

A Finite Element Iterative Solver for a PNP Ion Channel Model with Neumann Boundary Condition and Membrane Surface Charge

Dexuan Xie¹ and Zhen Chao

Department of Mathematical Sciences, University of Wisconsin-Milwaukee, Milwaukee, WI, 53201-0413, USA

Abstract

In this paper, an effective finite element iterative algorithm is presented for solving a Poisson-Nernst-Planck ion channel (PNPic) model with Neumann boundary value condition and a membrane surface charge density. It is constructed by a solution decomposition scheme to avoid singularity problems caused by atomic charges, an alternating block iterative scheme to sharply reduce computation complexity and computer memory requirement, and a Slotboom variable transformation scheme to significantly enhance numerical stability, as well as a modified Newton iterative scheme to efficiently solve each related nonlinear finite element equation. This PNPic finite element solver is then implemented as a software package that works for an ion channel protein with a crystallographic structure in a mixture solution of multiple ionic species. Furthermore, a finite element scheme is presented to compute a volume integral of a potential/concentration function over a block of a solvent region. This work can greatly improve the accuracy of a visualization tool for depicting the distribution pattern of a three-dimensional potential/concentration function across membrane in a simple two-dimensional curve. Numerical results for a mouse voltage-dependent anion-channel isoform (mVDAC1) in a solution of up to four ionic species are reported. They demonstrate the convergence of the PNPic iterative solver, the performance of the software package, and the valuable usage of the visualization tool in the comparison study of different potential and concentration functions. They also validate that this PNPic model can well retain the anion selectivity property of mVDAC1.

Keywords: Poisson-Nernst-Planck equations, finite element method, ion channel, electrostatics, Neumann boundary condition, membrane surface charge

2000 MSC: 92-08, 65N30, 35J66, 65K10

1. Introduction

A system of Poisson-Nernst-Planck equations (PNP) is commonly used in the development of ion channel models since it can qualitatively capture several important macroscopic properties such as electric current, current-voltage relations, conductance rectification, and membrane potentials [6, 15, 20, 25, 28]. To reflect ionic size effects [13, 22, 27, 34] and variable dielectric properties [18], etc., several PNP ion channel models were developed, and solved numerically by finite difference schemes [15, 41] and finite element schemes [10, 21, 31, 32]. However, as a system of strongly coupled nonlinear partial differential equations, a PNP ion channel model is often convection-dominated, and demands a positive function of ionic concentration, which cause many numerical difficulties in the development of PNP numerical solvers. How to solve a PNP model numerically remains an important research topic in order to better overcome the singularity caused by atomic charges, to further reduce highly-demanded computer memory requirement, and to more effectively deal with the related strong nonlinearity. We also noted that the current PNP ion channel models mostly ignored the influence of membrane charges, membrane boundary, and the ion channel proteins outside a PNP simulation domain.

In fact, a PNP ion channel model is commonly defined on a box domain, which is partitioned into a protein region D_p , a membrane region D_m , and a solvent region D_s . Here, D_m separates a box of ionic solution into two solvent

¹Corresponding author. E-mail address: dxie@uwm.edu

compartments, and D_p hosts one ion channel protein molecule as the conduit for ion transport from one solvent compartment to the other. Such a single-molecule model clearly needs either proper periodic boundary value conditions or proper Neumann boundary value conditions in order to reflect the influence of membrane and ion channels outside the box domain. But, in the current PNP ion channel models, periodic/Neumann boundary value conditions were rarely employed due to implementation difficulties. With the advance of numerical techniques, periodic boundary conditions started to be used to construct Poisson-Boltzmann (PB) ion channel models in [2, 14]. Toward this direction of efforts, recently, Neumann boundary value conditions were applied to the construction of a size modified PB ion channel model [36], and a periodic boundary condition was applied to the construction of a PNP ion channel model [38], together with their effective finite element solvers. As the continuation of these efforts, in this paper, we construct a PNP ion channel ion channel (PNPic) model using Neumann boundary value conditions and a membrane surface charge density to reflect the influence of both membrane boundary and membrane charges on the calculation of electrostatics and ionic concentrations. We then develop an effective and efficient nonlinear iterative algorithm for solving such a PNPic model by finite element techniques and other advanced mathematical techniques, which include a solution decomposition, an alternating block iterative scheme, and the Slotboom variables transformation.

In fact, one major difficulty of solving a PNP ion channel model comes from the solution singularity caused by atomic charges. It has been known that such a difficulty cannot be overcome unless all the singularity points can be isolated by a solution decomposition scheme [39, Figure 3]. Currently, two solution decomposition schemes were proposed in [7, 35] for solving the Poisson-Boltzmann equation (PBE), and the one from [7] was adapted to the development of a PNP finite difference solver in [41] and a PNP finite element solver in [31]. In this solution decomposition scheme, an electrostatic potential function, u , is split into three component functions, u^s , u^h , and u^r , within a protein region D_p only, resulting in a Laplace boundary value problem of u^h in D_p and a nonlinear interface boundary value problem of u^r in the box domain. Since D_p is a strongly non-convex domain with a complicated non-smooth boundary (i.e., a molecular surface), especially for an ion channel protein, solving such a Laplace boundary value problem may cause serious problems in solution accuracy and solution regularity. The equation of u^r is also difficult to solve due to involving flux interface conditions with jump discontinuities on the interface between D_p and a solvent region D_s . Fortunately, these potential numerical and mathematical problems do not occur in the solution decomposition scheme from [35], since in this scheme, u is split into three component functions, G , Ψ , and $\tilde{\Phi}$, over the whole box domain such that $u = G + \Psi + \tilde{\Phi}$ with G , Ψ , and $\tilde{\Phi}$ being the electrostatic potentials induced by the atomic charges, the potentials from the interfaces and boundary, and the ionic charges from a solvent region, D_s , respectively. Since G contains all the singularity points of u , both Ψ and $\tilde{\Phi}$ become smooth within the solvent and protein regions. Note that $u^r = u$ within D_s and $u = G + \Psi + \tilde{\Phi}$. Hence, $\tilde{\Phi} = u^r - G - \Psi$, which shows that $\tilde{\Phi}$ is much smoother than u^r . As a result, the interface boundary value problem of $\tilde{\Phi}$ can be much easier to solve numerically than that of u^r . Hence, in this paper, we will adapt the splitting scheme from [35] to the construction of our PNPic solution decomposition scheme.

A PNP ion channel model is a system of partial differential equations for one electrostatic potential function u and n ionic concentration functions c_i when a solvent contains n ionic species. In order to reduce numerical complexity and computer memory requirement, classic successive relaxation iterative techniques [26] (or related Gummel's iterative technique [11]) is often adopted to construct a PNP iterative algorithm, resulting in n equations that involve gradient calculation for a given potential function. It has been known that such a gradient calculation may decay the calculation accuracy in one order [3]. Moreover, the positivity requirement of c_i produces a further numerical difficulty. Several variable transform techniques were developed to overcome these difficulties [24]. One of them is the Slotboom variables transformation, which was introduced in the early semiconductor device system simulation [30]. It will be used in this paper to develop our PNPic numerical solver since it can transform each Nernst-Planck equation as the one that does not involve any gradient of a potential function, and can produce positive concentrations. However, this transformation results in a system of strongly nonlinear equations, requiring to develop efficient nonlinear iterative solvers. In this paper, we will develop an efficient modified Newton iterative scheme for solving such a nonlinear equation. We also solve each related linear system using a generalized minimal residual method using incomplete LU preconditioning (GMRES-ILU). To this end, we derive an efficient PNPic solver.

Since each ionic concentration function c_i is defined in the solvent region D_s while an electrostatic potential function u in the whole box domain Ω , c_i and u belong to two different finite element function spaces. Thus, a communication operator is required to carry out operations between c_i and u , causing additional difficulty in the implementation of a PNPic numerical solver. To avoid such a difficulty, each c_i was currently extended from D_s to

Ω through setting $c_i = 0$ at the mesh nodes outside D_s such that both c_i and u are defined on the same mesh of Ω . But, this simple treatment may decay the accuracy of a PNP numerical solution significantly since c_i can be nonzero outside D_s on a layer of tetrahedra along the interface between D_s and a protein-membrane region. In this paper, we will directly construct a finite element function space for each ionic concentration function c_i based on an irregular tetrahedral mesh of D_s . We then derive the required communication operators to complete the function operations between two different finite element function spaces without losing any solution accuracy.

Different selections of PNPic parameters may generate different electrostatic potential u and ionic concentrations c_i . How to display the differences among these three dimensional (3D) potential and concentration functions is one important research topic. One way to do so is to display the values of u (or c_i) on a cross section of a solvent region D_s (or other regions of the box domain) in a color mapping. Another way is to simply plot the function values of u (or c_i) in a curve along a line that passes an ion channel pore [20, 41]. However, both ways only reflect the function values of potential and concentration locally. They are too prolix to use to display the function values over a whole solvent region. To gain a visualization tool that can reflect the profile of a 3D potential u (or concentration c_i) over a 3D domain, several efforts were done recently (see [36, 38] for example). In this paper, we introduce one more scheme for computing a volume integral of a 3D function over a block from a block partition of D_s along the membrane normal direction. Because the shape of D_s is very complex, such blocks may have very irregular shapes, especially within an ion channel pore, causing difficulties in the calculation of related volume integrals. In this paper, we overcome these difficulties through extracting a tetrahedral mesh of each block from a tetrahedral mesh of D_s . To this end, we can visualize the distribution pattern of a 3D function (u or c_i) across membrane in a simple 2D curve along the membrane normal direction. Since each point of such a curve represents an average value of the 3D function over a block, the curve produced from our scheme reflects the profile of the 3D function over the whole solvent region. It is particularly valuable for us to compare the electrostatic and ionic concentration functions generated from the PNPic model using different parameter values.

The rest of the paper is organized as follows. In Section 2, we present the PNPic model. In Section 3, we present a PNPic solution decomposition. In Section 4, we reformulate each equation of the PNPic solution decomposition into a variational problem. In Section 5, we construct a PNPic finite element solver. In Section 6, we describe a scheme for computing volume integrals over a set of solvent blocks for both potential and ionic concentration functions. In Section 7, we present our PNPic software package, and report numerical results. Finally, conclusions are made in Section 8.

2. Construction of the PNPic model

Let a rectangular box domain, Ω , be defined by

$$\Omega = \{(x, y, z) \mid L_{x1} < x < L_{x2}, L_{y1} < y < L_{y2}, L_{z1} < z < L_{z2}\}, \quad (1)$$

where $L_{x1}, L_{x2}, L_{y1}, L_{y2}, L_{z1}$, and L_{z2} are sufficiently large real numbers such that the following domain partition holds:

$$\Omega = D_p \cup D_m \cup D_s \cup \Gamma_p \cup \Gamma_m \cup \Gamma_{pm}. \quad (2)$$

Here, D_p denotes a protein region, which holds an ion channel protein molecule with n_p atoms embedded in a membrane region D_m , D_s is a solvent region containing a solution of n ionic species, Γ_p an interface between D_p and D_s , Γ_m an interface between D_m and D_s , and Γ_{pm} an interface between D_p and D_m . We further split the boundary $\partial\Omega$ of Ω by

$$\partial\Omega = \Gamma_D \cup \Gamma_N, \quad (3)$$

where Γ_D consists of the bottom and top surfaces of Ω , and Γ_N consists of the four side surfaces of Ω . See Figure 1 for an illustration of the box domain partition (1), where we have set the normal direction of the membrane surface in the z -axis direction, and the membrane location by two numbers $Z1$ and $Z2$.

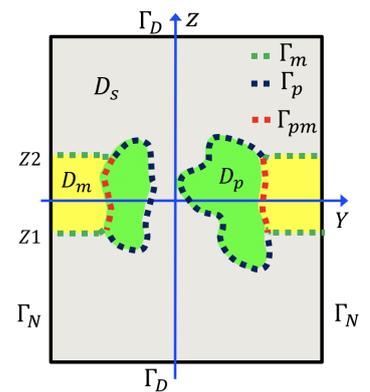


Figure 1: An illustration of a box domain partition defined in (2) and (3).

A Poisson-Nernst-Planck ion channel (PNPic) model is a system of n Nernst-Planck equations and one Poisson dielectric equation subject to proper interface and boundary value conditions. Here the three regions D_p , D_m , and D_s have been treated as homogenous dielectric media with permittivity constants ϵ_p , ϵ_m , and ϵ_s , respectively. The PNPic solution contains a dimensionless electrostatic potential function, u , and an ionic concentration function, c_i , of species i , in terms of spatial variable $\mathbf{r} = (x, y, z)$ and time variable t . The n Nernst-Planck equations are given by

$$\frac{\partial c_i(t, \mathbf{r})}{\partial t} = \nabla \cdot \mathcal{D}_i(t, \mathbf{r}) [\nabla c_i(t, \mathbf{r}) + Z_i c_i(t, \mathbf{r}) \nabla u(t, \mathbf{r})], \quad \mathbf{r} \in D_s, \quad t > 0, \quad i = 1, 2, \dots, n, \quad (4)$$

where \mathcal{D}_i and Z_i denote the diffusion coefficient function and charge number of species i , respectively, and $\nabla = \langle \frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \rangle$ (a gradient operator). The right hand side term of (4) can be reformulated into the expression

$$\nabla \cdot \mathcal{D}_i(t, \mathbf{r}) [\nabla c_i(t, \mathbf{r}) + Z_i c_i(t, \mathbf{r}) \nabla u(t, \mathbf{r})] = \nabla \cdot \mathcal{D}_i(t, \mathbf{r}) c_i(t, \mathbf{r}) \nabla \left[\ln(c_i(t, \mathbf{r})/c_i^b) + Z_i u(t, \mathbf{r}) \right],$$

where c_i^b denotes a bulk concentration of species i . From the above expression we get an electrochemical potential of ionic species i , denoted by μ_i , as follows:

$$\mu_i = \ln(c_i(t, \mathbf{r})/c_i^b) + Z_i u(t, \mathbf{r}), \quad i = 1, 2, \dots, n.$$

In the steady state, we have that

$$\frac{\partial c_i(t, \mathbf{r})}{\partial t} = 0, \quad i = 1, 2, \dots, n,$$

so that both u and c_j become independent of time t , and the Nernst-Planck equations (4) are simplified as

$$\nabla \cdot \mathcal{D}_i(\mathbf{r}) [\nabla c_i(\mathbf{r}) + Z_i c_i(\mathbf{r}) \nabla u(\mathbf{r})] = 0, \quad i = 1, 2, \dots, n.$$

Correspondingly, the electrochemical potential μ_i is also simplified as a function of \mathbf{r} ,

$$\mu_i = \ln(c_i(\mathbf{r})/c_i^b) + Z_i u(\mathbf{r}), \quad i = 1, 2, \dots, n.$$

Setting the above $\mu_i = 0$, we get the well known Boltzmann distribution density function:

$$c_i(\mathbf{r}) = c_i^b e^{-Z_i u}, \quad i = 1, 2, \dots, n,$$

so that a PNP model can be reduced to a Poisson-Boltzmann model. Hence, a Poisson-Boltzmann model can be regarded as a special case of a PNP model. Its solution gives an electrostatic potential u and ionic concentrations c_i in an equilibrium state in the sense of minimizing an electrostatic free energy functional F , since the electrochemical potential μ_i is exactly equal to the variation of F with respect to c_i as proved mathematically in [37, Eq. (30)].

In this paper, we focus on the numerical solution of a steady PNPic model in the nonequilibrium state (i.e., $\mu_i \neq 0$). To reflect the membrane charge influence and membrane boundary effect, we use Neumann boundary conditions on Γ_N and a membrane surface charge density function, σ , in our PNPic model. That is, our PNPic model consists of n steady Nernst-Planck boundary value problems as follows,

$$\nabla \cdot \mathcal{D}_i(\mathbf{r}) [\nabla c_i(\mathbf{r}) + Z_i c_i(\mathbf{r}) \nabla u(\mathbf{r})] = 0, \quad \mathbf{r} \in D_s, \quad (5a)$$

$$\frac{\partial c_i(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} + Z_i c_i(\mathbf{s}) \frac{\partial u(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} = 0, \quad \mathbf{s} \in \Gamma_p \cup \Gamma_m, \quad (5b)$$

$$c_i(\mathbf{s}) = g_i(\mathbf{s}), \quad \mathbf{s} \in \Gamma_D, \quad (5c)$$

$$\frac{\partial c_i(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} = 0, \quad \mathbf{s} \in \Gamma_N \cap \partial D_s, \quad i = 1, 2, \dots, n, \quad (5d)$$

where \mathbf{n}_s and \mathbf{n}_b are the unit outward normal directions of D_s and Ω , respectively, and g_i is a nonzero boundary value

function, together with one Poisson dielectric interface boundary value problem as follows:

$$-\epsilon_p \Delta u(\mathbf{r}) = \alpha \sum_{j=1}^{n_p} z_j \delta_{\mathbf{r}_j}, \quad \mathbf{r} \in D_p, \quad (6a)$$

$$-\epsilon_m \Delta u(\mathbf{r}) = 0, \quad \mathbf{r} \in D_m, \quad (6b)$$

$$-\epsilon_s \Delta u(\mathbf{r}) = \beta \sum_{i=1}^n Z_i c_i(\mathbf{r}), \quad \mathbf{r} \in D_s, \quad (6c)$$

$$u(\mathbf{s}^-) = u(\mathbf{s}^+), \quad \epsilon_p \frac{\partial u(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_s \frac{\partial u(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})}, \quad \mathbf{s} \in \Gamma_p, \quad (6d)$$

$$u(\mathbf{s}^-) = u(\mathbf{s}^+), \quad \epsilon_m \frac{\partial u(\mathbf{s}^-)}{\partial \mathbf{n}_m(\mathbf{s})} = \epsilon_s \frac{\partial u(\mathbf{s}^+)}{\partial \mathbf{n}_m(\mathbf{s})} + \tau \sigma, \quad \mathbf{s} \in \Gamma_m, \quad (6e)$$

$$u(\mathbf{s}^-) = u(\mathbf{s}^+), \quad \epsilon_p \frac{\partial u(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_m \frac{\partial u(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})}, \quad \mathbf{s} \in \Gamma_{pm}, \quad (6f)$$

$$u(\mathbf{s}) = g(\mathbf{s}), \quad \mathbf{s} \in \Gamma_D, \quad (6g)$$

$$\frac{\partial u(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} = 0, \quad \mathbf{s} \in \Gamma_N, \quad (6h)$$

where z_j and \mathbf{r}_j denote the charge number and position vector of atom j , \mathbf{n}_p and \mathbf{n}_m are the unit outward normal directions of D_p and D_m , respectively, $\delta_{\mathbf{r}_j}$ is the Dirac delta distribution at \mathbf{r}_j , g is a boundary value function, α , β and τ are physical constants, $\frac{\partial u(\mathbf{s})}{\partial \mathbf{n}(\mathbf{s})}$ denotes the directional derivative of u along a unit outside normal direction \mathbf{n} (say, $\mathbf{n} = \mathbf{n}_p$), and $u(\mathbf{s}^\pm) = \lim_{t \rightarrow 0^+} u(\mathbf{s} \pm t\mathbf{n}(\mathbf{s}))$, which are the two sided limits along a direction \mathbf{n} of a region (say a protein region, D_p) from the inside and outside the region.

In Physics, the boundary condition (5b) reflects the fact that none of ionic particles enter the protein and membrane regions D_p and D_m across the interface $\Gamma_p \cup \Gamma_m$; the Dirichlet boundary value conditions (5c) and (6g) on the top and bottom surfaces Γ_D of the simulation box Ω can be used to mimic an external voltage across the membrane; and the Neumann boundary condition (6h) naturally reflects the fact that none of the charges enter the box domain Ω from the four side surface Γ_N .

In calculation, the permittivity constants ϵ_m and ϵ_s can be simply set as $\epsilon_m = 2$ and $\epsilon_s = 80$, but the permittivity constant ϵ_p of the protein region D_p can be adjusted between 2 and 20 as indicated by experiments [8]. Experiments also indicate that the membrane surface charge density σ can be selected between 0.2 to 30 ($\mu\text{C}/\text{cm}^2$) [12, 16].

In the PNPic model defined in (5) and (6), we have used the SI units to measure length in angstroms (\AA), ionic concentrations c_i in moles per liter (mol/L), temperature in Kelvins (K), charges in Coulombs (C), diffusion coefficients \mathcal{D}_i in $\text{\AA}^2/\text{ps}$, and σ in $\mu\text{C}/\text{cm}^2$, and have defined the three physical constants α , β and τ by

$$\alpha = \frac{10^{10} e_c^2}{\epsilon_0 k_B T}, \quad \beta = \frac{N_A e_c^2}{10^{17} \epsilon_0 k_B T}, \quad \tau = \frac{10^{-12} e_c}{\epsilon_0 k_B T}, \quad (7)$$

such that u is dimensionless. Here, ϵ_0 is the permittivity of the vacuum, e_c is the elementary charge, k_B is the Boltzmann constant, T is the absolute temperature, and N_A is the Avogadro number, which estimates the number of ions per mole. After finding u , an electrostatic potential function, Φ , can be derived in units volts by the formula

$$\Phi(\mathbf{r}) = \frac{k_B T}{e_c} u(\mathbf{r}), \quad \mathbf{r} \in \Omega.$$

At $T = 298.5\text{K}$, we estimate the values of α , β , τ , and $\frac{k_B T}{e_c}$ by

$$\alpha \approx 7042.9399, \quad \beta \approx 4.2414, \quad \tau \approx 4.392, \quad \frac{k_B T}{e_c} \approx 0.026 \text{ volts.}$$

Thus, $u = 1$ means an electrostatic potential in about 0.026 volts.

How to select the diffusion coefficient function $\mathcal{D}_i(\mathbf{r})$ and the boundary value functions $g_i(\mathbf{r})$ and $g(\mathbf{r})$ is a research topic by its own. In the numerical tests of this paper, we simply set them as follows:

$$g_i(\mathbf{r}) = \begin{cases} c_i^b & \text{at } z = L_{z1} \text{ (bottom surface),} \\ c_i^t & \text{at } z = L_{z2} \text{ (top surface),} \end{cases} \quad g(\mathbf{r}) = \begin{cases} u_b & \text{at } z = L_{z1} \text{ (bottom surface),} \\ u_t & \text{at } z = L_{z2} \text{ (top surface),} \end{cases} \quad (8)$$

and

$$\mathcal{D}_i(\mathbf{r}) = \begin{cases} D_{i,b}, & z < Z1 \text{ or } z > Z2 \text{ (bulk part),} \\ D_{i,c} + (D_{i,c} - D_{i,b})f_i(\mathbf{r}), & Z2 - \eta \leq z \leq Z2 \text{ (top buffer part),} \\ D_{i,c}, & Z1 + \eta \leq z \leq Z2 - \eta \text{ (channel pore),} \\ D_{i,c} + (D_{i,c} - D_{i,b})f_b(\mathbf{r}), & Z1 \leq z \leq Z1 + \eta \text{ (bottom buffer part),} \end{cases} \quad (9)$$

where u_b and u_t denote two given boundary potential functions on the bottom and top surfaces of the box domain Ω , respectively, $D_{i,b}$ and $D_{i,c}$ are the diffusion constants of species i in the bulk and channel pore regions, respectively, f_b and f_t are two interpolation functions such that each diffusion function is sufficiently smooth over the whole solvent region D_s , and η is a parameter for adjusting the buffering region size. Two particular expressions of f_b and f_t can be found in [31, eq. (27)], and $D_{i,c}$ can be estimated by

$$D_{i,c} = \theta D_{i,b} \quad \text{for } 0 < \theta \leq 1. \quad (10)$$

We can set $u_t = 0$ and $u_b = 0$ when the box dimension in the z -axis direction is sufficiently large according to the fact that $u(\mathbf{r}) \rightarrow 0$ as $|z| \rightarrow \infty$ for $\mathbf{r} = (x, y, z)$.

3. A PNPic solution decomposition reformulation

In this section, we present a solution decomposition reformulation for the PNPic model to avoid the singularity difficulty caused by atomic charges. One key step of this reformulation is to decompose the electrostatic potential function u as a sum of three component potential functions, G , Ψ , and $\tilde{\Phi}$,

$$u(\mathbf{r}) = G(\mathbf{r}) + \Psi(\mathbf{r}) + \tilde{\Phi}(\mathbf{r}) \quad \forall \mathbf{r} \in \Omega, \quad (11)$$

such that G , Ψ , and $\tilde{\Phi}$ represent the potentials induced by the atomic charges from the protein region D_p , the related boundary and interface conditions, and the ionic charges from the solvent region D_s , respectively. Following what is done in [35], we define G by the Poisson dielectric equation over the whole space \mathbb{R}^3 ,

$$-\epsilon_p \Delta G(\mathbf{r}) = \alpha \sum_{j=1}^{n_p} z_j \delta_{\mathbf{r}_j}, \quad \mathbf{r} \in \mathbb{R}^3, \quad (12)$$

and find G analytically in the expression

$$G(\mathbf{r}) = \frac{\alpha}{4\pi\epsilon_p} \sum_{j=1}^{n_p} \frac{z_j}{|\mathbf{r} - \mathbf{r}_j|}. \quad (13)$$

We next describe the deviation of Ψ and $\tilde{\Phi}$ from the Poisson dielectric interface boundary value problem (6). Clearly, substituting (11) to the Neumann boundary value condition (6h) gives

$$\frac{\partial \Psi(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} + \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} + \frac{\partial \tilde{\Phi}(t, \mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} = 0, \quad \mathbf{s} \in \Gamma_N. \quad (14)$$

From the above equation we can get that Ψ satisfies the Neumann boundary value condition

$$\frac{\partial \Psi(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} = -\frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})}, \quad \mathbf{s} \in \Gamma_N,$$

provided that we set $\tilde{\Phi}$ to satisfy the Neumann boundary value condition

$$\frac{\partial \tilde{\Phi}(t, \mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} = 0, \quad \mathbf{s} \in \Gamma_N.$$

Applying (11) and the above two Neumann boundary conditions to the Poisson dielectric interface boundary value problem (6), we can obtain that Ψ satisfies the following linear interface boundary value problem,

$$\left\{ \begin{array}{ll} \Delta\Psi(\mathbf{r}) = 0, & \mathbf{r} \in D_m \cup D_p \cup D_s, \\ \Psi(\mathbf{s}^-) = \Psi(\mathbf{s}^+), & \epsilon_p \frac{\partial\Psi(\mathbf{s}^-)}{\partial\mathbf{n}_p(\mathbf{s})} = \epsilon_s \frac{\partial\Psi(\mathbf{s}^+)}{\partial\mathbf{n}_p(\mathbf{s})} + (\epsilon_s - \epsilon_p) \frac{\partial G(\mathbf{s})}{\partial\mathbf{n}_p(\mathbf{s})}, \quad \mathbf{s} \in \Gamma_p, \\ \Psi(\mathbf{s}^-) = \Psi(\mathbf{s}^+), & \epsilon_m \frac{\partial\Psi(\mathbf{s}^-)}{\partial\mathbf{n}_m(\mathbf{s})} = \epsilon_s \frac{\partial\Psi(\mathbf{s}^+)}{\partial\mathbf{n}_m(\mathbf{s})} + (\epsilon_s - \epsilon_m) \frac{\partial G(\mathbf{s})}{\partial\mathbf{n}_m(\mathbf{s})} + \tau\sigma, \quad \mathbf{s} \in \Gamma_m, \\ \Psi(\mathbf{s}^-) = \Psi(\mathbf{s}^+), & \epsilon_p \frac{\partial\Psi(\mathbf{s}^-)}{\partial\mathbf{n}_p(\mathbf{s})} = \epsilon_m \frac{\partial\Psi(\mathbf{s}^+)}{\partial\mathbf{n}_p(\mathbf{s})} + (\epsilon_m - \epsilon_p) \frac{\partial G(\mathbf{s})}{\partial\mathbf{n}_p(\mathbf{s})}, \quad \mathbf{s} \in \Gamma_{pm}, \\ \Psi(\mathbf{s}) = g(\mathbf{s}) - G(\mathbf{s}), & \mathbf{s} \in \Gamma_D, \\ \frac{\partial\Psi(\mathbf{s})}{\partial\mathbf{n}_b(\mathbf{s})} = -\frac{\partial G(\mathbf{s})}{\partial\mathbf{n}_b(\mathbf{s})}, & \mathbf{s} \in \Gamma_N, \end{array} \right. \quad (15)$$

and $\tilde{\Phi}$ satisfies the following linear interface boundary value problem:

$$\left\{ \begin{array}{ll} \Delta\tilde{\Phi}(\mathbf{r}) = 0, & \mathbf{r} \in D_m \cup D_p, \\ -\epsilon_s \Delta\tilde{\Phi}(\mathbf{r}) = \beta \sum_{i=1}^n Z_i c_i(\mathbf{r}), & \mathbf{r} \in D_s, \\ \tilde{\Phi}(\mathbf{s}^+) = \tilde{\Phi}(\mathbf{s}^-), & \epsilon_s \frac{\partial\tilde{\Phi}(\mathbf{s}^+)}{\partial\mathbf{n}_p(\mathbf{s})} = \epsilon_p \frac{\partial\tilde{\Phi}(\mathbf{s}^-)}{\partial\mathbf{n}_p(\mathbf{s})}, \quad \mathbf{s} \in \Gamma_p, \\ \tilde{\Phi}(\mathbf{s}^+) = \tilde{\Phi}(\mathbf{s}^-), & \epsilon_s \frac{\partial\tilde{\Phi}(\mathbf{s}^+)}{\partial\mathbf{n}_m(\mathbf{s})} = \epsilon_m \frac{\partial\tilde{\Phi}(\mathbf{s}^-)}{\partial\mathbf{n}_m(\mathbf{s})}, \quad \mathbf{s} \in \Gamma_m, \\ \tilde{\Phi}(\mathbf{s}^-) = \tilde{\Phi}(\mathbf{s}^+), & \epsilon_p \frac{\partial\tilde{\Phi}(\mathbf{s}^-)}{\partial\mathbf{n}_p(\mathbf{s})} = \epsilon_m \frac{\partial\tilde{\Phi}(\mathbf{s}^+)}{\partial\mathbf{n}_p(\mathbf{s})}, \quad \mathbf{s} \in \Gamma_{pm}, \\ \tilde{\Phi}(\mathbf{s}) = 0, & \mathbf{s} \in \Gamma_D, \\ \frac{\partial\tilde{\Phi}(\mathbf{s})}{\partial\mathbf{n}_b(\mathbf{s})} = 0, & \mathbf{s} \in \Gamma_N. \end{array} \right. \quad (16)$$

Here, $\frac{\partial G(\mathbf{s})}{\partial\mathbf{n}(\mathbf{s})} = \nabla G(\mathbf{s}) \cdot \mathbf{n}(\mathbf{s})$, and ∇G can be found in the expression

$$\nabla G(\mathbf{s}) = -\frac{\alpha}{4\pi\epsilon_p} \sum_{j=1}^{n_p} z_j \frac{(\mathbf{s} - \mathbf{r}_j)}{|\mathbf{s} - \mathbf{r}_j|^3}. \quad (17)$$

Similarly to what is done in [35], we then can show that the sum of G , Ψ , and $\tilde{\Phi}$ gives a solution of the Poisson dielectric interface boundary value problem (6). Since the singularity points of u are all located at atomic position points \mathbf{r}_j , and have been collected by G , Ψ and $\tilde{\Phi}$ must be well defined without any singularity problem.

Note that the equation (16) of $\tilde{\Phi}$ is independent of both G and Ψ . Hence, we can first calculate G , Ψ , and their gradient vectors $\nabla G(\mathbf{r})$ and $\nabla\Psi(\mathbf{r})$, and then treat them as known functions. To this end, we can set $w = G + \Psi$ to rewrite the Nernst-Planck boundary value problems of (5) as the equations of c_i and $\tilde{\Phi}$: For $i = 1, 2, \dots, n$,

$$\nabla \cdot \mathcal{D}_i(\mathbf{r}) \left[\nabla c_i(\mathbf{r}) + Z_i c_i(\mathbf{r}) \nabla w(\mathbf{r}) + Z_i c_i(\mathbf{r}) \nabla \tilde{\Phi}(\mathbf{r}) \right] = 0, \quad \mathbf{r} \in D_s, \quad (18a)$$

$$\frac{\partial c_i(\mathbf{s})}{\partial\mathbf{n}_s(\mathbf{s})} + Z_i c_i(\mathbf{s}) \frac{\partial(w + \tilde{\Phi})(\mathbf{s})}{\partial\mathbf{n}_s(\mathbf{s})} = 0, \quad \mathbf{s} \in \Gamma_p \cup \Gamma_m, \quad (18b)$$

$$c_i(\mathbf{s}) = g_i(\mathbf{s}), \quad \mathbf{s} \in \Gamma_D, \quad (18c)$$

$$\frac{\partial c_i(\mathbf{s})}{\partial\mathbf{n}_b(\mathbf{s})} = 0, \quad \mathbf{s} \in \Gamma_N \cap \partial D_s. \quad (18d)$$

The above system can be much easier to solve numerically than the PNPic system defined by (5) and (6) since it avoids the solution singularity problem induced by the Dirac delta distributions for atomic charges. A combination of (18) with (16) gives a system of equations for computing $\tilde{\Phi}$ and c_i . After $\tilde{\Phi}$ is found, we construct u by the solution decomposition formula (11).

4. Variational formulations

In this section, we reformulate the boundary value problems (5), (15), and (16) into variational problems as required to develop a finite element iterative algorithm for solving the PNPic model numerically.

Let $H^1(\Omega)$ and $H^1(D_s)$ denote the two regular Sobolev function spaces defined in the box domain Ω and solvent region D_s , respectively [1]. We define their subspaces U_0 and V_0 by

$$U_0 = \{u \in H^1(\Omega) \mid u = 0 \text{ on } \Gamma_D\}, \quad V_0 = \{v \in H^1(D_s) \mid v = 0 \text{ on } \Gamma_D\}. \quad (19)$$

A variational formulation of the linear interface boundary value problem (15) is presented in Theorem 4.1.

Theorem 4.1. *The interface boundary value problem (15) can be reformulated as the variational problem: Find $\Psi \in H^1(\Omega)$ satisfying $\Psi = g - G$ on Γ_D such that*

$$\begin{aligned} a(\Psi, v) = & (\epsilon_s - \epsilon_p) \int_{\Gamma_p} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} + (\epsilon_s - \epsilon_m) \int_{\Gamma_m} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_m(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} + (\epsilon_m - \epsilon_p) \int_{\Gamma_{pm}} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} \\ & - \epsilon_m \int_{\Gamma_N \cap \partial D_m} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} - \epsilon_s \int_{\Gamma_N \cap \partial D_s} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} + \tau \int_{\Gamma_m} \sigma v(\mathbf{s}) d\mathbf{s} \quad \forall v \in U_0, \end{aligned} \quad (20)$$

where $a(\cdot, \cdot)$ is defined by

$$a(\Psi, v) = \epsilon_p \int_{D_p} \nabla \Psi \cdot \nabla v d\mathbf{r} + \epsilon_m \int_{D_m} \nabla \Psi \cdot \nabla v d\mathbf{r} + \epsilon_s \int_{D_s} \nabla \Psi \cdot \nabla v d\mathbf{r}. \quad (21)$$

Proof. For a test function $v \in U_0$, we multiply the first equation of (15) with v , integrate it over D_p , D_m , and D_s respectively, and then add them together to get

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

Applying the Green's first identity to the above three integrals, we can rewrite the above equation as

$$\begin{aligned} & \epsilon_p \int_{D_p} \nabla \Psi(\mathbf{r}) \cdot \nabla v(\mathbf{r}) d\mathbf{r} + \epsilon_m \int_{D_m} \nabla \Psi(\mathbf{r}) \cdot \nabla v(\mathbf{r}) d\mathbf{r} + \epsilon_s \int_{D_s} \nabla \Psi(\mathbf{r}) \cdot \nabla v(\mathbf{r}) d\mathbf{r} \\ & = \epsilon_p \int_{\partial D_p} \frac{\partial \Psi(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} + \epsilon_m \int_{\partial D_m} \frac{\partial \Psi(\mathbf{s})}{\partial \mathbf{n}_m(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} + \epsilon_s \int_{\partial D_s} \frac{\partial \Psi(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} v(\mathbf{s}) d\mathbf{s}, \end{aligned} \quad (22)$$

where ∂D_p , ∂D_m , and ∂D_s denote the boundaries of D_p , D_m , and D_s , respectively. Note that the normal vectors \mathbf{n}_s , \mathbf{n}_p , \mathbf{n}_m , and \mathbf{n}_b have the following relations:

$$\mathbf{n}_s = -\mathbf{n}_p \text{ on } \Gamma_p, \quad \mathbf{n}_s = -\mathbf{n}_m \text{ on } \Gamma_m, \quad \mathbf{n}_m = -\mathbf{n}_p \text{ on } \Gamma_{pm}, \quad \mathbf{n}_m = \mathbf{n}_b \text{ on } \Gamma_N \cap \partial D_m, \quad \mathbf{n}_s = \mathbf{n}_b \text{ on } \Gamma_N \cap \partial D_s,$$

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

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

Hence, the three surface integrals of (22) can be reformulated as

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

where the boundary value condition $v = 0$ on Γ_D has been used. Applying the above expressions and the interface and Neumann conditions of (15) to (22), we can get (20). This completes the proof. \square

Similarly, the interface boundary value problem (16) can be formulated as the variational problem:

$$\text{Find } \tilde{\Phi} \in U_0 \text{ such that } a(\tilde{\Phi}, v) = \beta \sum_{j=1}^n Z_j \int_{D_s} c_j(\mathbf{r})v(\mathbf{r})d\mathbf{r} \quad \forall v \in U_0, \quad (23)$$

where $a(\cdot, \cdot)$ is defined in (21).

We next consider a variational formulation of (5). We multiply a test function $v_i \in V_0$ on the both sides of the first equation of (5), integrate on the solvent region D_s , and use Green's first identity to get

$$\begin{aligned} 0 &= \int_{D_s} \nabla \cdot D_i [\nabla c_i + Z_i c_i \nabla u(\mathbf{r})] v_i d\mathbf{r} \\ &= \int_{\partial D_s} D_i \left(\frac{\partial c_i(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} + Z_i c_i \frac{\partial u(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} \right) v_i(\mathbf{s}) ds - \int_{D_s} D_i (\nabla c_i + Z_i c_i \nabla u) \nabla v_i d\mathbf{r}. \end{aligned} \quad (24)$$

By the Neumann boundary conditions (5d) and (6h) on Γ_N , Robin boundary value condition (5b), and $v_i = 0$ on Γ_D , the surface integral of (24) can be found to be zero so that we obtain a variational problem of (5) as follows:

Find $c_i \in H^1(D_s)$ satisfying $c_i = g_i$ on Γ_D such that

$$\int_{D_s} D_i [\nabla c_i(\mathbf{r}) + Z_i c_i(\mathbf{r}) \nabla u(\mathbf{r})] \nabla v_i d\mathbf{r} = 0 \quad \forall v_i \in V_0, \quad i = 1, 2, \dots, n, \quad (25)$$

subject to the sign constraint conditions $c_i \geq 0$ for $i = 1, 2, \dots, n$.

In calculation, it is often appropriate to set $\epsilon_m = \epsilon_p$. In this case, the variational problem (20) can be simplified as follows:

Find $\Psi \in H^1(\Omega)$ satisfying $\Psi = g - G$ on Γ_D such that

$$a(\Psi, v) = (\epsilon_s - \epsilon_p) \int_{\Gamma} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}(\mathbf{s})} v(\mathbf{s}) ds - \epsilon_p \int_{\Gamma_N \cap \partial D_m} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) ds - \epsilon_s \int_{\Gamma_N \cap \partial D_s} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) ds + \tau \int_{\Gamma_m} \sigma v(\mathbf{s}) ds \quad \forall v \in V_0, \quad (26)$$

where \mathbf{n} denotes the unit outward normal direction of the solute region $D_{pm} = D_p \cup D_m$, and $\Gamma = \Gamma_m \cup \Gamma_p$, which gives the interface between D_{pm} and D_s .

To avoid the surface integral calculation on Γ , we use the Green's first identity, $\Delta G = 0$ in D_s , $\partial D_s = \Gamma \cup \Gamma_D \cup (\Gamma_N \cap \partial D_s)$, and $v = 0$ on Γ_D to get that

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

Since $\mathbf{n}_s = -\mathbf{n}$ on Γ , the above expression can be rewritten as

$$\int_{\Gamma} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}(\mathbf{s})} v(\mathbf{s}) ds = - \int_{D_s} \nabla G(\mathbf{r}) \cdot \nabla v(\mathbf{r}) d\mathbf{r} + \int_{\Gamma_N \cap \partial D_s} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) ds.$$

Using $\Gamma_N = (\Gamma_N \cap \partial D_s) \cup (\Gamma_N \cap \partial D_m)$, we then get

$$\int_{\Gamma_N \cap \partial D_s} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) ds + \int_{\Gamma_N \cap \partial D_m} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) ds = \int_{\Gamma_N} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) ds.$$

Substituting the above two expressions to (26), we can further simplify the variational problem (20) of Ψ as follows:

Find $\Psi \in H^1(\Omega)$ satisfying $\Psi = g - G$ on Γ_D such that

$$a(\Psi, v) = (\epsilon_p - \epsilon_s) \int_{D_s} \nabla G(\mathbf{r}) \cdot \nabla v(\mathbf{r}) d\mathbf{r} + \tau \int_{\Gamma_m} \sigma v(\mathbf{s}) ds - \epsilon_p \int_{\Gamma_N} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) ds \quad \forall v \in U_0. \quad (27)$$

5. A PNPic finite element solver

In this section, we use the solution decomposition (11), the classic nonlinear successive relaxation iterative techniques [26], and the Slotboom variables transformation [30] to develop a PNPic finite element solver in order to improve numerical accuracy, reduce computer memory requirement, enhance numerical stability, and retain the positivity of an ionic concentration. We also present an effective modified Newton iterative algorithm for solving each related nonlinear equation generated from the Slotboom variables transformation.

Let \mathcal{U} and \mathcal{V} denote the two Lagrange finite element function spaces with respect to an interface fitted irregular tetrahedral mesh, Ω_h , of Ω and a tetrahedral mesh, $D_{s,h}$, of D_s , respectively. We construct them as the finite dimensional subspaces of the function spaces $H^1(\Omega)$ and $H^1(D_s)$, respectively. We then define the finite dimensional subspaces \mathcal{U}_0 and \mathcal{V}_0 of \mathcal{U} and \mathcal{V} by

$$\mathcal{U}_0 = \{u \in \mathcal{U} \mid u = 0 \text{ on } \Gamma_D\}, \quad \mathcal{V}_0 = \{v \in \mathcal{V} \mid v = 0 \text{ on } \Gamma_D\}.$$

We also extract $D_{s,h}$ from Ω_h as a sub-mesh of Ω_h so that a restriction operator, $\mathcal{R} : \mathcal{U} \rightarrow \mathcal{V}$, and a prolongation operator, $\mathcal{P} : \mathcal{V} \rightarrow \mathcal{U}$, can be easily obtained to carry out functions operations for the functions from \mathcal{U} and \mathcal{V} .

Using the above finite element function spaces and the variational problems (23) and (25), we can obtain a system of $n + 1$ finite element equations for computing $\tilde{\Phi}$ and c_i as follows:

Find $\tilde{\Phi} \in \mathcal{U}_0$ and $c_i \in \mathcal{V}$ satisfying $c_i = g_i$ on Γ_D for $i = 1, 2, \dots, n$ such that

$$\begin{cases} \int_{D_s} D_i \left[\nabla c_i + Z_i c_i \nabla \mathcal{R}(G + \Psi + \tilde{\Phi}) \right] \nabla v_i d\mathbf{r} = 0 & \forall v_i \in \mathcal{V}_0, \quad i = 1, 2, \dots, n, \\ a(\tilde{\Phi}, v) - \beta \sum_{j=1}^n Z_j \int_{D_s} \mathcal{P} c_j v d\mathbf{r} = 0, & \forall v \in \mathcal{U}_0, \end{cases} \quad (28)$$

where G has been calculated by using (13), and Ψ has been found numerically as a solution of a linear finite element equation defined as follows: Find $\Psi \in \mathcal{U}$ satisfying $\Psi = g - G$ on Γ_D such that

$$\begin{aligned} a(\Psi, v) = & (\epsilon_s - \epsilon_p) \int_{\Gamma_p} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} + (\epsilon_s - \epsilon_m) \int_{\Gamma_m} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_m(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} + (\epsilon_m - \epsilon_p) \int_{\Gamma_{pm}} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} \\ & - \epsilon_m \int_{\Gamma_N \cap \partial D_m} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} - \epsilon_s \int_{\Gamma_N \cap \partial D_s} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} v(\mathbf{s}) d\mathbf{s} + \tau \int_{\Gamma_m} \sigma v(\mathbf{s}) d\mathbf{s} \quad \forall v \in \mathcal{U}_0, \end{aligned} \quad (29)$$

We recall that the Slotboom variable transformation is defined by

$$c_i = e^{-Z_i u} \bar{c}_i, \quad i = 1, 2, \dots, n, \quad (30)$$

where \bar{c}_i denotes the i th Slotboom variable. By the above transformation, we can get the identities

$$\nabla c_i + Z_i c_i \nabla u = e^{-Z_i u} \nabla \bar{c}_i, \quad i = 1, 2, \dots, n. \quad (31)$$

Using the boundary conditions (5b), (5d), (6h), and the fact that $e^{-Z_i u} > 0$ for any u in Ω , we further can obtain the Dirichlet boundary conditions

$$\bar{c}_i(\mathbf{s}) = \bar{g}_i(\mathbf{s}) \quad \forall \mathbf{s} \in \Gamma_D \quad \text{with } \bar{g}_i = e^{Z_i g} g_i,$$

and the Neumann boundary conditions

$$\frac{\partial \bar{c}_i(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} = 0, \quad \mathbf{s} \in \Gamma_N \cap \partial D_s, \quad \frac{\partial \bar{c}_i(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} = 0, \quad \mathbf{s} \in \Gamma_p \cup \Gamma_m.$$

To this end, we can transform the first n equations of (28) as the following equations:

$$\int_{D_s} \mathcal{D}_i e^{-Z_i \mathcal{R}(G + \Psi + \tilde{\Phi})} \nabla \bar{c}_i \nabla v_i d\mathbf{r} = 0 \quad \forall v_i \in \mathcal{V}_0, \quad i = 1, 2, \dots, n, \quad (32)$$

Since each of the above equations has a positive solution [9, page 27], from (30) it implies that each c_i is positive. Moreover, the above equations do not involve any gradient of a potential function.

Substituting (30) to the last equation of (28), we obtain a nonlinear equation of $\tilde{\Phi}$ as follows:

$$a(\tilde{\Phi}, v) - \beta \sum_{j=1}^n Z_j \int_{D_s} e^{-Z_i(G+\Psi+\tilde{\Phi})} \mathcal{P} \tilde{c}_j v d\mathbf{r} = 0 \quad \forall v \in \mathcal{U}_0. \quad (33)$$

Consequently, a combination of (32) and (33) gives a nonlinear system for computing $\tilde{\Phi}$ and \tilde{c}_i as follows:

Find $\tilde{\Phi} \in \mathcal{U}_0$ and $\tilde{c}_i \in \mathcal{V}$ satisfying $\tilde{c}_i = \bar{g}_i$ on Γ_D for $i = 1, 2, \dots, n$ such that

$$\begin{cases} \int_{D_s} \mathcal{D}_i e^{-Z_i \mathcal{R}(G+\Psi+\tilde{\Phi})} \nabla \tilde{c}_i \nabla v_i d\mathbf{r} = 0 & \forall v_i \in \mathcal{V}_0, i = 1, 2, \dots, n, \\ a(\tilde{\Phi}, v) - \beta \sum_{j=1}^n Z_j \int_{D_s} e^{-Z_i(G+\Psi+\tilde{\Phi})} \mathcal{P} \tilde{c}_j v d\mathbf{r} = 0 & \forall v \in \mathcal{U}_0, \end{cases} \quad (34)$$

where G and Ψ have been known, and $\bar{g}_i = e^{Z_i g_i}$.

After finding $\tilde{\Phi}$ and \tilde{c}_i , we use the formulas (11) and (30) to recover the electrostatic potential u and ionic concentrations c_i , yielding a finite element solution of our PNPic model defined in (5) and (6).

We now present a relaxation iterative scheme for solving the nonlinear finite element system (34).

Let $(\tilde{\Phi}^k, \tilde{c}^k)$ denote the k th iterate of this relaxation iterative scheme for $k \geq 0$ with $\tilde{c}^k = (\tilde{c}_1^k, \tilde{c}_2^k, \dots, \tilde{c}_n^k)$. When initial guesses $\tilde{\Phi}^0$ and \tilde{c}_i^0 are given, we define the updates \tilde{c}_i^{k+1} and $\tilde{\Phi}^{k+1}$ by

$$\tilde{c}_i^{k+1} = \tilde{c}_i^k + \omega(\bar{p}_i - \tilde{c}_i^k), \quad i = 1, 2, \dots, n, \quad (35)$$

$$\tilde{\Phi}^{k+1} = \tilde{\Phi}^k + \omega(\bar{q} - \tilde{\Phi}^k), \quad (36)$$

where ω is a relaxation parameter between 0 and 2, \bar{p}_i is a solution of the linear variational problem:

$$\text{Find } \bar{p}_i \in \mathcal{V} \text{ satisfying } \bar{p}_i = \bar{g}_i \text{ on } \Gamma_D \text{ such that } \int_{D_s} \mathcal{D}_i e^{-Z_i \mathcal{R}(G+\Psi+\tilde{\Phi}^k)} \nabla \bar{p}_i \nabla v_i d\mathbf{r} = 0 \quad \forall v_i \in \mathcal{V}_0, \quad (37)$$

and \bar{q} is a solution of the nonlinear variational problem:

$$\text{Find } \bar{q} \in \mathcal{U}_0 \text{ such that } a(\bar{q}, v) - \beta \sum_{i=1}^n Z_i \int_{D_s} e^{-Z_i(G+\Psi+\tilde{\Phi}^k)} \mathcal{P} \tilde{c}_i^{k+1} v d\mathbf{r} = 0 \quad \forall v \in \mathcal{U}_0. \quad (38)$$

We define the iteration termination rule by

$$\|\tilde{\Phi}^{k+1} - \tilde{\Phi}^k\| < \epsilon \quad \text{and} \quad \max_{1 \leq i \leq n} \|\tilde{c}_i^{k+1} - \tilde{c}_i^k\| < \epsilon, \quad (39)$$

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

We next present a good selection of initial guesses $\tilde{\Phi}^0$ and \tilde{c}_i^0 .

When we set the slotboom variable $\tilde{c}_i = c_i^b$, which is a bulk concentration of species i , the Slotboom variable transformation (30) is reduced to the classical Boltzmann distribution

$$c_i = c_i^b e^{-Z_i u}, \quad i = 1, 2, \dots, n, \quad (40)$$

and the nonlinear equation (33) is simplified as a nonlinear equation of $\tilde{\Phi}$ as follows:

$$\text{Find } \tilde{\Phi} \in \mathcal{U}_0 \text{ such that } a(\tilde{\Phi}, v) - \beta \sum_{j=1}^n Z_j c_j^b \int_{D_s} e^{-Z_i(G+\Psi+\tilde{\Phi})} v d\mathbf{r} = 0 \quad \forall v \in \mathcal{U}_0. \quad (41)$$

Interestingly, it can be easily shown that adding a solution of the above equation with G and Ψ gives a solution of

a Poisson-Boltzmann (PB) ion channel model as follows:

$$\left\{ \begin{array}{ll} -\epsilon_p \Delta u(\mathbf{r}) = \alpha \sum_{j=1}^{n_p} z_j \delta_{\mathbf{r}_j}, & \mathbf{r} \in D_p, \\ -\epsilon_m \Delta u(\mathbf{r}) = 0, & \mathbf{r} \in D_m, \\ \epsilon_s \Delta u(\mathbf{r}) + \beta \sum_{i=1}^n Z_i c_i^b e^{-Z_i u(\mathbf{r})} = 0, & \mathbf{r} \in D_s, \\ u(\mathbf{s}^-) = u(\mathbf{s}^+), \quad \epsilon_p \frac{\partial u(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_s \frac{\partial u(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})}, & \mathbf{s} \in \Gamma_p, \\ u(\mathbf{s}^-) = u(\mathbf{s}^+), \quad \epsilon_m \frac{\partial u(\mathbf{s}^+)}{\partial \mathbf{n}_m(\mathbf{s})} = \epsilon_s \frac{\partial u(\mathbf{s}^-)}{\partial \mathbf{n}_m(\mathbf{s})} + \tau \sigma, & \mathbf{s} \in \Gamma_m, \\ u(\mathbf{s}^-) = u(\mathbf{s}^+), \quad \epsilon_p \frac{\partial u(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_m \frac{\partial u(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})}, & \mathbf{s} \in \Gamma_{pm}, \\ u(\mathbf{s}) = g(\mathbf{s}), & \mathbf{s} \in \Gamma_D, \\ \frac{\partial u(\mathbf{s})}{\partial \mathbf{n}_b(\mathbf{s})} = 0, & \mathbf{s} \in \Gamma_N, \end{array} \right. \quad (42)$$

Hence, we denote the solution of (41) by $\tilde{\Phi}^{PB}$, and set $\tilde{c}_i^0 = c_i^b$ and $\tilde{\Phi}^0 = \tilde{\Phi}^{PB}$ as a good selection of initial guess for the nonlinear iterative scheme defined in (35) and (36).

Clearly, the efficiency of our nonlinear iterative scheme depends on that of a numerical algorithm for solving the nonlinear equations (33) and (41) as well as that of a numerical algorithm for all related linear finite element equations. Since there exist several effective numerical algorithms for solving a linear finite element equation in the literature, we here only present a modified Newton iterative method for solving the nonlinear variational problem (38) including the nonlinear variational problem (46) occurred in the initial step $k = 0$.

Let q_k^j denote the j -th iterative approximation to a solution of (38). For a given q_k^j , we define the update q_k^{j+1} by

$$q_k^{j+1} = q_k^j + \xi_k^j \quad \text{for } j = 0, 1, 2, \dots, \quad (43)$$

where ξ_k^j is a solution of the linear variational problem: Find $\xi_j \in \mathcal{U}_0$ such that

$$a(\xi_j, v) + \beta \int_{D_s} \sum_{i=1}^n Z_i^2 \mathcal{P} \tilde{c}_i^{k+1} e^{-Z_i(G+\Psi+q_k^j)} \xi_j v d\mathbf{r} = \beta \int_{D_s} \sum_{i=1}^n Z_i \mathcal{P} \tilde{c}_i^{k+1} e^{-Z_i(G+\Psi+q_k^j)} v d\mathbf{r} - a(q_k^j, v) \quad \forall v \in \mathcal{U}_0, \quad (44)$$

and the initial guess q_k^0 is selected by

$$q_k^0 = \begin{cases} q^0, & \text{for } k = 0, \\ \tilde{\Phi}^k, & \text{for } k > 0. \end{cases} \quad (45)$$

Here q^0 denotes a solution of a linearized problem of (41) as follows:

$$\text{Find } q^0 \in \mathcal{U}_0 \text{ such that } a(q^0, v) + \beta \sum_{i=1}^n Z_i^2 c_i^b \int_{D_s} q^0 v d\mathbf{r} = -\beta \sum_{i=1}^n Z_i^2 c_i^b \int_{D_s} (G + \Psi) v d\mathbf{r} \quad \forall v \in \mathcal{U}_0. \quad (46)$$

In the construction of the above equation, we have used the electroneutrality condition $\sum_{i=1}^n Z_i c_i^b = 0$.

In the iterative process of (43), we use the following iteration stopping criterion

$$\|q_k^{j+1} - q_k^j\| < \epsilon \quad (\epsilon = 10^{-5} \text{ by default}). \quad (47)$$

For clarity, we summarize our nonlinear relaxation iterative algorithm in a flowchart of Figure 2.

6. Visualization of potential and concentration functions in curves

In this section, we present a scheme to visualize a three-dimensional (3D) function — either an electrostatic potential function u or an ionic concentration function c_i — over a solvent region D_s in a 2D curve with each point of the curve representing an average function value over a block of D_s . Such a curve can be valuable in the study of potential and ionic distribution profiles across membrane and in the comparison of different potential and ionic concentration functions generated by either different models or different parameter values.

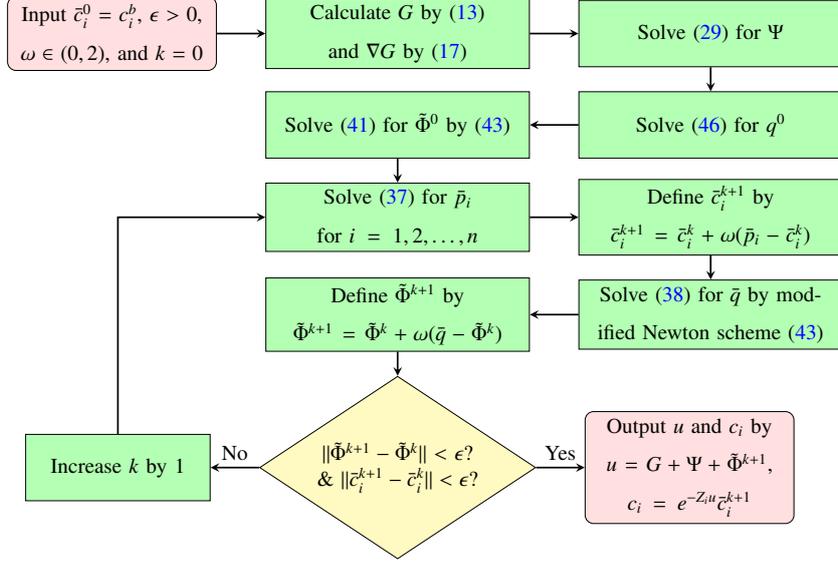


Figure 2: A flowchart of our nonlinear relaxation iterative algorithm for a finite element solution of our PNPic model, which is defined in (5) and (6). Here, u is an electrostatic potential function, c_i is a concentration function of ionic species i , c_i^b is a bulk concentration of species i , ω is a relaxation parameter between 0 and 2, and \bar{c}_i is the i th Slotboom variable. By default, each linear finite element equation of (29), (37), and (46) is solved approximately by the GMRES-ILU method.

Let \bar{h} be a quality controlling parameter of the 2D curve. We start with $m + 2$ partition numbers, $\{z^j\}$ for $j = 0, 1, 2, \dots, m + 1$, of the interval $[L_{z1}, L_{z2}]$ as follows:

$$z^0 = L_{z1}, \quad z^1 = L_{z1} + \bar{h}/2, \quad z^1 < z^2 < \dots < z^m, \quad z^m = L_{z2} - \bar{h}/2, \quad z^{m+1} = L_{z2},$$

where the set $\{z^j\}_{j=1}^m$ contains the membrane location numbers $Z1$ and $Z2$, the ion channel protein location numbers, and other location numbers that we want to know for the average values. We then construct m blocks B_j of a solvent region D_s by

$$B_j = ([z^j - \bar{h}/2, z^j + \bar{h}/2] \times [L_{x1}, L_{x2}] \times [L_{y1}, L_{y2}]) \cap D_s, \quad j = 1, 2, \dots, m, \quad (48)$$

where z^j is the mid-number of B_j in the z -axis direction, each B_j is allowed to overlap its neighboring blocks, and each B_j has a nonzero volume, denoted by $\|B_j\|$. It is clear that $\cup_{j=1}^m B_j = D_s$.

Using these blocks B_j , we define m average values c_i^j of the i -th ionic penetration function c_i by

$$c_i^j = \frac{1}{\|B_j\|} \int_{B_j} c_i(\mathbf{r}) d\mathbf{r}, \quad j = 1, 2, \dots, m. \quad (49)$$

In addition, we set $c_i^0 = c_i^b$ and $c_i^{m+1} = c_i^b$ to obtain $m + 2$ points (z^j, c_i^j) for $j = 0, 1, 2, \dots, m, m + 1$.

However, an electrostatic potential function u has positive and negative values. Thus, two curves are needed to display the distribution pattern of u — one for the positive values and the other for the negative values. To do so, we split u as the sum of a positive function u^+ and a negative function u^- by the formulas

$$u = u^+ + u^-, \quad u^+(\mathbf{r}) = \frac{u(\mathbf{r}) + |u(\mathbf{r})|}{2}, \quad u^-(\mathbf{r}) = \frac{u(\mathbf{r}) - |u(\mathbf{r})|}{2} \quad \forall \mathbf{r} \in D_s. \quad (50)$$

We then set $u_0^\pm = u_b$ and $u_{m+1}^\pm = u_t$, and calculate the m average values u_j^\pm of u^\pm by

$$u_j^\pm = \frac{1}{\|B_j\|} \int_{B_j} u^\pm(\mathbf{r}) d\mathbf{r}, \quad j = 1, 2, \dots, m, \quad (51)$$

resulting in $m + 2$ points (z^j, u_j^\pm) for $j = 0, 1, 2, \dots, m, m + 1$.

Using the points (z^j, c_i^j) and (z^j, u_j^\pm) for $j = 0, 1, 2, \dots, m, m + 1$, we can draw 2D curves as the visualization of concentration functions c_i and electrostatic potential function u over a solvent region D_s along the membrane normal direction (i.e., the z axis direction in this paper). Clearly, such two-dimensional curves are easy to view, and can be good approximations to the distribution profiles of c_i and u . Hence, they are valuable in the study of electrostatics and ionic concentrations.

The remaining question is how to calculate each volume integral of (49) and (51). Since each block B_j has a complex irregular shape (see Figure 5 for examples of $B_{j,h}$), such a volume integral is difficult to calculate. We overcome this difficulty through extracting a mesh of B_j , denoted by $B_{j,h}$, from a mesh $D_{s,h}$ of D_s . As a sub-mesh of $D_{s,h}$, $B_{j,h}$ consists of the tetrahedra from $D_{s,h}$. Thus, we can calculate c_i^j and u_j^\pm approximately as follows:

$$c_i^j \approx \frac{1}{\|B_{j,h}\|} \int_{B_{j,h}} c_i(\mathbf{r}) d\mathbf{r} = \frac{\sum_{T \in B_{j,h}} \int_T c_i(\mathbf{r}) d\mathbf{r}}{\sum_{T \in B_{j,h}} \|T\|}, \quad u_j^\pm \approx \frac{1}{\|B_{j,h}\|} \int_{B_{j,h}} u_j^\pm(\mathbf{r}) d\mathbf{r} = \frac{\sum_{T \in B_{j,h}} \int_T u^\pm(\mathbf{r}) d\mathbf{r}}{\sum_{T \in B_{j,h}} \|T\|}, \quad (52)$$

where T denotes a tetrahedron of the mesh domain $B_{j,h}$ of block B_j , and c_j is a finite element function defined on the mesh domain $D_{s,h}$. Note that $D_{s,h}$ is an irregular tetrahedral mesh (See Figure 3(b) for an example). Hence, a variable value of the parameter \bar{h} may be needed to ensure that each $B_{j,h}$ is a solid volume domain without any hole.

7. Numerical results

We implemented our finite element iterative algorithm as outlined in Figure 2 as a Python software package based on the state-of-the-art finite element library from the FEniCS project [19] and the PB finite element solver program package reported in [35]. In the software package, each related linear finite element equation is solved by a commonly-used iterative method — the GMRES-ILU method. The ion channel finite element mesh program package reported in [17] is the only one available in the public domain. We adapted it to our software package for generating an interface fitted irregular tetrahedral mesh of a box domain Ω .

To demonstrate the performance of our software package and validate our PNPic model, we did numerical tests on a mouse voltage-dependent anion-channel isoform (mVDAC1) in a salt solution of up to four ionic species. A crystallographic three-dimensional molecular structure of mVDAC1 is determined by X-ray crystallography in [33]. It can be downloaded from the Protein Data Bank (PDB) with the PDB identification number 3EMN. But, we used the one downloaded from the Orientations of Proteins in Membranes (OPM) database <https://opm.phar.umich.edu>, since the mVDAC1 molecular structure from this database has been rotated and embedded in membrane properly with the normal direction of the membrane being in the z -axis direction and the membrane location numbers Z1 and Z2 being given as $Z1 = -12$ and $Z2 = 12$. We then derived a PQR file of mVDAC1 from the PDB2PQR web server (<http://server.poissonboltzmann.org/pdb2pqr>) to get the missed data in a PDB file such as the hydrogen atoms, the atomic charge numbers, and atomic radii. Such a PQR file is a required input protein data file for implementing our software package. In biology, the voltage-dependent anion channel is the most abundant protein on outer mitochondrial membrane as the main conduit for the entry and exit of ionic species (ions and metabolites) across the outer membrane of the mitochondria. It plays a crucial role in regulating cell survival and cell death and characterizing health and diseases [4, 23, 29]. Since mVDAC1 has been known to have the anion-selectivity property, it provides us with a good test case for us to validate our PNPic model.

We defined a cubic box domain Ω with side length 100 by

$$L_{x1} = -50, \quad L_{x2} = 50, \quad L_{y1} = -50, \quad L_{y2} = 50, \quad L_{z1} = -50, \quad L_{z2} = 50.$$

Using $Z1 = -12$, $Z2 = 12$, and the PQR file, we generated an interface fitted irregular tetrahedral mesh Ω_h with 44939 mesh nodes. We then from Ω_h extracted a mesh $D_{s,h}$ of the solvent region D_s with 27412 mesh nodes.

Figure 3 give two views of the meshes of Ω_h and $D_{s,h}$ to demonstrate the geometrical shapes and complexities of Ω_h and $D_{s,h}$. Here the meshes of the protein region D_p and the membrane region D_m are also displayed in green and yellow colors, respectively.

Figure 4 displays a comparison of our selected protein region D_p with a cartoon backbone representation and a spheric space filling model of the mVDAC1 molecular structure, respectively. It shows that our selection of D_p can

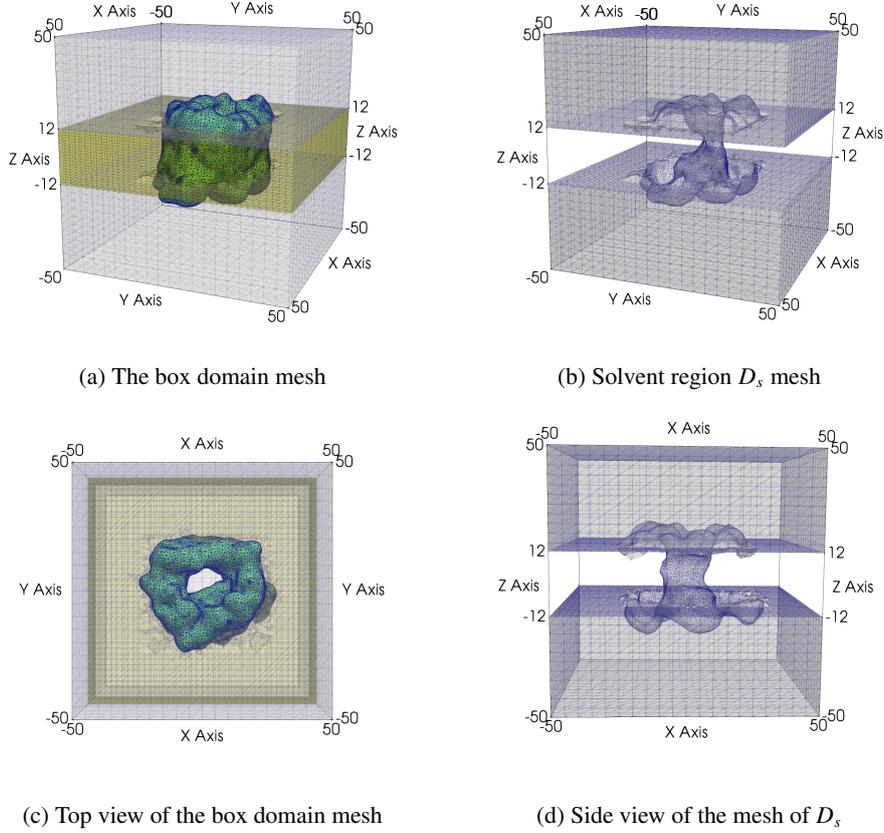


Figure 3: The interface fitted irregular tetrahedral meshes of the box domain Ω and solvent region D_s for the ion channel protein with PDB ID 3EMN used for PNPic calculation. In Plots (a,c), the meshes of the membrane region D_m and protein region D_p are colored in yellow and green, respectively, for clarity.

well fit a spacial shape of the mVDAC1 molecular structure. Here D_p was generated by using a Gaussian molecular surface [40]. In the cartoon backbone representation, alpha-helices are depicted via flat helical sheets while beta-sheets via flat level sheets. In the spheric space filling model, all of the atoms are depicted as spheres with their van der Waals radii, resulting in a van der Waals protein domain.

Figure 5 displays four block meshes $B_{j,h}$ that we extracted from the solvent mesh $D_{s,h}$ to show that the blocks B_j that are defined in (48) for the visualization of potential and concentration functions in curves (see Section 6) can have complex irregular shapes. The two blocks in Plots (b,c) were selected from the ion channel pore, the one in Plot (a) was constructed along the membrane surface $z = -12$, in which the holes were produced by the space occurred by the protein region D_p , and the one in Plot (d) was selected from the outside of the protein region D_p .

Using meshes Ω_h and $D_{s,h}$, we constructed the linear finite element function spaces \mathcal{U} and \mathcal{V} , and derived the linear and nonlinear finite element equations required for computing Ψ , $\tilde{\Phi}$, and c_i , the restriction operator \mathcal{R} , and prolongation operator \mathcal{P} . We then selected the following two mixture solutions for numerical tests:

1. A NaCaCl solution: A solution of two salts NaCl and CaCl_2 with bulk concentrations $c_1^b = 0.1$ mol/L for ionic species Na^+ , $c_2^b = 0.1$ for Ca^{2+} , and $c_3^b = 0.3$ for Cl^- .
2. A NaClKNO3 solution: A solution of two salts NaCl and KNO_3 (potassium nitrate) with bulk concentrations $c_1^b = 0.2$ mol/L for Na^+ , $c_2^b = 0.1$ for K^+ , $c_3^b = 0.2$ for Cl^- , and $c_4^b = 0.1$ for NO_3^- .

The NaCaCl solution is only used in our numerical tests to illustrate how divalent cations compete with monovalent cations for space within a region with negative potential. It is not physiological since divalent ions should be very dilute in the bulk; a 0.1 mol/L bulk concentration of calcium ions is too high for a physiological solution.

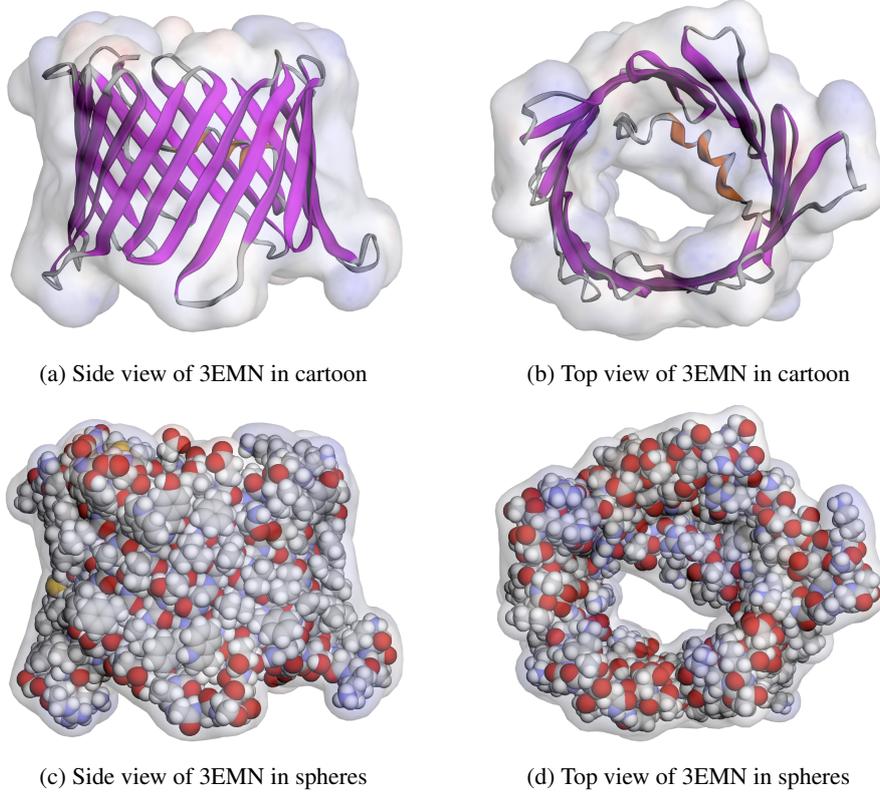


Figure 4: A comparison of our protein domain D_p with a molecular structure of the ion channel protein mVDAC1 (PDB ID 3EMN), which is depicted in cartoon (Plots (a,b)) and spheres (Plots (c,d)). In Plots (c,d), the atoms of oxygen, nitrogen, and carbon of the molecular structure have been depicted in red, blue, and grey spheres, respectively.

In all the numerical tests, we fixed $\epsilon_p = 2$, $\epsilon_m = 2$, $\epsilon_s = 80$, $u_b = 2$, $u_t = 0$, the bulk diffusion constants $D_{i,b} = 0.133$ for Na^+ , 0.196 for K^+ , 0.0793 for Ca^{2+} , 0.203 for Cl^- , and 0.19 for NO_3^- , and $\theta = 0.1$ for the channel pore diffusion constants $D_{i,c}$ defined by (10). Here the diffusion constants can be found on the website <https://www.aqion.de/site/194>. The boundary value functions g_i and g were set in (8), and the diffusion coefficient functions \mathcal{D}_i was set in (9). The iteration termination rules (39) and (47) were set with $\epsilon = 10^{-5}$. The absolute and relative residual error tolerances were set as 10^{-7} for the GMRES-ILU method for solving each finite element equation. All the numerical tests were done by using a linear finite element method on our iMac computer with one 4.2 GHz Intel core i7 processor and 64 GB memory. Test results are reported in Table 1 and Figures 6 to 9.

Table 1 reports the convergence and performance of our nonlinear iterative scheme (35) in terms of the number of iterations and computer CPU time. From the table it can be seen that the smallest number of iterations and best performance were reached at $\omega = 1$.

Figure 6 displays a convergence process of our nonlinear relaxation iterative scheme at $\omega = 1$ and that of the modified Newton iterative scheme (43) at the initial iteration step $k = 0$. From the figure it can be seen that both schemes have a fast convergence rate. Here the residual $F(\bar{q})$ of the nonlinear variational equation (38) is defined by

$$F(\bar{q}) = a(\bar{q}, v) - \beta \sum_{i=1}^n Z_i \int_{D_s} e^{-Z_i(G+\Psi+\bar{q})} \mathcal{P} \bar{c}_i^{k+1} v d\mathbf{r} \quad \forall v \in \mathcal{U}_0. \quad (53)$$

As shown in Figure 6(B), the initial residue norm $\|F(q_0^0)\|$ was as large as 10^{19} , but was quickly reduced by our modified Newton iterative scheme to 10^{-3} in 54 iterations, In these numerical tests, our selection (45) of initial iterates worked particularly well in the improvement of the performance of our modified Newton iterative algorithm; it enabled the number of modified Newton iterations to be reduced from 54 at the initial iteration step $k = 0$ to 16, 15,

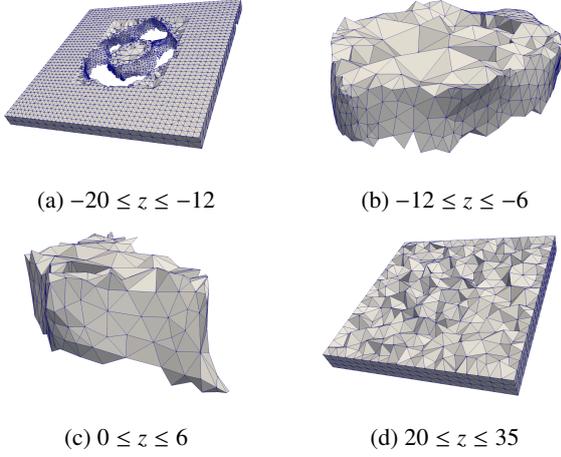


Figure 5: Four block meshes $B_{j,h}$ extracted from the solvent region mesh $D_{s,h}$ for computing the volume integrals (52). Here blocks B_j are defined in (48), and the holes in Plot (a) are produced by the protein region D_p .

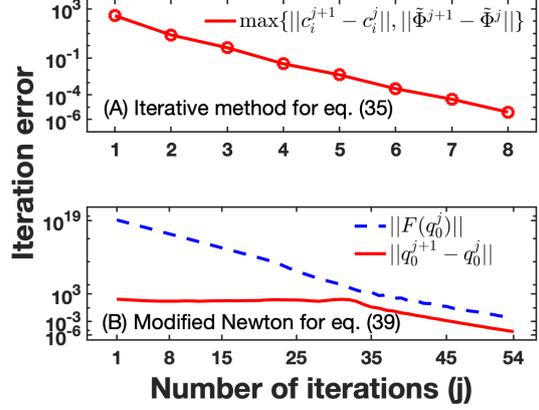


Figure 6: (A) Iteration errors for the PNPic relaxation iterative scheme defined in (35) and (36) using $\omega = 1$. (B) Iteration errors for the modified Newton iterative scheme (43) for solving the finite element equation (41). Here F is the residue defined in (53).

9, 6, 5, 3, 2, 1, respectively, at the iteration step $k = 1$ to 8.

ω	Case of NaCaCl		Case of NaClKNO3	
	Iteration number	CPU time (min.)	Iteration number	CPU time (min.)
0.9	15	7.40	11	4.45
1.0	12	6.12	8	3.41
1.1	14	7.26	12	4.70

Table 1: Convergence and performance of our nonlinear iterative scheme (35) for solving the PNPic finite element system (34) for the ion channel protein mVDAC1 in two mixtures NaCaCl and NaClKNO3. Here, $u_b = 2$, $u_t = 0$, and $\sigma = 20$.

Figure 7 displays the electrostatic potential u and the concentrations of Na^+ , Ca^{2+} , and Cl^- on a cross section ($x = 0$) of the solvent region D_s in color mapping for mVDAC1 in the NaCaCl solution. Here the membrane and protein regions are marked in yellow and green colors, respectively, to clearly highlight the values in the solvent region D_s . From Plot (a) it can be seen that the positive electrostatic potential values (in red) are distributed mostly within the channel pore, attracting many anions Cl^- to the channel pore as shown in Plot (d) while cations Na^+ and Ca^{2+} were mostly expelled from the channel pore area. Interestingly, in the area having negative electrostatic potential values, more calcium ions were found than sodium ions (see Plot (b, c)) since each calcium ion has more positive charges than a sodium ion. This shows that our PNPic model can well predict such a physical property — ionic species having larger charge numbers can be stronger in competition for space when their bulk concentrations are the same.

Figure 8 displays the case of a NaClKNO3 solution. Here we further plot the negative part u^- and positive part u^+ of the electrostatic potential u to illustrate how u^- and u^+ affect the concentration distributions of cations and anions. In this numerical test, the two cation species Na^+ and K^+ and the two anion species Cl^- and NO_3^- have the same charge numbers +1 and -1, respectively, but Na^+ and Cl^- have a bulk concentration of 0.2 mol/L, which is double the bulk concentrations of K^+ and NO_3^- . From Plots (d, e) and Plots (f, g) it can be seen that more Na^+ and Cl^- ions were found in the areas having negative and positive electrostatic potential values, respectively. This shows that our PNPic model can retain another physical property — ionic species having larger bulk concentration can be stronger in competition for space when their charge numbers are the same.

Figure 9 displays the curves that we depicted using the volume integrals of (51) and (49) with $m = 40$ for mVDAC1 in NaCaCl and NaClKNO3 mixture solutions. From these curves we now can clearly see that positive electrostatics and anions are dominating within the channel pore. In the case of the NaCaCl solution, in which the two cation species

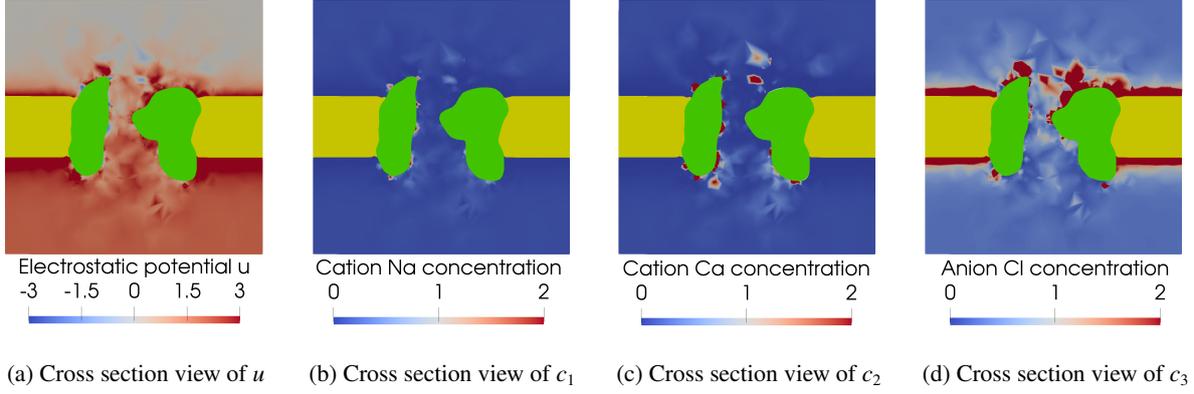


Figure 7: A color mapping of of the electrostatic potential u and the concentrations c_i produced by our PNPic model for mVDAC1 in a NaCaCl solution on a cross-section ($x = 0$). Here, the protein and membrane regions are colored in green and yellow, and the units of concentrations and electrostatic potential u are given in mol/L and $k_B T / e_c$ (≈ 0.026 volts), respectively.

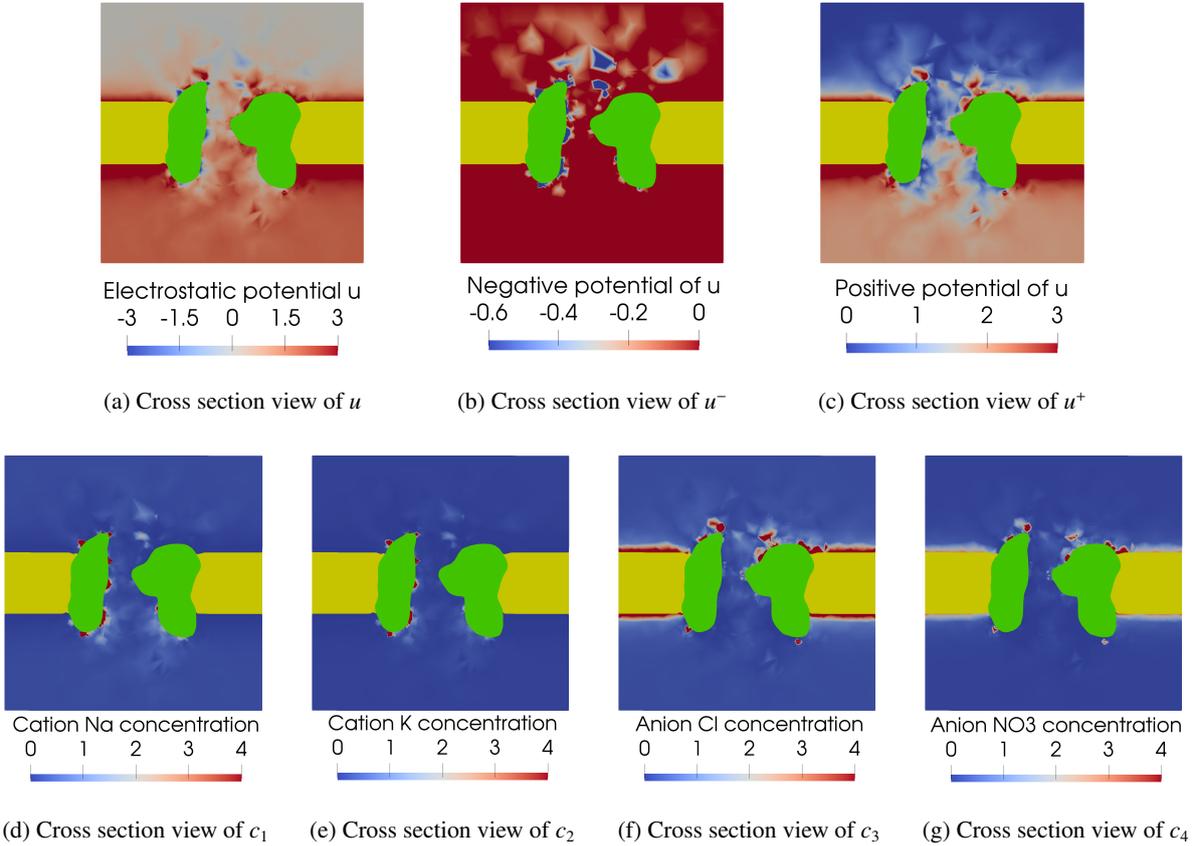


Figure 8: Color mappings of the electrostatic potential u , the negative potential u^- , the positive potential u^+ , and the four concentrations c_i produced by our PNPic solver for mVDAC1 in a NaClKNO3 solution onto a cross section ($x = 0$). Here, the partition $u = u^+ + u^-$ is given in (50), and the units of concentrations and potentials are the same as those given in Figure 7.

Na^+ and Ca^{2+} having the same bulk concentrations, the test results show that an ion with the higher charge (i.e., Ca^{2+}) is more competitive for space in the area with negative electrostatic potential values. In the case of the NaClKNO3 solution, the test results show that Na^+ and Cl^- ions are more competitive for space than K^+ and NO_3^- ions in the

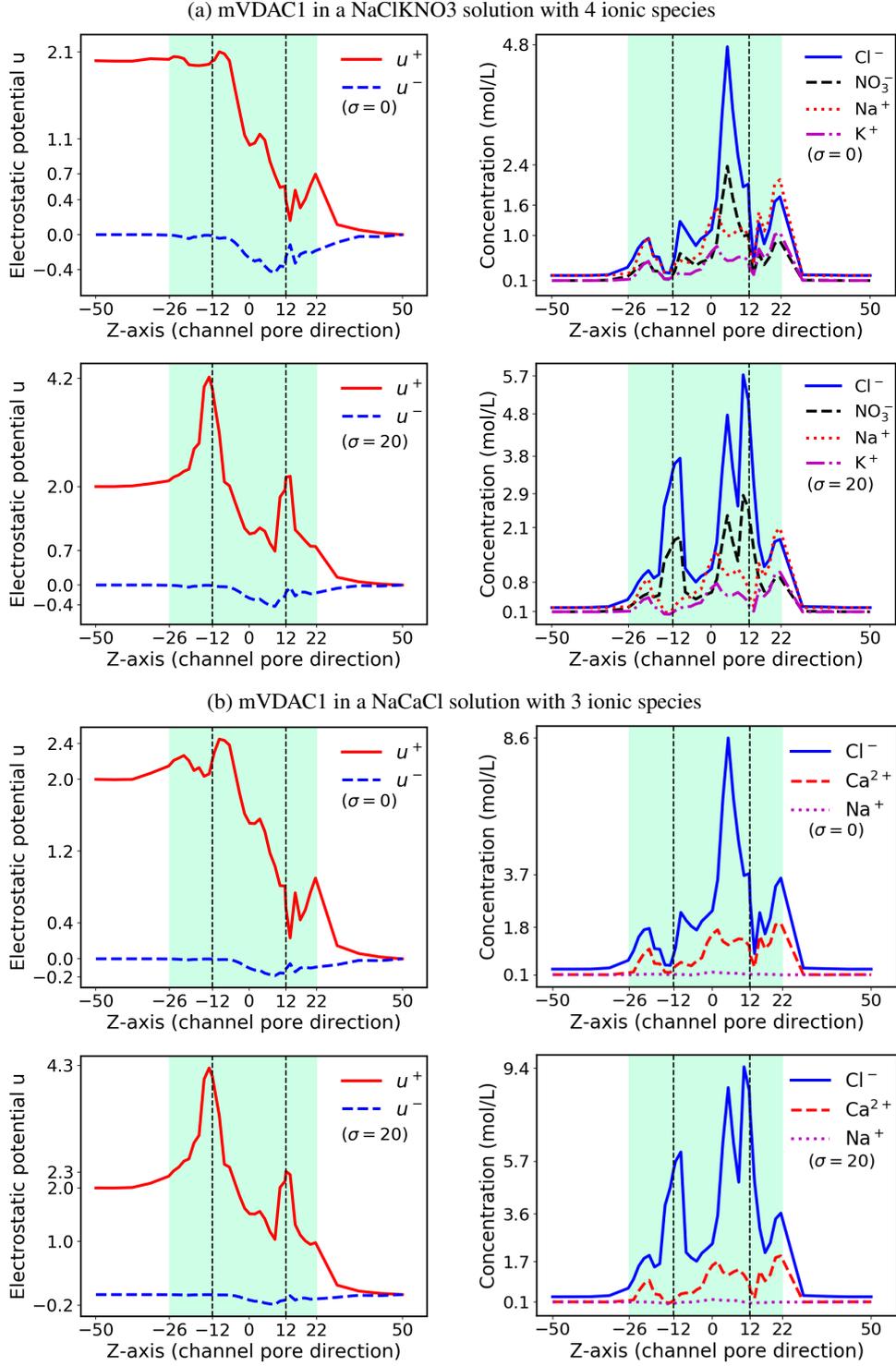


Figure 9: A comparison of the electrostatic potential u and ionic concentrations c_i calculated by our PNPic model using the membrane surface charge density $\sigma = 20 \mu\text{C}/\text{cm}^2$ with those without any membrane charge (i.e., $\sigma = 0$). Here, u^+ and u^- are defined in (50), the curves of u^+ , u^- , and c_i are depicted in their average values calculated by the block-integrals (49) and (51) with $m = 40$, the channel pore area is shaded in green color, and the membrane area is located between the two dotted lines $z = -12$ and $z = 12$.

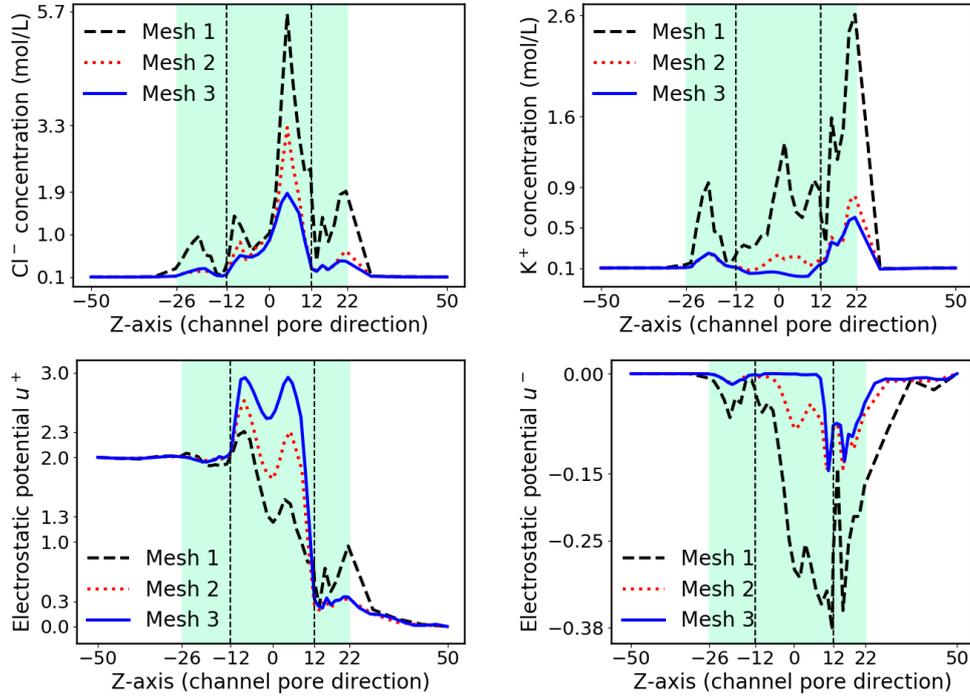


Figure 10: Mesh size influence on the convergence of our PNPic finite element solution — the electrostatic potentials u^\pm and ionic concentrations for Cl^- and K^+ for mVDAC1 in a 0.1 molar KCl solution, Here Meshes 1, 2, and 3 have 44939, 82127, and 111836 mesh nodes, respectively.

areas having negative and positive electrostatics because the species of Na^+ and Cl^- have larger bulk concentrations than the species of K^+ and NO_3^- (0.2 mol/L via 0.1 mol/L). Furthermore, from Figure 9 it can be clearly seen that the membrane surface charge density $\sigma = 20$ significantly increased the values of positive electrostatic potential function u^+ and anionic concentrations, enhancing the anion selectivity of mVDAC1. With our curve visualization tool, we clearly compared different potential and concentration functions in this figure. These numerical results validates that our PNPic model can well retain the anion selectivity property of mVDAC1.

Finally, we did numerical tests to check the influence of mesh sizes on the convergence and performance of our PNPic finite element solver. Due to the memory limitation of our iMac, these tests were done for mVDAC1 in a 0.1 molar KCl solution based on three different meshes, called Meshes 1, 2, and 3. Here, Mesh 1 is the mesh used in the previous tests. It was refined as Meshes 2 and 3 by adding new mesh nodes though adjusting the quality mesh generation option (-q) of TetGen. In these tests, we regarded Mesh 1 as the coarse mesh, Mesh 3 as the fine mesh, and Mesh 2 as the intermediate mesh. We used $u_b = 2$, $u_t = 0$, $\sigma = 0$, and other default parameters in these tests. The test results were reported in Table 2 and Figure 10.

Mesh	Mesh-node number	Iteration number	CPU time (min.)
Mesh 1	44,939	8	3.40
Mesh 2	82,127	9	7.34
Mesh 3	111,836	10	10.85

Table 2: Mesh size influence on the convergence and performance of our PNPic finite element solver for mVDAC1 in a 0.1 molar KCl solution.

From Table 2 it can be seen that the number of iterations was increased only slightly from 8 to 10 as the coarse mesh was changed to the fine mesh, indicating that the convergence rate of our PNPic finite element iterative solver is not sensitive to the mesh sizes, but the mesh sizes do significantly affect the performance of our PNPic solver. In fact, each iteration of our nonlinear iterative scheme (35) involves several linear systems. How to solve these linear

systems can significantly affect the performance of our PNPic solver. Note that the GMRES-ILU iterative method is the default linear system solver in our PNPic software package. Thus, one way to improve the performance is to develop more efficient linear system solvers than the GMRES-ILU method. We have started to work toward this direction (see [5] for some initial results).

Figure 10 shows that our PNPic finite element solution is sensitive to the mesh sizes. It also displays some convergence tendency of our PNPic finite element solution. Note that each point of a curve in Figure 10 represents an average value of a 3D concentration/potential function calculated by the volume integrals (52). Hence, each curve of Figure 10 well retains the distribution pattern of a corresponding concentration/potential function across the membrane. Interestingly, from Figure 10 we can see the reduction of a finite element approximate ionic concentration function and the increment of a finite element approximate electrostatic potential function as the mesh size was increased from the coarse grid level to the fine grid level, implying some convergence behaviors of a PNPic finite element solution sequence in terms of mesh sizes.

8. Conclusions

We have presented an efficient nonlinear finite element iterative algorithm for solving a Poisson-Nernst-Planck ion channel (PNPic) model with Neumann boundary conditions and a membrane surface charge density. Our PNPic finite element solver has been implemented as a software package, which works for an ion channel protein with a three-dimensional crystallographic structure and a mixture solution of multiple ionic species. Numerical test results for a voltage-dependent anion-channel (VDAC) in a mixture solution with up to four ionic species are reported to demonstrate the convergence of our nonlinear iterative algorithms and the performance of our software package.

So far, the influence of membrane charges and the ion channels from the outside of a simulation box domain was mostly ignored in the current PNP ion channel models due to the difficulties in numerical calculation and computer programming issues. In this paper, we considered such an influence by using Neumann boundary value conditions and a membrane surface charge density, and overcame the related numerical difficulties by introducing a novel solution decomposition scheme, a nonlinear alternating block iterative algorithm, and a Slotboom variable transformation technique. We then developed an efficient modified Newton iterative scheme for solving each nonlinear finite element equation arisen from the Slotboom variable transformation. To retain the numerical accuracy, we approximated a potential function and an ionic concentration, respectively, on their own physical domains — a box domain and a solvent region, and constructed proper communication operators to carry out function operations between the two related different finite element function spaces. To display the distribution patterns of electrostatics and ionic concentrations in simple 2D curves in the membrane normal direction, we introduced a numerical scheme for computing a volume integral of a three-dimensional potential or concentration over a block of a solvent region. We implemented this scheme based on a block partition of the solvent region along the ion channel pore such that each potential/concentration function can be visualized in a two-dimensional curve. Such a curve can well represent the potential/concentration profile over the whole solvent region since each point of the curve represents an average value of a potential/concentration in a block of the whole solvent region. Using such curves, we clearly validated that our PNPic model can well retain the VDAC anion selectivity property, and indicated that the VDAC anion selectivity property can be affected by membrane charges significantly.

Our PNPic software package will be a valuable tool for ion channel simulation and study. In the future, we plan to do more numerical tests on this package for various ion channel proteins and different ionic solvents. We will also use this package to study how and to what extent the Neumann boundary value conditions and ionic surface charges can affect ion transport and electric current across membrane or within an ion channel pore.

References

- [1] R. ADAMS AND J. FOURNIER, *Sobolev Spaces*, vol. 140 of Pure and Applied Mathematics, Amsterdam). Elsevier/Academic Press, Amsterdam., second ed., 2003.
- [2] W. M. BOTELLO-SMITH AND R. LUO, *Applications of MMPBSA to membrane proteins I: Efficient numerical solutions of periodic Poisson-Boltzmann equation*, *Journal of Chemical Information and Modeling*, 55 (2015), pp. 2187–2199.
- [3] S. BRENNER AND L. SCOTT, *The Mathematical Theory of Finite Element Methods*, Springer-Verlag, New York, third ed., 2008.
- [4] A. K. CAMARA, Y. ZHOU, E. TAJKHORSHID, AND W.-M. KWOK, *Mitochondrial VDAC1: A key gatekeeper as potential therapeutic target*, *Frontiers in Physiology*, 8 (2017), pp. 1–18.

- [5] Z. CHAO, D. XIE, AND A. H. SAMEH, *Preconditioners for nonsymmetric indefinite linear systems*, Journal of Computational and Applied Mathematics, 367 (2020), pp. 112436 (15 pages).
- [6] D. CHEN, J. LEAR, AND R. S. EISENBERG, *Permeation through an open channel: Poisson-Nernst-Planck theory of a synthetic ionic channel*, Biophysical Journal, 72 (1997), pp. 97–116.
- [7] I. CHERN, J. LIU, AND W. WANG, *Accurate evaluation of electrostatics for macromolecules in solution*, Methods and Applications of Analysis, 10 (2003), pp. 309–328.
- [8] B. E. COHEN, T. B. MCANANEY, S. P. EUN, N. J. YUH, AND E. AL, *Probing protein electrostatics with a synthetic fluorescent amino acid*, Science, 296 (2002), pp. 1700–1703.
- [9] L. C. EVANS, *Partial Differential Equations*, vol. 19 of Graduate studies in mathematics, American Mathematical Society, second ed., 2010.
- [10] H. GAO AND P. SUN, *A linearized local conservative mixed finite element method for Poisson-Nernst-Planck equations*, Journal of Scientific Computing, 77 (2018), pp. 793–817.
- [11] H. K. GUMMEL, *A self-consistent iterative scheme for one-dimensional steady state transistor calculations*, IEEE Transactions on electron devices, 11 (1964), pp. 455–465.
- [12] T. HEIMBURG, *Physical properties of biological membranes*, Wiley-VCH, 2009, ch. Handbook of Molecular Biophysics, pp. 593–616.
- [13] T.-L. HORNG, T.-C. LIN, C. LIU, AND R. S. EISENBERG, *PNP equations with steric effects: a model of ion flow through channels*, The Journal of Physical Chemistry B, 116 (2012), pp. 11422–11441.
- [14] N. JI, T. LIU, J. XU, L. SHEN, AND B. LU, *A finite element solution of lateral periodic Poisson-Boltzmann model for membrane channel proteins*, International Journal of Molecular Sciences, 19 (2018), pp. 1–17 (article 695).
- [15] M. G. KURNIKOVA, R. D. COALSON, P. GRAF, AND A. NITZAN, *A lattice relaxation algorithm for three-dimensional Poisson-Nernst-Planck theory with application to ion transport through the Gramicidin A channel*, Biophysical Journal, 76 (1999), pp. 642–56.
- [16] N. LAKSHMINARAYANIAH AND K. MURAYAMA, *Estimation of surface charges in some biological membranes*, The Journal of membrane biology, 23 (1975), pp. 279–292.
- [17] T. LIU, S. BAI, B. TU, M. CHEN, AND B. LU, *Membrane-channel protein system mesh construction for finite element simulations*, Computational and Mathematical Biophysics, 3 (2015), pp. 128–139.
- [18] X. LIU, Y. QIAO, AND B. LU, *Analysis of the mean field free energy functional of electrolyte solution with nonhomogenous boundary conditions and the generalized PB/PNP equations with inhomogeneous dielectric permittivity*, SIAM Journal on Applied Mathematics, 78 (2018), pp. 1131–1154.
- [19] A. LOGG, K.-A. MARDAL, AND G. N. WELLS, eds., *Automated Solution of Differential Equations by the Finite Element Method*, vol. 84 of Lecture Notes in Computational Science and Engineering, Springer Verlag, 2012.
- [20] B. LU, *Poisson-Nernst-Planck equations*, in Encyclopedia of Applied and Computational Mathematics, B. Engquist, ed., Berlin: Springer, 2013.
- [21] B. LU, M. J. HOLST, J. A. MCCAMMON, AND Y. ZHOU, *Poisson-Nernst-Planck equations for simulating biomolecular diffusion–reaction processes I: Finite element solutions*, Journal of Computational Physics, 229 (2010), pp. 6979–6994.
- [22] B. LU AND Y. C. ZHOU, *Poisson-Nernst-Planck equations for simulating biomolecular diffusion-reaction processes II: size effects on ionic distributions and diffusion-reaction rates*, Biophysical Journal, 100 (2011), pp. 2475–2485.
- [23] K. S. MCCOMMIS AND C. P. BAINES, *The role of VDAC in cell death: Friend or foe?*, Biochimica et Biophysica Acta (BBA)-Biomembranes, 1818 (2012), pp. 1444–1450.
- [24] M. S. METTI, J. XU, AND C. LIU, *Energetically stable discretizations for charge transport and electrokinetic models*, Journal of Computational Physics, 306 (2016), pp. 1–18.
- [25] G. MOY, B. CORRY, S. KUYUCAK, AND S.-H. CHUNG, *Tests of continuum theories as models of ion channels. I. Poisson-Boltzmann theory versus Brownian dynamics*, Biophysical Journal, 78 (2000), pp. 2349–2363.
- [26] J. M. ORTEGA AND W. C. RHEINOLDT, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, 1970.
- [27] Y. QIAO, X. LIU, M. CHEN, AND B. LU, *A local approximation of fundamental measure theory incorporated into three dimensional Poisson-Nernst-Planck equations to account for hard sphere repulsion among ions*, Journal of Statistical Physics, 163 (2016), pp. 156–174.
- [28] B. ROUX, T. ALLEN, S. BERNECHE, AND W. IM, *Theoretical and computational models of biological ion channels*, Q Rev Biophys, 37 (2004), pp. 15–103.
- [29] V. SHOSHAN-BARMATZ, V. DE PINTO, M. ZWECKSTETTER, Z. RAVIV, N. KEINAN, AND N. ARBEL, *VDAC, a multi-functional mitochondrial protein regulating cell life and death*, Molecular aspects of medicine, 31 (2010), pp. 227–285.
- [30] J. W. SLOTBOOM, *Computer-aided two-dimensional analysis of bipolar transistors*, IEEE Transactions on Electron Devices, 20 (1973), pp. 669–679.
- [31] B. TU, M. CHEN, Y. XIE, L. ZHANG, B. EISENBERG, AND B. LU, *A parallel finite element simulator for ion transport through three-dimensional ion channel systems*, Journal of computational chemistry, 34 (2013), pp. 2065–2078.
- [32] B. TU, Y. XIE, L. ZHANG, AND B. LU, *Stabilized finite element methods to simulate the conductances of ion channels*, Computer Physics Communications, 188 (2015), pp. 131–139.
- [33] R. UJWAL, D. CASCIO, J.-P. COLLETIER, S. FAHAM, J. ZHANG, L. TORO, P. PING, AND J. ABRAMSON, *The crystal structure of mouse VDAC1 at 2.3 Å resolution reveals mechanistic insights into metabolite gating*, Proceedings of the National Academy of Sciences, 105 (2008), pp. 17742–17747.
- [34] G.-W. WEI, Q. ZHENG, Z. CHEN, AND K. XIA, *Variational multiscale models for charge transport*, SIAM Review, 54 (2012), pp. 699–754.
- [35] D. XIE, *New solution decomposition and minimization schemes for Poisson-Boltzmann equation in calculation of biomolecular electrostatics*, J. Comput. Phys., 275 (2014), pp. 294–309.
- [36] D. XIE, S. H. AUDI, AND R. K. DASH, *A size-modified Poisson-Boltzmann ion channel model in a solvent of multiple ionic species: Application to voltage-dependent anion channel*, Journal of Computational Chemistry, 41 (2020), pp. 218–230.
- [37] D. XIE, *New finite element iterative methods for solving a nonuniform ionic size modified Poisson-Boltzmann equation*, International Journal of Numerical Analysis & Modeling, 14 (2017), pp. 688–711.
- [38] D. XIE AND B. LU, *An effective finite element iterative solver for a Poisson-Nernst-Planck ion channel model with periodic boundary condi-*

- tions. SIAM Journal on Scientific Computing, accepted (2020).
- [39] D. XIE, H. W. VOLKMER, AND J. YING, *Analytical solutions of nonlocal Poisson dielectric models with multiple point charges inside a dielectric sphere*, Physical Review E, 93 (2016), p. 043304 (11).
 - [40] Z. YU, M. HOLST, Y. CHENG, AND J. McCAMMON, *Feature-preserving adaptive mesh generation for molecular shape modeling and simulation*, Journal of Molecular Graphics and Modelling, 26 (2008), pp. 1370–1380.
 - [41] Q. ZHENG, D. CHEN, AND G.-W. WEI, *Second-order Poisson-Nernst-Planck solver for ion transport*, Journal of Computational Physics, 230 (2011), pp. 5239–5262.