IMAGE CHARGE SOLVATION MODEL (ICSM) FOR SIMULATING BIOMOLECULES AND KCSA ION-CHANNELS

by

Katherine A. Baker

A dissertation submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in
Applied Mathematics

Charlotte

2014

Approved by:

_____
Dr. Shaozhong Deng

_____
Dr. Wei Cai

_____
Dr. Duan Chen

_____
Dr. Xintao Wu

ABSTRACT

KATHERINE A. BAKER. Image Charge Solvation Model (ICSM) for simulating biomolecules and KcsA ion channels. (Under the direction of DR. WEI CAI)

We present an order N method for calculating electrostatic interactions that has been integrated into the molecular dynamics portion of the TINKER Molecular Modeling package. This method, termed the Image-Charge Solvation Model (ICSM), and introduced previously by Dr. Lin et al. (1) in 2009, is a hybrid electrostatic approach that combines the strengths of both explicit and implicit representations of the solvent. In this model, a multiple-image method is used to calculate reaction fields due to the implicit solvent while the Fast Multipole Method (FMM) is used to calculate the Coulomb interactions for all charges, including the charges in the explicit solvent part.

The integrated package is validated through simulations of liquid water. The results are compared with those obtained by the Particle Mesh Ewald (PME) method that is built in the TINKER package. Timing performance of TINKER with the integrated ICSM is benchmarked on bulk water as a function of the size of the system. In particular, timing analysis results show that the ICSM outperforms the PME for sufficiently large systems with the break-even point at around 30,000 particles in the simulated system. To demonstrate the capability of the package on large macromolecules, the model is used to simulate the potassium channel KcsA.

ACKNOWLEDGEMENTS

I would first like to thank Dr. Wei Cai, Dr. Andrei Baumketner, Dr. Shaozhong Deng, and Dr. Donald Jacobs for their immense patience especially in the beginning when I did not know anything about Fortran or Molecular Dynamics or even Potential Energy. Dr. Cai, as my advisor, provided what seemed like unlimited support in many areas including, but not limited to, educational, emotional, and financial. His encouragement throughout the long years of this research was just invaluable. Dr. Deng's insight in scholarly works made my first publication possible, and his knowledge of the LE and FMM made my completion of the coding possible.

I would also like to thank Dr. Yuchun Lin without whose assistance the transfer of the program from water to proteins would not have been possible. He was very generous with his time and insight even though he was busy trying to complete his own research. I also owe my gratitude to Mark Hamrick. Often overlooked, Mark has been inspirational in his ability to help me with hardware and software issues. He never lost patience through all of my constant questions and network issues.

I extend my gratitude to Dr. Chen and Dr. Wu, the other members of my committee, for their generous input on this thesis.

I must also thank my whole family. They each contributed to this paper in various ways. Especially my beautiful mother, Betty Cross, who was incredibly patient and supportive. Also my daughter, Cassie Damery, whose encouragement and cheerleading got me through many debugging nightmares. And last, but certainly not least, my wonderful granddaughter Payton. Her numerous hugs and kisses were instrumental in the completion of this dissertation.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

1.1.1 Protein Structure

One of the most exciting and challenging areas of study for scientists today is the investigation of proteins. Of all known molecules, proteins might be the most complex and functionally significant (2). Proteins have many important functions such as acting as catalysts, transportation and storage of other molecules such as oxygen, providing mechanical support and immune protection, as well as transmitting nerve impulses, and they also control growth and differentiation (3). The structure of a protein drives their dynamics, and therefore their function, so we begin with a short overview of that structure (4).

Proteins are built from a simple string of amino acids called the primary structure. Although there are at least 300 amino acids in nature, only 20 are found in the structures of proteins (Figure 2). Amino acids are defined as small molecules containing an amino group (NH_2), a carboxyl group (COOH) and an R group or side chain (Figure 1). Within a protein, multiple amino acids are linked together by peptide bonds which are formed when the amino group from one amino acid links to the carboxyl group of another. Each amino acid is unique based on the characteristics of its side chain (5).

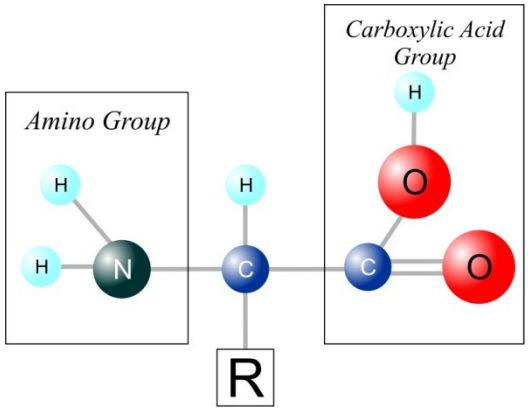

FIGURE 1: The basic structure of amino acids.

The largest group are the amino acids which have nonpolar side chains. These include alanine, valine, leucine, isoleucine, proline, phenylalanine, tryptophan, cysteine and methionine. Since these amino acids are nonpolar, they have a low propensity to be in contact with polar water and are called hydrophobic. Other amino acids have side chains that have a positive or negative charge. These include lysine and arginine, which are positively charged, and aspartate and glutamate, which are negatively charged. There are also six amino acids with polar side chains: serine, threonine, asparagine, glutamine, histidine and tyrosine. The final amino acid, glycine, has no side chain and often provides flexibility to the final protein structure (6). These side chains are instrumental in the formation of the secondary structure based on how they interact with the side chains of other amino acids in the protein. In general, charged side chains can form ionic bonds with other side chains while polar side chains might form hydrogen bonds. The nonpolar hydrophobic side chains have weak van der Waals interactions with other side chains.

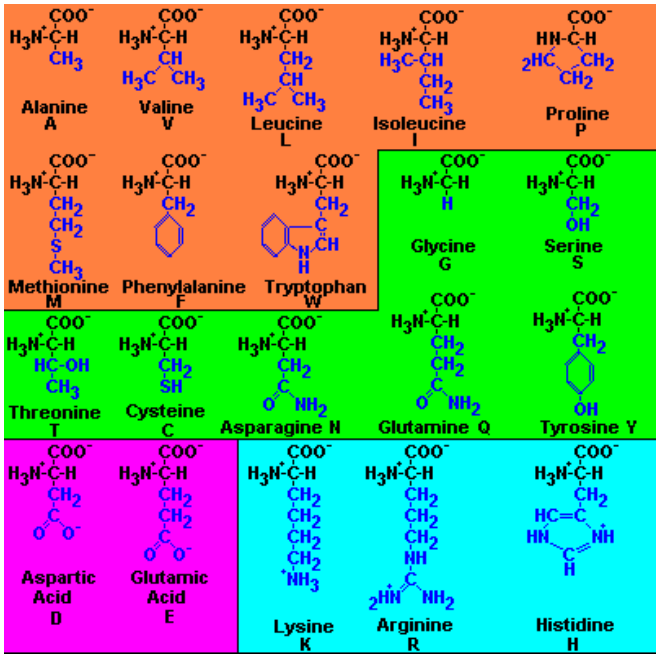

FIGURE 2: The twenty amino acids found in proteins. The amino acids in the orange area are non-polar (hydrophobic.) The ones in the green area have polar side chains. The pink area contains amino acids with a negative charge while the blue are basic. Image is by Dr. Robert J. Huskey (retired) University of Virginia **(7)**.

The secondary structure is made up of two main structures, the alpha helix and beta sheets (6). Alpha helices are formed when the carbonyl oxygen atoms point towards the amide NH groups of an amino acid four residues away in the chain. This interaction forms a hydrogen bond. On the other hand, beta sheets form when strands of the chain lie parallel to each other and the carbonyl oxygen in one strand form hydrogen bonds with the amino hydrogen in the adjacent strand. Beta sheets are classified as parallel or anti-parallel depending on whether the N-terminus is aligned with the N-terminus or the C-terminus of the other strand. Of these two, the anti-parallel beta sheet is considered to be the more stable conformation (8). Proteins normally have several of these structures connected by loops (Figure 3). The entire ensemble created from the folds and formations of these structures is called the tertiary structure.

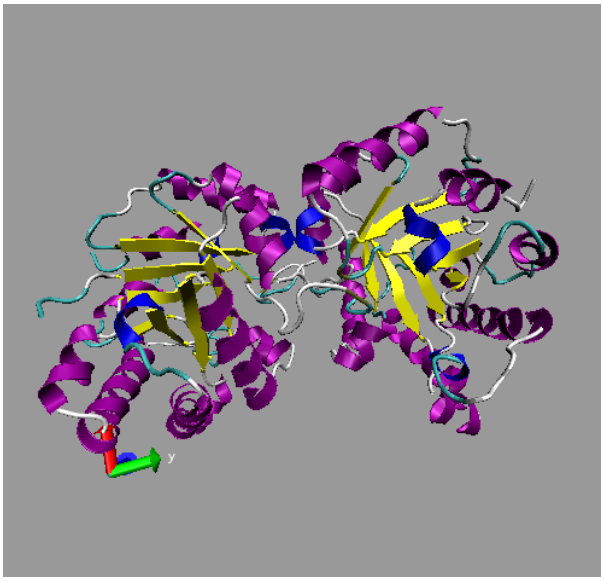

FIGURE 3: Image of Triose Phosphate Isomerase showing the secondary structure with alpha helices (blue and purple) and parallel beta sheets(yellow). Image created by VMD software.

The goal of the tertiary structure is to reach maximum stability and the lowest energy state. One important aspect for this stabilization is the formation of disulfide bridges by oxidation of the sulfhydryl groups on cysteine, which hold parts of the chain together covalently. In addition, salt bridges form due to ionic interaction of positively and negatively charged side chains, which provides further stabilization for the overall form of the protein (8). Nuclear Magnetic Resonance (NMR) has been instrumental in the investigation of the structure of proteins however it gives little in terms of the function. Molecular dynamics simulations provide a means to explore the link between the structure and dynamics of proteins (4).

1.1.2 Molecular Dynamics

Molecular Dynamic simulations are employed to calculate the positions of a group of atoms in a molecular system over time using the discretization of Newton's laws of motion over a period of time (9). In the earliest implementations of this concept, scientists attempted to demonstrate these trajectories using physical rods and balls. In the late 50's and early 60's, Alder and Wainwright and then later Rahman first introduced the method (10; 11) and in 1974, Rahman and Stillinger used the method to simulate a realistic liquid system (12). With the improvement of computer systems, MD simulations have been used to study systems ranging in size from tens to hundreds of thousands.

One standard numerical method for discretizing the Newton's laws of motions is the "leap-frog" or Velocity Verlet method. The steps of the algorithm are given in (13) as follows:

1. Calculate the acceleration of each particle *i* at the current time *t* using:

$$F_i = m_i a_i = -\frac{dE}{d\boldsymbol{r_i}}$$

2. Update the velocity of particle *i* at the half time step $(t + \frac{1}{2}\Delta t)$ using:

$$v_i\left(t + \frac{1}{2}\Delta t\right) = v_i\left(t - \frac{1}{2}\Delta t\right) + a_i(t)\Delta t$$

3. Update to new position of particle *i* at $(t + \Delta t)$ using:

$$r_i(t + \Delta t) = r_i(t) + v_i\left(t + \frac{1}{2}\Delta t\right)\Delta t$$

Although these equations first appear to be quite simple, the calculation of the force in step 1 has become a major research issue for many areas of study over the last few decades.

The formula used to calculate the potential energy in step 1 for *N* atoms can be written as:

$$E = \sum_{bonds} \frac{a_i}{2}\left(l_i - l_{i0}\right)^2 + \sum_{angles} \frac{b_i}{2}(\theta_i - \theta_{i0})^2 + \sum_{torsions} \frac{V_n}{2}(1+\cos(n\omega - \gamma)) + \frac{1}{2}\sum_{i=1}^{N}\sum_{j\neq i}^{N} 4\varepsilon_{ij}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}\right] + \frac{1}{2}\sum_{i=1}^{N}\sum_{j\neq i}^{N}\frac{q_i q_j}{r_{ij}} \quad (1.1)$$

where $r_{ij} = \left|\boldsymbol{r_i} - \boldsymbol{r_j}\right| (9)$.

In this equation, the first four terms are used to calculate mainly short-ranged interactions, whereas the final term is for long-range. The parameters of Eq. (1.1), which define a force field model, have been calculated and compiled for use in energy minimization and dynamic simulation packages (14). Some examples of these are Amber (15; 16), CHARMM (17; 18), GROMACS (19; 20) and OPLS (21). The final term, which represents the Coulomb electrostatic potential, is by far the most difficult to calculate as it represents $O(N^2)$ operations. For small systems, this calculation is trivial. However, a typical protein or virus may consist of thousands or perhaps millions of atoms, and, therefore, the expense of this calculation is inhibitive even for today's computers. Many variations of MD methods have sprung from attempting to simplify this calculation. These can be divided into two major sub-categories: explicit methods, which use an atomistic representation of the simulated system and periodic boundary conditions, and implicit methods, which use implicit representation of the solvent with non-periodic boundary conditions.

1.1.3 Explicit Representation of the System

In explicit solvation methods, shown in Figure 4, all particles in the system, including solute and solvent, are treated in atomistic detail.

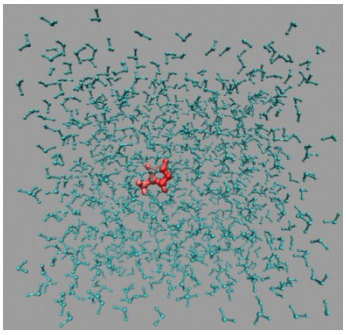

FIGURE 4: The explicit system where all atoms including the solute and solvent are treated in detail. The explicit method is considered to be more accurate but less efficient due to the large number of solvent molecules that must be included.

For the boundary conditions, these use periodic boundary conditions (PBC) where infinite copies of the central box, situated on a lattice, are used to simulate the solvent (1). Some examples of explicit algorithms include:

1. Simple cutoff methods (22) where the long range forces are truncated using artificial, non-bonded cutoffs (9).
2. Lattice-sum methods such as the original Ewald summation (23), the particle-mesh Ewald (PME) (24; 25), and the particle-particle particle-mesh Ewald (PPPM or P3M) (26; 27; 28). In these methods the Coulomb interaction is split into a screened short-range term, handled exactly by direct sum, plus a long-range, smoothly varying term, handled approximately in the reciprocal Fourier space (9), and

3. Fast Multipole Method (FMM) (29; 30) which divides the system into smaller cells and then calculates the Coulombic interactions exactly for particles within the same or neighboring sub-cells of the simulation cell and then uses multipole expansions to calculate the force for more distant particles (9). This method will be discussed in detail in Section (1.1.6)

Although PBCs makes it possible to find bulk properties for the simulated material in a finite system, they also introduce artifacts due to the interactions of atoms in different copies of the box. One method for improving the accuracy is to increase the box size for the system; however this will also increase the computational cost. These artifacts are not significant in homogeneous medium, such as pure solvent, with a large enough box size, but in the case of a solvated macromolecule these can be significant (1; 31) In their study of a polyalanine octapeptide in 2000, Weber et al showed that increasing the box size might actually destabilize the structure of a polypeptide (32)

1.1.4 Implicit Representation of the System

As in explicit systems, the atoms comprising the solute under study are treated in atomic detail in implicit systems (Figure 5). The solvent, however, is modeled as a dielectric continuum. The elimination of the solvent molecules reduces the number of degrees of freedom, which greatly improves computational costs. Either the Poisson equation or the Poisson-Boltzmann system equation is used to calculate the effect of the solvent on the solute (33; 34; 35; 36). Following is a detailed description of the PB electrostatic model for clarification as reviewed by Xu and Cai in SIAM Review (14).

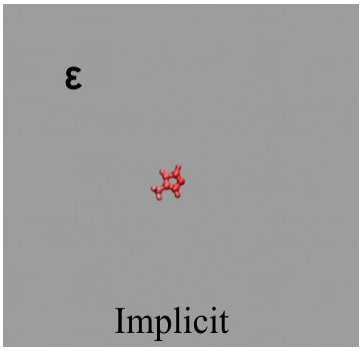

FIGURE 5: Image of the implicit system with the solute, shown in red, treated explicitly and the solute, shown as the gray area, treated as a dielectric continuum with dielectric constant ε. Image used with permission.

1.1.4.1 Poisson-Boltzmann Equation

For a classical continuum approach, the solute can be described as a low dielectric region with dielectric constant ε_i where ε_i typically ranges from $1{\sim}4$. In this region partial charges q_j located at atomic locations $\boldsymbol{r}_j$ have a charge density of:

$$\rho(\boldsymbol{r}) = \sum_{j=1}^{N} q_j \delta(\boldsymbol{r} - \boldsymbol{r_j}) \tag{1.2}$$

Typically the partial charges q_j from molecular mechanics force fields are used (37) but could also be calculated through quantum mechanics (38). These are considered to be point charges located at $\boldsymbol{r}_j$ which are taken as the nuclear centers of the atoms. As shown in Figure 6, the solute boundary Γ can be given by either the vdW surface, which is the sum of overlapping vdW spheres or the solvent accessible surface or SAS, which is generated by rolling a small sphere on the vdW surface (39). Outside of this surface is the solvent which has a higher dielectric constant. For instance, $\varepsilon_o \approx 80$ if the solvent is

water. This solvent is often an ionic liquid with ionic charges $n_i(\boldsymbol{r})$ for ions of type *i* (40).

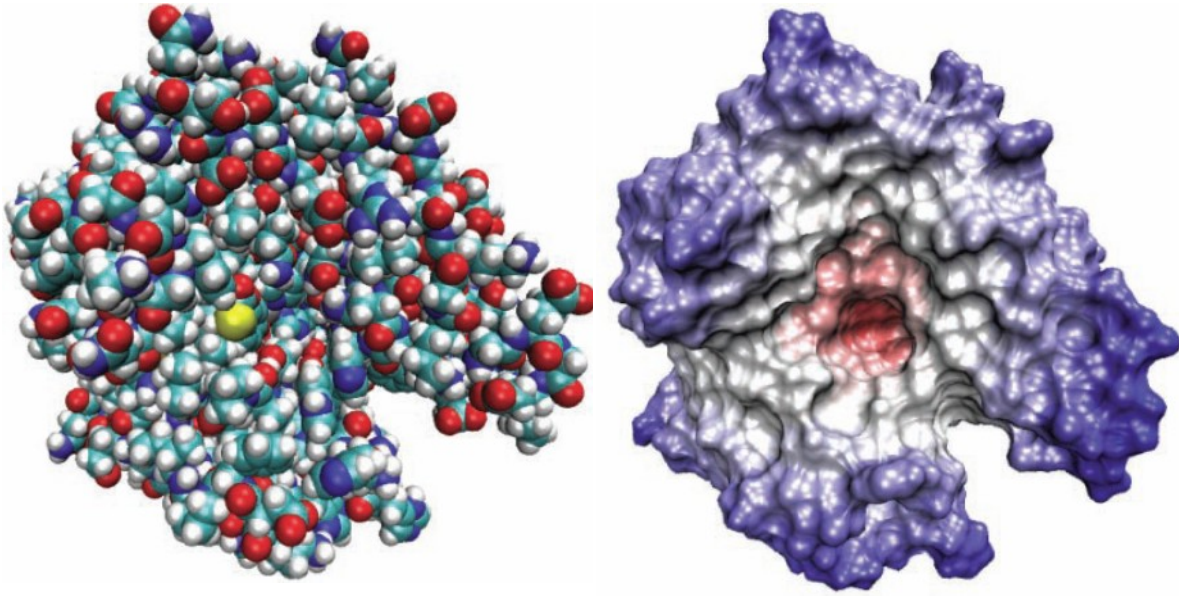

FIGURE 6: Molecular surfaces of the carbonic anhydrases-11 (39). (Left) The van der Waals (vdW) surface of the domain composed of the sum of overlapping vdW spheres. (Right) The solvent accessible surface (SAS) generated by rolling a small sphere on the vdW surface. In the macroscopic theory, the molecular domain inside the surface is given a low dielectric constant and the enclosed atoms are treated explicitly, while the exterior domain is treated as a homogeneous continuum medium with a higher dielectric constant. This image was created with VMD software. VMD is developed with NIH support by the Theoretical and Computational Biophysics group at the Beckman Institute, UIUC

In this case, the macroscopic potential $\Phi(\boldsymbol{r})$, due to the charges in the solute, the ionic charges in the solvent, and the polarization effects of the solvent from the solute charges, is governed by the Poisson equation:

$$-\nabla \cdot \epsilon(\boldsymbol{r})\nabla\Phi(\boldsymbol{r}) = 4\pi[\rho(\boldsymbol{r}) + n(\boldsymbol{r})] \tag{1.3}$$

where $n(\boldsymbol{r}) = \sum_i n_i(\boldsymbol{r})$.

There are two required boundary conditions for the interface on Γ to insure the continuity of the potentials and the normal displacements, namely:

$$\Phi(\boldsymbol{r}^-) = \Phi(\boldsymbol{r}^+), \epsilon_i \frac{\partial \Phi(\boldsymbol{r}^-)}{\partial \boldsymbol{n}} = \epsilon_o \frac{\partial \Phi(\boldsymbol{r}^+)}{\partial \boldsymbol{n}} \quad (1.4)$$

for $\boldsymbol{r} \in \Gamma$, where $\boldsymbol{r}^-$ and $\boldsymbol{r}^+$ are, respectively, the inner and outer limits at position $\boldsymbol{r}$ and $\boldsymbol{n}$ is the outward unit normal to the surface of the solute.

The solute charges will create an electric field that will affect the mobile ionic charges in the solvent. We assume that the ions will obey the Boltzmann distribution (41), under a potential of mean force (PMF) $\omega_i(\boldsymbol{r})$ (42), for the *i*th type of ions in the solvent,

$$n_i = n_i^0 e^{-\frac{\omega_i(\boldsymbol{r})}{k_B T}} \quad (1.5)$$

where n_i^0 is the number density of ions of type *i* in the bulk solvent in the absence of the solute, *T* is the temperature, and k_B is the Boltzmann constant.

The negative gradient of the PMF $\omega_i(\boldsymbol{r})$ gives the average force on a given ion of type *i* due to all other charges in the system. It is defined by a Gibbs average over all other ions and charge configurations, i.e., by a Boltzmann factor weighted integration over the positions of all other ions/charges in the phase space (42). According to the Debye–Hückel theory, the PMF on an ion of type *i,* $\omega_i(\boldsymbol{r})$*,* can then be approximated by multiplying the charge of the ion by the electrostatic potential Φ(**r**) of the solute-solvent system (43),

$$\omega_i(\boldsymbol{r}) = q_i \Phi(\boldsymbol{r}) \quad (1.6)$$

therefore the distribution of the *i*th type of ion is given by:

$$n_i = n_i^0 e^{-\frac{q_i \Phi(\mathbf{r})}{k_B T}} \quad (1.7)$$

We can now substitute Eq. (1.7) into Eq. (1.3) and we obtain a nonlinear PB equation for the electrostatic potential $\Phi(\mathbf{r})$ in the solute-solvent system:

$$-\nabla \cdot \epsilon(\boldsymbol{r}) \nabla \Phi(\boldsymbol{r}) = 4\pi\rho(\boldsymbol{r}) + 4\pi \sum_i q_i n_i^0 e^{-\frac{q_i \Phi(\mathbf{r})}{k_B T}} \quad (1.8)$$

Many methods exist for solving these equations including (1):

1. Finite Difference Method (44; 45; 46), where the derivatives in Poisson-Boltzmann equations are estimated using finite difference equations,
2. Finite Element Method (47; 48; 49), which divides the system into smaller elements where the equations can be approximated by simpler algebraic expressions, and
3. Boundary Element Method (50; 51; 52; 53) which is based on a surface integral representation of the linearized PB equation, which reduces the 3D problem into a 2D problem on the surface of the molecule.

Although a molecular system is simplified once the water molecules are integrated out, there are still several drawbacks to these methods. For instance, the arbitrary shape of macromolecules makes the solution of the three-dimensional PB equation computationally challenging (1). Another issue with this type of system is that the description of the sharp boundary between the low-dielectric medium of the solute and the high-dielectric medium of solvent might not be adequate to accurately describe

the physical system (1; 54). In 2007, Dai et al. showed that better accuracy could be attained by using a more gradual transition of the dielectric between the solute and solvent (55). It would appear that the optimal solution would be a combination of the explicit and implicit methods for the study of molecular systems.

1.1.5 Hybrid Representation of the System

Hybrid systems, which have attracted a great deal of attention in recent years (56; 57; 58) combine the explicit treatment with the implicit treatment by treating the solute and the first solvation layer explicitly and then treating the remaining solvent as a dielectric medium (Figure 7). A buffer region is added between these two areas where the particles are treated explicitly but the force is different from the other two areas. The potential in this area can be divided into two distinct parts. The first is the Coulomb potential (Φ_s) due to the interaction of the atoms in the explicit area with each other. The second is the potential due to the reaction field (Φ_{RF}) which is the polarization of the continuum solvent by the charges inside the explicit region. The total potential in the explicit region is therefore $\Phi = \Phi_s + \Phi_{RF}$. The main consideration for this type of system is the how to calculate Φ_{RF}.

One method, first applied in the context of solvation by Friedman (59), is to use image charges to approximate the reaction field. It was shown that for charges in a spherical cavity embedded in a dielectric medium with dielectric ε, one image charge could be used to approximate the reaction with an error less than $O(1/\varepsilon)$. Although this approximation is sufficient in systems with solvents with a high dielectric constant such as water, it may not be accurate enough in systems with low or moderate dielectric constant. We overcome this drawback of the one-image charge method by using multiple

image charges to approximate the reaction field and then using the Fast Multipole Method to calculate the Coulomb interactions including those from the reaction field (60; 40).

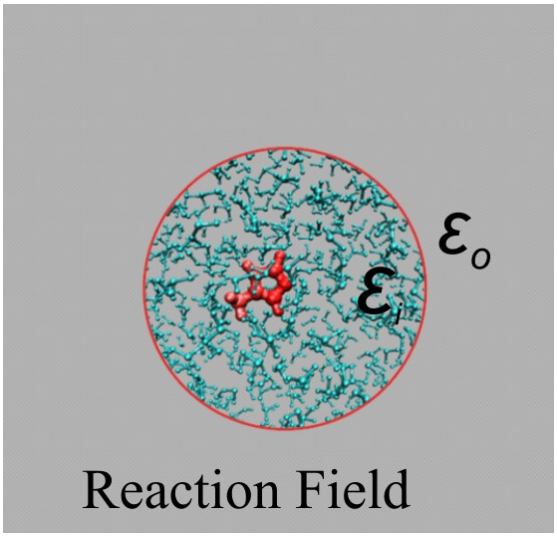

FIGURE 7: Image of a hybrid implicit/explicit system. All atoms inside the sphere are treated explicitly while the reaction field outside the sphere is considered a dielectric continuum with dielectric $\boldsymbol{\varepsilon_o}$.

1.1.6 Fast Multipole Method

The Fast Multipole Method (FMM), introduced by Greengard, et. al. in 1987, has become one of the most important algorithm developments in recent history and has many applications in both science and engineering such as the calculation of electrostatic potential and fast solvers for integral equations for electromagnetic scattering (29; 61; 62). The FMM relies on the analytical property of Green's functions for the Laplace operator for calculating the electrostatic potential. Therefore, the FMM has only been used for the cases of homogeneous media of the situations where Green's functions can be obtained by methods of images for layered media (62). Our method of using multiple

discrete images allows us to extend the FMM to the case of a dielectric sphere with source point charges inside the sphere. One important example of this type of system would be the calculation of the electric potential for a protein inside a spherical cavity with a dissimilar dielectric constant than the surrounding medium.

Our approach can be traced back to a result obtained over 100 years ago by C. Neumann in 1883 (63) and more recently by Lindell in 1994 (64; 65; 66), which extended the Kelvin image (67) for a conducting sphere to the case of a dielectric sphere. In the case of a protein inside a dielectric sphere, both the source charge and the field point are inside the sphere. Therefore, in addition to an image point charge at the Kelvin image inverse point, there is an image line charge along a ray extending from the inverse point to the center of the sphere. Lindell has provided the power law distribution for the line charge density along the ray. In the following sections, we will describe the development of the discrete image charges.

1.1.7 Theoretical Background of Multiple Image Charges

1.1.7.1 Point Charge in a Homogenous Medium

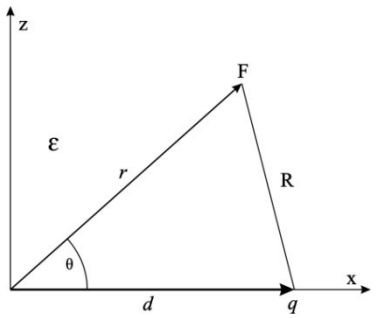

FIGURE 8: A point charge (q) embedded in a homogenous medium ε.

Before covering the details for calculating the reaction field due to multiple discrete image charges, we will consider the classical electrostatic theory to find the potential of a point charge inside a dielectric sphere. First consider a dielectric sphere of radius *a* with dielectric constant ε_i, centered at the origin and embedded in an infinite homogeneous medium of dielectric constant ε_o (60). In general terms, given a point charge q located on z at $z=d$, the electric potential at a field point $F(\boldsymbol{r})$ generated by q is (Figure 8):

$$V(\boldsymbol{r}) = \frac{q}{4\pi\varepsilon_i R} = \frac{q}{4\pi\varepsilon_i} \cdot \frac{1}{\sqrt{r^2 + d^2 - 2dr\cos\theta}} \tag{1.9}$$

The Legendre polynomial expansion of the inverse distance is:

$$\frac{1}{|\boldsymbol{x} - \boldsymbol{x}'|} = \frac{1}{\sqrt{r^2 + d^2 - 2dr\cos\theta}} = \sum_{n=0}^{\infty} \frac{d^n}{r^{n+1}} P_n \cos\theta \tag{1.10}$$

So,

$$V(\boldsymbol{r}) = \frac{q}{4\pi\varepsilon_i R} = \frac{q}{4\pi\varepsilon_i} \sum_{n=0}^{\infty} \frac{d^n}{r^{n+1}} P_n \cos\theta \tag{1.11}$$

Due to the polarization of the dielectric, which must be finite at $r = 0$, we must add to this a reaction potential (40). Therefore, the total potential would be:

$$V(r, \theta) = \frac{q}{4\pi\varepsilon_i R} + \sum_{n=0}^{\infty} B_n r^n P_n \cos\theta \tag{1.12}$$

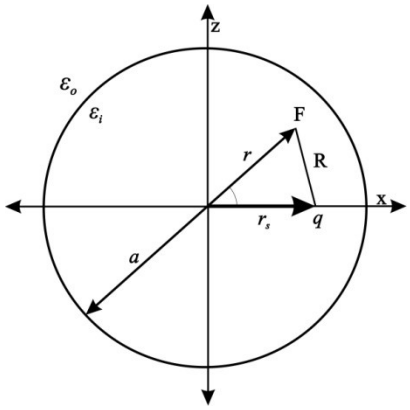

FIGURE 9: A point charge inside a dielectric sphere with dielectric $\boldsymbol{\varepsilon_i}$ embedded in a dielectric continuum of dielectric $\boldsymbol{\varepsilon_o}$.

As all of our point charges of the MD simulation will be inside a sphere defining the explicit region of the system, we consider a point charge q located on the x-axis inside a sphere at a distance r_s from the center of the sphere (40). So, depending on whether $r \geq r_s$ or $r \leq r_s$, we get (Figure 9):

$$\frac{q}{4\pi\varepsilon_i R} = \begin{cases} \dfrac{q}{4\pi\varepsilon_i r}\displaystyle\sum_{n=0}^{\infty}\left(\frac{r_s}{r}\right)^n P_n(\cos\theta), & r_s \leq r \leq a \\ \dfrac{q}{4\pi\varepsilon_i r_s}\displaystyle\sum_{n=0}^{\infty}\left(\frac{r}{r_s}\right)^n P_n(\cos\theta), & 0 \leq r \leq r_s \end{cases} . \tag{1.13}$$

Then the potentials inside the sphere take the form:

$$V(r,\theta) = \begin{cases} \displaystyle\sum_{n=0}^{\infty}\left[\frac{q}{4\pi\varepsilon_i r}\left(\frac{r_s}{r}\right)^n + B_n r^n\right]P_n(\cos\theta), & r_s \leq r \leq a \\ \displaystyle\sum_{n=0}^{\infty}\left[\frac{q}{4\pi\varepsilon_i r_s}\left(\frac{r}{r_s}\right)^n + B_n r^n\right]P_n(\cos\theta), & 0 \leq r \leq r_s \end{cases} \tag{1.14}$$

The expansion coefficient B_n can be determined using the boundary conditions that the potentials at the boundary of the sphere are equal and also the flux normal to the boundary is the same at either side of the boundary:

$$V\left(a^{+},\ \theta\right)=V\left(a^{-},\ \theta\right),$$
$$\varepsilon_o \frac{\delta V(r,\theta)}{\delta r}\Big|_{r=a^{+}} = \varepsilon_i \frac{\delta V(r,\theta)}{\delta r}\Big|_{r=a^{-}} \tag{1.15}$$

Using the orthogonality of the Legendre Polynomials, we get:

$$B_n = \frac{q}{4\pi\varepsilon_i}\cdot\frac{r_s^n}{a^{2n+1}}\cdot\left(2+\frac{2\gamma}{1-\gamma+2n}\right) \tag{1.16}$$

Where $n \geq 0$ and $\gamma = \varepsilon_i - \varepsilon_o/\varepsilon_i + \varepsilon_o$.

1.1.7.2 Image Charges and Image Line Charges

Carl Neumann derived the mathematical formulas for finding the image line charges for a point charge inside a dielectric sphere (63)in 1883(63). Over the last two decades, these results have been investigated independently by both Lindell (64; 65; 66) and Norris (68).

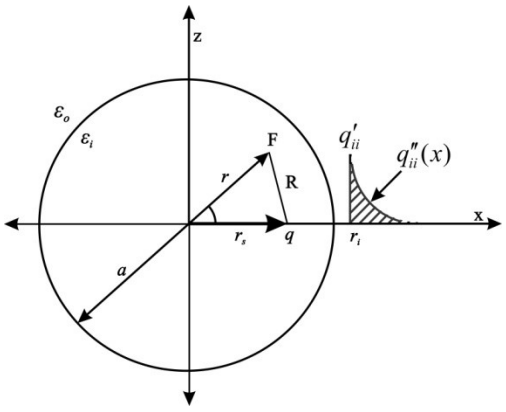

FIGURE 10: Discretization of the line image charge $\boldsymbol{q''_{ii}(x)}$.

To calculate the potential inside the sphere due to polarization, we substitute B_n from Eq. (1.16) into the last part of Eq. (1.14) to get:

$$\sum_{n=0}^{\infty} B_n r^n P_n(\cos\theta) = \sum_{n=0}^{\infty} \frac{q}{4\pi\varepsilon_i} \cdot \frac{r_s^n}{a^{2n+1}} \cdot \gamma \left(1 + \frac{1+\gamma}{1-\gamma+2n}\right) r^n P_n(\cos\theta)$$

$$= \frac{\gamma q}{4\pi\varepsilon_i r_i} \frac{a}{r_s} \sum_{n=0}^{\infty} \left(\frac{r}{r_i}\right)^n P_n(\cos\theta)$$

$$+ \frac{q}{4\pi\varepsilon_i} \frac{\gamma(1+\gamma)}{2} \sum_{n=0}^{\infty} \frac{r_s^n}{a^{2n+1}} \cdot \frac{2}{1-\gamma+2n} \cdot r^n P_n(\cos\theta) \tag{1.17}$$

$$= S_1 + S_2$$

Let $d = r_i$, $\varepsilon = \varepsilon_i$, then $S_1 = V(\boldsymbol{r})$ from eq. (1.11) for $q_{ii}^{'} = \frac{\gamma a q}{r_s}$.

Now,

$$S_2 = \frac{q}{4\pi\varepsilon_i} \frac{\gamma(1+\gamma)}{2} \sum_{n=0}^{\infty} \frac{r_s^n}{a^{2n+1}} \cdot \frac{2}{1-\gamma+2n} \cdot r^n P_n(\cos\theta) \tag{1.18}$$

Substitute in $r_i = \frac{a^2}{r_s}$, then:

$$\frac{r_s^n}{a^{2n+1}} = \left(\frac{a^{-\gamma}}{r_s^{\frac{1-\gamma}{2}}}\right)\left(\frac{r_s}{a^2}\right)^{\frac{1-\gamma}{2}+n} = \left(\frac{a^{-\gamma}}{r_s^{\frac{1-\gamma}{2}}}\right)\left(\frac{1}{r_i}\right)^{\frac{1-\gamma}{2}+n} \tag{1.19}$$

Using the following integral

$$\int_{r_i}^{\infty} \frac{1}{x^{\frac{1-\gamma}{2}+n+1}} dx = \frac{2}{1-\gamma+2n} \cdot \left(\frac{1}{r_i}\right)^{\frac{1-\gamma}{2}+n}$$

we get:

$$S_2 = \frac{q}{4\pi\varepsilon_i}\frac{\gamma(1+\gamma)}{2}\sum_{n=0}^{\infty}\left[\frac{a^{-\gamma}}{r_s^{\frac{1-\gamma}{2}}}\int_{r_i}^{\infty}\frac{1}{x^{\frac{1-\gamma}{2}+n+1}}dx\right]r^n P_n(\cos\theta) \tag{1.20}$$

Then, after moving the integral to the outside and substituting for $r_i = \frac{a^2}{r_s}$ again:

$$S_2 = \int_{r_i}^{\infty}\left[\frac{q}{4\pi\varepsilon_i x}\frac{1}{a}\frac{\gamma(1+\gamma)}{2}\left(\frac{x}{r_i}\right)^{-\frac{1-\gamma}{2}}\sum_{n=0}^{\infty}\left(\frac{r}{x}\right)^n P_n(\cos\theta)\right]dx \tag{1.21}$$

The inside of the above integral is Eq. (1.11) with $d = x$ and $\epsilon = \varepsilon_i$ and represents the potential generated by the charge $q_{ii}^{''}(x)$ at $\boldsymbol{x}$ **(**Figure 10**)**, where $q_{ii}^{''}(x) = \frac{q}{a}\frac{\gamma(1+\gamma)}{2}\left(\frac{x}{r_i}\right)^{-\frac{1-\gamma}{2}}, r_i \le x.$

1.1.7.3 Multiple Discrete Images

The total potential at a field point F(**r**) inside a sphere, due to an internal point charge q, consists of three components: the potential $V_s(\boldsymbol{r};q)$ from the original source point charge q at $\boldsymbol{r}_s$, the potential $V_{ip}(\boldsymbol{r};q)$ from the image point charge $q_{ii}^{'}$ at the inverse point $\boldsymbol{r}_i$, and the potential $V_{il}(\boldsymbol{r};q)$ which is due to the distributed image line charge $q_{ii}^{''}(x)$, which we developed previously (40). So we have:

$$\begin{aligned} V(\boldsymbol{r};q) &= V_s(\boldsymbol{r};q) + V_{ip}(\boldsymbol{r};q) + V_{il}(\boldsymbol{r};q) \\ &= \frac{q}{4\pi\varepsilon_i\cdot|\boldsymbol{r}-\boldsymbol{r}_s|} + \frac{q_{ii}'}{4\pi\varepsilon_i\cdot|\boldsymbol{r}-\boldsymbol{r}_i|} + \int_{r_i}^{\infty}\frac{q_{ii}''(x)}{4\pi\varepsilon_i\cdot|\boldsymbol{r}-\boldsymbol{x}|}dx \end{aligned} \tag{1.22}$$

where $r_i = \frac{a^2}{r_s}, q_{ii}' = \frac{\gamma a q}{r_s}, q_{ii}''(x) = \frac{q}{a}\frac{\gamma(1+\gamma)}{2}\left(\frac{x}{r_i}\right)^{-\frac{1-\gamma}{2}}, r_i \le x.$

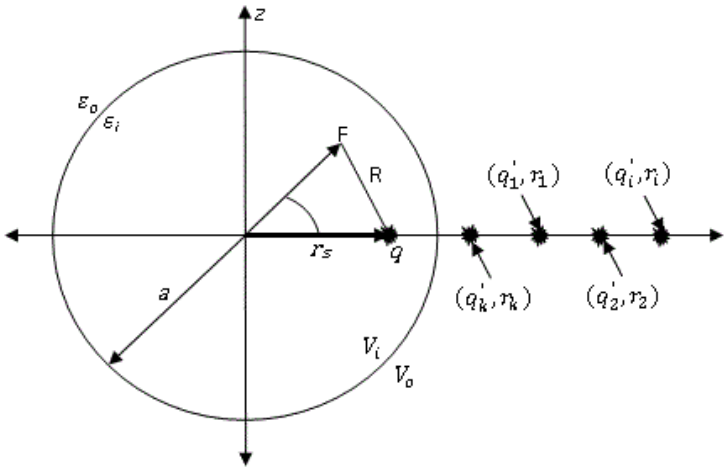

FIGURE 11: Visualization of the multiple image charges created from the discretization of the line image charge.

To approximate the potential due to the image line charge $q_{ii}''(x)$ by a small number of discrete points we start with the line integral (Figure 11):

$$I = \int_{r_i}^{\infty} \frac{1}{|\boldsymbol{r} - \boldsymbol{x}|} \left(\frac{x}{r_i}\right)^{-\frac{1-\gamma}{2}} dx \tag{1.23}$$

First, introduce a change of variable $\frac{r_i}{x} = \left(\frac{1-s}{2}\right)^{\tau}$ with $\tau > 0$. Then,

$$I = \tau \cdot 2^{\frac{\gamma-1}{2}\tau} \int_{-1}^{1} (1-s)^{\alpha} \cdot h(\boldsymbol{r}, s; \tau) ds \tag{1.24}$$

where $\alpha = \frac{(1-\gamma)\tau}{2} - 1$ and $h(\boldsymbol{r}, s; \tau) = \frac{2^{\tau} r_i}{|(1-s)^{\tau}\boldsymbol{r} - 2^{\tau}\boldsymbol{r}_i|}$.

Next, employ a numerical quadrature to approximate the integral I. The value $s = -1$ corresponds to the Kelvin image location $x = r_i$. Also, $\alpha > -1$ since $-1 < \gamma < 1$ and $\tau > 0$. Therefore, we can use Gauss-Radau quadrature based on Jacobi polynomials.

The Jacobi polynomials $P_n^{\alpha,\beta}(s)$ on [-1,1] are orthogonal polynomials under the Jacobi weight $\omega(s) = (1-s)^\alpha(1+s)^\beta$, i.e.

$$\int_{-1}^{1}(1-s)^\alpha(1+s)^\beta\, P_n^{\alpha,\beta}(s)ds = \delta_{jk}, \alpha > -1, \beta > -1 \tag{1.25}$$

Let $s_m, \omega_m,\ m = 0,1,2,\cdots,M$ be the Jacobi-Gauss-Radau points and weights on [-1,1] with $s_0 = -1$, $\alpha = \frac{(1-\gamma)\tau}{2} - 1,\ \beta = 0$. Then, the numerical quadrature for approximating integral I is:

$$I \approx \tau \cdot 2^{\frac{\gamma-1}{2}\tau} \sum_{m=0}^{M} \omega_m h(\boldsymbol{r}, s_m; \tau) \tag{1.26}$$

Then,

$$V_{il}(\boldsymbol{r}; q) = \int_{r_i}^{\infty} \frac{q_{ii}''(x)}{4\pi\varepsilon_i \cdot |\boldsymbol{r} - \boldsymbol{x}|}\, dx \approx \sum_{m=0}^{M} \frac{q_m^{ii}}{4\pi\varepsilon_i \cdot \left|\boldsymbol{r} - \boldsymbol{x}_m^{ii}\right|} \tag{1.27}$$

where $m = 0,1,2,\cdots,M;\ q_m^{ii} = 2^{\frac{\gamma-1}{2}\tau-1}\gamma(1+\gamma)\tau\omega_m q \cdot \frac{x_m^{ii}}{a}$, and $x_m^{ii} = r_i \cdot \left(\frac{2}{1-s_m}\right)^\tau$.

Note that $x_0^{ii} = r_i$. Therefore we can combine together the point image charge q_{ii}' with the first discrete point charge q_0^{ii} and

$$\begin{aligned} V(\boldsymbol{r}; q) &= V_s(\boldsymbol{r}; q) + V_{ip}(\boldsymbol{r}; q) + V_{il}(\boldsymbol{r}; q) \\ &= \frac{q}{4\pi\varepsilon_i \cdot |\boldsymbol{r} - \boldsymbol{r}_s|} + \frac{q_{ii}'}{4\pi\varepsilon_i \cdot |\boldsymbol{r} - \boldsymbol{r}_i|} + \int_{r_i}^{\infty} \frac{q_{ii}''(x)}{4\pi\varepsilon_i \cdot |\boldsymbol{r} - \boldsymbol{x}|}\, dx \end{aligned} \tag{1.28}$$

becomes:

$$V(\boldsymbol{r};q)=\frac{q}{4\pi\varepsilon_i\cdot|\boldsymbol{r}-\boldsymbol{r}_s|}+\frac{q'_{ii}+q_0^{ii}}{4\pi\varepsilon_i\cdot|\boldsymbol{r}-\boldsymbol{r}_i|}+\sum_{m=1}^{M}\frac{q_m^{ii}}{4\pi\varepsilon_i\cdot|\boldsymbol{r}-\boldsymbol{x}_m^{ii}|} \tag{1.29}$$

1.1.7.4 Images for Ionic Solvent

To this end, theoretically let us consider a spherical volume Γ with dielectric permittivity ε_i imbedded in an infinite solvent of dielectric permittivity ε_o, the total potential $\Phi(\boldsymbol{r})$ in this setting satisfies the following version of the Poisson-Boltzmann equation:

$$\begin{aligned}\varepsilon_i\nabla^2\Phi(\boldsymbol{r})&=-\rho_{in}(\boldsymbol{r}), \qquad \boldsymbol{r}\in V_{in}\\ [\nabla^2-\lambda^2]\Phi(\boldsymbol{r})&=0, \qquad \boldsymbol{r}\in V_{out}\end{aligned} \tag{1.30}$$

where the $\rho_{in}(\boldsymbol{r})=\sum_i q_i\delta(\boldsymbol{r}-\boldsymbol{r_i})$ is the charge distribution inside V_{in} which contains all explicit charges of the solute and solvent molecules and λ is the inverse Debye-Hückel screening length.

For the boundary conditions, the continuity of the tangential component of the electric field and the normal component of the displacement field require that:

$$\begin{aligned}\Phi_{in}&=\Phi_{out}\\ \varepsilon_i\frac{\delta\Phi_{in}}{\delta\boldsymbol{n}}&=\varepsilon_o\frac{\delta\Phi_{out}}{\delta\boldsymbol{n}}\end{aligned} \tag{1.31}$$

where **n** is the outward normal and Φ_{in} and Φ_{out} are the potentials inside and outside of the cavity.

Assume that a single point charge q is located at position $\boldsymbol{r}_s$ inside the spherical cavity. We can write the solution to the above PB equation in terms of the primary field, Φ_s, which is the field due to source charge, and the reaction field, Φ_{RF}, which is the potential created by the source-induced polarization of the solvent. Then the total potential can be written as $\Phi = \Phi_s + \Phi_{RF}$. For a spherical cavity with radius a, Φ_{RF} inside the cavity is given by the Kirkwood expansion (1). In the pure solvent case where $\lambda = 0$, we will have (as discussed in Eq. (1.17):

$$\phi_{RF}(r,\theta) = \frac{q}{4\pi\varepsilon_i a}\sum_{n=0}^{\infty}\frac{(\varepsilon_i - \varepsilon_o)(n+1)}{\varepsilon_i n + \varepsilon_o(n+1)}\left(\frac{r r_s}{a^2}\right)^n P_n(\cos\theta) \tag{1.32}$$

where P_n are the Legendre polynomials as in Section 1.1.7.1 and θ is the angle formed by vectors **r** and $\mathbf{r}_s$.

If we let $r_k = \frac{a^2}{r_s}$ be the location of the Kelvin image charge and $q_k = \gamma\frac{a}{r_s}q$ be the charge, then after some algebraic manipulation Eq. (1.32), the reaction field, can be approximated by (see Section 1.1.7 for details on the derivation):

$$\phi_{RF}(\boldsymbol{r}) = \frac{q_k}{4\pi\varepsilon_i r_k}\sum_{n=0}^{\infty}\left(\frac{r}{r_k}\right)^n P_n(\cos\theta) + \frac{\delta q}{4\pi\varepsilon_i a}\sum_{n=0}^{\infty}\frac{1}{n+\sigma}\left(\frac{r}{r_k}\right)^n P_n(\cos\theta) \tag{1.33}$$

In this case $\delta = \varepsilon_i\frac{\varepsilon_i - \varepsilon_o}{(\varepsilon_i+\varepsilon_o)^2}$ and $\sigma = \frac{\varepsilon_o}{(\varepsilon_i+\varepsilon_o)}$.

The first part of this equation is the potential of the Kelvin image charge.

After substituting in the integral identity $\frac{1}{n+\sigma} = r_K^{n+\sigma} \int_{r_K}^{\infty} \frac{1}{x^{n+\sigma+1}} dx$ the second part of the equation becomes:

$$\frac{\delta q}{4\pi\varepsilon_i a} \sum_{n=0}^{\infty} \frac{1}{n+\sigma} \left(\frac{r}{r_k}\right)^n P_n(\cos\theta) = \int_{r_k}^{\infty} \left[\frac{q'(x)}{4\pi\varepsilon_i x} \sum_{n=0}^{\infty} \left(\frac{r}{x}\right)^n P_n(\cos\theta) \right] dx \tag{1.34}$$

Which is the potential $V_{il}(\boldsymbol{r}; q)$ from Eq. (1.28) created by the line image charge $q'(x) = \frac{\delta q}{a} \left(\frac{x}{r_k}\right)^{-\sigma}, r_k \leq x.$

Similar to Eq. (1.28), we can find the multiple discrete image charges for the reaction field from an ionic solvent.

CHAPTER 2: IMAGE-CHARGE SOLVATION MODEL (ICSM)

2.1.1 The Setup of ICSM

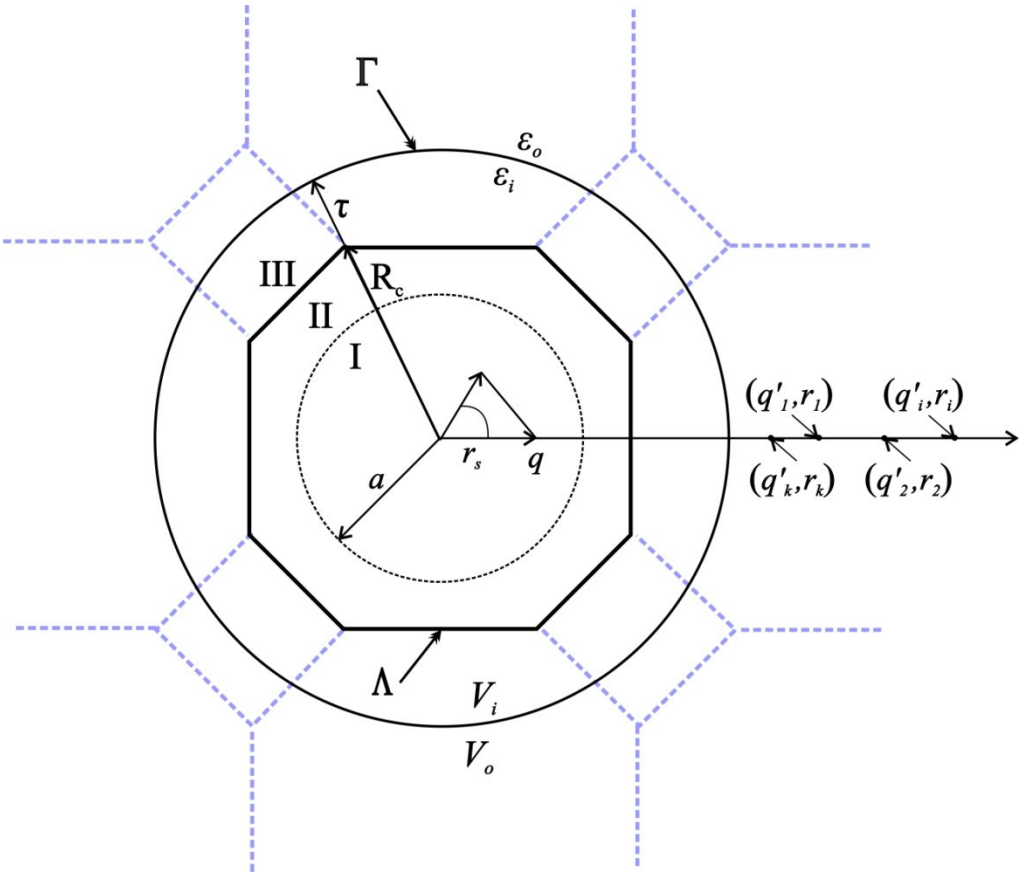

FIGURE 12: A schematic illustration of the ICSM. Note that the productive region, Region I, can accommodate a solute molecule of maximum diameter of $\boldsymbol{d = L(\sqrt{3} - \sqrt{5}/2) - 2\tau}$ where L is the size of the cube from which the regular TO simulation box Λ is built and τ is the thickness of the buffer zone. Particles in Region I are not periodically imaged. Region II contains particles which have periodic images in Region III. Region III, the so-called buffer zone, contains the nearest periodic images of the particles in Region II. The solvent outside the spherical cavity Γ is modeled as a dielectric continuum. Given a source charge q located at position $\boldsymbol{r_s}$ inside this spherical cavity Γ, the reaction field $\boldsymbol{\phi_{RF}(r)}$ at position r inside the cavity Γ due to the polarization of the implicit solvent can be approximated by the potential created by the image charges, $\boldsymbol{q'_k, q'_i, i \geq 2}$, located at positions $\boldsymbol{r_k, r_i, i \geq 2}$, respectively **(1)**.

The Image-Charge Solvation Model (ICSM) was first introduced in Ref. (1) as a new method for calculating the above electrostatic interactions. Central to the ICSM is, for a point charge inside a spherical cavity, how to find the image charges outside a central cavity that generate the reaction field inside the cavity, which has been shown above to be approximated by multiple image charges.

The setup of the model itself can be schematically illustrated by Figure 12. A regular truncated octahedron (TO) is employed as the main simulation box. The central part of the box, marked as Region I, contains the solute molecule under study, while the remaining part of the box (including Region II and the area in Region I not occupied by the solute molecule) is filled with solvent. The solute and the solvent molecules inside the main simulation box are to be treated in atomic detail. Region III represents a buffer zone whose purpose is to eliminate or reduce the surface effects that would otherwise be induced by the sharp boundary between explicit and implicit solvents.

The solvent molecules in Region III are just periodic images of the solvent particles in Region II, defined as in the usual periodic boundary condition with respect to the TO simulation box for long range electrostatic calculation. And the solvent outside the spherical cavity is modeled as a dielectric continuum whose effect on the solute is treated through reaction field corrections.

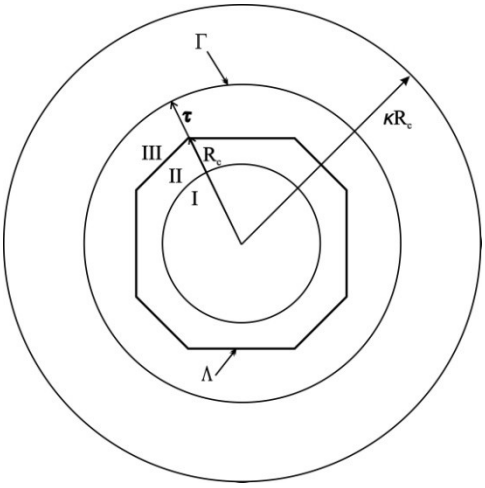

FIGURE 13: A schematic showing the division of the system into an outside system which is solved using LE and an inside system which is solved using FMM.

Note that the solute particles in Region I are not periodically imaged and only the solvent particles in Region II could have periodic images in Region III, leading to a most important feature of the ICSM that a solute may be solvated without suffering any artificial electrostatic solute-solute interactions. For this reason, Region I is also called the productive region in the ICSM.

After the calculation of the periodic images, all image charges are determined using

$$q_m^{ii} = 2^{\frac{\gamma-1}{2}\tau-1}\gamma(1+\gamma)\tau\omega_m q \cdot \frac{x_m^{ii}}{a}, \text{ and } x_m^{ii} = r_i \cdot \left(\frac{2}{1-s_m}\right)^{\tau} \text{ from Eq. (1.27).}$$

As initially proposed in reference (41), we could now complete the evaluation of these forces with a single FMM run using all of the charges in Λ, the periodic images in region III and the image charges outside the sphere Γ and assuming that all charges are located in a homogeneous medium of dielectric permittivity ϵ_i. However, the image

charges are not uniformly distributed, especially the image charges of the original charges which are close to the center of Λ. Therefore, we introduce an additional step which divides the system into two separate systems using a reference sphere S_r of radius κR_c as shown in Figure 13. The field with the TO box Λ due to charges inside S_r is calculated using an adaptive FMM and the field due to charges outside this sphere is calculated using a local expansion as described in reference (43).

Specifically, for any point outside the reference sphere at location $(\rho_l, \alpha_l, \beta_l)$, $L = 1,2,\cdots,L$ with charge $\hat{q}_l$, $L = 1,2,\cdots,L$ the potential on a charge inside the sphere located at $\boldsymbol{r} = (r, \theta, \varphi)$ can be described by the local expansion:

$$\Phi(\boldsymbol{r}) \approx \sum_{j=0}^{p} \sum_{k=-j}^{j} L_j^k \, Y_J^k(\theta, \varphi) r^j, \tag{2.1}$$

where $Y_J^k(\theta, \varphi)$ are the spherical harmonics and L_j^k are the local expansion coefficients given by:

$$L_j^k = \sum_{l=1}^{L} \hat{q}_l \cdot \frac{Y_j^{-k}(\alpha_l, \beta_l)}{\rho_l^{j+1}} \tag{2.2}$$

The force $\boldsymbol{f}(\boldsymbol{r}) = (f_x(\boldsymbol{r}), f_y(\boldsymbol{r}), f_z(\boldsymbol{r}))$ can then be calculated by (1):

$$f_x(\boldsymbol{r}) = -q\frac{\partial}{\partial x}\Phi(\boldsymbol{r}) = -q\mathrm{Re}(H_2 - H_3) \tag{2.3}$$

$$f_y(\boldsymbol{r}) = -q\frac{\partial}{\partial y}\Phi(\boldsymbol{r}) = -q\mathrm{Im}(H_2 + H_3) \tag{2.4}$$

$$f_z(\boldsymbol{r}) = -q\frac{\partial}{\partial z}\Phi(\boldsymbol{r}) = -q\big(H_0 - 2\,\mathrm{Re}(H_1)\big) \tag{2.5}$$

where $\mathrm{Re}(\cdots)$ and $\mathrm{Im}(\cdots)$ represent the real and imaginary parts of a complex number and

$$H_0 = \sum_{j=1}^{p} j\, L_j^0 P_{j-1}(\cos\theta) r^{j-1},$$

$$H_1 = \sum_{j=1}^{p}\sum_{k=1}^{j-1}(j+k) C_j^k L_j^k e^{ik\varphi} P_{j-1}^k(\cos\theta) r^{j-1},$$

$$H_2 = \sum_{j=1}^{p} L_j^0 e^{i\varphi} P_{j-1}^1(\cos\theta) r^{j-1} + \sum_{j=1}^{p}\sum_{k=1}^{j} C_j^k L_j^k e^{i(k+1)\varphi} P_{j-1}^{k+1}(\cos\theta) r^{j-1},$$

$$H_3 = \sum_{j=1}^{p} C_j^{-1} L_j^1 e^{i\varphi} P_{j-1}(\cos\theta) r^{j-1} + \sum_{j=1}^{p}\sum_{k=2}^{j} B_j^k C_j^k L_j^k e^{i(k-1)\varphi} P_{j-1}^{k-1}(\cos\theta) r^{j-1},$$

and

$$B_j^k = (j+k)(j+k-1), C_j^k = \sqrt{\frac{(j-k)!}{(j+k)!}}$$

2.1.2 Integration of ICSM into TINKER

As pointed out earlier, the ICSM has been integrated into the molecular dynamics portion of the Tinker package to provide a new order N method to calculate charge-charge electrostatic interactions (69). How the ICSM is integrated into TINKER is schematically shown in Figure 14. The changes needed to the original Tinker source files are minimal, and will be described in detail later. As a matter of fact, the only significant change required is to add in the Tinker source file *echarge1.f* a call to the ICSM subroutine *imgmethod()* and another subsequent call to the ICSM subroutine *fmmle()*. In this sense, the (modified) Tinker file *echarge1.f* could be regarded as the interface between Tinker and the ICSM. In simulations, the call for the ICSM is controlled by the presence of the keyword *imgmethod* in the keyword parameter file, and when the keyword *imgmethod* is present, the logical variable *use_images* is set to *.true.* in the (modified) TINKER subroutine *prmkey()*. It should also be mentioned that a new subroutine *echarge1h()* is added into *echarge1.f* for calculating electrostatic interactions by the ICSM without using the FMM. And whether or not to use the FMM in the ICSM is determined by whether another keyword *fmmle* is also present in the keyword parameter file, the logical variable *use_fmmle* is set to *.true.* accordingly.

If the keyword *imgmethod* is present in the keyword parameter file, the ICSM subroutine *imgchg()* is first called by the TINKER subroutine *mechanic()* to initialize or calculate the variables unique to the ICSM, including the Gauss–Radau quadrature points and weights $\{s_m, \omega_m\}_{m=1}^{N_i}$. Then later the ICSM subroutine *imgmethod()* is called by the TINKER subroutine *echarge1()* to (1) locate periodic images in Region III for those solvent particles in Region II of the simulation box Λ, and (2) find image charges of

reaction field for all charges inside the spherical cavity Γ . To this end, the atoms in each molecule inside the simulation box Λ are first checked to see if they are inside Region II. If any part of the molecule is inside Region II, the positions of the periodic images of its atoms are calculated and stored in an image-position array, and the charges of these periodic images are stored in an image-charge array accordingly. Remember that only those periodic images that are indeed located inside the buffer zone are kept and contribute to the reaction field corrections. Once all periodic images are found and stored, the locations as well as the magnitudes of image charges are then calculated and stored in corresponding arrays for all real and periodic image charges inside the spherical cavity Γ if $N_i \geq 1$. Lastly, the electrostatic forces exerted on all real ‘‘target’’ charges inside the simulation box Λ by all ‘‘source’’ charges, including real charges inside Λ, periodic images in Region III, and image charges of reaction fields outside the spherical cavity Γ, are calculated either by the pairwise method in *echarge1h()* or by the FMM, depending on whether the keyword *fmmle* is present in the keyword parameter file. Once this last step is done, the electrostatic forces are then added to the values in TINKER's force arrays and control is returned to TINKER by the ICSM.

If the keyword *fmmle* is present in the keyword parameter file, the ICSM subroutine *fmmle ()* is called to calculate the above electrostatic forces by the FMM together with the local expansion procedure described in Section 2.1.1. First, all ‘‘source’’ charges are split into two groups based on where they are located by calling the ICSM subroutine *twoGrp()*. If a charge is located inside the cut-off reference sphere S_r , it is put in the ‘‘inside’’ group; otherwise it is put in the ‘‘outside’’ group. Next, the electrostatic forces exerted on all ‘‘targets’’ inside the simulation box Λ by those

‘‘source’’ charges from the ‘‘inside’’ group are evaluated using the FMM. Then the forces exerted on the same ‘‘targets’’ by those ‘‘source’’ charges from the ‘‘outside’’ group are calculated using the local expansion (LE). The forces calculated by the FMM and the local expansion are then added together to obtain the final total electrostatic forces. Details concerning the installation and use of tinker with the ICSM are included in Appendix A: Installation Instructions**.**

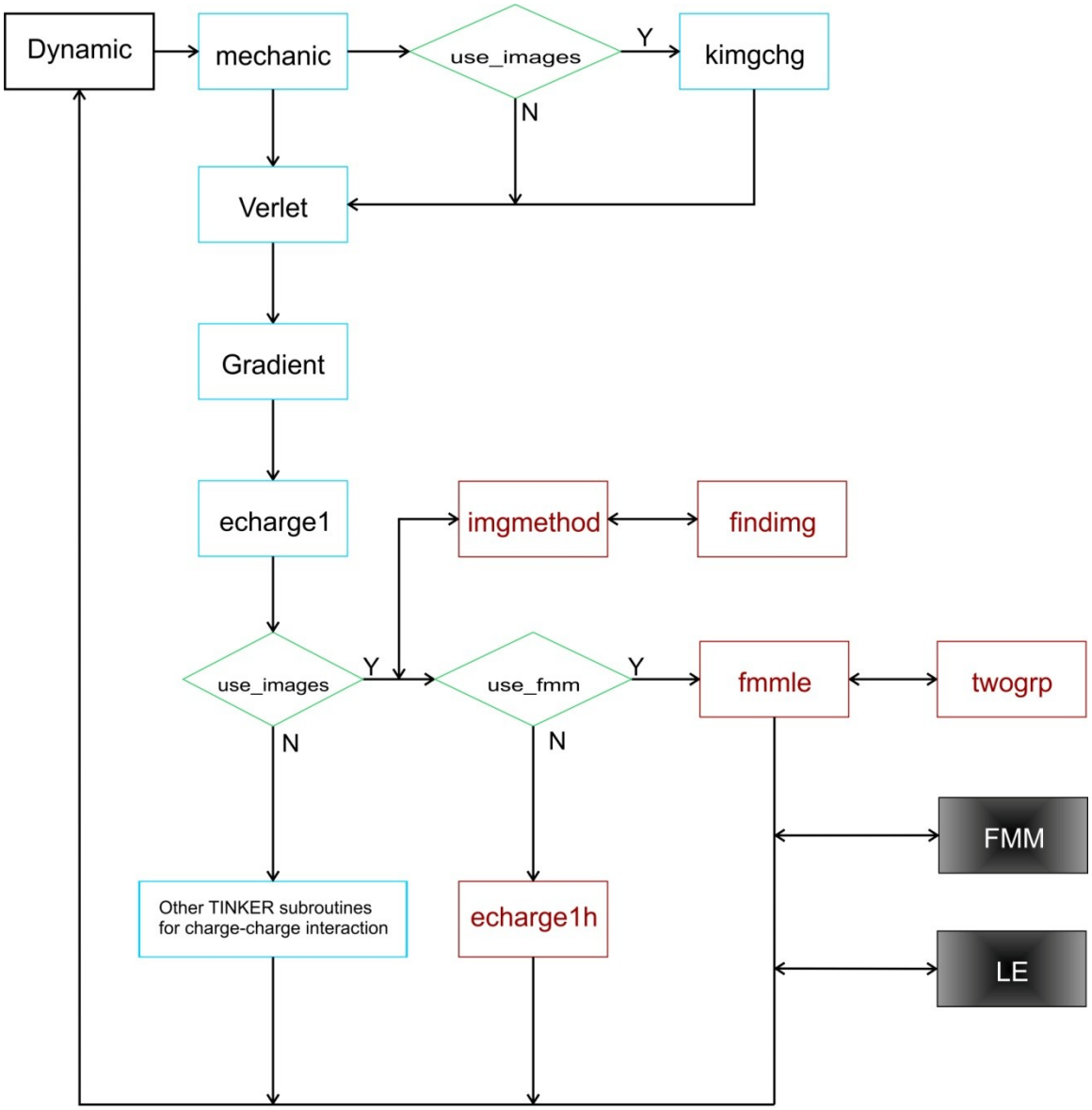

FIGURE 14: The schematic illustration of the flowchart of the integration of the ICSM into TINKER. The blue boxes represent the subroutines that belong to Tinker that were modified. The red boxes are subroutines belonging to ICSM that were added. The green boxes are conditional statements and the black boxes are modules which are called from ICSM.

CHAPTER 3: NUMERICAL RESULTS

In order to validate the ICSM integrated into the TINKER molecular modeling package, we applied TINKER 6.0 with the integrated ICSM to liquid water simulations. We ran the same benchmarks on the physical properties of bulk water as in our previous paper (1), including the density, the structural oxygen–oxygen radial distribution function, the self-diffusion coefficient, and the dielectric constant. The results are compared to the PME calculations included in the TINKER package and to the published results obtained by the original in-house ICSM software (1). That is, unless otherwise specified, for each test, three different kinds of simulations are presented: one by TINKER with the built-in PME using neighbor lists, one by TINKER with the integrated ICSM using one image charge ($N_i = 1$), and one by the original in-house ICSM program used in Ref. (1) also with one image charge. And for convenience, the corresponding results are marked as ‘‘TINKER–PME’’, ‘‘TINKER–ICSM’’, and ‘‘ICSM’’, respectively. The machine on which we compile the integrated package and perform the test simulations is a Red Hat Enterprise Linux 6 server with dual Xeon E5450 processors and 32 Gb of memory, and the compiler is gcc-4.4.5-6.

3.1.1 Testing on Water

3.1.1.1 Simulation Details

We use the TIP3P (70) all-atom model to characterize properties of bulk water. The implementation details of the in-house ICSM program can be found in Ref. (1). In all three approaches employed, positions and velocities of particles are calculated using the Velocity Verlet algorithm, coupled with the Nosé–Hoover thermostat. For each test, three different simulation boxes with L = 30, 45, and 60 Å are considered, the number of atoms contained in these boxes being 1224, 4224, and 10,275, respectively. The corresponding initial coordinates are the identical input files used in Ref. (1), which were generated from an equilibrium 200 ps simulation in NPT ensemble, performed by the GROMACS program package. Since it was found in reference (1) that the thickness of the buffer zone had to be at least 6 Å in order to yield the uniform density of the simulated bulk water inside the TO Λ, we use a six Angstrom buffer zone for all tests reported in this work, namely, τ = 6 Å. In addition, each dynamics simulation was 1.1 ns long with the first 0.1 ns used for equilibration time and the integration time step was set to 2 fs. The simulations were performed under constant temperature conditions at T = 300 K. The trajectories were recorded at every 0.2 ps for subsequent analysis. In all simulations, for the FMM-Yukawa software, the maximum number of particles in a leaf box of the adaptive oct-tree structure was set to 80 (nbox = 80). The number of terms in the multipole and local expansions and that in the plane wave expansion were both set to 9 for three-digit accuracy (nterms = 9, nlambs = 9).

For the proposed TIP3P liquid water tests, a typical command line for running TINKER with the integrated ICSM would be:

Dynamic water_30A.xyz 550000 2.0 0.2 2 300

where Dynamic is the TINKER executable, *water_30A.xyz* represents the input file for the 30 Å simulation box, 550,000 is the number of total time steps, 2.0 is the time step size in femtoseconds, 0.2 indicates the trajectory is recorded after every 0.2 ps, 2 indicates to use the NVT ensemble, and 300 is the temperature, respectively. A corresponding example keyword parameter file "*water_30A.key*" is included in Appendix B.

3.1.1.2 Results and Discussion

To validate TINKER with the integrated ICSM, as in Ref. (1), first we examine the local particle density across the simulation box as a measure of the homogeneity of the simulated bulk water. More specifically, we compute the relative density of oxygen atoms along the diagonal of the TO simulation box. First, due to symmetry, we can select two opposite vertexes on the TO box and then calculate the density along the line that connects them. To evaluate the density, we consider 11 equally spaced positions on this line, draw small spheres of radius r=5Å around each position and then count the number of water molecules in each sphere. This count is then converted to the particle density ρ_i and then normalized using the number density of the entire box to give ρ_r. The results of this experiment are shown in Figure 15. As can be seen, for all three box sizes, the relative densities obtained by the three different approaches are in good agreement with each other, all showing a uniform density distribution with some statistical variations which are quantitatively summarized in Table 1.

TABLE 1: Standard deviations of relative densities along the diagonal of TO simulation boxes.

	TINKER–ICSM	TINKER–PME	ICSM
30 Å	0.004	0.004	0.002
45 Å	0.004	0.003	0.003
60 Å	0.003	0.004	0.004

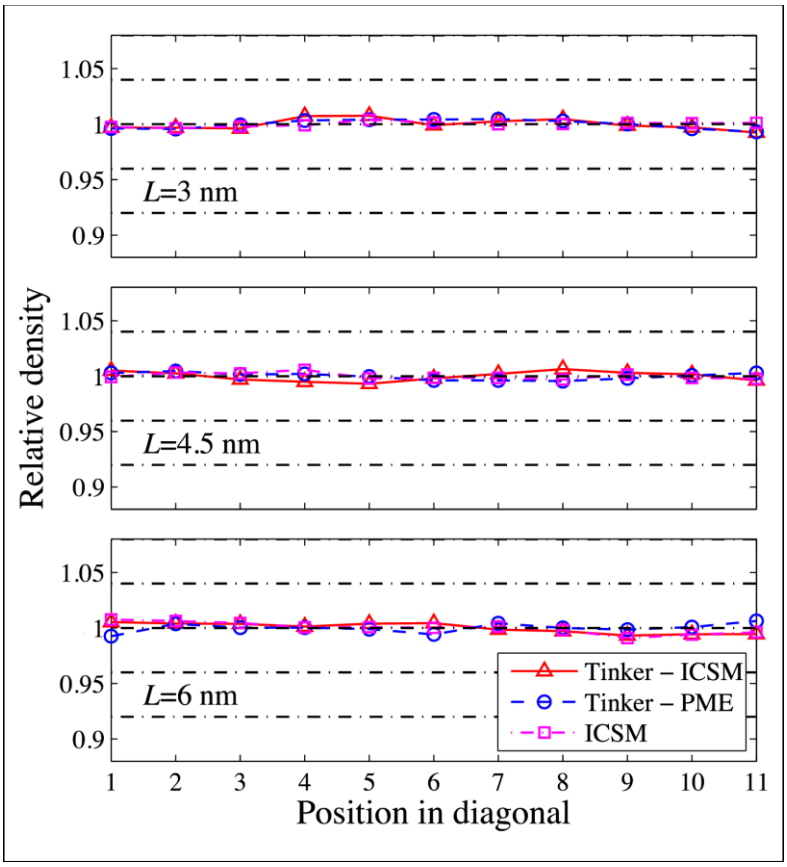

FIGURE 15: Computed relative density along the diagonal of the TO simulation for three different box sizes with L = 30, 45, and 60 Å, respectively.

Next, we examine the structure of the simulated bulk water. We do this by evaluating the structure using $g_{oo}(r)$, the oxygen–oxygen radial distribution function (RDF), over the entire simulation box.

The RDF is calculated as (14):

$$g_{oo}(r) = \frac{1}{4\pi\rho r^2 dr}\frac{1}{N}\langle \sum_{a}^{N} \sum_{r<r_{\alpha\beta}(\tau)\leq r+dr, \beta\neq\alpha}^{N} 1\rangle_{\tau}, g_{oo}(0) = 0 \quad (3.1)$$

where N is the total number of molecules, ρ is the number density, dr is the window size (0.8 Å), $r_{\alpha\beta}(\tau)$ is the minimum image oxygen to oxygen distance between molecules α and β at time τ and $\langle \cdots \rangle_{\tau}$ denotes averaging over all trajectory frames (1).

These results are plotted in Figure 16 - 17. Recall that the most important features of $g_{oo}(r)$ are the locations as well as the magnitudes of the first three density peaks and the first two density minima. As can be seen, for all three box sizes, the RDFs obtained by the three different approaches are in excellent agreement with each other, and a closer look at $g_{oo}(r)$ using a higher resolution, shown in the insets of Figure 16 – 17, reveals some noticeable but yet slight difference only for the first two maxima and the first minimum. For example, the first density maxima all lie at 2.76 Å for all box sizes for all three models, while their values are shown in Table 2. On the other hand, the first density minima are all located around 3.56 Å and their values are given in Table 3.

TABLE 2: Magnitudes of the first density maxima in the computed RDFs.

	TINKER–ICSM	TINKER–PME	ICSM
30 Å	2.740	2.750	2.770
45 Å	2.720	2.720	2.740
60 Å	2.700	2.700	2.710

TABLE 3: Magnitudes of the first density minima in the computed RDFs.

	TINKER–ICSM	TINKER–PME	ICSM
30 Å	0.864	0.866	0.858
45 Å	0.880	0.881	0.876
60 Å	0.900	0.901	0.897

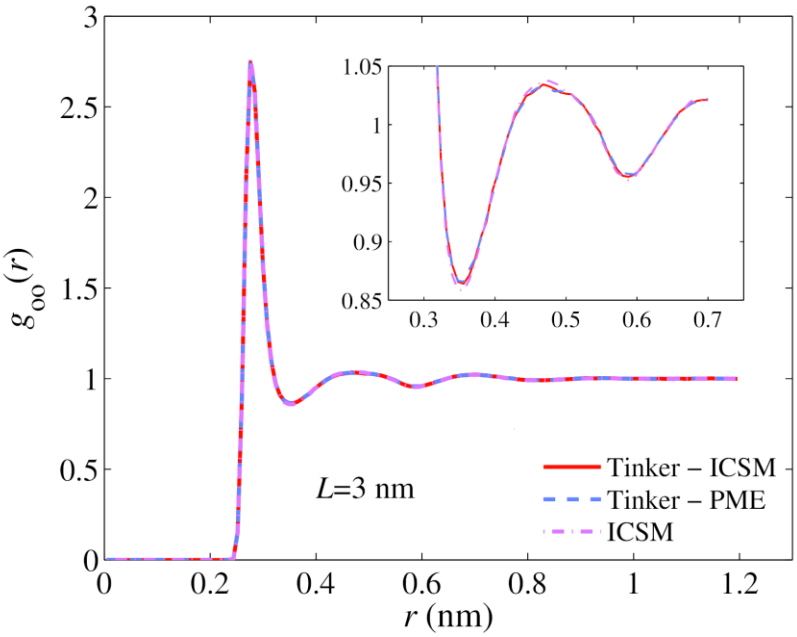

FIGURE 16: Computed Oxygen-Oxygen RDFs for a 30Å TO simulation box. The insert shows a close-up of the first two density minima.

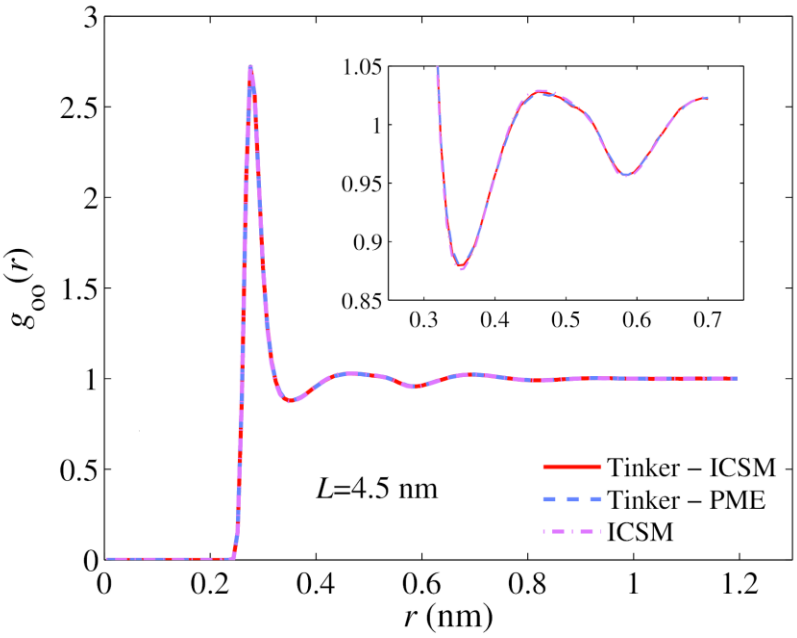

FIGURE 17: Computed Oxygen-Oxygen RDFs for a 45Å TO simulation box. The insert shows a close-up of the first two density minima.

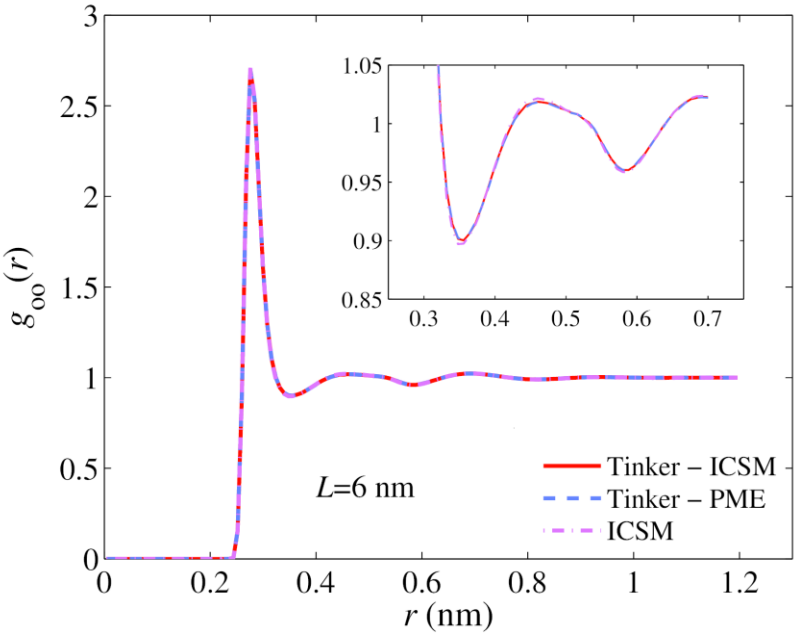

FIGURE 18: Computed Oxygen-Oxygen RDFs for a 60Å TO simulation box. The insert shows a close-up of the first two density minima.

Next, we examined the dynamical properties of the simulated water. More precisely, as in (1), we choose to evaluate the self-diffusion coefficient D, a transport coefficient characterizing how quickly equilibrium is established in particle density following a small perturbation. These diffusion coefficients were calculated from the mean square displacement (MSD) of all oxygen atoms using the Einstein relation (22):

$$\lim_{t\to\infty}\langle|\boldsymbol{r}(t'+t)-\boldsymbol{r}(t')|^2\rangle = 6Dt, \tag{3.2}$$

where **r**(*t*) is the position of the oxygen atom of the water molecule at time t, *D* is the diffusion coefficient, and the brackets denote averaging over all water molecules and time at origins *t'*. The diffusion coefficient was estimated from the slope of the linear part of the line at long times of MSD versus time plot. The initial part of the line is influenced by inertial effects and is not included in this calculation (1).

The self-diffusion coefficients obtained by the three different approaches are recorded in Table 4. As it shows, the self-diffusion coefficient obtained by TINKER with the integrated ICSM is consistent, within statistical errors, with both the values obtained by the PME included in TINKER and the initial findings from Ref. (1) which were obtained using the original in-house ICSM software. Also note that the standard deviation decreases as the box size increases.

TABLE 4: Computed self-diffusion coefficients *D* with standard deviations (unit: 10^{-9} m^2s^{-1}).

	TINKER–ICSM	TINKER–PME	ICSM
30 Å	6.15(±0.12)	6.16(±0.12)	6.28(±0.03)
45 Å	6.16(±0.04)	6.24(±0.07)	6.19(±0.02)
60 Å	6.03(±0.03)	6.06(±0.05)	6.02(±0.01)

Last, to validate TINKER with the integrated ICSM, we consider a model devised by Berendsen (71) in order to determine the dielectric properties of the simulated bulk water by computing its dielectric constant ε. In this model a central sphere with permittivity ε is centered in a spherical layer with permittivity ε' and then embedded in dielectric continuum with permittivity ε_R. The spherical layer allows cases where the dielectric constant at the explicit/implicit interface may differ from that on either side of the boundary. The original cavity model is recovered by setting ε' to either ε or ε_R. The dielectric constant is then given by (1):

$$\varepsilon = \frac{1 + \frac{B}{A} 2\varepsilon' \left[(2\varepsilon_R + \varepsilon') + \left(\frac{R}{R_c + \tau} \right)^3 (\varepsilon_R - \varepsilon') \right]}{1 - \frac{B}{A} \left[(2\varepsilon_R + \varepsilon') + 2 \left(\frac{R}{R_c + \tau} \right)^3 (\varepsilon_R - \varepsilon') \right]}, \tag{3.3}$$

where

$$A = (2\varepsilon_R + \varepsilon')(2\varepsilon' + 1) - 2 \left(\frac{R}{R_c + \tau} \right)^3 (\varepsilon_R - \varepsilon')(1 - \varepsilon')$$

and

$$B = \left(\frac{1}{\varepsilon_o} \right) \left(\frac{\langle \boldsymbol{M^2} \rangle}{3 k_b T V(R)} \right).$$

The expression for B describes the fluctuations on the total dipole moment **M**(*R*) of a spherical sample with radius *R*. Here k_b is the Boltzmann constant, ε_o is the vacuum permittivity, *T* is the temperature, and *V(R)* is the volume of the sample (1). Ballenegger and Hansen (72) show that if ε' is set to ε_R, Eq. (3.3) reduces to:

$$\varepsilon = \frac{1 + B\frac{2\varepsilon_R}{2\varepsilon_R + 1}}{1 - B\frac{1}{2\varepsilon_R + 1}} \tag{3.4}$$

Further reduction of this expression to the known Kirkwood–Frölich and the Clausius–Mossotti formulas is possible by setting ε_R to ε and 1, respectively. Expression (3.4) must be applied at relatively large *R* since it was derived based on continuum electrostatics (72). Although the upper bound for *R* would be $R_c + \tau$ in our system, we limit the radii in our calculations to $R \leq R\_max = (\sqrt{3}/4)L$ to ensure that no periodic images are included in the calculation. The dielectric constant relies on the quadratic fluctuations of the total dipole moment **M**(*R*) which scales linearly with the sample volume. Therefore, the inclusion of any periodic images in the sample would violate the linear scaling of $\langle \boldsymbol{M^2} \rangle$ and invalidate the fluctuation formula (1). The water molecules in the layer $R \leq R_{max} = (\sqrt{3}/4)L$ are, therefore, included as part of the dielectric continuum. The permittivity of the continuum $\varepsilon' = \varepsilon$ was set using Eq. **(**3.3) in our calculations and convergence was achieved in just a few iterations (1).

Once a crossover from a finite-size to macroscopic dimensions is achieved, the dielectric constant should be largely independent of the geometry of the investigated object. In the case of ε(R), a plateau in the graph is expected to start at some radius.

Figure 19 shows the computed dielectric constant ε(R) obtained, respectively, by TINKER with the integrated ICSM and the original in-house ICSM software. Again, good agreement between these two simulation results can be observed.

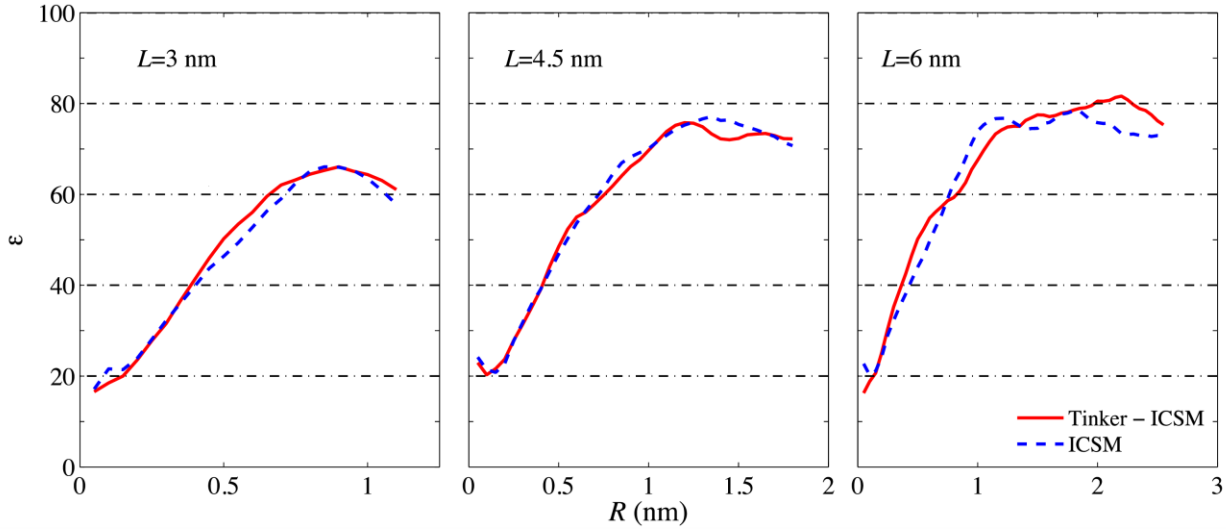

FIGURE 19: Dielectric constant ε(R) as a function of the spherical sample radius R computed using Formula (A3) of Ref. [1] for three different TO simulation boxes with L = 30, 45,and 60 Å, respectively. For sufficiently large R, ε(R) is seen to level off, and the plateau value in ε(R) is identified as the static dielectric constant of the material. Note that, as in Ref. **(1)**, this static dielectric constant is seen to grow from ~65 at L = 30 Å to ~80 at L = 60 Å.

Finally, to test the performance of the ICSM integrated into TINKER, we compare the timing required to run TINKER with the integrated ICSM using the FMM against the timing required to run TINKER with the built-in PME. To this end, in addition to the three simulation boxes used above, larger simulation boxes with = 80, 100, 120, and 140 Å are also considered. For each case, the corresponding simulation was run for 1000 time steps, and the time required for completion, as calculated using the TIME function built into UNIX, is plotted in Figure 20 as a function of the number of atoms in the system. As can be seen, for relatively small systems, TINKER with the built-

in PME runs faster than TINKER with the integrated ICSM, while for sufficiently large systems the latter clearly outperforms the former. The break-even point is at around 30,000 atoms (Figure 20.)

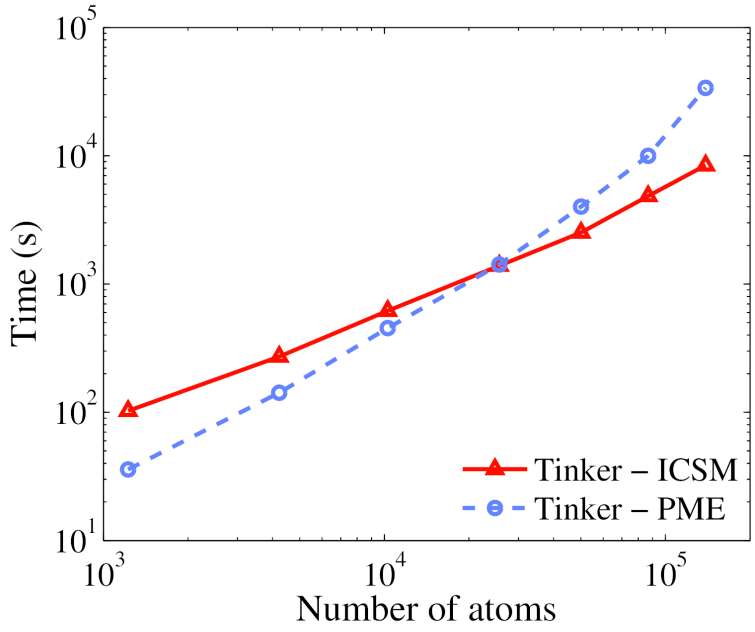

FIGURE 20: Results of timing tests using TINKER with the integrated ICSM and TINKER with the built-in PME, respectively. Test runs were carried out for 1000 time steps and the time used was calculated using the TIME function built into UNIX. TINKER with the integrated ICSM starts to outperform TINKER with the built-in PME at approximately 30,000 atoms.

3.1.2 Testing on KcsA Channel

3.1.2.1 Ion Channel/KcsA History and Description

Ion channels are membrane-spanning proteins that form a pathway for the movement of ions through cell. Some examples of their many functions include the secretion of hormones into the bloodstream, generating the electrical impulses that establish information transfer in the nervous system, and controlling the pace of the heart

and other muscles (73). Although the study of ion channels is relatively new, great strides in their structure and function have been made.

The supposition of the existence of means of transporting ions from the exterior of a cell to the interior began a mere 63 years ago with Hodgkin and Huxley's study of the electrical activity in squid giant axon (74; 75). They showed that both the sodium and potassium ions contributed to the ionic current and that the fluxes were opposed. Twenty years later Hladky and Haydon used the small antibiotic gramicidin to actually prove the existence of an ionic pathway (74; 76) . Great technical strides in many diverse areas of science culminated in the completion of the x-ray crystal structure of the potassium ion channel KcsA (potassium crystallographically-sited activation channel) by MacKinnon in 1998 (74; 77).

Potassium channels are specialized proteins able to facilitate and regulate the conduction of K+ ions in particular through cell membranes (78; 73). KcsA is comprised of 5,819 atoms (Figure 21) which form four identical proteins(Figure 22) each of which contain two alpha-helices connected by a loop of approximately 30 amino acids. These proteins combine to form three primary sections of the channel – the opening pore on the side of the cell interior, a small cavity which is filled with water and a mix of sodium (Na+) and potassium ions (K+) and the selectivity filter. The selectivity filter, which is comprised of four specific cation binding sites (S1 to S4) (Figure 25) formed by the backbone carbonyl groups of conserved residues common to all K+ channels (TVGYG), allows fast conduction of K+ while being highly selective for potassium ions over sodium ions (Figure 23) (78).

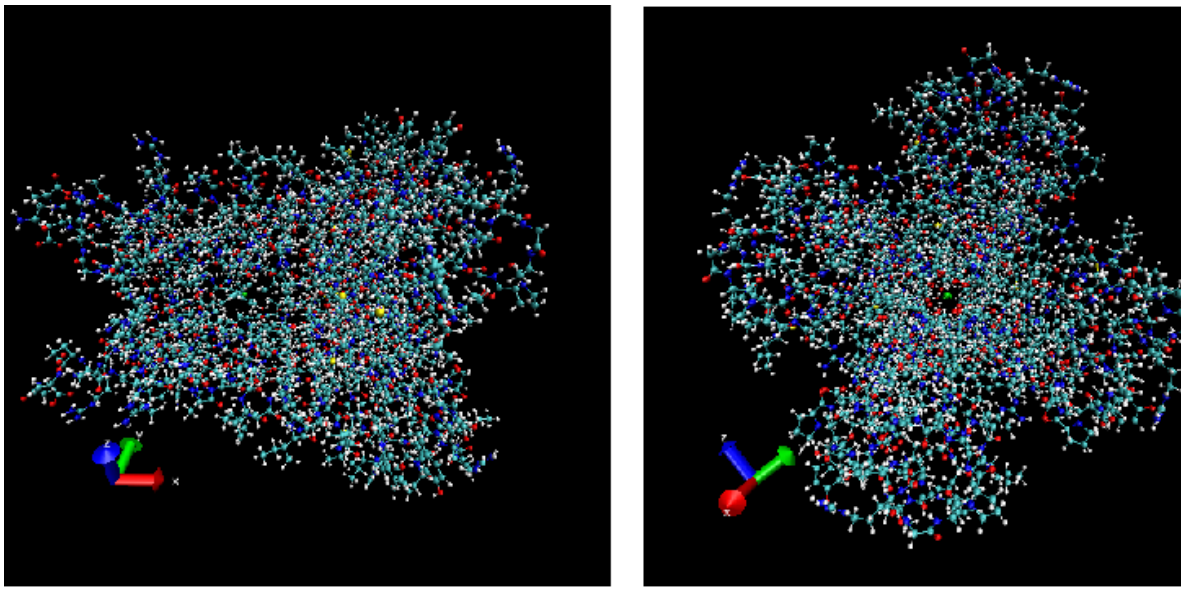

FIGURE 21: All atom model of KcsA from a) Side and b) extracellular end.

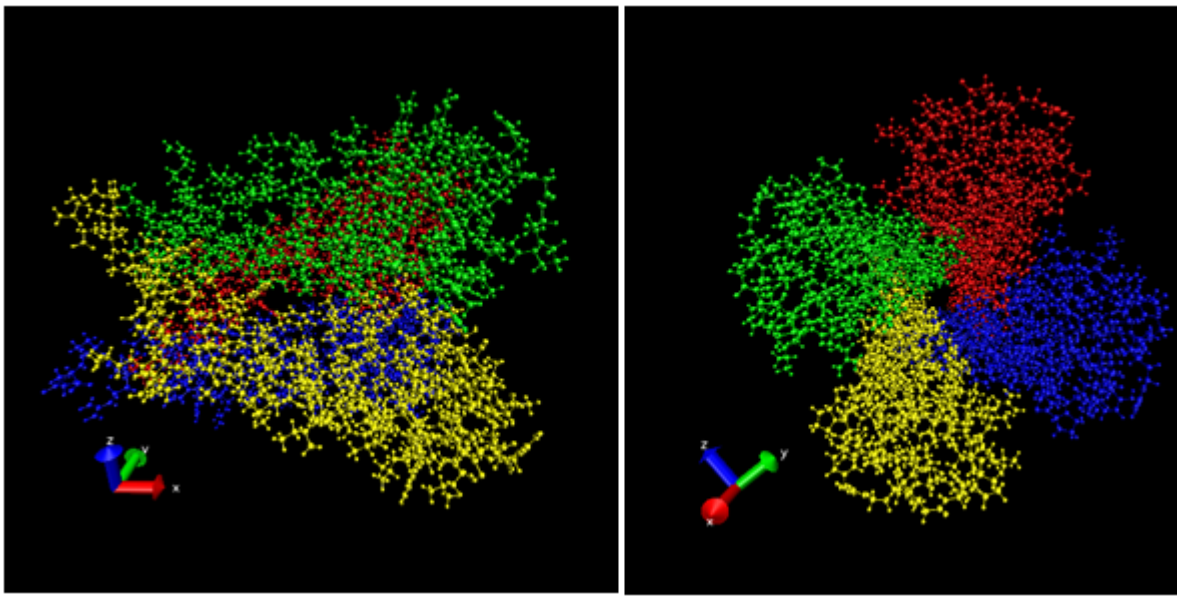

FIGURE 22: View of four identical proteins that make up the channel

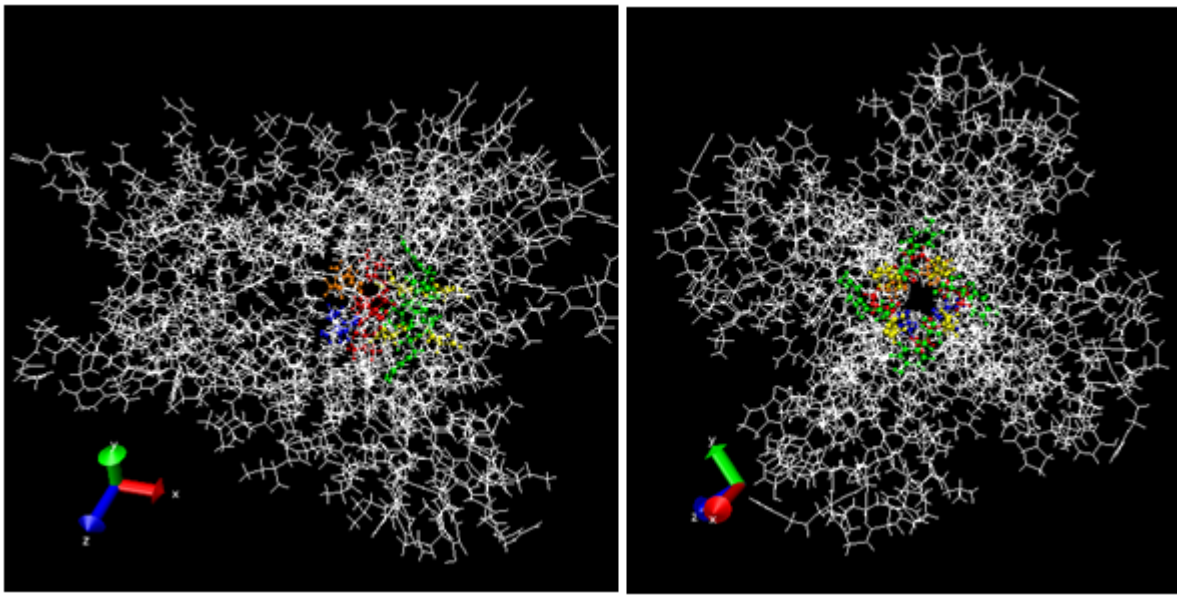

FIGURE 23: The amino acids that make up the selection filter – blue – threonine(T), red – valine(V), yellow – glycine(G), green – tyrosine(Y).

3.1.2.2 Simulation Details

For our testing, we used the file *2A9H.pdb* from the RCSB Protein Data Bank (www.pdb.org). It was entered into the PDB after discovery by Nuclear Magnetic Resonance(NMR) in 2005 (79). This version of the structure of KcsA is a closed structure with a high-affinity peptide antagonist charybdotoxin attached to the extracellular side of the channel. The data from the PDB was converted to a Tinker input file using the built in program *pdbxyz.x*. We then removed the toxin and added five waters to the central cavity. We also shifted all of the atoms in the file in order to center the channel in a 60 Angstrom TO box. The system was then minimized with all particles allowed to move freely using a program included in Tinker and then run for 200ps for initial equilibrium. While running the following simulations, we used the ACTIVE keyword in the key file to lock down all atoms of the channel except for the amino acids

making up the selection filter, the waters, and the ions. We used the parameters contained in the Amber99 force field (15; 16), also included in Tinker. We chose to use the Velocity Verlet algorithm and a Nose-Hoover bath set to 300K. Each time step is 2 fs, and the trajectory is printed out every .1 ps. For these simulations, the above system for the ICSM was modified slightly to accommodate this unique environment.

We considered only the amino acids (Figure 23) contained in the selection filter, the ions, and the waters in the pore as the solute. The protein and surrounding membrane are treated as the infinite dielectric continuum of dielectric constant $\varepsilon_o = 2$. Although there is water on both the extracellular side and the intercellular side of the channel, we considered the opening into the pore to be small enough to disregard this. The small size of the pore as compared to the rest of the protein and the surrounding membrane also allows us to treat the protein itself as the buffer zone so Tau in this case is set to 0.

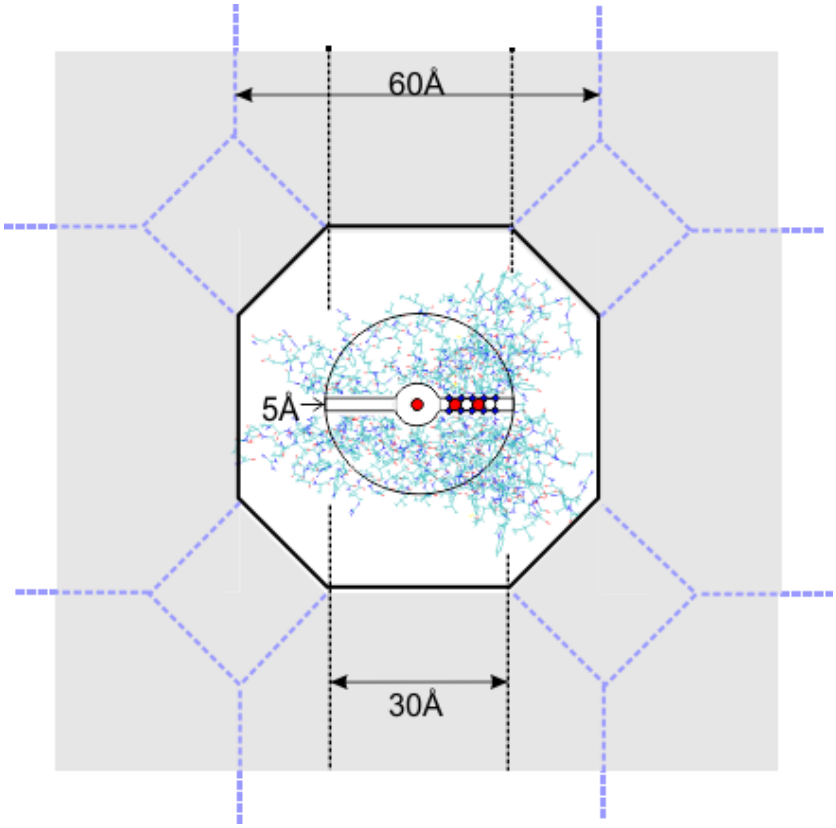

FIGURE 24: A schematic showing the setup for the KcsA system. The red dots are ions. The gray area is the dielectric continuum with $\varepsilon_o = 2$, the area in the circle is the productive region. The dielectric inside the TO box is $\varepsilon_i = 1$. The area between the dashed black lines represents the size of the cellular membrane.

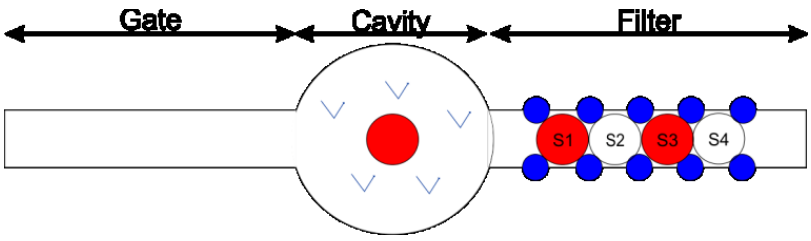

FIGURE 25: Example of channel with three ions. One ion typically remains in the water-filled cavity and two in the filter area. The blue dots represent the carbonyl oxygen from the amino acids that make up the selectivity filter.

3.1.2.3 Numerical Results

We first collected the data for channel permanent potential using the program *analyze.x* that is included in the Tinker package. This routine was modified to print the energy on a particular ion as it moved through the channel. The ion was moved 2 Angstrom at a time and then the potential energy was calculated on the ion at that position. The results for the channel prior to minimization and equilibration are shown in Figure 26. The result closely resembles the channel permanent potential found by Jung, Lu and Mascagni for their tests on ERINP (80). We then allowed all of the atoms to move freely in the system and again minimized the file using the minimization program built into Tinker. This file was used as the initial input for the remainder of the testing.

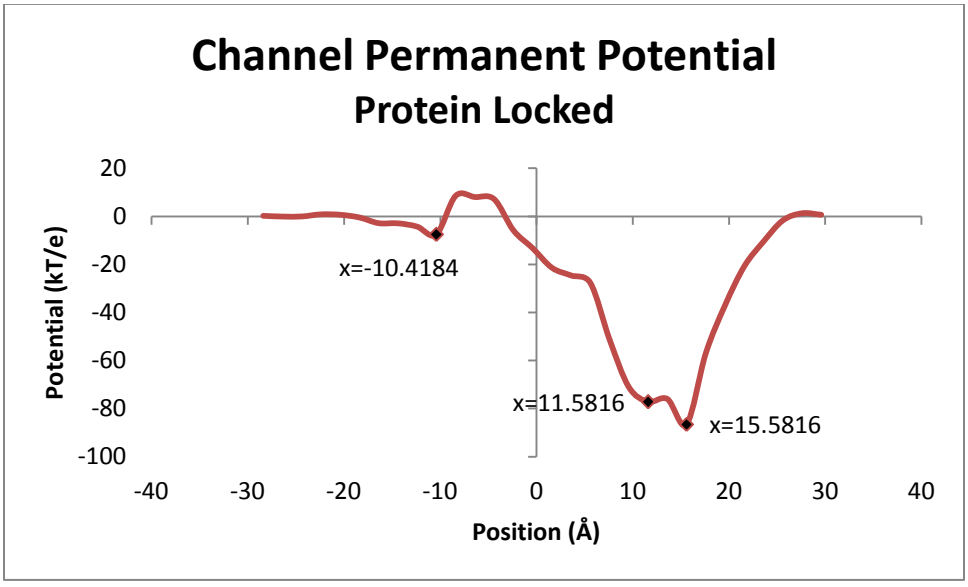

FIGURE 26: Potential energy profile of KcsA prior to minimization.

Next we tested whether or not this current set up of the ICSM will work with an ion channel buried in a membrane. We ran 100 ps simulations for every permutation of ion locations in the channel. We found that regardless of starting position in the filter the two ions will come to rest in S2 and S4. This relaxation period for the ions is very short. We found that by .3 ps, the ions and water in the filter had reached their approximate final positions. Table 5 shows the initial and final positions for two ions present in the channel. All of these files were started with one ion in the cavity and then a second ion was placed in the filter in positions approximating the binding sites. The original positions for the ions in the filter were calculated by finding the midpoint between the carbonyl oxygen of the filter amino acids. An example of the trajectories for four of these simulations can be seen in Figure 27. In this instance, both of the ions move into the filter and come to rest in positions S1 and S3. This configuration would be a transitional phase between an ion exiting the channel and another one entering. After adding a third ion into the cavity, the ion located in position S3 moves into close proximity to the carbonyl oxygen belonging to Tyrosine(T). Figure 28 shows the trajectories for four permutations (Cavity/s2/s4, Cavity/s1/s3, Cavity/s1/s4, and Cavity/s3/s4) of starting with three ions in the channel. The starting positions for the examples shown can be seen in Table 6.

TABLE 5: Initial and final positions for tests with two ions in the channel

	Cavity – S4		Cavity – S3		Cavity – S2		Cavity – S1	
	Initial	Final	Initial	Final	Initial	Final	Initial	Final
K1	0.418	8.635	0.418	9.049	0.418	8.957	0.418	8.923
K2	18.544	14.739	15.800	14.717	13.324	15.387	10.347	13.451

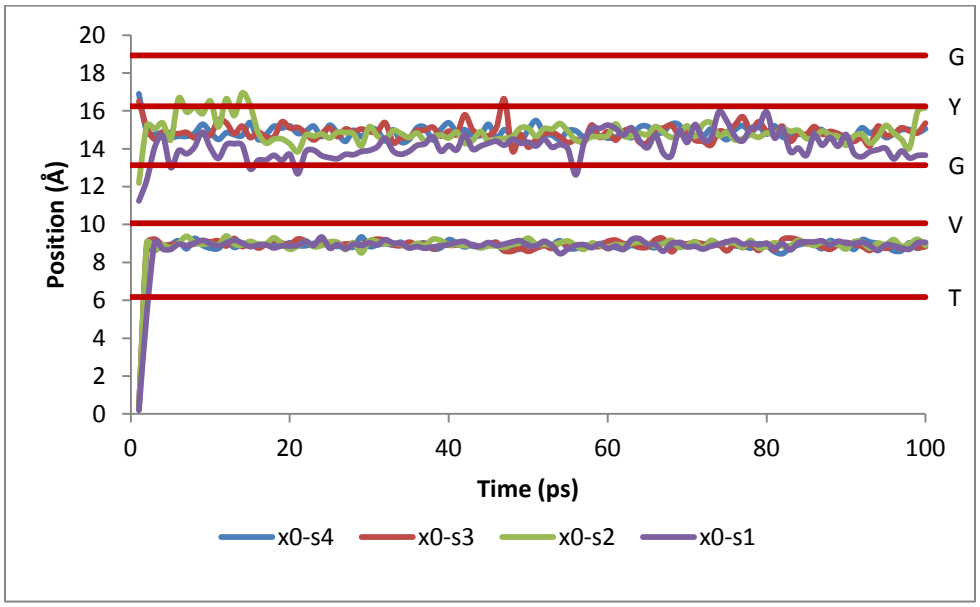

FIGURE 27: Trajectories for examples of simulations with two ions in the channel.

TABLE 6: Initial and final positions for three ions in the channel

	Cavity/S2/S4		Cavity/S1/S3		Cavity/S1/S4		Cavity/S3/S4	
	Initial	Final	Initial	Final	Initial	Final	Initial	Final
K1	0.418	5.310	0.418	5.404	0.418	8.562	0.418	5.092
K2	13.324	11.808	10.347	11.813	10.347	12.047	15.800	11.695
K3	18.544	16.280	15.800	15.727	18.544	16.656	18.544	16.385

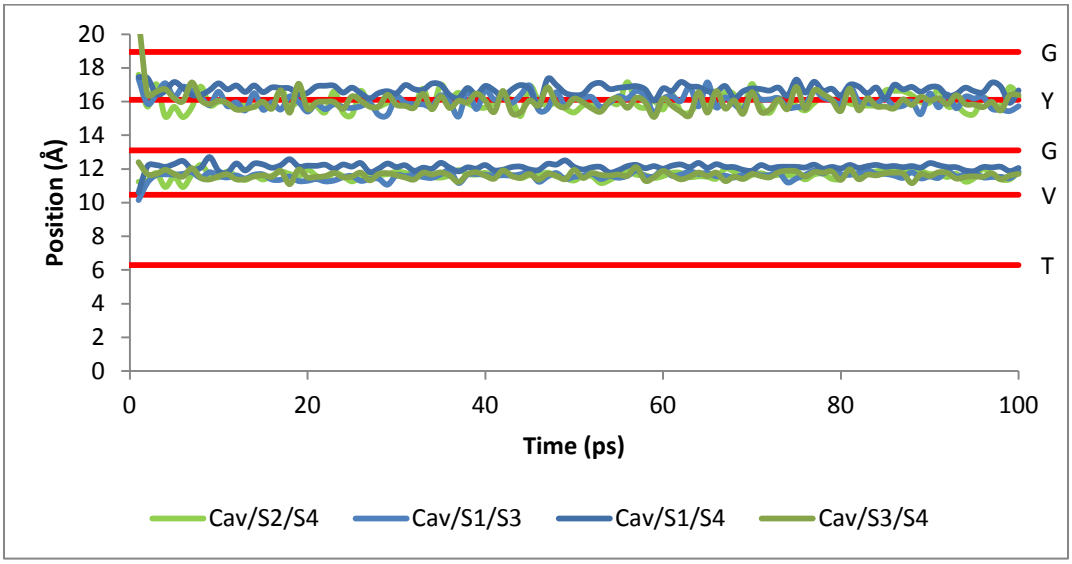

FIGURE 28: Trajectories for four examples of simulations with three ions in the channel.

We ran another test for 200 ps where the initial positions for the three ions are as shown in Table 7. In this test, two of the ions were in the cavity, and one was already located in the filter at approximately the location of S3. We note that in this case the second ion pushed a water into the channel. The results above for three ions in the channel have no water in the filter area. The trajectory for this case is shown in Figure 29. The difference in the final resting location for the filter ions is just slightly different with the ion that is located close to the Tyrosine shifting slightly into the binding location S3. This result is comparable to the results obtained by Caperner in (81). We note that their assignment for labels S1 to S4 is opposite of the standard used here (S1 is toward the exterior of the cell membrane.)

TABLE 7: Starting and Final Positions for the 3 ions in the channel for example case.

	Starting Position	Final Position
K1	-5.4184	5.14125
K2	2.302215	11.59929
K3	12.3509	16.36362

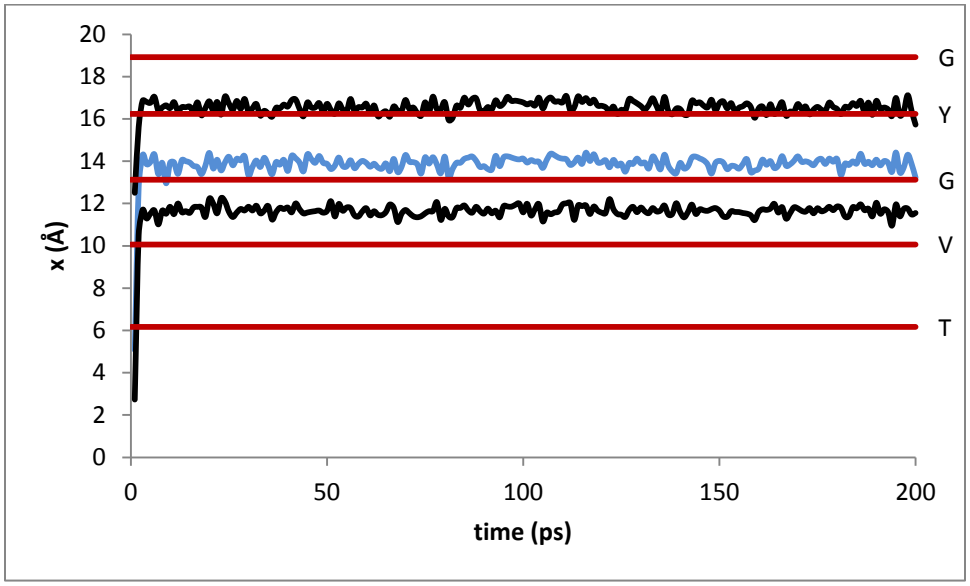

FIGURE 29: Relative positions of the ions and water in the selectivity filter to the carbonyl oxygen. The black lines are the two ions in the filter while the blue line is the water.

In summary, for the investigations into the positioning of the filter ions we find that in the transitional state of two ions in the channel the ions prefer locations S1 and S3 whereas in the three ion state, they come to a resting position of S2 and S4.

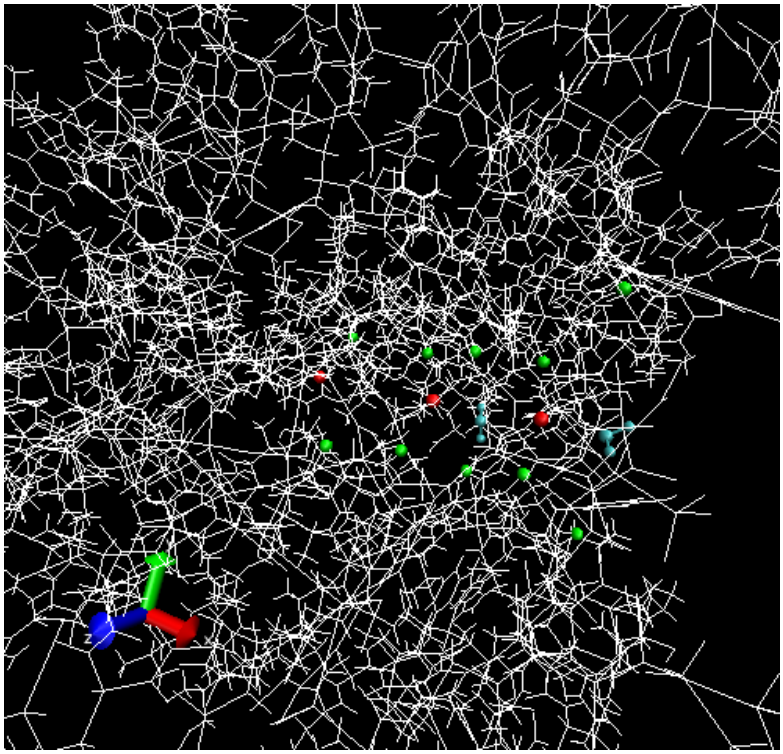

FIGURE 30: A close-up of the selection filter after the final time step showing the positions of the carbonyl oxygen from two chains (green), the potassium ions (red) and the two waters in the filter (cyan). These locations correspond to the ones shown in Figure 28. Image created using VMD.

.

CHAPTER 4: CONCLUSION AND FUTURE WORK

In this thesis, we studied molecular dynamics and discussed two different methods for simulation based on the type of boundary conditions used to calculate the electrostatic interactions – using periodic boundary conditions as in explicit models versus using non-periodic boundary conditions as in implicit models. We pointed out that both of these methods had inherent issues. Explicit models could introduce artifacts caused by interactions between different copies of the primary simulation box. Implicit models are difficult to implement due to the complexity of solving the 3D Poisson-Boltzmann equation and the loss of molecular interaction near biomolecular surfaces. To resolve these issues, we proposed a method, together with its theoretical foundations, which combines the explicit method and the implicit method by treating the solute and nearby solvation shells atomistically and then treating the remaining solvent as a dielectric continuum.

We then showed how this method (ICSM) can be integrated into the open-source Molecular Dynamics package TINKER written by J. Ponder (82). The method was extensively tested on bulk water, and it was shown to be able to calculate some physical properties of the water including the self-diffusion coefficient, the oxygen-oxygen radial distribution function, and the dielectric constant accurately as compared to the results from the PME included in TINKER. We further showed that our method, when

combined with the FMM, outperforms the PME for sufficiently large boxes where the number of atoms in the box is greater than approximately 30,000 particles.

As a further test of the capability of the modified Tinker code, we conducted a preliminary investigation of the ion channel KcsA to see if the ICSM will handle the inhomogeneity of media in the setting of ion channels. Initial results have shown the potential of the ICSM to simulate the physical property of the filter region of the KcsA channel and the dynamics of the ions in such a channel. Further investigation of the ion channel by the ICSM model will be conducted including the image charges due to the presence of the membrane interfaces.

BIBLIOGRAPHY

1. *An Image-based Reaction Field Method for Electrostatic Interactions in Molecular Dynamics Simulations of Aqueous Solutions.* Lin, Y., et al. 131, 2009, J. Chem. Phys.

2. Alberts, B., Johnson, A. and Lewis, J. *Molecular Biology of the Cell. 4th edition.* New York : Garland Science, 2002.

3. Berg, J.M., Tymoczko, J.L. and Stryer, L. Chapter 3, Protein Structure and Function. *Biochemistry. 5th edition.* New York : W H Freeman, 2002.

4. *Molecular dynamics and protein function.* Karplus, M. and Kuriyan, J. 19, 2005, Proc. Nat. Acad. Sci., Vol. 102, pp. 6679–6685.

5. Nature Education. Essentials of Cell Biology. *Scitable.* [Online] 2012. http://www.nature.com/scitable/ebooks/essentials-of-cell-biology-14749010/118239119.

6. Karadaghi, Salam Al. The 20 Amino Acids and Their Role in Protein Structures. *Basics of Protein Structure.* [Online] 2014. http://proteinstructures.com/Structure/Structure/amino-acids.html.

7. DEPARTMENT OF BIOCHEMISTRY AND MOLECULAR BIOLOGY. Rutgers New Jersey Medical School. *Pre Knowledge - Amino Acids.* [Online] http://njms2.umdnj.edu/biochweb/education/bioweb/PreK/AminoAcids.htm.

8. Particle Sciences Drug Development Services. *Protein Structure. Technical Brief. Volume 8.* 2009.

9. *Molecular Dynamics Simulations of Biomolecules.* Sagui, C. and Darden, T. A. 1999, Amer. Rev. Biophys. Biomol. Struct., Vol. 28, pp. 155-79.

10. *Studies in Molecuar Dynamics. I. General Methods.* Alder, B. J. and Wainwright, T. E. 2, 1959, The Journal of Chemical Physics, Vol. 31, pp. 459-466.

11. *Correlations in the Motion of Atoms in Liquid Argon.* Rahman, A. 1964, Phys. Rev., Vol. 136, pp. A405-11.

12. *Improved Simulation of Liquid Water by Molecular Dynamics.* Stillinger, F. H. and Rahman, A. 4, 1974, Jour. Chem. Phys., Vol. 60, pp. 1545-1557.

13. Standard, J. M. The Verlet Algorithm for Molecular Dynamic Simulations. [Online] May 2013. http://chemistry.illinoisstate.edu/standard/che38037/handouts/380.37verlet.pdf.

14. *Fast Analytical Methods for Macroscopic Electrostatic Models in Biomolecular Simulations.* Xu, Z. and Cai, W. 4, 2011, SIAM Review, Vol. 53, pp. 683-720.

15. *Amber: Assisted model-building with energy refinement.A general program for modeling molecules and their interactions.* Weiner, P. K. and Kollman, P. A. 3, 1981, J. Comput. Chem., Vol. 2, pp. 287-303.

16. *New-generation Amber united-atom force field.* Yang, L. J., et al. 2006, J. Phys.Chem. B, Vol. 110, pp. 13166-13176.

17. *All-atom empirical potential for molecular modeling and dynamics studies of proteins.* MacKerell, Jr., A. D., et al. 1998, J. Phys. Chem. B, Vol. 102, pp. 3586-3616.

18. *CHARMM: A program for macromolecular energy minimization and dynamics calculations.* B. R. Brooks, R. E. Bruccoleri, B. D. Olafson, D. J. States, S. Swaninathan, and M. Karplus. 1983, J. Comput. Chem., Vol. 4, pp. 187-217.

19. *GROMACS: Fast, flexible, and free.* D. van der Spoel, E. Lindahl, B. Hess, G. Groenhof, A. E. Mark, and H. J. C. Berendsen. 2005, J. Comput. Chem., Vol. 26, pp. 1701–1718.

20. *GROMACS 4: Algorithms for highly efficient, load-balanced, and scalable molecular simulation.* B. Hess, C. Kutzner, D. van der Spoel, and E. Lindahl. 2008, J. Chem. Theory Comput., Vol. 4, pp. 435-447.

21. *Evaluation and reparametrization of the OPLS-AA force field for proteins via comparison with accurate quantum chemical calculations on peptides.* Kaminski, G. A., et al. 2001, J. Phys. Chem. B, Vol. 105, pp. 6474-6487.

22. Allen, M. P. and Tildesley, D.J. Computer Simulation of Liquids. Oxford : Oxford University Press, 1987.

23. *Die Berechnung optischer und elektrostatischer Gitterpotentiale.* Ewald, P. 1921, Ann. Phys., Vol. 64, pp. 254-87.

24. *Particle mesh Ewald: An N·log(N) method for Ewald sums in large systems.* Darden, T., York, D. and Pederson, L. 12, 1993, J. Chem. Phys., Vol. 98, p. 10089.

25. *A smooth particle mesh Ewald method.* Essmann, U., et al. 19, 1995, Vol. 103, p. 8577.

26. Hockney, R. W. and Eastwood, J. W. *Computer simulation using particles.* New York : McGraw-Hill International Book Co., 1981.

27. Pollock, E. L. and Glosli, J. *Comments on PPPM, FMM, and the Ewald Method for Large Periodic Coulombic Systems.* Cornell University. p. 31. arXiv:cond-mat/9511134.

28. *Efficient calculations of coulombic interactions in biomolecular simulations with periodic boundary conditions.* Shimada, J., Kaneko, H. and Takada, T. 7, 1993, J. Comp. Chem., Vol. 14, pp. 867-878.

29. *A Fast Algorithm for Particle Simulations.* Greengard, L. and Rokhlin, V. s.l. : J. Comput.Phys., 1987, Vol. 73, pp. 325-348.

30. *A fast algorithm for particle simulations.* Greengard, L. and Rokhlin, V. 2, 1997, J. Comp. Phys., Vol. 135, pp. 280-292.

31. *Effect of artificial periodicity in simulations of biomolecules under Ewald boundary conditions: a continuum electrostatics study.* Hünenbergera, P.H. and McCammon, J.A. 1-2, 1999, Biophys. Chem., Vol. 78, pp. 69-88.

32. *Ewald Boundary Conditions: Influence of Artificial Periodicity on Peptide Conformation.* Weber, W., Hünenberger, P.H. and McCammon, J. A. 2000, J. Phys. Chem., Vol. 104, pp. 3668-3675.

33. *Poisson-Boltzmann Solvents in Molecular Dynamics Simulations.* Wang, J., et al. 5, 2008, Comm. Comp. Phys., Vol. 3, pp. 1010-1031.

34. Roux, B. *Implicit Solvent Models, Computational Biochemistry and Biophysics.* New York : Marcel Dekker, 2001.

35. *Classical Electrostatics in Biology and Chemistry.* Honig, B. and Nicholls, A. 5214, 1995, Science, Vol. 268, pp. 1144-1149.

36. *Improving implicit solvent simulations: a Poisson-centric view.* Baker, N. 2, 2005, Current Opinion in Structural Biology, Vol. 15, pp. 137–143.

37. Leach, A. R. *Molecular Modelling: Principles and Applications, 2nd ed.* s.l. : Prentice-Hall, 2001.

38. *Electrostatics in biomolecular structure and dynamics.* Davis, M. E. and McCammon, J. A. 1990, Chem. Rev., Vol. 90, pp. 509-521.

39. *Structure and mechanism of carbonic anhydrase.* Lindskog, S. 1997, Pharmacol. Ther., Vol. 74, pp. 1-20.

40. *Extending the fast multipole method for charges inside a dielectric sphere in an ionic solvent: High-order image approximations for reaction fields.* Cai, W. and Deng, S. 2007, Jour. Comp. Phys., Vol. 227, pp. 1246–1266.

41. Pathria, R. K. *Statistical Mechanics, 2nd ed.* Oxford : Butterworth–Heinemann, 1996.

42. Hill, T. L. *An Introduction to Statistical Thermodynamics.* New York : Dover, 1986.

43. *The theory of electrolytes. I. Lowering of freezing point and related phenomena.* Debye, P. and Huckel, E. 1923, Phys. Zeitschr., Vol. 24, pp. 185-206.

44. Warwicker, J. and Watson, H. C. 4, 1982, J. Mol.Biol., Vol. 157, pp. 671-679.

45. *Multigrid solution of the Poisson-Botzman equation.* Holst, M. and Saied, F. 1, 1993, Vol. 14, pp. 105-113.

46. *Accelerated Poisson-Boltzman Calculations for Static and Dynamic Systems.* Luo, R., David, L. and Gilson, M. K. 13, 2002, J. Comp. Chem., Vol. 23, pp. 1244-1253.

47. *DIRECT SOLUTION OF THE POISSON EQUATION FOR BIOMOLECULES OF ARBITRARY SHAPE, POLARIZABILITY DENSITY, AND CHARGE DISTRIBUTION.* Orttung, W. H. 1977, Ann. N.Y. Acad. Sci., Vol. 303, pp. 22-37.

48. *Adaptive Multilevel Finite Element.* Holst, M., Baker, N. and Wang, F. 15, 2000, J. Comp. Chem., Vol. 21, pp. 1319-1342.

49. *The Finite Element Approximation of the Nonlinear Poisson–Boltzmann Equation.* Chen, L. and Holst, M. J., Xu, J. 6, 2007, SIAM J. Numer. Anal., Vol. 45, pp. 2298–2320.

50. *Electrostatic calculations for an ion channel. I. Energy and potential profiles and interactions between ions.* Levitt, D. G. 2, 1978, Biophys. J., Vol. 22, pp. 209–219.

51. Vorobjev, Y. N. and Scheraga, H. A. *Theory of the Stability of Lyophobic Colloids.* Amsterdam : Elsevier, 1948.

52. *Fast Boundary Element Method for the Linear Poisson−Boltzmann Equation.* Boschitsch, A. H., Fenley, M. O. and Zhou, H. X. 10, 2002, J. Phys. Chem. B, Vol. 106, pp. 2741–2754.

53. *AFMPB: An adaptive fast multipole, Poisson-Boltzmann solver for calculating electrostatics in biomolecular systems.* Lu, B., et al. 2010, Comput. Phys. Commun, Vol. 181, pp. 1150-1160.

54. *Dielectric response of a polar fluid trapped in a spherical nanocavity.* Blaak, R. and Hansen, J. P. 14, 2006, J. Chem. Phys., Vol. 124.

55. *New Computational Models for Electrostatics of Macromolecules in Solvents.* Dai, J., et al. 4, 2007, IEEE Transactions on Magnetics, Vol. 43, pp. 1217 - 1220 .

56. *Chapter 6 Hybrid Explicit/Implicit Solvation Methods.* Okur, A. and Simmerling, C. 2006, Annual Reports in Computational Chemistry, Vol. 2, pp. 97–109.

57. *An efficient hybrid explicit/implicit solvent method for biomolecular simulations.* Lee, M. S., Salsbury, Jr., F. R. and Olson, M. A. 16, 2004, J. Comp. Chem., Vol. 25, pp. 1967-1978.

58. *Evaluation of Poisson Solvation Models Using a Hybrid Explicit/Implicit Solvent Method.* Lee, M. S. and Olson, M. A. 11, 2005, J. Phys. Chem. B, Vol. 109, pp. 5223-5236.

59. *Image approximation to the reaction field.* Friedman, H. R. 5, 1975, Mol. Phys., Vol. 29.

60. *Extending the Fast Multipole Method to Charges Inside or Outside a Dielectric Sphere.* Cai, W., Deng, S. and Jacobs, D. 2007, J. Comput. Phys. 223, pp. 846-864.

61. Chew, W.C. Fast Algorithms for Wave Scattering Developed at the University of Illinois' Electromagnetics Lagoratory. *IEEE Antenn. Propag. Mag.* 1993, 35, pp. 22-32.

62. Pan, Y.C., Chew, W.C. and Wan, L.X. A Fast Multipole Method Based Calculation of the Capacitance Matrix for Multiple Conductors Above Stratified Media. *IEEE Trans. Microw. Theory Tech.* 2001, 49, pp. 480-490.

63. *Hydrodynamische Untersuchen nebst einem Anhang uber die Probleme der Elecktrostatik und der magnetischen.* Neumann, C. 1883, Teubner, Leipzig, pp. 279-282.

64. *Electrostatic Image Theory for the Dielectric Sphere.* Lindell, I.V. s.l. : Radio Sci., 1992, Vol. 27, pp. 1-8.

65. *Image Theory for Electrostatic and Magnetostatic Problems Involving a Material Sphere.* Lindell, I.V. s.l. : Am. J. Phys., 1993, Vol. 61, pp. 39-44.

66. *Electrostatic Image Theory for Layered Dielectric Sphere.* Lindell, I.V., Ermutlu, M.E. and Sihvola, A.H. s.l. : IEE Proc.-H, 1992, Vol. 139, pp. 186-192.

67. Jackson, J.D. *Classical Electrodynamics.* New York : John Wiley, 1999.

68. *Charge Images in a Dielectric Sphere.* Norris, W.T. s.l. : IEE Proc.-Sci. Meas. Technol., 1995, Vol. 142, pp. 142-150.

69. *ICSM: An order N method for calculating electrostatic interactions added to TINKER.* **Baker, K.**, et al. 2013, Comp. Phys. Comm., Vol. 184, pp. 19-26.

70. *J. Chem. Phys.* Jorgenson, W. L., et al. 926, 1983, Vol. 79.

71. Berendsen, H. J. C. “Molecular dynamics and Monte Carlo calculations on water". *CECAM Report No. 29.* 1972.

72. *J. Chem. Phys.* Balleneger, V. and Hansen, J. P. 114711, 2005, Vol. 122.

73. University of Illinois at Urbana-Champaign. KcsA Potassium Channel. [Online] http://www.ks.uiuc.edu/Research/smd_imd/kcsa/.

74. *Fifty Years of Progress in Ion Channel Research.* Jordan, P.C. 1, 2005, IEEE TRANSACTIONS ON NANOBIOSCIENCE, Vol. 4, pp. 3-9.

75. *A quantitative description of membrane current and its application to conduction and excitation in nerve.* Hodgkin, A. L. and Huxley, A. F. 1952, J. Physiol., Vol. 117, pp. 500-544.

76. *Ion transfer across lipid membranes in the presence of gramicidin a. I. studies of the unit conductance channel.* Hladky, S. B. and Haydon, D. A. 1972, Biochim. Biophys. Acta, Vol. 274, pp. 294-312.

77. *The structure of the potassium channel: Molecular basis of K conduction and selectivity.* Doyle, D. A., et al. 1998, Science, Vol. 280, pp. 69-77.

78. Egwolf, B. and Roux, B. 2010, J. Chem. Phys. 2, Vol. 401, pp. 831-842.

79. *Nuclear Magnetic Resonance Structural Studies of a Potassium Channel-Charybdotoxin Complex.* Yu, L., et al. 48, 2005, Biochemistry, Vol. 44, pp. 15834–15841.

80. *A computational study of ion conductance in the KcsA K+ channel using a Nernst-Planck model with explicit resident ions.* Jung, Y., Lu, B. and Mascagni, M. 2009, J. Chem. Phys., p. 215101.

81. *Molecular Dynamics Simulations of a K Channel Model: Sensitivity to Changes in Ions, Waters, and Membrane Environment.* Capener, C. E. and Sansom, M. S. P. 2001, J. Phys. Chem. B, Vol. 106, pp. 4543-4551.

82. Ponder, J. W. TINKER—software tools for molecular design. [Online] 2004. http://dasher.wustl.edu/ffe/downloads/guide.pdf.

83. Xie, D. and Zhou, S. 853, 2007, BIT Numerical Mathematics, Vol. 47.

84. Kirkwood, J.G. 351, 1934, J. Chem. Phys 2.

85. *Potassium and sodium ions in a potassium channel studied by molecular dynamics simulations.* Biggin, P. C., et al. 2001, Biochim. Biophys. Acta , Vol. 1510, pp. 1-9.

86. *The Cavity and Pore Helices in the KcsA K+ Channel: Electrostatic Stabilization of Monovalent Cations.* Roux, B. and MacKinnon, R. 5424, 1999, Science, Vol. 285, pp. 100-102.

87. *The Structure of the Potassium Channel: Molecular Basis of K+ Conduction and Selectivity.* Doyle, D. A., et al. 1998, Science, Vol. 280, pp. 69-77.

88. Gullingsrud, J and Kosztin, D. Conduction of Ions through Channels. *University of Illinois at Urbana-Champaign.* [Online] 2011. http://www.ks.uiuc.edu/Research/Kchannel/.

APPENDIX A: INSTALLATION INSTRUCTIONS

To install the integrated TINKER and ICSM package, one needs first to download the TINKER package from the website http://dasher.wustl.edu/TINKER/, and extract the package to the directory that it will be run from. Note that all source files of the TINKER package are located in the source subdirectory. Then, one needs to download the ICSM package from http://www.math.uncc.edu/~wcai/TINKER-icsm. After extraction of the zip file, the ICSM package contains several modified TINKER files and four top-level modules: ICSM, FMM, LE, and EXAMPLES.

Next, one needs to copy all of the files and directories of the ICSM package into TINKER's source subdirectory. Finally, one needs to modify the make file makefile, located in the source directory, to include the run path and computer specifications for the system the integrated package will be compiled on. For detailed information on the installation see the manual included with TINKER (82).

As mentioned, several TINKER files were edited. More specifically, the following TINKER files have been modified in order to integrate the ICSM into TINKER.

(a) echarge1.f—added the call to the ICSM to calculate electrostatic interactions.

(b) prmkey.f—added a few lines required to use the ICSM.

(c) mechanic.f—added lines to call the initializing routine for the ICSM.

(d) sizes.i—added parameter settings required to use the ICSM.

(e) potent.i—added parameter settings required to use the ICSM.

(f) Makefile.

The subdirectory ICSM contains the following subroutines for the ICSM.

(a) imgmethod.f—the main call for the ICSM.

(b) fmmle.f—use the FMM and the local expansion to calculate the electrostatic force field in the TO box.

(c) kimgchg.f—initialize variables singular to the ICSM. Also calculate the Gauss–Radau quadrature points and weights $\{s_m, \omega_m\}_{m=1}^{N_i}$

(d) findimg.f—find image charges for charges inside the spherical cavity Γ as shown in Figure **12**.

(e) imgchgs.i—define variables unique to the ICSM.

The subdirectory LE contains the files needed for the special local expansion as pointed out in Section 2.1.1. On the other hand, the subdirectory FMM contains the FMM-Yukawa source files downloaded from the website http://www.fastmultipole.org/ but with a few files being modified and renamed. In particular, the new file ICSMFMMdriver.f serves as the interface between the ICSM and the FMM. The new file ICSMFMMadaplap.f, modified based on and renamed from fmmadaplap.f, now calculates both electrostatic potentials and forces. In addition, parm-alap.h is renamed as fmm.i for naming consistency. The last subdirectory EXAMPLES contains an input file water_30A.xyz and a keyword parameter file water_30A.key needed for a test simulation of liquid water for a 30 Å simulation box, using the integrated TINKER and ICSM package with one image charge and a buffer layer of thickness $\tau = 6$Å.

APPENDIX B: EXAMPLE OF THE KEYFILE USED FOR THE WATER TESTS

```
##############################################################################
##                                                                          ##
##    Test Job on Tinker + Image Charge                                     ##
##                                                                          ##
##############################################################################

parameters          none
#verbose

randomseed           123456789
octahedron
thermostat          Nose-Hoover
vdwtype             LENNARD-JONES
vdw-cutoff           10.0
vdw-list
radiusrule          geometric
radiustype           SIGMA
radiussize          DIAMETER
epsilonrule          geometric
dielectric          80.0

##############################################################################
##                                                                          ##
##         Keywords and Parameters for image charge method
##
##                                                                          ##
## values follows "imgmethod" line:                                         ##
##       Thickness of boundary,                                             ##
##       Number of images for each charge                                   ##
## values follows "fmmle" line:                                             ##
##       whichfmm                                                           ##
##       Number of terms in the local expansions,                           ##
##       Cut-off box type for LE: 1-sphere 2-cube,                          ##
##       Factor kap to define the cut-off sphere                            ##
##                                                                          ##
############################################################## #############

imgmethod            6.0    0
fmmle               2      10     1     2.0

a-axis            30.0000
b-axis            30.0000
c-axis            30.0000
```

```
integrate           verlet
rattle              water
tau-temperature     0.1

##############################################################################
##                                                                          ##
##              Water Parameters
##
##                                                                          ##
##############################################################################

atom        1    O     "O Water (TIP3P)"      8     15.9994     2
atom        2    H     "H Water (TIP3P)"      1      1.008      1
vdw         1          3.15061     0.152072595
vdw         2               0.000     0.000
bond        1    2           529.6     0.9572
angle       2    1    2       34.05     104.52
ureybrad    2    1    2       38.25     1.5139
charge      1              -0.834
charge      2               0.417
```