UNIVERSITY OF CALIFORNIA, SAN DIEGO

An Efficient Algorithm
Combining Cell Multipole and Multigrid Methods
for Rapid Evaluation of Dipole Iteration
in Polarizable Force Fields

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Chemistry

by

Thuy Linh Dinh-Truong

Committee in charge:

J. Andrew McCammon, Chair
Gary Huber, Co-chair
Michael Holst
Judy Kim
Elizabeth Komives
Katja Lindenberg

2007

The dissertation of Thuy Linh Dinh-Truong is approved, and it is acceptable in quality and form for publication on microfilm:

_____

_____

_____

_____

_____
Co-chair

_____
Chair

University of California, San Diego

2007

To my wonderful husband, Christopher, for all of his love & support.

To my parents who value my education above all else.

To my brother for his belief in me.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

This thesis would not have been possible without the support of so many people who have helped me in so many ways.

In particular, I would like to thank my advisors, Gary Huber and Andrew McCammon, for their support, guidance, patience, and continual understanding throughout this journey. Thank you for all of the valuable discussions, advice, and giving me the freedom to explore new research directions. I am also very grateful for your understanding after my car accident. Very few people would show so much kindness and patience during my long recovery.

I would also like to thank all of my committee members, Katja Lindenberg, Michael Holst, Judy Kim, and Elizabeth Komives who have given me much guidance, encouragement and insightful feedback.

I would like to thank my husband, Christopher, who is my everlasting strength and support. He is always there for me, even through my frustrations, freak-outs and total craziness. His constant encouragement and love helped me survive graduate school.

I would like to thank my parents and my brother for their love and encouragement. Thank you for always believing in me and supporting me in whatever I do.

I would like to also thank Stephen Schvaneveldt, who pointed me in the

right direction and encouraged me to pursue graduate studies.

I would also like to thank so many others including Don Steiger, Peter Taylor, Vanda Glezakou, Joakim Persson, Abhik Ghosh, Will Polik, Kim Baldridge, and Laura Brovold, who all bestowed valuable advice and guidance, taught me so many things that a book couldn't, and put up with all of my stupid questions in the past.

Sections 2.2.1, 3.3.1, and 4.1 are largely derived from the work published as "An Efficient Algorithm for Polarizable Interactions: A Uniformly Distributed One-Dimensional Case"[289] which is Copyright © Springer Netherlands.

I feel very fortunate to have met so many brilliant minds!

Thank you so much for this wonderful opportunity.

VITA

2000	Bachelor of Arts, Whitman College
2000-2001	Teaching Assistant, University of California, San Diego
2002	Master of Science, University of California, San Diego
2002-2006	Research Assistant, University of California, San Diego
2007	Doctor of Philosophy, University of California, San Diego

ABSTRACT OF THE DISSERTATION

An Efficient Algorithm

Combining Cell Multipole and Multigrid Methods

for Rapid Evaluation of Dipole Iteration

in Polarizable Force Fields

by

Thuy Linh Dinh-Truong

Doctor of Philosophy in Chemistry

University of California San Diego, 2007

J. Andrew McCammon, Chair

Gary Huber, Co-chair

There has been continuing effort to develop polarizable force fields for computational studies of biological systems. Applications of polarizable models in molecular dynamics simulations include liquid water, ionic systems, alcohols, solvated proteins, interfacial systems and membrane systems. An overview of the advances in development of these polarizable force fields to date is presented. Re-

cent studies have shown that the dynamic response to inhomogeneous environment represented by the explicit inclusion of polarization is necessary for more realistic descriptions of biosystems. Explicitly including polarization effects in force fields requires self-consistent iteration to evaluate induced dipole moments. However, the demanding computational cost using traditional solvers limits the system sizes that can be fully described with explicit polarization. To make this calculation more tractable for large-scale systems, an efficient method for computation of polarizable interactions is needed.

An algorithm combining hierarchical cell multipole (CMM) and multigrid (MG) schemes is developed for fast computation of these interactions, using polarizable point dipoles. This scheme separates polarizable interactions into direct and indirect components, where we derived the CMM electric field terms for dipolar systems to handle long-range interactions. A fast multigrid solver is applied to further increase computational efficiency in solving these induced dipolar calculations. Performance of various iterative solvers, Jacobi, Gauss-Seidel, successive over-relaxation, conjugate gradient, and our newly developed multigrid-multipole (MG-CMM) solver are compared for test cases of varying system sizes to demonstrate the efficiency of this algorithm for a uniform distribution. The MG-CMM algorithm achieves fast convergence with reasonable accuracy. A matrix version of the cell multipole method is derived and extended to include polarizable dipoles.

In order to extend MG-CMM to treat non-uniform distributions, we have casted the cell multipole method in matrix form and introduce an algebraic multigrid and matrix-based cell multipole (AMG-CMMm) scheme to reduce the number of iterations to self-consistency. For further speedup, AMG-CMMm can be parallelized and the sparse matrix storage can be optimized. An efficient implementation of this technique will significantly reduce the number of dipole iterations for large polarizable systems and help enhance the ability of force field methods to accurately describe biomolecular processes.

Chapter 1

Introduction

Polarization is the change in charge distribution due to an external electric field, namely the environment surrounding a given molecule.[1] Conventional non-polarizable force fields account for electrostatic polarization implicitly, describing induction in an average way with enhanced charges.[2, 3, 4] These fixed atomic charges are parameterized to reproduce molecular dipole moments that are roughly 10-20% larger than those observed in gas phase to compensate for the missing induction.[5, 6, 7, 8, 9] Treatment of polarization in an average way does not allow models to dynamically respond to a wide range of conditions within their environment. For example, non-polarizable models cannot simulate the variations in dielectric behavior in interfacial regions such as the liquid-vapor interface of water[10] or the area between proteins and solvent.[11] However, polarizable

models can capture the change in dipole moments seen by a molecule crossing from gas phase to liquid phase.[10, 12, 13] This response to environment is important in inhomogeneous systems such as ion channels where the interactions between proteins, lipids, ions and water within the channel environment are different from those in bulk environment.[14] Recent studies have shown that polarization effects are important in modeling structure and energetics of many processes involving ion solvation,[15, 16, 17, 18, 19, 20, 21, 22] hydrogen-bond formation,[23, 24, 25, 26] protein folding,[65] enzyme catalysis,[27] and ion permeation.[28, 29, 30] In particular, the proper treatment of these many-body effects are necessary in modeling highly polar interactions[20, 31, 32, 33, 34] and cation-π interactions,[35, 36, 37] due to the substantial polarization contributions to the total energy. By explicitly incorporating polarization effects, molecular mechanics force fields will have the added flexibility needed to respond to a wide range of surroundings, making them more realistic and transferable to different systems.[38]

The next section presents an overview of current models commonly used in molecular dynamics simulations to account for polarization effects using point dipoles, fluctuating charges, or classical Drude oscillators.

1.1 Polarization Models

Common approaches to explicitly include polarization are the polarizable point dipole model, classical Drude oscillator model, and fluctuating charge model.

1.1.1 Polarizable Point Dipole Model

Since the early seventies, the most widely used method to account for polarization is the polarizable point dipole (PD) model, also sometimes referred to as the atom dipole interaction model.[41, 42, 43, 44, 45, 46, 47, 15, 16, 48, 49] This model has been applied to ions,[50, 51, 20, 55, 52] liquid water,[53, 54, 55, 56] liquid-vapor interfaces,[10] surface tension studies,[57] proteins,[58, 59, 60, 61, 62] and DNA.[63] In the PD model, polarization is treated by the interaction of atomic induced dipole moments created by permanent charges and inducible dipoles. An induced dipole moment $\vec{\mu}_i$ on an atom i is given by:

$$\begin{aligned}
\vec{\mu}_{ind,i} &= \alpha_i \vec{E}_i \\
&= \alpha_i (\vec{E}_{i,perm} + \vec{E}_{i,ind}) \\
&= \alpha_i [\underbrace{\sum_{j=1;j\neq i} \frac{q_j \vec{r}_{ij}}{r_{ij}^3}}_{\vec{E}_{i,perm}} + \underbrace{\sum_{j=1;j\neq i} \mathbf{T}_{ij} \vec{\mu}_{j,ind}}_{\vec{E}_{i,ind}}]
\end{aligned} \tag{1.1}$$

where α_i is the isotropic atomic polarizability and $\vec{E}_i$ is the total electric field at atom i. The total electric field is composed of the electric field produced by

the permanent charges, $\vec{E}_{i,perm}$, and the electric field due to the other surrounding induced dipoles, $\vec{E}_{i,ind}$. The interaction between induced dipoles is described by the dipole tensor:

$$\mathbf{T}_{ij} = \frac{1}{r_{ij}^3}\left(\frac{3\vec{r}_{ij}\vec{r}_{ij}^{\top}}{r_{ij}^2} - \mathbf{I}\right) \tag{1.2}$$

where r_{ij} is the distance between of atoms i and j and $\mathbf{I}$ is the identity matrix.

The energy of the induced dipoles,[69, 23, 38] U_{ind}, is given by:

$$U_{ind} = U_{stat} + U_{\mu\mu} + U_{pol} \tag{1.3}$$

where U_{stat} is the energy of the induced dipoles in the field of permanent charges (charge-dipole interaction), $U_{\mu\mu}$ is the energy of the induced dipoles in the field due to other induced dipoles (dipole-dipole interaction), and U_{pol} is the polarization energy cost to distort the electron distributions of the atoms and generate induced dipoles. Components of U_{ind} can be expressed as:[70, 29, 38]

$$\begin{aligned} U_{stat} &= -\sum_{i=1}^{n} \vec{\mu}_i \cdot \vec{E}_{i,perm} \\ U_{\mu\mu} &= -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1;j\neq 1}^{n} \vec{\mu}_i \cdot \mathbf{T}_{ij} \cdot \vec{\mu}_j \\ U_{pol} &= \frac{1}{2}\sum_{i=1}^{n} \vec{\mu}_i \cdot \vec{E}_i = \frac{1}{2}\sum_{i=1}^{n} \vec{\mu}_i \cdot \alpha^{-1} \cdot \vec{\mu}_j = \sum_{i=1}^{n} \frac{\vec{\mu}_i^{\top}\vec{\mu}_i}{2\alpha_i} \end{aligned} \tag{1.4}$$

Combining all of the above terms for the energy of induced dipoles gives:

$$U_{ind} = -\sum_{i}^{n} \vec{\mu}_i \cdot \vec{E}_{i,perm} - \frac{1}{2}\sum_{i}^{n}\sum_{j\neq i} \vec{\mu}_i \cdot \mathbf{T}_{ij} \cdot \vec{\mu}_j + \frac{1}{2}\sum_{i}^{n} \vec{\mu}_i \cdot \vec{E}_i$$

This equation can be simplified to the form commonly seen in literature by making the following substitution ($E_i = E_{i,perm} + \sum_{j\neq i} \mathbf{T}_{ij} \cdot \mu_j$):

$$\begin{aligned} U_{ind} &= -\sum_{i}^{n} \vec{\mu}_i \cdot \vec{E}_{i,perm} - \frac{1}{2}\sum_{i}^{n}\sum_{j\neq i} \vec{\mu}_i \cdot \mathbf{T}_{ij} \cdot \vec{\mu}_j + \frac{1}{2}\sum_{i}^{n} \vec{\mu}_i \cdot (\vec{E}_{i,perm} + \sum_{j\neq i} \mathbf{T}_{ij} \cdot \vec{\mu}_j) \\ &= -\sum_{i}^{n} \vec{\mu}_i \cdot \vec{E}_{i,perm} - \frac{1}{2}\sum_{i}^{n}\sum_{j\neq i} \vec{\mu}_i \cdot \mathbf{T}_{ij} \cdot \vec{\mu}_j \\ &\quad + \frac{1}{2}\sum_{i}^{n} \vec{\mu}_i \cdot \vec{E}_{i,perm} + \frac{1}{2}\sum_{i}^{n}\sum_{j\neq i} \vec{\mu}_i \cdot \mathbf{T}_{ij} \cdot \vec{\mu}_j \\ &= -\frac{1}{2}\sum_{i}^{n} \vec{\mu}_i \cdot E_{i,perm} \end{aligned} \tag{1.5}$$

Therefore, if the induced dipoles $\vec{\mu}_i$ are known, the energetics of the system of permanent charges and polarizable dipoles can be computed.

Given a system of atoms, equation (1.1) can be rearranged to form a matrix equation for the computation of induced dipoles:

$$\vec{\mu}_{ind} = \mathbf{M}\vec{q} + \mathbf{N}\vec{\mu}_{ind} \tag{1.6}$$

where $\vec{q}$ and $\vec{\mu}_{ind}$ contain the charges and induced dipoles of the atoms, and the sub-blocks for $\mathbf{M}$ and $\mathbf{N}$ are:

$$\mathbf{M}_{ij} = \begin{cases} \frac{\alpha_i \vec{r}_{ij}}{r_{ij}^3}, & \text{if } r_i \neq r_j \\ 0, & \text{if } r_i = r_j \end{cases} \tag{1.7}$$

and

$$\mathbf{N}_{ij} = \begin{cases} \alpha_i \left(\frac{3\vec{r}_{ij}\vec{r}_{ij}^{\top}}{r_{ij}^5} - \frac{\mathbf{I}}{r_{ij}^3} \right), & \text{if } r_i \neq r_j \\ 0, & \text{if } r_i = r_j \end{cases} \tag{1.8}$$

This system of equations can be solved either by self-consistent iteration[43, 44] or direct matrix inversion,[69] both of which are computationally expensive. Matrix inversion of an $n \times n$ matrix gives the exact solution with the computational cost scaling as $O(n^3)$. Due to this expensive calculation, matrix inversion must be run in parallel to be feasible for large systems.[69, 71] With current computational resources, iterative solvers have been the common approach for computing induced dipoles.[72, 73, 10, 74]

This iterative process begins with an initial guess for induced dipoles, which is generally set to equal to $\vec{\mu} = \alpha \vec{E}_{perm}$ or the induced dipoles from the previous time step from the molecular dynamics simulation can be used.[43, 44] New induced dipoles are then refined by adding the electric field due to these approximate induced dipoles to the electric field from the permanent charges. This computation continues until the induced dipoles converge.

A shortcoming found in this model is that if two inducible dipoles come too close to each other, their mutual dipolar interaction approaches infinity. This model does not contain damping effects for close-range interactions to prevent this "polarization catastrophe."[41, 46] This is due to point charges being a poor approximation for electron distributions at small distances.[41, 46, 68, 69] This limitation

is generally eliminated by using screening functions to dampen the electric field from the induced dipoles at short distances which more closely imitates the electron clouds within a molecule.[46, 75, 76] These screening functions have been incorporated into some polarizable models,[53, 77, 54, 16, 18, 78, 72, 79, 80, 10, 56, 69, 81, 83] while others do not include any screening function.[50, 51, 48, 20, 34, 55, 13]

1.1.2 Classical Drude Oscillator Model

A variant of the polarizable point dipole model is the classical Drude oscillator (DO),[84, 92, 86] also known as the charge-on-spring[89] or shell model,[32] which utilizes dipoles of finite length to describe polarization. This scheme manipulates the geometry of the charge distribution to induce dipole moments.[70] The induced dipole is represented by a fixed core charge at a given atomic site connected to a free floating massless opposite charge by a harmonic spring. The movement of the Drude charge creates an induced dipole which is defined as:

$$\vec{\mu}_i = \alpha_i \vec{E}_i = -q_i \vec{d}_i \tag{1.9}$$

where α_i is the polarizability of the isotropic Drude atom, $\vec{d}_i$ is the vector pointing from the fixed charge q_i to its floating Drude charge, and $\vec{E}_i$ is the total electric field. By creating induced dipoles, the atomic charge is redistributed between the fixed charge and the floating charge. The harmonic spring constant k_i is related

to the charge of the Drude atom and the polarizability:[4, 84]

$$k_i = \frac{q_{i,Drude}^2}{\alpha_i}. \tag{1.10}$$

The U_{ind} for the classical Drude approach ($U_{ind} = U_{stat} + U_{\mu\mu} + U_{pol}$) includes the interaction between induced dipoles and permanent charges U_{stat}, the interaction between induced dipoles with other dipoles $U_{\mu\mu}$, and the polarization energy U_{pol} from the harmonic spring separating the two charges which is defined by the following equations:[38]

$$\begin{aligned} U_{stat} &= -\sum_{i=1}^{n} q_i[\vec{r_i} \cdot \vec{E}_{i,core} - (\vec{r_i} + \vec{d_i}) \cdot \vec{E}_{i,Drude}] \\ U_{\mu\mu} &= \frac{1}{2}\sum_{i=1}^{n}\sum_{j\neq i} q_i q_j \left[\frac{1}{|r_{ij}|} - \frac{1}{|r_{ij} - \vec{d_j}|} - \frac{1}{|r_{ij} + \vec{d_i}|} + \frac{1}{|r_{ij} - \vec{d_j} + \vec{d_i}|}\right] \\ U_{pol} &= \frac{1}{2}\sum_{i=1}^{n} k_i d_i^2 \end{aligned} \tag{1.11}$$

where $\vec{E}_{i,core}$ is electric field at the core charge, $\vec{r_i}$, and $\vec{E}_{i,Drude}$ is the field at the Drude charge, $\vec{r_i} + \vec{d_i}$. Notice that the DO model avoids the dipole-dipole calculation found in the PD model, by replacing this interaction with a sum of charge-charge interactions corresponding to core-core, Drude-Drude, and Drude-core charge contributions. This advantage has recently sparked interest in applying polarization based on this Drude oscillator formalism to simulations of liquid water,[87, 88, 89, 90, 4] ionic systems,[32, 91, 93] alkanes,[94] and liquid-alcohol mixtures.[95]

1.1.3 Fluctuating Charge Model

An alternative method to account for polarization is the fluctuating charge (FQ) model[96, 32, 97, 98, 99, 100, 11, 101] which simulates polarizability by allowing the atomic charges to fluctuate with environment according to the principle of electronegativity equalization.[110] This has been a popular method due to its low computational cost and has been used to study various ionic systems,[32, 102] liquid water,[96, 103, 104, 105] organic liquids,[106, 100, 107] alkanes,[108] and proteins.[11, 109] The efficiency of this model stems from the charge-charge interactions where the system dynamically responds to its environment by distributing charge between atoms in the system until the electronegativities of the atoms are all equal.[96] The energy of the charges is:[38]

$$U_q = \sum_i (U_i^0 + \chi_i^0 q_i + \frac{1}{2} J_{ii} q_i^2) + \sum_i \sum_{j>i} J_{ij}(r_{ij}) q_i q_j \tag{1.12}$$

where U_i^0 is the ground state energy of atom i, q_i is the partial charge on atom i, χ_i^0 is Mulliken's definition of electronegativity,[111] J_{ii} is the "hardness" of the atom,[112] and $J_{ij}(r_{ij})$ coefficient is dependent on the distance between two atoms, generally being equal to $1/r_{ij}$. To find the partial charges on each atom, the energy of the system is minimized by moving charges between atomic sites within a molecule, while keeping the total charge and configuration conserved.[96]

1.1.4 Comparison of Polarizable Models

We have reviewed three different ways to account for polarization. The fluctuating charge model describes the interaction between polarizable atoms by only considering charge-charge interactions, whereas the classical Drude oscillator model uses a sum of four charge-charge interactions. The polarizable point dipole model considers dipole-dipole interactions that greatly increase the complexity of code implementation.[38] Therefore, the FQ method has the advantage of computational efficiency as its computational cost is only a factor of 1.1 of non-polarizable models.[96] FQ also includes charge transfer from one atomic site to another to model charge redistribution, which is not modeled in either PD or DO models.

In recent work by Masia *et al*, the accuracy of all three polarizable models were examined in the investigation of small molecules.[113, 114] Their studies showed that the PD model had the best performance overall, accurately predicting properties compared to *ab initio* calculations due to its flexibility in description. The DO model yielded adequate results, and the FQ model gave poorer results. These results can stem from the limitation of the description of the polarizability. The molecular polarizability described by the DO method is limited to being isotropic, even for nonspherical systems as shown by equation (1.10), which is independent of direction of the electric field.[113] In contrast, the PD technique has isotropic atomic polarizabilities, but displays anisotropic molecular polarizabilities.[41] Stud-

ies by Berne and coworkers have shown that the fluctuating charge method cannot describe out-of-plane polarization which has been found inadequate for atomic ions[32] and systems containing cation-π interactions and bifurcated hydrogen bonds important for drug-protein interactions.[98, 115] To eliminate this weakness, hybrid models combining FQ and PD have been developed where point dipoles are added to improve the performance.[98, 116, 117, 118]

Each of these polarizable models have their strengths and weaknesses concerning computational efficiency and accuracy. Striking a balance between efficiency and accuracy is a major factor in choosing the appropriate approach for a given problem as well as considering the transferability and applicability of these methods in different biosystems. These methods have been applied to many studies to explore the polarization effects in biological systems, as is discussed in the following section.

1.2 Advances in the Development of Polarizable Force Fields

Over the last decade, there has been a continuing effort to develop polarizable force fields for computational studies of biological systems. The purpose of this section is to give an overview of the applications of polarizable models

in molecular dynamics simulations ranging from small molecules such as water, alcohols, and other organic functional groups to large membrane proteins.

1.2.1 Polarizable Models for Water

Since modeling water is essential for modeling biological systems, liquid water has been the primary focus of polarizable simulations for many years. To date, commonly used water models are non-polarizable and describe polarization in an average way. Popular non-polarizable water models, such as SPC (single point charge)[8] and TIPnP (transferable interaction potential),[129] are typically parameterized to reproduce bulk liquid properties. These non-polarizable models for liquids are unable to accurately reproduce properties over a wide range of densities and temperature, limiting their transferability to only simulate the environment for which they were parameterized.[80, 119] Extending these water models to explicitly include polarization response to their environment is naturally the next step in improving the quality and flexibility of these models. Including explicit polarization should allow the models to perform well in varying environments and conditions such as the liquid-vapor interface and surface solvation.[10] Due to the increased interest in polarization effects, many water models have been developed to validate the various polarization approaches by reproducing structural and dynamic properties. Polarizable water models are based on the SPC or TIPnP geometry and

account for polarization using polarizable point dipoles (PD), fluctuating charges (FQ), or classical Drude oscillators (DO).

Polarizable water models that explicitly include polarization effects date back to the 1970s with polarization being represented by partial charges and inducible point dipoles.[44, 53, 77, 120] Some models include atom-centered polarizabilities.[50, 15, 34] Other approaches place a single molecular polarizability site on the oxygen atom or along the bisector of the H-O-H angle.[20, 18] Caldwell *et al* found that including atomic polarizabilities on all atoms rather than a single center molecular polarizability leads to an improvement in modeling liquid water and water-ion clusters,[20] although it significantly increases the computational expense with the additional number of interaction sites. RPOL, a 3-site polarizable dipole water model, reproduced bulk water properties that are comparable to *ab initio* calculations and to experiment, but was not fully optimized for clusters and interfaces.[34, 121, 13] The addition of a fourth site and reparameterization led to better performance in a wide range of different environments including cluster, liquid and liquid/vapor interfacial regions. This revised potential is generally referred to as the Dang-Chang (DC) polarizable water model (Dang97). Two other recently developed water models are the Brodholt-Sampoli-Valluri (BSV)[79, 122, 123] and "Thole-type" model (TTM2).[124] Both of these models are also based on the TIP4P geometry like DC. TTM2 includes Thole's polarization damping scheme

to simulate smearing charge density for a better description of the molecular charge cloud.[74] To further improve the description of molecular properties, particularly dipoles and quadrupoles, Ren and Ponder's water model based on inducible dipoles, AMOEBA, uses atom-centered polarizable point multipoles in addition to monopole charges on atoms.[125] This greatly improves the description of polarization as shown by the excellent performance in yielding cluster and liquid phase properties.[125] However, including these atomic multipoles significantly increases the complexity and cost of the calculations.

In order to avoid computationally expensive dipole-dipole calculations and for easier incorporation into current force fields, polarizable water models based on Drude oscillators have been developed. All the interactions in these models are point charge interactions, rather than dipoles. This formulation maintains the simplicity of charge-charge calculations, but utilizes pairs of point charges separated by harmonic springs to describe polarization. Sprik and Klein modified the TIP4P water model using Drude oscillator formalism to reproduce water structure and binding energetics.[54] More recent water models based on Drude oscillators include the Charge-on-Spring (COS) model by Yu *et al*,[90] the simple water model (4-site) with Drude polarizability (SWM4-DP) developed by Lamoureux and coworkers[4], and shell water models (SW-FLEX).[88]

Another alternative to computing dipole interactions is accounting for po-

larization with fluctuating charges based on the electronegativity equalization principle.[110] Recently, fluctuating charges have been incorporated into two of the most commonly used water geometries to create SPC-FQ and TIP4P-FQ polarizable water models.[96] These fluctuating charge models proved to perform remarkably well for predicting bulk liquid properties and the aqueous solvation of ions and small molecules such as amides. These methods are highly efficient, only increasing the computational cost by a factor of 1.1 times compared to non-polarizable models.[96] However, this point charge model is limited to describing polarization on the molecular plane where charges are confined to the atomic positions, which proves to be inadequate in treating bifurcated hydrogen bonds and aromatic-π interactions.[96] This shortcoming can be eliminated by incorporating polarizable dipoles to form hybrid models that can describe both polarization and charge transfer.[117] This combined model has been applied to a 5-site model of water, POL5, that has been reported to perform reasonably well away from ambient conditions, reproducing a wide range of structural and thermodynamic properties.[118]

With all of the polarizable water models developed, it is important to determine the relative accuracy of these models. Table 1.1 contains a summary of reported liquid-state properties computed from current non-polarizable and polarizable water models compared to experimental data at ambient conditions. We compare various properties of liquid water to assess whether commonly used po-

larizable and non-polarizable models can provide a reasonable representation of water. At room temperature, the density of water is 0.997 g/cm^3 under 1 atm pressure.[138] Overall, the polarizable and non-polarizable water models shown in Table 1.1 produce densities and potential energies of the liquid in ambient conditions are in reasonable agreement with experiment.

Examination of computed radial distribution functions (RDFs), $g_{OO}(r)$, $g_{OH}(r)$, and $g_{HH}(r)$, provide insight into the liquid structure of water. A comparative study of RDFs predicted by various polarizable and non-polarizable models was done by Sorenson and coworkers.[127] They found that polarizable water models generally provided a better description of ambient water structure compared to non-polarizable models. An exception to this is TIP5P, which yielded the closest radial distribution functions compared to experimental RDFs completed by Sorenson *et al.*[127] The non-polarizable TIP5P water model includes off-center charges to better describe the directionality of hydrogen-bonding.[5, 126] Since the non-polarizable 5-site water model yielded the most accurate liquid structure, it is apparent that other water potentials can be improved by considering lone pairs or off-center charges. A general problem of all polarizable and non-polarizable water models is that the first peak is predicted to be much higher than that of experiment.[127, 122, 4, 124, 90] However, due to the various methods that can be used to extract RDFs from measured diffraction data, there is a large margin of exper-

imental uncertainty in the peak height.[128] The second and third peaks predicted by most water models are also in fairly good agreement with experiment, with the exception of the 3-site non-polarizable water models, SPC and TIP3P, that give very little structure beyond the first peak with unpronounced peaks showing very little definition.[129] It is observed that polarizable water models such as BSV, CC, and DC can show peak shifts to larger distances with increasing temperature and decreasing density that non-polarizable models cannot capture.[122]

There has been some debate on the true value of the liquid-state molecular dipole moment, as it is impossible to directly measure the dipole moment of an individual water molecule in condensed phases.[130, 118] Unfortunately, *ab initio* calculations yield a wide range of values varying from 2.4-3.0 D for the average dipole moment of liquid water depending on the partitioning of the molecular charge density.[131, 38] *Ab initio* calculations by Silvestrelli and Parinello predicted dipole moments ranging from 2.95-3.00 D.[133] Gubskaya and Kusalik reported the dipole moment of liquid water at ambient conditions to be 2.95 D which was extracted from the experimental refractive index data.[131] This is also in close agreement with results obtained by Batista *et al* from *ab initio* calculations.[132] From Table 1.1, the non-polarizable water models significantly underestimate the dipole moment, giving values ranging from 2.1-2.4 D, while polarizable potentials yield dipole moments within the range (2.5-2.8 D). Studies have shown that water model must

exhibit a dipole moment within the range of 2.4-2.6 D in order to reproduce the experimental value of the dielectric constant ϵ.[135] Subsequently, non-polarizable models also generally underestimate ϵ while polarizable models tend to overestimate ϵ, showing some correlation between the magnitude of the dipole moment with the dielectric constant.[122, 136] However, this guideline does not always prove true, as the computed dipole moment from the RPOL water model[34] falls within the given range, but still fails at predicting the dielectric constant. In addition, the AMOEBA water model[125] yields a larger dipole moment outside of this given range, but predicts a dielectric constant close to the experimental value of 78.0.[125, 3] TIP4P-FQ,[96] PPC[141] and SWM4-DP[4] accurately predict water's dielectric constant as well as yielding a dipole moment within the range of 2.4-2.6 D. Due to the general trend of polarizable models overestimating ϵ, it has been suggested by Alfredsson *et al* that the gas phase polarizabilities used in most methods are not appropriate for predicting liquid phase properties.[161] Rather, the polarizabilities should be reparameterized for the liquid phase. Because fluctuating charge models are successful in reproducing the dielectric environment, it has been proposed by Guillot that charge transfer needs to be included in the model for a more accurate description.[130] It is also speculated that most models do not predict the correct dipole moment and dielectric constant due to inaccurate description of molecular quadrupoles.[4] AMOEBA includes high-order atomic multipoles in their

model to improve the description of quadrupole moments.[125] To go beyond the point charge representation, diffuse charges can be used to properly simulate the electron cloud character. This is being explored by Jeon and coworkers with the Polarflex flexible (PFG) water model,[159] as well as Paricaud *et al* with polarizable smeared charges in the Gaussian charge polarizable model (GCPM).[160] The newly developed water potential, Gaussian smeared charge model, of Paricaud *et al* has demonstrated excellent modeling of phase equilibrium properties over a wide range of temperatures. It is beneficial to consider polarization methods because they can qualitatively describe the dielectric environment. These polarizable models can simulate the change in environment by capturing the change in water dipole moments near the liquid/vapor interface that non-polarizable methods cannot model. It is shown by Dang and Chang that far from the interface, the average dipole moments of these water molecules are about 2.75 D, reaching values of bulk solution. Approaching the interface, the dipole moments become smaller and approach the gas-phase value of 1.85 D, due to the change in the electric field induced by surrounding water molecules.[10]

The self-diffusion coefficient is a measure of the dynamic movement of water molecules. A number of water models, including BSV,[79] COS/G2,[90] and SWM4-DP,[4] yield values closest to the experimental value[139] of 2.3 x 10^{-5} cm^2/s while SPC/E,[144] RPOL,[121] TIP4P-FQ,[96] DC,[10] TTM2,[286] PFG,[159] AMOEBA,[125]

TIP4P/2005,[287] and GCPM[160] give reasonable self-diffusion coefficients. SPC-FQ[96] and POL5 underestimates the diffusivity by more than 0.4 cm^2/s. These water models (SPC[96], TIP3P[144], TIP4P[96] and SW-FLEX[88]) have much higher diffusion coefficients, resulting in faster mobility than real water. In general, the diffusion coefficients from polarizable water models are smaller than the non-polarizable water models, resulting in values much closer to experiment.

It is also important to examine the predictive power of these models in showing the temperature dependence of water properties. After all, the primary goal of any of these models is to be transferable, with the ability to accurately predict the variation of different properties over a large range of changing temperatures and densities well beyond ambient conditions. As the temperature increases to 277K, water reaches its maximum density of $0.99995 g/cm^3$.[90] Available data from reported simulation results for the temperature at maximum density are compared to experiment in Table 1.1. Recent water models, including PPC,[141] BSV,[122] TIP4P-FQ,[143] and TIP5P[145] have been shown to have a density maximum close to 277 K. Non-polarizable water models, SPC,[146] TIP3P,[146] TIP4P,[146] and SPC/E[147] fail to reproduce the density maximum, predicting the density maximum at temperatures well below experimental value by at least 30K, while POL5[118] overestimates the temperature at maximum density by about 15K. A general trend observed in most polarizable models, including those that yield accurate T_{MD}, is

that the densities decrease too quickly with increasing temperature compared to experimental data.[141, 122, 148, 123, 130, 90, 149] This excessively strong dependence on temperature, in turn, results in poor reproduction of the vapor-liquid coexistence properties.[150, 104] Reported values in literature for these critical properties from various models are given in Table 1.2. All of these reported water models underestimate the critical temperature, T_C, showing that polarizable models do not perform better than non-polarizable models in the case of phase transferability. These vapor-liquid equilibrium properties have been shown to be highly sensitive to changes in parameters by Kiyohara *et al.*[150, 152] Chen *et al* implemented various parameterizations and observed that either the structure or vapor-liquid equilibrium properties can be simulated accurately using current polarizable models, but not simultaneously.[104] It has been suggested by Jedlovszky and Richardi that reparameterization to fit these properties would greatly improve performance under varying conditions.[122] Patel and Brooks have proposed that variation in molecular polarizabilities across different phases is needed for improvement, as the molecular polarizability of water in condensed phase is 18% lower than in gas phase.[109, 155, 156]

Although most of the research has mainly focused on polarizable water models, there has been a move towards including explicit polarization to study more complicated systems beyond water such as solvated ions and interfaces, small molecules, proteins and DNA/RNA. Ongoing effort has extended into the parame-

terization of these polarizable force fields using small molecules that possess similar characteristics to protein backbone and side-chain groups for more accurate calculations of structural and dynamic properties of biomolecular systems under a wide range of conditions.

1.2.2 Studies of Solvated Ions, Interfaces, Ion Channels

The interactions in ion solvation have received considerable attention because of the important roles of ions as signaling molecules as well as their transport throughout proteins and lipid membranes. An early study by Lybrand and Kollman focused on the hydration of enthalpy for several ions.[50] An investigation into the solvation of sodium and chloride ions in water clusters was done by Berkowitz and coworkers.[162, 163, 164, 165] They found that polarization effects are an important factor in the reproduction of the correct ion-water cluster structure. Dang *et al* employed polarizable dipole models to study the solvation of various ions including lithium, sodium, chloride and fluoride ions in water clusters.[19, 34] They found that both fluoride and chloride ions are solvated near the surface of the water cluster. Stuart *et al* used the TIP4P-FQ[96] water model in conjunction with a polarizable Drude chloride ion to describe the hydration of chloride ions.[32, 102] They also observed that the chloride ion preferred to remain at the water cluster which was not shown with non-polarizable models. Grossfield *et al* computed solvation free

energies of chloride, potassium and sodium ions in liquid water and formamide using the AMOEBA[198] polarizable force field, giving values that suggested the solvation process in water is much more favorable for anions than cations.[31, 166] Comparison between the energetics and entropic calculations of ion-water clusters from polarizable and non-polarizable models show that surface solvation is due to polarization, ion size, and charge.[167] Continuing studies on the solvation behavior of ions are currently being explored, including work by Piquemal and coworkers[52] using AMOEBA for calcium and magnesium divalent cations in water and Lamoureux *et al* using SWM4-DP for the hydration of alkali and halide ions.[93]

The next step is to extend ion solvation models to describe the behavior of ions across liquid/vapor interface systems.[168, 169, 171, 172, 173, 174, 175] Archontis *et al* applied the polarizable force field based on classical Drude oscillators, SWM4-DP for the solvation of sodium and iodide ions near the water surface (air-water interface).[91, 30] It was observed that the iodide ions prefer surface solvation, while sodium ions formed an adjacent, interior layer. Further study of salt solutions, acids and bases at the liquid-vapor interface showed that heavy halides and hydroxides had an affinity towards the surface, while smaller halides and cations were repelled from the interface.[176, 169, 177, 178, 179, 180, 181, 182, 183] Examination of the free energies of solvation led to the understanding that large polarizable ions prefer the water surface due to the free energy gain when moving towards the interface as

well as an increased polarization, providing stronger water-ion interactions.[30] Using polarizable models for extensive studies in ion-specificity and ion transport across the liquid-vapor interfaces contribute to further understanding of the mechanisms in membrane ion channels. These techniques allow one to differentiate between the response of water molecules at the protein-solvent interface and that of other water molecules in bulk solution.

Recent studies in biological ion channels such as the gramicidin A (gA) channel have suggested that explicit polarization is needed to properly describe the heterogenous environment encountered by the ion during transport and reproduce the energetics accurately.[40, 184, 185, 186, 187] The gA dimer in a dimyristoylphosphatidylcholine (DMPC) bilayer forms a narrow potassium ion channel that rejects anions and binds to divalent cations.[187] Particularly, non-polarizable models cannot show that the water molecules closest to the ion are more polarized due to the electric field of the ion.[188] As with the water dipole moments, the backbone carbonyls on the protein do not show the induced polarization from the ion in non-polarizable models. Patel and coworkers have applied their polarizable protein force field[11] with polarizable TIP4P-FQ water[96] to study this hydrated protein system, while the membrane itself is treated with the standard non-polarizable CHARMM force field.[109] In examining the different regions of the membrane, they reported that a change in the average molecular dipole moment of water moving from the interior

of the bilayer towards the bulk solution is observed. The water dipole moments approach the gas phase values in the lipid bilayer interior and increase to bulk values away from the bilayer. The distribution of water dipole moments are also different within the ion channel compared to the bulk solution, reflecting the change in environment. Due to strong polarization between the gA dimer and water dipoles inside the ion channel, the water density profiles showed slower dynamics and the dipole moment distributions displayed a shift towards higher water dipole moments. Inspection of the dipole moments along the protein revealed a variation in polarization between different residues with water molecules, demonstrating that the polarizable force field is successful in describing the electrostatic interactions with a heterogeneous environment.

1.2.3 Studies of Liquid Alcohols

Modeling liquid alcohols such as methanol and ethanol provide understanding of hydrogen bonding behavior. These compounds possess amphiphilic character which is important in studies of hydroxyl groups in amino acid side chains such as serine, threonine, and tyrosine. One of the earliest contributions was made by Caldwell and Kollman who developed a polarizable methanol model using polarizable dipoles to account for the induction effects.[55] Gao *et al* studied the polarization effects on a series of liquid alcohols including methanol, ethanol

and propanol, observing that induction contributed about 10-20% to the total energy accompanied by a substantial increase in molecular dipole moment.[189] It has been demonstrated that the inclusion of explicit polarization is needed to capture the change in dipole moment in interfacial systems, where the computed dipole moments of alcohols approach gas phase values near the interface, and approach bulk values far from the interface.[172, 190] Dang and Chang's polarizable model for liquid methanol predicted values for structural properties, diffusion coefficient and surface tension properties in close agreement with experimental data.[172] Polarizable models can also reproduce the experimental dielectric constant of methanol, whereas most non-polarizable models have failed to do so.[95, 190] Additionally, polarizable models can capture the degree of hydrogen-bonding shown in the bimodal shape of the molecular dipole distribution of methanol.[175, 107] The small shoulder of the distribution represents methanol molecules that do not accept hydrogen bonds. Additional studies in liquid alcohols using polarizable models have examined different phases,[191] pressure effects on structural and dynamical properties,[175] computation of acidity constants,[192] alcohol-water mixtures,[193, 95] and liquid-vapor equilibrium properties.[107] Several groups have been involved in the development of polarizable force fields for liquid alcohols including Patel *et al*[107] using fluctuating charges to account for induction effects, Noskov and coworkers[193] using classical Drude oscillators, the COS model by Yu *et al*,[95] and the polarizable dipole DC

model by Dang and coworkers.[172, 175]

1.2.4 Studies of Alkanes

A limited number of computational studies focusing on alkane compounds using polarizable models have been reported in literature.[84, 94, 108] Vorobyov and coworkers have developed a polarizable alkane model based on classical Drude oscillators.[94] Parameterization of this alkane force field involved the use of model compounds that could be transferable for long chain alkanes, important for accurate treatment of lipid bilayers. Transferability of these parameters was tested on simulations of heptane and decane. Computed properties are in reasonable agreement with experiment, with the exception of the self-diffusion constants that underestimate experimental values by about 5-10%. Patel *et al* studied hexane-water interfacial properties using a polarizable force field based on fluctuating charges.[108] They found that explicitly including polarization in computing the self-diffusion constant improved the agreement with experiment. This polarizable hexane model also properly describes the dielectric environment and shows the shift in dipole moment for the change from gas to condensed phase.

1.2.5 Studies of Amines and Amides

Polarizable models have been applied to the study of structure, dynamics and hydrogen-bonding behavior of amines and amides.[55, 81, 194, 195] N-methylacetamide (NMA) is a common model compound used in force field studies due to the similarity of its structural and conformational characteristics to the peptide backbone of a protein.[81] Rick *et al* explored the effects of conformational changes on free energy calculations of solvated acetamide and N-methylacetamide using a polarizable model based on fluctuating charges.[106] A shift towards increasing dipole moment of water molecules near NMA is observed and NMA solvation free energies were closer to experiment than non-polarizable models. Similar studies in the methylation of small amines and amides also showed that polarizable models substantially improve the solvation free energies compared to non-polarizable models.[81, 82] The polarization effects of hydrogen bonding between NMA dimers was studied by Mannfors and coworkers.[196] Polarization proved to be important for transferability for more complex compounds and for the proper description of hydrogen-bonding when the computed properties (electric potentials, dipole moments, and polarizability) from polarizable models were compared to non-polarizable models.

1.2.6 Towards Studies of Macromolecular Systems

Some effort has been put forth towards producing a robust and reliable polarizable force field for proteins and DNA. Kaminski *et al* have developed a polarizable force field based on fluctuating bond-charge increments and atom-centered inducible dipoles that has been tested on twenty amino acid dipeptides in gas phase.[197, 118, 117] Short molecular dynamics simulations of 1-picosecond were performed on thirty-nine proteins in gas phase. This force field has since been parameterized for condensed-phase simulations for small molecules that represent functional groups found in organic compounds.[157] This polarizable force field has also been applied to bovine pancreatic trypsin inhibitor (BPTI) solvated in TIP4P-FQ and RPOL polarizable water for a 2-nanosecond simulation.[62] A fluctuating charge force field by Patel, Mackerell and Brooks has been applied to six small proteins in a polarizable TIP4P-FQ solvent.[11, 100, 32, 96] They performed molecular dynamics simulations on these proteins in solution, lasting for a few nanoseconds. Baucom and coworkers have carried out 25-nanosecond crystal simulations of a double-stranded DNA decamer using the polarizable ff02 field in conjunction with the POL3 polarizable water model.[85] Anisimov *et al* have also run molecular dynamic condensed-phase simulations for 1-nanosecond on a DNA octamer in SWM4-DP polarizable water with sodium ions, using a polarizable force field based on the classical Drude oscillator formalism.[84] There is also an ongoing ef-

fort to parameterize the AMOEBA force field for macromolecules.[198, 31, 125, 149, 52] The fluctuating charge polarizable force field has also been recently applied to the membrane protein, gramicidin A, by Patel and coworkers in a 5-nanosecond simulation.[109]

Current work has demonstrated the feasibility for these force fields to run stable simulations on small molecular systems on short timescales. Although there is still no fully parameterized polarizable force field for macromolecular systems that clearly eliminates all failures from non-polarizable force fields, polarizable models successfully simulate the variation in dielectric environment with charge redistribution and changing dipole moments that non-polarizable models cannot describe. Polarizable models can also improve the description of dynamic properties. Further studies of polarization effects in large biomolecular systems is impeded by the demanding computational cost of explicitly including polarizable interactions. The most expensive calculation is computing the induced dipoles at each movement of the system. This computational cost of traditional solvers limits the size of systems that can be fully described with the explicit inclusion of polarization.

1.3 Overview

The main objective of this study is to develop an efficient algorithm that accelerates the computation of induced dipole moments for large-scale problems. Due to all given limitations of the fluctuating charge and the Drude oscillator model, this thesis will focus on incorporating polarizable point dipoles for the description of the polarization. Since these dipole-dipole interactions are long-range, decaying as r^{-3}, an accurate and efficient treatment of long-range interactions needs to be considered. In the next chapter, we will review current methods for long-range electrostatic interactions including Ewald and multipole-based methods. We will primarily focus on a multipole-based method, the cell multipole method, and introduce our derivations for dipolar systems.

Including point dipoles when computing electrostatic interactions requires substantial computational effort, so fast solvers are explored to accelerate these calculations. Commonly used solvers such as the Jacobi, Gauss-Seidel, successive over-relaxation, conjugate gradient, and multigrid methods are described in Chapter 3. These techniques are generally implemented to increase the rate of convergence of brute force iterative calculations. We will concentrate on multigrid methods because of their optimal scaling and potential for parallelization.

In Chapter 4, we will describe the development of an algorithm that combines multigrid and cell multipole methods (MG-CMM) to accelerate the conver-

gence in computing induced dipole moments. We also demonstrate its practical implementation on a uniformly distributed one-dimensional model problem and further examine the performance of this method. This algorithm is then extended to include our matrix implementation of CMM with the algebraic multigrid method for more general applicability and transferability. We conclude our work with a summary of our algorithm development and perspectives for future work in Chapter 5.

Table 1.1: Density ρ in (g/cm^3), total potential energy U^{pot} in (kJ/mol), average molecular dipole moment μ^{mean} in (D), dielectric constant ϵ, and self-diffusion coefficient D(x10$^{-5}cm^2/s$), for the liquid-state water at ambient conditions as well as the temperature T$_{\rho_{max}}$ in (K) at the density maximum reported by various water models from literature compared to experiment.

Model	ρ	U^{pot}	μ^{mean}	ϵ	D	T$_{\rho_{max}}$
Non-Polarizable						
SPC[96, 146, 129]	0.971	-41.9	2.27	68	3.3	228
TIP3P[129]	0.982	-41.1	2.35	82	5.19	182
TIP4P[96, 146, 129]	0.999	-42.3	2.18	53	3.6	253
SPC/E[147, 2]	0.998	-41.4	2.35	71	2.5	235
TIP5P[144, 145]	0.999	-41.3	2.29	82	2.62	277
TIP4P/2005[287]	0.998	–	2.31	60	2.08	278
Polarizable						
RPOL[34, 121]	0.994	-41.6	2.62	106	2.4	–
SPC-FQ[96]	–	-41.4	2.83	116	1.7	–
PPC[141, 142]	0.997	-41.4	2.51	77	2.6	277
TIP4P-FQ[96, 143]	0.998	-41.4	2.62	79	1.9	280
BSV[79, 122]	–	-41.2	2.77	173.2	2.3	–
DC[10]	0.995	-41.9	2.75	–	2.1	–
SW-FLEX[88]	0.997	-41.7	2.69	116	3.66	–
POL5/TZ[118]	0.997	-41.5	2.71	98	1.81	293
TTM2-F[286]	1.046	-45.1	2.67	67.2	1.4	–
PRG[159]	–	-41.9	2.59	101	2.44	–
AMOEBA[125, 149]	1.00	–	2.78	81	2.02	290
COS/G2[90]	0.997	-41.3	2.59	87.8	2.3	–
SWM4-DP[4]	–	-41.6	2.46	79	2.30	–
GCPM[160]	1.004	-44.8	2.72	84.3	2.26	260
Experiment						
[Ref[138, 90, 133, 149, 139, 140]]	0.997	-41.5	2.9-3.0	78	2.30	277

Table 1.2: Reported critical properties, liquid vapor temperature T_c in (K) and critical density ρ_c in (g/cm^3), of various water models compared to experiment.

Model	T_c	ρ_c
Non-Polarizable		
SPC[151]	587	0.27
TIP4P[152]	580	0.33
SPC/E[160]	640	0.29
TIP5P[152]	538	0.29
TIP4P/2005[153]	640	0.31
Polarizable		
SPC-FQ[104]	540	0.33
PPC[151]	606	0.30
TIP4P-FQ[104]	570	0.35
BSV[151]	615	0.28
DC[170]	565	0.28
GCPM[160]	642	0.33
Experiment		
[Ref.[285]]	647	0.32

Chapter 2

Methods for Describing Long-range Electrostatic Interactions

Electrostatic interactions play an important role in the stability and function of biomolecular systems. In classical theory, electrostatic interactions are governed by Coulomb's law where the total electrostatic energy of the system is derived as the pairwise summation of Coulomb interactions.[1] For a charge distribution, the electrostatic potential at a given point charge i is defined as:

$$V(\vec{r}_i) = \sum_{j}^{n} \frac{q_j}{r_{ij}}. \tag{2.1}$$

Modeling long-range electrostatics has been a major challenge because of the high computational cost for large-scale systems that is proportional to the square of the number of particles. This chapter gives an overview of common methods for computing long-range interactions in simulation of large biomolecules.

There are numerous ways of reducing the computational effort of representing these electrostatic interactions, depending on the efficiency and accuracy needed for a given problem. The most efficient methods are cutoff schemes that simply ignore long-range interactions completely, which eliminates the expensive part of the calculation. This abrupt truncation has been shown to give rise to severe inaccuracies, energetic instabilities, and artifacts in simulating liquids,[199, 200, 201, 202] peptides/proteins,[203, 204] and DNA.[205] In order to overcome the limitations of these cutoff methods, various Ewald summation and multipole-based methods have been developed to treat electrostatic interactions.

2.1 Ewald Summation and Other Lattice Summation Schemes

The Ewald summation method was first developed to study properties of ionic crystals.[206] This technique places all particles in a given system into a central cell where each particle interacts with other particles in the system and all of their

images in an infinite array of simulation cells.[207, 208] Thus, a periodic system with multiple images of each particle is created as shown in Figure 2.1. The total energy

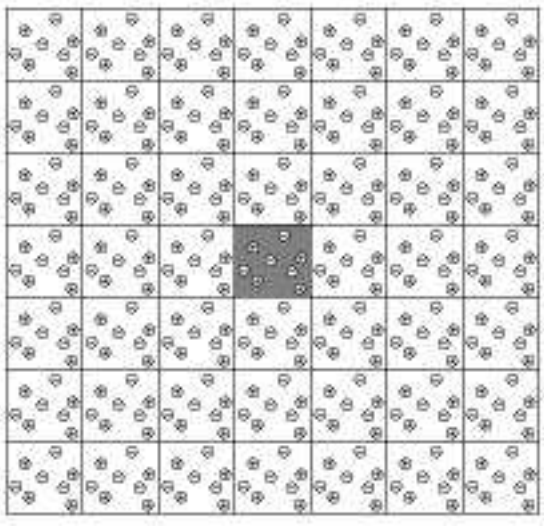


Figure 2.1: Ewald system of a central cell and all of its images.

from the interactions inside the central cell and the interactions of the central cell with all periodic images for a system of n atoms is:[207]

$$U = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{m\in Z^3}\frac{q_i q_j}{|r_{ij}+mL|} \tag{2.2}$$

where q is partial charge, $r_{ij} = \vec{r}_i - \vec{r}_j$, m is the index for all periodic cells excluding all self-interactions, and L is the length of the periodic cubic cell with dimensions $L \times L \times L$. The summation in equation (2.2) can be converted into two fast

converging series using the identity $\left(\frac{1}{r} = \frac{f(r)}{r} + \frac{1-f(r)}{r}\right)$ to form:

$$\frac{1}{r_{i,j,m}} = \underbrace{(\frac{1}{r_{i,j,m}} - \frac{\mathrm{erf}(\alpha r_{i,j,m})}{r_{i,j,m}})}_{\text{Real Space}} + \underbrace{\frac{\mathrm{erf}(\alpha r_{i,j,m})}{r_{i,j,m}}}_{\text{Reciprocal Space}} \tag{2.3}$$

where $r_{i,j,m} = |r_{ij} + mL|$ and the error function $\mathrm{erf}(r) = \frac{2}{\sqrt{\pi}} \int_0^r e^{-x^2} dx$. This decomposition divides the summation into short-range interactions represented by a direct sum in real space and the long-range interactions described in the reciprocal space by Fourier transforms. If the parameter α is chosen optimally, Ewald scales as $O(n^{3/2})$.[224] To reduce the cost of computing the reciprocal part, fast Fourier transform (FFT) methods can be applied. FFT requires the point charges to be replaced with a mesh-based charge density. Two common mesh implementations[209, 210] for the Ewald sum are known as the particle-particle-particle-mesh (P^3M) method by Hockney and Eastwood[211] and the particle mesh Ewald (PME) method by Darden and coworkers.[212] FFT-Ewald methods reduce the computational cost to $O(n \log n)$.[212, 213]

2.2 Multipole-based Methods

Hierarchical methods were originally devised in astrophysics for simulating many-body interactions in gravitational problems.[214, 215] The main strategy of this method is to decompose a system into a hierarchy of cells, forming a tree

structure with successive levels of refinement. A monopole center-of-mass approximation is used to compute the cell-cell interactions over large distances recursively throughout the entire tree structure, reducing the computational cost to the order of $O(n \log n)$ rather than $O(n^2)$. This scheme was extended to include higher-order multipole expansions and conversion to local Taylor expansions to further speed up the algorithm to form the fast multipole method (FMM) of order $O(n)$.[216, 217]

The Cartesian version of this method, called the cell multipole method (CMM), was devised in 1992 by Ding for large-scale molecular systems.[218] CMM can be more naturally incorporated into current molecular dynamics simulation software typically set in Cartesian coordinates in comparison to the original fast multipole methods described in spherical harmonics.[219] The main advantage of these multipole-based techniques is that they scale linearly with system size by computing interactions in a hierarchical way. Recent improvements in the cell multipole method include the treatment of periodic systems[220] and parallelization.[221]

There has been much debate about whether fast multipole methods or Ewald methods are more efficient. FMM scales linearly with n, which is better than the P^3M method's $O(n \log n)$ scaling. The particle mesh Ewald method is generally faster than multipole-based methods on single processor machines.[222] For systems larger than 20,000 atoms, FMM has been shown to be the most efficient by Figueirido.[223] FMM's efficiency for very large systems outweighs the cost of greater

code complexity. FMM parallel implementations also scale better than P^3M.[224] This stems from problems in parallelizing FFT in the Ewald-FFT methods.[222] Direct comparisons between Ewald summation and cell multipole methods have shown CMM to perform dramatically faster with similar accuracy.[225] The FMM approach is more efficient for highly non-uniform systems because of adaptive FMM approaches,[226, 227, 228] whereas the grid size grows dramatically faster as the system size increases in P^3M.[224] In choosing the appropriate method, we have to strike a balance between accuracy and efficiency to handle large-scale systems. Since our goal is to develop an efficient method that can handle large systems, we will concentrate on multipole-based methods, specifically the cell multipole method that scales linearly with the size of the system. The following sections will provide an overview of the theory and implementation of CMM for long-range electrostatic interactions.

2.2.1 Cell Multipole Method

CMM reduces the number of computations for long-range electrostatic Coulomb interactions in large systems by representing groups of distant atoms with multipole and Taylor series expansions. An advantage of this method is that it can be easily implemented in parallel.[221]

System Decomposition. Consider a collection of particles is placed in

a computational box, just large enough to enclose all of the atoms. This box of particles is successively divided into cells, shown in Figure 2.2.

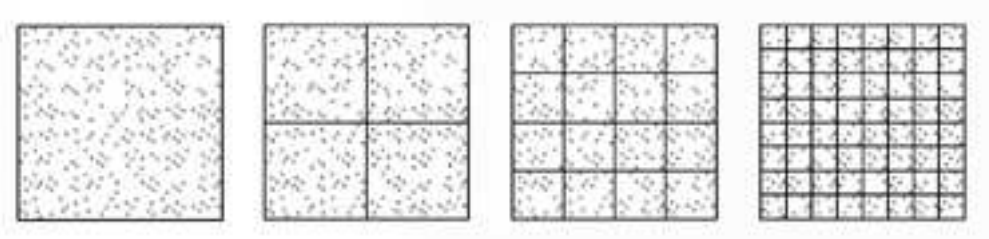


Figure 2.2: System decomposition, shown in two dimensions.

This hierarchy of cells forms a tree structure. This tree is a systematic way to differentiate between nearby and distant atoms. At level 0 (root) of the tree structure, the computational box is a single cell containing the entire system of atoms. At each level, each parent cell is subsequently divided into children cells. According to Ding, four atoms per cell achieves optimal accuracy in the calculations.[218] Thus, the number of levels is:

$$\text{Number of Levels} = log_8 \left(\frac{\text{n}}{4}\right). \tag{2.4}$$

At the deepest level of the tree structure, the number of cells is:

$$\text{Number of Cells} = 8^{\text{numLevels}} \tag{2.5}$$

with the size of each cell being:

$$\text{Cell Size} = \frac{\text{Size of Computational Box}}{\text{Number of Cells}}. \tag{2.6}$$

Cell Multipole Moments. After the system has been divided in a hierarchical way, each cell can be represented by a single potential at the center of the cell ($\vec{r}_A$). The potential is divided into near field and far field contributions. A cell's near field includes itself and its nearest neighbors (any cell sharing its boundary) whereas the far field refers to the rest of cells not included in the near field at the same level. In the cell multipole method, an approximation is made for the interactions between atoms in a given cell and other atoms more than one cell away. Assuming that the far cells are significantly farther than the near particles, this series expansion:[70]

$$\frac{1}{|\vec{R}-\vec{r}_i|} = \frac{1}{R}\left[1 + \frac{\vec{R}\cdot\vec{r}_i}{R^2} - \frac{1}{2}\left(\frac{r_i}{R}\right)^2 + \frac{3}{2}\frac{(\vec{R}\cdot\vec{r}_i)^2}{R^4} - \frac{3}{2}\frac{(\vec{R}-\vec{r}_i)r_i^2}{R^4} + \frac{3}{8}\left(\frac{r_i}{R}\right)^4 - \cdots\right] \tag{2.7}$$

where $\vec{R} = \vec{r} - \vec{r}_A$, is substituted in the Coulomb potential from equation (2.1). These terms can be combined and rearranged in the form:

$$V(\vec{r}) = V^{(1)}(\vec{r}) + V^{(2)}(\vec{r}) + V^{(3)}(\vec{r}) + \cdots \tag{2.8}$$

which yields the multipole expansion of the potential expressed in the form of an infinite series of multipole moments, which is commonly truncated to four terms, monopole (Z), dipole (μ), quadrupole (Q) and octopole (O):

$$\begin{aligned} V_A^{pole}(r) = &\frac{Z}{R} + \sum_{\alpha=x,y,z}\frac{\mu_\alpha R_\alpha}{R^3} + \sum_{\alpha=x,y,z}\sum_{\beta=x,y,z}\frac{Q_{\alpha\beta}R_\alpha R_\beta}{R^5} \\ &+ \sum_{\alpha=x,y,z}\sum_{\beta=x,y,z}\sum_{\gamma=x,y,z}\frac{O_{\alpha\beta\gamma}R_\alpha R_\beta R_\gamma}{R^7} + \cdots \end{aligned} \tag{2.9}$$

where $\vec{R}$ is equal to the difference between $\vec{r}$ and $\vec{r}_A$, $\vec{r}$ is the position vector of any atom outside of cell A, $\vec{r}_A$ is the center of cell A, and α, β, γ are any Cartesian coordinates, x, y, or z. The monopole, dipole, quadrupole and octopole terms for the deepest level are as follows:

$$\begin{aligned} Z &= \sum_i q_i \\ \mu_\alpha &= \sum_i q_i r_{i\alpha} \\ Q_{\alpha\beta} &= \sum_i \frac{q_i}{2} \left[3 r_{i\alpha} r_{i\beta} - \delta_{\alpha\beta} r_i^2\right] \\ O_{\alpha\beta\gamma} &= \sum_i \frac{q_i}{2} \left[5 r_{i\alpha} r_{i\beta} r_{i\gamma} - (r_{i\alpha}\delta_{\beta\gamma} + r_{i\beta}\delta_{\gamma\alpha} + r_{i\gamma}\delta_{\alpha\beta}) r_i^2\right] \end{aligned} \tag{2.10}$$

where $r_{i\alpha}$ is the α component of the position vector for atom i with respect to the center of cell A. The multipole moments of each cell are computed at the deepest level and translated to the higher levels. This is continued until the root level is reached, where a multipole expansion representing the field created by all atoms in the system is formed. In order to translate child cell moments to parent cell moments, we must make the substitution: $\vec{r}_i \leftarrow \vec{r}_{ik} + \vec{R}_k$, to shift the cells.

Therefore, the multipole moments of higher levels are given as:

$$
\begin{aligned}
Z^{(l-1)} &= \sum_{k=1}^{children} \sum_{i} q_{ik} = \sum_{k=1}^{children} Z_k^{(l)} \\
\vec{\mu}^{(l-1)} &= \sum_{k=1}^{children} \sum_{i} q_{ik}(\vec{r}_{ik} + \vec{R}_k) = \sum_{k=1}^{children} (\vec{\mu}_k^{(l)}) + Z_k^{(l)} \vec{R}_k \\
Q_{\alpha\beta}^{(l-1)} &= \sum_{k=1}^{children} \sum_{i} \frac{q_{ik}}{2} \left[3(r_{i\alpha k} + R_{\alpha k})(r_{i\beta k} + R_{\beta k}) - \delta_{\alpha\beta} |\vec{r}_{ik} + \vec{R}_k|^2 \right] \\
&= \sum_{k=1}^{children} Q_{\alpha\beta k}^{(l)} + \frac{1}{2} \sum_{k=1}^{children} \left[3(\mu_{\alpha k}^{(l)} R_{\beta k} + \mu_{\beta k}^{(l)} R_{\alpha k}) - 2\delta_{\alpha\beta}(\vec{\mu}_k^{(l)} \cdot \vec{R}_k) \right] \\
&\quad + \frac{1}{2} \sum_{k=1}^{children} Z_k^{(l)} \left[3R_{\alpha k} R_{\beta k} - \delta_{\alpha\beta} |\vec{R}_k|^2 \right] \\
O_{\alpha\beta\gamma}^{(l-1)} &= \sum_{k=1}^{children} \sum_{i} \frac{q_{ik}}{2} [5(r_{i\alpha k} + R_{\alpha k})(r_{i\beta k} + R_{\beta k})(r_{i\gamma k} + R_{\gamma k}) \\
&\quad - ((r_{i\alpha k} + R_{\alpha k})\delta_{\beta\gamma} + (r_{i\beta k} + R_{\beta k})\delta_{\gamma\alpha} + (r_{i\gamma k} + R_{\gamma k})\delta_{\alpha\beta})] |\vec{r}_{ik} + \vec{R}_k|^2 \\
&= \sum_{k=1}^{children} O_{\alpha\beta\gamma k}^{(l)} - \frac{2}{3} \sum_{k=1}^{children} \sum_{\lambda=x,y,z} \left(\delta_{\alpha\beta} Q_{\gamma\lambda k}^{(l)} + \delta_{\beta\gamma} Q_{\alpha\lambda k}^{(l)} + \delta_{\gamma\alpha} Q_{\beta\lambda k}^{(l)} \right) R_{\lambda k} \\
&\quad - \frac{1}{2} \sum_{k=1}^{children} (\delta_{\alpha\beta} \mu_{\gamma k}^{(l)} + \delta_{\beta\gamma} \mu_{\alpha k}^{(l)} + \delta_{\gamma\alpha} \mu_{\beta k}^{(l)}) |\vec{R}_k|^2 \\
&\quad - \sum_{k=1}^{children} ((\vec{\mu}_k^{(l)} \cdot \vec{R}_k)(\delta_{\alpha\beta} R_{\gamma k} + \delta_{\beta\gamma} R_{\alpha k} + \delta_{\gamma\alpha} R_{\beta k})) \\
&\quad + \frac{15}{6} \sum_{k=1}^{children} (\mu_{\alpha k}^{(l)} R_{\beta k} R_{\gamma k} + R_{\alpha k} \mu_{\beta k}^{(l)} R_{\gamma k} + R_{\alpha k} R_{\beta k} \mu_{\gamma k}^{(l)}) \\
&\quad + \frac{1}{2} \sum_{k=1}^{children} Z_k^{(l)} \left[5R_{\alpha k} R_{\beta k} R_{\gamma k} - (\delta_{\alpha\beta} R_{\gamma k} + \delta_{\beta\gamma} R_{\alpha k} + \delta_{\gamma\alpha} R_{\beta k}) |\vec{R}_k|^2 \right]
\end{aligned}
\tag{2.11}
$$

The general scheme for the upward pass is given in Algorithm 1.

Algorithm 1 : Hierarchical CMM upward pass.

```
\* For the deepest level *\
do all cells at the deepest level
    Compute cell multipole moments (Z, μ, Q, O)
end do
\* For the upper levels *\
do all remaining levels until root level is reached
    do all cells at current level
       Parent cell moments = translation & summation of child cell moments
    end do
end do
```

Far Field Taylor Series Coefficients. Once the multipole moments have been computed at all levels, the far field potential of all the atoms in a cell A may be obtained by summing all the contributions from far cells. This far field potential is represented by a Taylor series centered at the center of cell A. A Taylor series expansion at each cell is constructed by a downward pass through the tree structure. The root level and the next level down cannot be included in the downward pass because they have no far field contributions. Subsequently, at each cell in each level, Taylor series coefficients representing the distant cells are computed, using the cell multipole moments from the upward pass. The Taylor series expansion at each cell in the deepest level represents the weak, far field interactions. Consider the first level as level A with a local Taylor series expansion

being:

$$\begin{aligned} V_{A_0}^T(\vec{r}) &= \sum_A V_A^{pole}(\vec{r} - \vec{r}_A) \\ &= V^{(0)} + \sum_\alpha V_\alpha^{(1)} r_\alpha + \sum_{\alpha\beta} V_{\alpha\beta}^{(2)} r_\alpha r_\beta + \dots \end{aligned} \tag{2.12}$$

and the atom position $\vec{r}$ and cell position $\vec{r}_A$ are with respect to the center of cell C_0 at the deepest level, $V^{(0)}$ is the sum of all constant terms, $\sum_\alpha V_\alpha^{(1)}$ is the sum of all linear coefficients, and $\sum_{\alpha\beta} V_{\alpha\beta}^{(2)}$ is the sum of all second order coefficients. Then, the Taylor series expansion of the child cell will consist of two terms. The first term comes from shifting the parent's Taylor series expansion to the center of the corresponding child cell. The second term is the contribution from all far field cells at the current level that were not included in the parent's local expansion. Therefore, the local expansions in lower levels (level B, C...) are:

$$\begin{aligned} V_{B_0}^T(\vec{r}) &= V_{A_0}^T(\vec{r} + \vec{r}_0) + \sum_B V_B^{pole}(\vec{r} - \vec{r}_B) \\ V_{C_0}^T(\vec{r}) &= V_{B_0}^T(\vec{r} + \vec{r}_0) + \sum_C V_C^{pole}(\vec{r} - \vec{r}_C) \end{aligned} \tag{2.13}$$

where $V_{A_0}^T(\vec{r} + \vec{r}_0)$, $V_{B_0}^T(\vec{r} + \vec{r}_0)$, and so on are substituted with:

$$V^T(\vec{r} + \vec{r}_0) = V^{(0)} + V^{(1)}(\vec{r} + \vec{r}_0) + V^{(2)}(\vec{r} + \vec{r}_0)^2 + V^{(3)}(\vec{r} + \vec{r}_0)^3 \tag{2.14}$$

which represents shifting the center to the center of the child cell. Expanding each multipole term in $\vec{r}$ and taking the first four terms in each expansion gives the

charge terms:

$$\begin{aligned}\frac{Z}{|\vec{r}-\vec{r}_A|} =& \frac{Z}{r_A} + \frac{Z\vec{r}_A}{r_A}\cdot\vec{r} + \sum_{\alpha\beta}\frac{Z(3r_{A,\alpha}r_{A,\beta}-\delta_{\alpha\beta}r_A^2)}{2r_A^5}r_\alpha r_\beta \\ &+ \sum_{\alpha\beta\gamma}\frac{3Z(5r_{A,\alpha}r_{A,\beta}r_{A,\gamma}-(\delta_{\alpha\beta}r_{A,\gamma}+\delta_{\beta\gamma}r_{A,\alpha}+\delta_{\alpha\gamma}r_{A,\beta}))}{r_A^7}r_\alpha r_\beta r_\gamma\end{aligned} \tag{2.15}$$

the dipole terms:

$$\begin{aligned}\frac{\vec{\mu}\cdot(\vec{r}-\vec{r}_A)}{|\vec{r}-\vec{r}_A|^3} =& \frac{\vec{\mu}\cdot\vec{r}_A}{r_A^3} + \left[\frac{\vec{\mu}}{r_A^3} - \frac{3(\vec{\mu}\cdot\vec{r}_A)\vec{r}_A}{r_A^5}\right]\cdot\vec{r} \\ &+ \sum_{\alpha\beta}\left[\frac{3[(\mu_\alpha r_{A,\beta}+\mu_\beta r_{A,\alpha})+\delta_{\alpha\beta}(\vec{\mu}\cdot\vec{r}_A)]}{2r_A^5} - \frac{15r_{A,\alpha}r_{A,\beta}}{2r_A^7}\right]r_\alpha r_\beta \\ &+ \sum_{\alpha\beta\gamma}\Bigg[\frac{-(\mu_\alpha\delta_{\beta\gamma}+\mu_\beta\delta_{\alpha\gamma}+\mu_\gamma\delta_{\alpha\beta})}{2} \\ &\quad + \frac{5(\mu_\alpha r_{A,\beta}r_{A,\gamma}+\mu_\beta r_{A,\alpha}r_{A,\gamma}+\mu_\gamma r_{A,\alpha}r_{A,\beta})}{2r_A^7} \\ &\quad + \frac{5(\vec{\mu}\cdot\vec{r_A})(\delta_{\alpha\beta}r_{A,\gamma}+\delta_{\alpha\gamma}r_{A,\beta}+\delta_{\beta\gamma}r_{A,\alpha})}{2r_A^7} \\ &\quad + \frac{-35(\vec{\mu}\cdot\vec{r_A})r_{A,\alpha}r_{A,\beta}r_{A,\gamma}}{2r_A^9}\Bigg]r_\alpha r_\beta r_\gamma\end{aligned} \tag{2.16}$$

the quadrupole terms:

$$
\begin{aligned}
&\frac{Q\cdot(\vec{r}-\vec{r}_A)\cdot(\vec{r}-\vec{r}_A)}{|\vec{r}-\vec{r}_A|^5} = \\
&\sum_{\rho\nu}\frac{Q_{\rho\nu}r_{A,\rho}r_{A,\nu}}{r_A^5}+\sum_{\alpha\rho\nu}\left[\frac{-(Q_{\alpha\nu}r_{A,\nu}+Q_{\rho\alpha}r_{A,\rho})}{r_A^5}+\frac{5Q_{\rho\nu}r_{A,\rho}r_{A,\nu}r_{A,\alpha}}{r_A^7}\right]r_\alpha \\
&+\sum_{\alpha\beta}\left[\sum_{\rho\nu}\left(\frac{-5}{2r_A^7}[Q_{\rho\nu}r_{A,\rho}r_{A,\nu}\delta_{\alpha\beta}+Q_{\rho\alpha}r_{A,\rho}r_{A,\beta}+Q_{\alpha\nu}r_{A,\nu}r_{A,\beta}+Q_{\rho\beta}r_{A,\rho}r_{A,\alpha}\right.\right. \\
&\qquad\left.\left.+Q_{\beta\nu}r_{A,\nu}r_{A,\alpha}]+\frac{35Q_{\rho\nu}r_{A,\rho}r_{A,\nu}r_{A,\alpha}r_{A,\beta}}{2r_A^9}\right)+\frac{Q_{\alpha\beta}}{r_A^5}\right]r_\alpha r_\beta \\
&+\sum_{\alpha\beta\gamma}\left[\frac{5(Q_{\alpha\beta}r_{A,\gamma}+Q_{\beta\gamma}r_{A,\alpha}+Q_{\alpha\gamma}r_{A,\beta})}{3r_A^7}\right. \\
&\qquad+\sum_{\rho}\left(\frac{5(Q_{\alpha\rho}r_{A,\rho}\delta_{\beta\gamma}+Q_{\beta\rho}r_{A,\rho}\delta_{\gamma\alpha}+Q_{\gamma\rho}r_{A,\rho}\delta_{\gamma\alpha})}{3r_A^7}\right. \\
&\qquad\qquad\left.+\frac{-35(Q_{\alpha\rho}r_{A,\rho}r_{A,\beta}r_{A,\gamma}+Q_{\beta\rho}r_{A,\rho}r_{A,\alpha}r_{A,\gamma}+Q_{\gamma\rho}r_{A,\rho}r_{A,\alpha}r_{A,\beta})}{3r_A^9}\right) \\
&\qquad+\sum_{\rho\nu}\left(\frac{-35Q_{\rho\nu}r_{A,\rho}r_{A,\nu}(\delta_{\alpha\beta}r_{A,\gamma}+\delta_{\alpha\gamma}r_{A,\beta}+\delta_{\beta\gamma}r_{A,\alpha})}{3r_A^9}\right. \\
&\qquad\qquad\left.\left.+\frac{315Q_{\rho\nu}r_{A,\rho}r_{A,\nu}r_{A,\alpha}r_{A,\beta}r_{A,\gamma}}{r_A^{11}}\right)\right]r_\alpha r_\beta r_\gamma
\end{aligned}
\tag{2.17}
$$

and the octopole terms:

$$
\begin{aligned}
&\frac{O_{\alpha\beta\gamma}\cdot(\vec{r}-\vec{r}_A)\cdot(\vec{r}-\vec{r}_A)\cdot(\vec{r}-\vec{r}_A)}{|\vec{r}-\vec{r}_A|^7} = \\
&\sum_{\rho\nu\lambda}\frac{-O_{\rho\nu\lambda}r_{A,\rho}r_{A,\nu}r_{A,\lambda}}{r_A^7} \\
&+\sum_{\alpha}\left[\sum_{\rho\nu}\frac{3O_{\rho\nu\alpha}r_{A,\rho}r_{A,\nu}}{r_A^7}+\sum_{\rho\nu\lambda}\frac{-7O_{\rho\nu\lambda}r_{A,\rho}r_{A,\nu}r_{A,\lambda}r_{A,\alpha}}{r_A^9}\right]r_\alpha \\
&+\sum_{\alpha\beta}\left[\sum_{\rho}\frac{-3O_{\rho\alpha\beta}r_{A,\rho}}{r_A^7}+\sum_{\rho\nu}\frac{21(O_{\rho\nu\alpha}r_{A,\rho}r_{A,\nu}r_{A,\beta}+O_{\rho\nu\beta}r_{A,\rho}r_{A,\nu}r_{A,\alpha})}{2r_A^9}\right. \\
&\qquad\left.+\sum_{\rho\nu\lambda}O_{\rho\nu\lambda}r_{A,\rho}r_{A,\nu}r_{A,\lambda}\left(\frac{\delta_{\alpha\beta}}{r_A^9}-\frac{9r_{A,\alpha}r_{A,\beta}}{r_A^{11}}\right)\right]r_\alpha r_\beta \\
&+\sum_{\alpha\beta\gamma}\left[\frac{O_{\alpha\beta\gamma}}{r_A^7}+\sum_{\rho}\frac{-7r_{A,\nu}(O_{\rho\alpha\beta}r_{A,\gamma}+O_{\rho\beta\gamma}r_{A,\alpha}+O_{\rho\gamma\alpha}r_{A,\beta})}{r_A^9}\right. \\
&\qquad+\sum_{\rho\nu}\frac{-7r_{A,\rho}r_{A,\nu}(O_{\rho\nu\alpha}\delta_{\beta\gamma}+O_{\rho\nu\beta}\delta_{\alpha\gamma}+O_{\rho\nu\gamma}\delta_{\alpha\beta})}{2r_A^9} \\
&\qquad+\sum_{\rho\nu}\frac{63r_{A,\rho}r_{A,\nu}(O_{\rho\nu\alpha}r_{A,\beta}r_{A,\gamma}+O_{\rho\nu\beta}r_{A,\alpha}r_{A,\gamma}+O_{\rho\nu\gamma}r_{A,\alpha}r_{A,\beta})}{2r_A^{11}} \\
&\qquad+\sum_{\rho\nu\lambda}\frac{21r_{A,\rho}r_{A,\nu}r_{A,\lambda}O_{\rho\nu\lambda}(\delta_{\alpha\beta}r_{A,\gamma}+\delta_{\beta\gamma}r_{A,\alpha}+\delta_{\gamma\alpha}r_{A,\beta})}{2r_A^{11}} \\
&\qquad\left.+\sum_{\rho\nu\lambda}\frac{-231r_{A,\rho}r_{A,\nu}r_{A,\lambda}O_{\rho\nu\lambda}r_{A,\alpha}r_{A,\beta}r_{A,\gamma}}{2r_A^{13}}\right]r_\alpha r_\beta r_\gamma
\end{aligned}
\tag{2.18}
$$

where the indices α, β, γ, ρ, ν, and λ all correspond to Cartesian coordinates x, y, or z.

The Taylor coefficients are computed for each far cell by collecting all the constant terms for $V^{(0)}$, linear coefficients of $\vec{r}$ for $V^{(1)}$, and so on. By expanding each multipole term and collecting like terms, the Taylor coefficients can be computed at each level, and translated to the centers of the children cells. This

process is repeated for all far field cells that were not part of the far field at a higher level of the tree, which is demonstrated in Algorithm 2. When the deepest level is reached, the far field can be combined with the near field pairwise interactions to find the total interaction between all atoms in the system.

Algorithm 2 : Hierarchical CMM downward pass.

```
\* For the most upper level with far field contributions*\
do all cells at this level
    Compute Taylor series coefficients of far field cells (V^(0), V^(1), V^(2)...)
end do
\* For lower levels *\
do all remaining levels until deepest level is reached
    do all cells at current level
       Child cell coefficients = shifting far field from parent to child & addition
                                 of all far field interactions at the current level
                                 that were not accounted for at the parent level
    end do
end do
```

Near field pairwise interactions. The interactions between a cell and its nearest neighbors are calculated directly by atomic pairwise interactions. The charges and positions of atoms in each cell and its nearest neighbors are used to calculate the near field potential at the deepest level.

$$V_{near}(r_i) = \sum_{j \epsilon near} \frac{q_j}{|\vec{r}_i - \vec{r}_j|} \tag{2.19}$$

Combination of far field and near field effects. The ultimate goal is to evaluate the potential of each particle in the system. Thus, the far field and near field contributions are summed for each atom in order to find the total interaction

between atoms in the system:

$$V(\vec{r}_i) = V_{near}(\vec{r}_i) + V_{far}(\vec{r}_i) \tag{2.20}$$

2.2.2 Matrix Version of the Cell Multipole Method

The standard cell multipole method involves a great deal of bookkeeping of the interaction lists of near cells and far cells at all levels. Efficient implementation of this method involves complex coding. This can be avoided by implementing CMM in a matrix framework. The same computations can be performed by a series of sparse matrices. This is a similar technique to the matrix FMM method.[229] Casting CMM in matrix form simplifies the implementation by circumventing the use of pointers to tree cells and creates a more automated process for computation. Also, setting this algorithm in matrix form involves matrix-vector operations, which lends itself to being more parallelizable. The underlying matrix structure can be exploited for further improvements in efficiency and accuracy, such as further factorizations or simplifications using different matrix properties. In this section, we describe our matrix implementation of the cell multipole method (CMMm).

The hierarchical cell multipole method can be described by the following matrix operations to compute the CMM potential for a system of charges:

1. $\mathbf{R}$ is the charge-to-multipole matrix. The matrix-vector product $\mathbf{R}\vec{q}$ converts the charges of the individual atoms into cell multipoles at the deepest level

of the tree, where $\vec{q}$ is the vector containing all the atom charges.

2. $\mathbf{U}^{(l-1)}$ is the lower-to-upper multipole translation matrix, representing the CMM upward pass. The product $\mathbf{U}^{(l-1)}(\mathbf{R}\vec{q})$ converts the level l cell multipoles into level $l-1$ cell multipoles.

3. $\mathbf{T}^{(l-1)}$ is the multipole to Taylor series conversion matrix. The product $\mathbf{T}^{(l-1)}(\mathbf{U}^{(l-1)}\mathbf{R}\vec{q})$ converts the level $l-1$ cell multipoles into level $l-1$ Taylor series coefficients.

4. $\mathbf{D}^{(l-1)}$ is the upper-to-lower Taylor series translation matrix, representing the CMM downward pass. The product $\mathbf{D}^{(l-1)}(\mathbf{T}^{(l-1)}\mathbf{U}^{(l-1)}\mathbf{R}\vec{q})$ converts the level $l-1$ Taylor series coefficients into level l Taylor series coefficients.

5. $\mathbf{T}^{(l)}$ is the level l multipole to Taylor series matrix. The product $\mathbf{T}^{(l)}(\mathbf{R}\vec{q})$ converts the level l cell multipoles into level l Taylor series coefficients.

6. Adding $\mathbf{D}^{(l-1)}\mathbf{T}^{(l-1)}\mathbf{U}^{(l-1)}\mathbf{R}\vec{q}+\mathbf{T}^{(l)}\mathbf{R}\vec{q}=(\mathbf{U}^{(l-1)}\mathbf{T}^{(l-1)}\mathbf{U}^{(l-1)}+\mathbf{T}^{(l)})\mathbf{R}\vec{q}$ combines the component of the far field calculated at level $l-1$ with the component calculated at level l.

7. $\mathbf{R}^{\top}$ is the transpose of the charge-to-multipole matrix, which is used to incorporate the locations of the target atoms into the Taylor series expansions. Thus, the far-field component of the electric field can be written as $V_{far}=\mathbf{R}^{\top}(\mathbf{D}^{(l-1)}\mathbf{T}^{(l-1)}\mathbf{U}^{(l-1)}+\mathbf{T}^{(l)})\mathbf{R}\vec{q}$.

8. For each additional level of the matrix, the higher level $\mathbf{T}^{(l-1)}$ is replaced with a nested computation of steps 2 through 6, for example:

$$V_{far} = \mathbf{R}^{\top} \left(\mathbf{D}^{(l-1)} \left(\mathbf{D}^{(l-2)} \mathbf{T}^{(l-2)} \mathbf{U}^{(l-2)} + \mathbf{T}^{(l-1)} \right) \mathbf{U}^{(l-1)} + \mathbf{T}^{(l)} \right) \mathbf{R} \vec{q}.$$

The series of matrix operations described above are implicitly included in a single matrix.

Single-level CMMm in One Dimension

To demonstrate the structure of the single-level CMM matrix form, we will consider the simplest case in 1-D that does not have a hierarchy: a system of 16 atoms, corresponding to 4 cells with 4 atoms per cell. The particle-particle interactions of this system are represented as a matrix $\mathbf{A}$, where each element A_{ij} is the electrostatic interaction between source atom j and target atom i. With a system of 16 atoms, $\mathbf{A}$ is a 16x16 matrix. This can be divided into 4x4 blocks where each block $\mathbf{A}_{IJ}$ represents the interaction between two cells in the deepest level of the tree structure: source cell J and target cell I.

This block matrix can be separated into near-field and far-field interactions

$$\mathbf{A} = \mathbf{A}_{near} + \mathbf{A}_{far}$$

$$\mathbf{A} = \left[\begin{array}{c|c|c|c} \text{near} & \text{near} & \textbf{far} & \textbf{far} \\ \hline \text{near} & \text{near} & \text{near} & \textbf{far} \\ \hline \textbf{far} & \text{near} & \text{near} & \text{near} \\ \hline \textbf{far} & \textbf{far} & \text{near} & \text{near} \end{array}\right]$$

where the near field of a target atom includes atoms within the same cell and neighboring cells. Thus, the blocks of matrix $\mathbf{A}$ that correspond to the near field are the diagonal blocks $\mathbf{A}_{I,I}$ (interactions within the same cell) and the blocks next to the diagonal $\mathbf{A}_{I,I-1}$ and $\mathbf{A}_{I,I+1}$ (interactions with neighboring cells). The interactions within the near field are computed exactly, as in the standard CMM approach.

Now we will focus on a particular block that corresponds to far-field interactions. Block $\mathbf{A}_{1,4}$ represents the far-field interaction between source cell 4 and target cell 1.

$$V_{1,4} = \begin{bmatrix} A_{1,13} & A_{1,14} & A_{1,15} & A_{1,16} \\ A_{2,13} & A_{2,14} & A_{2,15} & A_{2,16} \\ A_{3,13} & A_{3,14} & A_{3,15} & A_{3,16} \\ A_{4,13} & A_{4,14} & A_{4,15} & A_{4,16} \end{bmatrix} \begin{bmatrix} q_{13} \\ q_{14} \\ q_{15} \\ q_{16} \end{bmatrix}$$

In order to derive the matrix elements, we first examine the expression for the far field component of the potential V_{far} for an atom i in cell 1 due to all atoms

in cell 4:

$$V_{far}(\vec{r}_i - \vec{r}_{c_4}) = V^{(0)} + V^{(1)} r_i + V^{(2)} r_i^2 + V^{(3)} r_i^3,$$

where

$$\begin{aligned}
V^{(0)} &= \frac{Z}{|r_{c_4}|} + \frac{-\mu r_{c_4}}{|r_{c_4}|^3} + \frac{Q r_A^2}{|r_{c_4}|^5} + \frac{-O r_{c_4}^3}{|r_{c_4}|^7} \\
V^{(1)} &= \frac{Z r_{c_4}}{|r_{c_4}|^3} + \frac{-2\mu}{|r_{c_4}|^3} + \frac{3 Q r_{c_4}}{|r_{c_4}|^5} + \frac{-4O}{|r_{c_4}|^5} \\
V^{(2)} &= \frac{2Z}{|r_{c_4}|^3} + \frac{-6\mu r_{c_4}}{|r_{c_4}|^5} + \frac{12Q}{|r_{c_4}|^5} + \frac{-20 O r_{c_4}}{|r_{c_4}|^7} \\
V^{(3)} &= \frac{6 Z r_{c_4}}{|r_{c_4}|^5} + \frac{-24\mu}{|r_{c_4}|^5} + \frac{60 Q r_{c_4}}{|r_{c_4}|^7} + \frac{-120 O}{|r_{c_4}|^7}
\end{aligned}$$

where r_{c_4} is the center of the source cell 4. This can be rearranged into an expression in terms of the multipole moments Z, μ, Q, and O:

$$\begin{aligned}
V_{far}(\vec{r}_i - \vec{r}_{c_4}) = &\left(\frac{1}{|r_{c_4}|} + \frac{r_{c_4}}{|r_{c_4}|^3} r_i + \frac{2}{|r_{c_4}|^3} r_i^2 + \frac{6 r_{c_4}}{|r_{c_4}|^5} r_i^3 \right) Z \\
&+ \left(\frac{-r_{c_4}}{|r_{c_4}|^3} + \frac{-2}{|r_{c_4}|^3} r_i + \frac{-6 r_{c_4}}{|r_{c_4}|^5} r_i^2 + \frac{-24}{|r_{c_4}|^5} r_i^3 \right) \mu \\
&+ \left(\frac{r_{c_4}^2}{|r_{c_4}|^5} + \frac{3 r_{c_4}}{|r_{c_4}|^5} r_i + \frac{12}{|r_{c_4}|^5} r_i^2 + \frac{60 r_{c_4}}{|r_{c_4}|^7} r_i^3 \right) Q \\
&+ \left(\frac{-r_{c_4}^3}{|r_{c_4}|^7} + \frac{-4}{|r_{c_4}|^5} r_i + \frac{-20 r_{c_4}}{|r_{c_4}|^7} r_i^2 + \frac{-120}{|r_{c_4}|^7} r_i^3 \right) O
\end{aligned}$$

This can be rewritten in terms of the individual particle charges q_j using the

definitions of Z, μ, Q, and O:

$$\begin{aligned}
V_{far}(\vec{r}_i - \vec{r}_{c_4}) = &\left(\frac{1}{|r_{c_4}|} + \frac{r_{c_4}}{|r_{c_4}|^3}r_i + \frac{2}{|r_{c_4}|^3}r_i^2 + \frac{6r_{c_4}}{|r_{c_4}|^5}r_i^3\right)\sum_{j=13}^{16} q_j \\
&+ \left(\frac{-r_{c_4}}{|r_{c_4}|^3} + \frac{-2}{|r_{c_4}|^3}r_i + \frac{-6r_{c_4}}{|r_{c_4}|^5}r_i^2 + \frac{-24}{|r_{c_4}|^5}r_i^3\right)\sum_{j=13}^{16} q_j r_j \\
&+ \left(\frac{r_{c_4}^2}{|r_{c_4}|^5} + \frac{3r_{c_4}}{|r_{c_4}|^5}r_i + \frac{12}{|r_{c_4}|^5}r_i^2 + \frac{60r_{c_4}}{|r_{c_4}|^7}r_i^3\right)\sum_{j=13}^{16} q_j r_j^2 \\
&+ \left(\frac{-r_{c_4}^3}{|r_{c_4}|^7} + \frac{-4}{|r_{c_4}|^5}r_i + \frac{-20r_{c_4}}{|r_{c_4}|^7}r_i^2 + \frac{-120}{|r_{c_4}|^7}r_i^3\right)\sum_{j=13}^{16} q_j r_j^3
\end{aligned}$$

Then q_j can be factored out of the equation giving a single summation.

$$\begin{aligned}
V_{far}(\vec{r}_i - \vec{r}_{c_4}) = \sum_{j=13}^{16} &\left[\left(\frac{1}{|r_{c_4}|} + \frac{r_{c_4}}{|r_{c_4}|^3}r_i + \frac{2}{|r_{c_4}|^3}r_i^2 + \frac{6r_{c_4}}{|r_{c_4}|^5}r_i^3\right)\right. \\
&+ \left(\frac{-r_{c_4}}{|r_{c_4}|^3} + \frac{-2}{|r_{c_4}|^3}r_i + \frac{-6r_{c_4}}{|r_{c_4}|^5}r_i^2 + \frac{-24}{|r_{c_4}|^5}r_i^3\right) r_j \\
&+ \left(\frac{r_{c_4}^2}{|r_{c_4}|^5} + \frac{3r_{c_4}}{|r_{c_4}|^5}r_i + \frac{12}{|r_{c_4}|^5}r_i^2 + \frac{60r_{c_4}}{|r_{c_4}|^7}r_i^3\right) r_j^2 \\
&\left.+ \left(\frac{-r_{c_4}^3}{|r_{c_4}|^7} + \frac{-4}{|r_{c_4}|^5}r_i + \frac{-20r_{c_4}}{|r_{c_4}|^7}r_i^2 + \frac{-120}{|r_{c_4}|^7}r_i^3\right) r_j^3\right] q_j
\end{aligned}$$

Using this equation, block $\mathbf{A}_{1,4}$ is decomposed into a product of three matrices in the following way:

$$\mathbf{A}_{1,4} = \begin{bmatrix} 1 & r_1 & r_1^2 & r_1^3 \\ 1 & r_2 & r_2^2 & r_2^3 \\ 1 & r_3 & r_3^2 & r_3^3 \\ 1 & r_4 & r_4^2 & r_4^3 \end{bmatrix} \begin{bmatrix} \frac{1}{|r_{c_4}|} & \frac{-r_{c_4}}{|r_{c_4}|^3} & \frac{r_{c_4}^2}{|r_{c_4}|^5} & \frac{-r_{c_4}^3}{|r_{c_4}|^7} \\ \frac{r_{c_4}}{|r_{c_4}|^3} & \frac{-2}{|r_{c_4}|^3} & \frac{3r_{c_4}}{|r_{c_4}|^5} & \frac{-4}{|r_{c_4}|^5} \\ \frac{2}{|r_{c_4}|^3} & \frac{-6r_{c_4}}{|r_{c_4}|^5} & \frac{12}{|r_{c_4}|^5} & \frac{-20r_{c_4}}{|r_{c_4}|^7} \\ \frac{6r_{c_4}}{|r_{c_4}|^5} & \frac{-24}{|r_{c_4}|^5} & \frac{60r_{c_4}}{|r_{c_4}|^7} & \frac{-120}{|r_{c_4}|^7} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ r_{13} & r_{14} & r_{15} & r_{16} \\ r_{13}^2 & r_{14}^2 & r_{15}^2 & r_{16}^2 \\ r_{13}^3 & r_{14}^3 & r_{15}^3 & r_{16}^3 \end{bmatrix},$$

or $\mathbf{A}_{1,4} = \mathbf{R}_1^\top * \mathbf{T}_4 * \mathbf{R}_4$, where $\mathbf{R}$ contains values of r for the atoms in a given cell, $\mathbf{T}_4$ is the block of the multipole-to-Taylor series matrix corresponding to cell

4, and $\mathbf{R}_4^\top$ is the transpose of $\mathbf{R}_4$. Each block of $\mathbf{A}$ can be decomposed in a similar manner. Since $\mathbf{T}_4$, or more precisely r_{c4}, depends on the location of the cell center, it can be computed once the size of the bounding box for the system is known. In this case, each $\mathbf{R}_I$ block is 4x4, but in general, the number of columns in $\mathbf{R}_I$ is the number of atoms in cell I.

The block decomposition of $\mathbf{A}_{far}$ for the four-cell case is:

$$\begin{bmatrix} \mathbf{R}_1^\top & 0 & 0 & 0 \\ 0 & \mathbf{R}_2^\top & 0 & 0 \\ 0 & 0 & \mathbf{R}_3^\top & 0 \\ 0 & 0 & 0 & \mathbf{R}_4^\top \end{bmatrix} \begin{bmatrix} 0 & 0 & \mathbf{T}_3 & \mathbf{T}_4 \\ 0 & 0 & 0 & \mathbf{T}_4 \\ \mathbf{T}_1 & 0 & 0 & 0 \\ \mathbf{T}_1 & \mathbf{T}_2 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R}_1 & 0 & 0 & 0 \\ 0 & \mathbf{R}_2 & 0 & 0 \\ 0 & 0 & \mathbf{R}_3 & 0 \\ 0 & 0 & 0 & \mathbf{R}_4 \end{bmatrix} \tag{2.21}$$

$$= \begin{bmatrix} 0 & 0 & \mathbf{R}_1^\top \mathbf{T}_3 \mathbf{R}_3 & \mathbf{R}_1^\top \mathbf{T}_4 \mathbf{R}_4 \\ 0 & 0 & 0 & \mathbf{R}_2^\top \mathbf{T}_4 \mathbf{R}_4 \\ \mathbf{R}_3^\top \mathbf{T}_1 \mathbf{R}_1 & 0 & 0 & 0 \\ \mathbf{R}_4^\top \mathbf{T}_1 \mathbf{R}_1 & \mathbf{R}_4^\top \mathbf{T}_2 \mathbf{R}_2 & 0 & 0 \end{bmatrix}$$

The four-cell case gives a simplified version of the algorithm, involving only one level of the tree structure.

Hierarchical CMMm in One Dimension

To illustrate the hierarchical algorithm with upward and downward passes, we will now consider a system with 32 atoms, corresponding to 8 cells and four atoms per cell. Instead of a simple decomposition into near and far fields at a

single level, the far field interactions are further divided into levels (root, level 1, level 2, and level 3 in this case). The multipoles for cell J at the deepest level (level 3 in this case) are found by the following equation:

$$\mathbf{R}_J \cdot \vec{q}_J = \begin{bmatrix} Z_J \\ \mu_J \\ Q_J \\ O_J \end{bmatrix}$$

where $\vec{q}_J$ is the block of the original $\vec{q}$ vector corresponding to cell J, for example:

$$\vec{q} = \begin{bmatrix} \vec{q}_1 \\ \vec{q}_2 \\ \vec{q}_3 \\ \vec{q}_4 \\ \vec{q}_5 \\ \vec{q}_6 \\ \vec{q}_7 \\ \vec{q}_8 \end{bmatrix}. \tag{2.22}$$

As with the standard CMM, multipoles of higher levels are found by trans-

lating and summing child cells:

$$Z^{l-1}_{parent\ cell} = \sum_{k \in children} Z^l_k$$

$$\mu^{l-1}_{parent\ cell,x} = \sum_{k \in children} (\mu^l_{k,x} + Z^l_k \vec{R}_k)$$

$$Q_{parent\ cell,xx} = \sum_{k \in children} (Q^l_k + 2\vec{R}_k \mu^l_k + \vec{R}^2_k Z^l_k)$$

$$O_{parent\ cell,xxx} = \sum_{k \in children} (O^l_k + 3\vec{R}_k Q^l_k + 3\vec{R}^2_k \mu^l_k + \vec{R}^3_k Z^l_k)$$

where $\vec{R}_k$ is the vector from the center of the higher-level cell to the center of the lower-level cell, l is the lower level, and $l-1$ is the higher level.

The transfer operator for the upward pass from level l to upper level $l-1$, $\mathbf{U}^{(l-1)}_J$ can be written as follows, where J is the index of the parent cell at level $l-1$:

$$\mathbf{U}^{(l-1)}_J = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \vec{R}_{2J-1} & 1 & 0 & 0 & \vec{R}_{2J} & 1 & 0 & 0 \\ \vec{R}^2_{2J-1} & 2\vec{R}_{2J-1} & 1 & 0 & \vec{R}^2_{2J} & 2\vec{R}_{2J} & 1 & 0 \\ \vec{R}^3_{2J-1} & 3\vec{R}^2_{2J-1} & 3\vec{R}_{2J} & 1 & \vec{R}^3_{2J} & 3\vec{R}^2_{2J} & 3\vec{R}_{2J} & 1 \end{bmatrix}.$$

$\mathbf{U}_J^{(l-1)}$ operates in the following manner:

$$\mathbf{U}_J^{(l-1)} \cdot \begin{bmatrix} Z_{2J-1}^l \\ \mu_{2J-1}^l \\ Q_{2J-1}^l \\ O_{2J-1}^l \\ Z_{2J}^l \\ \mu_{2J}^l \\ Q_{2J}^l \\ O_{2J}^l \end{bmatrix} = \begin{bmatrix} Z_J^{l-1} \\ \mu_J^{l-1} \\ Q_J^{l-1} \\ O_J^{l-1} \end{bmatrix}.$$

The entire matrix $\mathbf{U}^{(l-1)}$ is composed of the individual $\mathbf{U}_J^{(l-1)}$ blocks:

$$\mathbf{U}^{(l-1)} = \begin{bmatrix} \mathbf{U}_1^{(l-1)} & 0 & 0 & 0 \\ 0 & \mathbf{U}_2^{(l-1)} & 0 & 0 \\ 0 & 0 & \mathbf{U}_3^{(l-1)} & 0 \\ 0 & 0 & 0 & \mathbf{U}_4^{(l-1)} \end{bmatrix}.$$

At the highest level with far field contributions (level 2), the multipole-to-Taylor matrix has the same structure as in the four-cell example:

$$\mathbf{T}^{(2)} = \begin{bmatrix} 0 & 0 & \mathbf{T}_3^{(2)} & \mathbf{T}_4^{(2)} \\ 0 & 0 & 0 & \mathbf{T}_4^{(2)} \\ \mathbf{T}_1^{(2)} & 0 & 0 & 0 \\ \mathbf{T}_1^{(2)} & \mathbf{T}_2^{(2)} & 0 & 0 \end{bmatrix} \tag{2.23}$$

The lower-level (level 3) multipole-to-Taylor matrix computes the interactions that were not accounted for by the parent cells at level 2.

$$\mathbf{T}^{(3)} = \begin{bmatrix} 0 & 0 & \mathbf{T}_3^{(3)} & \mathbf{T}_4^{(3)} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{T}_4^{(3)} & 0 & 0 & 0 & 0 \\ \mathbf{T}_1^{(3)} & 0 & 0 & 0 & \mathbf{T}_5^{(3)} & \mathbf{T}_6^{(3)} & 0 & 0 \\ \mathbf{T}_1^{(3)} & \mathbf{T}_2^{(3)} & 0 & 0 & 0 & \mathbf{T}_6^{(3)} & 0 & 0 \\ 0 & 0 & \mathbf{T}_3^{(3)} & 0 & 0 & 0 & \mathbf{T}_7^{(3)} & \mathbf{T}_8^{(3)} \\ 0 & 0 & \mathbf{T}_3^{(3)} & \mathbf{T}_4^{(3)} & 0 & 0 & 0 & \mathbf{T}_8^{(3)} \\ 0 & 0 & 0 & 0 & \mathbf{T}_5^{(3)} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{T}_5^{(3)} & \mathbf{T}_6^{(3)} & 0 & 0 \end{bmatrix} \tag{2.24}$$

The matrix operator for the downward pass is defined similarly to the upward pass. The following expression is used to shift the Taylor series by $\vec{r}_0$ from a parent cell to a child cell:

$$V_{A_0}^T(r + r_0) = V^{(0)} + V^{(1)}(r + r_0) + V^{(2)}(r + r_0)^2 + V^{(3)}(r + r_0)^3$$

Rearranging terms, the expression can be written as a polynomial in r:

$$\begin{aligned} V_{A_0}^T(r + r_0) = &\left(V^{(0)} + V^{(1)}r_0 + V^{(2)}r_0^2 + V^{(3)}r_0^3\right) \\ &+ \left(V^{(1)} + 2V^{(2)}r_0 + 3V^{(2)}r_0^2\right) r \\ &+ \left(V^{(2)} + 3V^{(3)}r_0\right) r^2 \\ &+ V^{(3)}r^3 \end{aligned}$$

The coefficients on the powers of $\vec{r}$ can be stored in a matrix-vector product:

$$\begin{bmatrix} 1 & r_0 & r_0^2 & r_0^3 \\ 0 & 1 & 2r_0 & 3r_0^2 \\ 0 & 0 & 1 & 3r_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} V^{(0)} \\ V^{(1)} \\ V^{(2)} \\ V^{(3)} \end{bmatrix} = \begin{bmatrix} V^{(0)} + V^{(1)}r_0 + V^{(2)}r_0^2 + V^{(3)}r_0^3 \\ V^{(1)}r_0 + 2V^{(2)}r_0 + 3V^{(3)}r_0^2 \\ V^{(2)} + 3V^{(3)}r_0 \\ V^{(3)} \end{bmatrix}$$

The matrix on the left is denoted $\mathbf{D}_J^{(l-1)}$, where l is the level of the child cells, and J refers to a particular child cell. The entire matrix $\mathbf{D}^{(l-1)}$ can be written in the following block form:

$$\mathbf{D}^{(2)} = \begin{bmatrix} \mathbf{D}_1^{(2)} & 0 & 0 & 0 \\ \mathbf{D}_2^{(2)} & 0 & 0 & 0 \\ 0 & \mathbf{D}_3^{(2)} & 0 & 0 \\ 0 & \mathbf{D}_4^{(2)} & 0 & 0 \\ 0 & 0 & \mathbf{D}_5^{(2)} & 0 \\ 0 & 0 & \mathbf{D}_6^{(2)} & 0 \\ 0 & 0 & 0 & \mathbf{D}_7^{(2)} \\ 0 & 0 & 0 & \mathbf{D}_8^{(2)} \end{bmatrix}$$

The three-dimensional case can be defined similarly. To summarize, we have derived a CMMm algorithm where a series of matrix operations is implicitly included

in a single matrix as shown below.

$$V = V^{near} + \begin{bmatrix} \mathbf{R}^\top \end{bmatrix} \left(\begin{bmatrix} \mathbf{D} \end{bmatrix} \cdot \underbrace{\begin{bmatrix} \quad \end{bmatrix}} \cdot \begin{bmatrix} \mathbf{U} \end{bmatrix} + \begin{bmatrix} \mathbf{T} \end{bmatrix} \right) \begin{bmatrix} \mathbf{R} \end{bmatrix} \begin{bmatrix} \vec{q} \end{bmatrix}$$

$$([\mathbf{D}] \cdot \underbrace{[\ ]} \cdot [\mathbf{U}] + [\mathbf{T}])$$

$$([\mathbf{D}] \cdot \underbrace{[\ ]} \cdot [\mathbf{U}] + [\mathbf{T}])$$

$$\ddots$$

2.2.3 Treatment of Polarizable Systems

Now we can focus our attention on dipolar systems, as this will be directly applied to the dipole iteration problem. Kutteh and Nicholas give the CMM potential terms for a dipolar system[230, 219] which we will describe in this section. We will then derive the field equations necessary for the dipole iteration problem at hand using these potential terms. The total electric field at a particle i for a system of charges and induced dipoles is:

$$\vec{E}_i = \sum_{j=1;j\neq i}^{n} \left(\frac{q_j \vec{r}_{ij}}{r_{ij}^3} + \mathbf{T}_{ij} \vec{\mu}_j \right) \tag{2.25}$$

where

$$\mathbf{T}_{ij} = \frac{1}{r_{ij}^3} \left(\frac{3\vec{r}_{ij}\vec{r}_{ij}^{\top}}{r_{ij}^2} - \mathbf{I} \right) \tag{2.26}$$

The cell multipole method generates a hierarchy of cells, where the interaction of cells is divided into near field and far field contributions. A cell's near field includes itself and its nearest neighbors and is computed directly using equation

(2.25). The far-field potential at a position $\vec{r}$ outside cell f due to all polarizable dipoles inside the cell is:

$$\begin{aligned} V_f^{far}(\vec{r}) =& \frac{\sum_{\alpha} A_\alpha R_\alpha}{R^3} + \frac{\sum_{\alpha\beta} B_{\alpha\beta}\left[3R_\alpha R_\beta - \delta_{\alpha\beta}R^2\right]}{R^5} \\ &+ \frac{3}{2}\frac{\sum_{\alpha\beta\gamma} C_{\alpha\beta\gamma}\left[5R_\alpha R_\beta R_\gamma - (R_\alpha\delta_{\beta\gamma} + R_\beta\delta_{\alpha\gamma} + R_\gamma\delta_{\alpha\beta})R^2\right]}{R^7} + \cdots \end{aligned} \tag{2.27}$$

with the corresponding multipole moments of the charges and dipoles at the deepest level:

$$\begin{aligned} A_\alpha &= \sum_{i\in f} \mu_{i\alpha} \\ B_{\alpha\beta} &= \sum_{i\in f} \mu_{i\alpha} r_{i\beta} \\ C_{\alpha\beta\gamma} &= \sum_{i\in f} \mu_{i\alpha} r_{i\beta} r_{i\gamma} \end{aligned} \tag{2.28}$$

and $\vec{R} = \vec{r} - \vec{r}_f$, $\vec{r}_f$ is the center of cell f, R is the magnitude of $\vec{R}$, and α, β, γ are any Cartesian coordinates, x, y, or z. These lower cell multipole moments are translated to upper cell multipole moments in the upward pass using the following equations:

$$\begin{aligned} A_\alpha^{(l-1)} &= \sum_{k=1}^{children} A_{\alpha k}^{(l)} \\ B_{\alpha\beta}^{(l-1)} &= \sum_{k=1}^{children} \left(B_{\alpha\beta}^{(l)} + A_\alpha^{(l)} R_{\beta k}\right) \\ C_{\alpha\beta\gamma}^{(l-1)} &= \sum_{k=1}^{children} \left(C_{\alpha\beta\gamma}^{(l)} + B_{\alpha\beta}^{(l)} R_{\gamma k} + B_{\alpha\gamma}^{(l)} R_{\beta k} + A_\alpha^{(l)} R_{\beta k} R_{\gamma k}\right) \end{aligned} \tag{2.29}$$

The multipole expansion can be rewritten as a Taylor series expansion in

the following way:

$$V_f^T(\vec{r}) = (V_f)_0 + \sum_\alpha \left(\frac{\partial V_f}{\partial r_\alpha}\right)_0 r_\alpha + \frac{1}{2}\sum_{\alpha\beta}\left(\frac{\partial^2 V_f}{\partial r_\alpha \partial r_\beta}\right)_0 r_\alpha r_\beta + \cdots$$

Given the Taylor series of the potential, the Taylor series coefficients for $\vec{E}_f^{far}(\vec{r})$ can be found by using the definition $\vec{E} = -\vec{\nabla}V$, expanding the terms of the multipole expansion as power series, and adding the corresponding terms together. Each term of the multipole expansion is then expanded into a Taylor series at each cell. Thus,

$$\vec{E}_f^{far}(\vec{r}) = \begin{bmatrix} -\frac{\partial V_f^T}{\partial x} \\ -\frac{\partial V_f^T}{\partial y} \\ -\frac{\partial V_f^T}{\partial z} \end{bmatrix} = \begin{bmatrix} -\left(\frac{\partial V_f}{\partial r_x}\right)_0 - \sum_\alpha \left(\frac{\partial^2 V_f}{\partial r_\alpha \partial r_x}\right)_0 r_\alpha - \frac{1}{2}\sum_{\alpha\beta}\left(\frac{\partial^3 V_f}{\partial r_\alpha \partial r_\beta \partial r_x}\right)_0 r_\alpha r_\beta \\ -\left(\frac{\partial V_f}{\partial r_y}\right)_0 - \sum_\alpha \left(\frac{\partial^2 V_f}{\partial r_\alpha \partial r_y}\right)_0 r_\alpha - \frac{1}{2}\sum_{\alpha\beta}\left(\frac{\partial^3 V_f}{\partial r_\alpha \partial r_\beta \partial r_y}\right)_0 r_\alpha r_\beta \\ -\left(\frac{\partial V_f}{\partial r_z}\right)_0 - \sum_\alpha \left(\frac{\partial^2 V_f}{\partial r_\alpha \partial r_z}\right)_0 r_\alpha - \frac{1}{2}\sum_{\alpha\beta}\left(\frac{\partial^3 V_f}{\partial r_\alpha \partial r_\beta \partial r_z}\right)_0 r_\alpha r_\beta \end{bmatrix}$$

Expanding the dipole term in terms of $\vec{r}$ yields:

$$V_f^A = \frac{\sum_\alpha A_\alpha R_\alpha}{R^3} \tag{2.30}$$

$$\begin{aligned} \left(\frac{\partial V_f^A}{\partial r_x}\right)_0 &= \frac{A_x}{r_f^3} - \frac{3(\vec{A}\cdot\vec{r}_f)r_{f,x}}{r_f^5} \\ \left(\frac{\partial^2 V_f^A}{\partial r_x \partial r_y}\right)_0 &= \frac{3r_{f,y}A_x}{r_f^5} + \frac{3A_y r_{f,x}}{r_f^5} - \frac{15(\vec{A}\cdot\vec{r}_f)r_{f,x}r_{f,y}}{r_f^7} \\ \left(\frac{\partial^2 V_f^A}{\partial r_x r_x}\right)_0 &= \frac{6A_x r_{f,x}}{r_f^5} - \frac{12(\vec{A}\cdot\vec{r}_f)r_{f,x}^2}{r_f^7} \end{aligned} \tag{2.31}$$

$$
\begin{aligned}
\left(\frac{\partial^3 V_f^A}{\partial r_x \partial r_y \partial r_z}\right)_0 =& \frac{15}{r_f^7}(A_x r_{f,y} r_{f,z} + A_y r_{f,x} r_{f,z} + A_z r_{f,x} r_{f,y}) - \frac{105}{r_f^9}(\vec{A}\cdot\vec{r}_f) r_{f,x} r_{f,y} r_{f,z} \\
\left(\frac{\partial^3 V_A}{\partial r_x \partial r_y \partial r_y}\right)_0 =& \frac{3}{r_f^7}(5A_x r_{f,y} r_{f,y} + 5A_y r_{f,x} r_{f,y} + A_y r_{f,x} r_{f,y} - r_f^2 A_x + 5(\vec{A}\cdot\vec{r}_f) r_{f,x}) \\
&+ \frac{105}{r_f^9}(\vec{A}\cdot\vec{r}_f) r_{f,x}) - \frac{105}{r_f^9}(\vec{A}\cdot\vec{r}_f) r_{f,x} r_{f,y} r_{f,z} \\
\left(\frac{\partial^3 V_f^A}{\partial r_x \partial r_x \partial r_x}\right)_0 =& \frac{3}{r_f^7}(15A_x r_{f,x}^2 - 3A_x r_f^2 + 10(\vec{A}\cdot\vec{r}_f) r_{f,x}) + \frac{105}{r_f^9}(\vec{A}\cdot\vec{r}_f) r_{f,x}^3
\end{aligned}
\tag{2.32}
$$

All of these Taylor series coefficients will have the same form if any Cartesian direction x, y, or z is substituted into the formula. They are also symmetric with respect to the order of differentiation. The quadrupole and octopole terms can also be expanded in the same way. Due to the length of these expressions, we include them as well as their derivations in the appendix.

To compute the complete Taylor series coefficients, the A, B, and C terms are added together, for example:

$$
\left(\frac{\partial V_f}{\partial r_x}\right)_0 = \left(\frac{\partial V_f^A}{\partial r_x}\right)_0 + \left(\frac{\partial V_f^B}{\partial r_x}\right)_0 + \left(\frac{\partial V_f^C}{\partial r_x}\right)_0
$$

In the downward pass, the Taylor series coefficients are shifted by $\vec{r}_0$ from the parent cells to the child cells. These Taylor coefficients are then shifted to deeper levels in the downward pass:

$$\begin{aligned}\vec{E}_f^T(\vec{r}+\vec{r}_0) &= -\vec{\nabla}V_{A_0}^T(\vec{r}+\vec{r}_0) \\ &= -\vec{\nabla}V^{(0)} + \sum_{\alpha} -\vec{\nabla}V_{\alpha}^{(1)}(r_\alpha + r_{0,\alpha}) \\ &\quad + \sum_{\alpha\beta} -\vec{\nabla}V_{\alpha\beta}^{(2)}(r_\alpha + r_{0,\alpha})(r_\beta + r_{0,\beta})\end{aligned} \tag{2.33}$$

Each component of the electric field E_γ, where $\gamma = x$, y, or z can be rewritten as a polynomial in terms of r_x, r_y, and r_z:

$$\begin{aligned}E_{f,\gamma}^T(\vec{r}+\vec{r_0}) &= \left(\frac{-\partial V_f}{\partial r_\gamma}\right)_0 + \sum_\alpha \left(\frac{-\partial^2 V_f}{\partial r_\gamma \partial r_\alpha}\right)_0 (r_\alpha + r_{0,\alpha}) \\ &\quad + \sum_{\alpha\beta}\left(\frac{-\partial^3 V_f}{\partial r_\gamma \partial r_\alpha \partial r_\beta}\right)_0 (r_\alpha + r_{0,\alpha})(r_\beta + r_{0,\beta}) \\ &= \left[\left(\frac{-\partial V_f}{\partial r_\gamma}\right)_0 + \sum_\alpha \left(\frac{-\partial^2 V_f}{\partial r_\gamma \partial r_\alpha}\right)_0 r_{0,\alpha} + \sum_{\alpha\beta}\left(\frac{-\partial^3 V_f}{\partial r_\gamma \partial r_\alpha \partial r_\beta}\right)_0 r_{0,\alpha} r_{0,\beta}\right] \\ &\quad + \sum_\alpha \left(\left[\left(\frac{-\partial^2 V_f}{\partial r_\gamma \partial r_\alpha}\right)_0 + 2\sum_\beta \left(\frac{-\partial^3 V_f}{\partial r_\gamma \partial r_\alpha \partial r_\beta}\right)_0 r_{0,\beta}\right] r_\alpha\right) \\ &\quad + \sum_{\alpha\beta}\left(\left[\left(\frac{-\partial^3 V}{\partial r_\gamma \partial r_\alpha \partial r_\beta}\right)_0\right] r_\alpha r_\beta\right)\end{aligned} \tag{2.34}$$

At the deepest level, the total interaction between atoms in the system is the sum of far field and near field contributions for each atom: $\vec{E} = \vec{E}_{near} + \vec{E}_{far}$. The electric field computed by the cell multipole method is then inserted into equation (1.1) to compute the induced dipoles of the system.

CMMm for Dipolar Systems

The matrix formulation for CMM field terms for dipolar systems is similar to the matrix-based CMM for permanent charge systems in the previous section. The following matrix operations are performed to calculate the CMM electric field for a system of point dipoles:

1. $\mathbf{R}$ is the dipole-to-multipole matrix. The matrix-vector product $\mathbf{R}\vec{\mu}$ converts the charges of the individual atoms into cell multipoles at the deepest level of the tree, where $\vec{\mu}$ is the vector containing all the atom charges.

2. $\mathbf{U}^{(l-1)}$ is the lower-to-upper multipole translation matrix, representing the CMM upward pass. The product $\mathbf{U}^{(l-1)}(\mathbf{R}\vec{\mu})$ converts the level l cell multipoles into level $l-1$ cell multipoles.

3. $\mathbf{T}^{(l-1)}$ is the multipole to Taylor series matrix. The product $\mathbf{T}^{(l-1)}(\mathbf{U}^{(l-1)}\mathbf{R}\vec{\mu})$ converts the level $l-1$ cell multipoles into level $l-1$ Taylor series coefficients.

4. $\mathbf{D}^{(l-1)}$ is the upper-to-lower Taylor series translation matrix, representing the CMM downward pass. The product $\mathbf{D}^{(l-1)}(\mathbf{T}^{(l-1)}\mathbf{U}^{(l-1)}\mathbf{R}\vec{\mu})$ converts the level $l-1$ Taylor series coefficients into level l Taylor series coefficients.

5. $\mathbf{T}^{(l)}$ is the level l multipole to Taylor series matrix. The product $\mathbf{T}^{(l)}(\mathbf{R}\vec{\mu})$ converts the level l cell multipoles into level l Taylor series coefficients.

6. Adding $\mathbf{D}^{(l-1)}\mathbf{T}^{(l-1)}\mathbf{U}^{(l-1)}\mathbf{R}\vec{\mu}+\mathbf{T}^{(l)}\mathbf{R}\vec{\mu} = (\mathbf{U}^{(l-1)}\mathbf{T}^{(l-1)}\mathbf{U}^{(l-1)}+\mathbf{T}^{(l)})\mathbf{R}\vec{\mu}$ combines the component of the far field calculated at level $l-1$ with the component calculated at level l.

7. $\mathbf{R}^{\top}$ is the transpose of the charge-to-multipole matrix, which is used to plug the locations of the target atoms into the Taylor series expansions. Thus the far-field component of the electric field can be written as $\vec{E}_{far} = \mathbf{R}^{\top}(\mathbf{D}^{(l-1)}\mathbf{T}^{(l-1)}\mathbf{U}^{(l-1)} + \mathbf{T}^{(l)})\mathbf{R}\vec{\mu}$.

8. For each additional level of the matrix, the higher level $\mathbf{T}^{(l-1)}$ is replaced with a nested computation of steps 2 through 6, for example:

$$\vec{E}_{far} = \mathbf{R}^{\top}\left(\mathbf{D}^{(l-1)}\left(\mathbf{D}^{(l-2)}\mathbf{T}^{(l-2)}\mathbf{U}^{(l-2)} + \mathbf{T}^{(l-1)}\right)\mathbf{U}^{(l-1)} + \mathbf{T}^{(l)}\right)\mathbf{R}\vec{\mu}.$$

Single-level CMMm for Dipolar Systems in One Dimension

We again demonstrate the structure for this system with a 1-D example of 16 atoms, corresponding to 4 cells with 4 atoms per cell. The particle-particle interactions of this system are represented in a 16×16 matrix $\mathbf{A}$ in this case, where each element A_{ij} is the electrostatic interaction between source atom j and target atom i. This can be divided into 4x4 blocks where each block $\mathbf{A}_{IJ}$ represents the interaction between two cells in the deepest level of the tree structure: source cell J and target cell I.

This block matrix is separated into near-field and far-field interactions $\mathbf{A} =$

$\mathbf{A}_{near} + \mathbf{A}_{far}$, where the near field of a target atom includes atoms within the same cell and neighboring cells. Thus, the blocks of matrix $\mathbf{A}$ that correspond to the near field are the diagonal blocks $\mathbf{A}_{I,I}$ (interactions within the same cell) and the blocks next to the diagonal $\mathbf{A}_{I,I-1}$ and $\mathbf{A}_{I,I+1}$ (interactions with neighboring cells). The interactions within the near field are computed exactly, as in the standard CMM approach.

Now we will focus on a particular block that corresponds to far-field interactions. Block $\mathbf{A}_{1,4}$ represents the far-field interaction between source cell 4 and target cell 1.

$$E_{1,4} = \begin{bmatrix} A_{1,13} & A_{1,14} & A_{1,15} & A_{1,16} \\ A_{2,13} & A_{2,14} & A_{2,15} & A_{2,16} \\ A_{3,13} & A_{3,14} & A_{3,15} & A_{3,16} \\ A_{4,13} & A_{4,14} & A_{4,15} & A_{4,16} \end{bmatrix} \begin{bmatrix} \mu_{13} \\ \mu_{14} \\ \mu_{15} \\ \mu_{16} \end{bmatrix}$$

The expression for E for an atom i in cell 1 due to all atoms in cell 4 is:

$$\begin{aligned} E(r_i - r_{c_4}) = & \left(\frac{2}{|r_{c_4}|^3} - \frac{6r_{c_4}}{|r_{c_4}|^5} r_i + \frac{24}{|r_{c_4}|^5} r_i^2 \right) A_x \\ & + \left(\frac{-6r_{c_4}}{|r_{c_4}|^5} + \frac{24}{|r_{c_4}|^5} r_i - \frac{120 r_{c_4}}{|r_{c_4}|^7} r_i^2 \right) B_{xx} \\ & + \left(\frac{12}{|r_{c_4}|^5} - \frac{60 r_{c_4}}{|r_{c_4}|^7} r_i + \frac{360}{|r_{c_4}|^7} r_i^2 \right) C_{xxx}. \end{aligned}$$

where r_{c_4} is the center of the source cell 4. This can be rewritten in terms of the

individual particle dipoles μ_j using the definitions of A, B, and C:

$$\begin{aligned} E(r_i - r_{c_4}) = &\left(\frac{2}{|r_{c_4}|^3} - \frac{6r_{c_4}}{|r_{c_4}|^5} r_i + \frac{24}{|r_{c_4}|^5} r_i^2 \right) \sum_{j=13}^{16} \mu_j \\ &+ \left(\frac{-6r_{c_4}}{|r_{c_4}|^5} + \frac{24}{|r_{c_4}|^5} r_i - \frac{120 r_{c_4}}{|r_{c_4}|^7} r_i^2 \right) \sum_{j=13}^{16} \mu_j r_j \\ &+ \left(\frac{12}{|r_{c_4}|^5} - \frac{60 r_{c_4}}{|r_{c_4}|^7} r_i + \frac{36}{|r_{c_4}|^7} r_i^2 \right) \sum_{j=13}^{16} \mu_j r_j^2. \end{aligned}$$

Then μ_j can be factored out of the equation and it can be written as one summation.

$$\begin{aligned} E(r_i - r_{c_4}) = \sum_{j=13}^{16} \Bigg[&\left(\frac{2}{|r_{c_4}|^3} - \frac{6r_{c_4}}{|r_{c_4}|^5} r_i + \frac{24}{|r_{c_4}|^5} r_i^2 \right) \\ &+ \left(\frac{-6r_{c_4}}{|r_{c_4}|^5} + \frac{24}{|r_{c_4}|^5} r_i - \frac{120 r_{c_4}}{|r_{c_4}|^7} r_i^2 \right) r_j \\ &+ \left(\frac{12}{|r_{c_4}|^5} - \frac{60 r_{c_4}}{|r_{c_4}|^7} r_i + \frac{36}{|r_{c_4}|^7} r_i^2 \right) r_j^2 \Bigg] \mu_j. \end{aligned}$$

Block $\mathbf{A}_{1,4}$ can be decomposed into a product of three matrices in the following way:

$$\mathbf{A}_{1,4} = \begin{bmatrix} 1 & r_1 & r_1^2 \\ 1 & r_2 & r_2^2 \\ 1 & r_3 & r_3^2 \\ 1 & r_4 & r_4^2 \end{bmatrix} \begin{bmatrix} \frac{2}{|r_{c_4}|^3} & \frac{-6r_{c_4}}{|r_{c_4}|^5} & \frac{12}{|r_{c_4}|^5} \\ \frac{-6r_{c_4}}{|r_{c_4}|^5} & \frac{24}{|r_{c_4}|^5} & \frac{-60r_{c_4}}{|r_{c_4}|^7} \\ \frac{24}{|r_{c_4}|^5} & \frac{-120r_{c_4}}{|r_{c_4}|^7} & \frac{36}{|r_{c_4}|^7} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ r_{13} & r_{14} & r_{15} & r_{16} \\ r_{13}^2 & r_{14}^2 & r_{15}^2 & r_{16}^2 \end{bmatrix},$$

or $\mathbf{A}_{1,4} = \mathbf{R}_1^\top * \mathbf{T}_4 * \mathbf{R}_4$, where $\mathbf{R}$ contains values of r for the atoms in a given cell, $\mathbf{T}_4$ is block of the multipole-to-Taylor series matrix corresponding to cell 4, and $\mathbf{R}_4^\top$ is the transpose of $\mathbf{R}_4$. Each block of $\mathbf{A}$ can be decomposed in a similar manner. $\mathbf{T}_4$, or more precisely r_{c4}, depends on the location of the cell center, so

it can be computed once the size of the bounding box for the system is known. In this case, each $\mathbf{R}_I$ block is 3x4, but in general, the number of columns in $\mathbf{R}_I$ is the number of atoms in cell I.

The block decomposition of $\mathbf{A}_{far}$ for the four-cell case is:

$$\begin{bmatrix} \mathbf{R}_1^\top & 0 & 0 & 0 \\ 0 & \mathbf{R}_2^\top & 0 & 0 \\ 0 & 0 & \mathbf{R}_3^\top & 0 \\ 0 & 0 & 0 & \mathbf{R}_4^\top \end{bmatrix} \begin{bmatrix} 0 & 0 & \mathbf{T}_3 & \mathbf{T}_4 \\ 0 & 0 & 0 & \mathbf{T}_4 \\ \mathbf{T}_1 & 0 & 0 & 0 \\ \mathbf{T}_1 & \mathbf{T}_2 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R}_1 & 0 & 0 & 0 \\ 0 & \mathbf{R}_2 & 0 & 0 \\ 0 & 0 & \mathbf{R}_3 & 0 \\ 0 & 0 & 0 & \mathbf{R}_4 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & \mathbf{R}_1^\top \mathbf{T}_3 \mathbf{R}_3 & \mathbf{R}_1^\top \mathbf{T}_4 \mathbf{R}_4 \\ 0 & 0 & 0 & \mathbf{R}_2^\top \mathbf{T}_4 \mathbf{R}_4 \\ \mathbf{R}_3^\top \mathbf{T}_1 \mathbf{R}_1 & 0 & 0 & 0 \\ \mathbf{R}_4^\top \mathbf{T}_1 \mathbf{R}_1 & \mathbf{R}_4^\top \mathbf{T}_2 \mathbf{R}_2 & 0 & 0 \end{bmatrix}$$

The four-cell case involves only one level of the tree structure.

Hierarchical CMMm for Dipolar Systems in One Dimension

To illustrate the hierarchical algorithm with upward and downward passes, we will now consider a system with 32 atoms, corresponding to 8 cells and four atoms per cell. The multipoles for cell J at the deepest level are found by the

following equation:

$$\mathbf{R}_J \cdot \vec{\mu}_J = \begin{bmatrix} A_{J,x} \\ B_{J,xx} \\ C_{J,xxx} \end{bmatrix}$$

where $\vec{\mu}_J$ is the block of the original $\vec{\mu}$ vector corresponding to cell J. The entire vector $\vec{\mu}$ is decomposed as:

$$\vec{\mu} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \\ \mu_5 \\ \mu_6 \\ \mu_7 \\ \mu_8 \end{bmatrix}$$

The multipoles of higher levels are found by translating and adding child cells, accordingly:

$$A^{(l-1)}_{cell,x} = \sum_{k \in children} A^{(l)}_k,$$
$$B^{(l-1)}_{cell,xx} = \sum_{k \in children} B^{(l)}_{k,xx} + A^{(l)}_k R_k,$$
$$C^{(l-1)}_{cell,xxx} = \sum_{k \in children} C^{(l)}_{k,xxx} + 2B^{(l)}_k R_k + A^{(l)}_k R^2_k,$$

where R_k is the difference between the center of the higher-level cell to the center of the lower-level cell, l is the lower level, and $l-1$ is the higher level.

The transfer operator for the upward pass from level l to upper level $l-1$, $\mathbf{U}_J^{(l-1)}$ can be written as follows, where J is the index of the parent cell at level $l-1$:

$$\mathbf{U}_J^{(l-1)} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ \vec{R}_{2J-1} & 1 & 0 & \vec{R}_{2J} & 1 & 0 \\ \vec{R}_{2J-1}^2 & 2\vec{R}_{2J-1} & 1 & \vec{R}_{2J}^2 & 2\vec{R}_{2J} & 1 \end{bmatrix}.$$

$\mathbf{U}_J^{(l-1)}$ operates in the following manner:

$$\mathbf{U}_J^{(l-1)} \cdot \begin{bmatrix} A^l_{2J-1,x} \\ B^l_{2J-1,xx} \\ C^l_{2J-1,xxx} \\ A^l_{2J,x} \\ B^l_{2J,xx} \\ C^l_{2J,xxx} \end{bmatrix} = \begin{bmatrix} A^{l-1}_{J,x} \\ B^{l-1}_{J,xx} \\ C^{l-1}_{J,xxx} \end{bmatrix}.$$

The entire matrix $\mathbf{U}^{(l-1)}$ is composed of the individual $\mathbf{U}_J^{(l-1)}$ blocks:

$$\mathbf{U}^{(l-1)} = \begin{bmatrix} \mathbf{U}_1^{(l-1)} & 0 & 0 & 0 \\ 0 & \mathbf{U}_2^{(l-1)} & 0 & 0 \\ 0 & 0 & \mathbf{U}_3^{(l-1)} & 0 \\ 0 & 0 & 0 & \mathbf{U}_4^{(l-1)} \end{bmatrix}.$$

At the highest level with far field contributions (level 2), the multipole-to-

Taylor matrix is given as:

$$\mathbf{T}^{(2)} = \begin{bmatrix} 0 & 0 & \mathbf{T}_3^{(2)} & \mathbf{T}_4^{(2)} \\ 0 & 0 & 0 & \mathbf{T}_4^{(2)} \\ \mathbf{T}_1^{(2)} & 0 & 0 & 0 \\ \mathbf{T}_1^{(2)} & \mathbf{T}_2^{(2)} & 0 & 0 \end{bmatrix} \tag{2.35}$$

The lower-level (level 3) multipole-to-Taylor matrix computes the interactions that were not accounted for by the parent cells at level 2.

$$\mathbf{T}^{(3)} = \begin{bmatrix} 0 & 0 & \mathbf{T}_3^{(3)} & \mathbf{T}_4^{(3)} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{T}_4^{(3)} & 0 & 0 & 0 & 0 \\ \mathbf{T}_1^{(3)} & 0 & 0 & 0 & \mathbf{T}_5^{(3)} & \mathbf{T}_6^{(3)} & 0 & 0 \\ \mathbf{T}_1^{(3)} & \mathbf{T}_2^{(3)} & 0 & 0 & 0 & \mathbf{T}_6^{(3)} & 0 & 0 \\ 0 & 0 & \mathbf{T}_3^{(3)} & 0 & 0 & 0 & \mathbf{T}_7^{(3)} & \mathbf{T}_8^{(3)} \\ 0 & 0 & \mathbf{T}_3^{(3)} & \mathbf{T}_4^{(3)} & 0 & 0 & 0 & \mathbf{T}_8^{(3)} \\ 0 & 0 & 0 & 0 & \mathbf{T}_5^{(3)} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{T}_5^{(3)} & \mathbf{T}_6^{(3)} & 0 & 0 \end{bmatrix} \tag{2.36}$$

In the downward pass, the Taylor series for the electric field is shifted from parent cells to child cells:

$$E_{A_0}^T(r + r_0) = E^{(0)} + E^{(1)}(r + r_0) + E^{(2)}(r + r_0)^2$$

Rearranging terms, the expression can be written as a polynomial in $\vec{r}$:

$$\begin{aligned} E_{A_0}^T(r + r_0) = &\left(E^{(0)} + E^{(1)}r_0 + E^{(2)}r_0^2\right) \\ &+ \left(E^{(1)} + 2E^{(2)}r_0\right) r \\ &+ E^{(2)}r^2 \end{aligned}$$

The coefficients on the powers of $\vec{r}$ can be stored in a matrix-vector product in the following way:

$$\begin{bmatrix} 1 & r_0 & r_0^2 \\ 0 & 1 & 2r_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} E^{(0)} \\ E^{(1)} \\ E^{(2)} \end{bmatrix} = \begin{bmatrix} E^{(0)} + E^{(1)}r_0 + E^{(2)}r_0^2 \\ E^{(1)}r_0 + 2E^{(2)}r_0 \\ E^{(2)} \end{bmatrix}$$

The matrix on the left is denoted $\mathbf{D}_j^{(l-1)}$, where l is the level of the child cells, and j refers to a particular child cell. The entire matrix $\mathbf{D}^{(l-1)}$ can be written in the following block form:

$$\mathbf{D}^{(2)} = \begin{bmatrix} \mathbf{D}_1^{(2)} & 0 & 0 & 0 \\ \mathbf{D}_2^{(2)} & 0 & 0 & 0 \\ 0 & \mathbf{D}_3^{(2)} & 0 & 0 \\ 0 & \mathbf{D}_4^{(2)} & 0 & 0 \\ 0 & 0 & \mathbf{D}_5^{(2)} & 0 \\ 0 & 0 & \mathbf{D}_6^{(2)} & 0 \\ 0 & 0 & 0 & \mathbf{D}_7^{(2)} \\ 0 & 0 & 0 & \mathbf{D}_8^{(2)} \end{bmatrix}$$

Hierarchical CMMm for Dipolar Systems in Three Dimensions

We can now imagine the system to be contained in a three-dimensional cubic box. For simplicity, we will assume that the number of cells n_c in the system is a perfect cube (the system is $\sqrt[3]{n_c} \times \sqrt[3]{n_c} \times \sqrt[3]{n_c}$). We will further assume that $\sqrt[3]{n_c}$ is a power of 2, so that the child cells will be evenly divided into parent cells at each step of the upward and downward passes. Thus, the number of cells at level l of the system is $n_c^{(l)} = 8^l$.

Indexing cells. In three dimensions, it is not as easy to visualize the matrix elements. These matrices remain sparse, but their structures are much more complex. The way in which the cells are indexed plays a larger role in the efficiency of implementation. There are many ways to index the cells. Here, each cell is given an index number $< i, j, k >$ starting with $< 1, 1, 1 >$ at one corner of the 3-D box and counting in the x, y, and z directions respectively. The following would be the simplest ordering of the cells in the case of a system of 512 total cells: $< 1, 1, 1 >, < 1, 1, 2 >, < 1, 1, 3 >, < 1, 1, 4 >, < 1, 1, 5 >, < 1, 1, 6 >, < 1, 1, 7 >,$ $< 1, 1, 8 >, < 1, 2, 1 >, < 1, 2, 2 >, < 1, 2, 3 >, < 1, 2, 4 >, \ldots < 1, 8, 7 >,$ $< 1, 8, 8 >, < 2, 1, 1 >, < 2, 1, 2 >, \ldots , < 8, 8, 7 >, < 8, 8, 8 >$. This ordering was chosen for ease of determining which cell each atom belongs in. Further formulation in the ordering of atoms to represent nearby cells in the hierarchy will increase the efficiency of the data structure and enhance parallelizability. The

child cells of cell $< i, j, k >$ at level l are the following 8 cells at level $l + 1$: $< (2i-1), (2j-1), (2k-1) >$, $< (2i-1), (2j-1), 2k >$, $< (2i-1), 2j, (2k-1) >$, $< (2i-1), 2j, 2k >$, $< 2i, (2j-1), (2k-1) >$, $< 2i, (2j-1), 2k >$, $< 2i, 2j, (2k-1) >$, and $< 2i, 2j, 2k >$.

Upward Pass. The dimensions of the vector representing the cell multipoles of a cell I correspond to the following:

In 1-D, the vector is 3×1: $[A_{I,x}, B_{I,xx}, C_{I,xxx}]^\top$.

In 2-D, the vector is 14×1: $[A_{I,x},\ A_{I,y},\ B_{I,xx},\ B_{I,xy},\ B_{I,yx},\ B_{I,yy},\ C_{I,xxx},\ C_{I,xxy},\ C_{I,xyx},\ C_{I,xyy}, C_{I,yxx},\ C_{I,yxy},\ C_{I,yyx},\ C_{I,yyy}]^\top$.

In 3-D case, the vector is 39×1: $[A_{I,x},\ A_{I,y},\ A_{I,z},\ B_{I,xx},\ B_{I,xy},\ B_{I,xz},\ B_{I,yx},\ B_{I,yy},\ B_{I,yz},\ B_{I,zx},\ B_{I,zy},\ B_{I,zz},\ C_{I,xxx},\ C_{I,xxy},\ C_{I,xxz},\ C_{I,xyx},\ C_{I,xyy},\ C_{I,xyz},\ C_{I,xzx},\ C_{I,xzy}, C_{I,xzz}, C_{I,yxx}, C_{I,yxy}, C_{I,yxz}, C_{I,yyx}, C_{I,yyy}, C_{I,yyz}, C_{I,yzx}, C_{I,yzy}, C_{I,yzz}, C_{I,zxx},\ C_{I,zxy},\ C_{I,zxz},\ C_{I,zyx},\ C_{I,zyy},\ C_{I,zyz},\ C_{I,zzx},\ C_{I,zzy},\ C_{I,zzz}]^\top$.

This vector does not appear explicitly in the CMM matrix decomposition, but is implicitly present as the product $\mathbf{R}\vec{\mu}$, where $\vec{\mu}$ is the initial vector containing the dipoles of all atoms in the system:

$$\vec{\mu} = \begin{bmatrix} \vec{\mu}_1 \\ \vec{\mu}_2 \\ \vdots \\ \vec{\mu}_n \end{bmatrix}$$

and $\vec{\mu}_i = [\mu_{i,x}, \mu_{i,y}, \mu_{i,z}]^\top$ is the point dipole of atom i. Like the one-dimensional

version of the method, the dipole-to-multipole matrix $\mathbf{R}$ has one block per cell at the deepest level. Thus, the matrix $\mathbf{R}$ is $39n_c \times 3n$, where n_c is the number of cells and n is the number of atoms. Assuming four atoms per cell, block $\mathbf{R}_1$ of $\mathbf{R}$ has this structure:

$$\mathbf{R}_1 = \begin{bmatrix}
1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\
r_{1x} & 0 & 0 & r_{2x} & 0 & 0 & r_{3x} & 0 & 0 & r_{4x} & 0 & 0 \\
r_{1y} & 0 & 0 & r_{2y} & 0 & 0 & r_{3y} & 0 & 0 & r_{4y} & 0 & 0 \\
r_{1z} & 0 & 0 & r_{2z} & 0 & 0 & r_{3z} & 0 & 0 & r_{4z} & 0 & 0 \\
0 & r_{1x} & 0 & 0 & r_{2x} & 0 & 0 & r_{3x} & 0 & 0 & r_{4x} & 0 \\
0 & r_{1y} & 0 & 0 & r_{2y} & 0 & 0 & r_{3y} & 0 & 0 & r_{4y} & 0 \\
0 & r_{1z} & 0 & 0 & r_{2z} & 0 & 0 & r_{3z} & 0 & 0 & r_{4z} & 0 \\
0 & 0 & r_{1x} & 0 & 0 & r_{2x} & 0 & 0 & r_{3x} & 0 & 0 & r_{4x} \\
0 & 0 & r_{1y} & 0 & 0 & r_{2y} & 0 & 0 & r_{3y} & 0 & 0 & r_{4y} \\
0 & 0 & r_{1z} & 0 & 0 & r_{2z} & 0 & 0 & r_{3z} & 0 & 0 & r_{4z} \\
r_{1x}r_{1x} & 0 & 0 & r_{2x}r_{2x} & 0 & 0 & r_{3x}r_{3x} & 0 & 0 & r_{4x}r_{4x} & 0 & 0 \\
r_{1x}r_{1y} & 0 & 0 & r_{2x}r_{2y} & 0 & 0 & r_{3x}r_{3y} & 0 & 0 & r_{4x}r_{4y} & 0 & 0 \\
r_{1x}r_{1z} & 0 & 0 & r_{2x}r_{2z} & 0 & 0 & r_{3x}r_{3z} & 0 & 0 & r_{4x}r_{4z} & 0 & 0 \\
r_{1y}r_{1x} & 0 & 0 & r_{2y}r_{2x} & 0 & 0 & r_{3y}r_{3x} & 0 & 0 & r_{4y}r_{4x} & 0 & 0 \\
r_{1y}r_{1y} & 0 & 0 & r_{2y}r_{2y} & 0 & 0 & r_{3y}r_{3y} & 0 & 0 & r_{4y}r_{4y} & 0 & 0 \\
r_{1y}r_{1z} & 0 & 0 & r_{2y}r_{2z} & 0 & 0 & r_{3y}r_{3z} & 0 & 0 & r_{4y}r_{4z} & 0 & 0 \\
r_{1z}r_{1x} & 0 & 0 & r_{2z}r_{2x} & 0 & 0 & r_{3z}r_{3x} & 0 & 0 & r_{4z}r_{4x} & 0 & 0 \\
r_{1z}r_{1y} & 0 & 0 & r_{2z}r_{2y} & 0 & 0 & r_{3z}r_{3y} & 0 & 0 & r_{4z}r_{4y} & 0 & 0 \\
r_{1z}r_{1z} & 0 & 0 & r_{2z}r_{2z} & 0 & 0 & r_{3z}r_{3z} & 0 & 0 & r_{4z}r_{4z} & 0 & 0 \\
0 & r_{1x}r_{1x} & 0 & 0 & r_{2x}r_{2x} & 0 & 0 & r_{3x}r_{3x} & 0 & 0 & r_{4x}r_{4x} & 0 \\
0 & r_{1x}r_{1y} & 0 & 0 & r_{2x}r_{2y} & 0 & 0 & r_{3x}r_{3y} & 0 & 0 & r_{4x}r_{4y} & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & r_{1z}r_{1z} & 0 & 0 & r_{2z}r_{2z} & 0 & 0 & r_{3z}r_{3z} & 0 & 0 & r_{4z}r_{4z} & 0 \\
0 & 0 & r_{1x}r_{1x} & 0 & 0 & r_{2x}r_{2x} & 0 & 0 & r_{3x}r_{3x} & 0 & 0 & r_{4x}r_{4x} \\
0 & 0 & r_{1x}r_{1y} & 0 & 0 & r_{2x}r_{2y} & 0 & 0 & r_{3x}r_{3y} & 0 & 0 & r_{4x}r_{4y} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & r_{1z}r_{1z} & 0 & 0 & r_{2z}r_{2z} & 0 & 0 & r_{3z}r_{3z} & 0 & 0 & r_{4z}r_{4z}
\end{bmatrix}$$

where $r_{i\alpha}$ is the α-component of the position of particle i ($\alpha = x$, y, or z). Cell $< 1,1,1 >$ was chosen as an example to simplify the subscripts on the matrix

elements, but all subblocks of $\mathbf{R}$ have the same structure. The first three rows of $\mathbf{R}_1$ are used to compute the A_α multipoles, rows 4-12 correspond to the $B_{\alpha\beta}$ multipoles, and rows 13-39 correspond to the $C_{\alpha\beta\gamma}$ multipoles.

The sparse matrix $\mathbf{U}^{(l-1)}$ that translates the level l (child) multipoles to level $l-1$ (parent) multipoles is $(39n_c^{(l-1)}) \times (39n_c^{(l)})$, where $n_c^{(l)}$ is the number of cells at level l and $n_c^{(l-1)} = \frac{n_c^{(l)}}{8}$. Since the rows of $\mathbf{U}^{(l-1)}$ correspond to the parent cells at level $l-1$, we can simplify the notation for the elements of the matrix $\mathbf{U}^{(l-1)}$ by denoting $p^{(l-1)}_{<i,j,k>} = 39(k-1) + 39(n_c^{(l-1)})^{1/3}(j-1) + 39(n_c^{(l-1)})^{2/3}(i-1)$. This is a row index referring to the section of $\mathbf{U}^{(l-1)}$ that corresponds to cell $< i, j, k >$. Similarly, the columns of $\mathbf{U}^{(l-1)}$ correspond to the child cells at level l. Therefore, the following column indices correspond to the 8 level l child cells of the level $l-1$

parent cell $< i, j, k >$:

$$
\begin{aligned}
c^{(l,1)}_{<i,j,k>} &= 39(2k-2) + 39(n_c^{(l-1)})^{1/3}(2j-2) + 39(n_c^{(l-1)})^{2/3}(2i-2) \\
c^{(l,2)}_{<i,j,k>} &= 39(2k-2) + 39(n_c^{(l-1)})^{1/3}(2j-2) + 39(n_c^{(l-1)})^{2/3}(2i-1) \\
c^{(l,3)}_{<i,j,k>} &= 39(2k-2) + 39(n_c^{(l-1)})^{1/3}(2j-1) + 39(n_c^{(l-1)})^{2/3}(2i-2) \\
c^{(l,4)}_{<i,j,k>} &= 39(2k-2) + 39(n_c^{(l-1)})^{1/3}(2j-1) + 39(n_c^{(l-1)})^{2/3}(2i-1) \\
c^{(l,5)}_{<i,j,k>} &= 39(2k-1) + 39(n_c^{(l-1)})^{1/3}(2j-2) + 39(n_c^{(l-1)})^{2/3}(2i-2) \\
c^{(l,6)}_{<i,j,k>} &= 39(2k-1) + 39(n_c^{(l-1)})^{1/3}(2j-2) + 39(n_c^{(l-1)})^{2/3}(2i-1) \\
c^{(l,7)}_{<i,j,k>} &= 39(2k-1) + 39(n_c^{(l-1)})^{1/3}(2j-1) + 39(n_c^{(l-1)})^{2/3}(2i-2) \\
c^{(l,8)}_{<i,j,k>} &= 39(2k-1) + 39(n_c^{(l-1)})^{1/3}(2j-1) + 39(n_c^{(l-1)})^{2/3}(2i-1)
\end{aligned}
\tag{2.37}
$$

The nonzero elements of $\mathbf{U}^{(l-1)}$ are defined using these indices. Using equation 2.29, lower multipoles are translated to upper multipoles. The matrix elements used to compute the parent cell multipoles $A^{(l-1)}_{<i,j,k>,x}$, $A^{(l-1)}_{<i,j,k>,y}$ and $A^{(l-1)}_{<i,j,k>,z}$ are as follows:

$$
\begin{aligned}
\mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+1\right],\left[c^{(l,m)}_{<i,j,k>}+1\right]} &= 1 \\
\mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+2\right],\left[c^{(l,m)}_{<i,j,k>}+2\right]} &= 1 \\
\mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+3\right],\left[c^{(l,m)}_{<i,j,k>}+3\right]} &= 1
\end{aligned}
\tag{2.38}
$$

where $m = 1, 2, \ldots 8$.

These elements are multiplied by the multipole moments of the child cells to generate the parent cell multipole moments. For the parent cell quadrupole terms $B^{(l-1)}_{<i,j,k>,\alpha\beta}$, there are two matrix elements corresponding to each $\alpha\beta$ pair:

$$
\begin{aligned}
B^{(l-1)}_{<i,j,k>,xx}\colon\quad & \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+4\right],\left[c^{(l,m)}_{<i,j,k>}+1\right]} = R^{(l,m)}_{<i,j,k>,x} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+4\right],\left[c^{(l,m)}_{<i,j,k>}+4\right]} = 1 \\
B^{(l-1)}_{<i,j,k>,xy}\colon\quad & \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+5\right],\left[c^{(l,m)}_{<i,j,k>}+1\right]} = R^{(l,m)}_{<i,j,k>,y} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+5\right],\left[c^{(l,m)}_{<i,j,k>}+5\right]} = 1 \\
B^{(l-1)}_{<i,j,k>,xz}\colon\quad & \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+6\right],\left[c^{(l,m)}_{<i,j,k>}+1\right]} = R^{(l,m)}_{<i,j,k>,z} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+6\right],\left[c^{(l,m)}_{<i,j,k>}+6\right]} = 1 \\
B^{(l-1)}_{<i,j,k>,yx}\colon\quad & \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+7\right],\left[c^{(l,m)}_{<i,j,k>}+2\right]} = R^{(l,m)}_{<i,j,k>,x} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+7\right],\left[c^{(l,m)}_{<i,j,k>}+7\right]} = 1 \\
B^{(l-1)}_{<i,j,k>,yy}\colon\quad & \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+8\right],\left[c^{(l,m)}_{<i,j,k>}+2\right]} = R^{(l,m)}_{<i,j,k>,y} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+8\right],\left[c^{(l,m)}_{<i,j,k>}+8\right]} = 1 \\
B^{(l-1)}_{<i,j,k>,yz}\colon\quad & \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+9\right],\left[c^{(l,m)}_{<i,j,k>}+2\right]} = R^{(l,m)}_{<i,j,k>,z} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+9\right],\left[c^{(l,m)}_{<i,j,k>}+9\right]} = 1 \\
B^{(l-1)}_{<i,j,k>,zx}\colon\quad & \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+10\right],\left[c^{(l,m)}_{<i,j,k>}+3\right]} = R^{(l,m)}_{<i,j,k>,x} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+10\right],\left[c^{(l,m)}_{<i,j,k>}+10\right]} = 1 \\
B^{(l-1)}_{<i,j,k>,zy}\colon\quad & \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+11\right],\left[c^{(l,m)}_{<i,j,k>}+3\right]} = R^{(l,m)}_{<i,j,k>,y} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+11\right],\left[c^{(l,m)}_{<i,j,k>}+11\right]} = 1 \\
B^{(l-1)}_{<i,j,k>,zz}\colon\quad & \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+12\right],\left[c^{(l,m)}_{<i,j,k>}+3\right]} = R^{(l,m)}_{<i,j,k>,z} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+12\right],\left[c^{(l,m)}_{<i,j,k>}+12\right]} = 1
\end{aligned}
\tag{2.39}
$$

where $m = 1, 2, \ldots, 8$, and $R^{(l,m)}_{<i,j,k>}$ is the vector from the parent cell to the mth child cell. The value of $R^{(l,m)}_{<i,j,k>}$ is a property of the size of the cells and depends only on l and m, so it only needs to be computed once per level for each value of m.

For the parent cell octopole terms $C^{(l-1)}_{<i,j,k>,\alpha\beta\gamma}$, there are four matrix elements for each choice of $\alpha\beta\gamma$, or only three elements when $\beta = \gamma$. These matrix

elements are as follows:

$$
\begin{aligned}
C^{(l-1)}_{<i,j,k>,xxx}\text{:}\quad & \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+13\right],\left[c^{(l,m)}_{<i,j,k>}+1\right]} = R^{(l,m)}_{<i,j,k>,x} R^{(l,m)}_{<i,j,k>,x} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+13\right],\left[c^{(l,m)}_{<i,j,k>}+4\right]} = 2R^{(l,m)}_{<i,j,k>,x} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+13\right],\left[c^{(l,m)}_{<i,j,k>}+13\right]} = 1 \\
C^{(l-1)}_{<i,j,k>,xxy}\text{:}\quad & \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+14\right],\left[c^{(l,m)}_{<i,j,k>}+1\right]} = R^{(l,m)}_{<i,j,k>,x} R^{(l,m)}_{<i,j,k>,y} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+14\right],\left[c^{(l,m)}_{<i,j,k>}+4\right]} = R^{(l,m)}_{<i,j,k>,y} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+14\right],\left[c^{(l,m)}_{<i,j,k>}+5\right]} = R^{(l,m)}_{<i,j,k>,x} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+14\right],\left[c^{(l,m)}_{<i,j,k>}+14\right]} = 1 \\
C^{(l-1)}_{<i,j,k>,xxz}\text{:}\quad & \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+15\right],\left[c^{(l,m)}_{<i,j,k>}+1\right]} = R^{(l,m)}_{<i,j,k>,x} R^{(l,m)}_{<i,j,k>,z} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+15\right],\left[c^{(l,m)}_{<i,j,k>}+4\right]} = R^{(l,m)}_{<i,j,k>,z} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+15\right],\left[c^{(l,m)}_{<i,j,k>}+6\right]} = R^{(l,m)}_{<i,j,k>,x} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+15\right],\left[c^{(l,m)}_{<i,j,k>}+15\right]} = 1 \\
C^{(l-1)}_{<i,j,k>,xyx}\text{:}\quad & \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+16\right],\left[c^{(l,m)}_{<i,j,k>}+1\right]} = R^{(l,m)}_{<i,j,k>,x} R^{(l,m)}_{<i,j,k>,z} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+16\right],\left[c^{(l,m)}_{<i,j,k>}+5\right]} = R^{(l,m)}_{<i,j,k>,x} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+16\right],\left[c^{(l,m)}_{<i,j,k>}+4\right]} = R^{(l,m)}_{<i,j,k>,y} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+16\right],\left[c^{(l,m)}_{<i,j,k>}+16\right]} = 1 \\
\vdots \quad & \quad \vdots \\
C^{(l-1)}_{<i,j,k>,zzy}\text{:}\quad & \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+38\right],\left[c^{(l,m)}_{<i,j,k>}+3\right]} = R^{(l,m)}_{<i,j,k>,z} R^{(l,m)}_{<i,j,k>,y} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+38\right],\left[c^{(l,m)}_{<i,j,k>}+12\right]} = R^{(l,m)}_{<i,j,k>,y} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+38\right],\left[c^{(l,m)}_{<i,j,k>}+11\right]} = R^{(l,m)}_{<i,j,k>,z} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+38\right],\left[c^{(l,m)}_{<i,j,k>}+38\right]} = 1 \\
C^{(l-1)}_{<i,j,k>,zzz}\text{:}\quad & \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+39\right],\left[c^{(l,m)}_{<i,j,k>}+3\right]} = R^{(l,m)}_{<i,j,k>,z} R^{(l,m)}_{<i,j,k>,z} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+39\right],\left[c^{(l,m)}_{<i,j,k>}+12\right]} = 2R^{(l,m)}_{<i,j,k>,z} \\
& \mathbf{U}^{(l-1)}_{\left[p^{(l-1)}_{<i,j,k>}+39\right],\left[c^{(l,m)}_{<i,j,k>}+39\right]} = 1
\end{aligned}
\tag{2.40}
$$

where $m = 1, 2, \ldots, 8$.

The sparsity of this matrix can be examined by comparing the number of nonzero elements to the entire size of the matrix. This will give the percentage of elements that need to be stored. The sum of children multipole moments for a given parent cell yields 960 nonzero elements in $\mathbf{U}^{(l-1)}$. Since there are 8^{l-1} parent cells at level $l-1$, $\mathbf{U}^{(l-1)}$ has a total of $960 \cdot 8^{l-1}$ nonzero elements. With a total of $(39 \cdot 8^{l-1}) \times (39 \cdot 8^{l}) = 1521 \cdot 8^{2l-1}$ elements in $\mathbf{U}^{(l-1)}$, the proportion of nonzero elements in the matrix is $\frac{320}{507 \cdot 8^{l}}$. Therefore as the size of the system increases, the matrices become substantially more sparse.

Conversion from multipole to Taylor expansion. Each subblock of the multipole to Taylor series matrix $\mathbf{T}_f^{(l-1)}$ is 39×39, and corresponds to a cell f at level $l-1$. Each row represents one Taylor series coefficient and each column represents the contribution to the Taylor series coefficient by one multipole moment. Each term of the multipole expansion is expanded in terms of $\vec{r}$. The first row of the block $\mathbf{T}_f^{(l-1)}$ is then the sum of these constant terms. Subsequently, each row represents a Taylor coefficient. An example of this is given below:

$$\left(\frac{\partial V_f^A}{\partial r_x}\right)_0 = \left(\frac{1}{r_f^3} - \frac{3r_{f,x}^2}{r_f^5}\right) A_x + \left(-\frac{3r_{f,x}r_{f,y}}{r_f^5}\right) A_y + \left(-\frac{3r_{f,x}r_{f,z}}{r_f^5}\right) A_z \tag{2.41}$$

$$\begin{aligned}\left(\frac{\partial V_f^B}{\partial r_x}\right)_0 = &\left(\frac{15r_{f,x}^3}{r_f^7}+\frac{-9r_{f,x}}{r_f^5}\right)B_{xx}+\left(\frac{15r_{f,x}^2r_{f,y}}{r_f^7}+\frac{-3r_{f,y}}{r_f^5}\right)B_{xy}\\ &+\left(\frac{15r_{f,x}^2r_{f,z}}{r_f^7}+\frac{-3r_{f,z}}{r_f^5}\right)B_{xz}+\left(\frac{15r_{f,x}^2r_{f,y}}{r_f^7}+\frac{-3r_{f,y}}{r_f^5}\right)B_{yx}\\ &+\left(\frac{15r_{f,x}r_{f,y}^2}{r_f^7}+\frac{-3r_{f,x}}{r_f^5}\right)B_{yy}+\left(\frac{15r_{f,x}r_{f,y}r_{f,z}}{r_f^7}\right)B_{yz}\\ &+\left(\frac{15r_{f,x}^2r_{f,z}}{r_f^7}+\frac{-3r_{f,z}}{r_f^5}\right)B_{zx}+\left(\frac{15r_{f,x}r_{f,y}r_{f,z}}{r_f^7}\right)B_{zy}\\ &+\left(\frac{15r_{f,x}r_{f,z}^2}{r_f^7}+\frac{-3r_{f,x}}{r_f^5}\right)B_{zz}\end{aligned} \tag{2.42}$$

$$\begin{aligned}\left(\frac{\partial V_f^C}{\partial r_x}\right)_0 = &\left(\frac{-9}{2r_f^5}+\frac{90r_{f,x}^2}{2r_f^7}+\frac{-105r_{f,x}^4}{r_f^9}\right)C_{xxx}+\left(\frac{45r_{f,x}r_{f,y}}{2r_f^7}+\frac{-105r_{f,x}^3r_{f,y}}{r_f^9}\right)C_{xxy}\\ &+\left(\frac{45r_{f,x}r_{f,z}}{2r_f^7}+\frac{-105r_{f,x}^3r_{f,z}}{r_f^9}\right)C_{xxz}+\left(\frac{45r_{f,x}r_{f,y}}{2r_f^7}+\frac{-105r_{f,x}^3r_{f,y}}{r_f^9}\right)C_{xyx}\\ &+\left(\frac{-9}{2r_f^5}+\frac{15r_{f,x}^2+15r_{f,y}^2}{2r_f^7}+\frac{-105r_{f,x}^2r_{f,y}^2}{r_f^9}\right)C_{xyy}\\ &+\left(\frac{15r_{f,y}r_{f,z}}{2r_f^7}+\frac{-105r_{f,x}^2r_{f,y}r_{f,z}}{r_f^9}\right)C_{xyz}+\left(\frac{45r_{f,x}r_{f,z}}{2r_f^7}+\frac{-105r_{f,x}^3r_{f,z}}{r_f^9}\right)C_{xzx}\\ &+\left(\frac{15r_{f,y}r_{f,z}}{2r_f^7}+\frac{-105r_{f,x}^2r_{f,y}r_{f,z}}{r_f^9}\right)C_{xzy}+\left(\frac{-9}{2r_f^5}+\frac{15r_{f,x}^2+15r_{f,z}^2}{2r_f^7}+\frac{-105r_{f,x}^2r_{f,z}^2}{r_f^9}\right)C_{xzz}\\ &+\left(\frac{30r_{f,x}r_{f,y}}{2r_f^7}+\frac{-105r_{f,x}^3r_{f,y}}{r_f^9}\right)C_{yxx}+\left(\frac{-9}{2r_f^5}+\frac{15r_{f,x}^2+15r_{f,y}^2}{2r_f^7}+\frac{-105r_{f,x}^2r_{f,y}^2}{r_f^9}\right)C_{yxy}\\ &+\left(\frac{15r_{f,y}r_{f,z}}{2r_f^7}+\frac{-105r_{f,x}^2r_{f,y}r_{f,z}}{r_f^9}\right)C_{yxz}+\left(\frac{-9}{2r_f^5}+\frac{15r_{f,x}^2+15r_{f,y}^2}{2r_f^7}+\frac{-105r_{f,x}^2r_{f,y}^2}{r_f^9}\right)C_{yyx}\\ &+\left(45r_{f,x}r_{f,y}+\frac{-105r_{f,x}r_{f,y}^3}{r_f^9}\right)C_{yyy}+\left(\frac{15r_{f,x}r_{f,z}}{2r_f^7}+\frac{-105r_{f,x}r_{f,y}^2r_{f,z}}{r_f^9}\right)C_{yyz}\\ &+\left(\frac{15r_{f,y}r_{f,z}}{2r_f^7}+\frac{-105r_{f,x}^2r_{f,y}r_{f,z}}{r_f^9}\right)C_{yzx}+\left(\frac{15r_{f,x}r_{f,z}}{2r_f^7}+\frac{-105r_{f,x}r_{f,y}^2r_{f,z}}{r_f^9}\right)C_{yzy}\\ &+\left(\frac{15r_{f,x}r_{f,y}}{2r_f^7}+\frac{-105r_{f,x}r_{f,y}r_{f,z}^2}{r_f^9}\right)C_{yzz}+\left(\frac{45r_{f,x}r_{f,z}}{2r_f^7}+\frac{-105r_{f,x}^3r_{f,z}}{r_f^9}\right)C_{zxx}\\ &+\left(\frac{15r_{f,y}r_{f,z}}{2r_f^7}+\frac{-105r_{f,x}^2r_{f,y}r_{f,z}}{r_f^9}\right)C_{zxy}+\left(\frac{-9}{2r_f^5}+\frac{15r_{f,x}^2+15r_{f,z}^2}{2r_f^7}+\frac{-105r_{f,x}^2r_{f,z}^2}{r_f^9}\right)C_{zxz}\\ &+\left(\frac{15r_{f,y}r_{f,z}}{2r_f^7}+\frac{-105r_{f,x}^2r_{f,y}r_{f,z}}{r_f^9}\right)C_{zyx}+\left(\frac{15r_{f,x}r_{f,z}}{2r_f^7}+\frac{-105r_{f,x}r_{f,y}^2r_{f,z}}{r_f^9}\right)C_{zyy}\\ &+\left(\frac{15r_{f,x}r_{f,y}}{2r_f^7}+\frac{-105r_{f,x}r_{f,y}r_{f,z}^2}{r_f^9}\right)C_{zyz}+\left(\frac{-9}{2r_f^5}+\frac{15r_{f,z}^2}{2r_f^7}+\frac{-105r_{f,x}^2r_{f,z}^2}{r_f^9}\right)C_{zzx}\\ &+\left(\frac{15r_{f,x}r_{f,y}}{2r_f^7}+\frac{-105r_{f,x}r_{f,y}r_{f,z}^2}{r_f^9}\right)C_{zzy}+\left(\frac{45r_{f,x}r_{f,z}}{2r_f^7}+\frac{-105r_{f,x}r_{f,z}^3}{r_f^9}\right)C_{zzz}\end{aligned} \tag{2.43}$$

The elements in the first row of $\mathbf{T}_f^{(l-1)}$ are the 39 coefficients on the multipole components in the expressions for $\left(\frac{\partial V_f^A}{\partial r_x}\right)_0$, $\left(\frac{\partial V_f^B}{\partial r_x}\right)_0$, and $\left(\frac{\partial V_f^C}{\partial r_x}\right)_0$. The other rows of $\mathbf{T}_f^{(l-1)}$ are defined similarly. The equations for $\left(\frac{\partial V_f^B}{\partial r_x}\right)_0$, and $\left(\frac{\partial V_f^C}{\partial r_x}\right)_0$ are given in the appendix.

Downward pass. The matrix $\mathbf{D}^{(l-1)}$ representing the downward pass is defined in a similar way to the one-dimensional case. Equation (2.34) gives $E_{f,\gamma}^T(\vec{r}+\vec{r}_0)$ as a polynomial in r_x, r_y, and r_z. For a given γ, the coefficients on the powers of r_x, r_y, and r_z can be stored in a matrix-vector product. The terms involving $\vec{r}_0$ are stored in the matrix block $\mathbf{D}_{<i,j,k>,m,\gamma}^{(l-1)}$, where l is the level of the child cells,

and $m = 1, \ldots, 8$ refers to a particular child cell of parent cell $< i, j, k >$.

$$\begin{bmatrix} & & \\ 1 & 0 & 0 \\ r_{0,x} & 1 & 0 \\ r_{0,y} & 1 & 0 \\ r_{0,z} & 1 & 0 \\ r_{0,x}^2 & 2r_{0,x} & r_{0,x}^2 \\ r_{0,x}r_{0,y} & 2r_{0,y} & r_{0,x}r_{0,y} \\ r_{0,x}r_{0,z} & 2r_{0,z} & r_{0,x}r_{0,z} \\ r_{0,y}r_{0,x} & 2r_{0,x} & r_{0,y}r_{0,x} \\ r_{0,y}^2 & 2r_{0,y} & r_{0,y}^2 \\ r_{0,y}r_{0,z} & 2r_{0,z} & r_{0,y}r_{0,z} \\ r_{0,z}r_{0,x} & 2r_{0,x} & r_{0,z}r_{0,x} \\ r_{0,z}r_{0,y} & 2r_{0,y} & r_{0,z}r_{0,y} \\ r_{0,z}^2 & 2r_{0,z} & r_{0,z}^2 \end{bmatrix}^\top \tag{2.44}$$

This gives a 3×1 matrix of the coefficients for $E_{f,\gamma}^T(\vec{r} + \vec{r}_0)$ when it is multiplied

by the vector $\vec{t}_{f,\gamma}$ of Taylor series coefficients for cell f:

$$\vec{t}_{f,\gamma} = \begin{bmatrix} \left(\frac{-\partial V_f}{\partial r_\gamma}\right)_0 \\ \left(\frac{-\partial^2 V_f}{\partial r_\gamma \partial r_x}\right)_0 \\ \left(\frac{-\partial^2 V_f}{\partial r_\gamma \partial r_y}\right)_0 \\ \left(\frac{-\partial^2 V_f}{\partial r_\gamma \partial r_z}\right)_0 \\ \left(\frac{-\partial^3 V_f}{\partial r_\gamma \partial r_x \partial r_x}\right)_0 \\ \left(\frac{-\partial^3 V_f}{\partial r_\gamma \partial r_x \partial r_y}\right)_0 \\ \left(\frac{-\partial^3 V_f}{\partial r_\gamma \partial r_x \partial r_z}\right)_0 \\ \left(\frac{-\partial^3 V_f}{\partial r_\gamma \partial r_y \partial r_x}\right)_0 \\ \left(\frac{-\partial^3 V_f}{\partial r_\gamma \partial r_y \partial r_y}\right)_0 \\ \left(\frac{-\partial^3 V_f}{\partial r_\gamma \partial r_y \partial r_z}\right)_0 \\ \left(\frac{-\partial^3 V_f}{\partial r_\gamma \partial r_z \partial r_x}\right)_0 \\ \left(\frac{-\partial^3 V_f}{\partial r_\gamma \partial r_z \partial r_y}\right)_0 \\ \left(\frac{-\partial^3 V_f}{\partial r_\gamma \partial r_z \partial r_z}\right)_0 \end{bmatrix} \tag{2.45}$$

The vector $\vec{t}$ appears implicitly in the CMMm factorization as the result of $\mathbf{T}^{(l-1)}$ operating on the result of the upward pass. The vector $\vec{t}_{f,\gamma}$ is the block of $\vec{t}$ corresponding to $E^T_{f,\gamma}(\vec{r}+\vec{r}_0)$. Like the matrix $\mathbf{U}^{(l-1)}$, the matrix $\mathbf{D}^{(l-1)}$ is sparse. The locations of the nonzero elements of $\mathbf{D}^{(l-1)}$ can be indexed similarly to the upward pass matrix. Since the columns of $\mathbf{U}^{(l-1)}$ correspond to the parent cells at level $l-1$, let $pp^{(l-1)}_{<i,j,k>} = 13(k-1) + 13(n_c^{(l-1)})^{1/3}(j-1) + 13(n_c^{(l-1)})^{2/3}(i-1)$.

This is a column index referring to the section of $\mathbf{D}^{(l-1)}$ that corresponds to cell $< i, j, k >$. Similarly, the columns of $\mathbf{U}^{(l-1)}$ correspond to the child cells at level l. The following row indices correspond to the 8 level l child cells of the level $l-1$ parent cell $< i, j, k >$:

$$\begin{aligned}
cc^{(l,1)}_{<i,j,k>} &= 9(2k-2) + 9(n_c^{(l-1)})^{1/3}(2j-2) + 9(n_c^{(l-1)})^{2/3}(2i-2) \\
cc^{(l,2)}_{<i,j,k>} &= 9(2k-2) + 9(n_c^{(l-1)})^{1/3}(2j-2) + 9(n_c^{(l-1)})^{2/3}(2i-1) \\
cc^{(l,3)}_{<i,j,k>} &= 9(2k-2) + 9(n_c^{(l-1)})^{1/3}(2j-1) + 9(n_c^{(l-1)})^{2/3}(2i-2) \\
cc^{(l,4)}_{<i,j,k>} &= 9(2k-2) + 9(n_c^{(l-1)})^{1/3}(2j-1) + 9(n_c^{(l-1)})^{2/3}(2i-1) \\
cc^{(l,5)}_{<i,j,k>} &= 9(2k-1) + 9(n_c^{(l-1)})^{1/3}(2j-2) + 9(n_c^{(l-1)})^{2/3}(2i-2) \\
cc^{(l,6)}_{<i,j,k>} &= 9(2k-1) + 9(n_c^{(l-1)})^{1/3}(2j-2) + 9(n_c^{(l-1)})^{2/3}(2i-1) \\
cc^{(l,7)}_{<i,j,k>} &= 9(2k-1) + 9(n_c^{(l-1)})^{1/3}(2j-1) + 9(n_c^{(l-1)})^{2/3}(2i-2) \\
cc^{(l,8)}_{<i,j,k>} &= 9(2k-1) + 9(n_c^{(l-1)})^{1/3}(2j-1) + 9(n_c^{(l-1)})^{2/3}(2i-1)
\end{aligned} \tag{2.46}$$

The symbols pp and cc are used for these indices to distinguish them from the indices in the upward pass matrix $\mathbf{U}^{(l-1)}$.

For the mth child cell of parent cell $< i, j, k >$, the upper left corners of the blocks $\mathbf{D}^{(l-1)}_{<i,j,k>,m,\gamma}$ are located at the following row-column indices in the matrix $\mathbf{D}^{(l-1)}$:

$$\begin{aligned}
\mathbf{D}^{(l-1)}_{<i,j,k>,m,x}&: \ (cc^{(l,m)}_{<i,j,k>} + 1, pp^{(l-1)}_{<i,j,k>} + 1) \\
\mathbf{D}^{(l-1)}_{<i,j,k>,m,y}&: \ (cc^{(l,m)}_{<i,j,k>} + 4, pp^{(l-1)}_{<i,j,k>} + 1) \\
\mathbf{D}^{(l-1)}_{<i,j,k>,m,z}&: \ (cc^{(l,m)}_{<i,j,k>} + 7, pp^{(l-1)}_{<i,j,k>} + 1)
\end{aligned} \tag{2.47}$$

In summary, we have presented in detail a new matrix version of the cell multipole method for a system of charges as well as for a system of point dipoles. For a polarizable point dipole system, the CMMm scheme takes this form:

$$\vec{E} = \vec{E}^{near} + \begin{bmatrix} \mathbf{R}^\top \end{bmatrix} \left(\begin{bmatrix} \mathbf{D} \end{bmatrix} \cdot \underbrace{\begin{bmatrix} \quad \end{bmatrix}} \cdot \begin{bmatrix} \mathbf{U} \end{bmatrix} + \begin{bmatrix} \mathbf{T} \end{bmatrix} \right) \begin{bmatrix} \mathbf{R} \end{bmatrix} \begin{bmatrix} \vec{\mu} \end{bmatrix}$$

$$([\mathbf{D}] \cdot \underbrace{[\ ]} \cdot [\mathbf{U}] + [\mathbf{T}])$$

$$([\mathbf{D}] \cdot \underbrace{[\ ]} \cdot [\mathbf{U}] + [\mathbf{T}])$$

$$\ddots$$

where this is plugged into the equation $\vec{\mu}_i = \alpha_i \vec{E}_i$.

Chapter 3

Standard Iterative Methods for Solving Linear Equations

Systems of linear equations in the form $\mathbf{A}\vec{x} = \vec{b}$ can be solved directly or via iterative methods. Obtaining the exact solution can be a daunting task for large systems since the computational cost is proportional to n^3. Iterative methods generally have a lower computational cost, making it possible to solve very large systems that would have been intractable to compute directly. Starting with an initial approximate solution, these techniques improve the accuracy of this estimate by generating a sequence of better approximations until convergence is achieved. In this section, we will give an overview of various iterative methods, primarily focusing on the fast solvers using the multigrid formulation. An extensive

survey of iterative solvers is given by Saad and van der Vorst[231] and several useful texts.[232, 233, 234, 235, 236, 237, 238, 239, 240]

3.1 Jacobi and Gauss-Seidel Iteration Methods

The simplest iterative schemes are the Jacobi and Gauss-Seidel methods. The Jacobi iterative solver begins with an initial guess or approximation of the solution vector $\vec{x}^{(0)}$ in $\mathbf{A}\vec{x} = \vec{b}$. This is then substituted back into the system of equations to generate new approximate solutions. This process is continued until convergence is reached. An algorithm for the Jacobi iteration is as follows:[234]

Algorithm 3 : Jacobi Iteration.

Given: initial guess $\vec{x}^{(0)}$ for the solution
* a_{ij}=element in row i and column j of matrix **A** *\
* x_i^k=i^{th} element of the approximate solution in k^{th} iteration *\
do k=1,2,...until convergence is reached
  for i=1 to n

$$x_i^{k+1} = \frac{b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}}{a_{ii}}$$

  end for
end do

To speed up the convergence, the rows of the starting matrix can be rearranged so that the magnitudes of the diagonal elements are as large as possible relative to the other matrix elements in the row. This is because these schemes converge more rapidly when the matrix **A** is diagonally dominant:[232] $|a_{ii}| > \sum_{j=1;j\neq i}^{n} |a_{ij}|$ for $i = 1, 2, \ldots n$.

For faster convergence, this algorithm can be revised to use the most recent estimates of the exact solution as soon as they are available, to obtain the Gauss-Seidel iteration method:[234]

Algorithm 4 : Gauss-Seidel Iteration.

Given: initial guess $x^{(0)}$ for the solution
* a_{ij}=element in row i and column j of matrix **A** *\
* x_i^k=i^{th} element of the approximate solution in k^{th} iteration *\
do k=1,2,...until convergence is reached
  for i=1 to n

$$x_i^{k+1} = \frac{b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)}}{a_{ii}}$$

  end for
end do

An advantage of using the most updated values as soon as they are computed is to reduce the storage for the approximate vector to n locations which becomes more important as the size of the system grows.[232] When the most recently updated values are used, the order in which the values are updated becomes more important. An effective ordering strategy, known as the Red-Black Gauss-Seidel (RBGS) method, involves first updating all of the even components of the solution, then updating the odd components of the solution. This order of updating lends itself to being parallelized.[232] A common alternative is to modify the Gauss-Seidel equation to: $x_i^{k+1} = \omega \frac{b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)}}{a_{ii}} + (1-\omega) x_i^{(k)}$, with a relaxation parameter ω. A proper choice of ω can speed up the convergence rate. This modification is generally known as Successive Over-Relaxation (SOR).

3.2 Conjugate Gradient Method

The conjugate gradient method is one of the most widely used methods for solving large, symmetric positive definite linear systems.[234] This iterative scheme generates a succession of search directions $\vec{p}^{(i)}$ in order to rapidly minimize the error.[232] When the exact solution is not known, the residual is used as a measure of how close the approximate solution is to the exact solution, which is given by:

$$\vec{d} = \vec{b} - \mathbf{A}\vec{x}_{approx}. \tag{3.1}$$

The starting value for the residual can be found using an initial approximation of the solution vector $\vec{x}^{(0)}$. CG computes improved approximations of the solutions using a multiple α_i of the search direction: $\vec{x}^{(i)} = \vec{x}^{(i-1)} + \alpha_i \vec{p}^{(i)}$, and reduces the residual recursively with $\vec{d}^{(i)} = \vec{d}^{(i-1)} - \alpha_i \vec{\omega}^{(i)}$ and $\vec{\omega}^{(i)} = A\vec{p}^{(i)}$, until the residual is smaller than a given tolerance. A conjugate gradient method[234] is shown in Algorithm 5.

The standard CG algorithm guarantees convergence for symmetric positive definite systems. When a matrix is nonsymmetric, the simplest way to guarantee convergence using CG is to find a transformation that can make the system symmetric positive definite. The simplest such transformation is the normal equations $\mathbf{A}^\top \mathbf{A}\vec{x} = \mathbf{A}^\top \vec{b}$. Then CG is used on the transformed system to enhance its numerical stability. This scheme is generally referred to as the Conjugate Gradient Normal Equations Residual method (CGNR). The drawback of doing this is that

Algorithm 5 : Conjugate Gradient Method.

Given: initial guess $x^{(0)}$ for the solution
* α_i = step size in the search direction *\
* $\vec{p}^{(i)}$ = search direction *\
* $\vec{d}$ = residual *\
$k = 0$
$\vec{d} = \vec{b} - A\vec{x}^{(0)}$
$\rho_0 = |\vec{d}|^2$
do until convergence is reached
  $k = k + 1$
  if $k = 1$
    $\vec{p} = \vec{d}$
  else
    $\beta_k = \rho_{k-1}/\rho_{k-2}$
    $\vec{p} = \vec{d} + \beta_k \vec{p}$
  end if
  $\vec{\omega} = A\vec{p}$
  $\alpha_k = \frac{\rho_{k-1}}{\vec{p}^\top \vec{\omega}}$
  $\vec{x} = \vec{x} + \alpha_k \vec{p}$
  $\vec{d} = \vec{d} - \alpha_k \vec{\omega}$
  $\rho_k = |\vec{d}|^2$
end do

it will substantially slow the convergence rate. However, it is simple to code and provides a regular pattern of convergence. More complex methods are described by Barrett[232] and Golub.[234]

3.3 Multigrid Methods

Another approach for solving large systems of equations comprises multigrid methods which use a succession of grids to represent the approximate solution at different levels. The idea is to increase computational efficiency by reducing the number of variables and solving a smaller problem for a better initial guess of the solution. The error of this approximate solution is reduced and transferred to the finest grid for a more refined solution. The first multigrid method can be traced back to the 1960s where Fedorenko applied this formalism to solve Poisson's equation.[241, 242] It was proven to be effective for solving partial differential equations by Brandt in 1977.[243] Since then, it has been applied to numerous problems, including eigenfunction problems,[244, 245, 246, 247] wave propagation equations,[248, 249] and fluid dynamics[250, 251, 252] among others. Parallel implementations have also been made available for numerous applications.[253, 254]

3.3.1 Standard Multigrid Method

Consider a system of linear equations in matrix-vector form:

$$\mathbf{A}\vec{x} = \vec{b}$$

where the grid operator is $\mathbf{A} = (A_{ij})_{n\times n}$, the solution vector is $\vec{x} = (x_1, x_2, ...x_n)^\top$, and the right-hand side is $\vec{b} = (b_1, b_2, ...b_n)^\top$. The original problem $\mathbf{A}\vec{x} = \vec{b}$ is referred to as the fine-grid problem. From this fine grid, the multigrid method generates a hierarchy of smaller structured grids to represent a series of smaller approximate problems. The basic idea is to increase computational efficiency by reducing the number of variables and solving a smaller problem for a better initial guess of the solution.

This method is effective in computing a solution for a problem at the finest grid by combining the strengths of relaxation methods and coarse grids to eliminate the error. Through Fourier analysis, it can be seen that error consists of smooth (low-frequency) and non-smooth (high-frequency) components. Relaxation methods are iterative methods that effectively reduce the high frequency modes, but have little effect on low frequency modes. When the problem is brought to a coarse grid, it is reduced to a simpler problem where the low-frequency components can be effectively reduced. Repeating this process eventually results in a problem with one unknown that can be solved directly. Then this solution can be brought back to the finer grids, using additional relaxation to reduce any new error introduced

during interpolation. Approximating the problem onto coarse grids allows cheaper and much more effective relaxation on low frequency modes. Additional details on the theory and convergence properties of multigrid methods can be found in these texts.[255, 256, 257, 258, 259] The basic components of multigrid (relaxation, intergrid transfer operators, and coarse grid correction) are described below.

Relaxation Schemes

Relaxation methods effectively reduce high frequency error modes. The most common smoother is Gauss-Seidel[257] discussed in Section 3.1. Other robust smoothers have been created, but Gauss-Seidel is generally used because of its simplicity and low storage requirement. This relaxation method performs smoothing using the most current estimate of the solution. In order for Gauss-Seidel computation to be valid at a given grid level, it is assumed that the diagonal elements of the matrix $\mathbf{A}$ at that grid level must be nonzero. Updated solution grid points are computed by taking a weighted average of the corresponding right-hand side value and values at neighboring grid points. After each new x-value is computed, it is used in the next step to determine the next x-value. Red-black Gauss-Seidel (RBGS) decomposes the grid into two sets of points like a checkerboard to form a different ordering of updates, rather than the sequential updates of standard Gauss-Seidel. The grid points are updated in two sweeps, corresponding to sweeping all of the red points and then all the black points, which is described in Algorithm 6.

RBGS ordering generally converges faster than standard Gauss-Seidel and has the capability of being parallelized.[232]

Algorithm 6 : Red-Black Gauss-Seidel Smoother

Given: initial guess x_0 for the solution
* Red sweep *\
for all red points
$$x_i^{new} = \frac{b_i - \sum_{j=1}^{i-1} A_{ij} x_j^{new} - \sum_{j=i+1}^{n} A_{ij} x_j^{old}}{A_{ii}}$$
end for
* Black sweep *\
for all black points
$$x_i^{new} = \frac{b_i - \sum_{j=1}^{i-1} A_{ij} x_j^{new} - \sum_{j=i+1}^{n} A_{ij} x_j^{old}}{A_{ii}}$$
end for

Intergrid Transfer and Coarse-grid Operators

In order to move from one grid to another, transfer operators are used. The prolongation operator (**P**), also known as the interpolation operator, maps a vector from the coarse grid to the fine grid, whereas the restriction operator (**R**) maps the vector from the fine grid to the coarse grid. The prolongation operator is defined by linear interpolation demonstrated in Algorithm 7. In prolongation, the coarse grid values are directly copied to the fine grid, and the points between them are an average of the neighboring coarse grid points.

Algorithm 7 : Prolongation.

* Copies *\
for all coarse grid points
    $x_j^{fine} = x_j^{coarse}$
end for
* All Others *\
for all fine grid points that are not directly copied from coarse grid
    $x_j^{fine} = \frac{1}{2}x_{j-1}^{coarse} + \frac{1}{2}x_{j+1}^{coarse}$
end for

The restriction operator is defined by a full-weighted average shown in Algorithm 8. In restriction, the boundary points are copied directly to the coarse grid while the interior points of the coarse grid are weighted averages of fine grid points. The prolongation operator is related to the restriction operator by: $\mathbf{R} = \frac{1}{2}\mathbf{P}^{\top}$.

Algorithm 8 : Restriction.

for all coarse grid points
    $x_j^{coarse} = \frac{1}{4}x_{j-1}^{fine} + \frac{1}{2}x_j^{fine} + \frac{1}{4}x_{j+1}^{fine}$
end for

The coarse grid operator is found using the Galerkin coarse grid approximation (GCA) which is given by $\mathbf{A}^{coarse} = \mathbf{R}\mathbf{A}^{fine}\mathbf{P}$. Specifically, the coarse grid operator is computed using this equation:

$$\mathbf{A}^{coarse}(i,n) = \sum_{m \in S_R} \sum_{k \in S_A} \mathbf{R}(i,m)\mathbf{A}^{fine}(2i+m,k)\mathbf{P}^{*}(i+n,m+k-2n) \tag{3.2}$$

where $(\mathbf{A}\vec{x})_i = \sum_{j \in \mathbb{Z}} \mathbf{A}(i,j)x_{i+j}$, and S_R and S_A are the structures of $\mathbf{R}$ and $\mathbf{A}$, respectively. Extensive formulation and implementation of this approximation is discussed in Wesseling's text.[259]

Coarse Grid Correction Scheme

An important component of the multigrid method is the coarse grid correction. Relaxation methods alone are not the most effective way to find a solution to a matrix equation. However, they can efficiently smooth the error, leaving the error to be primarily composed of low-frequency error modes. These error modes can be more easily approximated on a coarser grid. To take advantage of this, the multigrid transfers the residual $\vec{d}$ of a solution to coarser grid levels, where the low-frequency error is quickly reduced due to a smaller number of unknowns. In the process of coarse grid correction, the residual is restricted to a coarser grid and the problem $\mathbf{A}\vec{v} = \vec{d}$ is solved for $\vec{v}$, a correction used to improve the approximate solution for $\vec{x}_{approx}$. This $\vec{v}$ is interpolated to a finer grid to correct the fine grid approximation. In our implementation, Cramer's rule was used to find the coarse grid solution. Cramer's rule finds the solution $\vec{x}$ of $\mathbf{A}\vec{x} = \vec{b}$ using the equation:

$$x_i = \frac{det\ \mathbf{A_i}(\vec{b})}{det\ \mathbf{A}} \tag{3.3}$$

where i = 1,2...k, and $\mathbf{A_i}(\vec{b})$ is the matrix $\mathbf{A}$ with its i^{th} column replaced by $\vec{b}$.[260] For larger coarse grid systems, either Gaussian elimination or conjugate gradient is generally used as the coarse grid solver.[257, 234]

Cycling Algorithms

The multigrid method is an iterative scheme that utilizes all of these components: relaxation, coarse grid generation, intergrid transfer operators, coarse grid operators, and correction scheme. Given an initial guess on the finest grid, the multigrid method generates a hierarchy of smaller structured grids to represent a series of smaller approximations. The simplest multigrid cycle is the v-cycle[261] described in Algorithm 9 and Figure 3.1. Using the initial guess, the residual is computed and transferred to a coarser grid with the restriction operator. On each coarser grid, the residual equation continues to be relaxed before being restricted to the next coarser grid recursively until the coarsest grid is reached. At the coarsest level, the direct solution of the residual is calculated, giving the error that is then interpolated. This correction is used to improve the approximate solution and relaxed. The interpolation operator successively brings this problem to finer grids where the correction is added and refined with relaxation methods at each grid level. Efficiency is improved by using nested v-cycles to achieve convergence, providing the best approximation possible with these inexpensive v-cycle computations. This nested iteration is called the full multigrid v-cycle[265] which is demonstrated in Algorithm 10 and Figure 3.2.

Algorithm 9 : V-cycle

do until reach coarse grid
  Presmoothing with relaxation method ($\vec{x}^{new} = \mathbf{S}\vec{x}^{old}$)
  Compute residual $\vec{d} = \vec{b} - \mathbf{A}\vec{x}$
  Restrict residual to coarse grid ($\vec{d}^{coarse} = \mathbf{R}\vec{d}^{fine}$)
end do
Solve coarse grid problem ($\mathbf{A}\vec{e} = \vec{d}$)
do until reach finest grid
  Interpolate coarse grid error to finer grid ($\vec{v}^{fine} = \mathbf{P}\vec{v}^{coarse}$)
  Correct approximate solution ($\vec{x}^{new} = \vec{x}^{old} + \vec{v}$)
  Postsmoothing with relaxation method ($\vec{x}^{new} = \mathbf{S}\vec{x}^{old}$)
end do

Figure 3.1: V-cycle.

Algorithm 10 : FMG V-cycle

```
if at coarsest grid
   Compute direct solution
else
   Restrict to coarsest grid
   Initial solution on coarsest grid
   do for all grids
     Interpolate from coarse grid to next finer grid
     Perform V-cycle (pre-smoothing, residual, restriction of residual,
           coarse grid solution, correct solution, post-smoothing)
   end do
end if
```

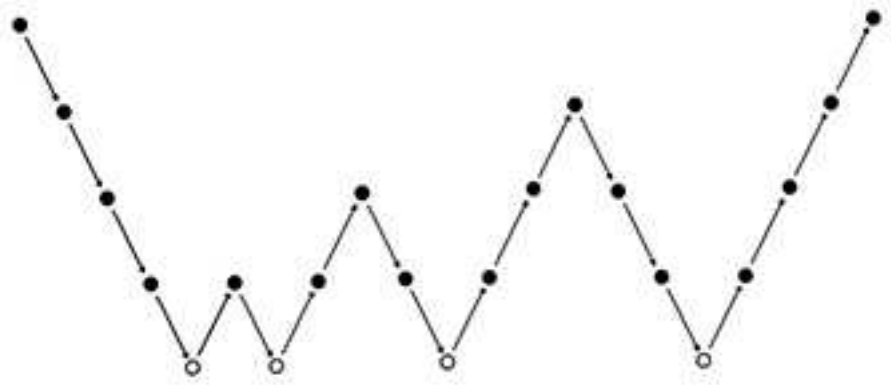


Figure 3.2: FMG V-cycle.

3.3.2 Algebraic Multigrid Method

Standard multigrid methods are geometry-dependent in nature, which are designed only for structured problems with regular meshes. They are sometimes referred to as geometric multigrid methods (GMG). Algebraic multigrid methods are

matrix-based methods that extend the scope of these techniques to solve unstructured linear systems.[262, 263, 264, 265] The algebraic multigrid method (AMG) was proposed by Ruge and Stüben in 1987.[263] Since then, AMG has been applied to various problems such as fluid dynamics,[264, 266, 267] electromagnetics,[268, 269] and eigenvalue problems.[270, 271] A number of parallel versions have also been derived.[272, 274, 275]

AMG is designed for a more general class of problems than GMG, which has the same fundamental components (sequence of grids, intergrid transfer operators, smoother, coarse-grid operators, coarse-grid solver). The advantage of AMG methods is that grid generation and solution computations are solely based on the variables from the given matrix equation $\mathbf{A}\vec{x} = \vec{b}$, rather than requiring any geometric information about the problem.[276] Therefore, the grids generated by AMG are defined by the matrix entries, rather than a geometric grid where the locations of grid points and their connections are known. However, the added flexibility of AMG comes with a price of some overhead for the setup phase. The set up phase is necessary to generate the coarse grids and operators, prior to applying the multigrid cycles in the solution phase. In the following section, we will focus on the coarsening process and determination of intergrid transfer operators. Smoothing and coarse-grid operators are previously covered in the standard geometric multigrid methods in Section 3.3.1, as Gauss-Seidel iteration is the commonly used relaxation method and the Galerkin method is used for the coarse-grid operator.

Coarsening

In the coarse-grid generation,[263] the grid points are partitioned into C-points (coarse-grid variables) and F-points (variables not on the coarse grid) based on the concept of strong matrix connections between all grids. The strength of these connections is defined as:

> Given a threshold of $0 < \theta < 1$, grid point i *strongly depends* on j or j *strongly influences* i if:

$$-a_{ij} \geq \theta \max_{k \neq i}(-a_{ik}).$$

where the matrix coefficient a_{ij} is compared to the largest off-diagonal coefficient in the i^{th} equation. This definition plays an important role in determining the coarse grid-points. Let us then define the following sets:[265]

$\mathbf{S_i}$: set of points in which point i strongly depends on

$\mathbf{S_i^T}$: set of points j that are influenced by i

$\mathbf{N_i}$: (neighborhood of i) set of all points $j \neq i$ and $a_{ij} \neq 0$

$\mathbf{C_i}$: coarse interpolatory set that includes the neighboring C-points that are strongly connected to i

$\mathbf{D_i^s}$: neighboring F-points that strongly influence i

$\mathbf{D_i^w}$: weakly connected neighbors that include points that do not strongly influence i

Two heuristic criteria are also imposed in this coarsening scheme:[263, 265]

C1: Given an F-point i, for all j that i strongly depends on, each of these j should either be included in C_i or strongly depend on some point in C_i.

C2: Set of C-points needs to form an optimal subset of all the given grid points with no two C-points strongly connected.

In order to insure proper description of the matrix problem on the coarse grid, the first criterion is enforced. This criterion implies that strong connections are important in accurately representing the relationship between neighboring grid points. Therefore, if one grid point does not strongly influence another, it is not important in the construction of the coarse grid or interpolation operator. The second criterion is implemented to avoid creating relatively large coarse grids that may slow down the convergence rate. Using these guidelines, the coarsening procedure is formulated in two passes:

First Pass: We first measure each point's number of "influences" which is the number of points that are strongly influenced by point i. This measure is denoted by λ_i. The point with the most "influences" (largest λ) is assigned to be a C-point, and all of the points that strongly depend on this C-point then become F-points. The λ_j of all the other j points that strongly influence these

F-points are incremented because they are more likely to be important as C-points. This process is repeated until all the points are partitioned into F-points and C-points.

Second Pass: To insure that all of the criteria are enforced, an additional pass searches through all of the pairs of F-points that do not meet this requirement and changes one of them into a C-point.

The coarsening algorithm is further described in Algorithm 11.

Algorithm 11 : AMG Coarse Grid Generation[257]

```
\* First Pass *\
Find S_i and S_i^T for all points i on grid
Find λ_i for all points i on grid
for i=1,2...until all points are either designated as a C-point or F-point
   Pick i with maximal λ_i and this becomes a C-point
   For all j in S_i^T that strongly depend on this C-point becomes an F-point
   Increase λ_j for all other j that strongly influence these F-points
end for
\* Second Pass *\
for all points i on grid in F
   for all j in D_i^S for current i
      if
         j does not strongly depend on a point in C_i and
         all other j in D_i^S depend strongly on points in C_i
      then
         j becomes a C-point
      end if
   end for
end for
```

Defining Intergrid Operators and Coarse-grid Operators

In AMG, the interpolation operator is characterized by the set of C-points and F-points found in the coarsening scheme. The components of the interpolation operator that interpolates the error at the F-point is then defined by this interpolation formula:

$$P_{ij} = \begin{cases} w_{ij} & \text{if } j \in C_i \\ 0 & \text{otherwise} \end{cases} \tag{3.4}$$

where w_{ij} is the interpolation weight and C_i is the subset of C-points whose values are used to interpolate a value at i. The interpolation weight is:

$$w_{ij} = -\frac{a_{ij} + \sum\limits_{m \epsilon D_i^s} \left(\frac{a_{im} a_{mj}}{\sum\limits_{k \epsilon C_i} a_{mk}} \right)}{a_{ii} + \sum\limits_{n \epsilon D_i^w} a_{in}} \tag{3.5}$$

From the interpolation operator, the restriction operator is found using the property $R = P^\top$ and the coarse-grid operator is constructed from the Galerkin method that imposes the condition: $A^{coarse} = RA^{fine}P$.[263]

Algorithm 12 : AMG Interpolation Operator Generation[265]

```
for all i^fine on fine grid
    for all j^fine in C
      if i^fine = j^fine
        then P_{i,fine} j_coarse=1
      else if i^fine is in F and j^fine is in C_{i,fine}
        then P_{i,fine} j_coarse=w_{i,fine} * j_coarse
      else
        P_{i,fine} j_coarse=0
      end if
    end for
end for
```

AMG algorithm

Using the components defined in the previous sections, the algebraic multigrid algorithm given below can be formulated to include a setup phase for construction of coarse grids, intergrid operators and coarse-grid operators, and a solution phase that includes multigrid cycle. For further detailed discussion on algebraic multigrid methods, we refer to texts by Trottenberg[257] and Briggs.[265]

Algorithm 13 : Algebraic Multigrid Method.

* Setup phase: *\
for k=1, 2...until all coarse grids have been constructed
    Partition grid into C and F sets to find coarse grid-points (Coarsening)
    Construct interpolation operator
    Set $R = P^{\top}$ and $A^c = RA^fP$ to construct restriction, coarse-grid operators
end for
* Solution phase: *\
* Same process as standard MG, assumed as FMG V-cycle here *\
If at coarsest grid
  Compute direct solution
Else
  Restrict to coarsest grid
  Initial solution on coarsest grid
  Do for all grids
    Interpolate from coarse grid to next finer grid
    Do V-cycle (pre-smoothing, residual, restriction of residual,
        coarse grid solution, correct solution, post-smoothing)
  End Do
End If

3.4 Comparison of Iterative Solvers

All of these iterative methods have their strengths and weaknesses. Table 3.4 shows the storage requirement and complexity of various iterative solvers discussed in this chapter. The scaling of the multigrid methods are optimal compared to other iterative methods. Jacobi and Gauss-Seidel methods are easy to use and the storage required for these methods is minimal, but convergence is generally very slow compared to conjugate gradient or multigrid methods.[232] One way to reduce the computational cost is through parallel implementation. Jacobi iteration can be parallelized because each matrix-vector product is independent of ordering and can be done simultaneously.[234] Gauss-Seidel can also be run in parallel if a multi-color ordering of unknowns is used.[232] Variants of the conjugate gradient method have been parallelized as well.[280] As mentioned in Section 3.3, multigrid methods are well-suited for parallelization. If sufficient storage is available, multigrid methods may be the best candidates for further exploration because of their optimal scaling and potential for parallelism.

Table 3.1: Comparison of Iterative Solvers.

Iterative Solver	Storage Requirement	Complexity
Jacobi[232, 233]	$n^2 + 3n$	$\mathrm{O}(n^2)$
Gauss-Seidel,SOR[232, 233]	$n^2 + 2n$	$\mathrm{O}(n^2)$
Conjugate Gradient[232, 255]	$n^2 + 6n$	$\mathrm{O}(n^{3/2})$
Multigrid[265]	$\frac{2n^d}{1-2^{-d}}$, $d = 1, 2, 3$	$\mathrm{O}(n)$

Chapter 4

Multipole & Multigrid Methods for Computing Polarizable Interactions

We have described different methods for accounting for polarization in Chapter 1, methods for the treatment of long-range electrostatic interactions in Chapter 2, and fast solvers in Chapter 3. With this foundation, we can now focus our efforts towards eliminating the bottleneck in computing polarizable interactions. In this study, we will describe polarization using polarizable point dipoles due to the flexibility of this representation. This widely-used model has been shown to yield accurate properties in many applications.[125, 149] Because of their

optimal scaling, the cell multipole and multigrid techniques described in Chapter 2 and 3 are combined to form a new algorithm to solve the dipole iteration found in polarizable point dipole force fields.[125, 31, 172, 198] In this chapter, we will present these algorithms to accelerate the rate of convergence for this computation.

4.1 MG-CMM:

A Multigrid-Multipole Based Algorithm

4.1.1 Formulation

Given a collection of particles' positions, charges, and polarizabilities (derived by Thole[46]), the induced dipoles of this system can be computed using this system of equations:

$$\vec{\mu}^{ind} = \mathbf{M}\vec{q} + \mathbf{N}\vec{\mu}^{ind} \tag{4.1}$$

where $\vec{q}$ and $\vec{\mu}^{ind}$ contain the charges and induced dipoles of the atoms, $\mathbf{M}$ is the induction-from-charges operator and $\mathbf{N}$ is the induction-from-dipoles operator. The matrix sub-blocks for $\mathbf{M}$ and $\mathbf{N}$ are:

$$\mathbf{M}_{ij} = \begin{cases} \frac{\alpha_i \vec{r}_{ij}}{r_{ij}^3}, & \text{if } r_i \neq r_j \\ 0, & \text{if } r_i = r_j \end{cases} \tag{4.2}$$

and

$$\mathbf{N}_{ij} = \begin{cases} \alpha_i \left(\frac{3\vec{r}_{ij}\vec{r}_{ij}^{\top}}{r_{ij}^5} - \frac{\mathbf{I}}{r_{ij}^3} \right), & \text{if } r_i \neq r_j \\ 0, & \text{if } r_i = r_j \end{cases} \tag{4.3}$$

This equation is solved iteratively until convergence is reached. The MG-CMM algorithm[288, 289] was formulated to speed up the convergence rate for large systems. We applied the MG-CMM algorithm on one-dimensional model systems where all the particles are uniformly distributed. Although these systems are not directly related to any physical applications, it serves as a useful model to demonstrate the acceleration in convergence compared to other standard iterative solvers. This algorithm separates polarizable particle interactions into direct and indirect components, where the cell multipole method is used to approximate the indirect component. The overall interaction is then evaluated by a multigrid solver. For our multigrid method, we use RBGS smoothing, linear interpolation and full-weighting restriction. The atoms are assumed to be far enough apart to avoid the "polarization catastrophe",[41, 46, 75] as this does not occur at physically reasonable distances. Thus, no screening function is used. For further simplification, the term approximated by CMM is included on the right-hand side of the matrix equation to reduce the cost per iteration by eliminating the need to reevaluate the CMM field terms at each iteration. This is a reasonable approximation as it was observed by Kutteh and Nicholas that the dipole moments and the Taylor series coefficients varied slowly between iterations.[219] Since the contributions from the far field of the induced dipoles are much smaller than the near field contribution, they reduced the number of far field updates to decrease the computational cost

and also showed that less frequent updates did not affect the accuracy. Therefore, to utilize the CMM far field scheme, we modified equation (4.1) by dividing the induction matrix **N** into direct and indirect interactions:

$$\vec{\mu}^{ind} = \mathbf{M}\vec{q} + (\mathbf{N}^{direct} + \mathbf{N}^{indirect})\vec{\mu}^{ind} \tag{4.4}$$

Subsequently, this can be rearranged to form a matrix-vector equation:

$$\begin{array}{ccccc} \underbrace{(\mathbf{I} - \mathbf{N}^{direct})}_{\mathbf{A}} & \underbrace{\vec{\mu}^{ind}}_{\vec{x}} & = & \underbrace{\mathbf{M}\vec{q} + \mathbf{N}^{indirect}\vec{\mu}^{ind}}_{\vec{b}} \end{array} \tag{4.5}$$

In this matrix equation, **A** will be referred to as the grid operator, $\vec{x}$ as the solution, and $\vec{b}$ as the right-hand side. The right-hand side includes $\mathbf{M}\vec{q}$ and the indirect interactions computed by the cell multipole method multiplied by each atom's polarizability. CMM equations in terms of the potential for dipolar systems were formulated by Kutteh and Nicholas.[219] We have derived the CMM field terms for dipolar systems, as discussed in Section 2.2.1. The resulting one-dimensional equations that include polarizable dipoles are given below for the multipole moments at the deepest level:

$$\begin{aligned} \mu^{(l)}_{cell,x} &= \sum_i \mu_i \\ Q^{(l)}_{cell,xx} &= \sum_i \mu_i r_{ix} \\ O^{(l)}_{cell,xxx} &= \sum_i \mu_i r^2_{ix} \end{aligned} \tag{4.6}$$

and translation and summation of the child cell multipole moments yields the

higher level moments:

$$\begin{aligned} \mu_{cell,x}^{(l-1)} &= \sum_{k=1}^{children} \mu_{k,x}^{(l)} \\ Q_{cell,xx}^{(l-1)} &= \sum_{k=1}^{children} Q_{k,xx}^{(l)} + \mu_{k,x}^{(l)} R_k \\ O_{cell,xxx}^{(l-1)} &= \sum_{k=1}^{children} O_{k,xxx}^{(l)} + 2Q_{k,xx}^{(l)} R_k + \mu_{k,x}^{(l)} R_k^2 \end{aligned} \tag{4.7}$$

The electric field expansion for the far field used in the downward pass is given by:

$$\vec{E}_{i,far} = E^{(0)} + E^{(1)} r + E^{(2)} r^2 \tag{4.8}$$

where,

$$\begin{aligned} E_0 &= \frac{2\mu}{|r_A|^3} + \frac{-6Qr_A}{|r_A|^5} + \frac{12O}{|r_A|^5} \\ E_1 &= \frac{-6\mu r_A}{|r_A|^5} + \frac{24Q}{|r_A|^5} + \frac{-60Or_A}{|r_A|^7} \\ E_2 &= \frac{24\mu}{|r_A|^5} + \frac{-120Qr_A}{|r_A|^7} + \frac{360O}{|r_A|^7} \end{aligned}$$

and the following expression for the near dipolar field:

$$E_{near} = \sum_i \frac{2\mu_i}{r^3}. \tag{4.9}$$

The steps of the MG-CMM scheme are summarized in Algorithm 14.

4.1.2 Data Structures

The data structure used to compute the indirect interaction from CMM includes a doubly-linked list storing each atom's position, induced dipole, and near

Algorithm 14 : MG-CMM Method.

Given: Particle positions, charges, polarizabilities
Initial guess: $\mu_i \approx (Mq)_i$
MG setup
CMM field computation for indirect interactions:$(N_{indirect}\vec{\mu}_{ind})_i = \alpha_i \vec{E}_{CMM,i}$
System decomposition
Cell multipole moments:μ, Q, O
Far field Taylor series Coefficients:$E^T_{A_0}$, $E^T_{B_0}$, $E^T_{C_0}$
Grid operator coarsening of $\mathbf{A}$ at all levels using GCA ($\mathbf{A}^{coarse} = \mathbf{R}\mathbf{A}^{fine}\mathbf{P}$)
FMG solution cycles: $(\mathbf{I} - \mathbf{N}^{direct})\vec{\mu}^{ind} = \mathbf{M}\vec{q} + \mathbf{N}^{indirect}\vec{\mu}^{ind}$

and far field contributions of the CMM field. This interaction list is connected to a binary tree where pointers allow easy access to each tree node. Each node of the tree structure holds the cell's multipole moments and Taylor coefficients. All the matrices used in the multigrid method are stored in dynamically allocated multidimensional arrays. Further improvement in code efficiency would include sparse matrix storage schemes for faster computations, an indexing system instead of an explicit binary tree, and parallel implementation to reduce the run-time.

4.1.3 Performance

MG-CMM was applied to several test problems of varying sizes ranging from about 100 to 4000 particles. These results are compared to various iterative solvers (Jacobi, Gauss-Seidel, successive over-relaxation, conjugate gradient normal residual equation) and the matrix inversion method. The Jacobi-based methods, Jacobi, GS, and SOR are the traditional iterative solvers used for the

dipole iteration. Figure 4.1 shows the number of iterations needed for convergence of the Jacobi, GS, SOR, CGNR, and MG-CMM methods with various system sizes, evaluated with a convergence limit of 10^{-5}D. For the dipole iteration, the common choice of overrelaxation parameter, $\omega = 0.75$ was used for SOR.[6] The number of iterations slowly increased with increasing system size for Jacobi, GS, SOR, CGNR, whereas the number of iterations necessary for convergence for MG-CMM remained constant and independent of system size. MG-CMM has the lowest iteration count with only four FMG-cycles being sufficient to reach convergence in all test systems. The best results for the multigrid-multipole method were obtained with 2-pre and 2-post smoothing steps. In terms of iteration counts, the MG-CMM outperforms other commonly used iterative approaches, showing that MG-CMM successfully accelerates the convergence rate. It must also be mentioned that the number of iterations depend on the choices of iteration parameters and the type of smoother used. Further optimization of these factors can substantially increase efficiency.

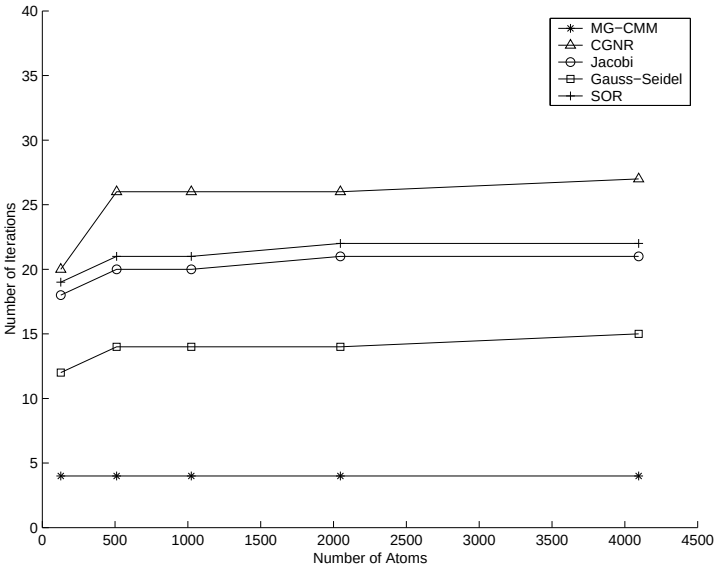


Figure 4.1: Number of iterations necessary for convergence for various iterative solvers.

The computed induced dipole moments from these iterative solvers are compared to the exact solution given by the matrix inversion. Figures 4.2-4.6 show the maximum absolute error for all methods at each iteration. The maximum absolute error is defined as: $\Delta\vec{x}^{max} = \max_i |x_i^{computed} - x_i^{exact}|$. This plot also shows the relative speed at which each solver converges. Similar results were found with all system sizes. In the Jacobi iteration and its variants GS and SOR, the error between iterations slowly converged to give small error compared to the exact solution. The error of the Jacobi, GS, and SOR ranged from 3.5×10^{-5}D to 3.4×10^{-4}D, 2.9×10^{-5}D to 6.5×10^{-5}D, and 5.4×10^{-5}D to 6.5×10^{-5}D, respectively. The conjugate gradient method steadily reduced the error, giving a substantial error

ranging from 3.7×10^{-1}D to 5.3×10^{-1}D. The MG-CMM method yielded reasonable results with errors within the range 5.0×10^{-3}D to 1.0×10^{-2}D.

In summary, MG-CMM substantially accelerated the convergence rate of the dipole iteration compared to other iterative approaches. The Jacobi-based methods had slow convergence rates, but converged with very small error. CGNR was slow to converge and yielded the least accurate values for the induced dipole compared to other iterative solvers. With all iterative problems, the initial guess can greatly dictate the convergence rate. After the first step of the MG-CMM, the error was quickly reduced by about two orders of magnitude and then declined slowly compared to the first step. This is probably because one FMG-cycle provided an excellent approximation to the exact solution. Since the initial guess in our algorithm included the CMM long-range interactions, it is expected to provide a more accurate starting approximation to the solution that can lead to faster convergence. MG-CMM has the fastest convergence rate with the most reduction in iterations and gave reasonable values for the induced dipole moments.

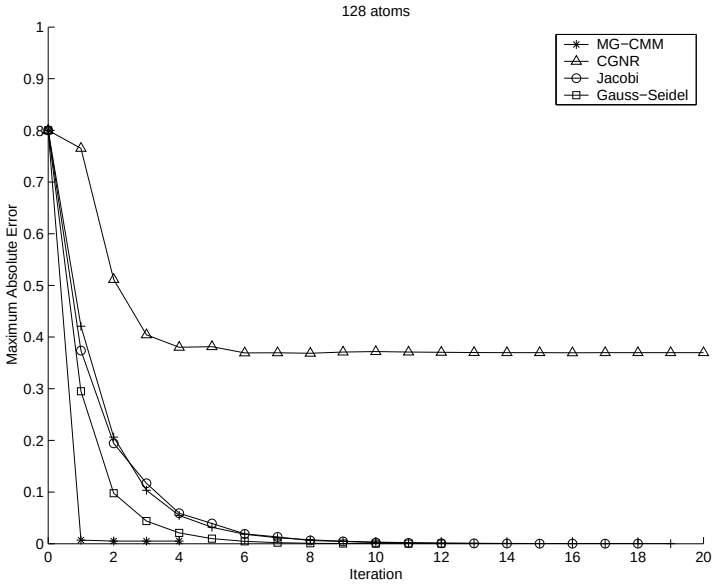


Figure 4.2: Comparison of Maximum Error of Iterative Solvers for System of 128 atoms.

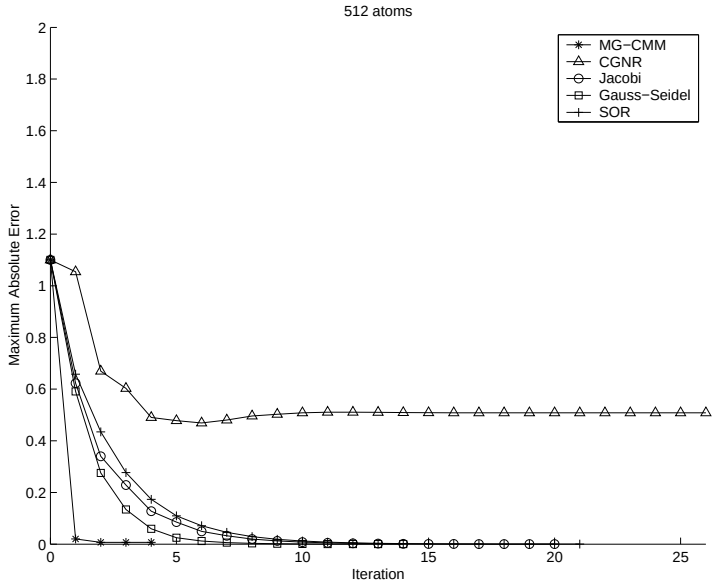


Figure 4.3: Comparison of Maximum Error of Iterative Solvers for System of 512 atoms.

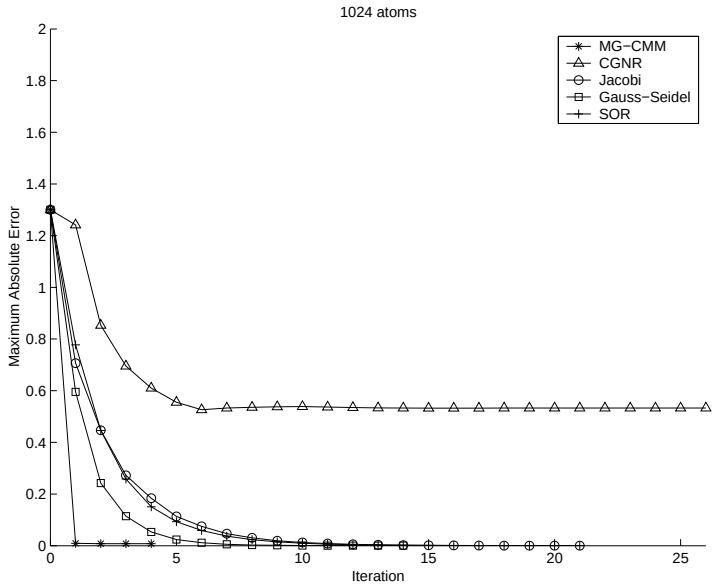


Figure 4.4: Comparison of Maximum Error of Iterative Solvers for System of 1024 atoms.

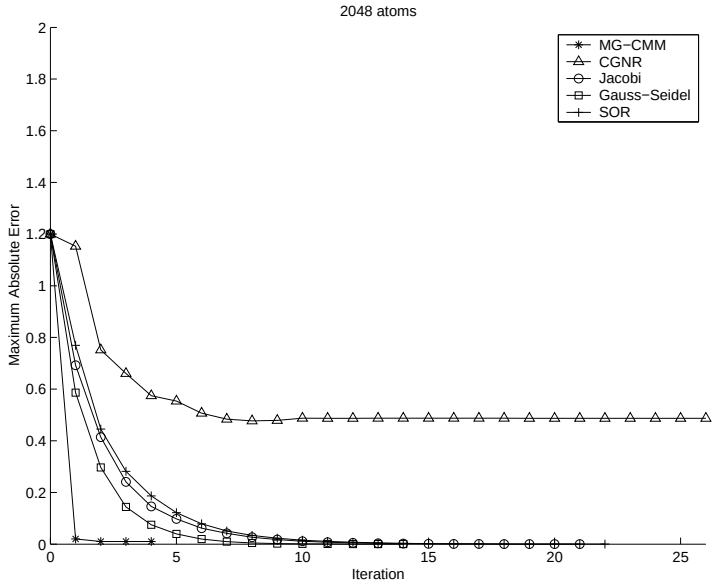


Figure 4.5: Comparison of Maximum Error of Iterative Solvers for System of 2048 atoms.

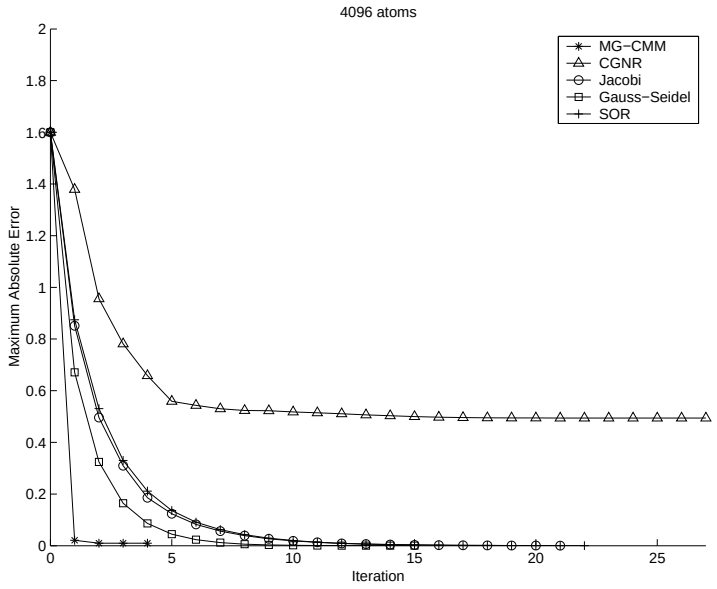


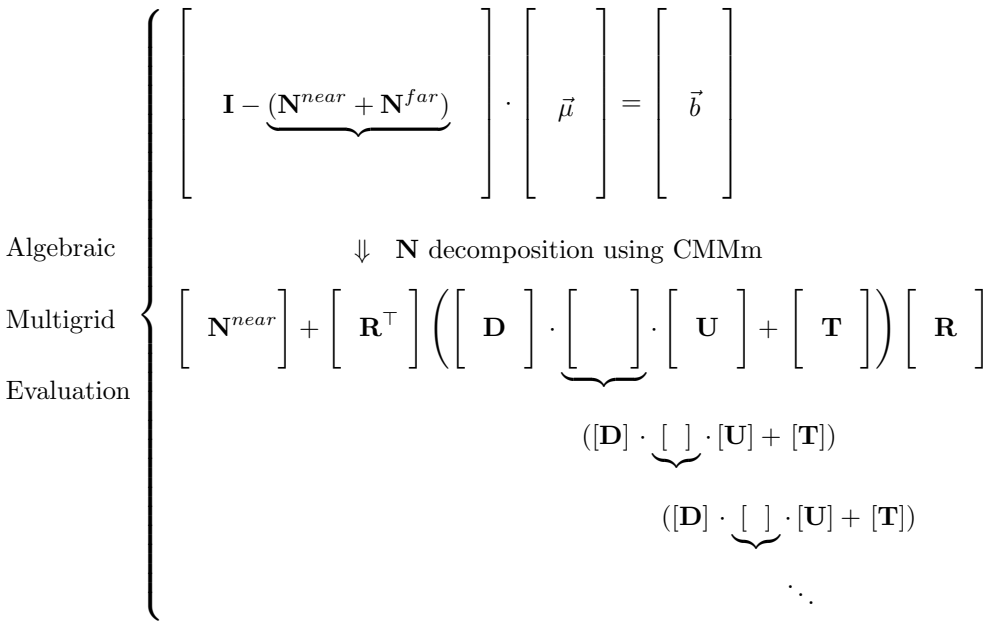
Figure 4.6: Comparison of Maximum Error of Iterative Solvers for System of 4096 atoms.

4.2 Matrix Framework for AMG-CMMm Algorithm

Extension of this MG-CMM method is necessary to treat non-uniform distributions by implementing an algebraic multigrid to replace the geometric multigrid component of the algorithm. This scheme follows the same concepts of multigrid techniques without the dependence on the physical location of the grid points and includes an automated coarsening procedure for unstructured grids (see Reference 263 for review). In Section 2.2.2, we have also casted the cell multipole method

for induced dipolar interactions in matrix form representing a series of operations that condensed into a single matrix. The combination of this matrix version of the cell multipole method and the algebraic multigrid method provides a matrix-based algorithm that encompasses the benefits of MG-CMM with further flexibility and automated computations. The CMMm matrix implicitly contains all of the cell multipole method operations. Using the CMMm matrix within the induced dipolar matrix equation for the far field component, the algebraic multigrid method is used to coarsen, smooth, and refine the solution with matrix operations. Therefore, the algebraic multigrid and cell multipole components are simultaneously solved for the induced dipole moments.

The framework of this algorithm AMG-CMMm is summarized in the following diagram.



Chapter 5

Concluding Remarks and Perspectives for the Future

The drive for the development of polarizable force fields for biomolecular systems has been facilitated by the need for more flexible methods that can handle inhomogeneous environments.[23, 277, 278, 279] Chapter 1 presents an overview of the advances made towards a polarizable force field for large biosystems. Most of the early polarizable force fields primarily focused on water models. The push towards modeling macromolecular systems is the central focus of current polarizable force field development. However, improvements to current algorithms are needed to make large-scale systems more tractable.

Using cell multipole and multigrid methods reviewed in Chapter 2 and 3,

we have developed an efficient algorithm to accelerate the rate of convergence in computing induced dipolar interactions. This scheme separates polarizable interactions into direct and indirect components. The cell multipole method is used to approximate the indirect component, where our derivations for the CMM field for dipolar systems,[219] described by Kutteh and Nicholas are used. CMM speeds up long-range electrostatic calculations by approximating the distant particles with multipole and Taylor expansions. The overall interaction is then evaluated by a multigrid solver where multigrid methods further speed up the convergence by using successive grids and relaxation methods to provide a better approximate solution. Another advantage of using these methods is that they both scale linearly with the size of the system.

The performance of this new algorithm, MG-CMM, has been investigated on uniformly-distributed one-dimensional systems of varying sizes and compared to other popular iterative solvers. MG-CMM showed a speed-up of the convergence rate by substantially reducing the number of iterations necessary for convergence, as well as showing that its rate of convergence was independent of system size. This formulation was then extended to an algorithm utilizing matrix-based cell multipole method with an algebraic multigrid (AMG-CMMm) for added flexibility in the treatment of 3-D non-uniform systems. This was accomplished by deriving a matrix form of the cell multipole method and modifying it for the treatment of dipolar

interactions. This matrix formalism is better suited to be parallelized for further efficiency. We can use parallel processing to generate the induced dipolar interactions for large polarizable systems. Parallel codes have already been implemented for both FMM[281, 282] and CMM,[221] as well as various multigrid applications.[253, 254]

Future improvements to AMG-CMMm include optimized coarsening scheme, parallel implementation, optimized data structure and efficient sparse matrix storage as this algorithm is primarily composed of matrix-vector products. The coarsening scheme used in the standard classical algebraic multigrid presented in Chapter 3 is sequential coarsening and cannot be parallelized. However, a modification of this coarsening scheme can be explored to allow parallel implementation such as the CLJP or Falgout coarsening methods.[273, 274] CLJP coarsening scheme adds random numbers between 0 and 1 to each measure, making the points distinct from one another.[273] Falgout coarsening is a hybrid of RS and CLJP coarsening strategies.[274] Further examination in the performance of other parallelizable coarsening methods in AMG is needed to find the optimal coarsening scheme for our matrix problem.[265] It is crucial to optimize the coarsening, as it greatly dictates the speed of the setup phase as well as determining the accuracy of the approximation. This algorithm involves a series of matrix operations that may be parallelized by subdividing the matrix operations among many processors to speed up calculations. Parallel implementation will further improve the efficiency

of computation. Additionally, sparse matrix storage schemes should be explored for optimal efficiency in the multiplication of nonzero matrix elements, rather than dense matrix operations, to minimize memory requirements and to avoid wasted storage of zeroes.[283, 284]

Extending our approach to include charge transfer via fluctuating charges as well as polarizable dipoles will give a better description of the system. Hybrid PD-FQ models have been shown to accurately predict the energetics of small molecule systems.[117, 118] With improved accuracy and efficiency, this approach will have general applicability in polarizable force fields for large-scale systems. The acceleration in evaluating induced dipoles found in this work is a crucial step towards the explicit inclusion of polarization in force fields of macromolecular systems, and improving the understanding of biological processes such as ion permeation, protein folding, DNA-protein interactions, and drug-protein interactions.

Appendix A

Derivations for Taylor series expansion of CMM dipole equations

All these Taylor series coefficients will have the same form if any Cartesian direction x, y, or z is substituted into the formula. They are also symmetric with respect to the order of differentiation.

For the dipole term,

$$V_f^A = \frac{\sum_\alpha A_\alpha R_\alpha}{R^3} = \frac{A_x(r_x - r_{f,x}) + A_y(r_y - r_{f,y}) + A_z(r_z - r_{f,z})}{((r_x - r_{f,x})^2 + (r_y - r_{f,y})^2 + (r_z - r_{f,z})^2)^{1/2}} \tag{A.1}$$

$$\begin{aligned}\frac{\partial V_f^A}{\partial r_x} =& \frac{R^3 A_x - (\vec{A}\cdot\vec{R})(\frac{3}{2})R(2(r_x - r_{f,x}))}{R^6} \\ =& \frac{R^2 A_x - 3 - (\vec{A}\cdot\vec{R})(r_x - r_{f,x})}{R^5} \\ \left(\frac{\partial V_f^A}{\partial r_x}\right)_0 =& \frac{6A_x r_{f,x}}{r_f^5} - \frac{12(\vec{A}\cdot\vec{r})r_{f,x}^2}{r_f^7}\end{aligned} \tag{A.2}$$

$$\begin{aligned}\left(\frac{\partial^2 V_f^A}{\partial r_x \partial r_y}\right) =& \frac{1}{R^{10}}[R^5(2(r_y - r_{f,y})A_x - 3A_y(r_x - r_{f,x})) \\ & - (R^2 A_x - 3(\vec{A}\cdot\vec{R})(r_x - r_{f,x}))\frac{5}{2}R^3 2(r_y - r_{f,y})] \\ =& \frac{1}{R^7}[R^2[(2(r_y - r_{f,y}) - 5(r_y - r_{f,y}))]A_x - 3R^2 A_y(r_x - r_{f,x}) \\ & + 15(\vec{A}\cdot\vec{R})(r_x - r_{f,x})(r_y - r_{f,x})] \\ \left(\frac{\partial^2 V_f^A}{\partial r_x \partial r_y}\right)_0 =& \frac{3r_{f,y}A_x}{r_f^5} + \frac{3A_y r_{f,x}}{r_f^5} - \frac{15(\vec{A}\cdot\vec{r_f})r_{f,x}r_{f,y}}{r_f^7}\end{aligned} \tag{A.3}$$

$$\begin{aligned}\left(\frac{\partial^2 V_f^A}{\partial r_x \partial r_x}\right) =& \frac{1}{R^{10}}[R^5(2(r_x - r_{f,x})A_x - 3A_x(r_x - r_{f,x})) - 3(\vec{A}\cdot\vec{R})) \\ & - (R^2 A_x - 3(\vec{A}\cdot\vec{R})(r_x - r_{f,x}))\frac{5}{2}R^3 2(r_x - r_{f,x})] \\ =& \frac{1}{R^7}[R^2[-3(r_x - r_{f,x})A_x) - 3A_x R^2(r_x - r_{f,x}) - 3(\vec{A}\cdot\vec{R})R^2 \\ & + 15(\vec{A}\cdot\vec{R})(r_x - r_{f,x})(r_x - r_{f,x})] \\ \left(\frac{\partial^2 V_f^A}{\partial r_x r_x}\right)_0 =& \frac{6A_x r_{f,x}}{r_f^5} - \frac{12(\vec{A}\cdot\vec{r})r_{f,x}^2}{r_f^7}\end{aligned} \tag{A.4}$$

$$
\begin{aligned}
\frac{\partial^3 V_f^A}{\partial r_x \partial r_y \partial r_z} &= \frac{\partial}{\partial r_z}\left(\frac{1}{R^7}[-3R^2(r_y - r_{f,y})A_x - 3R^2(r_x - r_{f,x})A_y \right. \\
&\left. +15(\vec{A}\cdot\vec{R})(r_x - r_{f,x})(r_y - r_{f,y})]\right) \\
&= \frac{1}{R^{14}}[R^7(-3\times 2(r_z - r_{f,z})(r_y - r_{f,y})A_x - 3\times 2(r_z - r_{f,z})(r_x - r_{f,x})A_y \\
&+ 15A_z(r_x - r_{f,x})(r_y - r_{f,y})) - (-3R^2(r_y - r_{f,y})A_x - 3R^2(r_x - r_{f,x})A_y \\
&+ 15(\vec{A}\cdot\vec{R})(r_x - r_{f,x})(r_y - r_{f,y}))\frac{7}{2}R^5\times 2(r_z - r_{f,z})] \\
&= \frac{1}{R^9}[R^2(15(r_z - r_{f,z})(r_y - r_{f,y}) + 15(r_z - r_{f,z})(r_x - r_{f,x})A_y \\
&+ 15A_z(r_x - r_{f,x})(r_y - r_{f,y}))] - \frac{105}{R^9}(\vec{A}\cdot\vec{R})(r_x - r_{f,x})(r_y - r_{f,y})(r_z - r_{f,z}) \\
\left(\frac{\partial^3 V_f^A}{\partial r_x \partial r_y \partial r_z}\right)_0 &= \frac{15}{r_f^7}(A_x r_{f,y} r_{f,z} + A_y r_{f,x} r_{f,z} + A_z r_{f,x} r_{f,y}) - \frac{105}{r_f^9}(\vec{A}\cdot\vec{r_f}) r_{f,x} r_{f,y} r_{f,z}
\end{aligned}
\tag{A.5}
$$

$$
\begin{aligned}
\frac{\partial^3 V_A}{\partial r_x \partial r_y \partial r_y} &= \frac{1}{R^{14}}[R^7(-3\times 2(r_y - r_{f,y})^2 A_x - 3R^2 A_x - 3\times 2(r_y - r_{f,y})(r_x - r_{f,x})A_y \\
&+ 15A_y(r_x - r_{f,x})(r_y - r_{f,y}) + 15(\vec{A}\cdot\vec{R})(r_x - r_{f,x}) - (-3R^2(r_y - r_{f,y})A_x \\
&- 3R^2(r_x - r_{f,x})A_y + 15(\vec{A}\cdot\vec{R})(r_x - r_{f,x})(r_y - r_{f,y}))\frac{7}{2}R^5\times 2(r_y - r_{f,y})] \\
&= \frac{1}{R^9}[R^2(15(r_y - r_{f,y})(r_y - r_{f,y})A_x + 15(r_y - r_{f,y})(r_x - r_{f,x})A_y \\
&+ 15(r_x - r_{f,x})(r_y - r_{f,y})A_y - 3R^2 A_x + 15(\vec{A}\cdot\vec{R})(r_x - r_{f,x}))] - \\
&\frac{109}{R^9}(\vec{A}\cdot\vec{R})(r_x - r_{f,x})(r_y - r_{f,y})(r_y - r_{f,y}) \\
\left(\frac{\partial^3 V_A}{\partial r_x \partial r_y \partial r_y}\right)_0 &= \frac{3}{r_f^7}(5A_x r_{f,y} r_{f,y} + 5A_y r_{f,x} r_{f,y} + A_y r_{f,x} r_{f,y} - r_f^2 A_x + 5(\vec{A}\cdot\vec{r_f}) r_{f,x}) \\
&+ \frac{105}{r_f^9}(\vec{A}\cdot\vec{r_f}) r_{f,x}) - \frac{105}{r_f^9}(\vec{A}\cdot\vec{r_f}) r_{f,x} r_{f,y} r_{f,z}
\end{aligned}
\tag{A.6}
$$

$$\begin{aligned}
\frac{\partial^3 V_f^A}{\partial r_x \partial r_x \partial r_x} &= \frac{\partial}{\partial r_x}\left(\frac{-6A_x R^2(r_x - r_{f,x}) - 3(\vec{A}\cdot\vec{R})R^2 + 15(\vec{A}\cdot\vec{R})(r_x - r_{f,x})}{R^7}\right) \\
&= \frac{1}{R^{14}}[R^7(-6A_x \times 2(r_x - r_{f,x})^2 - 6A_x R^2 - 3(\vec{A}\cdot\vec{R}) \times 2(r_x - r_{f,x}) \\
&+ 15A_x(r_x - r_{f,x})^2 + 15(\vec{A}\cdot\vec{R})2(r_x - r_{f,x})) - (-6A_x 2(r_x - r_{f,x}) \\
&- 3(\vec{A}\cdot\vec{R})R^2 + 15(\vec{A}\cdot\vec{R})(r_x - r_{f,x})^2)\frac{7}{2}R^5 \times 2(r_x - r_{f,x})] \\
&= \frac{R^2(45A_x(r_x - r_{f,x})^2 - 9A_x R^2 + 30(\vec{A}\cdot\vec{R})(r_x - r_{f,x})) - 105(\vec{A}\cdot\vec{R})(r_x - r_{f,x})^3}{R^9} \\
\left(\frac{\partial^3 V_f^A}{\partial r_x \partial r_x \partial r_x}\right)_0 &= \frac{3}{r_f^7}(15A_x r_{f,x}^2 - 3A_x r_f^2 + 10(\vec{A}\cdot\vec{r_f})r_{f,x}) + \frac{105}{r_f^9}(\vec{A}\cdot\vec{r}_f)r_{f,x}^3
\end{aligned} \tag{A.7}$$

For the quadrupole term,

$$V_f^B = \frac{\sum\limits_{\alpha\beta} 3B_{\alpha\beta}R_\alpha R_\beta - \sum\limits_{\alpha} B_{\alpha\alpha}R^2}{R^5} \tag{A.8}$$

$$\begin{aligned}
\frac{\partial V_f^B}{\partial r_x} =& \frac{1}{R^{10}}[R^5(6B_{xx}R_x + 3B_{xy}R_y + 3B_{xz}R_z + 3B_{yx}R_y + 3B_{zx}R_z - 2B_{xx}R_x - 2B_{yy}R_z \\
&+ 2B_{zz}R_x) - (3\sum_{\alpha\beta} B_{\alpha\beta}R_\alpha R_\beta - (B_{xx} + B_{yy} + B_{zz})R^2)\frac{5}{2}R^3 \cdot 2R_x] \\
\left(\frac{\partial V_f^B}{\partial r_x}\right)_0 =& \frac{-3}{r_f^5}(3B_{xx}r_{f,x} + B_{yy}r_{f,x} + B_{zz}r_{f,x} + (B_{xy} + B_{yx})r_{f,y} + (B_{xz} + B_{zx})r_{f,z}) \\
&+ \frac{15}{r_f^7}(\sum_{\alpha\beta} B_{\alpha\beta}r_{f,\alpha}r_{f,\beta})r_{f,x}
\end{aligned} \tag{A.9}$$

$$
\begin{aligned}
\frac{\partial^2 V_f^B}{\partial r_x \partial r_y} =& \frac{1}{R^9}(R^2[-45B_{xx}R_xR_y - 45B_{yy}R_xR_y - 15B_{zz}R_xR_y + 3R^2(B_{xy}+B_{yx}) \\
& - 15(B_{xy}+B_{yx})R_x^2 - 15(B_{xy}+B_{yx})R_y^2 - 15(B_{yz}+B_{zy})R_xR_z \\
& - 15(B_{xz}+B_{zx})R_yR_z] + 105(\sum_{\alpha\beta} B_{\alpha\beta}R_\alpha R_\beta)R_xR_y) \\
\left(\frac{\partial^2 V_f^B}{\partial r_x \partial r_y}\right)_0 =& \frac{-3}{r_f^7}(15B_{xx}r_{f,x}r_{f,y} + 5B_{yy}r_{f,x}r_{f,y} + 5B_{zz}r_{f,x}r_{f,y} - r_f^2(B_{xy}+B_{yx}) \\
& + 5(B_{xy}+B_{yx})r_{f,x}^2 + 5(B_{xy}+B_{yx})r_{f,y}^2 + 5(B_{yz}+B_{zy})r_{f,x}r_{f,z} \\
& + 5(B_{xz}+B_{zx})r_{f,y}r_{f,z}) + \frac{105}{r_f^9}(\sum_{\alpha\beta} B_{\alpha\beta}r_{f,\alpha}r_{f,\beta})r_{f,x}r_{f,y}
\end{aligned}
\tag{A.10}
$$

$$
\begin{aligned}
\frac{\partial^2 V_f^B}{\partial r_x \partial r_x} =& \frac{1}{R^9}(R^2[R^2(9B_{xx}+3B_{yy}+3B_{zz}) - 75B_{xx}R_x^2 - 15B_{zz}R_x^2 \\
& - 30(B_{xy}+B_{yx})R_xR_y - 30(B_{xz}+B_{zx})R_xR_z - 15(\sum_{\alpha\beta} B_{\alpha\beta}R_\alpha R_\beta)R_x^2)] \\
& + 105(\sum_{ij} B_{\alpha\beta}R_\alpha R_\beta)R_x^2) \\
\left(\frac{\partial^2 V_f^B}{\partial r_x \partial r_x}\right)_0 =& \frac{-3}{r_f^7}(30B_{xx}r_{f,x}^2 + 5B_{yy}r_{f,x}^2 + 5B_{zz}r_{f,x}^2 + 5(B_{xy}+B_{yx})r_{f,x}r_{f,y} \\
& + 5(B_{xz}+B_{zx})r_{f,x}r_{f,z} - r_f^2(3B_{xx}+B_{yy}+B_{zz}) - 5B_{xx}r_{f,x}^2 \\
& - 5(\sum_{\alpha\beta} B_{\alpha\beta}r_{f,\alpha}r_{f,\beta})r_x) + \frac{105}{r_f^9}(\sum_{\alpha\beta} B_{\alpha\beta}r_{f,\alpha}r_{f,\beta})r_{f,x}^2
\end{aligned}
\tag{A.11}
$$

$$
\begin{aligned}
\frac{\partial^3 V_f^B}{\partial r_x \partial r_y \partial r_z} =& \frac{1}{R^{11}}(R^2[315B_{xx}R_xR_yR_z + 315B_{yy}R_xR_yR_z + 315B_{zz}R_xR_yR_z \\
&+ 105(B_{xy}+B_{yx})R_x^2R_z + 105(B_{xy}+B_{yx})R_y^2R_z + 105(B_{yz}+B_{zy})R_xR_z^2 \\
&+ 105(B_{xz}+B_{zx})R_yR_z^2 + 105(B_{xz}+B_{zx})R_x^2R_y + 105(B_{yz}+B_{zy})R_xR_y^2 \\
&- 15R^2(B_{xy}+B_{yx})R_z - 15R^2(B_{yz}+B_{zy})R_x - 15R^2(B_{xz}+B_{zx})R_y] \\
&- 945(\sum_{\alpha\beta} B_{\alpha\beta}R_\alpha R_\beta)R_xR_yR_z) \\
\left(\frac{\partial^3 V_f^B}{\partial r_x \partial r_y \partial r_z}\right)_0 =& \frac{15}{r_f^9}(-21(B_{xx}+B_{yy}+B_{zz})r_{f,x}r_{f,y}r_{f,z} - 7(B_{xy}+B_{yx})r_{f,x}^2r_{f,z} \\
&- 7(B_{xy}+B_{yx})r_{f,y}^2r_{f,z} - 7(B_{yz}+B_{zy})r_{f,x}r_{f,z}^2 - 7(B_{xz}+B_{zx})r_{f,y}r_{f,z}^2 \\
&- 7(B_{xz}+B_{zx})r_{f,x}^2r_{f,y} - 7(B_{yz}+B_{zy})r_{f,x}r_{f,y}^2 + r_f^2(B_{xy}+B_{yx})r_{f,z} \\
&+ r_f^2(B_{yz}+B_{zy})r_{f,x} + r_f^2(B_{xz}+B_{zx})r_{f,y}) + \frac{945}{r_f^{11}}(\sum_{\alpha\beta} B_{\alpha\beta}r_{f,\alpha}r_{f,\beta})r_{f,x}r_{f,y}r_{f,z}
\end{aligned}
\tag{A.12}
$$

$$
\begin{aligned}
\frac{\partial^3 V_f^B}{\partial r_x \partial r_y \partial r_y} = & \frac{1}{R^{11}}(R^2[315B_{xx}R_xR_y^2 + 315B_{yy}R_xR_y^2 + 210B_{yy}R_xR_y^2 + 105B_{zz}R_xR_y^2 \\
& + 210(B_{xy}+B_{yx})R_x^2R_y + 105(B_{xy}+B_{yx})R_y^3 + 210(B_{yz}+B_{zy})R_xR_yR_z \\
& + 105(B_{xz}+B_{zx})R_y^2R_z - 21R^2(B_{xy}+B_{yx})R_y - 45R^2B_{xx}R_x - 45R^2B_{yy}R_x \\
& - 45R^2B_{zz}R_x - 24R^2(B_{xy}+B_{yx})R_y - 15R^2(B_{xz}+B_{zx})R_z] \\
& - 945(\sum_{\alpha\beta} B_{\alpha\beta}R_\alpha R_\beta)R_xR_y^2) \\
\left(\frac{\partial^3 V_f^B}{\partial r_x \partial r_y \partial r_y}\right)_0 = & \frac{3}{r_f^9}(-105B_{xx}r_{f,x}r_{f,y}^2 - 105B_{yy}r_{f,x}r_{f,y}^2 - 70B_{yy}r_{f,x}r_{f,y}^2 - 35B_{zz}r_{f,x}r_{f,y}^2 \\
& - 70(B_{xy}+B_{yx})r_{f,x}^2r_{f,y} - 35(B_{xy}+B_{yx})r_{f,x}^3 - 70(B_{yz}+B_{zy})r_{f,x}r_{f,y}r_z \\
& + 105(B_{xz}+B_{zx})R_y^2R_z - 21R^2(B_{xy}+B_{yx})R_y - 45R^2B_{xx}R_x \\
& - 45R^2B_{yy}R_x - 45R^2B_{zz}R_x - 24R^2(B_{xy}+B_{yx})R_y - 15R^2(B_{xz}+B_{zx})R_z) \\
& + \frac{945}{r_f^{11}}(\sum_{\alpha\beta} B_{\alpha\beta}r_{f,\alpha}r_{f,\beta})r_{f,x}r_{f,y}^2)
\end{aligned}
\tag{A.13}
$$

$$\begin{aligned}
\frac{\partial^3 V_f^B}{\partial r_x \partial r_x \partial r_x} &= \frac{1}{R^{11}}(R^2[-5R^2R_x(9B_{xx}+3B_{yy}+3B_{zz}) - R^2R_x(150B_{xx}+30B_{yy}+30B_{zz}) \\
&\quad - 30R^2(B_{xy}+B_{yx})R_y - 30R^2(B_{xz}+B_{zx})R_z - 15R^2(\sum_{\alpha\beta} B_{\alpha\beta}R_\alpha R_\beta) \\
&\quad - 15R^2(6B_{xx}R_x^2 + 3(B_{xy}+B_{yx})R_xR_y + 3(B_{xz}+B_{zx})R_xR_z) + 1155B_{xx}R_x^3 \\
&\quad + 105B_{yy}R_x^3 + 525(B_{xy}+B_{yx})R_x^2R_y + 525(B_{xz}+B_{zx})R_x^2R_z \\
&\quad + 105(\sum_{\alpha\beta} B_{\alpha\beta}R_\alpha R_\beta)R_x^2 + 210(\sum_{\alpha\beta} B_{\alpha\beta}R_\alpha R_\beta)R_x] - 945(\sum_{\alpha\beta} B_{\alpha\beta}R_\alpha R_\beta)R_x^3) \\
\left(\frac{\partial^3 V_f^B}{\partial r_x \partial r_x \partial r_x}\right)_0 &= \frac{3}{r_f^9}[(65B_{xx}+15B_{yy}+15B_{zz})r_f^2 r_{f,x} + 10r_f^2(B_{xy}+B_{yx})r_{f,y} \\
&\quad + 10r_f^2(B_{xz}+B_{zx})r_{f,z} - 5r_f^2(\sum_{\alpha\beta} B_{\alpha\beta}r_{f,\alpha}r_{f,\beta}) - 5r_f^2(6B_{xx}r_{f,x}^2 \\
&\quad + 3(B_{xy}+B_{yx})r_{f,x}r_{f,y} + 3(B_{xz}+B_{zx})r_{f,x}r_{f,z}) - 385B_{xx}r_{f,x}^3 \\
&\quad - 35B_{yy}r_{f,x}^3 - 175(B_{xy}+B_{yx})r_{f,x}^2 r_{f,y} - 175(B_{xz}+B_{zx})r_{f,x}^2 r_{f,z} \\
&\quad + (\sum_{\alpha\beta} B_{\alpha\beta}r_{f,\alpha}r_{f,\beta})(35r_{f,x}^2 - 70r_{f,x})] + \frac{945}{r_f^{11}}(\sum_{\alpha\beta} B_{\alpha\beta}r_{f,\alpha}r_{f,\beta})r_{f,x}^3)
\end{aligned} \tag{A.14}$$

For the octopole term,

$$V_f^C = \frac{3\sum_{\alpha\beta\gamma} C_{\alpha\beta\gamma}\left[5R_\alpha R_\beta R_\gamma - (R_\alpha\delta_{\beta\gamma} + R_\beta\delta_{\alpha\gamma} + R_\gamma\delta_{\alpha\beta})R^2\right]}{2R^7} \tag{A.15}$$

When this summation is fully expanded, there are a total of 48 terms. In order to express this more compactly, we will define the following coefficients:

$$
\begin{aligned}
D_x =& 5C_{xxx} \\
D_y =& 5C_{yyy} \\
D_z =& 5C_{zzz} \\
D_{xx} =& -\left[3C_{xxx} + C_{xyy} + C_{xzz} + C_{yxy} + C_{yyx} + C_{zxz} + C_{zzx}\right] \\
D_{yy} =& -\left[3C_{yyy} + C_{xxy} + C_{xyx} + C_{yxx} + C_{yzz} + C_{zyz} + C_{zzy}\right] \\
D_{zz} =& -\left[3C_{zzz} + C_{xxz} + C_{xzx} + C_{yyz} + C_{yzy} + C_{zxx} + C_{zyy}\right] \\
D_{xxy} =& 5(C_{xxy} + C_{xyx} + C_{yxx}) \\
D_{xxz} =& 5(C_{xxz} + C_{xzx} + C_{zxx}) \\
D_{yyx} =& 5(C_{xyy} + C_{yxy} + C_{yyx}) \\
D_{yyz} =& 5(C_{zyy} + C_{yzy} + C_{yyz}) \\
D_{zzx} =& 5(C_{xzz} + C_{zxz} + C_{zzx}) \\
D_{zzy} =& 5(C_{yzz} + C_{zyz} + C_{zzy}) \\
D_{xyz} =& 5(C_{xyz} + C_{xzy} + C_{yxz} + C_{yzx} + C_{zxy} + C_{zyx})
\end{aligned}
\tag{A.16}
$$

With these substitutions,

$$
\begin{aligned}
V_f^C = \frac{3}{2R^7}[&D_x R_x^3 + D_y R_y^3 + D_z R_z^3 + D_{xx} R_x R^2 + D_{yy} R_y R^2 + D_{zz} R_z R^2 \\
&+ D_{xxy} R_x^2 R_y + D_{xxz} R_x^2 R_z + D_{yyx} R_y^2 R_x + D_{yyz} R_y^2 R_z \\
&+ D_{zzx} R_z^2 R_x + D_{zzy} R_z^2 R_y + D_{xyz} R_x R_y R_z]
\end{aligned}
\tag{A.17}
$$

$$
\begin{aligned}
\left(\frac{\partial V_f^C}{\partial r_x}\right)_0 =& \frac{3}{2r_f^7}[(3D_x - 5D_{xx}) r_{f,x}^2 + (2D_{xxy} - 5D_{yy}) r_{f,x} r_{f,y} \\
&+ (2D_{xxz} - 5D_{zz}) r_{f,x} r_{f,z} + D_{yyx} r_{f,y}^2 + D_{zzx} r_{f,z}^2 + D_{xyz} r_{f,y} r_{f,z} \\
&+ D_{xx} r_f^2] - \frac{105}{r_f^9} \left(\sum_{\alpha\beta\gamma} C_{\alpha\beta\gamma} r_{f,\alpha} r_{f,\beta} r_{f,\gamma}\right) r_{f,x}
\end{aligned}
\tag{A.18}
$$

$$\left(\frac{\partial^2 V_f^C}{\partial r_x \partial r_y}\right)_0 = \frac{3}{2r_f^9}[(21D_x - 35D_{xx} + 14D_{yyx})r_{f,x}^2 r_{f,y} + (21D_y - 35D_{yy} + 14D_{xxy})r_{f,x} r_{f,y}^2$$
$$+ (14D_{xxz} + 14D_{yyz} - 35D_{zz})r_{f,x} r_{f,y} r_{f,z} + 7D_{yyx} r_{f,y}^3 + 7D_{xyz} r_{f,y}^2 r_{f,z}$$
$$+ 7D_{zzx} r_{f,y} r_{f,z}^2 + 7D_{xxy} r_{f,x}^3 + 7D_{zzy} r_{f,x} r_{f,z}^2 + 7D_{xyz} r_{f,x}^2 r_{f,z}$$
$$+ r^2((5D_{xx} - 2D_{yyx})r_{f,y} + (5D_{yy} - 2D_{xxy})r_{f,x} - D_{xyz} r_{f,z})]$$
$$- \frac{945}{2r^{11}}(\sum_{\alpha\beta\gamma} C_{\alpha\beta\gamma} r_{f,\alpha} r_{f,\beta} r_{f,\gamma}) r_{f,x} r_{f,y} \tag{A.19}$$

$$\left(\frac{\partial^2 V_f^C}{\partial r_x \partial r_x}\right)_0 = \frac{3}{2r_f^9}[(42D_x - 35D_{xx})r_{f,x}^3 + (28D_x xy - 35D_{yy})r_{f,x}^2 r_{f,y}$$
$$+ (28D_x xz - 35D_{zz})r_{f,x}^2 r_{f,z} + 14D_{yyx} r_{f,x} r_{f,y}^2 + 14D_{zzx} r_{f,x} r_{f,z}^2$$
$$+ 14D_{xyz} r_{f,x} r_{f,y} r_{f,z} + 35(\sum_{\alpha\beta\gamma} C_{\alpha\beta\gamma} r_{f,\alpha} r_{f,\beta} r_{f,\gamma}) + r_f^2((15D_{xx} - 6D_x)r_{f,x}$$
$$+ (5D_{yy} - 2D_{xxy})r_{f,y}(5D_{zz} - 2D_{xxz})r_{f,z})] - \frac{945}{2r^{11}}(\sum_{\alpha\beta\gamma} C_{\alpha\beta\gamma} r_{f,\alpha} r_{f,\beta} r_{f,\gamma}) r_{f,x}^2 \tag{A.20}$$

$$\left(\frac{\partial^3 V_f^C}{\partial r_x \partial r_y \partial r_z}\right)_0 = \frac{3}{2r_f^{11}}[(189D_x - 315D_{xx} + 126D_{yyx} + 126D_{zzx})r_{f,x}^2 r_{f,y} r_{f,z}$$
$$+ (189D_y - 315D_{yy} + 126D_{xxy} + 126D_{zzy})r_{f,x} r_{f,y}^2 r_{f,z}$$
$$+ (189D_z - 315D_{zz} + 126D_{xxz} + 126D_{yyz})r_{f,x} r_{f,y} r_{f,z}^2$$
$$+ 63D_{xxz} r_{f,x}^3 r_{f,y} + 63D_{yyz} r_{f,x}^3 r_{f,z} + 63D_{yyx} r_{f,y}^3 r_{f,z} + 63D_{yyz} r_{f,x} r_{f,y}^3$$
$$+ 63D_{zzy} r_{f,x} r_{f,z}^3 + 63D_{zzx} r_{f,y} r_{f,z}^3 + 63D_{xyz} r_{f,x}^2 r_{f,y}^2 + 63D_{xyz} r_{f,x}^2 r_{f,z}^2$$
$$+ 63D_{xyz} r_{f,y}^2 r_{f,z}^2 + r_f^2((35D_{zz} - 14D_{xxz} - 14D_{yyz})r_{f,x} r_{f,y}$$
$$+ (35D_{yy} - 14D_{xxy} - 14D_{zzz})r_{f,x} r_{f,z} + (35D_{xx} - 14D_{zzx} - 14D_{yyx})r_{f,y} r_{f,z}$$
$$- 7D_{xyz}(r_{f,x}^2 + r_{f,y}^2 + r_{f,z}^2) + D_{xyz} r_f^2)]$$
$$- \frac{10395}{2r_f^{13}}(\sum_{\alpha\beta\gamma} C_{\alpha\beta\gamma} r_{f,\alpha} r_{f,\beta} r_{f,\gamma}) r_{f,x} r_{f,y} r_{f,z} \tag{A.21}$$

$$\left(\frac{\partial^3 V_f^C}{\partial r_x \partial r_y \partial r_y}\right)_0 = \frac{3}{2r_f^{11}}[(189D_x - 315D_{xx} + 252D_{yyx})r_{f,x}^2 r_{f,y}^2$$
$$+ (378D_y - 315D_{yy} + 126D_{xxy})r_{f,x} r_{f,y}^3$$
$$+ (315D_{zz} + 126D_{xxz} + 252D_{yyz})r_{f,x} r_{f,y}^2 r_{f,z}$$
$$+ 63D_{yyx} r_{f,y}^4 + 63D_{xyz} r_{f,y}^3 r_{f,z} + 63D_{zzx} r_{f,y}^2 r_{f,z}^2 + 126D_{xxy} r_{f,x}^3 r_{f,y}$$
$$+ 126D_{zzy} r_{f,x} r_{f,y} r_{f,z}^2 + 126D_{xyz} r_{f,x}^2 r_{f,y} r_{f,z}$$
$$+ r_f^2((-21D_x + 35D_{xx} - 14D_{yyx})r_{f,x}^2 + (35D_{xx} - 35D_{yyx})r_{f,y}^2$$
$$- 7D_{zzx} r_z^2 + (-42D_y + 535D_{yy} - 42D_{xxy})r_{f,x} r_{f,y} - 21D_{xyz} r_{f,y} r_{f,z}$$
$$+ (35D_{zz} - 14D_{xxz} - 14D_{yyz})r_{f,x} r_{f,z} + (2D_{yyx} - 5D_{xx})r_f^2$$
$$+ 315(\sum_{\alpha\beta\gamma} C_{\alpha\beta\gamma} r_{f,\alpha} r_{f,\beta} r_{f,\gamma}) r_{f,x})] - \frac{10395}{2r_f^{13}}(\sum_{\alpha\beta\gamma} C_{\alpha\beta\gamma} r_{f,\alpha} r_{f,\beta} r_{f,\gamma}) r_{f,x} r_{f,y}^2 \tag{A.22}$$

$$\left(\frac{\partial^3 V_f^C}{\partial r_x \partial r_x \partial r_x}\right)_0 = \frac{3}{2r_f^{11}}[(567D_x - 315D_{xx})r_{f,x}^4 + (-315D_{yy} + 378D_{xxy})r_{f,x}^3 r_{f,y}$$
$$+ (-315D_{zz} + 378D_{xxz})r_{f,x}^2 r_{f,z} + 189D_{yyx}r_{f,x}^2 r_{f,y}^2 + 189D_{zzx}r_{f,x}^2 r_{f,z}^2$$
$$+ 63D_{xyz}r_{f,x}^2 r_{f,y} r_{f,z} - 945(\sum_{\alpha\beta\gamma} C_{\alpha\beta\gamma} r_{f,\alpha} r_{f,\beta} r_{f,\gamma}) r_{f,x}$$
$$+ r_f^2((-189D_x + 210D_{xx})r_{f,x}^2 + (105D_{yy} - 84D_{xxy})r_{f,x} r_{f,y}$$
$$+ (105D_{zz} - 84D_{xxz})r_{f,x} r_{f,z} - 21D_{yyx}r_{f,y}^2$$
$$- 21D_{zzx}r_{f,z}^2 + r_f^2(6D_x - 15D_x x))] - \frac{10395}{2r_f^{13}}(\sum_{\alpha\beta\gamma} C_{\alpha\beta\gamma} r_{f,\alpha} r_{f,\beta} r_{f,\gamma}) r_{f,x}^3 \tag{A.23}$$

References

[1] Griffiths, D.J. *Introduction to Electrodynamics.* Prentice Hall: Englewood Cliffs, (3rd Edn) 1989.

[2] Berendsen, H.J.C.; Grigera, J.R.; Straatsma, T.P. The missing term in effective pair potentials. *J. Phys. Chem.*, **1987**, *91(24)*, 6269-6271.

[3] Watanabe, K.; Klein, M.L. Effective pair potentials and the properties of water. *Chem. Phys.*, **1989**, *131(2-3)*, 157-167.

[4] Lamoureux, G., MacKerell, AD. and Roux, B.: A simple polarizable model of water based on classical Drude oscillators. *J. Chem. Phys.*, **2003**, *119(10)*, 5185-5197.

[5] Halgren, T.A.; Damm, W. Polarizable force fields. *Curr. Opin. Struct. Biol.*, **2001**, *11(2)*, 236-242.

[6] Ponder, J.W.; Case, D.A. Force fields for protein simulations. *Advances in Protein Chemistry*, **2003**, *66*, 27-85.

[7] Cornell, W.D.; Cieplak, P.; Bayly, C.I.; Gould, I.R.; Merz, K.M.; Ferguson, D.M.; Spellmeyer, D.C.; Fox, T.; Caldwell, J.W.; Kollman, P.A. A 2nd generation force field for the simulation of proteins, nucleic acids, and organic molecules. *J. Am. Chem. Soc.*, **1995**, *117(19)*, 5179-5197.

[8] Berendsen, H.J.C.; Postma, J.P.M.; van Gunsteren, W.F.; Hermans, J. Interaction models for water in relation to protein hydration, *Intermolecular Forces*, **1981**, 331-342.

[9] Jorgensen, W.L. Quantum and statistical mechanical studies of liquids .10. Transferable intermolecular potential functions for water, alcohols, and ethers - Application to liquid water. *J. Am. Chem. Soc.*, **1981**, *103(2)*, 335-340.

[10] Dang, L. X.; Chang, T.-M. Molecular Dynamics Study of Water clusters, liquid, and liquid–vapor interface of water with Many-Body Potentials. *J. Chem Phys.*, **1997**, *106*, 8149-8159.

[11] Patel, S.; Mackerell, A.D.; Brooks, C.L. CHARMM fluctuating charge force field for proteins: II - Protein/solvent properties from molecular dynamics simulations using a nonadditive electrostatic model. *J. Comput. Chem.*, **2004**, *25(12)*, 1504-1514.

[12] Chang, T.M.; Dang, L.X. Molecular dynamics simulations of CCl4-H2O liquid-liquid interface with polarizable potential models. *J. Chem. Phys.*, **1996**, *104(17)*, 6772-6783.

[13] Chang, T.M.; Peterson, K.A.; Dang, L.X. Molecular dynamics of liquid, interface, and ionic solvation of polarizable carbon-tetrachloride *J. Chem. Phys.*, **1995**, *103(17)*, 7502-7513.

[14] Kuyucak, S.; Andersen, O.S.; Chung, S.H. Models of permeation in ion channels. *Rep. Prog. Phys.*, **2001**, *64(11)*, 1427-1472.

[15] Rullmann, J.A.C.; Van Duijnen, P.Th. A Polarizable Water Model for Calculation of Hydration Energies. *Mol. Phys.*, **1988**, *63(3)*, 451-475.

[16] Ahlström, P.; Wallqvist, A.; Engström, S.; Jönsson, B. A Molecular-Dynamics Study of Polarizable Water. *Mol. Phys.*, **1989**, *68(3)*, 563-581.

[17] Sprik, M. Computer-simulation of the Dynamics of Induced Polarization Fluctuations in Water. *J. Phys. Chem.*, **1991**, *95(8)*, 2283-2291.

[18] Kuwajima, S.; Warshel, A. Incorporating electric polarizabilities in water-water interaction potentials. *J. Phys. Chem.*, **1990**, *94(1)*, 460-466.

[19] Dang, L.X; Rics, J.E.; Caldwell, J.; Kollman, P.A. Ion solvation in polarizable water - Molecular dynamics simulations. *J. Am. Chem. Soc.*, **1991**, *113(7)*, 2481-2486.

[20] Caldwell, J.; Dang, L.X.; Kollman, P.A. Implementation of Nonadditive Intermolecular Potentials by Use of Molecular Dynamics - Development of a Water Water Potential and Water Ion Cluster Interactions. *J. Am. Chem. Soc.*, **1990**, *112(25)*, 9144-9147.

[21] Van Belle, D.; Froeyen, M.; Lippens, G.; Wodak, S.J. Molecular dynamics simulation of polarizable water by an extended Lagrangian method. *Mol. Phys.*, **1992**, *77(2)*, 239-255.

[22] Gao, J.L. A molecular-orbital derived polarization potential for liquid water. *J. Chem. Phys.*, **1998**, *109(6)*, 2346-2354.

[23] Dang, L.X. Importance of polarization effects in modeling the hydrogen bond in water using classical molecular dynamics techniques. *J. Phys. Chem. B*, **1998**, *102(3)*, 620-624.

[24] Guo, H.; Salahub, D.R. Cooperative hydrogen bonding and enzyme catalysis. *Angew. Chem., Int. ed. Eng* **1998**, *37(21)*, 2985-2990.

[25] Ludwig, R.; Weinhold, F.; Farrar, T.C. Theoretical study of hydrogen bonding in liquid and gaseous N-methylformamide. *J. Chem. Phys.*, **1997**, *107(2)*, 499-507.

[26] Xu, H.F.; Stern, H.A.; Berne, B.J. Can water polarizability be ignored in hydrogen bond kinetics? *J. Phys. Chem. B*, **2002**, *106(8)*, 2054-2060.

[27] Guo, H.; Gresh, N.; Roques, B.P.; Salahub, D.R. Many-body effects in systems of peptide hydrogen-bonded networks and their contributions to ligand binding: A comparison of the performances of DFT and polarizable molecular mechanics. *J. Phys. Chem. B*, **2000**, *104(41)*, 9746-9754.

[28] Dang, L.X. Computer simulation studies of ion transport across a liquid/liquid interface. *J. Phys. Chem. B*, **1999**, *103(39)*, 8195-8200.

[29] Dang, L.X.; Feller, D. Molecular dynamics study of water-benzene interactions at the liquid/vapor interface of water. *J. Phys. Chem. B*, **2000**, *104(18)*, 4403-4407.

[30] Archontis, G.; Leontidis, E. Dissecting the stabilization of iodide at the air-water interface into components: A free energy analysis. *Chem. Phys. Lett.*, **2006**, *420(1-3)*, 199-203.

[31] Grossfield, A.; Ren, P.Y.; Ponder, J.W. Ion solvation thermodynamics from simulation with a polarizable force field. *J. Am. Chem. Soc.*, **2003**, *125(50)*, 15671-15682.

[32] Stuart, S.J.; Berne, B.J. Effects of polarizability on the hydration of the chloride ion. *J. Phys. Chem*, **1996**, *100(29)*, 11934-11943.

[33] Dang, L.X.; Smith, D.E. Molecular dynamics simulations of aqueous ionic clusters using polarizable water. *J. Chem. Phys.*, **1993**, *99(9)*, 6950-6956.

[34] Dang, L.X. The Nonadditive Intermolecular Potential for Water Revised. *J. Chem. Phys.*, **1992**, *97(4)*, 2659-2660.

[35] Roux, B.; Karplus, M. Potential energy function for cation-peptide interactions - An ab initio study *J. Comp. Chem.*, **1995**, *16(6)*, 690-704.

[36] Kim, K.S.; Im, K.S.; Lee, J.Y.; Lee, S.J.; Ha, T.K.; Kim, D.H. On binding forces between aromatic ring and quarternary ammonium compound. *J. Am. Chem. Soc.*, **1994**, *116(16)*, 7399-7400.

[37] Roux, B. Nonadditivity in cation-peptide interactions - A molecular dynamics and ab initio study of Na+ in the gramicidin channel. *Chem. Phys. Lett.*, **1993**, *212(3-4)*, 231-240.

[38] Rick, S.W.; Stuart, S.J. Potentials and algorithms for incorporating polarizability in computer simulations. *Rev. Comput. Chem.*, **2002**, *18*, 89-146.

[39] Roux, B. Computational studies of the gramicidin channel. *Acc. Chem. Res.*, **2002**, *35(6)*, 366-375.

[40] Bastug, T.; Patra, S.M.; Kuyucak, S. Molecular dynamics simulations of gramicidin A in a lipid bilayer: From structure-function relations to force fields. *Chem. Phys. Lipids*, **2006**, *141(1-2)*, 197-204.

[41] Applequist, J.; Carl, J.R.; Fung, K.-K. An Atom Dipole Interaction Model for Molecular Polarizability. Application to Polyatomic Molecules and Determination of Atom Polarizabilities. *J. Am. Chem. Soc.*, **1972**, *94*, 2952-2960.

[42] Applequist, J. An Atom Dipole Interaction Model for Molecular Optical Properties. *Acc. Chem. Res*, **1977**, *10*, 79-85.

[43] Pollock, E.L.; Alder, B.J. Effective field of a dipole in polarizable fluids. *Phys. Rev. Lett.*, **1977**, *39(5)*, 299-302.

[44] Vesely, F.J. N-Particle Dynamics of Polarizable Stockmayer-Type Molecules. *J. Comput. Phys.*, **1977**, *24*, 361.

[45] Bode, K.A.; Applequist, J. A New Optimization of Atom Polarizabilities in Halomethanes, Aldehydes, Ketones, and Amides by Way of the Atom Dipole Interaction Model. *J. Phys. Chem.*, **1996**, *100*, 17820-17824.

[46] Thole, B.T. Molecular Polarizabilities Calculated with a Modified Dipole Interaction. *Chem. Phys.*, **1981**, *59(3)*, 341-350.

[47] Birge, R.R. Calculation of molecular polarizabilities using an anisotropic atom point dipole interaction model which includes the effect of electron repulsion. *J. Chem. Phys.*, **1980**, *72*, 5312-5319.

[48] Cieplak, P.; Kollman, P.; Lybrand, T. A New Water Potential Including Polarization - Application to Gas-Phase, Liquid, and Crystal. Properties of Water *J. Chem. Phys.*, **1990**, *92(11)*, 6755-6760.

[49] Dinur, U. Molecular Polarizabilities From Electronectivity Equalization Models. *J. Phys. Chem.*, **1993**, *97(30)*, 7894-7898.

[50] Lybrand, T.P.; Kollman, P.A. Water-water and water-ion potential functions including terms for many-body effects. *J. Chem. Phys.*, **1985**, *83(6)*, 2923-2933.

[51] Cieplak, P.; Lybrand, T.P.; Kollman, P.A. Calculation of free energy changes in ion water clusters using nonadditive potentials and the Monte Carlo method. *J. Chem. Phys.*, **1987**, *86(11)*, 6393-6403.

[52] Piquemal, J.P.; Perera, L.; Cisneros, G.A.; Ren, P.Y.; Pedersen, L.G.; Darden, T.A. Towards accurate solvation dynamics of divalent cations in water using the polarizable amoeba force field: From energetics to structure. *J. Chem. Phys.*, **2006**, *125(5)*, Art. No. 054511.

[53] Stillinger, F.H., David, C.W. Polarization Model for Water and its Ionic Dissociation Products. *J. Chem. Phys.*, **1978**, *69(4)*, 1473-1484.

[54] Sprik, M.; Klein, M.L. A Polarizable Model for Water using Distributed Charge Sites. *J. Chem. Phys.*, **1988**, *89(12)*, 7556-7560.

[55] Caldwell, J.W.; Kollman, P.A. Structure and Properties of Neat Liquids using Nonadditive Molecululular Dynamics - Water, Methanol, and N-methylacetamide. *J. Phys. Chem.*, **1995**, *99(16)*, 6208-6219.

[56] Ojamae, L.; Shavitt, I.; Singer, S.J. Potential models for simulations of the solvated proton in water. *J. Chem. Phys.*, **1998**, *109(13)*, 5547-5564.

[57] Rivera, J.L.; Starr, F.W.; Paricaud, P.; Cummings, P.T. Polarizable contributions to the surface tension of liquid water. *J. Chem. Phys.*, **2006**, *125(9)*, Art. No. 094712.

[58] Warshel, A.; Levitt, M. Theoretical Studies of Enzymic Reactions - Dielectric, Electrostatic and Steric Stabilization of Carbonium-ion in Reaction of Lysozyme. *J. Mol. Biol.*, **1976**, *103(2)*, 227-249.

[59] Russell, S.T.; Warshel, A. Calculations of electrostatic energies in proteins - The energetics of ionized groups in bovine pancreatic trypsin-inhibitor *J. Mol. Biol.*, **1985**, *185(2)*, 389-404.

[60] Van Belle, D.; Couplet, I.; Prevost, M.; Wodak, S.J.; Calculations of Electrostatic Properties in Proteins - Analysis of Contributions from Induced Protein Dipoles. *J. Mol. Biol.*, **1987**, *198(4)*, 721-735.

[61] Wang, Z.X.; Zhang, W.; Wu, C.; Lei, H.X.; Cieplak, P.; Duan, Y. Strike a balance: Optimization of backbone torsion parameters of AMBER polarizable force field for simulations of proteins and peptides. *J. Comput. Chem.*, **2006**, *27(6)*, 781-790.

[62] Harder, E.; Kim, B.C.; Friesner, R.A.; Berne, B.J. Efficient simulation method for polarizable protein force fields: Application to the simulation of BPTI in liquid. *Journal of Chemical Theory and Computation*, **2005**, *1(1)*, 169-180.

[63] Babin, V.; Baucom, J.; Darden, T.A.; Sagui, C. Molecular dynamics simulations of DNA with polarizable force fields: Convergence of an ideal B-DNA structure to the crystallographic structure. *J. Phys. Chem. B*, **2006**, *110(23)*, 11571-11581.

[64] Guo, H.; Karplus, M. Solvent influence on the stability of the peptide hydrogen bond - A supramolecular cooperative effect. *J. Phys. Chem.*, **1994**, *98(29)*, 7104-7105.

[65] van der Vaart, A.; Bursulaya, B.D.; Brooks, C.L.; Merz, K.M. Are many-body effects important in protein folding? *J. Phys. Chem. B*, **2000**, *104(40)*, 9554-9563.

[66] Cubero, E.; Luque, F.J.; Orozco, M. Is polarization important in cation-pi interactions? *Proc. Nat. Acad. Sci. USA*, **1998**, *95(11)*, 5976-5980.

[67] Gresh, N.; Guo, H.; Salahub, D.R.; Roques, B.P.; Kafafi, S.A. Critical role of anisotropy for the dimerization energies of two protein-protein recognition motifs: cis-N-methylacetamide versus a beta-sheet conformer of alanine dipeptide. A joint ab initio, density functional theory, and molecular mechanics investigation *J. Am. Chem. Soc.*, **1999**, *121(34)*, 7885-7894.

[68] Stillinger, F.H. Dynamics and ensemble averages for the polarization models of molecular interactions. *J. Chem. Phys.*, **1979**, *71(4)*, 1647-1651.

[69] Bernardo, D. N.; Ding, Y.; Krogh-Jespersen, K.; Levy, R.M. An Anisotropic Polarizable Water Model: Incorporation of All-Atom Polarizabilities into Molecular Mechanics Force Fields. *J. Phys. Chem.*, **1994**, *98*, 4180-4187.

[70] Yu, H.B.; van Gunsteren, W.F. Accounting for polarization in molecular simulation. *Comp. Phys. Comm.*, **2005**, *172(2)*, 69-85.

[71] Bernardo, D. N.; Ding, Y.; Krogh-Jespersen, K.; Levy, R.M. Evaluating Polarizable Potentials on Distributed Memory Parallel Computers: Program Development and Applications. *J. Comput. Chem.*, **1995**, *16*, 1141-1152.

[72] Wallqvist, A.; Berne, B. J. Effective Potentials for Liquid Water Using Polarizable and Nonpolarizable Models. *J. Phys. Chem.*, **1993**, *97*, 13841-13851.

[73] Gao, J.; Pavelites, J.J.; Habibollazadeh, D. Simulation of Liquid Amides Using a Polarizable Intermolecular Potential Function. *J. Phys. Chem.*, **1996**, *100*, 2689-2697.

[74] Burnham, C. J.; Li, J.; Xantheas, S.S.; Leslie, M. The Parametrization of a Thole-type All-atom Polarizable Water Model from first principles and its application to the study of water clusters (n=2-21) and the phonon spectrum of ice Ih. *J. Chem. Phys.*, **1999**, *110*, 4566-4581.

[75] Jensen, L.; Åstrand, P.-O.; Osted, A.; Kongsted, J.; Mikkelsen, K.V. Polarizability of molecular clusters as calculated by a dipole interaction model. *J. Chem. Phys.*, **2002**, *116*, 4001-4010.

[76] Gresh, N. Development, validation, and applications of anisotropic polarizable molecular mechanics to study ligand and drug-receptor interactions. *Current Pharmaceutical Design*, **2006**, *12(17)*, 2121-2158.

[77] Barnes, P.; Finney, J.L.; Nicholas, J.D.; Quinn, J.E. Cooperative Effects in Simulated Water. *Nature*, **1979**, *282(5738)*, 459-464.

[78] Kozack, R.E.; Jordan, P.C. Polarizability effects in a 4-charge model for water. *J. Chem. Phys.*, **1992**, *96(4)*, 3120-3130.

[79] Brodholt, J,; Sampoli, M.; Vallauri, R. Parameterizing a polarizable intermolecular potential for water. *Mol. Phys.*, **1995**, *86(1)*, 149-158.

[80] Chialvo, A.A.; Cummings, P.T. Engineering a simple polarizable model for the molecular simulation of water applicable over wide ranges of state conditions. *J. Chem. Phys*, **1996**, *105(18)*, 8274-8281.

[81] Ding, Y.B.; Bernardo, D.N., Kroghjespersen, K.; Levy, R.M. Solvation Free energies of small amides and amines from molecular dynamics free energy perturbation simulations using pairwise additive and many-body polarizable potentials. *J. Phys. Chem.*, **1995**, *99(29)*, 11575-11583.

[82] Meng, E.C.; Caldwell, J.W.; Kollman, P.A. Investigating the anomalous solvation free energies of amines with a polarizable potential. *J. Phys. Chem.*, **1996**, *100(6)*, 2367-2371.

[83] Sakharov, D.V.; Lim, C. Zn protein simulations including charge transfer and local polarization effects. J. Am. Chem. Soc., **2005**, *127(13)*, 4921-4929.

[84] Anisimov, V.M.; Lamoureux, G.; Vorobyov, I.V.; Huang, N.; Roux, B.; MacKerell, A.D. Determination of electrostatic parameters for a polarizable force field based on the classical Drude oscillator. *J. Chem. Theory Comput.*, **2005**, *1(1)*, 153-168.

[85] Baucom, J.; Transue, T.; Fuentes-Cabrera, M.; Krahn, J.M.; Darden, T.A.; Sagui, C. Molecular dynamics simulations of the d(CCAACGTTGG)(2) decamer in crystal environment: Comparison of atomic point-charge, extra-point, and polarizable force fields. *J. Chem. Phys.*, **2004**, *121(14)*, 6998-7008.

[86] Lamoureux, G.; Harder, E.; Vorobyov, I.V.; Roux, B.; MacKerell, A.D. A polarizable model of water for molecular dynamics simulations of biomolecules. *Chem. Phys. Lett.*, **2006**, *418(1-3)*, 245-249.

[87] de Leeuw N.H.; Parker, S.C. Molecular dynamics simulation of MgO surfaces in liquid water using a shell model potential for water. *Phys. Rev. B*, **1998**, *58(20)*, 13901-13908.

[88] van Maaren, P.J.; van der Spoel, D. Molecular dynamics simulations of water with novel shell-model potentials. *J. Phys. Chem. B*, **2001**, *105(13)*, 2618-2626.

[89] Yu, H.B.; Hansson, T.; van Gunsteren, W.F. Development of a simple, self-consistent polarizable model for liquid water. *J. Chem. Phys.*, **2003**, *118(1)*, 221-234.

[90] Yu, H.B.; van Gunsteren, W.F. Charge-on-spring polarizable water models revisited: From water clusters to liquid water to ice. *J. Chem. Phys.*, **2004**, *121(19)*, 9549-9564.

[91] Archontis, G.; Leontidis, E.; Andreou, G. Attraction of iodide ions by the free water surface, revealed by simulations with a polarizable force field based on drude oscillators. *J. Phys. Chem. B*, **2005**, *109(38)*, 17957-17966.

[92] Lamoureux, G.; Roux, B. Modeling induced polarization with classical Drude oscillators: Theory and molecular dynamics simulation algorithm. *J. Chem. Phys.*, **2003**, *119(6)*, 3025-3039.

[93] Lamoureux, G.; Roux, B. Absolute hydration free energy scale for alkali and halide ions established from simulations with a polarizable force field. *J. Chem. Phys. B*, **2006**, *110(7)*, 3308-3322.

[94] Vorobyov, I.V.; Anisimov, V.M.; MacKerell, A.D. Polarizable empirical force field for alkanes based on the classical drude oscillator model. *J. Phys. Chem. B*, **2005**, *109(40)*, 18988-18999.

[95] Yu, H.B.; Geerke, D.P.; Liu, H.Y.; van Gunsteren, W.E. Molecular dynamics simulations of liquid methanol and methanol-water mixtures with polarizable models. *J. Comput. Chem.*, **2006**, *27(13)*, 1494-1504.

[96] Rick, S.W.; Stuart, S.J.; Berne, B.J. Dynamical Fluctuating Charge Force-Fields - Application to Liquid Water. *J. Chem. Phys.*, **1994**, *101(7)*, 6141-6156.

[97] Rick, S.W.; Berne, B.J. Free energy of the hydrophobic interaction from molecular dynamics simulations: The effects of solute and solvent polarizability. *J. Phys. Chem. B*, **1997**, *101(49)*, 10488-10493.

[98] Banks, J.L.; Kaminski, G.A.; Zhou, R.H.; Mainz, D.T.; Berne, B.J.; Friesner, R.A. Parametrizing a polarizable force field from ab initio data. I. The fluctuating point charge model. *J. Chem. Phys.*, **1999**, *110(2)*, 741-754.

[99] Rappe, A.K.; Goddard, W.A. Charge Equilibration for Molecular Dynamics Simulations. *J. Phy. Chem.*, **1991**, *95(8)*, 3358-3363.

[100] Patel, S.; Brooks, C.L. CHARMM fluctuating charge force field for proteins: I parameterization and application to bulk organic liquid simulations. *J. Comput. Chem.*, **2004**, *25(1)*, 1-15.

[101] Ando, K. A stable fluctuating-charge polarizable model for molecular dynamics simulations: Application to aqueous electron transfers. *J. Chem. Phys.*, **2001**, *115(11)*, 5228-5237.

[102] Stuart, S.J.; Berne, B.J. Surface curvature effects in the aqueous ionic solvation of the chloride ion. *J. Phys. Chem. A*, **1999**, *103(49)*, 10300-10307.

[103] Chen, B.; Potoff, J.J.; Siepmann, J.I. Adiabatic nuclear and electronic sampling Monte Carlo simulations in the Gibbs ensemble: Application to polarizable force fields for water. *J. Phys. Chem. B*, **2000**, *104(10)*, 2378-2390.

[104] Chen, B.; Xing, J.H.; Siepmann, J.I. Development of polarizable water force fields for phase equilibrium calculations. *J. Phys. Chem. B*, **2000**, *104(10)*, 2391-2401.

[105] Krishnan, M.; Verma, A.; Balasubramanian, S. Computer simulation study of water using a fluctuating charge model. *Proc. Indian Acad. Sci. Chem. Sci.*, **2001**, *113(5-6)*, 579-590.

[106] Rick, S.W.; Berne, B.J. Dynamical fluctuating charge force fields: The aqueous solvation of amides. *J. Am. Chem. Soc.*, **1996**, *118(3)*, 672-679.

[107] Patel, S.; Brooks, C.L. Structure, thermodynamics, and liquid-vapor equilibrium of ethanol from molecular dynamics simulations using nonadditive interactions. *J. Chem. Phys.*, **2005**, *123(16)*, Art. No. 164502.

[108] Patel, S.A.; Brooks, C.L. Revisiting the hexane-water interface via molecular dynamics simulations using nonadditive alkane-water potentials. *J. Chem. Phys.*, **2006**, *124(20)*, Art. No. 204706.

[109] Patel, S.; Brooks, C.L. Fluctuating charge force fields: Recent developments and applications from small molecules to macromolecular biological systems. *Molecular Simulation*, **2006**, *32(3-4)*, 231-249.

[110] Sanderson, R.T. An Interpretation of Bond Lengths and a Classification of Bonds. *Science*, **1951**, *114*, 670.

[111] Mulliken, R.S. A New Electronegativity Scale: Together with Data on Valence States and an Ionization Potential and Electron Affinities. *J. Chem. Phys.*, **1934**, *2*, 782-793.

[112] Parr, R.G.; Pearson, R.G. Absolute Hardness: Companion Parameter to Absolute Electronegativity. *J. Am. Chem. Soc.*, **1983**, *105*, 7512-7516.

[113] Masia, M.; Probst, M.; Rey, R. On the performance of molecular polarization methods. I. Water and carbon tetrachloride close to a point charge. *J. Chem. Phys.*, **2004**, *121(15)*, 7362-7378.

[114] Masia, M.; Probst, M.; Rey, R. On the performance of molecular polarization methods close to a point charge. *Comp. Phys. Comm.*, **2005**, *169(1-3)*, 331-334.

[115] Banks, J.L.; Beard, H.S.; Cao, Y.X.; Cho, A.E.; Damm, W.; Farid, R.; Felts, A.K.; Halgren, T.A.; Mainz, D.T.; Maple, J.R.; Murphy, R.; Philipp, D.M.; Repasky, M.P.; Zhang, L.Y.; Berne, B.J.; Friesner, R.A.; Gallicchio, E.; Levy, R.M. Integrated modeling program, applied chemical theory (IMPACT). *J. Comp. Chem.*, **2005**, *26(16)*, 1752-1780.

[116] Field, M.J. Hybrid quantum mechanical molecular mechanical fluctuating charge models for condensed phase simulations. *Mol. Phys.*, **1997**, *91(5)*, 835-845.

[117] Stern, H.A.; Kaminski, G.A.; Banks, J.L.; Zhou, R.H.; Berne, B.J.; Friesner, R.A. Fluctuating charge, polarizable dipole, and combined models: Parameterization from ab initio quantum chemistry. *J. Phys. Chem. B*, **1999**, *103(22)*, 4730-4737.

[118] Stern, H.A.; Rittner, F.; Berne, B.J.; Friesner, R.A. Combined fluctuating charge and polarizable dipole models: Application to a five-site water potential function. *J. Chem. Phys.*, **2001**, *115(5)*, 2237-2251.

[119] Chialvo, A.A.; Cummings, P.T. Microstructure of ambient and supercritical water. Direct comparison between simulation and neutron scattering experiments. *J. Phys. Chem.*, **1996**, *100(4)*, 1309-1316.

[120] Warshel, A. Calculations of chemical processes in solutions. *J. Phys. Chem.*, **1979**, *83(12)*, 1640-1652.

[121] Smith, D.E.; Dang, L.X. Computer Simulations of NaCl association in polarizable water. *J. Chem. Phys.*, **1994**, *100(5)*, 3757-3766.

[122] Jedlovszky, P.; Richardi, J. Comparison of different water models from ambient to supercritical conditions: A Monte Carlo simulation and molecular Ornstein-Zernike study. *J. Chem. Phys.*, **1999**, *110(16)*, 8019-8031.

[123] Jedlovszky, P.; Vallauri, R. Thermodynamic and structural properties of liquid water around the temperature of maximum density in a wide range of pressures: A computer simulation study with a polarizable potential model. *J. Chem. Phys.*, **2001**, *115(8)*, 3750-3762.

[124] Burnham, C.J.; Xantheas, S.S. Development of transferable interaction models for water. III. Reparametrization of an all-atom polarizable rigid model (TTM2-R) from first principles. *J. Chem. Phys.*, **2002**, *116(4)*, 1500-1510.

[125] Ren, P.Y.; Ponder, J.W. Polarizable atomic multipole water model for molecular mechanics simulation. *J. Phys. Chem. B*, **2003**, *107(24)*, 5933-5947.

[126] Dixon, R.W.; Kollman, P.A. Advancing beyond the atom-centered model in additive and nonadditive molecular mechanics. *J. Comp. Chem.*, **1997**, *18(13)*, 1632-1646.

[127] Sorenson, J.M.; Hura, G.; Glaeser, R.M.; Head-Gordon, T. What can x-ray scattering tell us about the radial distribution functions of water? *J. Chem. Phys.*, **2000**, *113(20)*, 9149-9161.

[128] Soper, A.K. The radial distribution functions of water and ice from 220 to 673 K and at pressures up to 400 MPa. *Chem. Phys.*, **2000**, *258(2-3)*, 121-137.

[129] Jorgensen, W.L.; Chandrasekhar, J.; Madura, J.D.; Impey, R.W.; Klein, M.L. Comparison of Simple Potential Functions for Simulating Liquid Water. *J. Chem. Phys.*, **1983**, *79(2)*, 926-935.

[130] Guillot, B. A reappraisal of what we have learnt during three decades of computer simulations on water. *J. Mol. Liq.*, **2002**, *101(1-3)*, 219-260.

[131] Gubskaya, A.V.; Kusalik, P.G. The total molecular dipole moment for liquid water. *J. Chem. Phys.*, **2002**, *117(11)*, 5290-5302.

[132] Batista, E.R.; Xantheas, S.S.; Jonsson, H. Molecular multipole moments of water molecules in ice Ih. *J. Chem. Phys.*, **1998**, *109(11)*, 4546-4551.

[133] Silvestrelli, P.L.; Parrinello, M. Water molecule dipole in the gas and in the liquid phase. *Phys. Rev. Lett.*, **1999**, *82(16)*, 3308-3311.

[134] Batista, E.R.; Xantheas, S.S.; Jonsson, H. Multipole moments of water molecules in clusters and ice Ih from first principles calculations. *J. Chem. Phys.*, **1999**, *111(13)*, 6011-6015.

[135] Sprik, M. Hydrogen bonding and the static dielectric constant in liquid water. *J. Chem. Phys.*, **1991**, *95(9)*, 6762-6769.

[136] Soetens, J.C.; Costa, M.T.C.M.; Millot, C. Static dielectric constant of the polarizable NCC water model. *Mol. Phys.*, **1998**, *94(3)*, 577-579.

[137] Guissani, Y.; Guillot, B. A Ccomputer Simulation Study of the Liquid-vapor coexistence curve of water. *J. Chem. Phys*, **1993**, *98(10)*, 8221-8235.

[138] Kell, G.S. Precise representation of volume properties of water at 1 atmosphere *J. Chem. Eng. Data*, **1967**, *12(1)*, 66-69.

[139] Krynicki, K.; Green, C.D.; SAWYER, D.W. Pressure and temperature dependence of self-diffusion in water. *Faraday Discussions*, **1978**, *66*, 199-208.

[140] Kaatze, U. Complex permittivity of water as a function of frequency and temperature. *J. Chem. Eng. Data*, **1989**, *34(4)*, 371-374.

[141] Svishchev, I.M.; Kusalik, P.G.; Wang, J.; Boyd, R.J. Polarizable point-charge model for water: Results under normal and extreme conditions. *J. Chem. Phys.*, **1996**, *105(11)*, 4742-4750.

[142] Kusalik, P.G.; Svishchev, I.M. The spatial structure in liquid water. *Science*, **1994**, *265(5176)*, 1219-1221.

[143] Rick, S.W. Simulations of ice and liquid water over a range of temperatures using the fluctuating charge model. *J. Chem. Phys.*, **2001**, *114(5)*, 2276-2283.

[144] Mahoney, M.W.; Jorgensen, W.L. Diffusion constant of the TIP5P model of liquid water. *J. Chem. Phys.*, **2001**, *114(1)*, 363-366.

[145] Mahoney, M.W.; Jorgensen, W.L. A five-site model for liquid water and the reproduction of the density anomaly by rigid, nonpolarizable potential functions. *J. Chem. Phys.*, **2000**, *112(20)*, 8910-8922.

[146] Vega, C.; Abascal, J.L.F. Relation between the melting temperature and the temperature of maximum density for the most common models of water. *J. Chem. Phys.*, **2005**, *123(14)*, Art. No. 144504.

[147] Baez, L.A.; Clancy, P. Existence of a density maximum in extended simple point charge water. *J. Chem. Phys.*, **1994**, *101(11)*, 9837-9840.

[148] Jedlovszky, P.; Mezei, M.; Vallauri, R. A molecular level explanation of the density maximum of liquid water from computer simulations with a polarizable potential model. *Chem. Phys. Lett.*, **2000**, *318(1-3)*, 155-160.

[149] Ren, P.Y.; Ponder, J.W. Temperature and pressure dependence of the AMOEBA water model. *J. Phys. Chem. B*, **2004**, *108(35)*, 13427-13437.

[150] Kiyohara, K.; Gubbins, K.E.; Panagiotopoulos, A.Z. Phase coexistence properties of polarizable water models. *Mol. Phys.*, **1998**, *94(5)*, 803-808.

[151] Jedlovszky, P.; Vallauri, R. Liquid-vapor and liquid-liquid phase equilibria of the Brodholt-Sampoli-Vallauri polarizable water model. *J. Chem. Phys.*, **2005**, *122(8)*, Art. No. 081101.

[152] Brovchenko, I; Geiger, A; Oleinikova, A; Liquid-liquid phase transitions in supercooled water studied by computer simulations of various water models. *J. Chem. Phys.*, **2005**, *123(4)*, Art. No. 044515.

[153] Vega, C.; Abascal, J.L.F.; Nezbeda, I. Vapor-liquid equilibria from the triple point up to the critical point for the new generation of TIP4P-like models: TIP4P/Ew, TIP4P/2005, and TIP4P/ice. *J. Chem. Phys.*, **2006**, *125(3)*, Art. No. 034503.

[154] Panhuis, M.I.H.; Popelier, P.L.A.; Munn, R.W.; Angyan, J.G. Distributed polarizability of the water dimer: Field-induced charge transfer along the hydrogen bond. *J. Chem. Phys.*, **2001**, *114(18)*, 7951-7961.

[155] Morita, A. Water polarizability in condensed phase: Ab initio evaluation by cluster approach. *J. Comp. Chem.*, **2002**, *23(15)*, 1466-1471.

[156] Morita, A.; Kato, S. An ab initio analysis of medium perturbation on molecular polarizabilities. *J. Chem. Phys.*, **1999**, *110*, 11987.

[157] Kaminski, G.A.; Stern, H.A.; Berne, B.J.; Friesner, R.A. Development of an accurate and robust polarizable molecular mechanics force field from ab initio quantum chemistry. *J. Phys. Chem. A*, **2004**, *108(4)*, 621-627.

[158] Giese, T.J.; York, D.M. Many-body force field models based solely on pairwise Coulomb screening do not simultaneously reproduce correct gas-phase and condensed-phase polarizability limits. *J. Chem. Phys.*, **2004**, *120(21)*, 9903-9906.

[159] Jeon, J.; Lefohn, A.E.; Voth, G.A. An improved Polarflex water model. *J. Chem. Phys.*, **2003**, *118(16)*, 7504-7518.

[160] Paricaud, P.; Predota, M.; Chialvo, A.A.; Cummings, P.T. From dimer to condensed phases at extreme conditions: Accurate predictions of the properties of water by a Gaussian charge polarizable model. *J. Chem. Phys.*, **2005**, *122(24)*, Art. No. 244511.

[161] Alfredsson, M.; Brodholt, J.P.; Hermanson, K.; Vallauri, R. The use of a point polarizable dipole in intermolecular potentials for water. *Mol. Phys.*, **1998**, *94(5)*, 873-876.

[162] Perera, L.; Berkowitz, M.L. Many body effects in molecular dynamics simulations of NA+(H2O)N and CL-(H2O)N clusters. *J. Chem. Phys.*, **1991**, *95(3)*, 1954-1963.

[163] Perera, L.; Berkowitz, M.L. Structure and dynamics of CL-(H2O)20 clusters - The effect of the polarizability and the charge of the ion *J. Chem. Phys.*, **1992**, *96(11)*, 8288-8294.

[164] Perera, L.; Berkowitz, M.L. Stabilization energies of CL-, BR-, and I- ions in water clusters. *J. Chem. Phys.*, **1993**, *99(5)*, 4222-4224.3

[165] Perera, L.; Berkowitz, M.L. Structures of CL-(H2O)(N) and F-(H2O)(N) (N=2,3,...,15) clusters - Molecular dynamics computer simulations. *J. Chem. Phys.*, **1994**, *100(4)*, 3085-3093.

[166] Grossfield, A. Dependence of ion hydration on the sign of the ion's charge. *J. Chem. Phys.*, **2005**, *122(2)*, Art. No. 024506.

[167] Herce, D.H.; Perera, L.; Darden, T.A.; Sagui, C. Surface solvation for an ion in a water cluster. *J. Chem. Phys.*, **2005**, *122(2)*, Art. No. 024513.

[168] Marrink, S.J.; Berendsen, H.J.C. Simulation of water transport through a lipid-membrane. *J. Phys. Chem.*, **1994**, *98(15)*, 4155-4168.

[169] Dang, L.X. Computational study of ion binding to the liquid interface of water. *J. Phys. Chem. B*, **2002**, *106(40)*, 10388-10394.

[170] Dang, L.X.; Chang, T.M.; Panagiotopoulos, A.Z. Gibbs ensemble Monte Carlo simulations of coexistence properties of a polarizable potential model of water. *J. Chem. Phys.*, **2002**, *117(7)*, 3522-3523.

[171] Dang, L.X.; Chang, T.M. Molecular mechanism of ion binding to the liquid/vapor interface of water. *J. Phys. Chem. B*, **2002**, *106(2)*, 235-238.

[172] Dang, L.X.; Chang, T.M. Many-body interactions in liquid methanol and its liquid/vapor interface: A molecular dynamics study. *J. Chem. Phys.*, **2003**, *119(18)*, 9851-9857.

[173] Paul, S.; Chandra, A. Dynamics of water molecules at liquid-vapour interfaces of aqueous ionic solutions: effects of ion concentration. *Chem. Phys. Lett.*, **2003**, *373(1-2)*, 87-93.

[174] Liu, P.; Harder, E.; Berne, B.J. On the calculation of diffusion coefficients in confined fluids and interfaces with an application to the liquid-vapor interface of water. *J. Phys. Chem. B*, **2004**, *108(21)*, 6595-6602.

[175] Wick, C.D.; Dang, L.X. Investigating pressure effects on structural and dynamical properties of liquid methanol with many-body interactions. *J. Chem. Phys.*, **2005**, *123(18)*, Art. No. 184503.

[176] Jungwirth, P.; Tobias, D.J. Molecular structure of salt solutions: A new view of the interface with implications for heterogeneous atmospheric chemistry. *J. Phys. Chem. B*, **2001**, *105(43)*, 10468-10472.

[177] Jungwirth, P.; Tobias, D.J. Ions at the air/water interface. *J. Phys. Chem B*, **2002**, *106(25)*, 6361-6373.

[178] Vrbka, L.; Mucha, M.; Minofar, B.; Jungwirth, P.; Brown, E.C.; Tobias, D.J. Propensity of soft ions for the air/water interface. *Current opinion in colloid & interface science*, **2004**, *9(1-2)*, 67-73.

[179] Brown, E.C.; Mucha, M.; Jungwirth, P.; Tobias, D.J. Structure and vibrational spectroscopy of salt water/air interfaces: Predictions from classical molecular dynamics simulations. *J. Phys. Chem. B*, **2005**, *109(16)*, 7934-7940.

[180] Gopalakrishnan, S.; Jungwirth, P.; Tobias, D.J.; Allen, H.C. Air-liquid interfaces of aqueous solutions containing ammonium and sulfate: Spectroscopic and molecular dynamics studies. *J. Phys. Chem. B*, **2005**, *109(18)*, 8861-8872.

[181] Jungwirth, P.; Tobias, D.J. Surface effects on aqueous ionic solvation: A molecular dynamics simulation study of NaCl at the air/water interface from infinite dilution to saturation. *J. Phys. Chem. B*, **2000**, *104(32)*, 7702-7706.

[182] Salvador, P.; Curtis, J.E.; Tobias, D.J.; Jungwirth, P. Polarizability of the nitrate anion and its solvation at the air/water interface. *Phys. Chem. Chem. Phys.*, **2003**, *5(17)*, 3752-3757.

[183] Tuma, L.; Jenicek, D.; Jungwirth, P. Propensity of heavier halides for the water/vapor interface revisited using the Amoeba force field. *Chem. Phys. Lett.*, **2005**, *411(1-3)*, 70-74.

[184] Bastug, T.; Kuyucak, S. Test of molecular dynamics force fields in gramicidin A. *European Biophysics Journal with Biophysics Letters*, **2005**, *34(5)*, 377-382.

[185] Allen, T.W.; Andersen, O.S.; Roux, B. Energetics of ion conduction through the gramicidin channel. *Proc. Natl. Acad. Sci. U.S.A.*, **2004**, *101(1)*, 117-122.

[186] Allen, T.W.; Bastug, T.; Kuyucak, S.; Chung, S.H. Gramicidin A channel as a test ground for molecular dynamics force fields. *Biophys. J.*, **2003**, *84(4)*, 2159-2168.

[187] Edwards, S.; Corry, B.; Kuyucak, S.; Chung, S.H. Continuum electrostatics fails to describe ion permeation in the gramicidin channel. *Biophys. J.*, **2002**, *83(3)*, 1348-1360.

[188] Duca, K.A.; Jordan, P.C. Comparison of selectively polarizable force fields for ion-water-peptide interactions: Ion translocation in a gramicidin-like channel. *J. Phys. Chem. B*, **1998**, *102(45)*, 9127-9138.

[189] Gao, J.L.; Habibollazadeh, D.; Shao, L. A Polarizable Intermolecular Potential Function for Simulation of Liquid Alcohols. *J. Phys. Chem.*, **1995**, *99(44)*, 16460-16467.

[190] Patel, S.; Brooks, C.L. A nonadditive methanol force field: Bulk liquid and liquid-vapor interfacial properties via molecular dynamics simulations using a fluctuating charge model. *J. Chem. Phys.*, **2005**, *122(2)*, Art. No. 024508.

[191] Gonzalez, M.A.; Enciso, E.; Bermejo, F.J.; Bee, M. Ethanol force fields: A molecular dynamics study of polarization effects on different phases. *J. Chem. Phys.*, **1999**, *110(16)*, 8045-8059.

[192] Kaminski, G.A. Accurate prediction of absolute acidity constants in water with a polarizable force field: Substituted phenols, methanol, and imidazole. *J. Phys. Chem. B*, **2005**, *109(12)*, 5884-5890.

[193] Noskov, S.Y.; Lamoureux, G.; Roux, B. Molecular dynamics study of hydration in ethanol-water mixtures using a polarizable force field. *J. Phys. Chem. B*, **2005**, *109(14)*, 6705-6713.

[194] Iuchi, S.; Morita, A.; Kato, S. Molecular dynamics simulation with the charge response kernel: Vibrational spectra of liquid water and N-methylacetamide in aqueous solution. J. Phys. Chem. B, **2002**, *106(13)*, 3466-3476.

[195] Kaminski, G.A.; Friesner, R.A.; Zhou, R.H. A computationally inexpensive modification of the point dipole electrostatic polarization model for molecular simulations. *J. Comp. Chem.*, **2003**, *24(3)*, 267-276.

[196] Mannfors, B.; Mirkin, N.G.; Palmo, K.; Krimm, S. A polarizable electrostatic model of the N-methylacetamide dimer. *J. Comp. Chem.*, **2001**, *22(16)*, 1933-1943.

[197] Kaminski, G.A.; Stern, H.A.; Berne, B.J.; Friesner, R.A.; Cao, Y.X.X.; Murphy, R.B.; Zhou, R.H.; Halgren, T.A. Development of a polarizable force field for proteins via ab initio quantum chemistry: First generation model and gas phase tests. *J. Comput. Chem.*, **2002**, *23(16)*, 1515-1531.

[198] Ren, P.Y.; Ponder, J.W. Consistent treatment of inter- and intramolecular polarization in molecular mechanics calculations. *J. Comput. Chem.*, **2002**, *23(16)*, 1497-1506.

[199] Mark, P.; Nilsson, L. Structure and dynamics of liquid water with different long-range interaction truncation and temperature control methods in molecular dynamics simulations. *J. Comp. Chem.*, **2002**, *23(13)*, 1211-1219.

[200] van der Spoel, D.; van Maaren, P.J.; Berendsen, H.J.C. A systematic study of water models for molecular simulation: Derivation of water models optimized for use with a reaction field. *J. Chem. Phys.*, **1998**, *108(24)*, 10220-10230.

[201] Feller, S.E.; Pastor, R.W.; Rojnuckarin, A.; Bogusz, S.; Brooks, B.R. Effect of electrostatic force truncation on interfacial and transport properties of water. *J. Phys. Chem*, **1996**, *100(42)*, 17011-17020.

[202] Roberts, J.E.; Schnitker, J. Boundary-conditions in Simulations of Aqueous Ionic-Solutions - A Systematic Study. *J. Phys. Chem.*, **1995**, *99(4)*, 1322-1331.

[203] Schreiber, H.; Steinhauser, O. Molecular Dynamics Studies of Solvated Polypeptides - Why the Cutoff Scheme does not work. Chem. Phys., **1992**, *168(1)*, 75-89.

[204] Saito, M. Molecular Dynamics Simulations of Proteins in solution - Artifacts caused by the cutoff approximation. *J. Chem. Phys.*, **1994**, *101(5)*, 4055-4061.

[205] Norberg, J.; Nilsson, L. On the truncation of long-range electrostatic interactions in DNA. *Biophys. J.*, **2000**, *79(3)*, 1537-1553.

[206] Ewald, P.P. Die Berechnung optischer und elektrostatischer Gitterpotentiale. *Ann. Phys.*, **1921**, *64*, 253-287.

[207] Koehl, P. Electrostatics calculations: latest methodological advances. *Curr. Opin. Struct. Biol.*, **2006**, *16(2)*, 142-151.

[208] Frenkel, D.; Smit, B. *Understanding Molecular Simulation*, Academic Press: San Diego, 2001.

[209] Deserno, M.; Holm, C. How to mesh up Ewald sums. I. A theoretical and numerical comparison of various particle mesh routines. *Journal of Chemical Physics*, **1998**, *109(18)*, 7678-7693.

[210] Deserno, M.; Holm, C. How to mesh up Ewald sums. II. An accurate error estimate for the particle-particle-particle-mesh algorithm. *Journal of Chemical Physics*, **1998**, *109(18)*, 7694-7701.

[211] Hockney, R.W.; Eastwood, J.W. *Computer Simulation using Particles*, Institute of Physics Publishing: Bristol, UK, 1988.

[212] Darden, T., York, D., Pedersen, L. Particle Mesh Ewald - An N.Log(N) Method for Ewald Sums in Large Systems. *J. Chem. Phys.*, **1993**, *98(12)*, 10089-10092.

[213] Toukmaji, A.; Sagui, C.; Board, J.; Darden, T., Efficient particle-mesh Ewald based approach to fixed and induced dipolar interactions. *J. Chem. Phys.*, **2000**, *113*, 10913.

[214] Appel, A. An efficient program for many-body simulation. *SIAM J. Sci. Stat. Comput.*, **1985**, *6(1)*, 85-103.

[215] Barnes, J.; HUT, P. A hierarchical O(N-LOG-N) force calculation algorithm. *Nature*, **1986**, *324(6096)*, 446-449.

[216] Greengard, L.; Rokhlin, V. A fast algorithm for particle simulations. *J. Comp. Phys.*, **1987**, *73*, 325-348

[217] Greengard, L. Fast algorithms for classical physics. *Science*, **1994**, *265(5174)*, 909-914.

[218] Ding, H.-Q.; Karasawa, N.; Goddard, W. A. Atomic level simulations on a million particles: The cell multipole method for Coulomb and London nonbond interactions. *J. Chem. Phys.*, **1992**, *97*, 4309-4315.

[219] Kutteh, R.; Nicholas, J. B. Implementing the cell multipole method for dipolar and charged dipolar systems. *Comput. Phys. Commun.*, **1995**, *86*, 236-254.

[220] Ding, H.-Q.; Karasawa, N.; Goddard, W. A. The reduced cell multipole method for Coulomb interactions in periodic systems with million-atom unit cells. *Chem. Phys. Lett.*, **1992**, *196*, 6-13.

[221] Lim, K.-T.; Brunett, S.; Iotov, M.; McClurg, R.B.; Vaidehi, N.; Dasgupta, S.; Taylor, S.; Goddard, W. A. Molecular Dynamics for Very Large Systems on Massively Parallel Computers: The MPSim Program. *J. Comput. Chem.*, **1997**, *18*, 501-521.

[222] Pollock, E.L.; Glosli, J. Comments on P(3)M, FMM, and the Ewald method for large periodic coulombic systems *Comp. Phys. Comm.*, **1996**, *95(2-3)*, 93-110.

[223] Figueirido, F.; Levy, R.M.; Zhou, R.H.; Berne, B.J. Large scale simulation of macromolecules in solution: Combining the periodic fast multipole method with multiple time step integrators. *J. Chem. Phys.*, **1997**, *106(23)*, 9835-9849.

[224] Sagui, C.; Darden, T.A. Molecular dynamics simulations of biomolecules: Long-range electrostatic effects, *Annual Review of Biophysics and Biomolecular Structure*, **1999**, *28*, 155-179.

[225] Ennari, J.; Neelov, I.; Sundholm, F. Simulation of a PEO based solid polyelectrolyte, comparison of the CMM and the Ewald summation method. *Polymer*, **2000**, *41(6)*, 2149-2155.

[226] Carrier, J.; Greengard, L.; Rokhlin, V. A fast adaptive multipole algorithm for particle simulations. *SIAM J. Sci. Stat. Comput.*, **1988**, *9(4)*, 669-686.

[227] Esselink, K. A comparison of algorithms for long range interactions. *Comp. Phys. Comm.*, **1995**, *87(3)*, 375-395.

[228] Fenley, M.O.; Olson, W.K.; Chua, K.; Boschitsch, A.H. Fast adaptive multipole method for computation of electrostatic energy in simulations of polyelectrolyte DNA. *J. Comp. Chem.*, **1996**, *17(8)*, 976-991.

[229] Sun, X.B.; Pitsianis, N.P. A matrix version of the fast multipole method. *SIAM Review*, **2001**, *43(2)*, 289-300.

[230] Kutteh, R.; Nicholas, J.B. Efficient dipole iteration in polarizable charged systems using the cell multipole method and application to polarizable water. *Comp. Phys. Comm.*, **1995**, *86(3)*, 227-235.

[231] Saad, Y.; van der Vorst, H.A. Iterative solution of linear systems in the 20-th century *J. Comput. Appl. Math.* , **2000**, *123*, 1-33.

[232] Barrett, R.; Berry, M.; Chan, T.F.; Demmel, J.; Donato,J.M.; Dongarra, J.; Eijkhout, V.; Pozo, R.; Romine, C.; van der Vorst,H.A. Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, SIAM: Philadalphia, 1994.

[233] Sewell, G. *Computational methods of linear algebra,* Wiley-Interscience: Hoboken, 2005.

[234] Golub,G.H.; Van Loan, C.F. *Matrix Computations,* The Johns Hopkins University Press, 3rd edition, 1996.

[235] Golub, G.H.; Ortega, J.M. *Scientific Computing And Differential Equations : An Introduction To Numerical Methods,* Academic Press:Boston, 1992.

[236] Saad, Y. *Iterative Methods for Sparse Linear Systems,* SIAM, Second edition, 2003.

[237] Kumar, V.; Grama, A.; Gupta, A.; Karypis, G. *Introduction to Parallel Computing: Design and Analysis of Algorithms,* Benjamin/Cumming Publishing Company: Redwood City, 1994.

[238] Axelsson, O. *Iterative Solution Methods,* Cambridge University Press: New York, 1994.

[239] Hackbusch, W. *Iterative solution of large sparse systems of equations,* Springer Verlag: New York, 1994.

[240] Greenbaum, A. *Iterative Methods for Solving Linear Systems,* SIAM: Philedelphia, 1997.

[241] Fedorenko, R.P. A relaxation method for solving elliptic difference equations. *USSR Comput. Math. math. Phys.*, **1961**, *1*, 1092-1096.

[242] Fedorenko, R.P. The speed of convergence of one iterative process. *USSR Comput. Math. math. Phys.*, **1965**, *4*, 227-235.

[243] Brandt, A. Multi-Level Adaptive Solutions to Boundary-Value Problems. *Math. Comp.*, **1977**, *31*, 333-390.

[244] Beck, T.L. Real-space multigrid solution of electrostatics problems and the Kohn-Sham equations. *Int. J. Quant. Chem.*, **1997**, *65(5)*, 477-486.

[245] Heiskanen, M.; Torsti, T.; Puska, M.J.; Nieminen, R.M. Multigrid method for electronic structure calculations. *Phys. Rev. B* , **2001**, *63(24)*, Art. No. 245106.

[246] Torsti, T.; Heiskanen, M.; Puska, M.J.; Nieminen, R.M. MIKA: Multigrid-based program package for electronic structure calculations. *Int. J. Quant. Chem.*, **2003**, *91(2)*, 171-176.

[247] Kummel, S. Damped gradient iteration and multigrid relaxation: tools for electronic structure calculations using orbital density-functionals. *J. Comp. Phys.*, **2004**, *201(1)*, 333-343.

[248] Livshits, I.; Brandt, A. Accuracy properties of the wave-ray multigrid algorithm for Helmholtz equations. *SIAM J. Sci. Comp.*, **2006** , *28 (4)* , 1228-1251 .

[249] Weiss, B.; Biro, O.; Caldera, P.; Hollaus, K.; Paoli, G.; Preis, K. A multigrid solver for time harmonic three-dimensional electromagnetic wave problems. *IEEE Trans. Mag.*, **2006** , *42 (4)* , 639-642 .

[250] Mavriplis, D.J. Exploring alternative approaches to computational fluid dynamics. *International Journal of Computational Fluid Dynamics*, **2005**, *19 (8)* , 613-620.

[251] Caughey, D.A. Symmetric Gauss-Seidel multigrid solution of the Euler equations on structured and unstructured grids. *International Journal of Computational Fluid Dynamics*, **2005** , *19 (8)* , 605-612.

[252] Hafez, M.; Wahba, E. Multigrid acceleration for transonic aerodynamic flow simulations based on a hierarchical formulation. *International Journal for Numerical methods in fluids*, **2005** , *47 (6-7)* , 517-541.

[253] Dolean, V.; Lanteri, S. Parallel multigrid methods for the calculation of unsteady flows on unstructured grids: algorithmic aspects and parallel performances on clusters of PCs. *Parallel Computing*, **2004**, *30(4)*, 503-525.

[254] Izaguirre, J.A.; Hampton, S.S.; Matthey, T. Parallel multigrid summation for the N-body problem. *Journal of Parallel and Distributed Computing*, **2005**, *65(8)*, 949-962.

[255] Demmel, J.A. *Applied Numerical Linear Algebra,* Society for Industrial and Applied Mathematics: Philadelphia, 1997.

[256] Press, W.H.; Teukolsky, S.A.; Vetterling, W.T.; Flannery, B.P. *Numerical Recipes in C++,* Cambridge University Press: Cambridge, 2002.

[257] Trottenberg, U.; Oosterlee, C.; Schüller, A. *Multigrid,* Academic Press: San Diego, 2001.

[258] Briggs, W.L.; Henson, V.E.; McCormick, S.F. *A Multigrid Tutorial,* Society for Industrial and Applied Mathematics: Philadelphia, 2000.

[259] Wesseling., P. *An Introduction to Multigrid Methods,* John Wiley & Sons: Chichester, 1992.

[260] Lay, D.C. *Linear Algebra and Its Applications,* Pearson Addison Wesley: Reading, 1998.

[261] Yavneh, I. Why Multigrid Methods Are So Efficient. *Computing in Science & Engineering*, textbf2006, *8(6)*, 12-22.

[262] Brandt,A.; McCormick,S.F.; Ruge, J.W. Algebraic Multigrid(AMG) for Automatic Multigrid Solutions with Application to Geodetic Computations, Report, Inst. For Computational studies, fort Collins, CO, 1982.

[263] Ruge, W.; Stüben, K. Algebraic multigrid, in Multigrid Methods, Frontiers Appl. Math. 3, S.F. McCormick, ed. SIAM, Philedelphia, 1987, pp.73-130.

[264] Stüben, K. Algebraic multigrid (AMG): An introduction with applications, in Multigrid, U Trottenberg, C. Ooosterlee, and A. Schuller, eds, Academic Press, San Diego, 2000.

[265] Briggs, W.L.; Henson. V.E.; McCormick, S.F. *A Multigrid Tutorial*, 2nd ed., SIAM, Philedelphia, 2000.

[266] Okusanya, T.; Darmofal, D.L.; Peraire, J. Algebraic multigrid for stabilized finite element discretizations of the Navier-Stokes equations. *Computer methods in applied mechanics and engineering*, **2004**, *193(33-35)*, 3667-3686.

[267] Cleary, A.J.; Falgout, R.D.; Henson, V.E.; Jones, J.E.; Manteufeel, T.A.; Robustness and scalability of algebraic multigrid. *SIAM J. Sci. Comput.*, **2000**, *21*, 1886-1908.

[268] Steinmetz, T.; Helias, M.; Wimmer, G.; Fichte, L.O.; Clemens, M. Electroquasistatic field simulations based on a discrete electromagnetism formulation. *IEEE Trans. Mag.*, **2006**, *42(4)*, 755-758.

[269] Liu, Y.Q.; Yuan, J.S. A finite element domain decomposition combined with algebraic multigrid method for large-scale electromagnetic field computation. *IEEE Trans. Mag.*, **2006**, *42(4)*, 655-658.

[270] Arbenz, P.; Hetmaniuk, U.L.; Lehoucq, R.B.; Tuminaro, R.S. A comparison of eigensolvers for large-scale 3D modal analysis using AMG-preconditioned iterative methods. *International Journal for Numerical Methods in Engineering*, **2005**, *64(2)*, 204-236.

[271] Borzi, A.; Borzi, G. Algebraic multigrid methods for solving generalized eigenvalue problems. *International Journal for Numerical Methods in Engineering*, **2006**, *65 (8)*: 1186-1196 FEB 19 2006

[272] Krechel, A.; Stüben, K. Parallel algebraic multigrid based on subdomain blocking. *Parallel Computing*, **2001**, *27*, 1009-1031.

[273] Cleary, A.; Falgout, R.; Henson, V.; Jones, J. Coarse-grid selection for parallel algebraic multigrid. *Proceedsing of the Fifth International Symposium on Solving Irregularly Structured Problems in Parallel*, Springer-Verlag Lecture Notes in Computer Science, New York, August 1998.

[274] Henson, V.E.; Yang, U.M. newblock BoomerAMG: A parallel algebraic multigrid solver and preconditioner. *Appl. Numer. Math.*, **2002**, *41*, 155-177.

[275] Yang, U.M. Parallel Algebraic Multigrid Methods - High Performance Preconditioners, Numerical Solution of Partial Differential Equations on Parallel Computers, LNCS 51, A.M. Bruaset and A. Tveito, eds., Springer-Verlag, (2006), pp. 209-233.

[276] Falgout, R.D. An introduction to algebraic multigrid. *Computing in Science & Engineering*, **2006**, *8(6)*, 24-33.

[277] York, D.M.; Lee, T.-S.; Yang, W. Quantum mechanical study of aqueous polarization effects on biological macromolecules. *J. Am. Chem. Soc.*, **1996**, *118*, 10940-10941.

[278] van der Vaart, A.; Bursulaya, B.D.; Brooks, C.L.; Merz, K.M. Are many-body effects important in protein folding? *J. Phys. Chem. B*, **2000**, *104(40)*, 9554-9563.

[279] Soto, P.; Mark, A.E. The Effect of the Neglect of Electronic Polarization in Peptide Folding Simulations. *J. Phys. Chem B*, **2002**, *106*, 12830-12833.

[280] Ciegis, R. Analysis of parallel preconditioned conjugate gradient algorithms. *Informatica*, **2005**, *16(3)*, 317-332.

[281] Kurzak, J.; Pettitt, B.M. Massively parallel implementation of a fast multipole method for distributed memory machines. *Journal of Parallel and Distributed Computing*, **2005**, *65(7)*, 870-881.

[282] Wu, F.; Zhang, Y.J.; Oo, Z.Z.; Li, E.P. Parallel multilevel fast multipole method for solving large-scale problems. *IEEE Antennas and Propagation Magazine*, **2005**, *47(4)*, 110-118.

[283] Saad, Y. *SPARSKIT:A basic tool kit for sparse matrix computation.* Tech. Rep. CSRD TR 1029, CSRD, University of Illinois, Urbana, IL, 1990.

[284] Eijkhout, V. *LAPACK working not 50: Distributed sparse data structures for linear algebra operations.* Tech. Rep. CS 92-169, Computer Science Department, University of Tennessee, Knoxville, TN, 1992.

[285] Grigull, U.; Straub, J.; Schiebener, P. *Steam Tables in SI Units* , Springer:Berlin, 1984.

[286] Fanourgakis, G.S.; Xantheas, S.S. The flexible, polarizable, thole-type interaction potential for water (TTM2-F) revisited. *J. Phys. Chem. A*, **2006**, *110(11)*, 4100-4106.

[287] Abascal, J.L.F.; Vega, C. A general purpose model for the condensed phases of water: TIP4P/2005. *J. Chem. Phys.*, **2005**, *123(23)*, Art. No. 234505.

[288] Dinh, T.-L.; Huber, G.A. Development of an Algorithm to Compute Polarizable Interactions using Multigrid and Multipole Methods, In: *Proc. International Conference on Mathematics and Engineering Techniques in Medicine and Biological Sciences*, 2003, 329-335.

[289] Dinh, T.-L.; Huber, G.A. An Efficient Algorithm for Polarizable Interactions: A Uniformly Distributed One-Dimensional Case. *Journal of Mathematical Modelling and Algorithms*, **2005**, *4*, 111-128.