i in units $\text{\AA}^2/\text{ps}$. In physics, the Robin boundary condition
216 (2.10) reflects the fact that none of ionic particles cross the interface $\Gamma_p \cup \Gamma_m$ to enter
217 the protein and membrane regions D_p and D_m ; the boundary value functions g and g_i
218 can be properly selected, as shown in (6.1) in Section 6, to mimic an external voltage
219 across the membrane.

220 When u is known, an electrostatic potential function, Φ , is found by

$$221 \quad \Phi(t, \mathbf{r}) = \frac{k_B T}{e_c} u(t, \mathbf{r}), \quad \mathbf{r} \in \Omega, \quad t > 0,$$

222 in volts. Due to the above relation, the dimensionless potential u can be viewed as
223 an electrostatic potential with the constant $k_B T/e_c$ as its physical unit.

224 At $T = 298.15$, the values of α , β , and $\frac{k_B T}{e_c}$ can be estimated as

$$225 \quad \alpha \approx 7042.9399, \quad \beta \approx 4.2413, \quad k_B T/e_c \approx 0.0257 \text{ volts.}$$

226 Thus, $u = 1$ is about 0.0257 volts or 25.7 millivolts (mV).

227 **3. PNPic solution decomposition.** To overcome the singularity difficulty
 228 caused by atomic charges, we split the electrostatic potential function u into three
 229 component functions, G , Ψ , and $\tilde{\Phi}$, such that

$$230 \quad (3.1) \quad u(t, \mathbf{r}) = G(\mathbf{r}) + \Psi(\mathbf{r}) + \tilde{\Phi}(t, \mathbf{r}), \quad \mathbf{r} \in \Omega, \quad t \geq 0,$$

231 where G is a potential induced by atomic charges from the protein region D_p , Ψ is
 232 a potential induced by potentials from interface and boundary, and $\tilde{\Phi}$ is a potential
 233 induced by ionic charges from the solvent region D_s .

234 In particular, G can be found in the analytical expression

$$235 \quad (3.2) \quad G(\mathbf{r}) = \frac{\alpha}{4\pi\epsilon_p} \sum_{j=1}^{n_p} \frac{z_j}{|\mathbf{r} - \mathbf{r}_j|}$$

236 as a solution of the Poisson equation in the whole space \mathbb{R}^3 :

$$237 \quad (3.3) \quad -\epsilon_p \Delta G(\mathbf{r}) = \alpha \sum_{j=1}^{n_p} z_j \delta_{\mathbf{r}_j}, \quad \mathbf{r} \in \mathbb{R}^3.$$

238 Since G and Ψ are independent of ionic concentrations c_i , they can be calculated
 239 prior to the calculation of c_i and $\tilde{\Phi}$ so that we can treat them as two given functions
 240 during an iterative process of searching for c_i and $\tilde{\Phi}$. With this observation, we
 241 construct a linear interface boundary value problem of Ψ such that it collects all the
 242 jumpily discontinuous interface conditions produced by the splitting formula (3.1) and
 243 the related inhomogeneous boundary value conditions for the purpose of making the
 244 equation of $\tilde{\Phi}$ as simple as possible. Clearly, $\tilde{\Phi}$ is periodic on the four side surfaces of
 245 the box domain Ω . To get its periodic boundary value conditions, we set Ψ to satisfy
 246 the Dirichlet boundary value condition $\Psi + G = 0$ on Γ_N . In this way, we derive a
 247 linear interface boundary value problem of Ψ ,

$$248 \quad (3.4) \quad \left\{ \begin{array}{ll} \Delta \Psi(\mathbf{r}) = 0, & \mathbf{r} \in D_m \cup D_p \cup D_s, \\ \Psi(\mathbf{s}^-) = \Psi(\mathbf{s}^+), & \epsilon_p \frac{\partial \Psi(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_s \frac{\partial \Psi(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})} + (\epsilon_s - \epsilon_p) \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})}, \quad \mathbf{s} \in \Gamma_p, \\ \Psi(\mathbf{s}^-) = \Psi(\mathbf{s}^+), & \epsilon_m \frac{\partial \Psi(\mathbf{s}^-)}{\partial \mathbf{n}_m(\mathbf{s})} = \epsilon_s \frac{\partial \Psi(\mathbf{s}^+)}{\partial \mathbf{n}_m(\mathbf{s})} + (\epsilon_s - \epsilon_m) \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_m(\mathbf{s})}, \quad \mathbf{s} \in \Gamma_m, \\ \Psi(\mathbf{s}^-) = \Psi(\mathbf{s}^+), & \epsilon_p \frac{\partial \Psi(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_m \frac{\partial \Psi(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})} + (\epsilon_m - \epsilon_p) \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})}, \quad \mathbf{s} \in \Gamma_{pm}, \\ \Psi(\mathbf{s}) = g(\mathbf{s}) - G(\mathbf{s}), & \mathbf{s} \in \Gamma_D, \\ \Psi(\mathbf{s}) = -G(\mathbf{s}), & \mathbf{s} \in \Gamma_N, \end{array} \right.$$

249 and a linear interface boundary value problem of $\tilde{\Phi}$, which has continuous interface
 250 conditions, a homogeneous Dirichlet boundary condition, and periodic boundary con-
 251 ditions, as follows:

$$252 \quad (3.5) \quad \left\{ \begin{array}{ll} \Delta \tilde{\Phi}(t, \mathbf{r}) = 0, & \mathbf{r} \in D_m \cup D_p, \\ -\epsilon_s \Delta \tilde{\Phi}(t, \mathbf{r}) = \beta \sum_{i=1}^n Z_i c_i(t, \mathbf{r}), & \mathbf{r} \in D_s, \\ \tilde{\Phi}(t, \mathbf{s}^+) = \tilde{\Phi}(t, \mathbf{s}^-), & \epsilon_s \frac{\partial \tilde{\Phi}(t, \mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_p \frac{\partial \tilde{\Phi}(t, \mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})}, \quad \mathbf{s} \in \Gamma_p, \\ \tilde{\Phi}(t, \mathbf{s}^+) = \tilde{\Phi}(t, \mathbf{s}^-), & \epsilon_s \frac{\partial \tilde{\Phi}(t, \mathbf{s}^+)}{\partial \mathbf{n}_m(\mathbf{s})} = \epsilon_m \frac{\partial \tilde{\Phi}(t, \mathbf{s}^-)}{\partial \mathbf{n}_m(\mathbf{s})}, \quad \mathbf{s} \in \Gamma_m, \\ \tilde{\Phi}(t, \mathbf{s}^-) = \tilde{\Phi}(t, \mathbf{s}^+), & \epsilon_p \frac{\partial \tilde{\Phi}(t, \mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_m \frac{\partial \tilde{\Phi}(t, \mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})}, \quad \mathbf{s} \in \Gamma_{pm}, \\ \tilde{\Phi}(t, \mathbf{s}) = 0, & \mathbf{s} \in \Gamma_D, \\ \tilde{\Phi}(t, \mathbf{s}) \text{ is periodic,} & \mathbf{s} \in \Gamma_N. \end{array} \right.$$

253 Here $\frac{\partial G(\mathbf{s})}{\partial \mathbf{n}(\mathbf{s})} = \nabla G(\mathbf{s}) \cdot \mathbf{n}(\mathbf{s})$ with ∇G being given by

$$254 \quad (3.6) \quad \nabla G(\mathbf{s}) = -\frac{\alpha}{4\pi\epsilon_p} \sum_{j=1}^{n_p} z_j \frac{(\mathbf{s} - \mathbf{r}_j)}{|\mathbf{s} - \mathbf{r}_j|^3}.$$

255 It can be easy to validate that the sum of G with Ψ and $\tilde{\Phi}$ gives the solution of
 256 the Poisson ion channel interface boundary value problem (2.4). Clearly, G contains
 257 all the singular points of u . Thus, both Ψ and $\tilde{\Phi}$ are smooth within D_p, D_m , or D_s .

258 Using the given G and Ψ , we can treat each Nernst-Planck equation of (2.5) as
 259 an equation of c_i and $\tilde{\Phi}$,

$$260 \quad (3.7) \quad \frac{\partial c_i(t, \mathbf{r})}{\partial t} = \nabla \cdot \mathcal{D}_i \left[\nabla c_i + Z_i c_i \mathbf{w} + Z_i c_i \nabla \tilde{\Phi}(t, \mathbf{r}) \right], \quad \mathbf{r} \in D_s, \quad t > 0,$$

261 for $i = 1, 2, \dots, n$. Here $\mathbf{w} = \nabla G(\mathbf{r}) + \nabla \Psi(\mathbf{r})$, which has been calculated.

262 Consequently, a combination of (3.7) with (3.5) gives a system of equations for
 263 solving $\tilde{\Phi}$ and c_i for $i = 1, 2, \dots, n$, together with the initial and boundary value
 264 conditions (2.7)–(2.10). Note that this new system is much easier to solve numerically
 265 than the original PNPic system since it avoids the solution singularity difficulties
 266 induced by atomic charges, and $\tilde{\Phi}$ is much smoother than u because the tough parts
 267 G and Ψ of u have been removed from the construction of $\tilde{\Phi}$.

268 In the remaining part of this paper, we only consider the steady state of PNPic.
 269 Since in the steady state, c_i , u , and $\tilde{\Phi}$ become independent of time t , the system for
 270 $\tilde{\Phi}$ and c_i is simplified as n steady Nernst-Planck boundary value problems,

$$271 \quad (3.8) \quad \begin{cases} \nabla \cdot \mathcal{D}_i(\mathbf{r}) \left[\nabla c_i(\mathbf{r}) + Z_i c_i(\mathbf{r}) \mathbf{w}(\mathbf{r}) + Z_i c_i(\mathbf{r}) \nabla \tilde{\Phi}(\mathbf{r}) \right] = 0, & \mathbf{r} \in D_s, \\ \frac{\partial c_i(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} + Z_i c_i(\mathbf{s}) \frac{\partial u(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} = 0, & \mathbf{s} \in \Gamma_p \cup \Gamma_m, \\ c_i(\mathbf{s}) = g_i(\mathbf{s}), & \mathbf{s} \in \Gamma_D, \\ \tilde{\Phi}(\mathbf{s}) \text{ is periodic,} & \mathbf{s} \in \Gamma_N, \end{cases}$$

272 for $i = 1, 2, \dots, n$, plus one interface boundary value problem,

$$273 \quad (3.9) \quad \begin{cases} \Delta \tilde{\Phi}(\mathbf{r}) = 0, & \mathbf{r} \in D_m \cup D_p, \\ -\epsilon_s \Delta \tilde{\Phi}(\mathbf{r}) = \beta \sum_{i=1}^n Z_i c_i(\mathbf{r}), & \mathbf{r} \in D_s, \\ \tilde{\Phi}(\mathbf{s}^+) = \tilde{\Phi}(\mathbf{s}^-), \quad \epsilon_s \frac{\partial \tilde{\Phi}(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_p \frac{\partial \tilde{\Phi}(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})}, & \mathbf{s} \in \Gamma_p, \\ \tilde{\Phi}(\mathbf{s}^+) = \tilde{\Phi}(\mathbf{s}^-), \quad \epsilon_s \frac{\partial \tilde{\Phi}(\mathbf{s}^+)}{\partial \mathbf{n}_m(\mathbf{s})} = \epsilon_m \frac{\partial \tilde{\Phi}(\mathbf{s}^-)}{\partial \mathbf{n}_m(\mathbf{s})}, & \mathbf{s} \in \Gamma_m, \\ \tilde{\Phi}(\mathbf{s}^-) = \tilde{\Phi}(\mathbf{s}^+), \quad \epsilon_p \frac{\partial \tilde{\Phi}(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_m \frac{\partial \tilde{\Phi}(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})}, & \mathbf{s} \in \Gamma_{pm}, \\ \tilde{\Phi}(\mathbf{s}) = 0, & \mathbf{s} \in \Gamma_D, \\ \tilde{\Phi}(\mathbf{s}) \text{ is periodic,} & \mathbf{s} \in \Gamma_N. \end{cases}$$

274 When $\tilde{\Phi}$ is known, we obtain u by the formula

$$275 \quad u(\mathbf{r}) = G(\mathbf{r}) + \Psi(\mathbf{r}) + \tilde{\Phi}(\mathbf{r}), \quad \mathbf{r} \in \Omega.$$

276 **4. Variational formulations.** One key step in the development of a finite
 277 element algorithm for solving the PNPic model is to derive the variational forms of
 278 interface boundary value problems (3.4) and (3.9) and Nernst-Planck system (3.8). In
 279 this section, we obtain these forms and give them detailed proofs since their derivations

280 are nontrivial due to the complicated interface conditions and periodic boundary
 281 value conditions. We then obtain a variational form of the system of (3.8) and (3.9).
 282 Furthermore, we simplify the variational form of (3.4) into a variational problem
 283 without involving any surface integral when the membrane permittivity constant ϵ_m
 284 is set to be equal to the protein permittivity constant ϵ_p .

285 Let $H^1(\Omega)$ and $H^1(D_s)$ be the Sobolev function spaces based on the box domain
 286 Ω and solvent region D_s , respectively [1]. We define their subspaces, $U, U_0, H_0^1(\Omega), V$,
 287 and V_0 , as follows:

$$288 \quad (4.1) \quad U = \{u \in H^1(\Omega) \mid u \text{ is periodic on } \Gamma_N\}, \quad U_0 = \{u \in U \mid u = 0 \text{ on } \Gamma_D\},$$

$$289 \quad H_0^1(\Omega) = \{v \in H^1(\Omega) \mid v = 0 \text{ on } \partial\Omega\},$$

$$291 \quad (4.2) \quad V = \{v \in H^1(D_s) \mid v \text{ is periodic on } \Gamma_N \cap \partial D_s\}, \quad V_0 = \{v \in V \mid v = 0 \text{ on } \Gamma_D\}.$$

292 We first present a variational form of the interface boundary value problem (3.9)
 293 in Theorem 4.1.

294 **THEOREM 4.1.** *The linear interface boundary value problem (3.9) has the follow-*
 295 *ing variational form:*

$$296 \quad (4.3) \quad \text{Find } \tilde{\Phi} \in U_0 \text{ such that } a(\tilde{\Phi}, v) = \beta \sum_{i=1}^n Z_i \int_{D_s} c_i v d\mathbf{r} \quad \forall v \in U_0,$$

297 where U_0 is defined in (4.1) and $a(\tilde{\Phi}, v)$ is defined by

$$298 \quad (4.4) \quad a(\tilde{\Phi}, v) = \epsilon_p \int_{D_p} \nabla \tilde{\Phi} \cdot \nabla v d\mathbf{r} + \epsilon_m \int_{D_m} \nabla \tilde{\Phi} \cdot \nabla v d\mathbf{r} + \epsilon_s \int_{D_s} \nabla \tilde{\Phi} \cdot \nabla v d\mathbf{r}.$$

299 *Proof.* We multiply the first and second equations of (3.9) with a test function
 300 $v \in U_0$; integrate it over D_p, D_m , and D_s , respectively; and then add them together
 301 to get

$$302 \quad -\epsilon_p \int_{D_p} \Delta \tilde{\Phi}(\mathbf{r}) v(\mathbf{r}) d\mathbf{r} - \epsilon_m \int_{D_m} \Delta \tilde{\Phi}(\mathbf{r}) v(\mathbf{r}) d\mathbf{r} - \epsilon_s \int_{D_s} \Delta \tilde{\Phi}(\mathbf{r}) v(\mathbf{r}) d\mathbf{r}$$

$$303 \quad = \beta \sum_{i=1}^n Z_i \int_{D_s} c_i(\mathbf{r}) v(\mathbf{r}) d\mathbf{r}.$$

304 Using Green's first identity, we can rewrite the above equation as

$$305 \quad (4.5) \quad \epsilon_p \int_{D_p} \nabla \tilde{\Phi}(\mathbf{r}) \cdot \nabla v(\mathbf{r}) d\mathbf{r} + \epsilon_m \int_{D_m} \nabla \tilde{\Phi}(\mathbf{r}) \cdot \nabla v(\mathbf{r}) d\mathbf{r} + \epsilon_s \int_{D_s} \nabla \tilde{\Phi}(\mathbf{r}) \cdot \nabla v(\mathbf{r}) d\mathbf{r}$$

$$= \epsilon_p \int_{\partial D_p} \frac{\partial \tilde{\Phi}(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} + \epsilon_m \int_{\partial D_m} \frac{\partial \tilde{\Phi}(\mathbf{s})}{\partial \mathbf{n}_m(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} + \epsilon_s \int_{\partial D_s} \frac{\partial \tilde{\Phi}(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} v(\mathbf{s}) d\mathbf{s}$$

$$+ \beta \sum_{i=1}^n Z_i \int_{D_s} c_i(\mathbf{r}) v(\mathbf{r}) d\mathbf{r},$$

306 where $\partial D_p, \partial D_m$, and ∂D_s denote the boundaries of D_p, D_m , and D_s and $\mathbf{n}_p, \mathbf{n}_m$,
 307 and \mathbf{n}_s denote the unit outward normal vectors of D_p, D_m , and D_s , respectively. Note
 308 that the normal vectors have the relations

$$309 \quad \mathbf{n}_s = -\mathbf{n}_p \text{ on } \Gamma_p, \quad \mathbf{n}_s = -\mathbf{n}_m \text{ on } \Gamma_m, \quad \mathbf{n}_m = -\mathbf{n}_p \text{ on } \Gamma_{pm},$$

$$310 \quad \mathbf{n}_m = \mathbf{n}_b \text{ on } \Gamma_N \cap \partial D_m, \quad \mathbf{n}_s = \mathbf{n}_b \text{ on } \Gamma_N \cap \partial D_s,$$

311 and the boundaries ∂D_p , ∂D_m , and ∂D_s can be expressed as

$$312 \quad \partial D_p = \Gamma_p \cup \Gamma_{pm}, \quad \partial D_m = \Gamma_m \cup (\Gamma_N \cap \partial D_m) \cup \Gamma_{pm}, \quad \partial D_s = \Gamma_m \cup \Gamma_p \cup \Gamma_D \cup (\Gamma_N \cap \partial D_s).$$

313 Hence, by $v = 0$ on Γ_D , the three surface integrals of (4.5) can be simplified as follows:

$$\begin{aligned} & \int_{\partial D_p} \frac{\partial \tilde{\Phi}(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} = \int_{\Gamma_p} \frac{\partial \tilde{\Phi}(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} + \int_{\Gamma_{pm}} \frac{\partial \tilde{\Phi}(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s}, \\ & \int_{\partial D_m} \frac{\partial \tilde{\Phi}(\mathbf{s})}{\partial \mathbf{n}_m(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} = \int_{\Gamma_m} \frac{\partial \tilde{\Phi}(\mathbf{s}^-)}{\partial \mathbf{n}_m(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} - \int_{\Gamma_{pm}} \frac{\partial \tilde{\Phi}(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} \\ 314 & \quad + \int_{\Gamma_N \cap \partial D_m} \frac{\partial \tilde{\Phi}(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) d\mathbf{s}, \\ & \int_{\partial D_s} \frac{\partial \tilde{\Phi}(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} = - \int_{\Gamma_m} \frac{\partial \tilde{\Phi}(\mathbf{s}^+)}{\partial \mathbf{n}_m(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} - \int_{\Gamma_p} \frac{\partial \tilde{\Phi}(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} \\ & \quad + \int_{\Gamma_N \cap \partial D_s} \frac{\partial \tilde{\Phi}(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) d\mathbf{s}, \end{aligned}$$

315 where \mathbf{n}_b denotes the unit outward normal vector of the box domain Ω . Applying the
316 above expressions and the interface conditions of (3.9)–(4.5), we obtain

$$\begin{aligned} 317 \quad a(\tilde{\Phi}, v) &= \beta \sum_{i=1}^n Z_i \int_{D_s} c_i v d\mathbf{r} + \epsilon_m \int_{\Gamma_N \cap \partial D_m} \frac{\partial \tilde{\Phi}(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} \\ & \quad + \epsilon_s \int_{\Gamma_N \cap \partial D_s} \frac{\partial \tilde{\Phi}(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) d\mathbf{s}. \end{aligned}$$

318 Clearly, the normal vectors $\mathbf{n}_b = (\pm 1, 0, 0)$ and $(0, \pm 1, 0)$ on the four side surfaces
319 of Γ_N . Thus, the surface integral $\int_{\Gamma_N \cap \partial D_s} \frac{\partial \tilde{\Phi}(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) d\mathbf{s}$ can be written as

$$\begin{aligned} 320 & \int_{\Gamma_N \cap \partial D_s} \frac{\partial \tilde{\Phi}(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} \\ 321 &= \int_{Lz1}^{Z1} \int_{Ly1}^{Ly2} \left[\frac{\partial \tilde{\Phi}(Lx2, y, z)}{\partial x} v(Lx2, y, z) - \frac{\partial \tilde{\Phi}(Lx1, y, z)}{\partial x} v(Lx1, y, z) \right] dy dz \\ 322 & \quad + \int_{Z2}^{Lz2} \int_{Ly1}^{Ly2} \left[\frac{\partial \tilde{\Phi}(Lx2, y, z)}{\partial x} v(Lx2, y, z) - \frac{\partial \tilde{\Phi}(Lx1, y, z)}{\partial x} v(Lx1, y, z) \right] dy dz \\ 323 & \quad + \int_{Lz1}^{Z1} \int_{Lx1}^{Lx2} \left[\frac{\partial \tilde{\Phi}(x, Ly2, z)}{\partial y} v(x, Ly2, z) - \frac{\partial \tilde{\Phi}(x, Ly1, z)}{\partial y} v(x, Ly1, z) \right] dx dz \\ 324 & \quad + \int_{Z2}^{Lz2} \int_{Lx1}^{Lx2} \left[\frac{\partial \tilde{\Phi}(x, Ly2, z)}{\partial y} v(x, Ly2, z) - \frac{\partial \tilde{\Phi}(x, Ly1, z)}{\partial y} v(x, Ly1, z) \right] dx dz, \end{aligned}$$

325 where $Z1$ and $Z2$ denote the starting and ending numbers of the membrane in the Z -
326 axis direction, respectively. Since each test function v satisfies the periodic boundary

327 conditions, the above expression becomes

$$\begin{aligned}
328 \quad (4.6) \quad & \int_{\Gamma_N \cap \partial D_s} \frac{\partial \tilde{\Phi}(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) d\mathbf{s}. \\
329 \quad & = \int_{Lz1}^{Z1} \int_{Ly1}^{Ly2} \left[\frac{\partial \tilde{\Phi}(Lx2, y, z)}{\partial x} - \frac{\partial \tilde{\Phi}(Lx1, y, z)}{\partial x} \right] v(Lx1, y, z) dy dz \\
330 \quad & + \int_{Z2}^{Lz2} \int_{Ly1}^{Ly2} \left[\frac{\partial \tilde{\Phi}(Lx2, y, z)}{\partial x} - \frac{\partial \tilde{\Phi}(Lx1, y, z)}{\partial x} \right] v(Lx1, y, z) dy dz \\
331 \quad (4.7) \quad & + \int_{Lz1}^{Z1} \int_{Lx1}^{Lx2} \left[\frac{\partial \tilde{\Phi}(x, Ly2, z)}{\partial y} - \frac{\partial \tilde{\Phi}(x, Ly1, z)}{\partial y} \right] v(x, Ly1, z) dx dz \\
332 \quad & + \int_{Z2}^{Lz2} \int_{Lx1}^{Lx2} \left[\frac{\partial \tilde{\Phi}(x, Ly2, z)}{\partial y} - \frac{\partial \tilde{\Phi}(x, Ly1, z)}{\partial y} \right] v(x, Ly1, z) dx dz.
\end{aligned}$$

333 From the periodicity of $\tilde{\Phi}$ on Γ_N , it can imply that the partial derivatives $\frac{\partial \tilde{\Phi}}{\partial x}$ and
334 $\frac{\partial \tilde{\Phi}}{\partial y}$ satisfy the following periodic boundary conditions:

$$\begin{aligned}
335 \quad & \frac{\partial \tilde{\Phi}(Lx1, y, z)}{\partial x} = \frac{\partial \tilde{\Phi}(Lx2, y, z)}{\partial x} \quad \forall (y, z) \in D_1, \\
336 \quad & \frac{\partial \tilde{\Phi}(x, Ly1, z)}{\partial y} = \frac{\partial \tilde{\Phi}(x, Ly2, z)}{\partial y} \quad \forall (x, z) \in D_2.
\end{aligned}$$

337 Applying the above equations to (4.6) immediately gives

$$338 \quad (4.8) \quad \int_{\Gamma_N \cap \partial D_s} \frac{\partial \tilde{\Phi}(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} = 0.$$

339 Similarly, we can prove that $\int_{\Gamma_N \cap \partial D_m} \frac{\partial \tilde{\Phi}(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} = 0$. This completes the proof. \square

340 We next present a variational formulation of the Nernst-Planck system (3.8) in
341 Theorem 4.2.

342 **THEOREM 4.2.** *The system (3.8) of n steady Nernst-Planck equations has the*
343 *following variational form: Find $c_i \in V$ satisfying $c_i = g_i$ on Γ_D such that*

$$344 \quad (4.9) \quad \int_{D_s} \mathcal{D}_i(\mathbf{r}) (\nabla c_i(\mathbf{r}) + Z_i c_i(\mathbf{r}) \nabla u(\mathbf{r})) \nabla v_i(\mathbf{r}) d\mathbf{r} = 0 \quad \forall v_i \in V_0, \quad i = 1, 2, \dots, n,$$

345 where V and V_0 are given in (4.2).

346 *Proof.* We multiply a test function $v_i \in V_0$ on both sides of the first equation of
347 (3.8), integrate on the solvent region D_s , and use Green's first identity to get

$$348 \quad (4.10) \quad \int_{\partial D_s} \mathcal{D}_i \left(\frac{\partial c_i(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} + Z_i c_i \frac{\partial u(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} \right) v_i(\mathbf{s}) d\mathbf{s} - \int_{D_s} \mathcal{D}_i (\nabla c_i + Z_i c_i \nabla u) \nabla v_i d\mathbf{r} = 0.$$

349 Since the boundary ∂D_s of D_s can be expressed as

$$350 \quad \partial D_s = \Gamma_m \cup \Gamma_p \cup \Gamma_D \cup (\Gamma_N \cap \partial D_s),$$

351 we can use the second equation of (3.8) and $v_i = 0$ on Γ_D to get

$$\begin{aligned}
352 \quad \int_{\partial D_s} \mathcal{D}_i \left(\frac{\partial c_i(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} + Z_i c_i \frac{\partial u(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} \right) v_i(\mathbf{s}) d\mathbf{s} &= \mathcal{D}_i \int_{\Gamma_N \cap \partial D_s} \frac{\partial c_i(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v_i(\mathbf{s}) d\mathbf{s} \\
353 \quad &+ \mathcal{D}_i Z_i \int_{\Gamma_N \cap \partial D_s} c_i \frac{\partial u(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v_i(\mathbf{s}) d\mathbf{s} \quad \forall v_i \in U_0,
\end{aligned}$$

354 where we have used the fact that $\mathbf{n}_s = \mathbf{n}_b$ on Γ_N and \mathcal{D}_i is a constant on the side
355 surface $\Gamma_N \cap \partial D_s$. Clearly, from the periodicities of c_i and u , it can imply the peri-
356 odicities of the partial derivatives $\frac{\partial c_i}{\partial x}$, $\frac{\partial c_i}{\partial y}$, $\frac{\partial u}{\partial x}$, and $\frac{\partial u}{\partial y}$ on the side surfaces $\Gamma_N \cap \partial D_s$
357 and Γ_N , respectively. Similarly to what is done in the proof of (4.8), we can use the
358 periodicities of c_i , v_i , $\frac{\partial c_i}{\partial x}$, and $\frac{\partial c_i}{\partial y}$ on $\Gamma_N \cap \partial D_s$ and the periodicities of u , $\frac{\partial u}{\partial x}$, and
359 $\frac{\partial u}{\partial y}$ on Γ_N to get

$$360 \quad \int_{\Gamma_N \cap \partial D_s} \frac{\partial c_i(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} v_i(\mathbf{s}) d\mathbf{s} = 0, \quad \int_{\Gamma_N \cap \partial D_s} c_i \frac{\partial u(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} v_i(\mathbf{s}) d\mathbf{s} = 0.$$

361 Thus, we obtain

$$362 \quad \int_{\partial D_s} \mathcal{D}_i \left(\frac{\partial c_i(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} + Z_i c_i \frac{\partial u(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} \right) v_i(\mathbf{s}) d\mathbf{s} = 0.$$

363 Applying the above equation to (4.10) gives the weak form (4.9). This completes the
364 proof. \square

365 Furthermore, a variational form of the interface boundary value problem (3.4) is
366 presented in Theorem 4.3.

367 **THEOREM 4.3.** *The linear interface boundary value problem (3.4) has the follow-*
368 *ing variational form: Find $\Psi \in H^1(\Omega)$ satisfying $\Psi = g - G$ on Γ_D and $\Psi = -G$ on*
369 *Γ_N such that*

$$\begin{aligned}
370 \quad (4.11) \quad a(\Psi, v) &= (\epsilon_s - \epsilon_p) \int_{\Gamma_p} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} + (\epsilon_s - \epsilon_m) \int_{\Gamma_m} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_m(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} \\
371 \quad &+ (\epsilon_m - \epsilon_p) \int_{\Gamma_{pm}} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} \quad \forall v \in H_0^1(\Omega),
\end{aligned}$$

372 where \mathbf{n}_m and \mathbf{n}_p denote the unit outward normal vectors of D_m and D_p , respectively,
373 and $a(\cdot, \cdot)$ is defined in (4.4).

374 *Proof.* We multiply the first equation of (3.4) with a test function $v \in H_0^1(\Omega)$;
375 integrate it over D_p , D_m , and D_s , respectively; and then add them together to get

$$376 \quad \epsilon_p \int_{D_p} \Delta \Psi(\mathbf{r}) v(\mathbf{r}) d\mathbf{r} + \epsilon_m \int_{D_m} \Delta \Psi(\mathbf{r}) v(\mathbf{r}) d\mathbf{r} + \epsilon_s \int_{D_s} \Delta \Psi(\mathbf{r}) v(\mathbf{r}) d\mathbf{r} = 0.$$

377 Applying Green's first identity to each of the above three integrals, we can get

$$\begin{aligned}
378 \quad (4.12) \quad &\epsilon_p \int_{D_p} \nabla \Psi(\mathbf{r}) \cdot \nabla v(\mathbf{r}) d\mathbf{r} + \epsilon_m \int_{D_m} \nabla \Psi(\mathbf{r}) \cdot \nabla v(\mathbf{r}) d\mathbf{r} + \epsilon_s \int_{D_s} \nabla \Psi(\mathbf{r}) \cdot \nabla v(\mathbf{r}) d\mathbf{r} \\
&= \epsilon_p \int_{\partial D_p} \frac{\partial \Psi(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} + \epsilon_m \int_{\partial D_m} \frac{\partial \Psi(\mathbf{s})}{\partial \mathbf{n}_m(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} + \epsilon_s \int_{\partial D_s} \frac{\partial \Psi(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} v(\mathbf{s}) d\mathbf{s}.
\end{aligned}$$

379 By $v = 0$ on $\Gamma_D \cup \Gamma_N$ (i.e., the boundary $\partial\Omega$), the three surface integrals of (4.12)
 380 can be simplified as follows:

$$\begin{aligned}
 & \int_{\partial D_p} \frac{\partial \Psi(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} = \int_{\Gamma_p} \frac{\partial \Psi(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} + \int_{\Gamma_{pm}} \frac{\partial \Psi(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s}, \\
 381 \quad & \int_{\partial D_m} \frac{\partial \Psi(\mathbf{s})}{\partial \mathbf{n}_m(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} = \int_{\Gamma_m} \frac{\partial \Psi(\mathbf{s}^-)}{\partial \mathbf{n}_m(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} - \int_{\Gamma_{pm}} \frac{\partial \Psi(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s}, \\
 & \int_{\partial D_s} \frac{\partial \Psi(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} = - \int_{\Gamma_m} \frac{\partial \Psi(\mathbf{s}^+)}{\partial \mathbf{n}_m(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} - \int_{\Gamma_p} \frac{\partial \Psi(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s}.
 \end{aligned}$$

382 Applying the above expressions and the interface conditions of (3.4)–(4.12), we obtain
 383 (4.11). This completes the proof. \square

384 In PNP ion channel simulations, it is often to set $\epsilon_m = \epsilon_p$. In this case, the weak
 385 form (4.11) can be simplified as follows: Find $\Psi \in H^1(\Omega)$ satisfying $\Psi = g - G$ on Γ_D
 386 and $\Psi = -G$ on Γ_N such that

$$387 \quad (4.13) \quad a(\Psi, v) = (\epsilon_s - \epsilon_p) \int_{\Gamma} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} \quad \forall v \in H_0^1(\Omega),$$

388 where \mathbf{n} denotes the unit outward normal direction of the protein-membrane region
 389 $D_{pm} = D_p \cup D_m \cup \Gamma_{pm}$, $\Gamma = \Gamma_m \cup \Gamma_p$, which is the interface between D_{pm} and D_s ,
 390 and $a(u, v)$ is simplified as follows:

$$391 \quad (4.14) \quad a(u, v) = \epsilon_p \int_{D_{pm}} \nabla u \cdot \nabla v d\mathbf{r} + \epsilon_s \int_{D_s} \nabla \tilde{\Phi} \cdot \nabla v d\mathbf{r}.$$

392 **THEOREM 4.4.** *Let the gradient vector ∇G be given in (3.6). If $\epsilon_m = \epsilon_p$ and*
 393 *$\Gamma = \Gamma_m \cup \Gamma_p$, then*

$$394 \quad (4.15) \quad \int_{\Gamma} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} = - \int_{D_s} \nabla G(\mathbf{r}) \cdot \nabla v(\mathbf{r}) d\mathbf{r}.$$

395 *Proof.* Using Green's first identity, $\Delta G = 0$ in D_s , $\partial D_s = \Gamma \cup \Gamma_D \cup (\Gamma_N \cap \partial D_s)$,
 396 and $v = 0$ on $\Gamma_D \cup \Gamma_N$, we get

$$\begin{aligned}
 397 \quad 0 &= \int_{D_s} \Delta G v d\mathbf{r} = \int_{\partial D_s} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} - \int_{D_s} \nabla G(\mathbf{r}) \cdot \nabla v(\mathbf{r}) d\mathbf{r} \\
 &= \int_{\Gamma} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} - \int_{D_s} \nabla G(\mathbf{r}) \cdot \nabla v(\mathbf{r}) d\mathbf{r}.
 \end{aligned}$$

398 Since $\mathbf{n}_s = -\mathbf{n}$ on Γ , from the above expression, it gives the identity (4.15). This
 399 completes the proof. \square

400 Applying (4.15) to the variational problem (4.13), we obtain another variational
 401 form of Ψ as follows: Find $\Psi \in H^1(\Omega)$ satisfying $\Psi = g - G$ on Γ_D and $\Psi = -G$ on
 402 Γ_N such that

$$403 \quad (4.16) \quad a(\Psi, v) = (\epsilon_p - \epsilon_s) \int_{D_s} \nabla G(\mathbf{r}) \cdot \nabla v(\mathbf{r}) d\mathbf{r} \quad \forall v \in H_0^1(\Omega).$$

404 The above weak form simplifies the numerical calculation of Ψ since it does not involve
 405 any surface integral. A surface integral can be more difficult to calculate than a

406 corresponding volume integral since a geometrical shape of the interface Γ is very
407 complicated in an ion channel simulation.

408 In summary, we have obtained a variational form of the system of (3.8) and (3.9)
409 as follows: Find $\tilde{\Phi} \in V_0$ and $c_i \in U$ with $c_i = g_i$ on Γ_D for $i = 1, 2, \dots, n$ such that

$$410 \quad (4.17) \quad \begin{cases} \int_{D_s} \mathcal{D}_i \left[\nabla c_i + Z_i c_i (\mathbf{w} + \nabla \tilde{\Phi}) \right] \nabla v_i d\mathbf{r} = 0 & \forall v_i \in U_0 \text{ for } i = 1, 2, \dots, n, \\ a(\tilde{\Phi}, v) - \beta \sum_{i=1}^n Z_i \int_{D_s} c_i v d\mathbf{r} = 0 & \forall v \in V_0, \end{cases}$$

411 where $\mathbf{w} = \nabla G(\mathbf{r}) + \nabla \Psi(\mathbf{r})$ with ∇G being given in (3.6) and Ψ is a solution of (4.11)
412 (or (4.16) in the case that $\epsilon_m = \epsilon_p$).

413 **5. A PNPic finite element solver.** Let Ω_h be an interface fitted irregular
414 tetrahedral mesh of a box domain Ω . We use Ω_h to construct two linear Lagrange finite
415 element function spaces, \mathcal{U}_1 and \mathcal{U}_2 , as two finite-dimensional subspaces of the function
416 spaces $H^1(\Omega)$ and U , respectively. From Ω_h , we extract an irregular tetrahedral mesh,
417 $D_{s,h}$, of D_s to construct two linear Lagrange finite element function spaces, \mathcal{V}_1 and \mathcal{V}_2 ,
418 as two finite dimensional subspaces of the function spaces $H^1(D_s)$ and V , respectively.
419 We also define three subspaces, $\mathcal{U}_{1,0}$, $\mathcal{U}_{2,0}$, and $\mathcal{V}_{2,0}$, by

$$420 \quad \mathcal{U}_{1,0} = \{u \in \mathcal{U}_1 \mid u = 0 \text{ on } \partial\Omega\}, \quad \mathcal{U}_{2,0} = \{u \in \mathcal{U}_2 \mid u = 0 \text{ on } \Gamma_D\},$$

$$421 \quad \mathcal{V}_{2,0} = \{v \in \mathcal{V}_2 \mid v = 0 \text{ on } \Gamma_D\}.$$

422 Here U and V have been defined in (4.1) and (4.2), respectively.

423 Since Ψ , $\tilde{\Phi}$, and c_i belong to three different finite element spaces, \mathcal{U}_1 , \mathcal{U}_2 , and \mathcal{V}_2 ,
424 respectively, we construct three communication operators P_1 , P_2 , and P_3 by

$$425 \quad P_1 : \mathcal{U}_2 \rightarrow \mathcal{U}_1, \quad P_2 : \mathcal{U}_1 \rightarrow \mathcal{V}_1, \quad P_3 : \mathcal{V}_2 \rightarrow \mathcal{U}_2.$$

426 For example, we map $\tilde{\Phi}$ from the periodic boundary constrained finite element space
427 \mathcal{U}_2 onto the original finite element space \mathcal{U}_1 by linear operator P_1 to complete the
428 addition of $\tilde{\Phi}$ with G and Ψ . Using these linear operators, we approximate the system
429 (4.17) by a system of finite element equations as follows: Find $\tilde{\Phi} \in \mathcal{U}_{2,0}$ and $c_i \in \mathcal{V}_2$
430 satisfying $c_i = g_i$ on Γ_D for $i = 1, 2, \dots, n$ such that

$$431 \quad (5.1) \quad \begin{cases} \int_{D_s} \mathcal{D}_i \left[\nabla c_i + Z_i c_i \nabla P_2(G + \Psi + P_1 \tilde{\Phi}) \right] \nabla v_i d\mathbf{r} = 0 & \forall v_i \in \mathcal{V}_{2,0} \\ & \text{for } i = 1, 2, \dots, n, \\ a(\tilde{\Phi}, v) - \beta \sum_{j=1}^n Z_j \int_{D_s} P_3 c_j v d\mathbf{r} = 0 & \forall v \in \mathcal{U}_{2,0}, \end{cases}$$

432 where G is given in (3.2) and Ψ has been calculated through solving a finite element
433 approximation of the variational problem (4.11). For example, in the case that $\epsilon_m =$
434 ϵ_p , the finite element equation for computing Ψ is given as follows: Find $\Psi \in \mathcal{U}_1$
435 satisfying $\Psi = g - G$ on Γ_D and $\Psi = -G$ on Γ_N such that

$$436 \quad (5.2) \quad a(\Psi, v) = (\epsilon_p - \epsilon_s) \int_{D_s} \nabla G(\mathbf{r}) \cdot \nabla v(\mathbf{r}) d\mathbf{r} \quad \forall v \in \mathcal{U}_{1,0},$$

437 where the bilinear form $a(\cdot, \cdot)$ is given in (4.14).

438 We recall that the Slotboom variable transformation is defined by

$$439 \quad (5.3) \quad c_i = e^{-Z_i u} \bar{c}_i, \quad i = 1, 2, \dots, n,$$

440 where \bar{c}_i denotes the i -th Slotboom variable [47]. From the periodicity of u and c_i on
 441 $\Gamma_N \cap \partial D_s$, it can imply that \bar{c}_i is periodic on $\Gamma_N \cap \partial D_s$. Using (5.3), we can get

$$442 \quad (5.4) \quad \nabla c_i + Z_i c_i \nabla u = e^{-Z_i u} \nabla \bar{c}_i, \quad i = 1, 2, \dots, n,$$

443 and then transform the system (5.1) into a new system of $\tilde{\Phi}$ and \bar{c}_i as follows: Find
 444 $\tilde{\Phi} \in \mathcal{U}_{2,0}$ and $\bar{c}_i \in \mathcal{V}_2$ satisfying $\bar{c}_i = \bar{g}_i$ on Γ_D for $i = 1, 2, \dots, n$ such that

$$445 \quad (5.5) \quad \begin{cases} \int_{D_s} \mathcal{D}_i e^{-Z_i P_2(G+\Psi+P_1\tilde{\Phi})} \nabla \bar{c}_i \nabla v_i d\mathbf{r} = 0 & \forall v_i \in \mathcal{V}_{2,0} \\ & \text{for } i = 1, 2, \dots, n, \\ a(\tilde{\Phi}, v) - \beta \sum_{i=1}^n Z_i \int_{D_s} e^{-Z_i(G+\Psi+P_1\tilde{\Phi})} P_3 \bar{c}_i v d\mathbf{r} = 0 & \forall v \in \mathcal{U}_{2,0}, \end{cases}$$

446 where $\bar{g}_i = e^{Z_i g} g_i$, which is derived from the boundary value conditions $u = g$ and
 447 $c_i = g_i$ on Γ_D . After finding \bar{c}_i , we recover c_i using (5.3) for $i = 1, 2, \dots, n$.

448 We now construct a relaxation iterative scheme for solving the nonlinear finite
 449 element system (5.5) using the classic successive relaxation iterative techniques [42].
 450 Let $\tilde{\Phi}^k$ and \bar{c}_i^k denote the k th iterative approximations to $\tilde{\Phi}$ and \bar{c}_i , respectively. We
 451 define them for $k = 0, 1, 2, \dots$ by

$$452 \quad (5.6) \quad \bar{c}_i^{k+1} = \bar{c}_i^k + \omega(\bar{p}_i - \bar{c}_i^k), \quad i = 1, 2, \dots, n,$$

$$453 \quad (5.7) \quad \tilde{\Phi}^{k+1} = \tilde{\Phi}^k + \omega(\bar{q} - \tilde{\Phi}^k),$$

454 where $\bar{p}_i \in \mathcal{V}_2$ satisfying $\bar{p}_i = \bar{g}_i$ on Γ_D such that

$$455 \quad (5.8) \quad \int_{D_s} \mathcal{D}_i e^{-Z_i P_2(G+\Psi+P_1\tilde{\Phi}^k)} \nabla \bar{p}_i \nabla v_i d\mathbf{r} = 0 \quad \forall v_i \in \mathcal{V}_{2,0}, \quad i = 1, 2, \dots, n,$$

456 \bar{q} is a solution of the nonlinear variational problem: Find $\bar{q} \in \mathcal{U}_{2,0}$ such that

$$457 \quad (5.9) \quad a(\bar{q}, v) - \beta \sum_{i=1}^n Z_i \int_{D_s} e^{-Z_i(G+\Psi+P_1\tilde{\Phi}^k)} P_3 \bar{c}_i^{k+1} v d\mathbf{r} = 0 \quad \forall v \in \mathcal{U}_{2,0},$$

458 \bar{c}_i^0 and $\tilde{\Phi}^0$ are given initial iterates, and ω is a relaxation parameter between 0 and 1.

459 By default, we set that $\bar{c}_i^0 = c_i^b$, and $\tilde{\Phi}^0$ is a solution of the variational problem:
 460 Find $\tilde{\Phi}^0 \in \mathcal{U}_{2,0}$ such that

$$461 \quad (5.10) \quad a(\tilde{\Phi}^0, v) - \beta \sum_{i=1}^n Z_i c_i^b \int_{D_s} e^{-Z_i(G+\Psi+P_1\tilde{\Phi}^0)} v d\mathbf{r} = 0 \quad \forall v \in \mathcal{U}_{2,0}.$$

462 We stop this iteration process whenever the following criteria hold:

$$463 \quad (5.11) \quad \|\tilde{\Phi}^{k+1} - \tilde{\Phi}^k\| < \epsilon \quad \text{and} \quad \max_{1 \leq i \leq n} \|\bar{c}_i^{k+1} - \bar{c}_i^k\| < \epsilon.$$

464 where ϵ is a tolerance (e.g. $\epsilon = 10^{-5}$) and $\|\cdot\|$ denotes the L_2 norm.

465 In order to solve the nonlinear variational problem (5.9) in the k th iteration, we
 466 construct an iterative sequence, $\{q_k^j\}$, by

$$467 \quad (5.12) \quad q_k^{j+1} = q_k^j + \xi_k^j \quad j = 0, 1, 2, \dots,$$

468 where $q_k^0 = \tilde{\Phi}^k$ and ξ_k^j is a solution of the variational problem: Find $\xi_k^j \in \mathcal{U}_{2,0}$ such
469 that

$$470 \quad (5.13) \quad a(\xi_k^j, v) + \beta \int_{D_s} \sum_{i=1}^n Z_i^2 P_3 \bar{c}_i^{k+1} e^{-Z_i(G+\Psi+P_1 q_k^j)} \xi_k^j v d\mathbf{r}$$

$$471 \quad = \beta \int_{D_s} \sum_{i=1}^n Z_i e^{-Z_i(G+\Psi+P_1 q_k^j)} P_3 \bar{c}_i^{k+1} v d\mathbf{r} - a(q_k^j, v) \quad \forall v \in \mathcal{U}_{2,0}.$$

472 To get the initial iterate $\tilde{\Phi}^0$, we construct an iterative sequence, $\{q^j\}$, for solving
473 the nonlinear variational problem (5.10) by

$$474 \quad (5.14) \quad q^{j+1} = q^j + \xi^j, \quad j = 0, 1, 2, \dots,$$

475 where initial iterate q^0 is set as a solution of a linearized problem of (5.10),

$$476 \quad (5.15) \quad a(\phi, v) + \beta \sum_{i=1}^n Z_i^2 c_i^b \int_{D_s} \phi v d\mathbf{r} = -\beta \sum_{i=1}^n Z_i^2 c_i^b \int_{D_s} (G + \Psi) v d\mathbf{r} \quad \forall v \in \mathcal{U}_{2,0},$$

477 and ξ^j is a solution of the linear variational problem: Find $\xi^j \in \mathcal{U}_{2,0}$ such that

$$478 \quad (5.16) \quad a(\xi^j, v) + \beta \int_{D_s} \sum_{i=1}^n Z_i^2 c_i^b e^{-Z_i(G+\Psi+P_1 q_k^j)} \xi^j v d\mathbf{r}$$

$$479 \quad = \beta \int_{D_s} \sum_{i=1}^n Z_i c_i^b e^{-Z_i(G+\Psi+P_1 q_k^j)} v d\mathbf{r} - a(q_k^j, v) \quad \forall v \in \mathcal{U}_{2,0}.$$

480 In (5.15), we have used the electroneutrality condition $\sum_{i=1}^n Z_i c_i^b = 0$.

481 In the iterative process of (5.12), we use the iteration stopping criterion:

$$482 \quad (5.17) \quad \text{either } j > \text{Ite_max} \quad \text{or} \quad \|q_k^{j+1} - q_k^j\| < \tau,$$

483 where Ite_max denotes the maximum allowable number of iterations and τ is a tol-
484 erance. In calculation, we set Ite_max = 10 and $\tau = 10^{-5}$ by default. Similarly, we
485 stop the iterative process of (5.14) whenever

$$486 \quad (5.18) \quad \text{either } j > \text{Ite_max} \quad \text{or} \quad \|q^{j+1} - q^j\| < \tau.$$

487 For clarity, we summarize our relaxation iterative scheme in Algorithm 1.

488 **Algorithm 1.** *Our finite element relaxation iterative scheme for solving the*
489 *steady PNPic system of (3.8) and (3.9) for the electrostatic potential u and ionic*
490 *concentrations c_i can be implemented in five steps:*

491 **Step 1.** *Initialization: Calculate G by (3.2); calculate Ψ by solving a finite element*
492 *approximation problem of (4.11) (or (5.2) when $\epsilon_m = \epsilon_p$); set the initial*
493 *iterates $\bar{c}_i^0 = c_i^b$ for $i = 1, 2, \dots, n$; calculate $\tilde{\Phi}^0$ as a solution of the nonlinear*
494 *problem (5.10) by the iterative scheme (5.14); and set $k = 0$.*

495 **Step 2.** *Define \bar{c}_i^{k+1} by (5.6) with \bar{p}_i being a solution of the linear variational problem*
496 *(5.8) for $i = 1, 2, \dots, n$.*

497 **Step 3.** *Define $\tilde{\Phi}^{k+1}$ by (5.7) with \bar{q} being an iterate q_k^j of the iterative scheme (5.12)*
498 *for solving the nonlinear variational problem (5.9) satisfying the iteration stop*
499 *rule (5.17).*

500 **Step 4.** Check the convergence: If the iteration stop criteria of (5.11) hold, go to
 501 Step 5 with $\bar{c}_i = \bar{c}_i^{k+1}$ for $i = 1, 2, \dots, n$ and $\tilde{\Phi} = \tilde{\Phi}^{k+1}$; otherwise, increase k
 502 by 1, and go back to Step 2.

503 **Step 5.** Define the steady PNPic solution: $u = G + \Psi + \tilde{\Phi}$ and $c_i = e^{-Z_i u} \bar{c}_i$ for
 504 $i = 1, 2, \dots, n$.

505 **Remark 1.** The iterative scheme defined in (5.12) is a Newton iterative method
 506 for minimizing the functional

$$507 \quad J(v) = \frac{1}{2}a(v, v) + \beta \int_{D_s} \sum_{i=1}^n \bar{c}_i^{k+1} e^{-Z_i(G+\Psi+P_1 v)} d\mathbf{r}.$$

508 It can be shown that the minimizer of J gives a solution of the nonlinear variational
 509 problem (5.9). This statement is true for the iterative scheme defined in (5.14) if
 510 Slotboom iterates \bar{c}_i^{k+1} of J are replaced by the bulk concentrations c_i^b .

511 **Remark 2.** The iterative scheme of (5.14) is actually a finite element Newton
 512 iterative scheme for solving a PB ion channel model using the periodic boundary con-
 513 ditions given in (2.3). That is, this PB ion channel model is defined by the equations
 514 of (3.3), (3.4), and (3.5) using $c_i = c_i^b e^{-Z_i u}$ for $i = 1, 2, \dots, n$. It can be shown that
 515 the solution u of this PB ion channel model can be constructed by

$$516 \quad (5.19) \quad u = G + \Psi + \tilde{\Phi}^{PB},$$

517 where $\tilde{\Phi}^{PB}$ denotes a solution of the nonlinear variational problem (5.10). This PB
 518 ion channel model and finite element solver are different from those reported in [24].

519 **6. Numerical results.** We implemented Algorithm 1 in Python as a software
 520 package based on the state-of-the-art finite element library from the FEniCS project
 521 [35] and the PB finite element solver program package reported in [52]. We used the
 522 ion channel finite element mesh program package developed by Lu's research group
 523 [10, 30, 31] to generate interface fitted irregular tetrahedral meshes for a box domain Ω
 524 as illustrated in Figure 1. From a mesh of Ω , we extracted the meshes of solvent region
 525 D_s , membrane region D_m , and protein region D_p , denoted by $D_{s,h}$, $D_{m,h}$, and $D_{p,h}$,
 526 respectively. We then used these meshes to define the finite element function spaces
 527 \mathcal{U}_1 and \mathcal{V}_1 . Furthermore, we modified \mathcal{U}_1 and \mathcal{V}_1 as the finite element function spaces
 528 \mathcal{U}_2 and \mathcal{V}_2 using the periodic boundary value conditions. In this software package, we
 529 set boundary value functions $g_i(\mathbf{r})$ and $g(\mathbf{r})$ with $\mathbf{r} = (x, y, z)$ for ionic concentration
 530 functions c_i and electrostatic potential function u , respectively, as follows:

$$531 \quad (6.1) \quad g_i(\mathbf{r}) = \begin{cases} c_i^b & \text{at } z = L_{z1} \text{ (bottom),} \\ c_i^b & \text{at } z = L_{z2} \text{ (top),} \end{cases} \quad g(\mathbf{r}) = \begin{cases} u_b & \text{at } z = L_{z1} \text{ (bottom),} \\ u_t & \text{at } z = L_{z2} \text{ (top),} \end{cases}$$

532 where c_i^b is a bulk concentration of species i and the difference between electrostatic
 533 potential values u_b and u_t can be regarded as a voltage across the membrane. We
 534 also followed what was done in [49, Equation (27)] to define the diffusion coefficient
 535 function $\mathcal{D}_i(\mathbf{r})$ with $\mathbf{r} = (x, y, z)$ by

$$536 \quad \mathcal{D}_i(\mathbf{r}) = \begin{cases} D_{i,b}, & z < Z1 \text{ or } z > Z2 \text{ (bulk part),} \\ D_{i,c} + (D_{i,c} - D_{i,b})f_t(\mathbf{r}), & Z2 - \eta \leq z \leq Z2 \text{ (top buffer part),} \\ D_{i,c}, & Z1 + \eta \leq z \leq Z2 - \eta \text{ (channel pore),} \\ D_{i,c} + (D_{i,c} - D_{i,b})f_b(\mathbf{r}), & Z1 \leq z \leq Z1 + \eta \text{ (bottom buffer part),} \end{cases}$$

537 where $D_{i,b}$ and $D_{i,c}$ are the diffusion constants of species i for the bulk and channel
 538 pore regions, respectively; f_b and f_t are the interpolation functions given in [49,

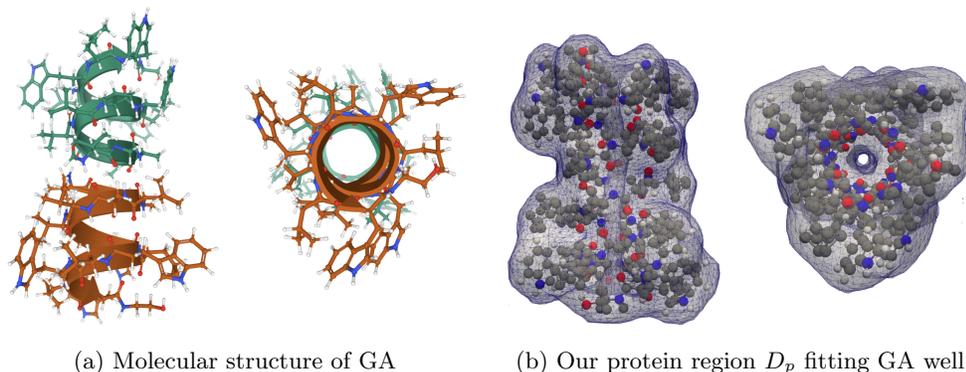


Fig. 3: (a) Two views of GA (PDB identification code 1MAG) depicted in sticks for the molecular structure and cartoons for the two helical subunits. (b) Two views of our protein region D_p , along with the GA molecular structure depicted in balls for oxygen atoms (in red), nitrogen atoms (in blue), and carbon atoms (in gray).

Equation (27)] such that each diffusion function is sufficiently smooth in the solvent region D_s ; and η is a parameter for adjusting the buffering region size. By default, each finite element equation of (5.8) and (5.13) is solved, approximately, by the generalized minimal residual method using incomplete LU preconditioning with the absolute and relative residual errors being less than 10^{-6} .

We did numerical tests on an ion channel protein, a gramicidin A (GA), in a solution of anions Cl^- and cations K^+ to demonstrate the convergence of our nonlinear relaxation iterative scheme and the computer performance of our program package. Here the charge numbers $Z_1 = 1$ and $Z_2 = -1$. The GA channel is a small protein 0.4 nm in diameter and 2.5 nm in length composed of symmetric dimers of two β -helical subunits. Two views of its molecular structure are given in Figure 3(a).

GA is an antibiotic peptide produced by *Bacillus brevis* and has been extensively studied in experiments and various modelings [3, 46]. Due to the cation-selective property and the simplicity in molecular structure compared with other ion channel proteins [2], the GA channel has been a typical molecular force probe to explore how changes in bilayer properties alter protein function [39]. With an X-ray crystallographic molecular structure [25] and the experimental data [12], the GA channel is often selected to construct numerical tests for validating PNP ion channel models [49, 54].

We downloaded the GA molecular structure file 1mag.pdb from the protein data bank (PDB, <https://www.rcsb.org>). We then derived its PQR file that contains the data missed in the PDB file, such as the hydrogen atoms, the atomic charge numbers, and the atomic radii. The total number n_p of atoms is 280. We rotated the ion channel and assembled it with a membrane, as illustrated in Figure 1, for a rectangular box Ω of dimensions $40 \times 40 \times 60$ defined by $L_{x1} = -20.323$, $L_{x2} = 19.677$, $L_{y1} = -20.0$, $L_{y2} = 20.0$, $L_{z1} = -33.421$, $L_{z2} = 26.579$, $Z_1 = -11$, and $Z_2 = 6$ for a membrane thickness of 17 Å. The meshes Ω_h and $D_{s,h}$ have 24686 and 15828 mesh points, respectively. We display them in Figure 4(a,b) to show their geometrical complexities. Because of the periodic boundary conditions, the dimensions 24686 and 15828 of \mathcal{U}_1 and \mathcal{V}_1 were reduced to the dimensions 22541 and 14203 of \mathcal{U}_2 and \mathcal{V}_2 , respectively.

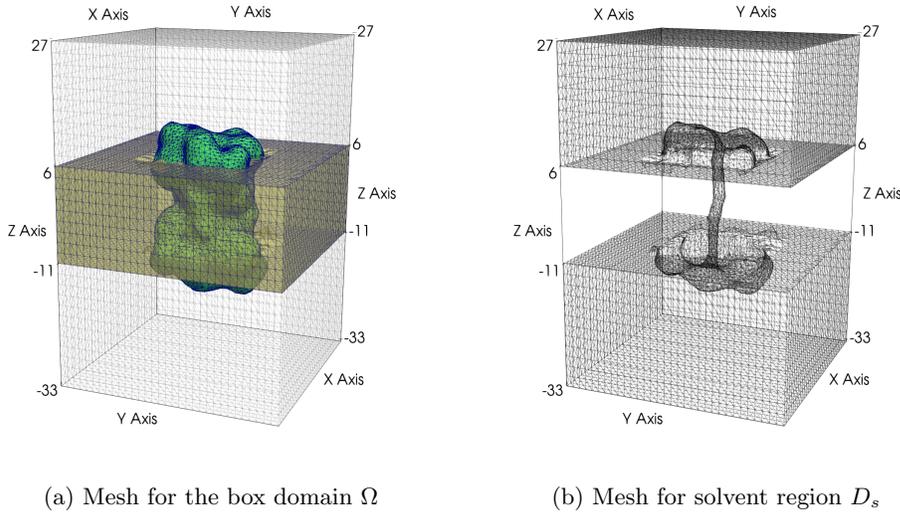


Fig. 4: The interface fitted irregular tetrahedral meshes of the box domain Ω and solvent region D_s for the ion channel protein Gramicidin A (PDB identification code 1MAG) for our numerical tests. Here the meshes of the membrane region D_m and protein region D_p are colored in yellow and green, respectively, for clarity.

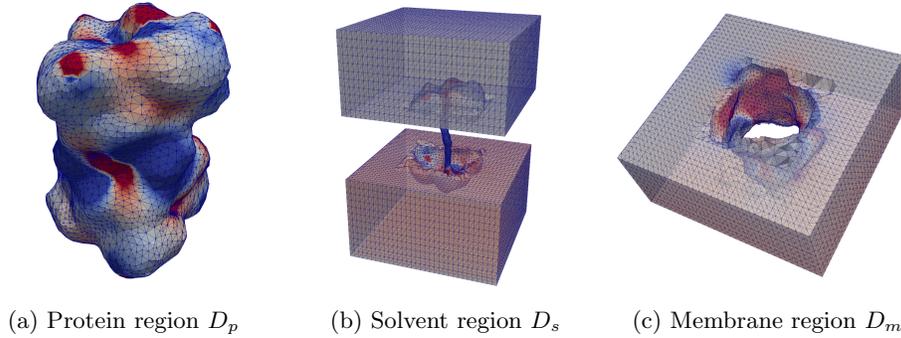


Fig. 5: The electrostatic potential u produced by the PNPic finite element solver on the triangular surface meshes of the protein, solvent, and membrane regions D_p , D_s , and D_m in color mapping from blue for -2 to red for 2 .

569 In the numerical tests, we set $\epsilon_s = 80$, $\epsilon_p = 2$, and $\epsilon_m = 2$; $D_{1,b} = 0.196$,
 570 $D_{1,c} = 0.0196$ (for K^+ ions), $D_{2,b} = 0.203$, and $D_{2,c} = 0.0203$ (for Cl^- ions); and
 571 $\eta = 3$ (for the diffusion coefficient function $\mathcal{D}_i(\mathbf{r})$). Since $\epsilon_m = \epsilon_p$, we calculated Ψ by
 572 solving the finite element variational problem (5.2). All the numerical tests were done
 573 on our iMac computer with one 4.2-GHz Intel core i7 processor and 64 GB memory.

574 One important feature of our PNPic software package is to be able to visualize

Table 1: Parameter values for the boundary value functions g_i for $i = 1, 2$ and g defined in (6.1) and the performance of our PNPic finite element solver.

u_b	u_t	c_i^b	Iteration number	CPU time (seconds)
-1	1	0.5	15	86.10
-1	1	0.1	15	85.41
-3	3	0.5	24	140.86

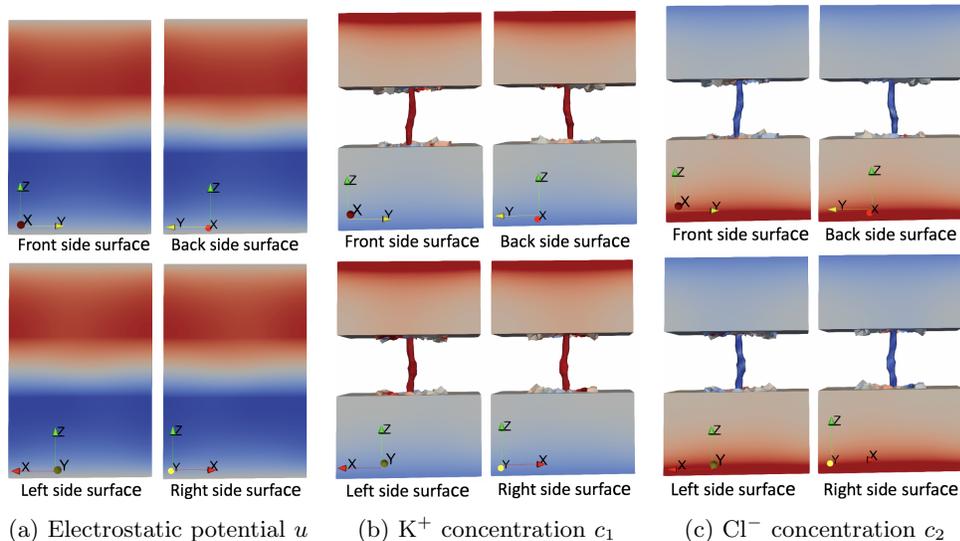


Fig. 6: The periodic boundary value conditions (2.9) well retained in the PNPic finite element solution (u, c_1, c_2) . Here the color mapping ranges for u and c_i are $[-1, 1]$ and $[0, 1]$, respectively, from blue to red.

575 the values of ionic concentrations c_i and electrostatic potential function u produced
 576 by our PNPic finite element solver in color mapping on a surface mesh of ion channel
 577 protein region D_p , membrane region D_m , or solvent region D_s . This feature makes
 578 our PNPic software package particularly useful in the study of ion channel properties.
 579 As an example, Figure 5 displays the values of u on the surface meshes of D_p , D_s ,
 580 and D_m , respectively. The three surface mesh plots of Figure 5 also display the
 581 complicated shapes of the interfaces Γ_p , Γ_{pm} , and Γ_m . From Figure 3(b), it can be
 582 seen that our protein region D_p wraps well the molecular structure of GA.

583 Figure 6 displays the boundary values of the electrostatic potential u and con-
 584 centrations c_1 and c_2 on the four side surfaces Γ_N of the box domain Ω and the four
 585 side surfaces $\Gamma_N \cap \partial D_s$ of the solvent region D_s in color mapping. Here u , c_1 and c_2
 586 were generated by our PNPic finite element software package using $u_b = -1$, $u_t = 1$,
 587 and $c_i^b = 0.5$ mol/L for $i = 1, 2$. The plots from this figure confirm that our PNPic
 588 finite element solution can well retain the periodic boundary value conditions (2.9).

589 Figure 7 displays the convergence of our relaxation iterative scheme, defined in
 590 (5.6) and (5.7) in terms of iteration numbers and the performance of our software

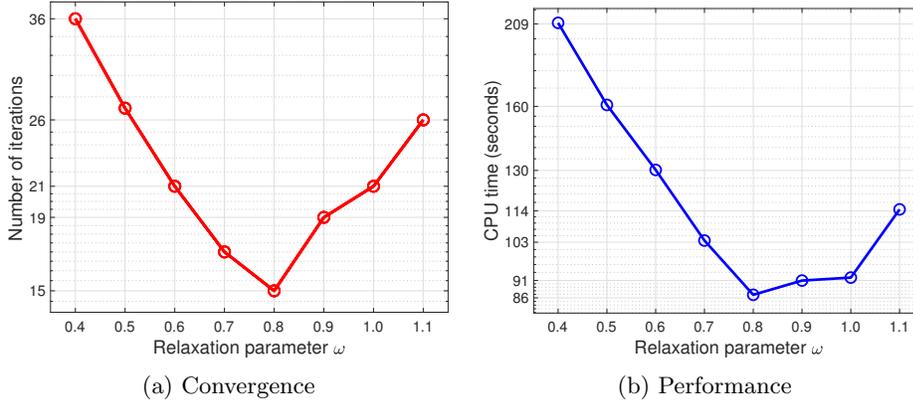


Fig. 7: Convergence and performance of our relaxation iterative scheme (5.6) for solving the PNPic finite element system (5.5) as a function of ω for a GA (PDB identification code 1MAG) in the 0.1 molar KCl solution with $u_b = 1$ and $u_t = 0$.

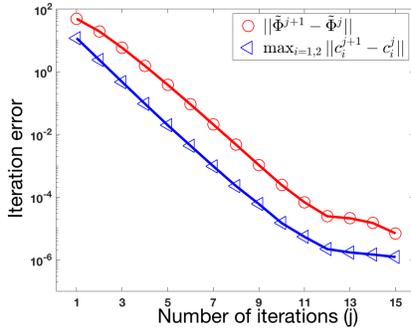


Fig. 8: Iteration errors $\max_{i=1,2} \|c_i^{j+1} - c_i^j\|$ and $\|\tilde{\Phi}^{j+1} - \tilde{\Phi}^j\|$ of iteration j for the PNPic relaxation iterative scheme defined in (5.6) and (5.7) using $\omega = 0.8$.

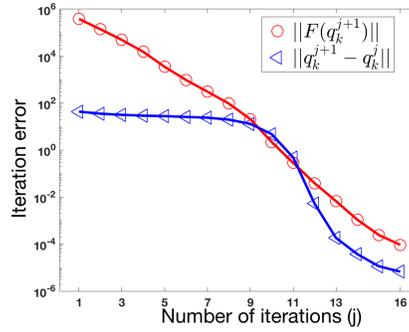


Fig. 9: Iteration errors $\|F(q_k^{j+1})\|$ and $\|q_k^{j+1} - q_k^j\|$ of iteration j for Newton scheme (5.12) for finite element equation $F(\tilde{\Phi}) = 0$ of (5.9) at $k = 0$.

591 package in terms of computer CPU time, as a function of the relaxation parameter
 592 ω . Here we set $u_b = 1$, $u_t = 0$, and $c_1^b = c_2^b = 0.1$ mol/L. From the figure, it
 593 can be seen that the number of iterations was reduced from 36 at $\omega = 0.4$ to 15
 594 at $\omega = 0.8$ and that the corresponding computer CPU time was reduced from 209
 595 seconds to 86 seconds. These test results show that the convergence and performance
 596 of our relaxation iterative scheme can be improved sharply through properly selecting
 597 a relaxation parameter value.

598 Figure 8 reports the convergence processes of our PNPic relaxation iterative
 599 scheme. From the figure, it can be seen that the iteration errors for both $\tilde{\Phi}$ and
 600 c_i were reduced from 10^2 to 10^{-6} in 15 iterations, showing that our PNPic relaxation
 601 iterative scheme has a fast rate of convergence.

602 Figure 9 reports a convergence process of our Newton iterative scheme (5.12) for

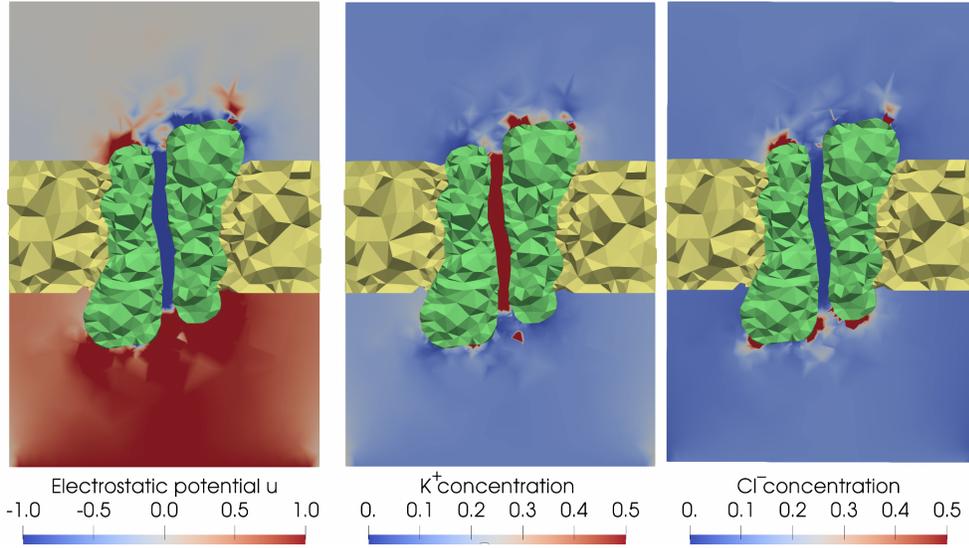


Fig. 10: The electrostatic potential u and the concentrations c_1 and c_2 of K^+ and Cl^- ions in color mapping on a cross section ($x = 0$) of the solvent region D_s . Here the protein and membrane regions are colored in green and yellow, respectively; concentrations are in mol/L; and electrostatic potential u is in $k_B T/e_c$ (≈ 0.0257 volts).

603 solving the nonlinear finite element equation of (5.9) for $\tilde{\Phi}$ at the initial iteration
 604 $k = 0$. Here the initial iterate $\tilde{\Phi}^0$ was generated by the modified Newton iterative
 605 scheme (5.14) for solving our PB ion channel model. From this figure it can be seen
 606 that the iteration errors were reduced quickly from 10^6 to 10^{-6} in 16 iterations only.
 607 Furthermore, as the iteration number k was increased for $k \geq 1$, the total number
 608 of iterations determined by the criteria (5.11) was further reduced due to using the
 609 previous iterate $\tilde{\Phi}^k$ as the initial guess. It is this fast rate of convergence of our
 610 modified Newton iterative scheme that makes our PNPic relaxation iterative scheme
 611 particularly efficient.

612 Figure 10 displays the concentrations of anions Cl^- and cations K^+ and the
 613 electrostatic potential u on a cross section ($x = 0$) of the solvent region D_s in color
 614 mapping. Here we marked the membrane and protein regions in yellow and green
 615 colors, respectively, to clearly show the values in the solvent region D_s . From the
 616 figure, it can be seen that the electrostatic potential values are almost all negative (in
 617 blue) within the channel pore, repelling the anions Cl^- away from the channel pore
 618 (in blue) while attracting the cations K^+ to the channel pore (in red).

619 To visualize a three-dimensional concentration function as a curve across the
 620 channel pore, we construct a rectangular box domain B such that B contains the
 621 channel pore part fully. We then divide B uniformly into m sub-boxes, $\{B_j\}_{j=1}^m$, in
 622 the z -axis direction and calculate a volume integral as follows:

$$623 \quad (6.2) \quad c_{i,j} = \int_{B_j} c_i(\mathbf{r}) d\mathbf{r}, \quad i = 1, 2, \dots, n, \quad j = 1, 2, \dots, m,$$

624 where c_i has been set to be zero outside the solvent region D_s to ensure the definition
 625 of the above integrals. Clearly, $c_{i,j}$ gives the total amount of the ions of species i in

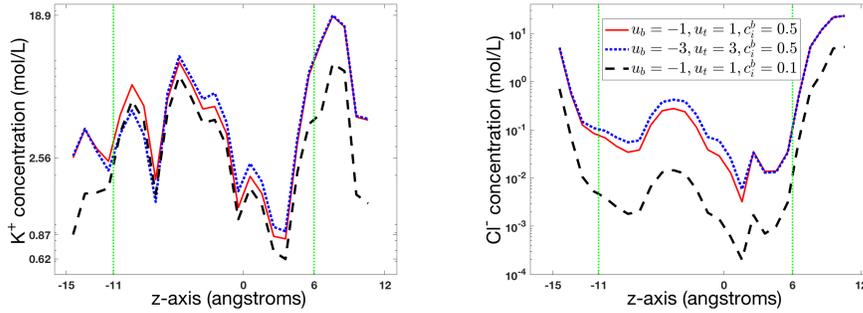


Fig. 11: A comparison of the concentrations of K^+ and Cl^- ions within and near the channel pore ($-11 < z < 6$) generated by the PNPic model for GA (PDB identification code 1MAG) using three different boundary value functions g_i and g defined in (6.1).

Table 2: A comparison of the currents estimated by our new formula (6.4) with the experimental data reported in [12] for GA (PDB identification code 1MAG) in a 0.1 molar NaCl solution. Here voltages are in mV and currents in pA.

Voltage across the membrane	50	100	150	200
Averaged current by formula (6.4)	0.5878	1.2026	1.8430	2.5072
Experimental current reported in [12]	0.65	1.2	1.71	2.12
Relative error	0.0956	0.0022	0.0778	0.1826

626 the sub-box B_j . We next set z^j to be the z -coordinate of a midpoint of B_j to produce
 627 m points, $(z^j, c_{i,j})$ for $j = 1, \dots, m$. Linking these points results in a curve of c_i as a
 628 function of z from z^1 to z^m . Clearly, such a curve provides us with a simple tool for
 629 visualizing the distribution of an ionic species within the channel pore. It can also be
 630 valuable for us to compare concentration functions.

631 We did numerical tests to study the effect of Dirichlet boundary value conditions
 632 on the concentrations c_1 and c_2 . Here $B = [-1.791, 1.2125] \times [-0.8262, 1.6595] \times$
 633 $[-14.4, 10.6]$ and B was uniformly divided into 28 sub-boxes B_j (i.e., $m = 26$) to
 634 produce 26 points $(z_j, c_{i,j})$. We solved the PNPic model using three different boundary
 635 value functions as listed in Table 1, along with the performance data of our relaxation
 636 iterative scheme. A comparison of the concentrations is displayed in Figure 11.

637 Figure 11 shows that changing the boundary value function of an electrostatic
 638 potential u (i.e., changing a voltage across the membrane) has an impact on concen-
 639 tration functions within and near the channel pore. We also see that changing the
 640 bulk concentrations c_i^b caused significant changes outside the channel pore for cations
 641 K^+ and inside the channel pore for anions Cl^- .

642 The test results of Figures 10 and 11 validate our PNPic model since they clearly
 643 describe the distribution patterns of cations and anions, which match the well-known
 644 fact that the GA is cation selective.

645 Finally, as an application of PNPic, we present a new formula for computing the
 646 electric current across the membrane and compare computed values with experimental
 647 data. It is known that the electric current I_S passing a cross section S of the channel

648 pore can be calculated by

$$649 \quad (6.3) \quad I_S = -\frac{e_c N_A}{10^3} \sum_{i=1}^n Z_i D_{i,c} \int_S \left[\frac{\partial c_i(\mathbf{s})}{\partial z} + Z_i c_i(\mathbf{s}) \frac{\partial u(\mathbf{s})}{\partial z} \right] ds$$

650 provided that the normal direction of the cross section S coincides with the z -axis
 651 direction, each ionic concentration c_i is measured in mol/L, $D_{i,c}$ is a diffusion coefficient
 652 within the channel pore in $\text{\AA}/\text{ps}$ (pico-second), and the current is measured in
 653 pA (pico-ampere). In the steady state, I_S only varies with the cross-surface S within
 654 the channel pore since both $\frac{\partial c_i(\mathbf{s})}{\partial z}$ and $\frac{\partial u(\mathbf{s})}{\partial z}$ with $\mathbf{s} = (x, y, z)$ are independent of z .
 655 In calculation, different values of I_S can be derived due to either numerical errors or
 656 S having different sizes. Thus, an average value I_{ave} of I_S is often calculated using
 657 several cross sections. However, for an irregular tetrahedral mesh of the solvent region
 658 D_s , the calculation of I_S is difficult since the calculation of a surface integral over S
 659 requires a mesh of S and an interpolation of both $\frac{\partial c_i(\mathbf{s})}{\partial z}$ and $\frac{\partial u(\mathbf{s})}{\partial z}$ onto this surface
 660 mesh, which are very difficult tasks to be done numerically. To avoid these difficulties,
 661 we present a new formula for computing I_{ave} as follows:

$$662 \quad (6.4) \quad I_{ave} = -\frac{\theta e_c N_A}{h_B 10^3} \sum_{i=1}^n Z_i D_{i,b} \int_B \left[\frac{\partial c_i(\mathbf{r})}{\partial z} + Z_i c_i(\mathbf{r}) \frac{\partial u(\mathbf{r})}{\partial z} \right] d\mathbf{r},$$

663 where B is a piece of the ion channel pore with height h_B in the z -axis direction,
 664 $0 < \theta \leq 1$, and $D_{i,b}$ is the diffusion coefficient of species i in the bulk solution region.
 665 Here $D_{i,c}$ has been set as $D_{i,c} = \theta D_{i,b}$.

666 In fact, since $B \approx S \times [z_1, z_2]$ with $z_2 - z_1 = h_B$, we can get that

$$667 \quad \int_B \left[\frac{\partial c_i(\mathbf{r})}{\partial z} + Z_i c_i(\mathbf{r}) \frac{\partial u(\mathbf{r})}{\partial z} \right] d\mathbf{r} \approx \int_{z_1}^{z_2} \int_S \left[\frac{\partial c_i(\mathbf{s})}{\partial z} + Z_i c_i(\mathbf{s}) \frac{\partial u(\mathbf{s})}{\partial z} \right] ds dz$$

$$668 \quad = h_B \int_S \left[\frac{\partial c_i(\mathbf{s})}{\partial z} + Z_i c_i(\mathbf{s}) \frac{\partial u(\mathbf{s})}{\partial z} \right] ds,$$

669 where we have used the fact that the surface integral is independent of z . Applying
 670 the above identity to (6.3), we show that I_{ave} is an approximation to I_S .

671 In the tests, we set B with the bottom surface at $z = -8$ and the top surface
 672 at $z = 2$ since the buffer size η was set as 3 (i.e., $h_B = 10 \text{ \AA}$), $c_i^b = 0.1 \text{ mol/L}$,
 673 $\theta = 0.0245$, $u_t = 0$, and $u_b = 50, 100, 150$, and 200 mV ($1 \text{ mV} = 0.001 \text{ volts}$). The
 674 test results are reported in Table 2. From these test results, it can be seen that
 675 the currents computed by our PNPic finite element software package match well the
 676 experimental data reported in [12]. These test results further validate our PNPic
 677 model and software package.

678 **7. Conclusions.** We have presented a new PNP ion channel model using per-
 679 iodical boundary value conditions, called PNPic, and developed an effective finite
 680 element relaxation iterative algorithm for solving PNPic. We then implemented this
 681 PNPic finite element algorithm as a software package for the calculation of electro-
 682 static potential density function, ionic concentration functions, and the distribution
 683 of ions and electric current within an ion channel pore. This PNPic software pack-
 684 age works for an ion channel protein with a three-dimensional X-ray crystallographic
 685 molecular structure in an ionic solvent with multiple ionic species.

686 In particular, because of the periodic boundary value conditions, our PNPic model
 687 can reflect the influence of ion channels from outside a simulation box on the cal-
 688 culation of ionic concentrations and an electrostatic potential. Using our solution

689 decomposition scheme, we simplify the PNPic system as a new system that does
690 not involve any singularity and can be much easier to solve numerically so that the
691 complexity of PNPic is reduced remarkably. We also show that the accuracy of the
692 finite element solver can be well retained by using the Slotboom variable transforma-
693 tion technique. We have developed an efficient modified Newton iterative scheme for
694 solving each nonlinear finite element equation that is generated from the Slotboom
695 variable transformation. Through constructing proper communication operators, we
696 have successively carried out function operations between different finite element func-
697 tion spaces, which are defined on different physical domains (a solvent region for ionic
698 concentrations and a box domain for potential functions) and subject to periodic
699 boundary constraints. As applications, we have obtained new formulas for visualizing
700 the distribution of an ionic species within the channel pore in a simple curve (see
701 (6.2)) and for computing the electric current passing on average a cross section of
702 an ion channel pore (see (6.4)). Moreover, we did numerical tests on an ion channel
703 protein and reported the numerical results that demonstrate the convergence and per-
704 formance of our PNPic finite element solver. Finally, we validated our PNPic model
705 using the cation selectivity property of an ion channel protein and the experimental
706 data from a chemical laboratory.

707 In this work, we have mainly focused on the presentation of our new PNPic model
708 and its effective finite element solver and only reported numerical results on a small
709 ion channel protein in a symmetric 1:1 ionic solvent. But our PNPic software package
710 can be applied to the calculation of electrostatic potential and ionic concentrations for
711 a large ion channel protein in ionic solvents with multiple species. It also can be used
712 to study the various properties of our PNPic model. For example, we will study how
713 and to what extent the periodic boundary value conditions can affect ion transport
714 and electric current across membrane or within an ion channel pore. Moreover, our
715 PNPic software package can be used to make various numerical experiments to justify
716 the novelty and advantage of our PNPic model in comparison to those reported in
717 [36, 49]. We will further improve the convergence and performance of our PNPic
718 finite element solver using other advanced numerical techniques to make our PNPic
719 software package a powerful tool for ion channel simulations.

720 Finally, it is worth noting that a repetition of one type of ion channel protein
721 along the membrane, as done in our construction of periodic boundary value condi-
722 tions, has been routinely used in state-of-the-art molecular dynamics for calculating
723 long-range electrostatic interactions by means of a simulation box containing a single
724 protein molecule. This treatment reduces the complexity of membrane modeling re-
725 markably, making it possible for us to count the electrostatic interactions outside a
726 simulation box. On the other hand, it does produce modeling errors since a real cell
727 membrane consists of various ion channel proteins as passage conduits for different
728 ionic species. In order to improve the reliability of our PNPic model in the calculation
729 of electrostatics and ionic concentrations, it is important to estimate such modeling
730 errors either theoretically or numerically via the experimental data from chemical
731 laboratories and molecular dynamics simulations. We plan to do so in the future.

732 **Acknowledgement.** B. Lu was supported by the National Key Research and
733 Development Program of China (2016YFB0201304), the National Natural Science
734 Foundation of China (NSFC 21573274, 11771435), and Science Challenge Program
735 (TZ2016003).

- 737 [1] R. ADAMS AND J. FOURNIER, *Sobolev Spaces*, Vol. 140 of Pure and Applied Mathematics,
738 Elsevier/Academic Press, Amsterdam, 2nd ed., 2003.
- 739 [2] O. S. ANDERSEN AND R. E. KOEPPE, *Molecular determinants of channel function*, *Physiological*
740 *reviews*, 72 (1992), pp. S89–S158.
- 741 [3] O. S. ANDERSEN, R. E. KOEPPE, AND B. ROUX, *Gramicidin channels*, *IEEE Transactions on*
742 *NanoBioscience*, 4 (2005), pp. 10–20.
- 743 [4] D. BODA, M. VALISKÓ, D. HENDERSON, R. S. EISENBERG, D. GILLESPIE, AND W. NONNER, *Ionic*
744 *selectivity in L-type calcium channels by electrostatics and hard-core repulsion*, *Journal of*
745 *General Physiology*, 133 (2009), pp. 497–509.
- 746 [5] W. M. BOTELLO-SMITH AND R. LUO, *Applications of MMPBSA to membrane proteins I: Ef-*
747 *ficent numerical solutions of periodic Poisson-Boltzmann equation*, *Journal of Chemical*
748 *Information and Modeling*, 55 (2015), pp. 2187–2199.
- 749 [6] S. BRENNER AND L. SCOTT, *The Mathematical Theory of Finite Element Methods*, Springer-
750 Verlag, New York, 3rd ed., 2008.
- 751 [7] J. H. CHAUDHRY, J. COMER, A. AKSIMENTIEV, AND L. N. OLSON, *A stabilized finite element*
752 *method for modified Poisson-Nernst-Planck equations to determine ion flow through a*
753 *nanopore*, *Communications in Computational Physics*, 15 (2014), pp. 93–125.
- 754 [8] D.-P. CHEN, J. LEAR, AND R. S. EISENBERG, *Permeation through an open channel: Poisson-*
755 *Nernst-Planck theory of a synthetic ionic channel*, *Biophysical Journal*, 72 (1997), pp. 97–
756 116.
- 757 [9] M. CHEN AND B. LU, *TSMESH: A robust method for molecular surface mesh generation using*
758 *a trace technique*, *Journal of Chemical Theory and Computation*, 7 (2010), pp. 203–212.
- 759 [10] M. CHEN, B. TU, AND B. LU, *Triangulated manifold meshing method preserving molecular*
760 *surface topology*, *Journal of Molecular Graphics and Modelling*, 38 (2012), pp. 411–418.
- 761 [11] I. CHERN, J. LIU, AND W. WANG, *Accurate evaluation of electrostatics for macromolecules in*
762 *solution*, *Methods and Applications of Analysis*, 10 (2003), pp. 309–328.
- 763 [12] C. D. COLE, A. S. FROST, N. THOMPSON, M. COTTEN, T. A. CROSS, AND D. D. BUSATH,
764 *Noncontact dipole effects on channel permeation. VI. 5F- and 6F-Trp Gramicidin channel*
765 *currents*, *Biophysical Journal*, 83 (2002), pp. 1974–1986.
- 766 [13] R. S. EISENBERG, *Ionic channels in biological membranes: natural nanotubes*, *Accounts of*
767 *Chemical Research*, 31 (1998), pp. 117–123.
- 768 [14] A. FLAVELL, J. KABRE, AND X. LI, *An energy-preserving discretization for the Poisson-Nernst-*
769 *Planck equations*, *Journal of Computational Electronics*, 16 (2017), pp. 431–441.
- 770 [15] A. FLAVELL, M. MACHEN, R. S. EISENBERG, J. KABRE, C. LIU, AND X. LI, *A conservative*
771 *finite difference scheme for Poisson-Nernst-Planck equations*, *Journal of Computational*
772 *Electronics*, 13 (2014), pp. 235–249.
- 773 [16] H. GAO AND P. SUN, *A linearized local conservative mixed finite element method for Poisson-*
774 *Nernst-Planck equations*, *Journal of Scientific Computing*, 77 (2018), pp. 793–817.
- 775 [17] D. GILLESPIE, *A review of steric interactions of ions: Why some theories succeed and others*
776 *fail to account for ion size*, *Microfluidics and Nanofluidics*, 18 (2015), pp. 717–738.
- 777 [18] D. GILLESPIE, W. NONNER, AND R. S. EISENBERG, *Coupling Poisson-Nernst-Planck and den-*
778 *sity functional theory to calculate ion flux*, *Journal of Physics Condensed Matter*, 14 (2002),
779 pp. 12129–12145.
- 780 [19] H. K. GUMMEL, *A self-consistent iterative scheme for one-dimensional steady state transistor*
781 *calculations*, *IEEE Transactions on electron devices*, 11 (1964), pp. 455–465.
- 782 [20] D. HE AND K. PAN, *An energy preserving finite difference scheme for the Poisson-Nernst-*
783 *Planck system*, *Applied Mathematics and Computation*, 287 (2016), pp. 214–223.
- 784 [21] U. HOLLERBACH, D.-P. CHEN, AND R. S. EISENBERG, *Two- and three-dimensional Poisson-*
785 *Nernst-Planck simulations of current flow through Gramicidin A*, *Journal of Scientific*
786 *Computing*, 16 (2001), pp. 373–409.
- 787 [22] T.-L. HORNG, T.-C. LIN, C. LIU, AND R. S. EISENBERG, *PNP equations with steric effects:*
788 *a model of ion flow through channels*, *Journal of Physical Chemistry B*, 116 (2012),
789 pp. 11422–11441.
- 790 [23] W. IM AND B. ROUX, *Ion permeation and selectivity of ompf porin: a theoretical study based on*
791 *molecular dynamics, brownian dynamics, and continuum electrodiffusion theory*, *Journal*
792 *of molecular biology*, 322 (2002), pp. 851–869.
- 793 [24] N. JI, T. LIU, J. XU, L. SHEN, AND B. LU, *A finite element solution of lateral periodic Poisson-*
794 *Boltzmann model for membrane channel proteins*, *International Journal of Molecular Sci-*
795 *ences*, 19 (2018), p. 695.
- 796 [25] R. R. KETCHUM, K. C. LEE, S. HUO, AND T. A. CROSS, *Macromolecular structural elucidation*

- 797 with solid-state NMR-derived orientational constraints, *Journal of Biomolecular NMR*, 8
798 (1996), pp. 1–14.
- 799 [26] M. G. KURNIKOVA, R. D. COALSON, P. GRAF, AND A. NITZAN, *A lattice relaxation algorithm for*
800 *three-dimensional Poisson-Nernst-Planck theory with application to ion transport through*
801 *the Gramicidin A channel*, *Biophysical Journal*, 76 (1999), pp. 642–56.
- 802 [27] H. LIU AND Z. WANG, *A free energy satisfying finite difference method for Poisson-Nernst-*
803 *Planck equations*, *Journal of Computational Physics*, 268 (2014), pp. 363–376.
- 804 [28] J.-L. LIU AND R. S. EISENBERG, *Poisson-Nernst-Planck-Fermi theory for modeling biological*
805 *ion channels*, *Journal of Chemical Physics*, 141 (2014), p. 12B640.1.
- 806 [29] P. LIU, X. JI, AND Z. XU, *Modified Poisson-Nernst-Planck model with accurate Coulomb*
807 *correlation in variable media*, *SIAM Journal on Applied Mathematics*, 78 (2018), pp. 226–
808 245.
- 809 [30] T. LIU, S. BAI, B. TU, M. CHEN, AND B. LU, *Membrane-channel protein system mesh con-*
810 *struction for finite element simulations*, *Computational and Mathematical Biophysics*, 3
811 (2015).
- 812 [31] T. LIU, M. CHEN, AND B. LU, *Efficient and qualified mesh generation for Gaussian molecular*
813 *surface using adaptive partition and piecewise polynomial approximation*, *SIAM Journal*
814 *on Scientific Computing*, 40 (2018), pp. B507–B527.
- 815 [32] W. LIU, X. TU, AND M. ZHANG, *Poisson-Nernst-Planck systems for ion flow with density*
816 *functional theory for hard-sphere potential: I-V relations and critical potentials. Part II:*
817 *numerics*, *Journal of Dynamics and Differential Equations*, 24 (2012), pp. 985–1004.
- 818 [33] X. LIU AND B. LU, *Incorporating Born solvation energy into the three-dimensional Poisson-*
819 *Nernst-Planck model to study ion selectivity in KcsA K⁺ channels*, *Physical Review E*, 96
820 (2017), 062416.
- 821 [34] X. LIU, Y. QIAO, AND B. LU, *Analysis of the mean field free energy functional of electrolyte*
822 *solution with nonhomogenous boundary conditions and the generalized PB/PNP equations*
823 *with inhomogeneous dielectric permittivity*, *SIAM Journal on Applied Mathematics*, 78
824 (2018), pp. 1131–1154.
- 825 [35] A. LOGG, K.-A. MARDAL, AND G. N. WELLS, eds., *Automated Solution of Differential Equa-*
826 *tions by the Finite Element Method*, Vol. 84 of *Lecture Notes in Computational Science*
827 *and Engineering*, Springer Verlag, 2012.
- 828 [36] B. LU, M. J. HOLST, J. A. MCCAMMON, AND Y. ZHOU, *Poisson-Nernst-Planck equations for*
829 *simulating biomolecular diffusion–reaction processes I: Finite element solutions*, *Journal*
830 *of Computational Physics*, 229 (2010), pp. 6979–6994.
- 831 [37] B. LU AND Y. C. ZHOU, *Poisson-Nernst-Planck equations for simulating biomolecular diffusion-*
832 *reaction processes II: Size effects on ionic distributions and diffusion-reaction rates*, *Bio-*
833 *physical Journal*, 100 (2011), pp. 2475–2485.
- 834 [38] B. LU, Y. C. ZHOU, G. A. HUBER, S. D. BOND, M. J. HOLST, AND J. A. MCCAMMON, *Elec-*
835 *trodiffusion: a continuum modeling framework for biomolecular systems with realistic spa-*
836 *tiotemporal resolution*, *Journal of Chemical Physics*, 127 (2007), pp. 10B604–78.
- 837 [39] JENS AUGUST LUNDBÆK, *Regulation of membrane protein function by lipid bilayer elasticity:*
838 *a single molecule technology to measure the bilayer properties experienced by an embedded*
839 *protein*, *Journal of Physics: Condensed Matter*, 18 (2006), pp. S1305–S1344.
- 840 [40] S. R. MATHUR AND J. Y. MURTHY, *A multigrid method for the Poisson-Nernst-Planck equa-*
841 *tions*, *International Journal of Heat and Mass Transfer*, 52 (2009), pp. 4031–4039.
- 842 [41] M. S. METTI, J. XU, AND C. LIU, *Energetically stable discretizations for charge transport and*
843 *electrokinetic models*, *Journal of Computational Physics*, 306 (2016), pp. 1–18.
- 844 [42] J. M. ORTEGA AND W. C. RHEINOLDT, *Iterative Solution of Nonlinear Equations in Several*
845 *Variables*, Academic Press, 1970.
- 846 [43] Y. QIAO, C. LIAN, B. LU, AND J. WU, *Modeling selective ion adsorption into cylindrical*
847 *nanopores*, *Chemical Physics Letters*, 709 (2018), pp. 116–124.
- 848 [44] Y. QIAO, X. LIU, M. CHEN, AND B. LU, *A local approximation of fundamental measure theory*
849 *incorporated into three-dimensional Poisson-Nernst-Planck equations to account for hard*
850 *sphere repulsion among ions*, *Journal of Statistical Physics*, 163 (2016), pp. 156–174.
- 851 [45] B. ROUX, T. ALLEN, S. BERNECHE, AND W. IM, *Theoretical and computational models of*
852 *biological ion channels*, *Quarterly Reviews of Biophys*, 37 (2004), pp. 15–103.
- 853 [46] B. ROUX AND M. KARPLUS, *Molecular Dynamics Simulations of the Gramicidin Channel*,
854 *Annu. Rev. Biomol. Struct. Dyn.* 23 (1994), pp. 731–761.
- 855 [47] J. W. SLOTBOOM, *Computer-aided two-dimensional analysis of bipolar transistors*, *IEEE Trans-*
856 *actions on Electron Devices*, 20 (1973), pp. 669–679.
- 857 [48] S. TANIZAKI AND M. FEIG, *A generalized Born formalism for heterogeneous dielectric environ-*
858 *ments: application to the implicit modeling of biological membranes*, *Journal of chemical*

- 859 physics, 122 (2005), p. 124706.
- 860 [49] B. TU, M. CHEN, Y. XIE, L. ZHANG, R. S. EISENBERG, AND B. LU, *A parallel finite element*
861 *simulator for ion transport through three-dimensional ion channel systems*, Journal of
862 Computational Chemistry, 34 (2013), pp. 2065–2078.
- 863 [50] B. TU, Y. XIE, L. ZHANG, AND B. LU, *Stabilized finite element methods to simulate the con-*
864 *ductances of ion channels*, Computer Physics Communications, 188 (2015), pp. 131–139.
- 865 [51] G.-W. WEI, Q. ZHENG, Z. CHEN, AND K. XIA, *Variational multiscale models for charge trans-*
866 *port*, SIAM Review, 54 (2012), pp. 699–754.
- 867 [52] D. XIE, *New solution decomposition and minimization schemes for Poisson-Boltzmann equa-*
868 *tion in calculation of biomolecular electrostatics*, Journal of Computational Physics, 275
869 (2014), pp. 294–309.
- 870 [53] D. XIE, H. W. VOLKMER, AND J. YING, *Analytical solutions of nonlocal Poisson dielectric*
871 *models with multiple point charges inside a dielectric sphere*, Physical Review E, 93 (2016),
872 pp. 043304.
- 873 [54] Q. ZHENG, D. CHEN, AND G.-W. WEI, *Second-order Poisson-Nernst-Planck solver for ion*
874 *transport*, Journal of Computational Physics, 230 (2011), pp. 5239–5262.