Analysis on Some Basic Ion Channel Modeling Problems

by

Zhen Chao

A Dissertation Submitted in

Partial Fulfillment of the

Requirements for the Degree of

Doctor of Philosophy

in Mathematics

at

The University of Wisconsin–Milwaukee

August 2020

ABSTRACT

Analysis on Some Basic Ion Channel Modeling Problems

by

Zhen Chao

The University of Wisconsin-Milwaukee, 2020
Under the Supervision of Professor Dexuan Xie

The modeling and simulation of ion channel proteins are essential to the study of many vital physiological processes within a biological cell because most ion channel properties are very difficult to address experimentally in biochemistry. They also generate a lot of new numerical issues to be addressed in applied and computational mathematics. In this dissertation, we mainly deal with some numerical issues that are arisen from the numerical solution of one important ion channel dielectric continuum model, Poisson-Nernst-Planck (PNP) ion channel model, based on the finite element approximation approach under different boundary conditions and unstructured tetrahedral meshes. In particular, we present the derivation of an improved PNP ion channel model using Dirichlet boundary value conditions and membrane surface charges, and obtain its variational formulations. To solve this PNP ion channel model numerically, we develop a fast finite element iterative method and program it as a software package by using effective numerical techniques. This work makes it possible for us to carry out numerical tests in order to study the affection of different boundary value conditions on the PNP numerical solutions.

To solve a PNP ion channel model by the finite element method, one important task is to generate an interface fitted unstructured tetrahedral mesh but it is very challenging to complete since the PNP ion channel model involves three physical regions – a protein region, a membrane region, and a solvent region, and the interfaces between these three regions are very complex. To address this mesh challenge, in this dissertation, we develop a new algorithm for generating a triangular surface mesh of a simulation box domain and a new algorithm for constructing a tetrahedral mesh of the membrane region, such that we can easily split a mesh of the simulation box domain into three submeshes — the meshes of

protein, membrane, and solvent regions in high quality. Remarkably, our membrane mesh generation algorithm works for an ion channel protein with an irregular ion channel pore provided that a triangular mesh of the interface between the membrane and protein regions does not have any hole. Furthermore, we implement these two new mesh algorithms based on the state-of-the-art package FEniCS, and then adapt them to one commonly-used ion channel mesh generation package.

With our PNP ion channel program package, we study the impacts of boundary value conditions, membrane surface changes, and simulation box sizes on the quality of a PNP ion channel model. Such studies are done numerically by using crystallographic molecular structures of ion channel proteins in a solution of multiple ionic species. We visualize the three dimensional electrostatic potential and ionic concentrations not only in color mapping on a cross-section of protein, membrane, or solvent region but also in two dimensional curves with curve values being the average values of potential and concentration functions over a block partition of the solvent region along the membrane normal direction.

To
my parents,
my wife Jianmin,
and my daughter Aimee

Table of Contents

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I would never have been able to finish my dissertation without help from many people at the University of Wisconsin-Milwaukee (UWM). First and foremost, I want to express my deep and sincerest gratitude to my advisor Prof. Dexuan Xie for supervising my dissertation with guidance and patience and training me how to become a good researcher, I have learned a lot from him in many ways. His insightful advice and immense knowledge have been greatly supportive during the past four years. He also gave me a lot of advice and help when I was looking for a post-doctoral job. Besides my advisor, a big thanks also goes out to Prof. Ahmed Sameh for his inspired discussions.

I am taking this opportunity to thank Prof. Hans Volkmer, Prof. Peter Hinow, Prof. Chao Zhu, and Prof. Lei Wang for serving in my dissertation committee.

I'd like to thank Prof. Yimin Zou and my previous supervisor Prof. Guoliang Chen, who gave me lots of help and suggestion when I applied for the Ph.D. program at UWM. I especially want to thank Kathleen Wehrheim, Xianwei Van Harpen, Rebecca Bourn, Kimberly Lacking-Quinn, Suzanne Boyd, Jeb Willenbring, who are really nice people and provided great help during the past four years. I also would like to thank my friends Ye Guo, Lianlian Zhou, and Jingchun Chen, they helped me a lot in my daily life and made me feel happy in the United States.

My heartfelt thanks go to my wife, Jianmin Liu, for her support and love. She has put a lot of effort into taking care of the family and my daughter. Without her effort and encouragement, I can not finish my Ph.D. study. Also, I want to share this important moment with my lovely daughter, Aimee. I also would like to dedicate this research to my parents. All of them are, and will always be, my source of inspiration in my life.

Last but not the least, I would like to thank the Department of Mathematical Sciences for its services and support. I am really honored to study here, and I appreciated the support fund I received from UWM during my study.

Chapter 1

Introduction

1.1 Motivation and current research

All living things are composed of cells. Although cells have differences in structure and function, all of them have a membrane. The biological cell membrane separates the cell from its exterior environment. It is the protective barrier that prevents unwanted material from passing in the cell and plays an important function to transport materials into the cell to support necessary life functions. There are many different proteins embedded within the membrane that have various functions and structures. One important type of these proteins is the pore-forming membrane protein; because these proteins are concerned with ion transport, they are referred to as ion channels.

In living organisms, transport of ions through channel pores in cell membranes is a fundamental and important process to cell biology. It is facilitated by ion channels that utilize the ionic flux to perform various biological functions, such as blood pressure, cell-cell signaling, maintaining homeostasis, muscle contraction, etc. Moreover, the action of ion channels is responsible for most of what we perceive as reality in the form of sound, smell, sight, taste, touch, and forms the physiological basis for thought. They also represent important and challenging drug targets [50]. Therefore, well understanding of the ion channel is one of the utmost biomedical significance. Although there exist probably 300 types of ion channels [66], they all share some characteristics, such as ions pass through their channels by electrical potential and concentration gradient across the cell membrane, ion channels are selective, which means that only certain types of ions can pass through their channels, and ion channels play a role via their open and close states in response to chemical or electrical

stimuli.

The study of ion channels has been done for a long time, and great progress has been made in understanding the functions and structures of them, but there still exist lots of key open questions and basic issues [23], for example, there are many ion channels whose gating mechanisms are still very obscure. In fact, ion channel study is a multidisciplinary research including lots of fields of science, such as biology, chemical, mathematics. Thus, cooperation between different disciplines is very important, and it is necessary for us to develop new methods and tools to understand ion channel functions.

Over the past several decades, ion channel methodological progresses in X-ray crystallography and electron microscopy have led to tremendous progress in determining the structure of ion channels [14, 63] in high resolution, which can help us to better understand and capture some important properties, such as ion channel conductance, selectivity, and gating property. However, some microscopic activities that happen as ions pass through a channel, such as, conformational changes in the protein, solvation/desolvation of ions along the journey, are almost impossible to obtain by the experimental approach. Fortunately, some of the shortcomings of experimental methods can be obtained by computational approaches.

There exist a large number of computational methods for studies of ion channels. Here are three widely-used computational methods:

1) Molecular Dynamics (MD): it is a fully microscopic description with all atoms treated explicitly [8, 10]. The history of MD simulation for small biological molecules dates back to the 1970s [42]. MD simulations have been widely used to study functions of ion channels, such as ion conductance, channel gating, ion selectivity, or larger molecules, see [2, 24, 25, 40, 49] for more details. However, there exist some challenges in MD simulations, the accuracy of force fields (AMBER [70], CHARMM [30], GROMOS [60]) is not good enough due to that it is an approximation, and the effect of electronic polarization is not considered during the simulation. The overall MD simulation is too expensive to investigate ion currents, so it is very difficult to use MD simulation to run up to a long time scale of ion permeation pass through the membrane.

2) Poisson-Boltzmann (PB) equation: ion channels are very complicated systems with huge degrees of freedom, hence, they are challenging to model in atomistic detail. The PB equation is one popular continuum model for describing the electrostatic interaction around a charged molecule in ionic solution at the equilibrium state, a brief history of the PB theory can be found in [17]. In the PB model, the distribution of charges in the solution is assumed to satisfy Boltzmann distribution. A solution of the PB equation presents the

electrostatic potential and equilibrium concentration of ions in the solution. This model has been extensively used to estimate the free energy of transferring the ions from the bulk solution to the ion channel pore [29, 47, 71] and to calculate the electrostatic potential and ionic concentrations [55, 56]. Recently, a size-modified PB model has been studied for ion channels in a solution of multiple ionic species [73]. To solve the PB equation numerically, there exist several popular software packages, such as DelPhi [53], APBS [1], UHBD [39], PBSA [38], PBEQ [28], MIBPB [5], SDPBS [27]. However, these packages need to be modified significantly for modeling ion channels due to the complicated shape of the ion channel proteins.

3) Poisson-Nernst-Planck (PNP) equations: while the PB model only gives insights into the equilibrium state of an ion channel, PNP can be used to simulate the flow of ions across the biological membrane in a non-equilibrium state. It also can be used to study ion permeation events, which are expensive to obtain by using MD. Mathematical analyses of PNP equations have been studied for a long time since it was introduced by German theoretical physicists Nernst and Planck [48, 52] at the end of 19th century. The solution existence of the PNP equations was studied for the simple case in [26, 45]. For multiple ion species case, the existence and uniqueness of solution were studied in [34] for the 1D steady-state PNP system. Analytic solutions of PNP equations in the 1D case were found in [19, 20].

However, due to the nonlinearity of the strong coupled system of PNP equations, it is challenging to find the analytic solution of PNP equations in the realistic bimolecular context. Therefore, numerical methods are required to find approximate solutions in general. Several numerical techniques and algorithms for solving PNP equation were developed in the past two decades including finite difference methods [3, 4], recently, a second-order finite difference method by using matched interface and boundary method [76] has been developed to solve PNP system in gA ion channel [78]. The application of the finite difference method to the PNP system has some limitations on the description of the ionic channel structure. Finite volume method [41] is also very popular to solve PNP equations in some irregular domains, but it was still limited due to the low convergence rate. And a hybrid finite-difference/finite-volume method for solving PNP equations in irregular domains by utilizing a conservative discretization of the diffusive has been presented in [44]. Hollerbach et al. [21] simulated sodium chloride currents through the gramicidin A (gA) channel by using the spectral element method to solve PNP equations. Another important and widely used method is the finite element methods [37, 65]. Of these numerical methods, the finite element

method is the most suitable to handle irregular molecular surfaces of an ion channel protein and different types of boundary conditions. Current PNP models used Dirichlet boundary value conditions, ignored any effect of membrane surface charges. Recently, in order to reflect the affection of ions and ion channel proteins outside a simulation box, Xie and Lu [75] applied periodic boundary conditions to a PNP ion channel model. We also constructed an improved PNP ion channel model using Neumann boundary value condition and membrane surface charges in [74].

Each of the three approaches for ion channel simulations has limitations and advantages. The MD method is considered the most computationally expensive but also the most accurate. The PNP approaches are, in general, less computationally expensive but also provide less detail. They easily incorporate certain types of boundary value conditions that arise in physical systems. In addition, the accuracy of a PNP finite element solution depends on the quality of a related finite element mesh.

To solve a PNP ion channel model by the finite element method, we need an ion channel mesh generation package to generate an interface fitted unstructured tetrahedral mesh for a box simulation domain. Quite often, people give up the finite element method because they can not generate a tetrahedral mesh, which can fit well the three related physical regions, called the membrane, protein, and solvent region, respectively. Generally speaking, the first step of generating an ion channel mesh is to produce a high quality protein surface mesh. There exist several molecular surface mesh generation software packages, such as TMSmesh [6], NanoShaper [11], GAMer [77], and MolSurf [64]. Recently, in order to improve the efficiency, robustness, and mesh quality of the molecular surface mesh, Lu's group [33] proposed an updated version of TMSmesh, TMSmesh 2.0, to handle large molecules and speedup the computer performance of TMSmesh. Then the box triangular surface mesh of the domain is needed to produce, this can be done by Gmesh [18] or FEniCS [35]. In [32], Liu et al. used the uniform mesh with the fixed density, which can not fit the interface between membrane and solvent region well, and may reduce the quality of the corresponding volume mesh. Once the triangular surface meshes of molecular and box domain are obtained, the volume mesh can be generated by the three dimensional (3D) volume mesh generation packages such as TetGen [62, 61], CGAL [15], and Gmsh [18]. Finally, the membrane region is needed to construct to simulate the effect of the membrane surface charges. In 2014, Tu et al. [67] used cylinders or balls to generate a membrane mesh manually because the structure of each ion channel is different. Recently, this method has been improved by Lu's group [32] via the walk-detect method to yield a computer algorithm for generating the membrane

region mesh. Even so, this method works only for a single ion channel protein with the channel pore having a regular shape. It does not work well or even fails for an ion channels with a complicated channel shape.

1.2 Dissertation outline

Following what is done in [74, 75], in this dissertation, we present a finite element iterative method for solving the PNP ion channel model using Dirichlet boundary value conditions and a membrane surface charge density function. We then propose a new numerical scheme for generating a membrane mesh to improve the ion channel tetrahedron mesh generation package reported in [67, 32]. Furthermore, we implement our new PNP finite element iterative solver and new membrane mesh scheme as a software package so that we can explore some important numerical issues arisen from PNP ion channel simulations. The remaining parts of the dissertation are organized as follows:

In Chapter 2, we present the PNP ion channel model using Dirichlet boundary conditions and membrane surface charges. For clarity, this model is denoted by the PNPic model.

In Chapter 3, we follow the work in [74] to derive the variational forms related to the PNPic model and then use the novel solution decomposition method [72] to derive an efficient finite element method for solving the PNPic model approximately.

In Chapter 4, we present a new algorithm for generating a triangular surface mesh of a simulation box domain and a new algorithm for constructing a tetrahedral mesh of the membrane region. We then implement these two new mesh algorithms based on the state-of-the-art package FEniCS, and use them to modify Lu's ion channel mesh generation package.

In Chapter 5, we present the results of numerical experiments to study the effects of boundary value conditions, simulation domain sizes, and membrane surface charges.

The conclusions and future work are given in Chapter 6.

Chapter 2

Possion-Nernst-Planck ion channel models

In this chapter, we describe Possion-Nernst-Planck ion channel (PNPic) equations and the related interface and boundary value conditions, and then derive a dimensionless PNPic model.

2.1 A simulation box domain partition

Let a rectangular box open domain $\Omega \in \mathbb{R}^3$ be defined by

$$\Omega = \{(x, y, z) | L_{x_1} < x < L_{x_2}, L_{y_1} < y < L_{y_2}, L_{z_1} < x < L_{z_2}\}, \tag{2.1}$$

where L_{x_1}, L_{x_2}, L_{y_1}, L_{y_2}, L_{z_1}, and L_{z_2} are real numbers.

In order to construct the PNPic model, the domain Ω is divided as follows:

$$\Omega = D_p \cup D_m \cup D_s \cup \Gamma_m \cup \Gamma_p \cup \Gamma_{pm}, \tag{2.2}$$

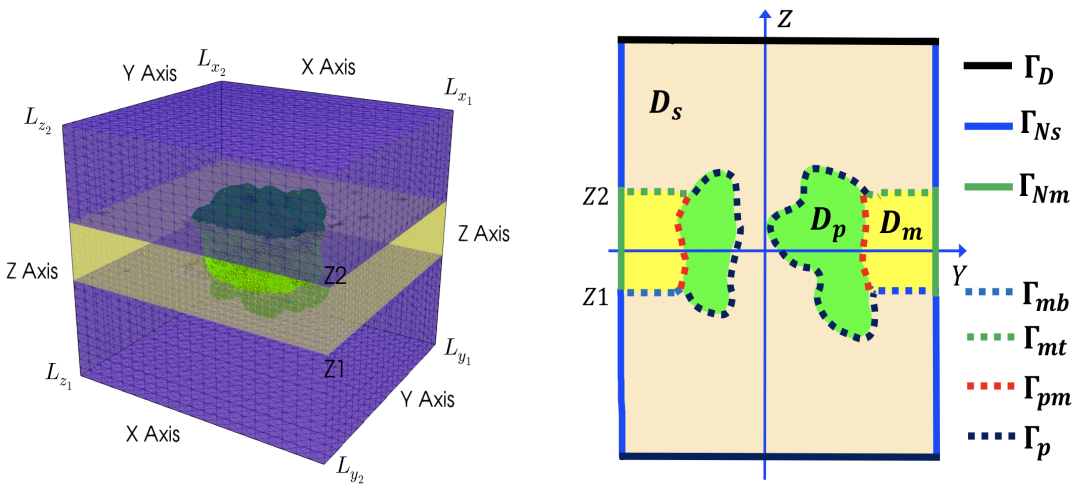
where D_p represents a protein domain, D_s is a solvent domain, which contains a solution of ionic species, D_m is a membrane domain. Γ_m, Γ_p, and Γ_{pm} denote the interfaces between D_m and D_s, D_p and D_s, and D_p and D_m, respectively. We further split the interface Γ_m and

the boundary $\partial\Omega$ by

$$\Gamma_m = \Gamma_{mb} \cup \Gamma_{mt}, \quad \partial\Omega = \Gamma_D \cup \Gamma_{Ns} \cup \Gamma_{Nm}, \tag{2.3}$$

where Γ_{mb} and Γ_{mt} represent the bottom and top surfaces of membrane, Γ_{Ns} and Γ_{Nm} consist of the lateral surfaces of the boundary ∂D_s of D_s and ∂D_m of D_m, respectively, and Γ_D consists of the bottom and top surfaces of $\partial\Omega$.

A cross-section of the domain partitions (2.2) and (2.3) is shown in Figure 2.1(b). Here the normal direction of the membrane surface is set in the z-axis direction, and the membrane location is determined by two numbers $Z1$ and $Z2$. Figure (2.1(a)) is a view of the mesh of a box domain generated from our mesh software package while Figure (2.1(b)) is simply a diagram to illustrate the region partition and related interfaces and boundaries.



(a) A 3D view

(b) A 2D cross-section

Figure 2.1: An illustration of a domain partition of (2.2).

2.2 Poisson dielectric equation

One of the cornerstones of electrostatics is setting up and solving problems described by the Poisson dielectric equation. Solving the Poisson equation amounts to finding the electric potential for a given charge distribution. Now, we give a brief description of it, the differential

form of time dependent Gauss's law in electrostatic is:

$$\nabla \cdot \epsilon(\mathbf{r})\boldsymbol{E}(\mathbf{r}) = \rho(\mathbf{r}), \tag{2.4}$$

where $\rho(\mathbf{r})$ is the total charge density that consists of both fixed and mobile charges, $\epsilon(\mathbf{r})$ is permittivity of the medium, in reality, it maybe space-dependent [51]. $\nabla\cdot$ is the divergence operator, and $\mathbf{E}$ is an electric field induced by charge density ρ. Generally, the electric permittivity $\epsilon(\mathbf{r})$ is conventionally expressed as

$$\epsilon(\mathbf{r}) = \epsilon_r(\mathbf{r})\epsilon_0,$$

where $\epsilon_r(\mathbf{r})$ is the relative permittivity, ϵ_0 is the permittivity of vacuum.

Since $\mathbf{E}$ is conservative, there exists an electrostatic potential function ϕ such that

$$\boldsymbol{E}(\mathbf{r}) = -\nabla\phi(\mathbf{r}).$$

Applying the above identity to (2.4), we obtain the Poisson dielectric equation as follows:

$$-\nabla \cdot \epsilon_r(\mathbf{r})\nabla\phi(\mathbf{r}) = \frac{\rho(\mathbf{r})}{\epsilon_0}. \tag{2.5}$$

Normally, different subdomains have different charge densities. In the PNPic model, the charge density function $\rho(\mathbf{r})$ is defined by

$$\rho(\mathbf{r}) = \begin{cases} 0, & \mathbf{r} \in D_m, \\ e_c \sum_{i=1}^{n} Z_i c_i(\mathbf{r}), & \mathbf{r} \in D_s, \\ \sum_{j=1}^{n_p} q_j \delta(\mathbf{r} - \mathbf{r}_j), & \mathbf{r} \in D_p, \end{cases} \tag{2.6}$$

where $q_j = z_j e_c$ with z_j being the valence of the j-th atom of the protein at position $\mathbf{r}_j$, e_c is the elementary charge, c_i is the concentration of the ith ionic species, Z_i is the charge number of the ith ionic species, n is the number of ionic species in the solvent, n_p is the number of atoms of the protein, and $\delta(\mathbf{r} - \mathbf{r}_j)$ is the Dirac delta function [57].

Due to the membrane chemical compositions, both protein and membrane can have

different relative electrical permittivities from the relative permittivity of the solvent. So $\epsilon_r(\mathbf{r})$ is discontinuous across the different regions, and defined as a piecewise function by

$$\epsilon_r(\mathbf{r}) = \begin{cases} \epsilon_m, & \mathbf{r} \in D_m, \\ \epsilon_s(\mathbf{r}), & \mathbf{r} \in D_s, \\ \epsilon_p, & \mathbf{r} \in D_p, \end{cases} \tag{2.7}$$

where ϵ_m and ϵ_p are the dielectric functions in the membrane and protein regions, respectively, which can be chosen as constants, and $\epsilon_s(\mathbf{r})$ denotes the dielectric function of the solvent region, which can vary from the bulk region to the ion channel pore.

From (2.6), (2.7), and (2.5), the Poisson dielectric equation for ion channel cases can be defined by

$$\begin{cases} -\epsilon_p \Delta\phi(\mathbf{r}) = \frac{e_c}{\epsilon_0} \sum_{j=1}^{n_p} z_j \delta(\mathbf{r} - \mathbf{r}_j), & \mathbf{r} \in D_p, \\ -\epsilon_m \Delta\phi(\mathbf{r}) = 0, & \mathbf{r} \in D_m, \\ -\nabla \cdot \epsilon_s(\mathbf{r}) \nabla\phi(\mathbf{r}) = \frac{e_c}{\epsilon_0} \sum_{i=1}^{n} Z_i c_i(\mathbf{r}), & \mathbf{r} \in D_s. \end{cases} \tag{2.8}$$

2.3 Nernst-Planck equation

The Nernst–Planck equation is a conservation of mass equation used to describe the flux of a charged chemical species in a fluid medium under the influence of both an ionic concentration gradient and an electric field. It is named after Walther Nernst and Max Planck.

Let's start with a general derivation of the Nernst-Planck equation. Suppose there is a flux density $\boldsymbol{J}_i$ of the i-th ionic species across the boundary $\partial\Omega_s$ of a solvent domain Ω_s, then the total ionic amount flowing out the boundary is

$$\int_{\partial\Omega_s} \boldsymbol{J}_i \cdot d\boldsymbol{S}.$$

By the law of mass conservation, the above expression should be equal to the rate of decrease of the total amount of ions in the solvent domain Ω_s, i.e.,

$$-\frac{dQ_i}{dt} = \int_{\partial\Omega_s} \boldsymbol{J}_i \cdot d\boldsymbol{S},$$

where

$$Q_i(t) = \int_{\Omega_s} c_i(t, \mathbf{r}) d\mathbf{r},$$

$c_i(t, \mathbf{r})$ is the concentration of the i-th ionic species in the solvent. Thus, it has

$$-\int_{\Omega_s} \frac{\partial c_i(t, \mathbf{r})}{\partial t} d\mathbf{r} = \int_{\partial\Omega_s} \boldsymbol{J}_i \cdot d\boldsymbol{S}.$$

Now, by the Divergence Theorem, we can get

$$-\int_{\Omega_s} \frac{\partial c_i(t, \mathbf{r})}{\partial t} d\mathbf{r} = \int_{\Omega_s} \nabla \cdot \boldsymbol{J}_i d\mathbf{r},$$

from which it implies that

$$\frac{\partial c_i(t, \mathbf{r})}{\partial t} = -\nabla \cdot \boldsymbol{J}_i. \tag{2.9}$$

The above equation is the continuity equation, which states that the divergence of the flux density $\boldsymbol{J}_i$ is equal to the negative rate of the ionic concentration c_i for the i-th ionic species.

When the flux density $\boldsymbol{J}_i$ comes from two sources, the concentration gradient and the electric potential gradient, the effect of a concentration gradient of the ith ionic species on ion flow can be described by Fick's law of diffusion [16]:

$$\boldsymbol{J}_{i,d}(t, \mathbf{r}) = -D_i(\mathbf{r}) \nabla c_i(t, \mathbf{r}), \tag{2.10}$$

where $\boldsymbol{J}_{i,d}$ is the diffusion flux, D_i is the diffusion coefficient. And the effect of an electric potential gradient of the ith ionic species is described by Ohm's law of drift [46]:

$$\boldsymbol{J}_{i,e}(t, \mathbf{r}) = -\mu_i \frac{q_i}{|q_i|} c_i(t, \mathbf{r}) \nabla \phi(t, \mathbf{r}), \tag{2.11}$$

where μ_i is the mobility of the i-th ion. Since the diffusion and drift can be impeded by the same molecular processes, there exists a physical connection between the parameters μ_i and $D_i(\mathbf{r})$, i.e., Einstein relation:

$$\mu_i = \frac{|q_i|}{k_B T} D_i(\mathbf{r}),$$

where k_B and T are the Boltzmann constant and the absolute temperature, respectively.

Then, (2.11) becomes

$$\boldsymbol{J}_{i,e}(t,\mathbf{r}) = -\frac{q_i}{k_B T} D_i(\mathbf{r}) c_i(t,\mathbf{r}) \nabla\phi(t,\mathbf{r}). \tag{2.12}$$

When both diffusional and electric field forces are present in (2.10) and (2.12), the total flux density,

$$\boldsymbol{J}_i(t,\mathbf{r}) = \boldsymbol{J}_{i,d}(t,\mathbf{r}) + \boldsymbol{J}_{i,e}(t,\mathbf{r}),$$

is obtained in the expression

$$\boldsymbol{J}_i(t,\mathbf{r}) = -D_i(\mathbf{r}) \left[\nabla c_i(t,\mathbf{r}) + \frac{q_i}{k_B T} c_i(t,\mathbf{r}) \nabla\phi(t,\mathbf{r}) \right]. \tag{2.13}$$

Combining (2.13) with (2.9), we obtain the Nernst-Planck equation of the ith ionic species in SI units as follows:

$$\frac{\partial c_i(t,\mathbf{r})}{\partial t} = \nabla \cdot D_i(\mathbf{r}) \left[\nabla c_i(t,\mathbf{r}) + \frac{Z_i e_c}{k_B T} c_i(t,\mathbf{r}) \nabla\phi(t,\mathbf{r}) \right]. \tag{2.14}$$

2.4 A Poisson-Nernst-Planck ion channel system

We now consider the steady-state Poisson-Nernst-Planck ion channel system as follows:

$$\begin{cases} -\epsilon_m \Delta\phi(\mathbf{r}) = 0, & \mathbf{r} \in D_m, \\ -\nabla \cdot \epsilon_s(\mathbf{r}) \nabla\phi(\mathbf{r}) = \frac{e_c}{\epsilon_0} \sum_{i=1}^{n} Z_i c_i(\mathbf{r}), & \mathbf{r} \in D_s, \\ -\epsilon_p \Delta\phi(\mathbf{r}) = \frac{e_c}{\epsilon_0} \sum_{j=1}^{n_p} z_j \delta(\mathbf{r} - \mathbf{r}_j), & \mathbf{r} \in D_p, \\ \nabla \cdot D_i(\mathbf{r}) \left[\nabla c_i(\mathbf{r}) + \frac{Z_i e_c}{k_B T} c_i(\mathbf{r}) \nabla\phi(\mathbf{r}) \right] = 0, & \mathbf{r} \in D_s, \quad i = 1, 2, \cdots, n. \end{cases} \tag{2.15}$$

where ϕ is measured in Volts (V), c_i has the unit of the number of ions per cubic meter, e_c, ϵ_0, k_B, and T are given in Table 2.1, and D_i has the unit m^2/s. To obtain a dimensionless form of the PNP ion channel system (2.15), we express the potential ϕ as

$$\phi(\mathbf{r}) = \frac{k_B T}{e_c} u(\mathbf{r}), \tag{2.16}$$

Parameter	Name	Unit(abbr.)	Magnitude
T	Absolute temperature	Kelvin(K)	298.15
k_B	Boltzmann constant	Joule/Kelvin(J/K)	$1.380648813\times10^{-23}$
ϵ_0	Permittivity of vacuum	Farad/Meter(F/m)	$8.854187817\times10^{-12}$
e_c	Elementary charge	Coulomb(C)	$1.602176565\times10^{-19}$

Table 2.1: Magnitudes & units of PNP parameters in SI unit system

where u is a dimensionless potential function. Then the PNP ion channel system (2.15) can be changed as

$$\begin{cases} -\epsilon_m \Delta u(\mathbf{r}) = 0, & \mathbf{r} \in D_m, \\ -\nabla \cdot \epsilon_s(\mathbf{r}) \nabla u(\mathbf{r}) = \frac{e_c^2}{\epsilon_0 k_B T} \sum_{i=1}^{n} Z_i c_i(\mathbf{r}), & \mathbf{r} \in D_s, \\ -\epsilon_p \Delta u(\mathbf{r}) = \frac{e_c^2}{\epsilon_0 k_B T} \sum_{j=1}^{n_p} z_j \delta(\mathbf{r} - \mathbf{r}_j), & \mathbf{r} \in D_p, \\ \nabla \cdot D_i(\mathbf{r}) \left[\nabla c_i(\mathbf{r}) + Z_i c_i(\mathbf{r}) \nabla u(\mathbf{r})\right] = 0, & \mathbf{r} \in D_s, \quad i = 1, 2, \cdots, n. \end{cases} \tag{2.17}$$

In molecular calculation, the coordinates of atoms are in unit of angstrom (Å). We change the length units from meter to angstrom. Correspondingly, the time is measured in picosecond (ps), diffusion coefficient $D_i(\mathbf{r})$ in units Å^2/ps, and the concentration $c_i(\mathbf{r})$ is measured in mole/liter (M/L). Then (2.17) can be rewritten as

$$\begin{cases} -\epsilon_m \Delta u(\mathbf{r}) = 0, & \mathbf{r} \in D_m, \\ -\nabla \cdot \epsilon_s(\mathbf{r}) \nabla u(\mathbf{r}) = \beta \sum_{i=1}^{n} Z_i c_i(\mathbf{r}), & \mathbf{r} \in D_s, \\ -\epsilon_p \Delta u(\mathbf{r}) = \alpha \sum_{j=1}^{n_p} z_j \delta(\mathbf{r} - \mathbf{r}_j), & \mathbf{r} \in D_p, \\ \nabla \cdot D_i(\mathbf{r}) \left[\nabla c_i(\mathbf{r}) + Z_i c_i(\mathbf{r}) \nabla u(\mathbf{r})\right] = 0, & \mathbf{r} \in D_s, \quad i = 1, 2, \cdots, n, \end{cases} \tag{2.18}$$

where

$$\alpha = \frac{10^{10} e_c^2}{\epsilon_0 k_B T}, \quad \beta = \frac{N_A e_c^2}{10^{17} \epsilon_0 k_B T}, \tag{2.19}$$

and $N_A = 6.02214076 \times 10^{23}$, which is the Avogadro constant (an estimate of the number of ions per mole.

2.5 Interface and boundary conditions

In physics, the potential function and the normal derivative of the potential function should be continuous at the interface between different regions when the permittivity function $\epsilon_{\mathbf{r}}$ is discontinuous across the interfaces. A membrane surface charge density function $\sigma(\mathbf{r})$ is provided on the membrane in order to consider the membrane surface charge effect. Thus, we have the following interface conditions:

$$\begin{cases} u(\mathbf{s}^-) = u(\mathbf{s}^+), & \epsilon_p \frac{\partial u(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_s(\mathbf{s}^+) \frac{\partial u(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})}, & \mathbf{s} \in \Gamma_p, \\ u(\mathbf{s}^-) = u(\mathbf{s}^+), & \epsilon_m \frac{\partial u(\mathbf{s}^-)}{\partial \mathbf{n}_m(\mathbf{s})} = \epsilon_s(\mathbf{s}^+) \frac{\partial u(\mathbf{s}^+)}{\partial \mathbf{n}_m(\mathbf{s})} + \tau\sigma(\mathbf{s}), & \mathbf{s} \in \Gamma_m, \\ u(\mathbf{s}^-) = u(\mathbf{s}^+), & \epsilon_p \frac{\partial u(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_m \frac{\partial u(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})}, & \mathbf{s} \in \Gamma_{pm}, \end{cases} \tag{2.20}$$

where

$$\tau = \frac{10^{-12} e_c}{\epsilon_0 k_B T}, \tag{2.21}$$

$\mathbf{n}_m$ and $\mathbf{n}_p$ denote the unit outward normal direction of D_m and D_p, respectively,

$$u(\mathbf{s}^{\pm}) = \lim_{t \to 0^+} u(\mathbf{s} + t\mathbf{n}(\mathbf{s})),$$

which are limits along a normal direction $\mathbf{n}$ of a given region from inside to outside, $\sigma(\mathbf{s})$ is defined by

$$\sigma(\mathbf{s}) = \begin{cases} \sigma_t, & \mathbf{s} \in \Gamma_{mt}, \\ \sigma_b, & \mathbf{s} \in \Gamma_{mb}. \end{cases} \tag{2.22}$$

Moreover, since the channel walls are insulating, charged particles cannot penetrate them. To ensure this, we require that the flux satisfies the Robin boundary value conditions:

$$D_i(\mathbf{s}) \left(\frac{\partial c_i(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} + Z_i c_i(\mathbf{s}) \frac{\partial u(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} \right) = 0, \quad \mathbf{s} \in \Gamma_p \cup \Gamma_m, \tag{2.23}$$

where $\mathbf{n}_s$ denotes the unit outward normal direction of the solvent region D_s.

Different boundary conditions may be imposed on (2.17) depending on specific biological settings of ion channel problems. Setting $\Gamma_N = \Gamma_{Ns} \cup \Gamma_{Nm}$, we define the boundary conditions of u and c_i as follows:

- Dirichlet boundary conditions:

$$\begin{cases} u(\mathbf{s}) = g(\mathbf{s}), & \mathbf{s} \in \partial\Omega, \\ c_i(\mathbf{s}) = g_i(\mathbf{s}), & \mathbf{s} \in \Gamma_D \cup \Gamma_{Ns}, \quad i = 1, 2, \cdots, n. \end{cases} \tag{2.24}$$

- Mixed (Neumann) boundary conditions:

$$\begin{cases} u(\mathbf{s}) = g(\mathbf{s}), & \mathbf{s} \in \Gamma_D, \\ \frac{\partial u(\mathbf{s})}{\partial \mathbf{n}(\mathbf{s})} = 0, & \mathbf{s} \in \Gamma_N, \\ c_i(\mathbf{s}) = g_i(\mathbf{s}), & \mathbf{s} \in \Gamma_D, \\ \frac{\partial c_i(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} = 0, & \mathbf{s} \in \Gamma_{Ns}, \quad i = 1, 2, \cdots, n. \end{cases} \tag{2.25}$$

where $\mathbf{n}$ denotes the unit outward normal direction of Ω. In calculation, g_i can be simply given by the bulk concentration c_i^b of species i, which is selected to satisfy the electronneutrality

$$\sum_{i=1}^{n} Z_i c_i^b = 0. \tag{2.26}$$

In summary, the PNP ion channel (PNPic) system with Dirichlet boundary conditions consists of the Poisson dielectric boundary value problem:

$$\begin{cases} -\epsilon_p \Delta u(\mathbf{r}) = \alpha \sum\limits_{j=1}^{n_p} z_j \delta(\mathbf{r} - \mathbf{r}_j), & \mathbf{r} \in D_p, \\ -\epsilon_m \Delta u(\mathbf{r}) = 0, & \mathbf{r} \in D_m, \\ -\nabla \cdot \epsilon_s(\mathbf{r}) \nabla u(\mathbf{r}) = \beta \sum\limits_{i=1}^{n} Z_i c_i(\mathbf{r}), & \mathbf{r} \in D_s, \\ u(\mathbf{s}^-) = u(\mathbf{s}^+), \quad \epsilon_p \frac{\partial u(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_s(\mathbf{s}^+) \frac{\partial u(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})}, & \mathbf{s} \in \Gamma_p, \\ u(\mathbf{s}^-) = u(\mathbf{s}^+), \quad \epsilon_m \frac{\partial u(\mathbf{s}^-)}{\partial \mathbf{n}_m(\mathbf{s})} = \epsilon_s(\mathbf{s}^+) \frac{\partial u(\mathbf{s}^+)}{\partial \mathbf{n}_m(\mathbf{s})} + \tau\sigma(\mathbf{s}), & \mathbf{s} \in \Gamma_m, \\ u(\mathbf{s}^-) = u(\mathbf{s}^+), \quad \epsilon_p \frac{\partial u(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_m \frac{\partial u(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})}, & \mathbf{s} \in \Gamma_{pm}, \\ u(\mathbf{s}) = g(\mathbf{s}), & \mathbf{s} \in \partial\Omega, \end{cases} \tag{2.27}$$

and n Nernst-Planck boundary value problems:

$$\begin{cases} \nabla \cdot D_i(\mathbf{r}) \left[\nabla c_i(\mathbf{r}) + Z_i c_i(\mathbf{r}) \nabla u(\mathbf{r}) \right] = 0, & \mathbf{r} \in D_s, \\ D_i(\mathbf{s}) \left(\frac{\partial c_i(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} + Z_i c_i(\mathbf{s}) \frac{\partial u(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} \right) = 0, & \mathbf{s} \in \Gamma_p \cup \Gamma_m, \\ c_i(\mathbf{s}) = g_i(\mathbf{s}), & \mathbf{s} \in \Gamma_{Ns} \cup \Gamma_D, \end{cases} \tag{2.28}$$

for $i = 1, 2, \cdots, n$.

Chapter 3

Numerical solution of a Poisson-Nernst-Planck ion channel model

In this chapter, to overcome the singular difficulty caused by the distribution function, we present a solution decomposition technique for the PNPic model with Dirichlet boundary value conditions, and obtain its variational formulations. Then we develop a fast finite element solver for solving them. In order to display the potential and concentration functions in two dimensional curves across the membrane, we introduce one scheme for computing a volume integral of a three dimensional function over a block from a block partition of the solvent region.

3.1 A solution decomposition method

In order to avoid the singularities caused by the Dirac Delta function in Poisson-Boltzmann equation, Xie [72] introduced a solution decomposition scheme. Now we apply this scheme to the Poisson dielectric boundary value problem (2.27) to decompose the electrostatic potential function u into three functions:

$$u(\mathbf{r}) = G(\mathbf{r}) + \Psi(\mathbf{r}) + \tilde{\Phi}(\mathbf{r}), \tag{3.1}$$

where G is the Green's function, which can be given analytically

$$G(\mathbf{r}) = \frac{\alpha}{4\pi\epsilon_p} \sum_{j=1}^{n_p} \frac{z_j}{|\mathbf{r} - \mathbf{r}_j|}, \quad \mathbf{r} \in \mathbb{R}^3. \tag{3.2}$$

$\Psi(\mathbf{r})$ is the solution of a linear interface boundary value problem with the Dirichlet boundary value condition as follows:

$$\begin{cases} \Delta\Psi(\mathbf{r}) = 0, \quad \mathbf{r} \in D_p \cup D_s \cup D_m, & \\ \Psi(\mathbf{s}^-) = \Psi(\mathbf{s}^+), \quad \epsilon_p \frac{\partial\Psi(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_s(\mathbf{s}) \frac{\partial\Psi(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})} + (\epsilon_s(\mathbf{s}) - \epsilon_p) \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})}, & \mathbf{s} \in \Gamma_p, \\ \Psi(\mathbf{s}^-) = \Psi(\mathbf{s}^+), \quad \epsilon_m \frac{\partial\Psi(\mathbf{s}^-)}{\partial \mathbf{n}_m(\mathbf{s})} = \epsilon_s(\mathbf{s}) \frac{\partial\Psi(\mathbf{s}^+)}{\partial \mathbf{n}_m(\mathbf{s})} + (\epsilon_s(\mathbf{s}) - \epsilon_m) \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_m(\mathbf{s})} + \tau\sigma(\mathbf{s}), & \mathbf{s} \in \Gamma_m, \\ \Psi(\mathbf{s}^-) = \Psi(\mathbf{s}^+), \quad \epsilon_p \frac{\partial\Psi(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_m \frac{\partial\Psi(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})} + (\epsilon_m - \epsilon_p) \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})}, & \mathbf{s} \in \Gamma_{pm}, \\ \Psi(\mathbf{s}) = g(\mathbf{s}) - G(\mathbf{s}), & \mathbf{s} \in \partial\Omega, \end{cases} \tag{3.3}$$

and $\tilde{\Phi}(\mathbf{r})$ is the solution of a linear interface boundary value problem with the Dirichlet boundary value condition as follows:

$$\begin{cases} \Delta\tilde{\Phi}(\mathbf{r}) = 0, & \mathbf{r} \in D_p \cup D_m, \\ -\nabla \cdot \epsilon_s(\mathbf{r}) \nabla\tilde{\Phi}(\mathbf{r}) = \beta \sum_{i=1}^{n} Z_i c_i(\mathbf{r}), & \mathbf{r} \in D_s, \\ \tilde{\Phi}(\mathbf{s}^-) = \tilde{\Phi}(\mathbf{s}^+), \quad \epsilon_p \frac{\partial\tilde{\Phi}(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_s(\mathbf{s}) \frac{\partial\tilde{\Phi}(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})}, & \mathbf{s} \in \Gamma_p, \\ \tilde{\Phi}(\mathbf{s}^-) = \tilde{\Phi}(\mathbf{s}^+), \quad \epsilon_m \frac{\partial\tilde{\Phi}(\mathbf{s}^-)}{\partial \mathbf{n}_m(\mathbf{s})} = \epsilon_s(\mathbf{s}) \frac{\partial\tilde{\Phi}(\mathbf{s}^+)}{\partial \mathbf{n}_m(\mathbf{s})}, & \mathbf{s} \in \Gamma_m, \\ \tilde{\Phi}(\mathbf{s}^-) = \tilde{\Phi}(\mathbf{s}^+), \quad \epsilon_p \frac{\partial\tilde{\Phi}(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_m \frac{\partial\tilde{\Phi}(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})}, & \mathbf{s} \in \Gamma_{pm}, \\ \tilde{\Phi}(\mathbf{s}) = 0, & \mathbf{s} \in \partial\Omega. \end{cases} \tag{3.4}$$

Actually, G is the solution of the Poisson dielectric equation

$$-\epsilon_p \Delta G(\mathbf{r}) = \alpha \sum_{j=1}^{n_p} z_j \delta(\mathbf{r} - \mathbf{r}_j), \quad \mathbf{r} \in \mathbb{R}^3. \tag{3.5}$$

From (3.3), (3.4), and (3.5)we notice that $\tilde{\Phi}$ is independent of G and Ψ, and G and Ψ are independent of c_i, so we can calculate G and Ψ first, then set $w(\mathbf{r}) = G(\mathbf{r}) + \Psi(\mathbf{r})$ to

rewrite the Nernst-Planck boundary value problems as follows:

$$\begin{cases} \nabla \cdot D_i(\mathbf{r}) \left[\nabla c_i(\mathbf{r}) + Z_i c_i(\mathbf{r}) \nabla w(\mathbf{r}) + Z_i c_i(\mathbf{r}) \nabla \tilde{\Phi}(\mathbf{r}) \right] = 0, & \mathbf{r} \in D_s, \\ D_i(\mathbf{s}) \left(\frac{\partial c_i(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} + Z_i c_i(\mathbf{s}) \frac{\partial w(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} + Z_i c_i(\mathbf{s}) \frac{\partial \tilde{\Phi}(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} \right) = 0, & \mathbf{s} \in \Gamma_p \cup \Gamma_m, \\ c_i(\mathbf{s}) = g_i(\mathbf{s}), & \mathbf{s} \in \Gamma_{Ns} \cup \Gamma_D, \end{cases} \tag{3.6}$$

for $i = 1, 2, \cdots, n$.

By introducing the effective densities (also referred to as Slotboom variables in semiconductor literature)

$$\bar{c}_i(\mathbf{r}) = c_i(\mathbf{r}) e^{Z_i u(\mathbf{r})}, \quad \bar{D}_i(\mathbf{r}) = D_i(r) e^{-Z_i u(\mathbf{r})}, \tag{3.7}$$

the Nernst-Planck equations (3.6) with the Dirichlet boundary value conditions can be transformed as

$$\begin{cases} -\nabla \cdot \left(\bar{D}_i(\mathbf{r}) \nabla \bar{c}_i(\mathbf{r}) \right) = 0, & \mathbf{r} \in D_s, \\ \bar{D}_i(\mathbf{s}) \left(\frac{\partial \bar{c}_i(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} \right) = 0, & \mathbf{s} \in \Gamma_p \cup \Gamma_m, \\ \bar{c}_i(\mathbf{s}) = g_i(\mathbf{s}) e^{Z_i g(s)}, & \mathbf{s} \in \Gamma_{Ns} \cup \Gamma_D, \end{cases} \tag{3.8}$$

where $i = 1, 2, \ldots, n$.

From the above system, we notice that the Slotboom transformation removes the convection term and results in self-adjoint Laplace equations. However, the transformed diffusion coefficient $\bar{D}_i(\mathbf{r})$ could result in a large condition number for the stiffness matrix of a finite element approximate equation of (3.8). Furthermore, the Slotboom transformation may cause overflow problems in numerical implementation due to the exponential term.

3.2 Variational formulation

To solve the PNPic model defined in (2.27) and (2.28) by the finite element method, the equivalent variational problems of them are needed, they can be obtained by following method in [74].

Let $H^1(\Omega)$ and $H^1(D_s)$ be the Sobolev function spaces defined in Ω and D_s, respectively.

Then, we introduce their subspaces, $H_0^1(\Omega)$ and V_0, as follows:

$$H_0^1(\Omega) = \left\{\nu \in H^1(\Omega) | \nu = 0 \text{ on } \partial\Omega\right\}, \quad V_0 = \left\{\mu \in H^1(D_s) | \mu = 0 \text{ on } \Gamma_{Ns} \cup \Gamma_D\right\}.$$

A finite element solution Ψ of (3.3) can be found by solving the equivalent linear variational problem:

$$\text{Find } \Psi \in H^1(\Omega) \text{ satisfying } \Psi(\mathbf{s}) = g(\mathbf{s}) - G(\mathbf{s}) \text{ on } \partial\Omega \text{ such that}$$

$$a(\Psi, \nu) = L_1(\nu) \ \ \forall \nu \in H_0^1(\Omega), \tag{3.9}$$

where

$$a(\Psi, \nu) = \epsilon_p \int_{D_p} \nabla\Psi(\mathbf{r}) \cdot \nabla\nu(\mathbf{r}) d\mathbf{r} + \epsilon_m \int_{D_m} \nabla\Psi(\mathbf{r}) \cdot \nabla\nu(\mathbf{r}) d\mathbf{r} + \int_{D_s} \nabla\Psi(\mathbf{r}) \cdot \nabla(\epsilon_s(\mathbf{r})\nu(\mathbf{r})) d\mathbf{r}, \tag{3.10}$$

and

$$\begin{aligned} L_1(\nu) = &\int_{\Gamma_p} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})} (\epsilon_s(\mathbf{s}) - \epsilon_p)\nu(\mathbf{s}) d\mathbf{s} + \int_{\Gamma_m} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_m(\mathbf{s})} (\epsilon_s(\mathbf{s}) - \epsilon_m)\nu(\mathbf{s}) d\mathbf{s} \\ &+ (\epsilon_m - \epsilon_p) \int_{\Gamma_{pm}} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})} \nu(\mathbf{s}) d\mathbf{s} + \tau \int_{\Gamma_m} \sigma(\mathbf{s})\nu(\mathbf{s}) d\mathbf{s}. \end{aligned} \tag{3.11}$$

Specially, if $\epsilon_m = \epsilon_p$, we have

$$L_1(\nu) = \int_{D_s} \nabla G(\mathbf{r}) \cdot \nabla((\epsilon_p - \epsilon_s(\mathbf{r}))\nu(\mathbf{r})) d\mathbf{r} + \tau \int_{\Gamma_m} \sigma(\mathbf{s})\nu(\mathbf{s}) d\mathbf{s}. \tag{3.12}$$

Actually, by the Green's first identity, Gauss's Theorem, $\Delta G(\mathbf{r}) = 0$ in D_s, and $\nu = 0$ on $\partial\Omega$, the volume integral of (3.12) can be reformulated as follows:

$$\begin{aligned} &\int_{D_s} \nabla G(\mathbf{r}) \cdot \nabla((\epsilon_p - \epsilon_s(\mathbf{r}))\nu(\mathbf{r})) d\mathbf{r} \\ &\quad = \int_{\partial D_s} (\epsilon_p - \epsilon_s(\mathbf{s}))\nu(\mathbf{r}) \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} d\mathbf{s} \\ &\quad = \int_{\Gamma_p} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} (\epsilon_p - \epsilon_s(\mathbf{s}))\nu(\mathbf{s}) d\mathbf{s} + \int_{\Gamma_m} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_s(\mathbf{s})} (\epsilon_p - \epsilon_s(\mathbf{s}))\nu(\mathbf{s}) d\mathbf{s} \\ &\quad = \int_{\Gamma_p} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_p(\mathbf{s})} (\epsilon_s(\mathbf{s}) - \epsilon_p)\nu(\mathbf{s}) d\mathbf{s} + \int_{\Gamma_m} \frac{\partial G(\mathbf{s})}{\partial \mathbf{n}_m(\mathbf{s})} (\epsilon_s(\mathbf{s}) - \epsilon_p)\nu(\mathbf{s}) d\mathbf{s}. \end{aligned}$$

Due to the complexity of the molecular surface, it may be difficult to do the interface integral calculation. We can avoid this, we can difficulty by using (3.12) when $\epsilon_m = \epsilon_p$.

After using the Slotboom transformation, the linear interface boundary value problem (3.4) can be formulated as the following nonlinear variational problem:

$$\text{Find} \quad \tilde{\Phi} \in H_0^1(\Omega) \quad \text{such that} \quad a(\tilde{\Phi}, \nu) - L_2(\tilde{\Phi}, \nu) = 0 \; \forall \nu \in\in H_0^1(\Omega), \tag{3.13}$$

where

$$a(\tilde{\Phi}, \nu) = \epsilon_p \int_{D_p} \nabla\tilde{\Phi}(\mathbf{r}) \cdot \nabla\nu(\mathbf{r}) d\mathbf{r} + \epsilon_m \int_{D_m} \nabla\tilde{\Phi}(\mathbf{r}) \cdot \nabla\nu(\mathbf{r}) d\mathbf{r} + \int_{D_s} \nabla\tilde{\Phi}(\mathbf{r}) \cdot \nabla(\epsilon_s(\mathbf{r})\nu(\mathbf{r})) d\mathbf{r} \tag{3.14}$$

and

$$L_2(\tilde{\Phi}, \nu) = \beta \sum_{i=1}^{n} Z_i \int_{D_s} e^{-Z_i(G(\mathbf{r})+\Psi(\mathbf{r})+\tilde{\Phi}(\mathbf{r}))} \bar{c}_i(\mathbf{r})\nu(\mathbf{r}) d\mathbf{r}. \tag{3.15}$$

Finally, the Nernst-Planck boundary value problem of (3.6) can be formulated as the following variational problems:

$$\text{Find } \bar{c}_i \in H^1(D_s) \text{ satisfying } \bar{c}_i = g_i e^{Z_i g} \text{ on } \Gamma_{Ns} \cup \Gamma_D \text{ such that}$$

$$\int_{D_s} \bar{D}_i(\mathbf{r}) \nabla\bar{c}_i(\mathbf{r}) \cdot \mu_i(\mathbf{r}) d\mathbf{r} = 0, \; \forall \mu_i \in V_0, i = 1, 2, \cdots, n. \tag{3.16}$$

3.3 A finite element iterative solver

In this section, we describe our finite element solver for the PNPic model based on the variational problems. Notice that the values of c_i are zero in the domains D_m and D_p, so c_i can only be defined in the solvent domain D_s while an electrostatic potential function u in the whole box domain Ω. In order to couple functions u and c_i when define Nernst-Planck equation, most of the researchers tried to extended c_i from D_s to Ω by setting $c_i(\mathbf{r}) = 0$ at the mesh nodes outside D_s. 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 interfaces Γ_p and Γ_m due to the continuity of c_i. To improve this, we define a finite element function space for c_i or $\bar{c}_i$ based on an irregular tetrahedral mesh of D_s directly.

Let $\mathcal{U}$ and $\mathcal{V}$ be the finite dimensional subspaces of $H^1(\Omega)$ and $H^1(D_s)$, respectively, and

$$\mathcal{U}_0 = \{\nu \in \mathcal{U} | \nu = 0 \text{ on } \partial\Omega\}, \quad \mathcal{V}_0 = \{\mu \in \mathcal{V} | \mu = 0 \text{ on } \Gamma_{Ns} \cup \Gamma_D\}.$$

After calculating G and Ψ from (3.2) and (3.9), we let $w = G + \Psi$, and compute $\tilde{\Phi}$ and $\bar{c}_i$ by solving the following nonlinear system:

Find $\tilde{\Phi} \in \mathcal{U}_0$ and $\bar{c}_i \in \mathcal{V}$ satisfying $\bar{c}_i = g_i e^{Z_i g}$ on $\Gamma_{Ns} \cup \Gamma_D$ such that

$$\begin{cases} \int_{D_s} D_i(\mathbf{r}) e^{-Z_i(w+\tilde{\Phi})} \nabla \bar{c}_i(t, \mathbf{r}) \cdot \nabla \nu_i(\mathbf{r}) d\mathbf{r} = 0, & \forall \nu_i \in \mathcal{V}_0, i = 1, 2, \cdots, n, \\ a(\tilde{\Phi}, \nu_{n+1}) - \beta \sum_{i=1}^n Z_i \int_{D_s} \bar{c}_i(\mathbf{r}) e^{-Z_i(w+\tilde{\Phi})} \nu_{n+1}(\mathbf{r}) d\mathbf{r} = 0, & \forall \nu_{n+1} \in \mathcal{U}_0. \end{cases} \tag{3.17}$$

After finding $\tilde{\Phi}$ and $\bar{c}_i$, we recover u and c_i by the formulas (3.1) and (3.7), respectively.

The nonlinear system (3.17) can be solved efficiently by adopting the nonlinear relaxation method reported in [74]. Initial iterates are essentially to determine the convergent property of the relaxation method, and how to choose good initial guesses $\bar{c}_i^{(0)}$ and $\tilde{\Phi}^{(0)}$ for the relaxation method is an important topic. Interestingly, when we let the slotboom variable $\bar{c}_i = c_i^b$, which is a bulk concentration function of the ith ion species, then

$$c_i = c_i^b e^{-Z_i u}, \quad i = 1, 2, \ldots, n, \tag{3.18}$$

and the nonlinear equation (3.9) can be simplified as s follows:

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

It can be easily shown that adding the solution $\tilde{\Phi}$ of above equation with w gives a solution $\tilde{\Phi}^{PB}$ of a Poisson-Boltzmann (PB) ion channel model with Dirichlet boundary value

conditions as follows:

$$\begin{cases} -\epsilon_m \Delta u(\mathbf{r}) = 0, & \mathbf{r} \in D_m, \\ -\nabla \cdot \epsilon_s(\mathbf{r}) \nabla u(\mathbf{r}) = \beta \sum_{i=1}^{n} Z_i c_i^b e^{-Z_i u}(\mathbf{r}), & \mathbf{r} \in D_s, \\ -\epsilon_p \Delta u(\mathbf{r}) = \alpha \sum_{j=1}^{n_p} z_j \delta(\mathbf{r} - \mathbf{r}_j), & \mathbf{r} \in D_p, \\ u(\mathbf{s}^-) = u(\mathbf{s}^+), \quad \epsilon_p \frac{\partial u(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_s(\mathbf{s}) \frac{\partial u(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})}, & \mathbf{s} \in \Gamma_p, \\ u(\mathbf{s}^-) = u(\mathbf{s}^+), \quad \epsilon_m \frac{\partial u(\mathbf{s}^-)}{\partial \mathbf{n}_m(\mathbf{s})} = \epsilon_s(\mathbf{s}) \frac{\partial u(\mathbf{s}^+)}{\partial \mathbf{n}_m(\mathbf{s})} + \tau\sigma(\mathbf{s}), & \mathbf{s} \in \Gamma_m, \\ u(\mathbf{s}^-) = u(\mathbf{s}^+), \quad \epsilon_p \frac{\partial u(\mathbf{s}^-)}{\partial \mathbf{n}_p(\mathbf{s})} = \epsilon_m \frac{\partial u(\mathbf{s}^+)}{\partial \mathbf{n}_p(\mathbf{s})}, & \mathbf{s} \in \Gamma_{pm}, \\ u(\mathbf{s}) = g(\mathbf{s}), & \mathbf{s} \in \partial\Omega, \end{cases} \tag{3.19}$$

Now, we can set $\bar{c}_i^{(0)} = c_i^b$ and $\tilde{\Phi}^{(0)} = \tilde{\Phi}^{PB}$ as a good initial guess for the relaxation method.

3.4 Visualization of potential and concentration functions in curves

After the calculation, we can visualize the values of 3D functions u (or c_i) on a cross-section of a region in a color mapping or in a line that passes through an ion channel pore as done in the current literature. However, such a visualization only reflects some values of u (or c_i) and may ignore some important values. To display a global property of u and c_i over a 3D region, we introduce a numerical method for calculating a set of average values of u (or c_i) over a block partition of the solvent region D_s along the normal direction of the membrane, which is the z-axis direction in this thesis. Using these average values, we then can plot a 2D curve. Such a curve is valuable in the study of potential and ionic distribution profiles passing through an ion channel pore and in the comparison of potential and ionic concentration functions generated by either different models or the same model but using different parameter values.

Let $\bar{h}$ be a positive parameter, which can be used to control the quality of the 2D curve. Set $m = m_1 + m_2 + m_3$, and start with $m+2$ partition numbers $\{z^j\}$ of the interval $[L_{z_1}, L_{z_2}]$

as follows:

$$z^0 < z^1 < \ldots < z^{m_1} < \ldots < z^{m_1+m_2} < \ldots < z^{m+1},$$

where

$$z^0 = L_{z1}, \quad z^{m_1} = P_{z_1}, \quad z^{m_1+m_2} = P_{z_2}, \text{ and } \quad z^{m+1} = L_{z2}.$$

In addition, the set $\{z^j\}_{j=1}^m$ contains the important location numbers that we want to know the average values such as the membrane location numbers $Z1$ and $Z2$. We also can set a larger value of m_2 to get more average values in the channel pore. Then we can build m mesh blocks B_j of D_s by

$$B_j = \big([z^j - \bar{h}/2, z^j + \bar{h}/2] \times [L_{x1}, L_{x2}] \times [L_{y1}, L_{y2}]\big) \cap D_s, \quad j = 1, 2, \ldots, m, \tag{3.20}$$

where z^j is the middle location of B_j in the z-direction, and B_j is allowed to be overlapped with its adjacent blocks in order to avoid missing any tetrahedron, and each B_j should have a nonzero volume $\|B_j\|$ by adjusting the parameter $\bar{h}$. It is clear that $\cup_{j=1}^m B_j = D_s$. Then, we can set $c_i^0 = c_{i,b}^b =$ and $c_i^{m+1} = c_{i,t}^b$ to obtain $m+2$ points (z^j, c_i^j) for $j = 0, 1, 2, \ldots, m, m+1$ by

$$c_i^j = \frac{1}{\|B_j\|} \int_{B_j} c_i(\mathbf{r}) d\mathbf{r}, \quad j = 1, 2, \ldots, m, \tag{3.21}$$

where $c_{i,b}^b$ and $c_{i,t}^b$ are bulk concentrations in the upper and lower bulk regions, respectively. Note that an electrostatic potential function u has positive and negative values. Hence, we rewrite u as the sum of a positive function u^+ and a negative function u^- by the formulas

$$u = u^+ + u^-, \quad u^+(\mathbf{r}) = \frac{u(\mathbf{r}) + |u(\mathbf{r})|}{2}, \quad u^-(\mathbf{r}) = \frac{u(\mathbf{r}) - |u(\mathbf{r})|}{2} \quad \forall \mathbf{r} \in D_s. \tag{3.22}$$

Now, we can set $u_0^\pm = g_b$ and $u_{m+1}^\pm = g_t$, and calculate the m average values $u_j^\pm$ of $u^\pm$ by

$$u_j^\pm = \frac{1}{\|B_j\|} \int_{B_j} u^\pm(\mathbf{r}) d\mathbf{r}, \quad j = 1, 2, \ldots, m, \tag{3.23}$$

which produce $m+2$ points $(z^j, u_j^\pm)$ for $j = 0, 1, 2, \ldots, m, m+1$.

After the above calculations, we get points (z^j, c_i^j) and $(z^j, u_j^\pm)$ for $j = 0, 1, 2, \ldots, m, m+1$, then 2D curves can be plotted as the visualization of concentration functions c_i and electrostatic potential function u over D_s along the z-axis driection. Since the mesh $\bar{D}_{s,\bar{h}}$

is unstructured, each block B_j, as a submesh of $\bar{D}_{s,\bar{h}}$, also has an irregular shape, thus, the volume integral is difficult to calculate. Fortunately, FEniCS has provided us with some useful functions to help us to do so. We can extract a mesh of B_j, denoted by $B_{j,h}$, from a mesh $\bar{D}_{s,\bar{h}}$ by the FEniCS function SubMesh, for example. As a submesh of $\bar{D}_{s,\bar{h}}$, $B_{j,\bar{h}}$ consists of tetrahedrons from $\bar{D}_{s,\bar{h}}$. Therefore, c_i^j and $u_j^{\pm}$ can be calculated approximately by:

$$c_i^j \approx \frac{1}{\|B_{j,\bar{h}}\|} \int_{B_{j,\bar{h}}} c_i(\mathbf{r}) d\mathbf{r} = \frac{\sum_{T \in B_{j,\bar{h}}} \int_T c_i(\mathbf{r}) d\mathbf{r},}{\sum_{T \in B_{j,\bar{h}}} \|T\|} \tag{3.24}$$

and

$$u_j^{\pm} \approx \frac{1}{\|B_{j,\bar{h}}\|} \int_{B_{j,\bar{h}}} u_j^{\pm}(\mathbf{r}) d\mathbf{r} = \frac{\sum_{T \in B_{j,\bar{h}}} \int_T u^{\pm}(\mathbf{r}) d\mathbf{r}}{\sum_{T \in B_{j,\bar{h}}} \|T\|}, \tag{3.25}$$

where T denotes a tetrahedron of the mesh domain $B_{j,\bar{h}}$.

Chapter 4

An improved ion channel finite element mesh generation scheme

To solve a PNP ion channel model by the finite element method, one of the most important tasks is to generate an interface fitted unstructured tetrahedral mesh for a box simulation domain. Actually, it is very difficult to construct such a mesh due to the irregular shape of proteins. In the PNPic model, to handle different boundary conditions and equations are defined in different regions, we need to mark triangular surface meshes on the interfaces/boundaries and tetrahedra meshes in different regions. In this chapter, we describe our algorithms and implementations for generating these meshes, and numerical tests for showing the robustness and efficiency of our mesh generation package are presented.

4.1 Our mesh generation method

We consider a closed box domain

$$\bar{\Omega} = \{(x, y, z) | L_{x_1} \leqslant x \leqslant L_{x_2}, L_{y_1} \leqslant y \leqslant L_{y_2}, L_{z_1} \leqslant x \leqslant L_{z_2}\} \tag{4.1}$$

and the partition

$$\bar{\Omega} = \bar{D}_p \cup \bar{D}_m \cup \bar{D}_s,$$

where $\bar{D}_p$, $\bar{D}_m$, and $\bar{D}_s$ are the closures of D_p, D_m, and D_s, respectively. Our purpose is to generate tetrahedral meshes for subdomains D_p, D_m, and D_s as well as triangular surface meshes for the interfaces Γ_p, Γ_m, and Γ_{pm} such that they are the submeshes of a box mesh $\bar{\Omega}_h$ to satisfy

$$\bar{\Omega}_h = \bar{D}_{p,h} \cup \bar{D}_{m,h} \cup \bar{D}_{s,h}. \tag{4.2}$$

We define the interface meshes by

$$\Gamma_{p,h} = \bar{D}_{p,h} \cap \bar{D}_{s,h},\ \Gamma_{m,h} = \bar{D}_{s,h} \cap \bar{D}_{m,h},\ \Gamma_{pm,h} = \bar{D}_{p,h} \cap \bar{D}_{m,h}.$$

Here, the normal direction of $\bar{D}_m$ is in the z-axis direction, and the location of D_m is set between the two planes $z = Z1$ and $z = Z2$.

According to the methodology of volume mesh generation, the tetrahedral mesh for a 3D domain can be generated from triangular surface meshes of its boundary. Thus, the box boundary triangular surface mesh $\partial\bar{\Omega}_h$, the protein surface mesh $\partial\bar{D}_{p,h}$, and the membrane surface mesh $\partial\bar{D}_{m,h}$ are needed. However, due to the irregular and complicated interface $\Gamma_{pm,h}$, $\bar{D}_{m,h}$ is very difficult to produce directly. To avoid this difficulty, it is usually to first generate a good quality mesh for an expanded solvent region, $\hat{D}_s$, defined by

$$\hat{D}_s = \bar{D}_s \cup \bar{D}_m,$$

such that

$$\bar{\Omega} = \hat{D}_s \cup \bar{D}_p \cup \hat{\Gamma}_{sp},$$

where $\hat{\Gamma}_{sp}$ is exactly a molecule surface of an ion channel protein, which can be given approximately by the triangular surface mesh $\partial\bar{D}_{p,h}$. We next construct the triangular surface mesh $\partial\bar{\Omega}_h$ by

$$\partial\bar{\Omega}_h = \Gamma_{D,h} \cup \Gamma_{N,h},$$

where $\Gamma_{D,h}$ and $\Gamma_{N,h}$ are the triangular meshes of Γ_D and Γ_N, respectively.

Using $\partial\bar{\Omega}_h$ and $\partial\bar{D}_{p,h}$, we can generate $\hat{D}_{s,h}$ and $\bar{D}_{p,h}$ such that

$$\hat{\Gamma}_{sp,h} = \hat{D}_{s,h} \cap \bar{D}_{p,h},$$

which is a triangular surface mesh of the interface $\hat{\Gamma}_{sp}$. The remaining work is to develop

algorithms to produce $\bar{D}_{m,h}$ as a submesh of $\hat{D}_{s,h}$, such that

$$\hat{D}_{s,h} = \bar{D}_{s,h} \cup \bar{D}_{m,h},$$

approximately. To do so, a scheme is used in the current mesh package developed by Lu's group. In this scheme, a list of additional mesh nodes is pre-allocated on the two planes $z = Z1$ and $z = Z2$, as a major part of the mesh nodes of $\Gamma_{m,h}$, and a mesh of the four side surfaces of D_m is also constructed as a submesh of $\partial\bar{\Omega}_h$.

Our major improvements on this mesh generation algorithm include a new scheme for the box boundary mesh $\partial\bar{\Omega}_h$ generation and a new scheme for generating the membrane region mesh $\bar{D}_{m,h}$. Moreover, we implement these two new schemes to yield a new mesh generation package, and then adopt it as a part of our PNPic software package.

4.2 A new triangular surface mesh for a simulation box

In this section, we present a new scheme for generating a high quality box surface mesh $\partial\bar{\Omega}_h$. By this new scheme, the quality of the membrane and solvent region meshes can be improved significantly.

We set the surface mesh partition

$$\Gamma_{N,h} = \Gamma_{Ns,h} \cup \Gamma_{Nm,h},$$

where $\Gamma_{Ns,h}$ and $\Gamma_{Nm,h}$ are the lateral surface meshes of $\bar{D}_s$ and $\bar{D}_m$, respectively. To guarantee that the two surface meshes $\Gamma_{N_s,h}$ and $\Gamma_{N_m,h}$ can match well, we further set the surface mesh partition

$$\Gamma_{Ns,h} = \Gamma_{Ns1,h} \cup \Gamma_{Ns2,h},$$

$\Gamma_{Ns2,h}$ represents the layer between $\Gamma_{Ns1,h}$ and $\Gamma_{Nm,h}$, $\Gamma_{Ns1,h}$ denotes the upper and lower parts of $\Gamma_{Ns,h}$ except the layer. We then generate $\partial\bar{\Omega}_h$ by the following steps:

- Step 1: Set the box domain $\bar{\Omega}$ by

$$\begin{cases} L_{x_1} = P_{x_1} - \delta_{x_1},\ L_{x_2} = P_{x_2} + \delta_{x_2}, \\ L_{y_1} = P_{y_1} - \delta_{y_1},\ L_{y_z} = P_{y_2} + \delta_{y_2}, \\ L_{z_1} = P_{z_1} - \delta_{z_1},\ L_{z_2} = P_{z_2} + \delta_{z_2}, \end{cases} \tag{4.3}$$

  where δ_{x_1}, δ_{x_2}, δ_{y_1}, δ_{y_2}, δ_{z_1}, δ_{z_2} are the input positive parameters to control the size of the box domain, and the box $[P_{x_1}, P_{x_2}] \times [P_{y_1}, P_{y_2}] \times [P_{z_1}, P_{z_2}]$ holds the protein region $\bar{D}_{p,h}$.

- Step 2: Specify a box surface mesh of the box domain by the inputs n_x, n_y, and n_z, which set the numbers of partitions in the x-, y-, and z-directions, respectively. We then simply set the number of partitions in the z-direction on the upper and lower parts of $\Gamma_{Ns1,h}$ as $n_z/2$, the number of partitions in the x-, y-, and z-directions on the surface mesh $\Gamma_{Nm,h}$ as $2n_x$, $2n_y$, and n_z. This setting leads to meshes of the membrane-protein region is finer than that of the solvent region and avoids any sharp triangular facet on the interface between the solvent and membrane regions. In addition, a parameter d_z is introduced to control the height of $\Gamma_{Ns2,h}$, which can be adjusted to further improve the quality of the mesh.

- Step 3: Generate the box triangular surface mesh data, vertices and triangles, and save these data for generating the volume mesh. In particular, we use the BoxMesh function from the state-of-the-art software package FEniCS [35] to generate the uniform volume meshes $\bar{\Omega}_{1,h}, \bar{\Omega}_{2,h}$, and $\bar{\Omega}_{3,h}$ of the top, middle, and bottom parts of the box domain as follows

$$\bar{\Omega}_{1,h} = \text{BoxMesh}(\text{Point}(L_{x_1}, L_{y_1}, L_{z_1}), \text{Point}(L_{x_2}, L_{y_2}, Z1 - d_z), n_x, n_y, n_z),$$

$$\bar{\Omega}_{2,h} = \text{BoxMesh}(\text{Point}(L_{x_1}, L_{y_1}, Z1), \text{Point}(L_{x_2}, L_{y_2}, Z2), 2n_x, 2n_y, n_z),$$

$$\bar{\Omega}_{3,h} = \text{BoxMesh}(\text{Point}(L_{x_1}, L_{y_1}, Z2 + d_z), \text{Point}(L_{x_2}, L_{y_2}, L_{z_2}), n_x, n_y, n_z).$$

  We then obtain a mesh of the box domain $\bar{\Omega}$ by

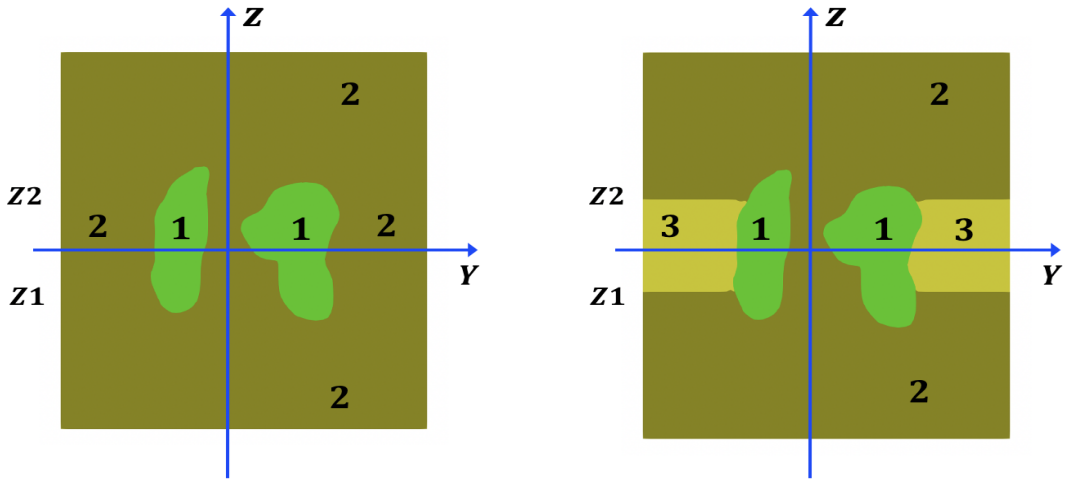
$$\bar{\Omega}_h = \bar{\Omega}_{1,h} \cup \bar{\Omega}_{2,h} \cup \bar{\Omega}_{3,h}.$$

We then extract the surface meshes $\Gamma_{Ns1,h}$, $\Gamma_{Nm,h}$ and $\Gamma_{D,h}$ by the function BoundaryMesh from FEniCS. Consequently, we produce the surface mesh $\partial\bar{\Omega}_h$ by

$$\partial\bar{\Omega}_h = \Gamma_{Ns1,h} \cup \Gamma_{Ns2,h} \cup \Gamma_{Nm,h} \cup \Gamma_{D,h}.$$

4.3 A new scheme for generating membrane tetrahedral meshes

Once the box surface mesh $\partial\bar{\Omega}_h$ and the protein surface mesh $\partial\bar{D}_{p,h}$ are obtained, then the corresponding tetrahedral meshes $\hat{D}_{s,h}$ and $\bar{D}_{p,h}$ can be generated by the volume mesh generation packages such as TetGen [62, 61]. We then need a scheme to determine $\bar{D}_{m,h}$ as a submesh of the extended solvent mesh $\hat{D}_{s,h}$. However, developing such a scheme is difficult since the interface between the membrane and the solvent regions is not given in an explicit function expression. To avoid such a difficulty, we can construct a scheme to re-mark the tetrahedra of $\hat{D}_{s,h}$ that belong to the membrane region D_m from the label number 2 to 3 as illustrated in Figure 4.1. Using these region label data as shown in Figure 4.1(b), we can produce the triangular meshes of the related interfaces and boundaries.



(a) The initial labels (b) The final labels

Figure 4.1: An illustration of the labels of protein, solvent, and membrane regions.

The most recent scheme for constructing $\bar{D}_{m,h}$ is reported in [32]. In this scheme, the walk-and-detect algorithm [12] is used to construct the membrane region mesh $\bar{D}_{m,h}$. This

method works well when the channel pore has a smooth and regular shape. It does not work well or even fails for an ion channel with a complicated channel pore. To improve this scheme, we develop a new algorithm for constructing $\bar{D}_{m,h}$. We then show that our new algorithm works for an ion channel protein even with an irregular ion channel pore provided that a surface mesh of the interface region Γ_{pm} does not have any hole.

We describe our new method in the following steps. Here we assume that $\bar{\Omega}_h$ is produced, and the tetrahedrons of the protein region and the solvent-membrane region have been marked by label numbers 1 and 2, respectively (see Figure 4.1(a)).

- Step 1: Construct a partition of the membrane region $\bar{D}_m$ as

$$\bar{D}_m = \bar{D}_{m1} \cup \bar{D}_{m2},$$

  where $\bar{D}_{m2}$ represents a subregion of $\bar{D}_m$ that intersects with a block region,

$$[l_{x_1}, l_{x_2}] \times [l_{y_1}, l_{y_2}] \times [Z1, Z2],$$

  such that $\bar{D}_{m1}$ has a regular shape, where l_{x_1}, l_{x_2}, l_{y_1}, and l_{y_2} are selected big enough not to touch the surface mesh of the protein. See Figure 4.2 for an illustration of this partition.

- Step 2: Produce a mesh $\bar{D}_{m1,h}$ of D_{m1} by remarking the tetrahedrons of the extend solvent region that belong to this region from 2 to 3.

- Step 3: Find a "seed" tetrahedron in $\bar{D}_{m2,h}$ by using either the walk-and-detect method or a direct selection method. Numerical tests show that the efficiency of these two methods is almost the same. Figure 4.3(a) shows one "seed" tetrahedron, which is colored in purple.

- Step 4: Relabel all the tetrahedrons belonging to the subregion D_{m2} starting from the "seed" tetrahedron via the following strategy: If the adjacent tetrahedron has label number 1 (i.e., in the protein region) or -1 (i.e., in the box boundary), the searching process is stopped; otherwise, this adjacent tetrahedron is re-labeled from 2 to 3, and continue to search for other adjacent tetrahedrons similarly until all the tetrahedrons belonging to D_{m2} are found.

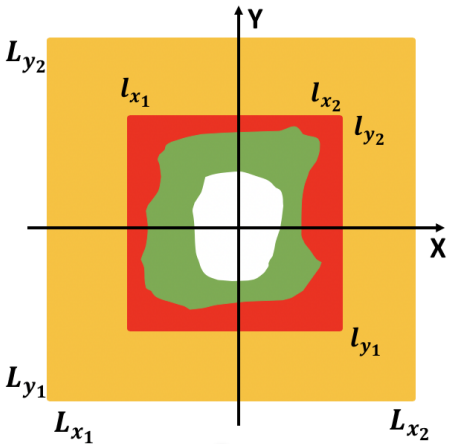
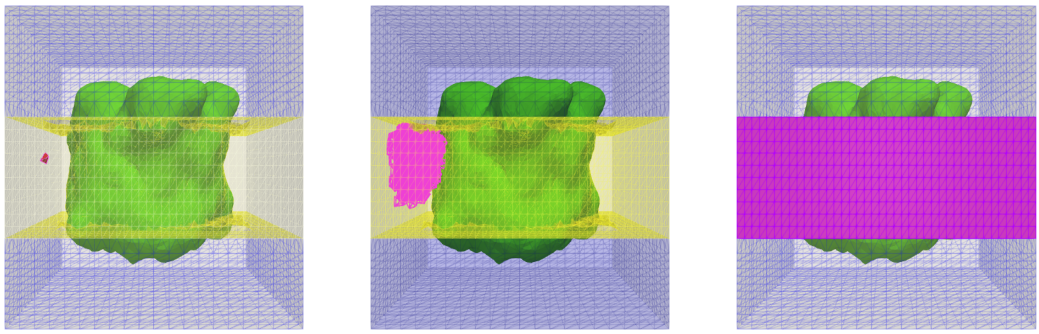


Figure 4.2: A cross-section view of the ion channel simulation box at $z = Z1$. Here, the protein region is colored in green, the membrane region D_m is split into two subdomains, D_{m1} and D_{m2}, colored in yellow and red, respectively, and the channel pore area is marked in white, which belongs to the solvent region D_s.



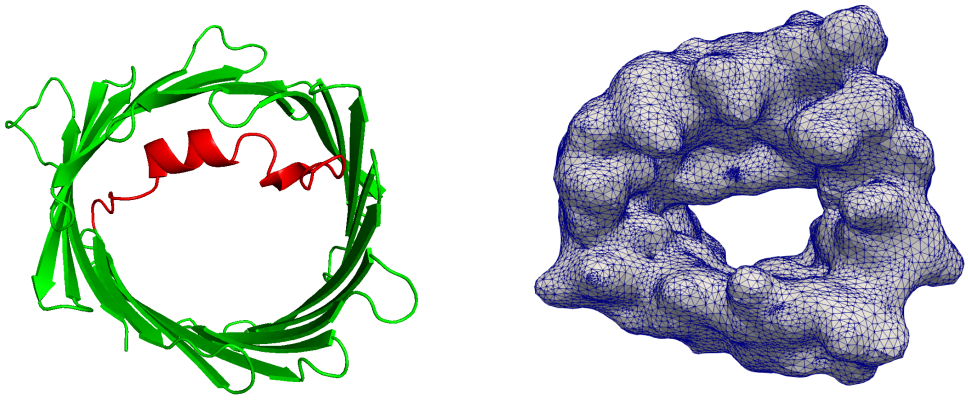
(a) A "seed" tetrahedron (b) After 16 iterations (c) Final restult

Figure 4.3: An iterative search process for the tetrahedrons that belong to the membrane region. (a) Select a "seed" tetrahedron as the initial iteration. (b) A volume mesh generated after 16 iterations. (c) The membrane mesh generated at the end of the iteration.

- Step 5: Obtain the membrane mesh $\bar{D}_{m,h}$ by combining $\bar{D}_{m1,h}$ and $\bar{D}_{m2,h}$, i.e.,

$$\bar{D}_{m,h} = \bar{D}_{m1,h} \cup \bar{D}_{m2,h}.$$

The search process of Step 4 is easy to carry out because all the data of adjacent tetrahedrons exist (e.g., in a .neigh file when the volume mesh is generated by TetGen). Figure 4.3(b) displays the tetrahedrons (colored in purple) found in the membrane region after 16 iterations, and Figure 4.3(c) shows all the tetrahedrons (colored in purple) that make up the membrane region D_m. Once the membrane mesh is constructed, we save the data of vertices,



(a) mVDAC1 in cartoon (b) Surface mesh of the mVDAC1

Figure 4.4: (a) A molecular structure of a mVDAC1 ion channel (PDB ID: 3EMN). Here the α-helix is colored in red. (b) A molecular surface mesh of mVDAC1.

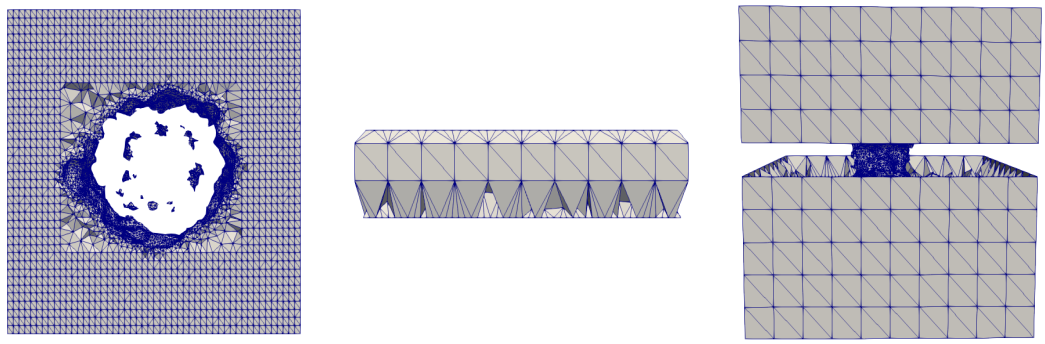
tetrahedrons, and tetrahedron labels to a .xml mesh file as required by FEniCS for a finite element mesh. We then can use this mesh directly for our PNPic package. In this way, we have obtained an improved mesh generation package.

4.4 Numerical results

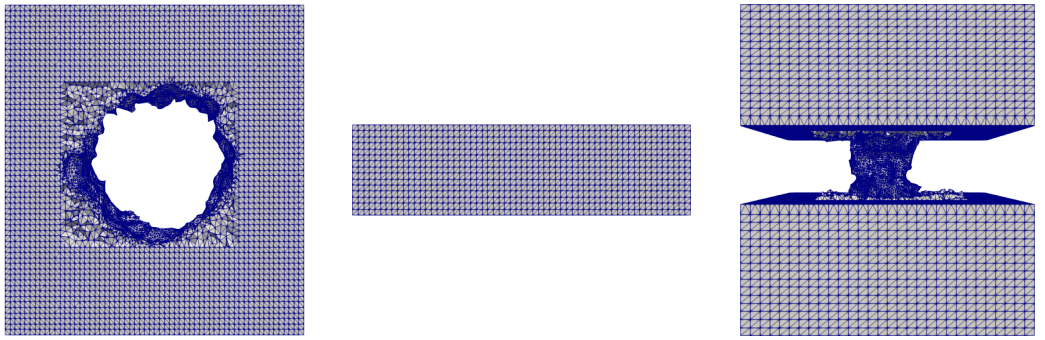
As mentioned at the start of this chapter, Lu's method works well when the channel pore has a smooth regular shape. However, it does not work well for ion channels with complicated channel pores. In this section, we present numerical tests to illustrate the robustness of our method compared to Lu's method.

Figure 4.4(a) shows the cartoon backbone representation of the original mouse voltage-dependent anion-channel isoform (mVDAC1) ion channel. Figure 4.4(b) is the corresponding triangular surface meshes, which are generated by TMSmesh [6, 33] with parameters $h = 0.4$, $d = 0.9$, and $c = 0.9$.

Figure 4.5 displays a comparison of the membrane mesh $\bar{D}_{m,h}$ and the solvent mesh $\bar{D}_{s,h}$ generated by our improved mesh package with that by Lu's mesh package for the original mVDAC1. From Figures 4.5(a, b, c) it can be seen that the membrane region mesh $\bar{D}_{m,h}$ generated by Lu's mesh package contains the tetrahedra of $\bar{D}_{s,h}$ while the solvent region mesh $\bar{D}_{s,h}$ contains some sharp triangles on $\partial\bar{D}_{s,h}$. These low quality meshes may affect the accuracy of a finite element solution. Figures 4.5(d, e, f) display the meshes $\bar{D}_{m,h}$ and $\bar{D}_{s,h}$



(a) Top view of $\bar{D}_{m,h}$ (b) Side view of $\bar{D}_{m,h}$ (c) Side view of $\bar{D}_{s,h}$



(d) Top view of $\bar{D}_{m,h}$ (e) Side view of $\bar{D}_{m,h}$ (f) Side view of $\bar{D}_{s,h}$

Figure 4.5: (a, b, c) The subdomain meshes generated by Lu's mesh package. (d, e, f) The subdomain meshes generated by our improved mesh package.

generated by our mesh generation method, which have much higher quality.

Chapter 5

Numerical studies on some ion channel modeling issues

In this chapter, we numerically study some important issues arisen from the PNP ion channel modeling. Our PNPic software package makes such a study possible. All numerical tests were done on an Intel Core i5 3.3 GHz MacBook Pro with 16GB RAM (macOS Catalina version 10.15.5) in double-precision floating-point format.

5.1 Preparation for numerical tests

From the physical point of view, the diffusion coefficient in the bulk region and the ion channel pore region can be different. In our PNPic software package, we use the diffusion coefficient defined by a piecewise expression as follows:

$$D_i(\mathbf{r}) = \begin{cases} D_i^p, & Z1 + h < z < Z2 - h \text{ (channel pore region)}, \\ D_i^p + (D_i^p - D_i^b) f_t(\mathbf{r}), & Z2 - h \leqslant z \leqslant Z2 \text{ (upper buffer region)}, \\ D_i^p + (D_i^p - D_i^b) f_b(\mathbf{r}), & Z1 \leqslant z \leqslant Z1 + h \text{ (lower buffer region)}, \\ D_i^b, & z < Z1 \text{ or } z > Z2 \text{ (bulk region)}, \end{cases} \tag{5.1}$$

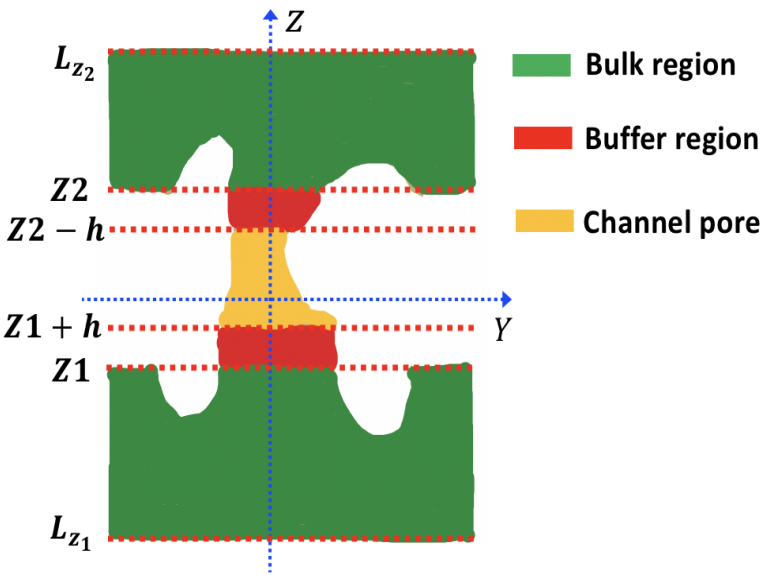


Figure 5.1: A cross-section representation of the ion channel-membrane system with the various regions for simulations.

where $\mathbf{r} = (x, y, z)$, the functions f_b and f_t are defined by

$$f_b(\mathbf{r}) = n\left(\frac{Z1 + h - z}{h}\right)^{n+1} - (n+1)\left(\frac{Z1 + h - z}{h}\right)^{n} \tag{5.2}$$

and

$$f_t(\mathbf{r}) = n\left(\frac{z - Z2 + h}{h}\right)^{n+1} - (n+1)\left(\frac{z - Z2 + h}{h}\right)^{n} \tag{5.3}$$

with n is an integer (the default value is 9), h is the buffer size (the default value is 3), D_i^b and D_i^p denotes the diffusion coefficients of the i-th ion species in the bulk and channel pore regions, respectively, and D_i^p is given by

$$D_i^p = \theta D_i^b \text{ for } 0 < \theta \leqslant 1. \tag{5.4}$$

A similar diffusion coefficient D_i can be found in [22].

We also use the following relative permittivity function ϵ_s in the solvent region:

$$\epsilon_s(\mathbf{r}) = \begin{cases} \epsilon_s^p, & Z1 + h < z < Z2 - h \text{ (channel pore region)}, \\ \epsilon_s^p + (\epsilon_s^p - \epsilon_s^b) f_t(\mathbf{r}), & Z2 - h \leqslant z \leqslant Z2 \text{ (upper buffer region)}, \\ \epsilon_s^p + (\epsilon_s^p - \epsilon_s^b) f_b(\mathbf{r}), & Z1 \leqslant z \leqslant Z1 + h \text{ (lower buffer region)}, \\ \epsilon_s^b, & z < Z1 \text{ or } z > Z2 \text{ (bulk region)}, \end{cases} \tag{5.5}$$

where ϵ_s^b and ϵ_s^p denote the relative permittivities in the bulk and channel pore regions, respectively.

The boundary potential value function g is defined by

$$g(\mathbf{s}) = \begin{cases} g_b(\mathbf{s}), & z = L_{z_1} \text{ (bottom surface of } \Omega), \\ g_t(\mathbf{s}), & z = L_{z_2} \text{ (top surface of } \Omega), \\ g_b(\mathbf{s}) + (g_t(\mathbf{s}) - g_b(\mathbf{s}))k(\mathbf{s}) & L_{z_1} < z < L_{z_2} \text{ (lateral surface of } \Omega), \end{cases} \tag{5.6}$$

where g_b and g_t denote the given potential functions, $\mathbf{s} = (x, y, z)$, and

$$k(\mathbf{s}) = \frac{z - Lz_1}{Lz_2 - Lz_1}.$$

The boundary concentration value function $g_i(\mathbf{r})$ can be set as

$$g_i = c_i^b,$$

where c_i^b is the bulk concentration of the i-th ionic species in the solvent region D_s, which is

defined by

$$c_i^b(\mathbf{r}) = \begin{cases} c_{i,p}^b(\mathbf{r}), & Z1 + h < z < Z2 - h \text{ (channel region)}, \\ c_{i,p}^b(\mathbf{r}) + (c_{i,p}^b(\mathbf{r}) - c_{i,t}^b(\mathbf{r}))f_t(\mathbf{r}), & Z2 - h \leqslant z \leqslant Z2 \text{ (upper buffer region)}, \\ c_{i,p}^b(\mathbf{r}) + (c_{i,p}^b(\mathbf{r}) - c_{i,b}^b(\mathbf{r}))f_b(\mathbf{r}), & Z1 \leqslant z \leqslant Z1 + h \text{ (lower buffer region)}, \\ c_{i,b}^b(\mathbf{r}), & z < Z1 \text{ (lower bulk region)}, \\ c_{i,t}^b(\mathbf{r}), & z > Z2 \text{ (upper bulk region)}, \end{cases} \tag{5.7}$$

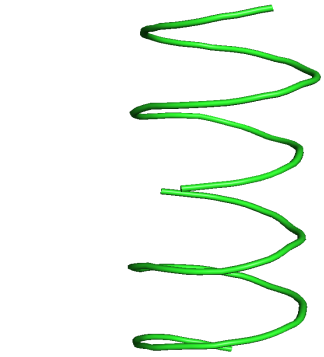
where $c_{i,p}^b$ is the concentration in the channel pore region (the default value is zero), $c_{i,b}^b$ and $c_{i,t}^b$ are the bulk concentrations in the upper and lower bulk regions, respectively.

Particularly, D_i^b, ϵ_s^p, ϵ_s^b, g_t, g_b, $c_{i,b}^b$, $c_{i,p}^b$, $c_{i,b}^b$, and $c_{i,t}^b$ are given as constants for the i-th species.

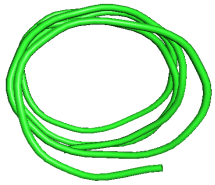
To validate our PNPic software package, we did numerical tests on two realistic ion channel proteins: Gramicidin A (gA) [31, 59] and murine voltage-dependent anion channel isoforms (mVDAC1) [58].

The gA is a natural antibiotic peptides with 15 amino acids (552 atoms) that form cation-permeable channels like pores in biological membranes and lipid bilayers of bacteria and animal cells. Inorganic monovalent ions, such as sodium (Na^+) and potassium (K^+), can pass through the channel pores via diffusion. This process can destroy vital ion concentration gradients between membranes and lipid bilayers. The pore is about 4 Å in diameter. Over the last fifty years, this channel has been studied on its ion selectivity, structure [69], etc., producing a lot of experimental data [9, 54]. Thus, the gA channel is regarded as a useful model for validating ion channel models. Figure 5.2 shows the 3D structure of gA in three different forms.

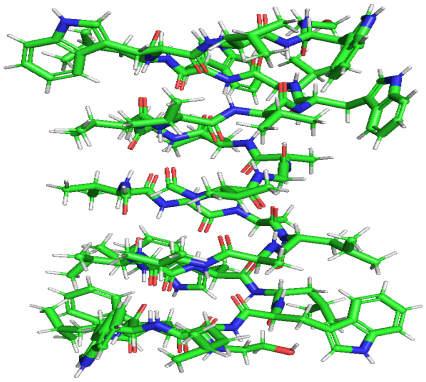
The mVDAC1 is one of the voltage-dependent anion channel isoforms of the mammalian VDAC, which are the most abundant proteins in the mitochondrial membrane [58]. The high-resolution crystal 3D structure of mVDAC1 contains an 19 stranded β barrel with an α-helix in the middle of the channel pore (see Figure 5.3) [68]. The pore is about 27 Å in diameter at the top and bottom mouths of the channel, but it narrows to about 14 Å near the helix at the center of the ion channel [7]. Although the structural information is available, the complexities of the conductance and the molecular mechanism are not understood well. Recent hypotheses show that the opening or closure of the channel could be controlled through a partial movement of the N-terminal helix with its flexible hinge region (amino



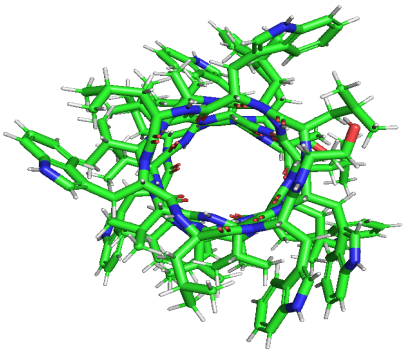
(a) Side view of 1MAG in cartoon



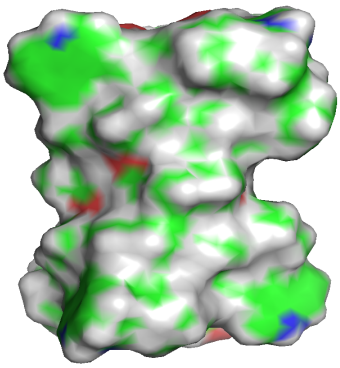
(b) Top view of 1MAG in cartoon



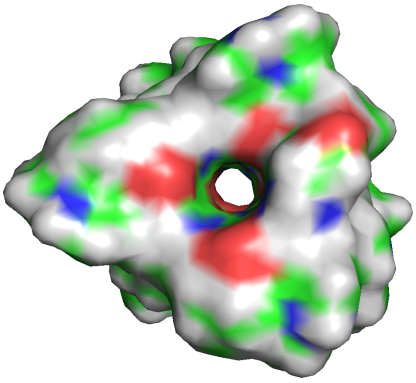
(c) Side view of 1MAG in stick



(d) Top view of 1MAG in stick



(e) Side view of 1MAG with Gaussian surface



(f) Top view of 1MAG with Gaussian surface

Figure 5.2: Illustrations of gA ion channel (PDB ID: 1MAG) geometry using PyMOL visualization software, color map indicates atoms.

acids 19–25) into or out of the pore. Although an inward α-helical movement could control the channel's activity through the blockage, it fails to account for the channel's ion selectivity, which an outward-movement model addresses more adequately [43]. Figure 5.3 shows the 3D structure of the mVDAC1 in three different forms, and the N-terminal α-helix is colored in red in Figure 5.3(a) and 5.3(b).

In our tests, the PDB files of proteins were downloaded from the Orientations of Proteins in Membranes (OPM) database [36] https://opm.phar.umich.edu, and converted to the PQR files by the online tool PDB2PQR [13], which prepares structures for further calculations by reconstructing some missing atoms and assigning atomic charges and radii from specified force fields. OPM provides us with proteins that have been rotated and transformed to match our requirement as illustrated in Figure 2.1, resulting in that $Z1 = -12$ and $Z2 = 12$ for mVDAC1 ion channel, $Z1 = -11$ and $Z2 = 6$ for gA ion channel.

Symbol	Value	Symbol	Value
ϵ_p	2	D_i for K^+	0.196
ϵ_m	2	D_i for Ca^{2+}	0.0793
ϵ_s^b	80	D_i for Na^+	0.133
$c_{i,p}^b$	0	D_i for NO_3^-	0.19
D_i for Cl^-	0.203		

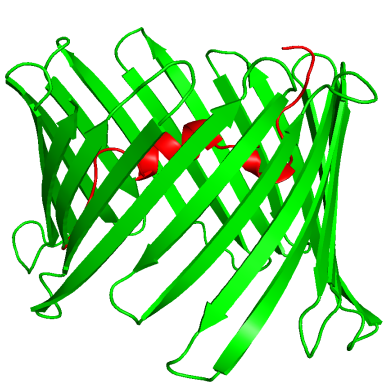
Table 5.1: Relative permittivity and diffusion coefficient constants used in our tests.

Some constants used in our numerical tests are summarized in Table 5.1. These values are collected from the experimental data and the website https://www.aqion.de/site/194.

	gA	mVDAC1
Parameter	Value	Value
TMSmesh: [h, d, c]	[0.6, 0.9, 0.9]	[0.2, 0.9, 0.9]
TetGen: [q, a, T]	[1.3, 0, 0]	[1.4, 0, 0]
BoxMesh: $[\delta_{z_1}, \delta_{z_2}]$	[10, 10]	[10, 10]
BoxMesh: $[n_x, n_y, n_z, d_z]$	[20, 40, 40, 1]	[10, 20, 20, 2]

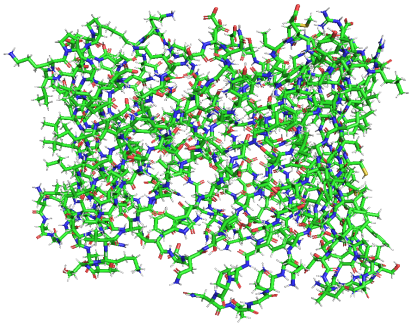
Table 5.2: Parameter values of mesh generation

The ion channel meshes used for our numerical tests were generated by the parameters in Table 5.2. Three different box domains defined in Table 5.3 were used to generate meshes,

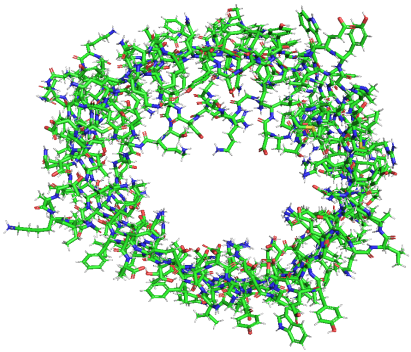


(a) Side view of 3EMN in cartoon

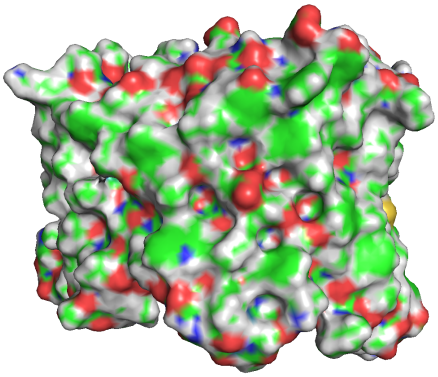
(b) Top view of 3EMN in cartoon



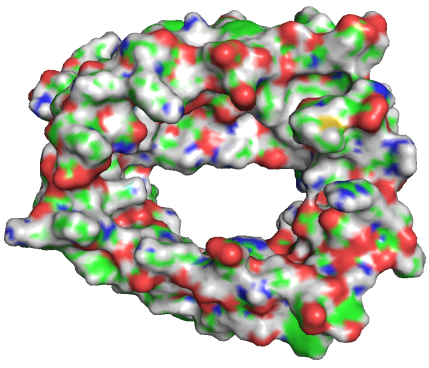
(c) Side view of 3EMN in stick



(d) Top view of 3EMN in stick



(e) Side view of 3EMN with Gaussian surface



(f) Top view of 3EMN with Gaussian surface

Figure 5.3: Illustrations of mVDAC1 ion channel (PDB ID: 3EMN) geometry using PyMOL visualization software, color map indicates atoms in (c, d, e ,f), and the α-helix is colored in red in (a, b).

and all the meshes in the channel pores were refined. To compare numerical results clearly, we used the method in Chapter 2 to depict the concentrations and potential functions in curves by using $m_1 = 10$, $m_2 = 30$, $m_3 = 10$, and $\bar{h} = 4$.

		gA		mVDAC1	
	$[\delta_{x_1}, \delta_{x_2}, \delta_{y_1}, \delta_{y_2}, \delta_{z_1}, \delta_{z_2}]$	Vertices	Tetrahedra	Vertices	Tetrahedra
Mesh 1	[10, 10, 10, 10, 10, 10]	68614	385740	49798	300921
Mesh 2	[30, 30, 30, 30, 10, 10]	72157	410080	52085	316009
Mesh 3	[50, 50, 50, 50, 10, 10]	76019	437349	53084	322728

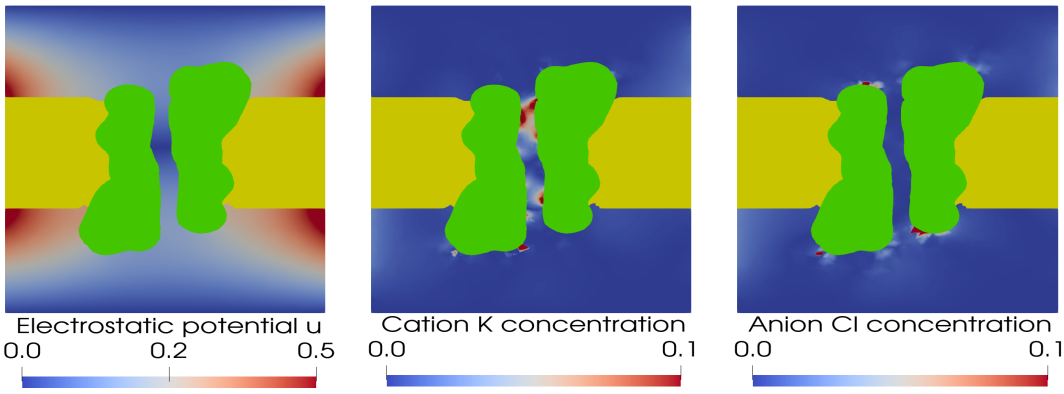
Table 5.3: Box domain dimensions and related mesh data for the three meshes used in numerical tests.

5.2 A comparison study on boundary value conditions

Several different boundary value conditions are used in PNP ion channel models. In this section, we present a comparison study on two commonly-used boundary value conditions — the Dirichlet and Neumann boundary value conditions that are stated in Chapter 2. The purpose of such a study is to show how boundary value conditions to affect the quality of a PNPic model. We used two ion channels, gA and mVDAC1, and Mesh 1 in Table 5.3 to do numerical tests, the membrane surface charge $\sigma_t = \sigma_b = 0$, and did the tests .

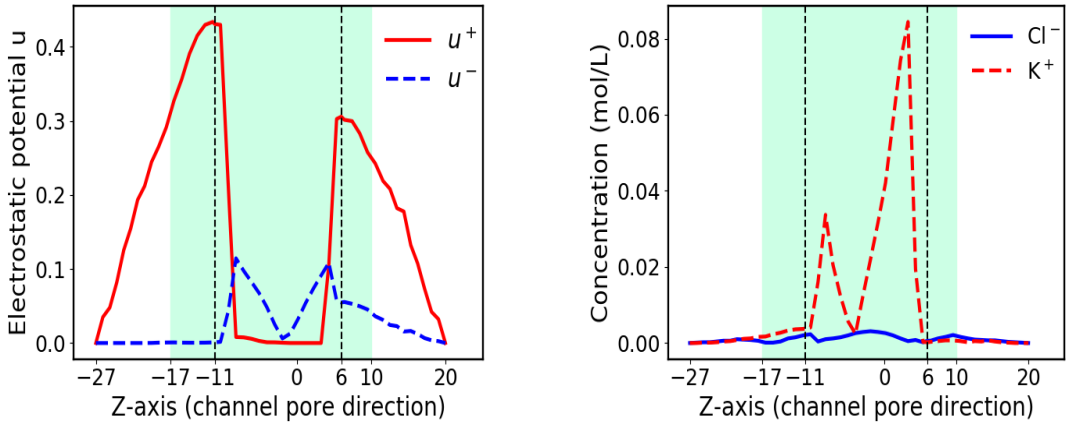
For gA in a symmetric salt solution of KCl (0.5 M/L) when $g_b = 1.946$ (i.e., 50mV) and $\theta = 0.055$, we calculated the differences between the electrostatic potentials u and concentrations c_i generated by the PNPic model using the Dirichlet and Neumann boundary value conditions, and plotted them in Figures 5.4 and 5.5. From Figure 5.4(a) it can be seen that there is a significant difference between potential results, especially in the solvent region. However, as shown in Figures 5.4(b) and 5.4(c), the differences in concentrations are not big except that the concentration of K^+ has significant changes in the entrance of the channel pore. In Figure 5.5 we depicted the differences in 2D curves, from which, we now can clearly see the trends of the differences in the whole solvent region.

We further produced current-voltage (I-V) curves using the PNPic models, and reported them in Figures 5.6 and 5.8. From these figures we can see that different boundary conditions can significantly affect the I-V curve calculation.



(a) Difference on potential (b) Difference on K^+ concentration (c) Difference on Cl^- concentration

Figure 5.4: Color mappings of the differences of the potentials and ionic concentrations generated by the PNPic model using Dirichlet and Neumann boundary value conditions on a cross-section $x = 0$ for gA in a salt solution of KCl (0.5 M/L) when $g_b = 1.946$ (i.e., 50mV). The protein and membrane regions are colored in green and yellow, respectively.



(a) Difference on potentials (b) Difference on concentrations

Figure 5.5: Differences of the potentials and ionic concentrations generated by the PNPic model using Dirichlet and Neumann boundary value conditions for gA in a salt solution of KCl (0.5 M/L) when $g_b = 1.946$ (i.e., 50mV) in 2D curves.

5.3 Simulation box size affection on ion channel simulation

The simulation box domain size is another possible affection factor for PNP ion channel simulations. To explore such an issue numerically, we set $\sigma_b = \sigma_t = 0$, $\epsilon_s^p = 80$, $g_t = 0$, $g_b = 3.892$ (i.e., 100 mV), and $\theta = 0.055$ and $\theta = 0.035$ for gA and mVDAC1 cases,

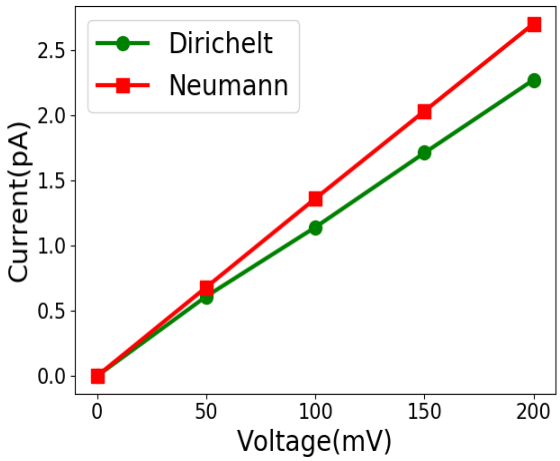
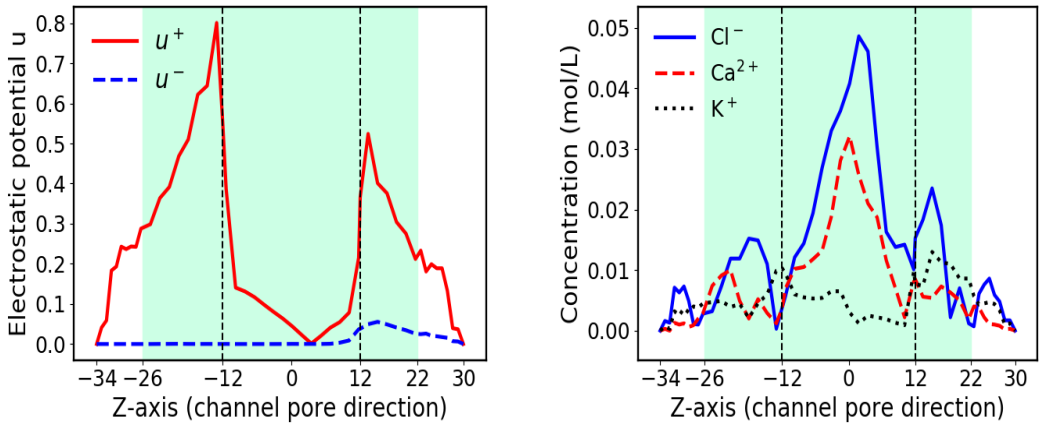


Figure 5.6: A comparison of the current-voltage curves generated by the PNPic model using Dirichlet and Neumann boundary value conditions for gA in a salt solution of KCl (0.5 M/L).



(a) Difference on potentials

(b) Difference on concentrations

Figure 5.7: Difference of potentials and ionic concentrations generated by using Dirichlet and Neumann boundary value conditions. All results are generated by PNPic model for mVDAC1 in a mixture solution of KCl (0.5 M/L) and $CaCl_2$ (0.1 M/L) when $g_b = 3.892$ (i.e., 100mV).

respectively, to control the value of diffusion constant D_i^p inside the channel pore. We used a salt solution with 0.5 M/L KCl for gA case and a mixture of 0.5 M/L KCl and 0.1 $CaCl_2$ for mVDAC1 case. The numerical test results are reported in Figures 5.9 and 5.10.

The differences in electrostatic potential and ionic concentrations are displayed in Figure 5.9 for the gA simulations. From this figure we can see that the absolute difference on potential and ionic concentration was reduced as the box domain size was increased. Fox example, the largest difference in potential and concentrations are 0.9 and 0.16, respectively,

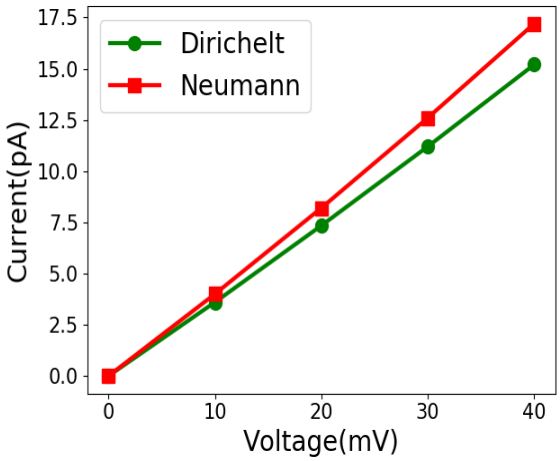


Figure 5.8: A comparison of I-V curves for different boundary value conditions.

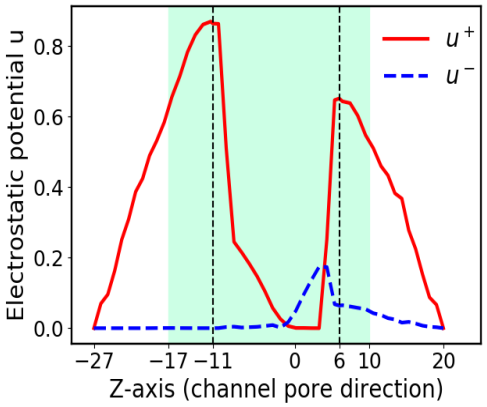
in the case of using Mesh 1, which are larger than that in the case of using a larger mesh, Mesh 3. As shown in Figure 5.10, the mVDAC1 case has the same trend as the gA case. These test results illustrate that the effects of boundary value conditions on the four side surfaces are insensitive on the PNPic model when the box dimension in the x/y-axis direction sufficiently exceeds the width of ion channel.

5.4 Effects of membrane surface charges

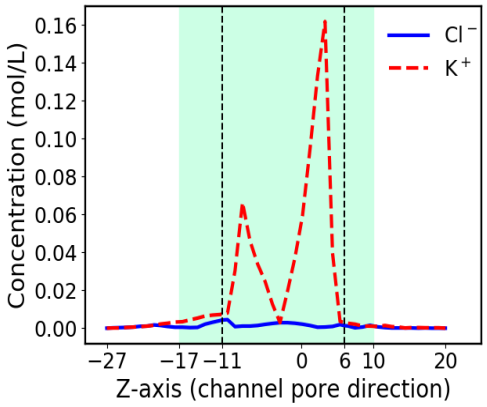
To study the effect of membrane surface charges, we did numerical tests by the PNPic model using the Dirichlet boundary value conditions and Mesh 1 defined in Table 5.3. The PNPic model was solved for a symmetric bulk salt concentration, i.e., $c^b_{i,p} = 0$ and $c^b_{i,t} = c^b_{i,b}$.

The electrostatic potential and concentrations on the cross-section $x = 0$ are shown in Figure 5.11 for different membrane surface charges under an applied voltage $g_b = 1.946$ (i.e., 50mV) for the gA ion channel in a salt solution of KCl (0.5 M/L). We also displayed the electrostatic potentials and ionic concentrations in 2D curves in Figure 5.12. From these figures, we can see that adding negative charges on one side surface of membrane attracted more cations K^+, and caused a larger negative potential in magnitude near this membrane surface. And adding positive charges on membrane surface increased positive potential, and attracted more anions Cl^- near the membrane surface, but can not enter the channel pore due to the cation selectivity of gA channel.

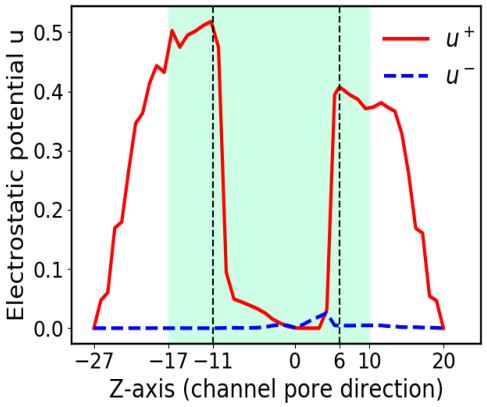
Furthermore, from Figure 5.12 it can be seen that the positive membrane surface charge $\sigma_b = 20$ significantly increased the positive electrostatic potential from 1.5 to 3 (see Fig-



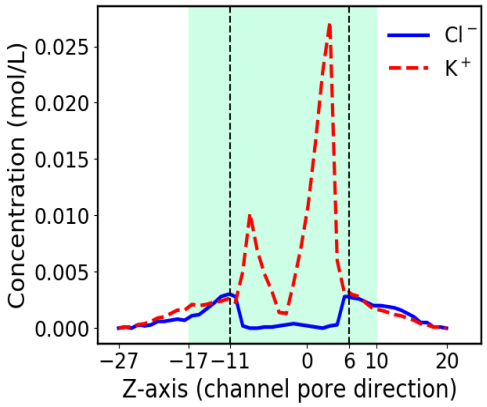
(a) Difference on potentials by Mesh 1



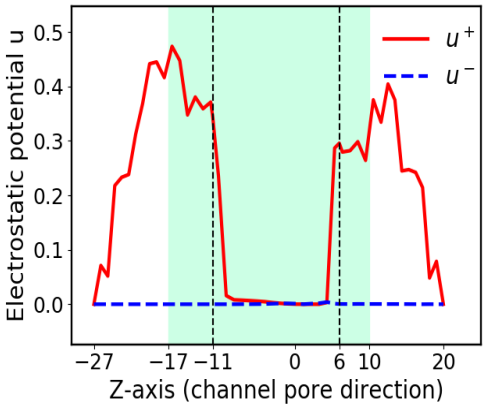
(b) Difference on concentrations by Mesh 1



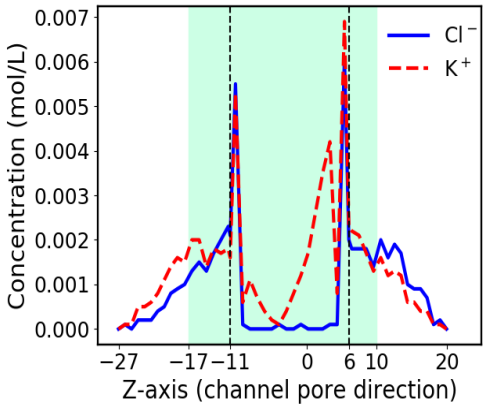
(c) Difference on potentials by Mesh 2



(d) Difference on concentrations by Mesh 2

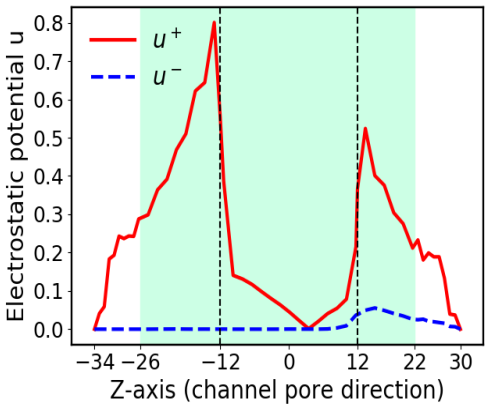


(e) Difference on potentials by Mesh 3

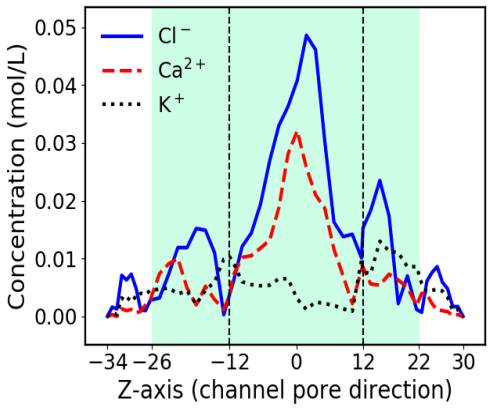


(f) Difference on concentrations by Mesh 3

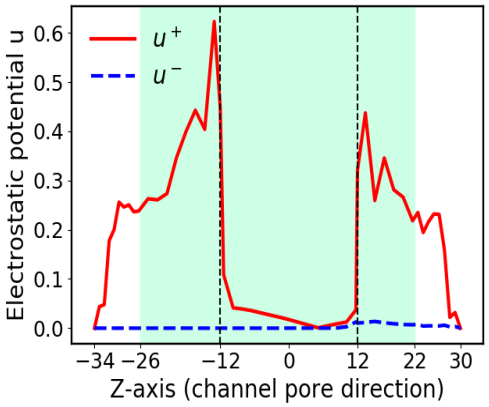
Figure 5.9: Differences of u and c_i generated by the PNPic model using Dirichlet and Neumann boundary value conditions for gA in a salt solution of KCl (0.5 M/L) on Meshes 1, 2, and 3 given in Table 5.3.



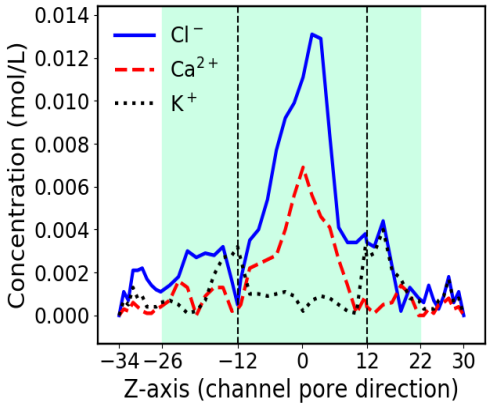
(a) Difference on potentials by Mesh 1



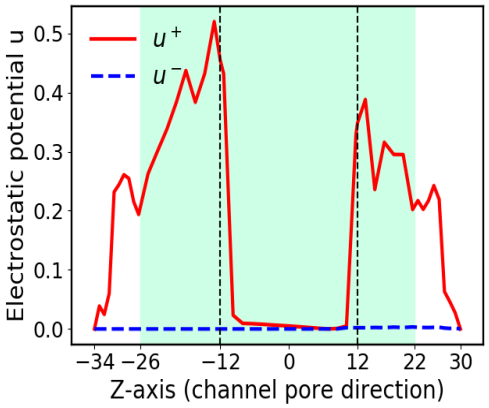
(b) Difference on concentrations by Mesh 1



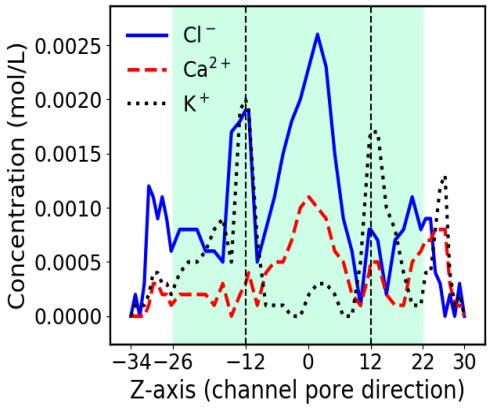
(c) Difference on potentials by Mesh 2



(d) Difference on concentrations by Mesh 2

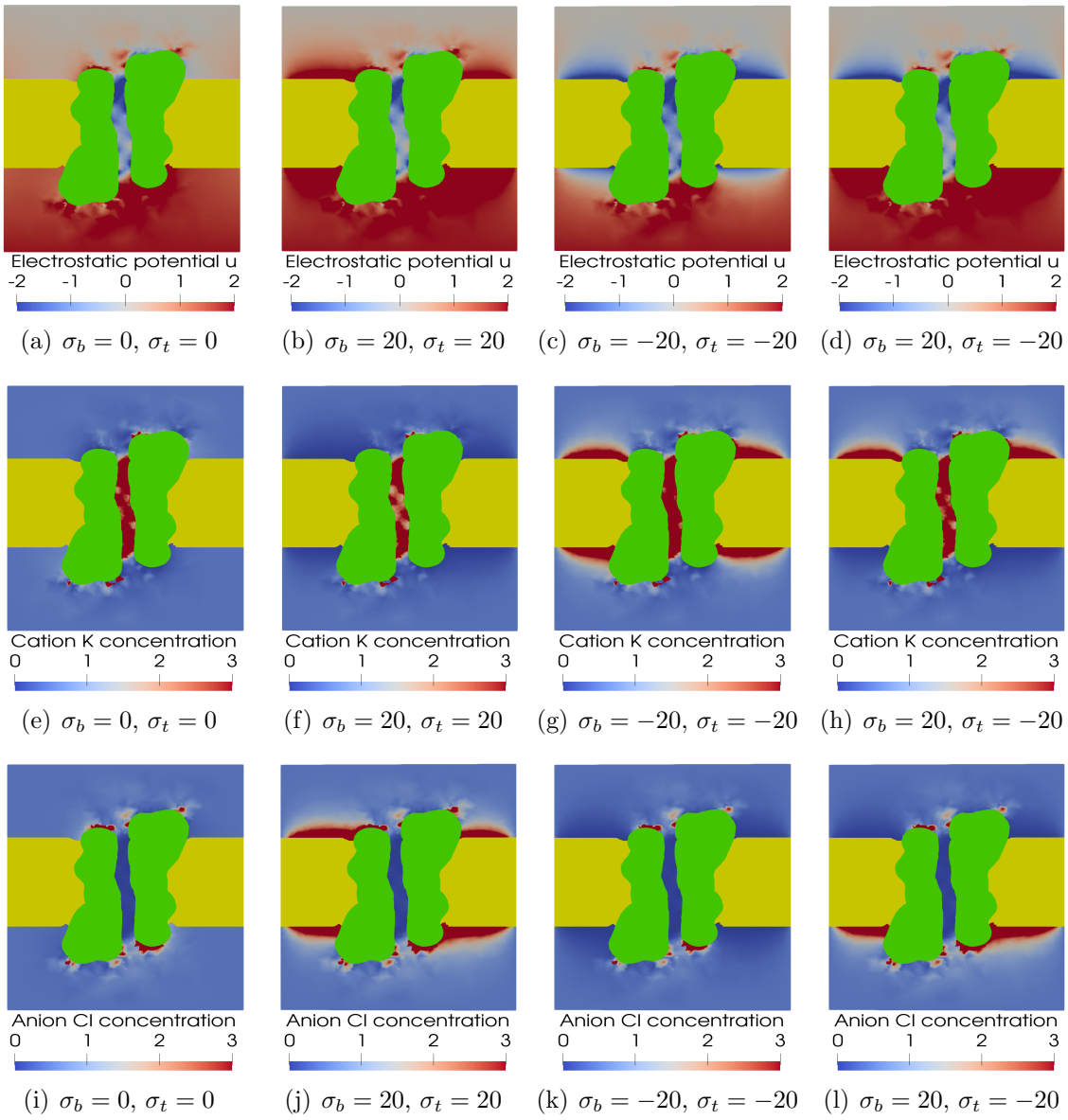


(e) Difference on potentials by Mesh 3



(f) Difference on concentrations by Mesh 3

Figure 5.10: Differences of u and c_i generated by the PNPic model using Dirichlet and Neumann boundary value conditions for mVDAC1 in a mixture solution of KCl (0.5 M/L) and $CaCl_2$ (0.1 M/L) on Meshes 1, 2, and 3 given in Table 5.3.



(a) $\sigma_b = 0$, $\sigma_t = 0$ (b) $\sigma_b = 20$, $\sigma_t = 20$ (c) $\sigma_b = -20$, $\sigma_t = -20$ (d) $\sigma_b = 20$, $\sigma_t = -20$

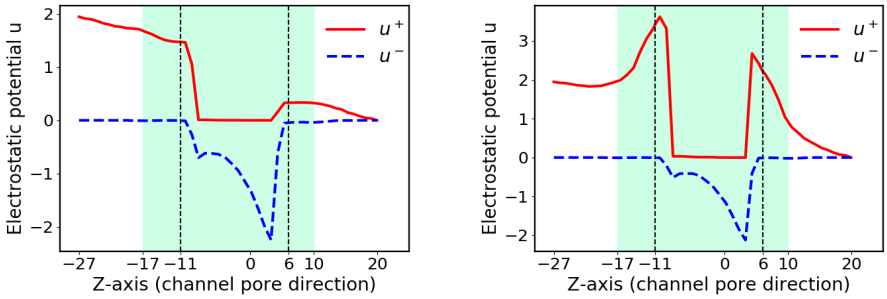
(e) $\sigma_b = 0$, $\sigma_t = 0$ (f) $\sigma_b = 20$, $\sigma_t = 20$ (g) $\sigma_b = -20$, $\sigma_t = -20$ (h) $\sigma_b = 20$, $\sigma_t = -20$

(i) $\sigma_b = 0$, $\sigma_t = 0$ (j) $\sigma_b = 20$, $\sigma_t = 20$ (k) $\sigma_b = -20$, $\sigma_t = -20$ (l) $\sigma_b = 20$, $\sigma_t = -20$

Figure 5.11: Comparisons of the electrostatic potential u and ion concentrations c_i by PNPic model with Dirichlet boundary value conditions and membrane surface charges for gA in a salt solution of KCl (0.5 M/L) on a cross-section $x = 0$. The protein and membrane regions are colored in green and yellow, respectively. Here $g_b = 1.946$ (i.e., 50mV).

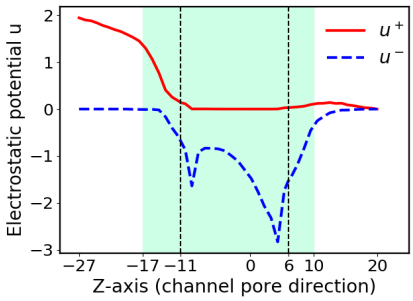
ures 5.12(b) and 5.12(d)), and anionic concentrations from 0 to 12 (see Figures 5.12(f) and 5.12(h)), and the anion Cl concentrations are essentially zero inside the channel for all cases. These numerical tests further validate that our PNPic model can predicate the important properties of gA channel, such as cation selectivity.

We also calculated the I-V curves of gA channel to further explore the effects of surface charges. The numerical results were plotted in Figure 5.13. They show that current-voltage

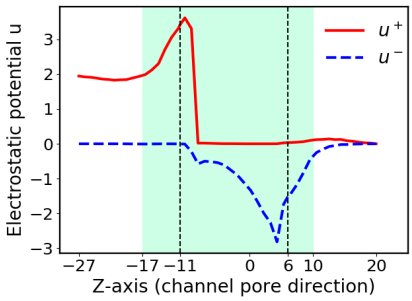


(a) $\sigma_b = 0$, $\sigma_t = 0$

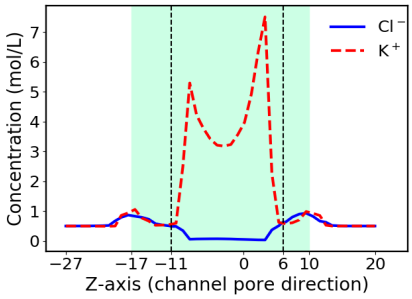
(b) $\sigma_b = 20$, $\sigma_t = 20$



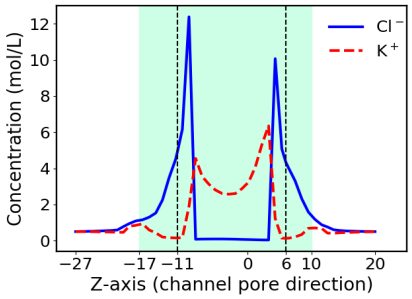
(c) $\sigma_b = -20$, $\sigma_t = -20$



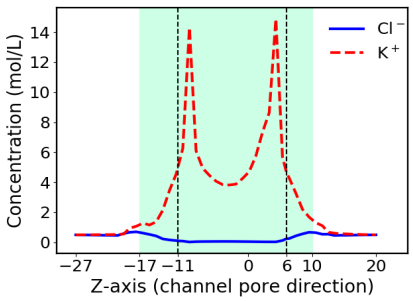
(d) $\sigma_b = 20$, $\sigma_t = -20$



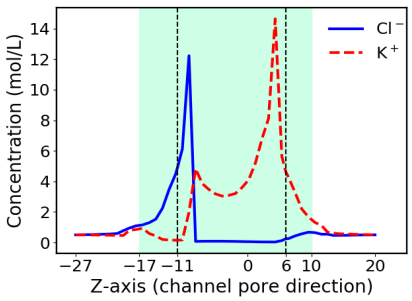
(e) $\sigma_b = 0$, $\sigma_t = 0$



(f) $\sigma_b = 20$, $\sigma_t = 20$



(g) $\sigma_b = -20$, $\sigma_t = -20$



(h) $\sigma_b = 20$, $\sigma_t = -20$

Figure 5.12: Comparisons of the electrostatic potential u and concentrations c_i generated by PNPic model using Dirichlet boundary value conditions and membrane surface charges for gA (Mesh 1) in a salt solution of KCl (0.5 M/L). Here $g_b = 1.946$ (i.e., 50mV).

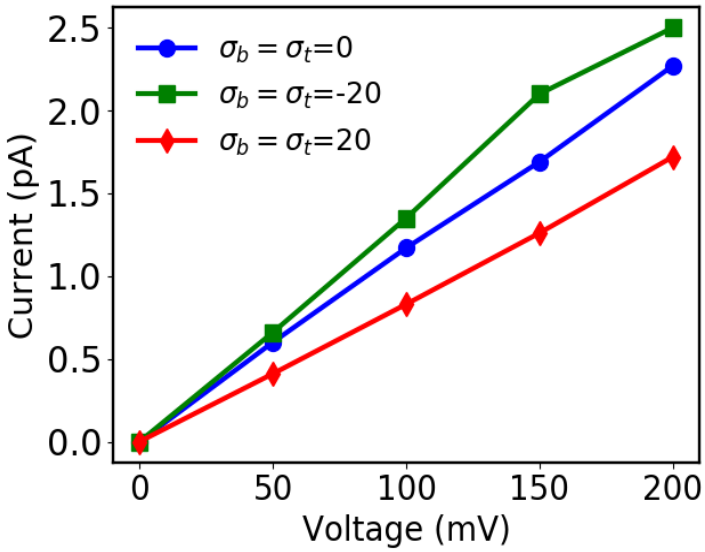
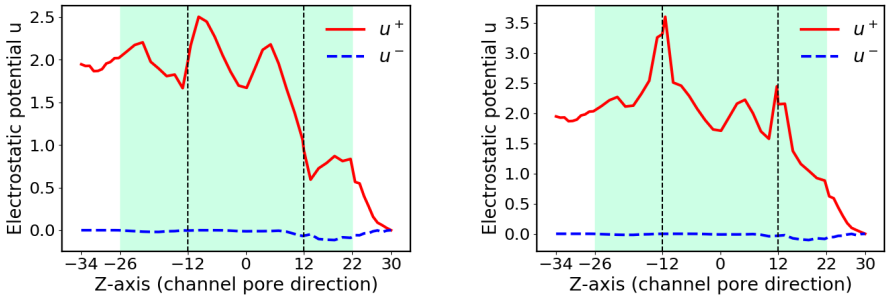


Figure 5.13: Current-voltage curves for gA ion channel with different membrane surface charges in a salt solution of KCl (0.1 M/L).

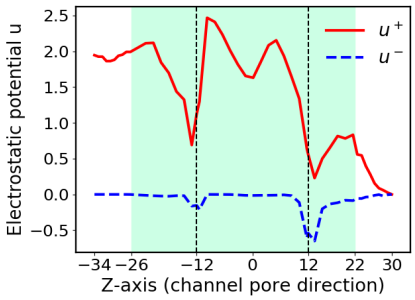
curves were affected significantly by membrane surface charges. Here a symmetrical 0.1M bulk concentration, i.e., $c^b_{i,t} = c^b_{i,b} = 0.1$ M/L was used. The presence of negative charges on the membrane surface increased the current value that passes through the ion channel pore, because more cations K^+ were attracted into the channel pore. Similarly, positive charges on the membrane surface can expel cations, producing a lower current. These test results are in agreement with experiment observations [54].

We also did tests for mVDAC1 in a mixture solution of KCl (0.1 M/L) and $CaCl_2$ (0.1 M/L). Here we set $\theta = 0.35$ and $g_b = 1.946$ (i.e., 50 mV). The numerical results were displayed in Figure 5.14. They are similar to the case of gA ion channel.

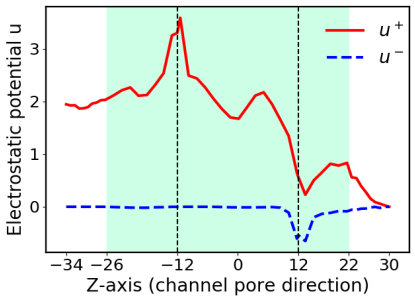


(a) $\sigma_b = 0$, $\sigma_t = 0$

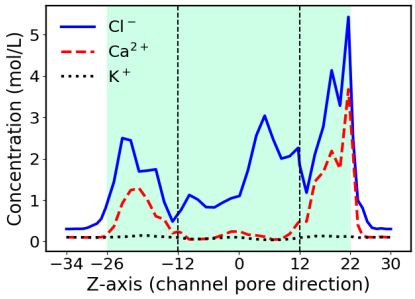
(b) $\sigma_b = 20$, $\sigma_t = 20$



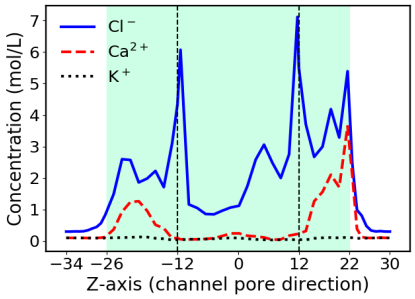
(c) $\sigma_b = -20$, $\sigma_t = -20$



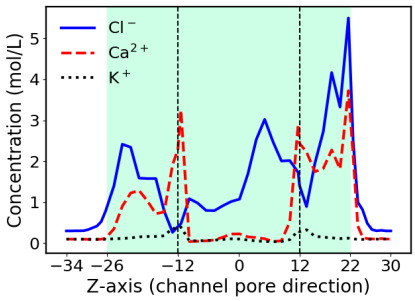
(d) $\sigma_b = 20$, $\sigma_t = -20$



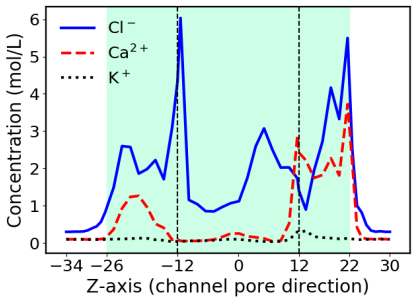
(e) $\sigma_b = 0$, $\sigma_t = 0$



(f) $\sigma_b = 20$, $\sigma_t = 20$



(g) $\sigma_b = -20$, $\sigma_t = -20$



(h) $\sigma_b = 20$, $\sigma_t = -20$

Figure 5.14: Comparisons of the electrostatic potential u and concentrations c_i generated by the PNPic model using Dirichlet boundary value conditions and membrane surface charges for mVDAC1 in a mixture solution of of KCl (0.1 M/L) and $CaCl_2$ (0.1 M/L). Here $g_b = 1.946$ (i.e., 50 mV).

Chapter 6

Conclusions and future work

In this dissertation, an effective finite element algorithm for solving a PNP ion channel model using Dirichlet boundary value conditions and an improved ion channel finite element mesh algorithm have been presented. They have been implemented as a high quality ion channel simulation software package. By using this package, many numerical tests were done to explore some important issues arisen from ion channel simulations by PNP ion channel models. Here is a summary of the main results of this dissertation:

• A PNP ion channel (PNPic) model using Dirichlet boundary value condition and membrane surface charge is derived.

• An effective PNPic finite element solver is presented.

• New numerical algorithms for generating a triangular surface mesh of the simulation box domain and for constructing a tetrahedral mesh of the membrane region are presented and implemented for an improved ion channel mesh generation package.

• The effects of boundary value conditions, simulation box domain sizes, and membrane surface charges on PNPic modeling are explored numerically.

Based on the current work, we will continue to do the following work in the future:

• In order to take the advantages of both finite difference and finite element methods, a new finite element and finite difference hybrid method for solving a PNP ion channel model will be developed.

• The PNPic model is effective but ignores the finite volume effect of ion particles. In fact, it has been known that ionic size is important for ion transport in highly confined channels. So an improved PNPic model needs to be constructed to consider ionic size effects.

• When we solve the PNPic system numerically, many highly indefinite linear systems

may be produced, which cannot be solved by any current iterative method such as the GMRES method preconditioned by ILU due to very poor convergent rates. In order to deal with this numerical issue, we will develop preconditioners and new iterative algorithms for our PNPic finite element solver.

Bibliography

[1] N. A. Baker, D. Sept, S. Joseph, M. J. Holst, and J. A. McCammon. Electrostatics of nanosystems: application to microtubules and the ribosome. *Proceedings of the National Academy of Sciences*, 98(18):10037–10041, 2001.

[2] S. Berneche and B. Roux. A microscopic view of ion conduction through the K^+ channel. *Proceedings of the National Academy of Sciences*, 100(15):8644–8648, 2003.

[3] D. S. Bolintineanu, A. Sayyed-Ahmad, H. T. Davis, and Y. N. Kaznessis. Poisson-Nernst-Planck models of nonequilibrium ion electrodiffusion through a protegrin transmembrane pore. *PLOS Computational Biology*, 5(1):e1000277, 2009.

[4] A. E. Cardenas, R. D. Coalson, and M. G. Kurnikova. Three-dimensional Poisson-Nernst-Planck theory studies: Influence of membrane electrostatics on gramicidin A channel conductance. *Biophysical Journal*, 79(1):80–93, 2000.

[5] D. Chen, Z. Chen, C. Chen, W. Geng, and G.-W. Wei. MIBPB: a software package for electrostatic analysis. *Journal of Computational Chemistry*, 32(4):756–770, 2011.

[6] M. Chen, B. Tu, and B. Lu. Triangulated manifold meshing method preserving molecular surface topology. *Journal of Molecular Graphics and Modelling*, 38:411–418, 2012.

[7] O. P. Choudhary, R. Ujwal, W. Kowallis, R. Coalson, J. Abramson, and M. Grabe. The electrostatics of VDAC: implications for selectivity and gating. *Journal of Molecular Biology*, 396(3):580–592, 2010.

[8] S.-H. Chung and B. Corry. Three computational methods for studying permeation, selectivity and dynamics in biological ion channels. *Soft Matter*, 1(6):417–427, 2005.

[9] C. D. Cole, A. S. Frost, N. Thompson, M. Cotten, T. A. Cross, and D. D. Busath. Noncontact dipole effects on channel permeation. VI. 5F-and 6F-Trp gramicidin channel currents. *Biophysical Journal*, 83(4):1974–1986, 2002.

[10] B. Corry. Computer simulation of ion channels. *Computational Biophysics of Membrane Proteins*, pages 161–96, 2016.

[11] S. Decherchi and W. Rocchia. A general and robust ray-casting-based algorithm for triangulating surfaces at the nanoscale. *PloS One*, 8(4):e59744, 2013.

[12] O. Devillers, S. Pion, and M. Teillaud. Walking in a triangulation. *Proceedings of the Seventeenth Annual Symposium on Computational Geometry*, pages 106–114, 2001.

[13] T. J. Dolinsky, J. E. Nielsen, J. A. McCammon, and N. A. Baker. PDB2PQR: an automated pipeline for the setup of Poisson-Boltzmann electrostatics calculations. *Nucleic acids research*, 32(suppl_2):W665–W667, 2004.

[14] D. A. Doyle, J. M. Cabral, R. A. Pfuetzner, A. Kuo, J. M. Gulbis, S. L. Cohen, B. T. Chait, and R. MacKinnon. The structure of the potassium channel: molecular basis of K^+ conduction and selectivity. *Science*, 280(5360):69–77, 1998.

[15] Q. Du, V. Faber, and M. Gunzburger. Centroidal Voronoi tessellations: Applications and algorithms. *SIAM Review*, 41(4):637–676, 1999.

[16] A. Fick. Ueber diffusion. *Annalen der Physik*, 170(1):59–86, 1855.

[17] F. Fogolari, A. Brigo, and H. Molinari. The Poisson-Boltzmann equation for biomolecular electrostatics: a tool for structural biology. *Journal of Molecular Recognition*, 15(6):377–392, 2002.

[18] C. Geuzaine, J.-F. Remacle, and P. Dular. Gmsh: a three-dimensional finite element mesh generator. *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331, 2009.

[19] A. Golovnev and S. Trimper. Steady state solution of the Poisson-Nernst-Planck equations. *Physics Letters A*, 374(28):2886–2889, 2010.

[20] A. Golovnev and S. Trimper. Analytical solution of the Poisson-Nernst-Planck equations in the linear regime at an applied dc-voltage. *The Journal of chemical physics*, 134(15):154902, 2011.

[21] U. Hollerbach, D.-P. Chen, and R. S. Eisenberg. Two-and three-dimensional Poisson-Nernst-Planck simulations of current flow through gramicidin a. *Journal of Scientific Computing*, 16(4):373–409, 2001.

[22] H. Hwang, G. C. Schatz, and M. A. Ratner. Incorporation of inhomogeneous ion diffusion coefficients into kinetic lattice grand canonical Monte Carlo simulations and application to ion current calculations in a simple model ion channel. *The Journal of Physical Chemistry A*, 111(49):12506–12512, 2007.

[23] E. Y. Isacoff, L. Y. Jan, and D. L. Minor Jr. Conduits of life's spark: a perspective on ion channel research since the birth of neuron. *Neuron*, 80(3):658–674, 2013.

[24] M. Ø. Jensen, D. W. Borhani, K. Lindorff-Larsen, P. Maragakis, V. Jogini, M. P. Eastwood, R. O. Dror, and D. E. Shaw. Principles of conduction and hydrophobic gating in K^+ channels. *Proceedings of the National Academy of Sciences*, 107(13):5833–5838, 2010.

[25] M. Ø. Jensen, V. Jogini, D. W. Borhani, A. E. Leffler, R. O. Dror, and D. E. Shaw. Mechanism of voltage gating in potassium channels. *Science*, 336(6078):229–233, 2012.

[26] J. W. Jerome. Consistency of semiconductor modeling: an existence/stability analysis for the stationary van Roosbroeck system. *SIAM Journal on Applied Mathematics*, 45(4):565–590, 1985.

[27] Y. Jiang, Y. Xie, J. Ying, D. Xie, and Z. Yu. Sdpbs web server for calculation of electrostatics of ionic solvated biomolecules. *Computational and Mathematical Biophysics*, 1(open-issue), 2015.

[28] S. Jo, M. Vargyas, J. Vasko-Szedlar, B. Roux, and W. Im. PBEQ-Solver for online visualization of electrostatic potential of biomolecules. *Nucleic Acids Research*, 36(suppl_2):W270–W275, 2008.

[29] V. Jogini and B. Roux. Electrostatics of the intracellular vestibule of K^+ channels. *Journal of Molecular Biology*, 354(2):272–288, 2005.

[30] J. B. Klauda, R. M. Venable, J. A. Freites, J. W. O'Connor, D. J. Tobias, C. Mondragon-Ramirez, I. Vorobyov, A. D. MacKerell Jr, and R. W. Pastor. Update of the CHARMM all-atom additive force field for lipids: validation on six lipid types. *The Journal of Physical Chemistry B*, 114(23):7830–7843, 2010.

[31] K.-C. Lee, S. Huo, and T. Cross. Lipid-peptide interface: valine conformation and dynamics in the gramicidin channel. *Biochemistry*, 34(3):857–867, 1995.

[32] 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*, 1(open-issue), 2015.

[33] T. Liu, M. Chen, and B. Lu. Efficient and qualified mesh generation for gaussian molecular surface using adaptive partition and piecewise polynomial approximation. *SIAM Journal on Scientific Computing*, 40(2):B507–B527, 2018.

[34] W. Liu. One-dimensional steady-state Poisson-Nernst-Planck systems for ion channels with multiple ion species. *Journal of Differential Equations*, 246(1):428–451, 2009.

[35] A. Logg, K.-A. Mardal, and G. Wells. Automated solution of differential equations by the finite element method the FEniCS book. *Springer*, 2012.

[36] M. A. Lomize, I. D. Pogozheva, H. Joo, H. I. Mosberg, and A. L. Lomize. OPM database and PPM web server: resources for positioning of proteins in membranes. *Nucleic Acids Research*, 40(D1):D370–D376, 2012.

[37] 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(19):6979–6994, 2010.

[38] R. Luo, L. David, and M. K. Gilson. Accelerated Poisson-Boltzmann calculations for static and dynamic systems. *Journal of Computational Chemistry*, 23(13):1244–1253, 2002.

[39] J. D. Madura, J. M. Briggs, R. C. Wade, M. E. Davis, B. A. Luty, A. Ilin, J. Antosiewicz, M. K. Gilson, B. Bagheri, L. R. Scott, et al. Electrostatics and diffusion of molecules in solution: simulations with the university of houston brownian dynamics program. *Computer Physics Communications*, 91(1-3):57–95, 1995.

[40] C. Maffeo, S. Bhattacharya, J. Yoo, D. Wells, and A. Aksimentiev. Modeling and simulation of ion channels. *Chemical Reviews*, 112(12):6250–6284, 2012.

[41] S. R. Mathur and J. Y. Murthy. A multigrid method for the Poisson-Nernst-Planck equations. *International Journal of Heat and Mass Transfer*, 52(17-18):4031–4039, 2009.

[42] J. A. McCammon, B. R. Gelin, and M. Karplus. Dynamics of folded proteins. *Nature*, 267(5612):585–590, 1977.

[43] B. Mertins, G. Psakis, W. Grosse, K. C. Back, A. Salisowski, P. Reiss, U. Koert, and L.-O. Essen. Flexibility of the N-terminal mVDAC1 segment controls the channel's gating behavior. *PLoS One*, 7(10), 2012.

[44] M. Mirzadeh and F. Gibou. A conservative discretization of the Poisson-Nernst-Planck equations on adaptive Cartesian grids. *Journal of Computational Physics*, 274:633–653, 2014.

[45] M. Mock. On equations describing steady-state carrier distributions in a semiconductor device. *Communications on Pure and Applied Mathematics*, 25(6):781–792, 1972.

[46] S. Moshtarikhah, N. Oppers, M. de Groot, J. Keurentjes, J. Schouten, and J. van der Schaaf. Nernst–Planck modeling of multicomponent ion transport in a Nafion membrane at high current density. *Journal of Applied Electrochemistry*, 47(1):51–62, 2017.

[47] 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(5):2349–2363, 2000.

[48] W. Nernst. Die elektromotorische wirksamkeit der jonen. *Zeitschrift für physikalische Chemie*, 4(1):129–181, 1889.

[49] S. Y. Noskov, S. Berneche, and B. Roux. Control of ion selectivity in potassium channels by electrostatic and dynamic properties of carbonyl ligands. *Nature*, 431(7010):830–834, 2004.

[50] J. P. Overington, B. Al-Lazikani, and A. L. Hopkins. How many drug targets are there? *Nat Rev Drug Discov*, 5:993–6, 2006.

[51] S. J. Paddison, D. W. Reagor, and T. A. Zawodzinski Jr. High frequency dielectric studies of hydrated Nafion. *Journal of Electroanalytical Chemistry*, 459(1):91–97, 1998.

[52] M. Planck. Ueber die erregung von electricität und wärme in electrolyten. *Annalen der Physik*, 275(2):161–186, 1890.

[53] W. Rocchia, E. Alexov, and B. Honig. Extending the applicability of the nonlinear Poisson-Boltzmann equation: multiple dielectric constants and multivalent ions. *The Journal of Physical Chemistry B*, 105(28):6507–6514, 2001.

[54] T. K. Rostovtseva, V. M. Aguilella, I. Vodyanoy, S. M. Bezrukov, and V. A. Parsegian. Membrane surface-charge titration probed by gramicidin A channel conductance. *Biophysical Journal*, 75(4):1783–1792, 1998.

[55] B. Roux, T. Allen, S. Berneche, and W. Im. Theoretical and computational models of biological ion channels. *Quarterly Reviews of Biophysics*, 37(1):15–103, 2004.

[56] B. Roux, S. Berneche, and W. Im. Ion channels, permeation, and electrostatics: insight into the function of KcsA. *Biochemistry*, 39(44):13295–13306, 2000.

[57] W. Rudin. Functional analysis, McGraw Hill. *Inc, New York*, 1991.

[58] M. J. Sampson, R. S. Lovell, and W. J. Craigen. The murine voltage-dependent anion channel gene family conserved structure and function. *Journal of Biological Chemistry*, 272(30):18966–18973, 1997.

[59] R. Sarges and B. Witkop. Gramicidin A. IV. Primary sequence of valine and isoleucine Gramicidin A. *Journal of the American Chemical Society*, 86(9):1862–1863, 1964.

[60] N. Schmid, A. P. Eichenberger, A. Choutko, S. Riniker, M. Winger, A. E. Mark, and W. F. van Gunsteren. Definition and testing of the GROMOS force-field versions 54A7 and 54B7. *European Biophysics Journal*, 40(7):843, 2011.

[61] H. Si. TetGen-A quality tetrahedral mesh generator and three-dimensional delaunay triangulator. *Weierstrass Institute for Applied Analysis and Stochastic, Berlin, Germany*, 81, 2006.

[62] H. Si and K. Gärtner. Meshing piecewise linear complexes by constrained Delaunay tetrahedralizations. *Proceedings of the 14th International Meshing Roundtable*, pages 147–163, 2005.

[63] A. K. Singh, L. L. McGoldrick, K. Saotome, and A. I. Sobolevsky. X-ray crystallography of TRP channels. *Channels*, 12(1):137–152, 2018.

[64] P. Sjoberg. MolSurf-A generator of chemical descriptors for QSAR. *Computer-Assisted Lead Finding and Optimization*, pages 83–92, 1997.

[65] Y. Song, Y. Zhang, T. Shen, C. L. Bajaj, J. A. McCammon, and N. A. Baker. Finite element solution of the steady-state Smoluchowski equation for rate constant calculations. *Biophysical Journal*, 86(4):2017–2029, 2004.

[66] C. Southan, J. L. Sharman, H. E. Benson, E. Faccenda, A. J. Pawson, S. P. Alexander, O. P. Buneman, A. P. Davenport, J. C. McGrath, J. A. Peters, et al. The IUPHAR/BPS Guide to PHARMACOLOGY in 2016: towards curated quantitative interactions between 1300 protein targets and 6000 ligands. *Nucleic Acids Research*, 44(D1):D1054–D1068, 2016.

[67] B. Tu, S. Bai, M. Chen, Y. Xie, L. Zhang, and B. Lu. A software platform for continuum modeling of ion channels based on unstructured mesh. *Computational Science & Discovery*, 7(1):014002, 2014.

[68] 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(46):17742–17747, 2008.

[69] D. Urry. Conformation of biological molecules and polymers. *The Fifth Jerusalem Symposium on Quantum Chemistry and Biochemistry*, pages 723–734, 1973.

[70] J. Wang, R. M. Wolf, J. W. Caldwell, P. A. Kollman, and D. A. Case. Development and testing of a general amber force field. *Journal of Computational Chemistry*, 25(9):1157–1174, 2004.

[71] P. Weetman, S. Goldman, and C. Gray. Use of the Poisson-Boltzmann equation to estimate the electrostatic free energy barrier for dielectric models of biological ion channels. *The Journal of Physical Chemistry B*, 101(31):6073–6078, 1997.

[72] D. Xie. New solution decomposition and minimization schemes for Poisson-Boltzmann equation in calculation of biomolecular electrostatics. *Journal of Computational Physics*, 275:294–309, 2014.

[73] 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(3):218–230, 2020.

[74] D. Xie and Z. Chao. A finite element iterative solver a PNP ion channel model by neumann boundary condition and membrane surface charge. *Journal of Computational Physics, Submitted*, 2020.

[75] D. Xie and B. Lu. An effective finite element iterative solver for a Poisson-Nernst-Planck ion channel model with periodic boundary conditions. *SIAM Journal on Scientific Computing, Accepted*, 2020.

[76] S. Yu and G. Wei. Three-dimensional matched interface and boundary (MIB) method for treating geometric singularities. *Journal of Computational Physics*, 227(1):602–632, 2007.

[77] Z. Yu, M. J. Holst, and J. A. McCammon. High–fidelity geometric modeling for biomedical applications. *Finite Elements in Analysis and Design*, 44(11):715–723, 2008.

[78] Q. Zheng, D. Chen, and G.-W. Wei. Second-order Poisson-Nernst-Planck solver for ion transport. *Journal of Computational Physics*, 230(13):5239–5262, 2011.

CURRICULUM VITAE

Education

- **Ph.D. in Applied Mathematics** **Sep. 2016 - Aug. 2020**
  UWM, Milwaukee, WI.
- **Visiting Scholar** **Oct. 2015 - Apr. 2016**
  UWM, Milwaukee, WI.
- **Doctoral study in Applied Mathematics** **Sep. 2013 - Sep. 2015**
  East China Normal University, Shanghai, China.
- **M.S. in Applied Mathematics** **Jun. 2013**
  Wenzhou University, Wenzhou, China.
- **B.S. in Applied Mathematics** **Jun. 2009**
  Hunan University of Science and Technology, Xiangtan, China.

Selected List of Awards

1. **Distinguished Graduate Student Fellowship**, Graduate school, UWM, August 2018.
2. **Morris and Miriam Marden Award in Mathematics**, Department of Mathematical Sciences, UWM, May 2020.
3. **AMS graduate student travel award**, AMS, January 2020.
4. **Travel award for the NSF-CBMS Conference**, University of Alabama, May 2019.
5. **Chancellor's Graduate Student Award**, UWM, September 2018, September 2019.
6. **Graduate Travel Award**, Graduate School, UWM, April 2017, May 2018, June 2019.
7. **Graduate Student Conference Travel Award**, Department of Mathematical Sciences, UWM, April 2017, May 2018, June 2019.
8. **Research Excellence Award**, Department of Mathematical Sciences, UWM, September 2016, September 2017, September 2018.

Selected List of Publications

1. Dexuan Xie, **Zhen Chao**. A Finite Element Iterative Solver for an Improved PNP Ion Channel Model by Neumann Boundary Condition and Membrane Surface Charge. **Journal of Computational Physics**, under revision.

2. **Zhen Chao**, Dexuan Xie and Ahmed Sameh. Preconditioners for nonsymmetric indefinite linear systems. **Journal of Computational and Applied Mathematics**, 367: 112436, 2020.

3. Yunying Huang, **Zhen Chao** and Guoliang Chen. Spectral properties of the matrix splitting preconditioners for generalized saddle point problems. **Journal of Computational and Applied Mathematics**, 332: 1-12, 2018.

4. **Zhen Chao**, Guoliang Chen and Ye Guo. On the semi-convergence of regularized HSS iteration methods for singular saddle point problems. **Computers and Mathematics with Applications**, 76: 438-450, 2018.

5. **Zhen Chao**, Kai Wang, Chao Zhu and Yanling Zhu. Almost Sure and Moment Exponential Stability of Regime-Switching Jump Diffusions. **SIAM Journal on Control and Optimization**, 55(6): 3458-3488, 2017.

6. **Zhen Chao** and Guoliang Chen. A generalized modified HSS method for singular complex symmetric linear systems. **Numerical Algorithms**, 73(1): 77-89, 2016.