

11-20-2014

High Dimensional Non-Linear Optimization of Molecular Models

Joseph C. Fogarty

University of South Florida, Jcfogart@mail.usf.edu

Follow this and additional works at: <https://digitalcommons.usf.edu/etd>

 Part of the [Physics Commons](#)

Scholar Commons Citation

Fogarty, Joseph C., "High Dimensional Non-Linear Optimization of Molecular Models" (2014). *USF Tampa Graduate Theses and Dissertations*.

<https://digitalcommons.usf.edu/etd/5618>

This Dissertation is brought to you for free and open access by the USF Graduate Theses and Dissertations at Digital Commons @ University of South Florida. It has been accepted for inclusion in USF Tampa Graduate Theses and Dissertations by an authorized administrator of Digital Commons @ University of South Florida. For more information, please contact digitalcommons@usf.edu.

High Dimensional Non-Linear Optimization of Molecular Models

by

Joseph C. Fogarty

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy in Applied Physics
Department of Physics
College of Arts and Sciences
University of South Florida

Major Professor: Sagar A. Pandit, Ph.D.
Ananth Y. Grama, Ph.D.
Martin Muschol, Ph.D.
Jianjun Pan, Ph.D.
Lilia M. Woods, Ph.D.

Date of Approval:
November 20, 2014

Keywords: Optimization, Simulation, Molecular Modeling

Copyright © 2014, Joseph C. Fogarty

Dedication

For my wife Erin whose patience and love have made everything possible.

For my son Benjamin, for his excitement, his curiosity, and his sweetness.

For my new son Theodore, whom I can't wait to get to know.

For my family, for all of their support.

Acknowledgments

This dissertation would not have been possible without the invaluable guidance given to me by my advisor, Dr. Sagar Pandit. In instances when I would have lost confidence in my approaches, Dr. Pandit would encourage me to stay the course, and not to abandon work due to initial challenges. Dr. Pandit's wise counsel and advice helped to keep the work presented here focused and productive. I also thank my fellow students in Dr. Pandit's lab especially C. Brad Bennett, James Kruczek, James Lyon, and Peter Kirby for encouragement, support, and camaraderie during my years of research.

I also thank my graduate committee for valuable insight, especially during the candidacy procedure. The frank evaluation of my dissertation proposal helped to chart the course for this work. I especially thank Dr. Ananth Grama for both guidance and financial support early in my graduate career, without which I would not have been able to produce the current work. I would like to acknowledge Dr. David Rabson, Dr. Sameer Varma, and Dr. David Rogers for valuable insight and feedback during group meetings. I thank Dr. Eric Jakobsson and Dr. See-Wing Chiu for fruitful collaborations. I thank Dr. Adri van Duin for providing data and code examples. I thank Dr. Bo Zeng for suggestions of optimization algorithms. I thank Dr. Ole Edholm, Dr. John Nagle, Dr. John Katsaras, Dr. Larry Scott and others for suggestions made during visits.

This work would also not have been possible without support from the Department of Physics faculty, staff and students. I thank the Fred L. and Helen M. Tharp Endowed Scholarship Fund for providing two summers of support.

Contents

| | |
|---|-----------|
| Abstract | ix |
| 1 Introduction | 1 |
| 1.1 Molecular Models | 1 |
| 1.1.1 Increasing model complexity | 2 |
| 1.1.2 Water | 5 |
| 1.2 Optimization | 6 |
| 1.2.1 Gradient Based Methods | 8 |
| 1.2.2 Direct Search Methods | 9 |
| 1.3 Dissertation Outline | 10 |
| 2 ParOpt | 13 |
| 2.1 A Note to the Reader | 13 |
| 2.2 Introduction | 13 |
| 2.3 Parameter Space Definition | 14 |
| 2.3.1 Parameter Constraints | 15 |
| 2.4 Target Function Definition | 16 |
| 2.5 Optimizers | 17 |
| 2.5.1 Steepest Descent | 17 |
| 2.5.2 Particle Swarm | 17 |
| 2.5.3 Nelder-Mead | 18 |
| 3 Coarse-Grained Water/Ion | 22 |
| 3.1 A Note to the Reader | 22 |
| 3.2 Introduction | 22 |
| 3.3 Model | 26 |
| 3.4 Target Data | 29 |
| 3.4.1 Water–Water | 29 |
| 3.4.2 Water–Ion | 31 |
| 3.5 Optimization | 34 |
| 3.5.1 Water–Water | 34 |
| 3.5.2 Water–Ion | 37 |
| 3.6 Validation | 39 |
| 3.6.1 Water–Water | 39 |
| 3.6.2 Water–Ion | 41 |
| 3.7 Conclusion | 46 |

| | | |
|----------|--|------------|
| 4 | Lipid Bilayer Structure Determination | 48 |
| 4.1 | A Note to the Reader | 48 |
| 4.2 | Introduction | 48 |
| 4.3 | Model and Methods | 50 |
| 4.3.1 | The Model (m) | 51 |
| 4.3.2 | The Data (d) | 54 |
| 4.3.3 | The Transformation (G) and the Comparison ($\ d' - d\ $) | 54 |
| 4.3.4 | The Regularization | 55 |
| 4.4 | Validation using Molecular Dynamics | 57 |
| 4.4.1 | Parameter Optimization | 60 |
| 4.5 | Results | 61 |
| 4.5.1 | Structural Properties | 68 |
| 4.6 | Conclusion | 75 |
| 5 | Reactive Water | 76 |
| 5.1 | Introduction | 76 |
| 5.2 | Model | 77 |
| 5.2.1 | Bond Orders | 78 |
| 5.2.2 | Bond Energy | 80 |
| 5.2.3 | Lone Pair Energy | 80 |
| 5.2.4 | C ₂ Correction | 81 |
| 5.2.5 | Triple Bond Stabilization | 81 |
| 5.2.6 | Overcoordination Energy | 81 |
| 5.2.7 | Undercoordination Energy | 81 |
| 5.2.8 | Valence Angle Energy | 82 |
| 5.2.9 | Penalty Energy | 82 |
| 5.2.10 | Three-body Conjugation Energy | 83 |
| 5.2.11 | Torsion Angle Energy | 83 |
| 5.2.12 | Four Body Conjugation Energy | 83 |
| 5.2.13 | Hydrogen Bond Energy | 84 |
| 5.2.14 | van der Waals Energy | 84 |
| 5.2.15 | Coulomb | 85 |
| 5.2.16 | QEq | 85 |
| 5.2.17 | Combination Rules | 85 |
| 5.3 | Parameters | 85 |
| 5.4 | Target Data | 91 |
| 5.4.1 | Partial Atomic Charges | 92 |
| 5.4.2 | <i>ab initio</i> Energies | 92 |
| 5.4.3 | Empirical Data | 92 |
| 5.5 | Optimization | 94 |
| 5.6 | Conclusion | 97 |
| 6 | Conclusion | 98 |
| | Bibliography | 101 |

| | | |
|----------|----------------------------------|------------|
| A | ParOpt Manual | 113 |
| A.1 | Introduction | 113 |
| A.2 | Usage | 113 |
| A.3 | Input | 113 |
| A.3.1 | Global | 115 |
| A.3.2 | Constant Parameters | 120 |
| A.3.3 | Variable Parameters | 121 |
| A.3.4 | Data | 121 |
| A.3.5 | Constraints | 122 |
| A.4 | Addressing | 123 |
| A.4.1 | Reax | 123 |
| A.4.2 | GROMACS | 124 |
| A.4.3 | Generic | 124 |
| A.5 | Scattering | 124 |
| A.5.1 | Scattering Data Lines | 124 |
| A.5.2 | Scattering Field Files | 126 |
| A.6 | Output | 126 |
| A.7 | Additional Tools | 127 |
| A.7.1 | Comparison Tool | 127 |
| A.7.2 | Forcefield Extraction | 127 |
| A.7.3 | Parameter Extraction | 128 |
| A.7.4 | Reax Profiles | 128 |
| A.7.5 | Luzzati Thickness | 129 |
| A.7.6 | Partial Volume | 129 |
| A.7.7 | Scattering | 129 |
| A.7.8 | Scattering Components | 129 |
| A.8 | Getting Help | 129 |
| B | Silica-Water Interface | 130 |
| B.1 | A Note to the Reader | 130 |
| B.2 | Introduction | 130 |
| B.3 | Methods | 132 |
| B.3.1 | System preparation | 133 |
| B.3.2 | Force field parameters | 136 |
| B.4 | Results | 137 |
| B.4.1 | Validation of models | 137 |
| B.4.2 | Silica–water interface | 141 |
| B.5 | Conclusion | 148 |
| C | Publications | 150 |
| D | Copyright Permissions | 151 |

List of Tables

| | | |
|-----|--|-----|
| 3.1 | Coarse-grained water optimal parameters. | 37 |
| 3.2 | Coarse-grained water target data results. | 38 |
| 3.3 | Water-ion interaction parameter ranges. | 38 |
| 3.4 | Water-ion optimal parameters. | 39 |
| 3.5 | Coarse-grained water ion target data results. | 39 |
| 3.6 | Coarse-grained water validation data. | 41 |
| 4.1 | Weights for calculation of electron and neutron scattering length densities. | 53 |
| 4.2 | Constraints on partial atomic volumes. | 55 |
| 4.3 | Lipid structural properties determined from ADP model fit averages. | 68 |
| 5.1 | ReaxFF global parameters | 86 |
| 5.2 | ReaxFF single body parameters | 87 |
| 5.3 | ReaxFF bond parameters | 88 |
| 5.4 | ReaxFF off-diagonal parameters | 88 |
| 5.5 | ReaxFF three body parameters | 89 |
| 5.6 | ReaxFF four body parameters | 90 |
| 5.7 | ReaxFF hydrogen bond parameters | 91 |
| B.1 | Free energy differences for force field validation. | 137 |
| B.2 | Water Data | 138 |
| B.3 | Silica Data. | 141 |

| | |
|--|-----|
| B.4 Silica-water Interface Data. | 144 |
|--|-----|

List of Figures

| | | |
|------|--|----|
| 1.1 | Comparison of interaction potential functions of increasing complexity | 5 |
| 2.1 | Schematic illustration of Nelder-Mead simplex transformations | 20 |
| 3.1 | Illustration of the MMCG water model. | 28 |
| 3.2 | MSD Scaling for SPC/E Water. | 30 |
| 3.3 | Coarse-grained ion solvation free energy pathway. | 32 |
| 3.4 | Coarse-grained water optimization results. | 35 |
| 3.5 | Water optimization parameter trajectory. | 36 |
| 3.6 | Water optimization observed data value trajectory. | 37 |
| 3.7 | Water-Ion optimization results. | 38 |
| 3.8 | Density as a function of NaCl concentration | 42 |
| 3.9 | Na-Na coordination and radial distribution functions | 43 |
| 3.10 | Cl-Cl coordination and radial distribution functions | 43 |
| 3.11 | Na-Cl coordination and radial distribution functions | 44 |
| 3.12 | Atomistic surface ion concentration profile | 45 |
| 3.13 | Coarse-grained surface ion concentration profile | 45 |
| 3.14 | Water radial distribution functions. | 46 |
| 4.1 | Form factors determined for MD and ADP model fit. | 58 |
| 4.2 | Electron and neutron scattering length densities for MD and ADP model fit. | 59 |
| 4.3 | Number densities for atomic groups from MD and ADP model fit. | 60 |

| | | |
|------|---|-----|
| 4.4 | Volume probabilities, number densities, and structural properties for DOPC. | 63 |
| 4.5 | Volume probabilities, number densities and structural properties for POPC. | 64 |
| 4.6 | Volume probabilities, number densities, and structural properties for DPPC. | 65 |
| 4.7 | Volume probabilities, number densities, and structural properties for POPG. | 66 |
| 4.8 | Volume probabilities, number densities, and structural properties for POPS. | 67 |
| 4.9 | Electron and neutron scattering length densities for DOPC | 69 |
| 4.10 | Electron and neutron scattering length densities for POPC | 70 |
| 4.11 | Electron and neutron scattering length densities for DPPC | 71 |
| 4.12 | Electron and neutron scattering length densities for POPG | 72 |
| 4.13 | Electron and neutron scattering length densities for POPS | 73 |
| 5.1 | QEq optimization results. | 95 |
| 5.2 | <i>ab initio</i> target optimization results. | 96 |
| 5.3 | Experimental target optimization results. | 96 |
| A.1 | ParOpt design illustration | 114 |
| B.1 | Surface of annealed silica. | 134 |
| B.2 | Silica-water system snapshot | 135 |
| B.3 | Dissociation of an oxygen group from silicon. | 136 |
| B.4 | Distribution of charges in water | 138 |
| B.5 | Water Radial Distribution Function | 139 |
| B.6 | Water Mean Squared Displacement | 140 |
| B.7 | Silica Radial Distribution Function | 142 |
| B.8 | Mass Densities | 143 |
| B.9 | Silica-Water interfacial area | 144 |
| B.10 | Silica-water species counts | 145 |

| | |
|---|-----|
| B.11 Water Dipole Moment | 146 |
| B.12 Electric potential across the silica layer | 147 |

Abstract

Molecular models allow computer simulations to predict the microscopic properties of macroscopic systems. Molecular modeling can also provide a fully understood test system for the application of theoretical methods. The power of a model lies in the accuracy of the parameter values which govern its mathematical behavior. In this work, a new software, called ParOpt, for general high dimensional non-linear optimization will be presented. The software provides a very general framework for the optimization of a wide variety of parameter sets. The software is especially powerful when applied to the difficult task of molecular model parameter optimization. Three applications of the ParOpt software, and the Nelder-Mead algorithm implemented within it, are presented: a coarse-grained (CG) water-ion model, a model for the determination of lipid bilayer structure via the interpretation of scattering data, and a reactive molecular dynamics (ReaxFF) model for oxygen and hydrogen. Each problem presents specific difficulties. The power and generality of the ParOpt software is illustrated by the successful optimization of such a diverse set of problems.

Chapter 1

Introduction

1.1 Molecular Models

Molecular models provide a framework for determining how the macroscopic characteristics of a system arise from the interactions of underlying constituents. Molecular simulation can provide a fully known system for testing the validity of a theory.¹ That is, the predictions of a theoretical framework applied to a model can be compared with the direct analysis of the model.² Molecular models can also be used to explore systems for which an experimental construction would be impractical.³ If a molecular model's macroscopic properties are equivalent to those of an experimental system, the microscopic details of the model may give an indication of the underlying interactions of the real system.² Molecular models are constructed by determining meaningful elementary units of the system (e.g. an electron in an atom,⁴ a molecule in a fluid,⁵ or an amino acid in a protein⁶) which represent fundamental components. Interactions among these units, and in some cases a background environment,⁷ are chosen to best reflect the underlying physics. The mathematical descriptions of these interactions contain parameters which determine the precise nature of the relationship between model components. In order to produce a physically relevant model, these parameters must be chosen, for instance via an optimization procedure, to reproduce known characteristics or behaviors of the system.⁸ For example, the positions of atoms within a crystal can be determined by reproducing scattering data.⁹ The spatial size and time duration of systems that can be simulated are restricted by the size of fundamental units and their fastest motions.¹⁰ For example, an atomistic water model can predict the alignment of molecules near a

surface, but not the behavior of ocean currents.

An important consideration in the construction of molecular models is the complexity.¹¹ Models with more complex mathematical formulation suffer from added difficulty in both parameter optimization and calculation of model properties and predictions. Added computational costs and parameterization complications are not the only drawbacks of increasing model complexity. In addition, models with an excess of free parameters tend to have less transferability than simpler models, i.e. more free parameters allows a model to be well tuned to a specific range of condition, but the model may not be applicable in other instances. The lack of transferability in optimization problems with large numbers of parameters for optimization is a consequence of over-fitting, where a model reproduces the target precisely but lacks accuracy in conditions which differ slightly from those in the target function.¹² Over-fitting can be avoided by having sufficient quantity of data, relative to the number of free parameters.

1.1.1 Increasing model complexity

The simplest model for a collection of molecules is the ideal gas. In this model, each molecule is treated as a point particle with no intermolecular interaction inside a finite volume. Since the ideal gas molecular model has no interaction parameters, the macroscopic observables are dependent only on the macroscopic state (and fundamental physical constants). Despite the extreme simplicity of the model, it agrees well with experimental values when applied to gases with very weak interactions (e.g. Nobel gases) and at high temperatures. As an example, the ideal gas law has been applied to a gas ignited by an electric spark¹³). The ideal gas exhibits no phase behavior.

The van der Waals approach in general is based on the separation of intermolecular interactions into short range repulsion (which arises due to the Pauli exclusion principle) and the much weaker long range attraction (arising in neutral, non polar molecules primarily as a result of the induced dipole dispersion forces).¹⁴ The simplest approach, the hard sphere model, assume no attractive forces and infinitely steep repulsion. Mean field approximation to the attractive interactions replace intermolecular interactions with an average interaction with a uniform field. Extensions of this approach include replacing the hard sphere repulsion and mean-field interaction with an explicit calculation of an intermolecular, distance dependent potential. The repulsive interaction becomes steep but finite, and the mean-field interaction is replaced with pair-wise interaction. More complex

systems can be modeled by including explicit electrostatics between non-neutral molecules/sub-molecular groups and the addition of bonded interactions within molecules.

The hard sphere model is an extension of the ideal gas, where gas molecules interact via the potential

$$V_{hard}(r) = \begin{cases} \infty & \text{if } r \leq \sigma \\ 0 & \text{if } r > \sigma \end{cases}. \quad (1.1)$$

Even this simple model, with one free parameter (σ) exhibits interesting behavior, with a first order transition between a solid phase and a single fluid phase.¹⁵ The hard sphere model is a good model for fluid systems at high density, a regime when the behavior of the system is dominated by repulsive interactions.¹⁶

The van der Waals equation of state can be derived by adding interaction between particles and a cohesive background field as a perturbation to the excluded volume model¹⁵ (equivalent to the hard sphere model for low densities). The equation of state is given by

$$P = \frac{RT}{v - b} - \frac{a}{v^2} \quad (1.2)$$

where P is the pressure, v is the per particle volume, R is the universal gas constant, and T the temperature.¹⁷ The parameters a and b are specific to the system being modeled. The van der Waals fluid which has two free parameters, a (a measure of the average intermolecular attraction) and b (a measure of the excluded volume per molecule) is the simplest model which exhibits a higher-order liquid-gas phase transition,¹⁸ but cannot represent a fluid-solid transition due the breakdown of the model at high particle concentrations.

The combination of the hard sphere model and a mean field cohesive interaction is able to represent both the solid-fluid transition of the hard sphere model and the liquid-gas transition of the van der Waals fluid.¹⁹ This model is able to reproduce much of the thermodynamic behavior of a weakly interacting gas (Argon in the case of Longuet-higgins and Wisdom).¹⁹

The van der Waals fluid and the hard sphere model have molecules as the fundamental interacting units. The atomic configuration can be taken into account by defining interactions between atoms. Within a molecule, covalent bonds are typically modeled via harmonic spring potentials defined between 2 - 4 atoms.¹ Further additions allow more diverse sets of molecular systems to be

modeled. For example, polar molecules such as water have uneven distributions of charge, due to the large differences in atomic electronegativities. This can be treated in classical molecular models by introducing partial atomic charges. In effect these partial charges assign fractions of electrons to different atoms within a molecule.²⁰

All of these additions to the simple molecular models introduce complexity into the mathematical description of the interactions and more free parameters which must be adjusted in order to model real world systems. For example, the addition of partial charges adds a free charge parameter for each atom in a molecule. The Lennard-Jones interaction (equation (1.3)) involves two parameters for each pair of atom types.²¹

$$V_{\text{LJ}}(r) = \epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - 2 \left(\frac{\sigma}{r} \right)^6 \right] \quad (1.3)$$

The Morse potential (equation (1.4)), which was developed to model the vibrational spectra of diatomic molecules,²² requires three parameters and provides greater flexibility than the Lennard-Jones interaction at the expense of computation cost in the evaluation of exponential functions.

$$V_{\text{Morse}}(r) = \epsilon \left[e^{-\alpha(r-\sigma)} - 2e^{-\alpha/2(r-\sigma)} \right] \quad (1.4)$$

A modification of the Morse potential (see chapter 3) replaces the constant parameter α with functions dependent on the interatomic separation r . This introduces extra parameters, increasing the cost of optimization and simulation, but also the flexibility of the model.²⁰⁷

$$V_{\text{Mod-Morse}}(r) = \epsilon \left[e^{-\alpha(r)(r-\sigma)} - 2e^{-\alpha(r)/2(r-\sigma)} \right] \quad (1.5)$$

$$\alpha(r) = \begin{cases} \alpha_L + (\alpha_0 - \alpha_L)(1 - r/\sigma) & \text{if } r \leq \sigma \\ \beta_R + (\beta_c - \beta_R) \frac{r-\sigma}{r_c-\sigma} & \text{if } r > \sigma \end{cases}$$

where r_c is the cutoff radius of the potential function evaluation, beyond which point the potential is assumed to vanish. Figure 1.1 shows a comparison of several functions with differing model complexity. The simplest of these, the hard sphere model, requires only one parameter (σ). The Lennard-Jones potential requires two, (σ and ϵ). The Morse potential adds a third parameter (σ ,

ϵ , and α) while a modified version of the potential has a total of six (σ , ϵ , α_L , α_0 , β_R , and β_c).

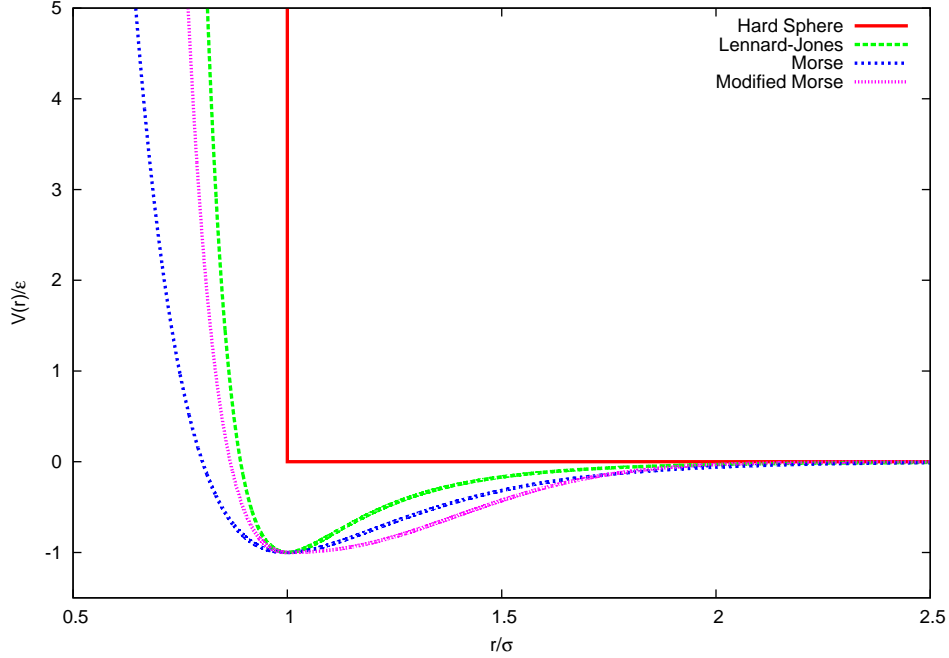


Figure 1.1: Comparison of interaction potential functions of increasing complexity. All functions are plotted as function of reduced separation ($r' = r/\sigma$) and energy ($V'(r) = V(r)/\epsilon$). The hard sphere model is plotted a sphere radius of $\sigma/2$. For the Morse potential, $\alpha = 7$ is used. For the modified Morse potential, values of used are $\alpha_0 = 13$, $\alpha_L = 10$, $\beta_c = 11$, $\beta_R = 3$, and $r'_c = r_c/\sigma = 2.5$.

Bonded interactions modeled via a harmonic potential require a “spring” constant energy parameter and an equilibrium spatial parameter (distance for a bond, angles for 3- and 4- body interactions).

1.1.2 Water

System can be studied via mathematical molecular modeling on a wide range of scales, from electron and nuclei to large collections of atoms and molecules. Independent of spatial/temporal scale, the complexity of the optimization procedure depends on the number of independent free parameters describing the interactions.

Water is a a rewarding system of study in many fields for several reasons. Water is a fundamental factor in a wide variety of fields, including biology, chemistry and physics.²⁴ Despite the ubiquity of water, its fundamental material properties are fairly atypical compared with other similar liquids.²⁵ Many of the anomalous properties of water, e.g. large heat capacity, density maximum above the

freezing point, and solvent characteristics are responsible for the importance of water.

Due to both the importance and difficulty in understanding the microscopic properties which lead to the anomalous macroscopic characteristics, many models exist which attempt to model water and its interactions with other molecules on a molecular and atomic level. The simplest of these models treat water as a background continuum. The most basic of these implicit solvent models reduce water to a constant dielectric background medium. More complex implicit solvent models utilize the Poisson-Boltzmann equation and approximation to model a water solvent with ionic concentration.²⁶ Coarse-grained models treat molecules²⁷ or groups of molecules^{28–33} as fundamental units. Three point water models typically treat water with a Lennard-Jones interaction centered on the oxygen atom. Harmonic bonds and angles between oppositely charged oxygen and hydrogen atoms give rise to a variable water molecule dipole moment, though these are often fixed for computational efficiency. Four and higher site models add extra interaction sites in order to model the electronic distribution (lone-pair electrons and atomic polarization) of the oxygen atom.^{34,35}

1.2 Optimization

Determination of parameters is vital for ensuring the accuracy of any model. In order for a mathematical model of a molecular system to have relevance to experimentally observable systems, the parameters governing their interactions must be chosen such that the model is a reasonably accurate representation of the real system. Parameters can be chosen by solving the constrained optimization problem:

$$\min_{p \in P} f(p) \tag{1.6}$$

where P is the parameter space to be explored, p is a point in the parameter space and f is a target function which measures the discrepancy between model predictions and target data.³⁶ The parameter space P may be bounded or unbounded. In the case of bounded optimization, limits are placed on parameter values, including limits on complex functions of parameter values. If a maximum value optimizes the model, the problem can be solved by minimizing $-f$.

The general problem of optimization arises in many different fields from physical sciences and engineering to social sciences and finance. Different models and methods for determining the

suitability of models lead to a wide variety of optimization problems. Because of the substantial differences in the computation costs of the comparison, the size of the parameter space, and the characteristics of the target function in optimization problems, there are many methods for obtaining solutions.

Target functions for molecular models typically have certain properties which make the optimization procedure a significant challenge. The computational cost of a target function evaluation is often very high. For the specific case of molecular models, the target function is very rarely a closed, analytic function. Rather, evaluation of the target function involves simulations of the model followed by comparison of model observations with those of either experiment or a more fundamental model (such as *ab initio* quantum mechanics in the case of atomistic molecular dynamics or atomistic models in the case of coarse-grained molecular dynamics). The comparison of simulation results can be inexpensive, if only single point calculations (e.g. partial charges, bond orders, energy) are used for tuning. When ensemble samples are required (e.g. density, diffusion), target function evaluations can take hours or even days to compute. Recent approaches to molecular model parameter optimization have included the usage of gradients of observables with respect to potential parameters.^{37,38} This typically involves the usage of second- and third-order fluctuations of MD observables in the optimization procedure, which is difficult to determine precisely.³⁸

The numerical characteristics of the target function also contribute to the difficulty of optimization. Since molecular simulation is often chaotic, very slight changes in parameter values can lead to large changes in target function values. This results in numerical instability in the target function and the possibility of stochasticity. In these cases, repeated evaluation of an identical point in parameter space can yield differing results. The potential for stochasticity and instability make the computation of a numerical derivative nearly impossible. The stochasticity of target functions also leads to a problem of uniqueness.

If the optimization lacks uniqueness, there may be many equally acceptable solutions. The uniqueness issue does not present a severe challenge in the case of molecular modeling since the target function is only one indication of the fitness or the accuracy of the model in representing a real system. Rather than requiring the absolute global minimum, we are interested instead in a point in space which has low target function value and performs well when compared with validation data not used in the optimization procedure.

When choosing an optimization method, therefore, the goal is to provide the most improvement in the parameter suitability with the fewest possible function evaluations. Additionally, the behavior of the method at high target function value is more important than the convergence properties very close to the minimum.

1.2.1 Gradient Based Methods

Gradient based optimization methods are a group of line search optimization algorithms which determine the direction for optimization in part by the computation of either a numerical or analytic gradient. The gradient (equation (1.7)) gives the direction of fastest decrease in the target function.

$$-\nabla_P f(p) = \sum_i^N \frac{\partial}{\partial P_i} f(p) \tag{1.7}$$

For the optimization of analytic functions, the computation of the gradient is a simple function evaluation, since a closed form of the gradient can be computed only once. For optimization of numerical methods, which includes almost all molecular models, the gradient is computed using at least $N + 1$ evaluations of the target function.

Steepest descent is a line search method where the direction of search at each iteration is chosen to be along the gradient, ensuring that the path is along the direction of maximal reduction of the function f at point p . The method has the advantage of requiring only once differentiable continuous functions (no higher order derivative are required). To compute the numerical derivative, steepest descent requires $\mathcal{O}(N)$ evaluation of f at each step. Steepest descent converges to only the local minimum with a rate of convergence which is very slow for certain problems.³⁶ Additional costs in computation arise when restrictions (e.g. Wolfe conditions) are imposed on step lengths.

Conjugate gradient interactively computes a new vector conjugate to all previous vectors, typically using the gradient at the initial point as the first vector. The linear conjugate gradient specifies a step length, thereby eliminating some of the extra costs of steepest descent. In the non-linear version of conjugate gradient, a one dimensional line search is performed to minimize the target function along the new conjugate vector. The method, for quadratic functions, converges in at most N iterations.³⁹

Newton’s method requires computation of the Hessian matrix and therefore a large set of second

derivatives.³⁶ This involves $\mathcal{O}(N^2)$ target function evaluations. It also requires the Hessian to be positive definite, placing further restrictions on the problems it can be applied to. The rate of convergence is quadratic. Quasi-Newton methods begin with an initial guess Hessian, typically the identity matrix, and improve the approximate as optimization progresses. It reduces the per step complexity to $\mathcal{O}(N)$ and still has a faster than linear rate of convergence.³⁹

1.2.2 Direct Search Methods

Computation of the gradient may be disadvantageous due to several possible target function properties. Instability in a target function, i.e., extreme sensitivity to small changes in parameter values, can make the determination of local gradient unreliable. Target function stochasticity contributes to instability. When the target function evaluation has a high computational cost, the N evaluations to compute a gradient at every optimization step may make such methods prohibitively expensive.⁴⁰ In these situations, direct search methods are preferred. These methods optimize by sampling the target function value itself without explicit computation of a derivative.

The genetic algorithm is a direct search method inspired by evolutionary principles.⁴¹ An individual in a population is a point in parameter space (a possible solution to the optimization problem). The fitness of each individual is defined as the target function value for the associated point in parameter space. A step in optimization consists of first selection of individuals based on fitness. Individuals with lower fitness values are culled from the population. A new population is generated from the surviving individuals by applying crossover (determination of offspring by the parameter values of one or more parents) and mutation (random modification of offspring parameter values). The fitness of the new generation is then determined by target function evaluation and the procedure iterates.

The particle swarm method is inspired by large groups of animals (e.g. swarms of insects, schools of fish) exhibiting cooperative behavior.⁴² Each individual in the swarm has knowledge of both its own lowest function value and also the lowest function value of some subset of the swarm. The velocity of the individual in parameter space is then modified to approach both of these points at each iteration. This method has the advantage of simultaneously considering both local and global information.

The Nelder-Mead algorithm is a simplex based method which iteratively evolves a set of $N + 1$

points on a surface defined on an N dimensional parameter space. The choice of a simplex is motivated by the goal of minimizing target function evaluations. The simplex has the fewest number of points required to estimate the gradient.⁴³ Since the algorithm is determined by the ordering of target function values, and not the values directly, it is well suited to the optimization of stochastic target functions. The Nelder-Mead algorithm is described in detail in chapter 2.

In the molecular models discussed in this work, target functions must be evaluated numerically. In the case of the reactive molecular dynamics forcefield ReaxFF (see chapter 5) and coarse-grained molecular dynamics (see chapter 3) target function evaluations can take hours to compute. These target functions, which require molecular dynamics simulations, also suffer from instability and stochasticity. Due to these target function properties, the Nelder-Mead method is well suited to the optimization of model parameters.

1.3 Dissertation Outline

This document presents the development of a software framework for parameter optimization, ParOpt, along with several applications. Chapter 2 presents the software design aspects and optimization algorithm implementations in ParOpt (the manual is included in appendix A). Optimization of molecular models is a high-dimensional and computationally expensive problem. The target functions also may suffer from stochasticity, instability, and a lack uniqueness. These properties make the usage of gradient based methods impractical. Therefore, development of ParOpt has focused on direct search optimization methods in general and Nelder-Mead in particular. The next several chapters, 3 through 5, demonstrate successful applications of the ParOpt to challenging molecular model optimizations, in the order of the complexity of the optimization defined by the size of the parameter space.

Chapter 3 covers the optimization of a coarse-grained (CG) model of water and ions (sodium and chloride). In this model, four water molecules are represented by one CG bead. CG water particles have internal charges separated by fixed bonds and a variable angle, thereby modeling a four-water cluster with a variable dipole moment. Ion CG beads represent partially solvated ions, with each bead consisting of one ion and four water molecules. This structure ensures similar size between beads and also allows for the future extension of the model to ion transfer between beads. All inter-

bead interactions are modeled by a modified Morse potential. The functional form of this potential is such that long-range attractive and short-range repulsive interactions are decoupled, giving the potential a high degree of flexibility in terms of parameters. This adds more parameters than the traditional Lennard-Jones interaction. The method reduces degrees of freedom in the simulated systems by increasing the degrees of freedom in the optimization procedure. Thus simulations are less computationally costly, while the parameterization, which needs to be performed only once, requires more time.

Real space molecular structure can be determined by analysis of the phase space information yielded by scattering data. Chapter 4 presents the application of ParOpt to the interpretation of lipid bilayer X-ray and neutron scattering data. The model of the lipid bilayers presented here (ADP) is atomically detailed, an extension over the widely used SDP model. Due to the large number of free parameters and small amount of experimental data, the methods developed to extract bilayer structural data are statistical in nature. Bilayer properties are determined from averages of ensembles of optimal models. The model and methods allow the future analysis of more complicated system, e.g. transmembrane proteins, bilayers with multiple lipid components, and asymmetric bilayers, which are not tractable with existing models and methods.

Chapter 5 presents an optimization of a reactive molecular dynamics forcefield (ReaxFF). ReaxFF calculates bond order, a measure of the degree of covalent bonding, for all pairs. All intramolecular, bonded interactions are written in terms of the bond order so that as interatomic distance increases, the bond order and related energies go smoothly to zero. This allows the simulation of chemical reactivity in a classical molecular dynamics framework. Additionally, since changing bonds implies changing electronic structure, atomic partial charges are calculated dynamically via a charge equilibration procedure, QEq. Though the ReaxFF model has been applied successfully to a constant volume water system (see appendix B), the model lacks accuracy in the isobaric ensemble. This limits the applicability of ReaxFF to biological problems, where constant pressure is normally required. Additionally, though hydrogen peroxide is in an important oxidative species in biological systems, the available ReaxFF parameter sets do not model its interactions well. Since the size of the parameter space for oxygen and hydrogen is on the order of a hundred and the calculation of macroscopic model properties is computationally expensive, the optimization procedure is broken into multiple phases. First, the QEq parameters are independently optimized

to reproduce quantum partial charges. Using these parameters, the bond energy and related terms are optimized against single point quantum energy calculations. A subset of the these parameters, which define the non-bonded interactions, is then optimization to reproduce experimental water properties. The final two steps in the procedure are performed iteratively to ensure that the model is able to simultaneously reproduce both *ab initio* calculation and experimental observations.

Appendix B presents an application of the ReaxFF to the interface between an nanoscale silica film and water. Due to the reactive nature of the model, many interesting properties of the system are determined, including the composition of silanol groups on the surface, a possible mechanism of water diffusion across the interface, and the rate of hydrogen diffusion via a bond hoping mechanism in the bulk silica region.

Chapter 2

ParOpt

2.1 A Note to the Reader

This chapter contains previously published material, reused with permission, from J. C. Fogarty, S.-W. Chiu, P. Kirby, E. Jakobsson, and S. A. Pandit. “Automated Optimization of Water–Water Interaction Parameters for a Coarse-Grained Model”. *J. Phys. Chem. B* 118.6 (2014), 1603–1611.

2.2 Introduction

A software for parameter optimization (ParOpt) was developed in order to provide a general framework for the treatment of challenging numerical target functions using various methods. The focus in the design of ParOpt was on generality and modularity. To this end, target function evaluations are performed external to the optimization routine, allowing users to employ custom or existing software to compute any desired target function. This chapter serves as an overview of the design and general structure of ParOpt, as well as a description of the primary methods implemented. The user manual describes in detail the usage of the software and is included in appendix A. In the late development stages, a similar software, GenOpt (Generic Optimization Program),⁴⁴ was found which has similar goals as ParOpt. The existence of GenOpt was missed due its exclusive application within the building optimization field and without widespread use or citation. The GenOpt software implements the Nelder-Mead simplex method, as well as particle swarm and other optimization approaches. The major differences between GenOpt and ParOpt are due to the

targeted user bases. GenOpt is written with portability in mind. To this end, it was developed in the Java programming language, to maximize the variety of computation platforms on which it can be executed. ParOpt on the other hand, was developed with large scale computing in mind. It is therefore written in C, and relies on the flexibility of a Unix based environment. ParOpt provides additional abstraction barriers to aid in executing long time simulations and multi-step analysis and comparison. Additionally, ParOpt provides an extremely general framework for defining constraints on parameter space, described in section A.3.5, which has been instrumental in tuning interrelated parameters in chapter 3 and chapter 4. ParOpt is available for download under the GNU public license at <https://csmllabfs1.cas.usf.edu/Sites>.²⁰⁷

The task for the ParOpt user is to define the parameter space over which optimization will proceed, determine the target function defined on that parameter space, choose an optimization method, and execute ParOpt. The software then performs target function calculations by executing external software, and communicates with those commands via the writing of force field files to disk. Because of this approach, target functions with a closed form or very computationally inexpensive calculations would suffer from the overhead costs of running ParOpt. These types of problems would be best approached using different software. An exception to this rule, the lipid structure calculations presented in chapter 4 are implemented internally, requiring no external procedures or disk operations. Since the Nelder-Mead algorithm is the method employed in the applications of ParOpt in this work, it will be described in greater detail (see section 2.5.3) than other optimization methods.

2.3 Parameter Space Definition

The parameter space in a ParOpt optimization is defined by determining how an external set of force field files which govern the interactions will be written. The user specifies the format and structure of these files. For each target function evaluation, the ParOpt software writes these files to disk. The number of variable parameters to be written sets the dimensionality of the problem. Users are able to define constant parameters which are used for force field file preparation but are not a part of the parameter space. These force field files are then used by later phases of the function evaluation. The parameter space can be restricted by the application of minimum and

maximum values on individual parameters or arbitrary functions in parameter space. How these constraints are treated depends on the optimizer utilized. In the case of Nelder-Mead, constraints may be handled in two ways. As suggested by Nelder-Mead,⁴⁵ an very high target function value can be assigned to points outside of the allowed parameter space, ensuring that these points will not be added to the simplex. Alternatively, points selected outside of the parameter space can be mapped onto the allowable space. In the case of minimum and maximum constraints, this involves setting the parameter value equal to the extreme value. For complex functional constraints (see section 2.3.1), this mapping is user defined, and is determined via similar calculations as the constraints themselves.

2.3.1 Parameter Constraints

Constraints, or bounds on the parameter space, can be specified in two ways. First, users can specify minimum and maximum values for each parameter. These ranges are used to define the construction of initial simplexes in Nelder-Mead, and are optionally used to define simple bounds, dependent on only individual parameters, on the parameter space. More general constraints can be applied using a custom constraint language. This language consists of operators, their arguments, and a symbol table. The symbol table consists of mappings from parameter names to their associated values. Operators include comparisons (e.g. `<`, `>=`) logical operations (`&`, `|`, `!`) a conditional operator (`if`), and an assignment operator. A user defines a constraint using the assignment operator: `(assign param_name value)`. The value to be assigned to the parameter is determined by the location of the point in parameter space, using comparisons and conditionals. The conditional operator takes three arguments, a boolean, the value to return if true, and the value to return if false. For example, a user may specify a minimum value, and a mapping from values outside of the bound onto the boundary using:

```
(assign param (if (> param MAX) MAX param)
```

This statement sets the value for the parameter with name `param` to `MAX` if the parameter value exceeds `MAX`. Otherwise, the parameter retains its initial value. Chapter 4 presents the optimization of an atomically detailed model for lipid bilayers. In this model, atoms are constrained to be within a bond length of each other. This requirement is applied via:


```

(assign atom1 (if (> (- atom1 atom2) BOND_LENGTH) \\
  (+ atom2 BOND_LENGTH) atom1))
(assign atom1 (if (> (- atom2 atom1) BOND_LENGTH) \\
  (- atom2 BOND_LENGTH) atom1))

```

where `atom1` and `atom2` are the mean positions of two atoms, and `BOND_LENGTH` is the maximum possible bond length. Using this language, the user can specify very complex bounds on the space which depend on any number of parameters. An important consideration when applying such constraints is the order in which they are performed, as changing the order of operation can change the resulting point in parameter space.

2.4 Target Function Definition

In ParOpt, a target function evaluation is divided into three distinct phases: simulation, analysis, and comparison. Since these phases have important logical differences, this abstraction makes for a more powerful framework. In optimization of molecular models, the simulation phase is typically the most computationally expensive. Therefore, the simulation phase in ParOpt is parallelized, with the assumption that all simulations are independent. Simulations can be defined with multiple phases, to be executed sequentially. Simulations should be dependent only on the point in parameter space being considered. After all simulations are completed, the analysis and comparison phases are performed. Since each target function component is treated independently, each component has both an analysis and a comparison phase. Target function component evaluations are performed serially, in the order in which they are defined by the user. During the analysis phase, aggregate information about the model is extracted from the full trajectories generated during the simulation phase. This data is then output in a format compatible with the target data. The comparison phase is the comparison of the target data with the observed data. This phase, which should require minimal processing of the data, returns a single real number which represents the function value of the associated target function component to ParOpt. The user has great freedom in the ParOpt framework and has the ability to reduce the entire target function calculation to an inadvisably complex comparison method which depends directly on the parameter values.

2.5 Optimizers

The focus of ParOpt is on the optimization of molecular models, which typically involves simulation phase calculations with very high computational costs. Applications of ParOpt on force field optimization have been for target function evaluations which take up to a day to complete (see chapter 5). Due to these long simulation times, the choice of optimizers in ParOpt has been driven by the reduction of number of function evaluations. Also, the focus of ParOpt development has been toward the most rapid reduction of target function value, with the least computational cost, as opposed to methods with ideal convergence properties.

2.5.1 Steepest Descent

A steepest descent like optimizer is implemented in ParOpt as a very basic optimization procedure.⁴⁶ Each dimension in parameter space is treated independently. For a given parameter, three target function evaluations are performed, for the initial value and for the initial value incremented and decremented by a given step size. An exact quadratic fit to these three points is performed, and a corresponding move towards the center of the parabolic is performed. This procedure is performed sequentially for each parameter, and iteratively over the entire set of parameters.

2.5.2 Particle Swarm

The particle swarm method is an optimization routine inspired by the natural behavior of large groups of insects. Each particle is a single point in parameter space for each trajectory step. Each particle records the both lowest target function value it has visited and the lowest value for a certain subset of the swarm. Each particle is able to explore the space in a manner determined both by the local and the global environments. The subset of the swarm that each particle communicates with defines a graph on the particles. Different connectivities in this graph yield different optimization methods. For ParOpt, the fully connected graph, i.e. all particles communicate the lowest point with all other particles, is implemented. An initial set of particles, with random positions within the allowed parameter space and zero velocities defines the initial set of particles. Particle velocity

is then updated at each trajectory step by

$$V_i(t) = \omega V_i(t-1) + [\Phi_p R_p(X'_i - X_i(t)) + \Phi_g R_g(X' - X_i(t))] \quad (2.1)$$

where X'_i is the lowest target function value explored by particle i , X' is the lowest value explored by all particles, Φ_p , Φ_g , ω are parameters which balance the optimization between local and global optimization, and R_p and R_g are random numbers uniformly distributed on the interval $[0, 1)$. The position of each particle is then updated by its velocity. Several different cutoff methods are available, including a cutoff on the root mean squared difference in particle positions.

2.5.3 Nelder-Mead

Nelder-Mead is a popular direct search method. The preference for direct search methods in the implementation of ParOpt was motivated primarily by the challenging properties of certain target functions, i.e. instability and stochasticity, which make the direct computation of a gradient impractical. There are drawbacks to the usage of the Nelder-Mead method, most importantly a lack of guaranteed convergence and the possibility of slow rates of convergence, especially near minima.⁴⁷ Nelder-mead can be shown to converge to non minimizing points in parameter space in certain situations.^{48,49} These issues may prove significant in certain situations, but when optimizing molecular models, the problems of convergence are not particularly serious. In molecular model optimization, the absolute minima is not always required or available. The stochasticity of the target functions, as well as the disparity in the types of target data in many cases means that no optimization method can approach arbitrarily close to a minimum. Since the Nelder-Mead algorithm relies on the ordering of points in parameter space, it is less sensitive to stochasticity than other methods. Additionally, the results of optimization will be applied to a much wider set of data, and the optimal model in reproducing the target data is not guaranteed to be the best model for other situations. Despite the lack of attractive convergence properties, the Nelder-Mead method is a powerful tool in the case of molecular model optimization due to the high cost of the target function and the inability to calculate gradients and higher order derivatives. Furthermore, the Nelder-Mead method provides significant improvement in the early stages of optimization, and has very good efficiency in terms of target function improvement per function evaluation.⁵⁰

The Nelder-Mead method produces a set of optimal parameter values by iteratively evolving $M + 1$ points on the M -dimensional parameter space using four basic moves⁴⁵ :

Reflect: Replace the highest point with a point reflected about the centroid of the remaining points.

$$P_r = (1 + \alpha)\bar{P} - \alpha P_h \quad (2.2)$$

Expand: If the reflected point is lower in value than all points in the centroid, consider a point further from the centroid.

$$P_e = \gamma P_r + (1 - \gamma)\bar{P} \quad (2.3)$$

Contract: If the reflected point would remain the highest, consider a point nearer to the simplex.

$$P_c = \beta P_h + (1 - \beta)\bar{P} \quad (2.4)$$

Shrink: Move all points nearer to the lowest point.

$$P_i = \delta P_i + P_l(1 - \delta) \text{ for all } P_i \text{ with } i \neq l \quad (2.5)$$

The $M + 1$ individual points in the in the M dimensional space are denoted by P_i while the subscripts h and l represent the lowest and highest points respectively. The centroid is defined by

$$\bar{P} = \frac{1}{M} \sum_{i \neq h}^{M+1} P_i. \quad (2.6)$$

Values used for optimization of water parameters are those suggested by Nelder-Mead:⁴⁵ $\alpha = 1.0$, $\gamma = 2.0$, $\beta = 0.5$, and $\delta = 0.5$. Figure 2.5.3 graphically illustrates the Nelder-Mead simplex transformations.

The Nelder-Mead algorithm replaces the highest point in the simplex with one of lower target function value by applying these moves. A Nelder-Mead step begins with calculation of a reflect move. If the target function value for this new reflected point is lower than all points in the simplex, an expansion move is attempted. If this expanded point is lower than all points in the simplex, it is accepted. If the reflected point is not higher than the highest point in the new simplex, it is accepted and replaces the highest point in the simplex. If the newly accepted reflected point is still

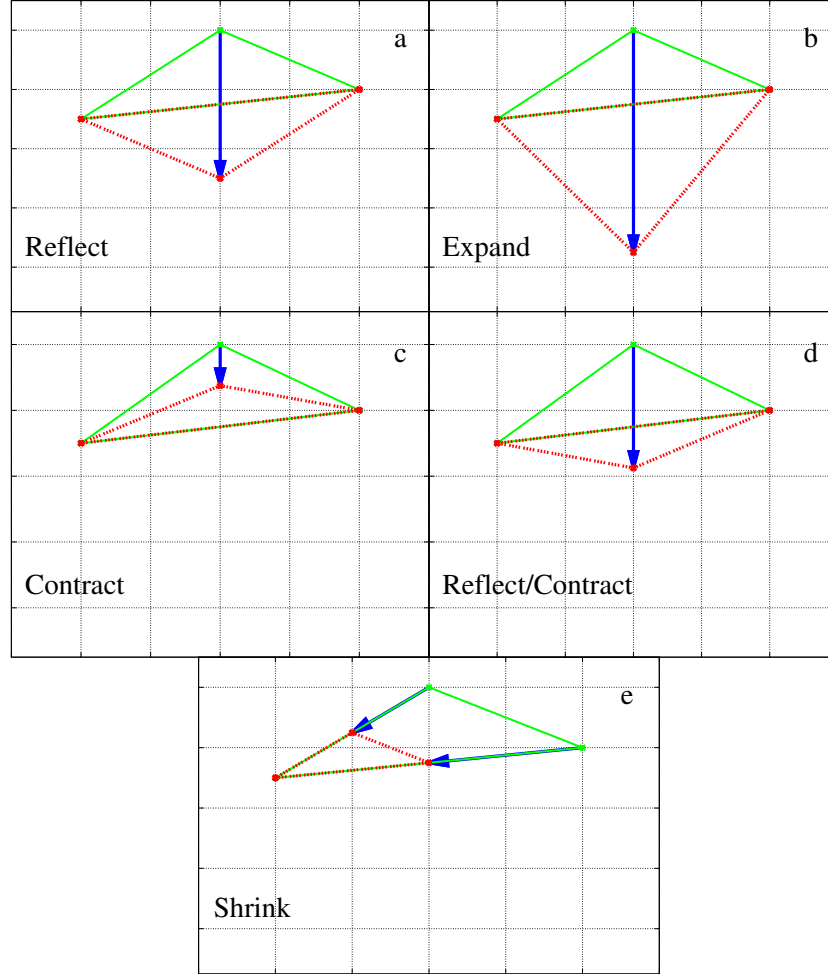


Figure 2.1: Schematic illustration of Nelder-Mead simplex transformations. The initial simplex is drawn solid green and its transformation in dotted red. Blue arrows are drawn from initial to the new point considered. Transformations are: (a) the reflection of the highest point about centroid of the remaining points (equation (2.2)), (b) the further expansion along the line of reflection (equation (2.3)), (c) the contraction along the line joining the centroid and the highest point (equation (2.4)), (d) a reflection followed by a contraction, and (e) the shrinking of the simplex by shifting all points nearer to the lowest point (equation (2.5)). Values for the transformation coefficients are those suggested by Nelder-Mead:⁴⁵ $\alpha = 1.0$, $\beta = 0.5$, $\gamma = 2.0$, and $\delta = 0.5$.

the highest point in the simplex, a contraction step is attempted. If the contracted point is higher than the highest point in the simplex, then a shrink move is applied, replacing all points except the lowest. If the contracted point is lower than at least one point, it is accepted into the simplex. This process is repeated until the stopping criterion is met. The stopping criterion for the Nelder-Mead method is a cutoff on the root-mean-squared target function value over the entire simplex. The number of occurrences of each move in an optimization depends on the characteristics of both the simplex and the target function, and is therefore problem specific.

The original description of Nelder-Mead has no set method for boundary conditions on the parameter space, though suggestions are made for modification of functional forms or assignment of a large penalty in the target function for points outside of the boundary. For the applications of the ParOpt software and the Nelder-Mead method in this work, points outside the boundary were replaced with a point inside the allowable space.

Chapter 3

Coarse-Grained Water/Ion

3.1 A Note to the Reader

This chapter contains previously published material, reused with permission, from J. C. Fogarty, S.-W. Chiu, P. Kirby, E. Jakobsson, and S. A. Pandit. “Automated Optimization of Water–Water Interaction Parameters for a Coarse-Grained Model”. *J. Phys. Chem. B* 118.6 (2014), 1603–1611.

3.2 Introduction

Water and aqueous electrolyte solutions play a fundamental role in a wide range of fields of study, including geology, atmospheric sciences, chemistry, biology, and physics.²⁴ Due to its relevance and ubiquity, water is the most common solvent in experimental and computational studies of biological systems.⁵¹ The hydrophobic effect, the segregation of molecules based on relative interactions with water, governs membrane self-assembly, protein folding, and many other biological processes.⁵² Despite its seemingly simple structure, the water molecule forms intricate and dynamic networks of hydrogen bonds which give it unique bulk and interfacial properties.^{53,54} The wealth of anomalous characteristics that water exhibits relative to most other liquids underscore its uniqueness. In part due to this enigmatic character, it has been the subject of many theoretical and modeling efforts in the last few decades.²⁵ Reproducing these anomalous characteristics poses a serious challenge for any effort to attain a detailed understanding of water dynamics and structure. Accurate modeling of ions in aqueous environments is vital for the simulation of biological systems.⁵⁵ Ionic characteristics

and structure are important in the stability of liposomes.⁵⁶ Monovalent cations have a strong effect on membrane stability.⁵⁷ Specific ion binding has been shown to affect the structure and function of proteins.⁵⁸ Ionic interactions play a vital role in the biological function of nucleic acids and RNA in particular.⁵⁹ Because ions of biological systems are primarily located in aqueous environments and fully solvated, the crucial first step in understanding the important role that ions play in biological processes is to determine the precise nature of their interactions with water. Existing molecular models for water and ion systems range in scale from *ab initio* electronic to atomistic molecular dynamics (MD) to coarse-grained (CG) MD and implicit electrolyte solvent methods. Ren et al. have presented an informative review of solvent-ion interaction models across a wide range of simulation scales.²⁰

Quantum mechanical dynamic models based on first principles (*ab initio*) evolve the electronic structure along with nuclear coordinates through time. The resolution of these methods is at the level of molecular orbitals. These *ab initio* models produce radial distribution functions (RDFs) and rotational/vibrational spectra which strongly agree with experimental data.^{60,61} Quantum mechanical methods are able to calculate, to a high degree of accuracy, the properties of small water-ion clusters.^{62,63} The computational complexity of quantum models restricts systems to spatial and temporal ranges too limited to accurately determine most statistical bulk quantities. These properties, such as diffusion coefficient and thermodynamic variables, are more accurately modeled by semi-empirical approaches. Additionally, the simulation of collections of large biomolecules remains beyond the reach of *ab initio* methods.

Water models with atomic level detail are the most common for simulation of biomolecules and their aggregates. In order to reproduce a wide range of complex water behavior, several models have been developed, consisting of four (Bernal-Fowler,³⁴ TIP4P,¹¹ TIPS2⁶⁴) and five (TIP5P³⁵) interaction sites. In the interest of reducing computational cost, most simulations use models (SPC,⁶⁵ SPC/E,⁶⁶ TIP3P⁶⁷) that are composed of three partially charged atoms connected with rigid bonds and a Lennard-Jones (6-12) intermolecular potential between oxygen atoms. Each model is reasonably accurate within its domain of applicability, but none reproduce a large portion of the anomalous properties of water. Various models⁶⁸ include flexible bonds which allow the water dipole moment to change with the surrounding environment. Atomistic methods for the simulation of ionic systems typically treat ion atoms as charged point particles interacting with the solvent via

long-range Coulombic interactions and short-range Lennard-Jones interactions. Though classical MD simulation of ionic interactions may lack qualitative agreement in certain properties with experimental or *ab initio* methods, it provides significant information on the nature of ion-solvent effects.⁶⁹ Polarizable molecular dynamics models, which includes variable multi-pole moments in a classical MD framework, are well suited to the study of ion-water interactions.⁷⁰ A comparison of *ab initio*, classical, and polarizable MD simulations for the potassium ion found that all three models make similar predictions for the solvation structure.⁷¹

Coarse-grained (CG) models, which model interactions between groups of atoms, are able to achieve longer time scales by averaging over the smallest timescale inter- and intramolecular motions, thereby enabling a larger timestep. Moreover, CG models are able to effectively represent a larger number of molecules by reducing the number of interaction sites and degrees of freedom considered. Due to the unusual properties of water, development of a CG model that reliably describes the bulk and medium-range properties of water and ionic interactions is a challenging project. Development of models of molecular assemblies with lower complexity that reproduce important qualities of water is ongoing.^{72,73} For an extensive review, see the book “Coarse-Graining of Condensed Phase and Biomolecular Systems”.⁷⁴ Johnson et al⁷⁵ have identified limitations of CG models by demonstrating the lack of transferability of a CG pair potential across different states. The work also shows that the pair potential cannot simultaneously resolve all the properties of the reference system for a given state. Despite these limitations, there is extensive development of various CG models for water.^{27–33,76–80} One common CG protocol is to group a number of water molecules into a single bead with its center as the interaction site.^{27–33} This class of CG models carry no charge. Electrostatic interactions are implicitly incorporated into the effective pairwise interaction potential which is either a Lennard-Jones (LJ)^{28–31} or a Morse like function.³² These models typically model a single ion atom along with several solvating water molecules as a single point particle, e.g. a single MARTINI ion bead with a screened partial charge of $0.7\ e$ represents one ion atom and six water molecules.²⁹ The free parameters in pair potentials are trained against thermodynamics properties such as density, surface tension, and solvation free energy. Molinero and Moore²⁷ built a coarse-grained water model (mW) around the tetrahedral properties shared by water, silicon, and carbon by adapting the Stillinger-Webber potential originally developed for silicon. The model maps one water molecule onto one interacting bead and describes important

properties of water across a wide range of temperatures. The extension of this model to sodium and chloride is done without using explicit charges in order to avoid calculation of long range electrostatics.⁸¹ Replacement of long range electrostatics with a short range interaction has also been performed by a coarse-graining of interactions.⁸² Using a clustering algorithm, Hadley and McCabe derive the effective potential through fitting to structural data.³³ The models of Shelley et al.²⁸ and Chiu et al.³² have large isothermal compressibility. Using a LJ 12-4 potential, Shinoda et al.³¹ have designed a CG water which has density and surface tension comparable to those of water. Electrostatic interactions can be introduced into CG water models with multiple charged sites per bead.⁷⁶⁻⁸⁰ These models are able to describe the dielectric properties of water. Despite incorporation of electrostatic interactions, both the BMW and polarizable MARTINI models predict densities different from experimental measurements.^{77,78} The CG representation by Darré et al.⁷⁹ associates approximately 11 water molecules with four tetrahedrally interconnected beads (WT4). The model explicitly accounts for long-range electrostatics. The parameter set of the WT4 model is tuned to match the experimental density. On the other hand, the isothermal compressibility and surface tension are modeled less accurately.⁷⁹ The recent GROMOS CG⁸⁰ is a 5-to-1 mapped water model with two electrostatic interaction sites. It delivers good results in density, surface tension, and dielectric constant as compared to those of real water but yields a coefficient of thermal expansion an order of magnitude higher than experiment. Adaptive multiscale models allow transfer between simulation domains with differing granularity, from quantum level up to CG.⁸³⁻⁸⁵ Adaptive methods face unique challenges due to the variability in degrees of freedom over the span of a simulation.

Continuum models replace explicit solvent molecules with a uniform background field and modified interatomic interactions. For example, the Generalized Born approximation to the Poisson-Boltzmann equation can replace both explicit water molecules and salt atoms (at the Debye-Hückel level of accuracy) with an implicit solvation medium.²⁶ A related approach involves deriving an effective interaction between ions without consideration of explicit water molecules..⁸⁶ Implicit solvent methods are a significant approximation of the interaction between water, ions, and biologically relevant molecules. Such models can be applied at large scales, but do not accurately model smaller scale effects.

In an extension of previous CG development based on the Morse potential³² (referred to as

CSJ), the CG water model presented here has been upgraded with polarizability and a more complex modified Morse potential with more flexibility in the functional form. Interactions between water beads are tuned to reproduce experimental observables: density, coefficient of diffusion, electric permittivity, and surface tension. The model is further extended to represent salt solutions, specifically systems containing sodium and chloride ions. The CG ion beads model ion atoms partially solvated by four water molecules. Interactions between all pairs of beads, including explicit cross terms, are simultaneously optimized to reproduce experimental solvation free energies for each ion and density at two different ionic concentrations. Additionally, a comparison of CG model RDFs to those determined from atomistic MD simulation is included for ionic species. This new model is referred to as Modified Morse Coarse-Grain (MMCG).

3.3 Model

Water CG beads in the MMCG model represent four water molecules. Though the solvation numbers for Na and Cl are usually determined to be between 5 and 6, the MMCG model maps one ion and four water molecules into one bead. This mapping, which considers CG ion beads to be partially solvated ions, was chosen for several reasons: 1) it allows for the future introduction of an ion transfer mechanism into the force field, whereby neighboring beads can exchange ions by switching CG types and 2) the mapping maintains a consistency of spatial size and mass which allows for efficient determination of timescales within the system. The interaction potential for CG bead interactions must model the average of several different interatomic and intermolecular forces. Therefore, the functional form is solely based on a phenomenological understanding of the system rather than a derivation from first principles, such as the r^{-6} behavior of the dispersion force included in the Lennard-Jones potential. Due to a high degree of functional flexibility, a modified Morse potential (eq. (3.1)) was used to describe interactions between CG beads.

$$V_{ij}(r_{ij}) = \begin{cases} D_{e_{ij}} \left[e^{\alpha(r_{ij}) \left(1 - \frac{r_{ij}}{R_{ij}}\right)} - 2e^{\frac{\alpha(r_{ij})}{2} \left(1 - \frac{r_{ij}}{R_{ij}}\right)} \right] & \text{if } r_{ij} \leq R_{ij} \\ D_{e_{ij}} \left[e^{\beta(r_{ij}) \left(1 - \frac{r_{ij}}{R_{ij}}\right)} - 2e^{\frac{\beta(r_{ij})}{2} \left(1 - \frac{r_{ij}}{R_{ij}}\right)} \right] & \text{if } r_{ij} > R_{ij} \end{cases} \quad (3.1a)$$

$$(3.1b)$$

where

$$\alpha(r_{ij}) = \alpha_{L_{ij}} + (\alpha_{0_{ij}} - \alpha_{L_{ij}}) \left(\frac{R_{ij} - r_{ij}}{R_{ij}} \right)$$

$$\beta(r_{ij}) = \beta_{R_{ij}} + (\beta_{c_{ij}} - \beta_{R_{ij}}) \left(\frac{r_{ij} - R_{ij}}{r_c - R_{ij}} \right)$$

where r_{ij} is the separation between bead of type i and j , R_{ij} is the equilibrium distance and $D_{e_{ij}}$ determines the well depth. Compared to the Morse potential, this modified form replaces α , the scaling parameter in the exponent, with two separate distance dependent variables $\alpha(r_{ij})$ and $\beta(r_{ij})$. This form decouples the repulsive (eq. (3.1a)) and attractive (eq. (3.1b)) parts of the potential. The shape of the repulsive term as r_{ij} tends to 0 is determined by α_0 . The exponential factor tends toward $\alpha_{L_{ij}}$ as r_{ij} approaches R_{ij} . The parameter $\alpha_{0_{ij}}$ determines the potential as $r_{ij} \rightarrow 0$, $\alpha_{R_{ij}}$ determines the curvature of the potential as $r_{ij} \rightarrow R_{ij}$ from the left, $\beta_{R_{ij}}$ determines the curvature of the potential as $r_{ij} \rightarrow R_{ij}$ from the right, and $\beta_{c_{ij}}$ determines the behavior of the potential as r_{ij} approaches the cutoff, r_c . This more complicated functional form allows a reduction of degrees of freedom in the simulated system while maintaining a similar accuracy by increasing the degrees of freedom during the optimization procedure. Due to the complexity of the potential function, function evaluations were performed using table lookups. Beyond the cutoff distance r_c , the potential is taken to be 0. The conventional value of $r_c = 1.6$ nm gives good results with reasonable computational efficiency. Comparing systems representing equal number of water molecules, the MMCG model achieves greater than an order of magnitude improvement in performance over an atomistic system. To model the charge screening due to partial solvation, ions were assigned charges (Q_{ion}) with magnitude less than 1.0. Polarizability of CG water beads is modeled similarly to the polarizable MARTINI water model,⁷⁷ through the addition of a harmonic angle potential term, with spring constant K_0 and equilibrium angle θ_0 , between three charged points. This model of polarizability allows for not only dipole, but also higher moments of the electrostatic energy.

Figure 3.1 provides an illustration of the MMCG model for water. The central point, which acts as the interaction site for the modified Morse potential, carries negative charge Q , while the two outer points each carry charge $-Q/2$. A mass equal to that of four water molecules is distributed evenly between the three points. The distance between the outer points and the

central point is a tunable parameter I . The change in the angle (θ) between the points represents the polarizable nature of the bead. The set of modified Morse potential parameters de-

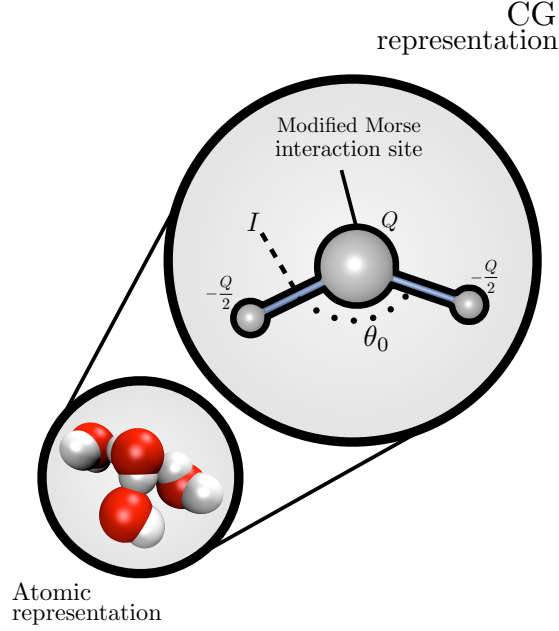


Figure 3.1: Illustration of the MMCG water model: Four water molecules map onto one CG bead. Each CG bead has three charge sites. The central site with charge Q provides the interaction site for the modified Morse potential. Outer charge sites (with charge $-Q/2$) are connected to the central site by a fixed bond of length I . The central angle is a determined by a harmonic potential with equilibrium angle θ_0 .

scribing water–water and water–ion interactions $\{D_{eij}, R_{ijij}, \alpha_{0ij}, \alpha_{Lij}, \beta_{Rij}, \beta_{cij}, \theta_0, K_0, Q, I, Q_{ion}\}$ determines the space for optimization. No combination rules are utilized in the model, therefore all interactions are tuned explicitly. To simplify the optimization procedure, water–water interactions were treated independently, followed by the simultaneous optimization of water–ion and ion–ion interaction parameters. For the water–water interactions, the set of parameters was $\{D_{eij}, R_{ij}, \alpha_{0ij}, \alpha_{Lij}, \beta_{Rij}, \beta_{cij}, \theta_0, K_0, Q, I\}$ where $i = j = \text{Water}$. For the water–ion and ion–ion interactions, the set was $\{D_{eij}, R_{ij}, \alpha_{0ij}, \alpha_{Lij}, \beta_{Rij}, \beta_{cij}, Q_{ion}\}$ where $i, j \in \{Na, Cl, Water\}$, excluding the water–water interaction.

3.4 Target Data

3.4.1 Water–Water

For optimization of the MMCG water model, the target function was defined by the weighted percent error in comparison of CG simulation results with four experimental quantities: density, dielectric constant, diffusion coefficient, and surface tension. To ensure that all comparison data contribute to the target function and thus the evolution of the simplex, percent errors are assigned scalar weights in order to yield similar orders of magnitude between all quantities. With this goal in mind, density was weighted 100 times higher than surface tension, while permittivity and diffusion coefficient were weighted 10 times higher. Experimental values used for comparison are presented in table 3.2.

Density: Density was computed by taking the ratio of the mass of the system and the ensemble average volume.

Diffusion Coefficient: The standard practice in comparing diffusion coefficients between CG models and experiment is to assume that diffusion for N -sized clusters scales as $1/N$, i.e. $D_N = \frac{D_1}{N}$ where D_N is the diffusion coefficient for an N -sized cluster.⁸⁷ This assumption neglects any correlation within the cluster. In order to numerically calculate the diffusion coefficient scaling, clusters in a simulation of SPC/E water were constructed from each molecule and its nearest $N - 1$ neighbors. The GROMOS CG work⁸⁰ compares CG bead diffusion coefficient with the diffusion coefficient for the center of mass of clusters in the SPC water model. In that work, static clusters in the atomistic representation were produced by adding distance restraints between oxygen atoms. The analysis produced diffusion coefficient scaling close to the inverse behavior usually assumed. In the current work, clusters were constructed similarly, though unrestrained. The MSD for such clusters was calculated using a window averaging method with eq. (3.2). Though the method presented here may not be generalizable to all types of clusters, it provides an improvement over current assumptions for homogeneous systems. Figure 3.2 shows the diffusion coefficient scaling for SPC/E water. The plot for atomistic water shows significant deviation from the scaling behavior of uncorrelated clusters, especially at larger N . Therefore approximations leading to $1/N$ scaling

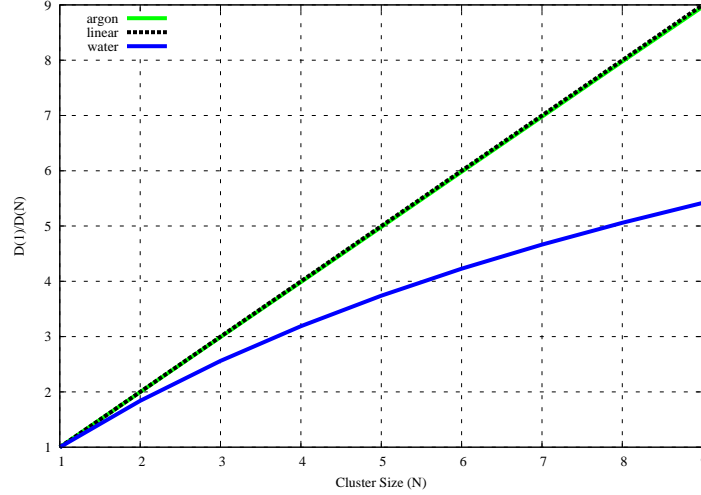


Figure 3.2: MSD Scaling for SPC/E Water: Linear shows the usual practice of taking the diffusion of an N cluster begin N times slower than the single molecule. SPC/E was determined by simulation of SPC/E at 303K. Clusters were chosen by the $N - 1$ nearest neighbors for each water molecule, and the center-of-mass diffusion was calculated. The weakly interacting Argon gas closely matches a linear scaling behavior.

are more appropriate at smaller cluster sizes. At a bead size of 4 water molecules, the difference is important. On the basis of this analysis, a scale factor of $s = 3.16$ is used to compare the CG diffusion coefficient with experiment, instead of the usual factor of 4.

$$D_1 = sD_N = s \lim_{t \rightarrow \infty} \frac{\langle [x(t) - x(0)]^2 \rangle}{6t} \quad (3.2)$$

where $x(t)$ gives the position of the central bead at time t and $\langle \dots \rangle$ denotes an ensemble average.

Relative permittivity: Relative permittivity (ϵ) was calculated using a Clausius-Mossotti-like equation with a reaction field of infinite dielectric constant (conducting boundary conditions).⁸⁸

$$\epsilon = 1 + \frac{\langle M^2 \rangle - \langle M \rangle^2}{3\epsilon_0 \langle V \rangle T} \quad (3.3)$$

where M is the total system dipole, ϵ_0 the permittivity of free space, $\langle V \rangle$ the ensemble average system volume, T is the target temperature for the thermostat, and ϵ_0 the permittivity of the vacuum.

Surface tension: Surface tension was taken from the GROMACS internal calculation:

$$\gamma(t) = \frac{L_z}{N_s} \left\{ P_{zz}(t) - \frac{P_{xx}(t) + P_{yy}(t)}{2} \right\} \quad (3.4)$$

where the z -axis is normal to the interface, P_{nn} are the pressure tensor diagonal elements, N_s is the number of interfaces, and L_z the size of the system along the z -axis.

3.4.2 Water–Ion

For the simultaneous optimization of water–ion and ion–ion interaction parameters, the training set was composed of densities for two NaCl concentrations, 0.1 and 0.25 mol kg^{−1}, atomistic radial distribution functions (RDF) for the same concentrations, and the experimental absolute solvation free energies for each ion.

Solvation Free Energy: In order to calculate the free energy of solvating an ion from the solvation of a partially solvated CG bead, a solvation pathway equivalent to that of a single ion in water was constructed. Figure 3.3 schematically illustrates this pathway. In order to compare to the absolute solvation free energy of an ion (ΔG_0), three intermediary solvation steps, ΔG_1 , ΔG_2 , and ΔG_3 , must be determined. The partial solvation of the ion (ΔG_1) is the binding energy, determined using experimental methods,⁸⁹ of the 4 water + 1 ion cluster. A value of $\Delta G_1 = 99.2$ kJ mol^{−1} was used for Cl, while a value of 196.2 was used for Na. Solvation of the resulting cluster in bulk water (ΔG_2) was determined from the solvation of the CG ion in the CG water solvent. The removal of four water molecules from the bulk ($\Delta G_3 = -37.6$ kJ mol^{−1}) was determined using the negative of the solvation free energy of a CG water bead, determined using the same procedure as the ions (equation (3.5)). The calculation of ΔG_2 was performed via thermodynamic integration at every target function evaluation, while the other contributions were constant over the parameter space. Thermodynamic integration was performed by splitting the integration

$$\Delta G_2 = \int_{\lambda=0}^1 \left\langle \frac{\partial V(\lambda)}{\partial \lambda} \right\rangle d\lambda \quad (3.5)$$

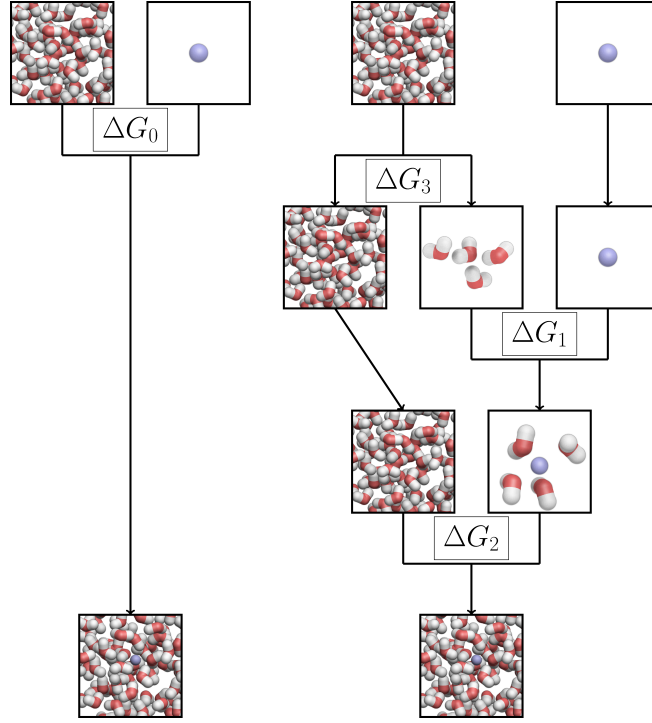


Figure 3.3: The solvation free energy pathway. In order to compare to the solvation free energy of an ion in bulk water (ΔG_0), a path is constructed by first separating a four-water cluster from the system (ΔG_3). A single ion is then partially solvating an ion with four water molecules (ΔG_1). This partially solvated ion is then solvated by a bulk water system. It is this solvation free energy (ΔG_2) which is directly calculated during optimization.

(eq. 3.5) into two phases.

$$\Delta G_2 = \int_{\lambda=0}^1 \left\langle \frac{\partial V_{coulomb}(\lambda)}{\partial \lambda} \right\rangle d\lambda + \int_{\lambda=0}^1 \left\langle \frac{\partial V_{morse}(\lambda)}{\partial \lambda} \right\rangle d\lambda \quad (3.6)$$

The λ -dependence of $V_{coulomb}$ is introduced as⁹⁰

$$V_{coulomb}(\lambda) = \frac{1}{4\pi\epsilon\epsilon_0 r_{ij}} \left[(1-\lambda)q_i^A q_j^A + \lambda q_i^B q_j^B \right]. \quad (3.7)$$

For total charging of the ion, $q_i^A = 0$ and $q_i^B = Q_{ion}$. The λ -dependence of V_{morse} is

$$V_{Morse}(r_{ij}, \lambda) = \begin{cases} \lambda D_{e_{ij}} \left[e^{\lambda \alpha(r_{ij}) \left(1 - \frac{r_{ij}}{R_{\lambda_{ij}}}\right)} - 2e^{\frac{\lambda \alpha(r_{ij})}{2} \left(1 - \frac{r_{ij}}{R_{\lambda_{ij}}}\right)} \right] & \text{if } r_{ij} \leq R_{\lambda_{ij}} \quad (3.8a) \\ \lambda D_{e_{ij}} \left[e^{\lambda \beta(r_{ij}) \left(1 - \frac{r_{ij}}{R_{\lambda_{ij}}}\right)} - 2e^{\frac{\lambda \beta(r_{ij})}{2} \left(1 - \frac{r_{ij}}{R_{\lambda_{ij}}}\right)} \right] & \text{if } r_{ij} > R_{\lambda_{ij}} \quad (3.8b) \end{cases}$$

where

$$\begin{aligned} \alpha(r_{ij}) &= \alpha_{L_{ij}} + (\alpha_{0_{ij}} - \alpha_{L_{ij}}) \left(\frac{R_{\lambda_{ij}} - r_{ij}}{R_{\lambda_{ij}}} \right) \\ \beta(r_{ij}) &= \beta_{R_{ij}} + (\beta_{c_{ij}} - \beta_{R_{ij}}) \left(\frac{r_{ij} - R_{\lambda_{ij}}}{r_c - R_{\lambda_{ij}}} \right) \end{aligned}$$

where the equilibrium separation ($R_{\lambda_{ij}}$) is given by

$$R_{\lambda_{ij}} = R \left(1 - \frac{1}{2} * (1.0 - \lambda) \right). \quad (3.9)$$

Gaussian quadrature with 8 points were used for the evaluation of each integral in equation (3.6).

See below for simulation details. Targets for optimization are shown in table 3.5.

Radial Distribution Function: The comparison data for RDFs was generated by simulating 68 Na-Cl ion pairs with a solvent consisting of 38020 water molecules for a molality of 0.1 mol kg⁻¹ and 170 pairs with 38420 waters for 0.25 mol kg⁻¹. The SPC/E model for water was used, along with ion-ion and ion-water interactions from Joung and Cheatham.⁹¹ The radial distribution function (RDF), both for atomistic and coarse-grained systems, was calculated using the built in GROMACS utilities.

The optimization procedure involves many target function evaluations (see figure 3.4 and figure 3.7). These evaluations, at a point in parameter space, involve the execution of two MD simulations for the water-water optimization and 32 simulations for the water-ion phase. This places practical limits on the size and length of simulations used for optimization. Systems used for

determination of diffusion coefficient, density, and electric field permittivity in pure water contained 512 CG water beads, corresponding to 2048 water molecules. This system is denoted as S_1 , and was simulated for 4 ns with a timestep of 20 fs. Due to large fluctuations, the calculation of surface tension requires a larger system: 4096 CG beads, implying 16,384 water molecules (system S_2). System S_2 was constructed, at every evaluation of the target function, with a slab near the average density computed from S_1 in contact with vacuum.⁹² System S_2 was simulated under the constant number, volume, and temperature (NVT) ensemble for 4 ns.

Four systems were constructed for the optimization of water-ion and ion-ion interactions. Two systems of NaCl in an aqueous solutions were constructed. The first consisted of 68 pairs of ion beads and 9369 water beads for an effective 68 ion pairs and 38020 water molecules and a molality of 0.1 mol kg^{-1} . The second contained 170 pairs of ion beads and 9265 water beads (effectively 170 ion pairs and 37740 water molecules) for a molality of 0.25 mol kg^{-1} . Additionally, two systems, with one ion bead and 511 CG water beads were constructed for the calculation of solvation free energy of sodium and chloride at infinite dilution. All water-ion systems began from thermalized water systems, generated using the optimal parameters in table 3.1, with ion beads replacing water beads. The NaCl systems were simulated for 4 ns with a 20 fs timestep, with the final 2 ns used for analysis. Single ion systems for the calculation of $\partial V/\partial \lambda$ were run for 2 ns each. The calculation of $\partial V_{\text{coulomb}}/\partial \lambda$ used internal GROMACS methods. The determination of $\partial V_{\text{morse}}/\partial \lambda$ required a simulation of the system using $V_{\text{morse}}(\lambda)$ for each λ followed by a recalculation of energies at each step in the resulting trajectory using interaction tables containing $\partial V_{\text{morse}}/\partial \lambda$.

All simulations were performed using the GROMACS software.⁹⁰ Electrostatic interactions were calculated using the particle mesh Ewald method for the long range (beyond 1.6 nm) contribution. The modified Morse potential interactions were computed using lookup tables, with a cutoff of 1.6 nm.

3.5 Optimization

3.5.1 Water-Water

The first step in optimization is the construction of an initial simplex. In this work, optimization started with a simplex consisting of random points within the domain of parameter space which

satisfies constraints. The parameter space was bounded by minimum and maximum values for each parameter. This simplex was optimized until convergence. In order to further improve the accuracy of the final point in the case of water–water interactions, the optimization was restarted with an initial simplex consisting of the converged point and random points around it. This procedure was then iterated two more times, with the final optimization resulting in the point presented in table 3.1.

Figure 3.4 shows the mean and rms target function value vs. Nelder-Mead step for the final optimization iteration. Due to roughness of the hypersurface generated by the target function, the simplex may occasionally include a vertex with abnormally high value, as observed in the mean and rms of the second shrink step of figures 3.4. Figure 3.5 shows the evolution of parameter values

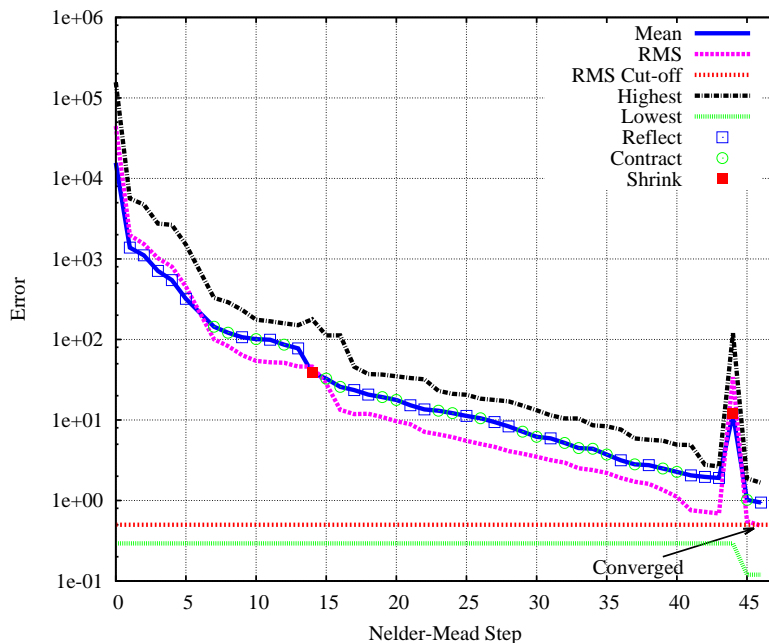


Figure 3.4: Water–Water parameter Optimization results: Mean error, rms error, and rms cutoff as a function of Nelder-Mead step. Reflection(empty squares), contraction (empty circles) and shrink (filled squares) steps labeled at the corresponding mean error point.

over the final run, while figure 3.6 shows the change in computed physical properties. The figures demonstrate the convergence of both the data and parameter values. Further, figure 3.5 demonstrates that although parameters did encounter the boundaries over the course of the optimization, the final simplex converged away from the boundary. Thus the parameter value boundaries chosen were not unreasonable.

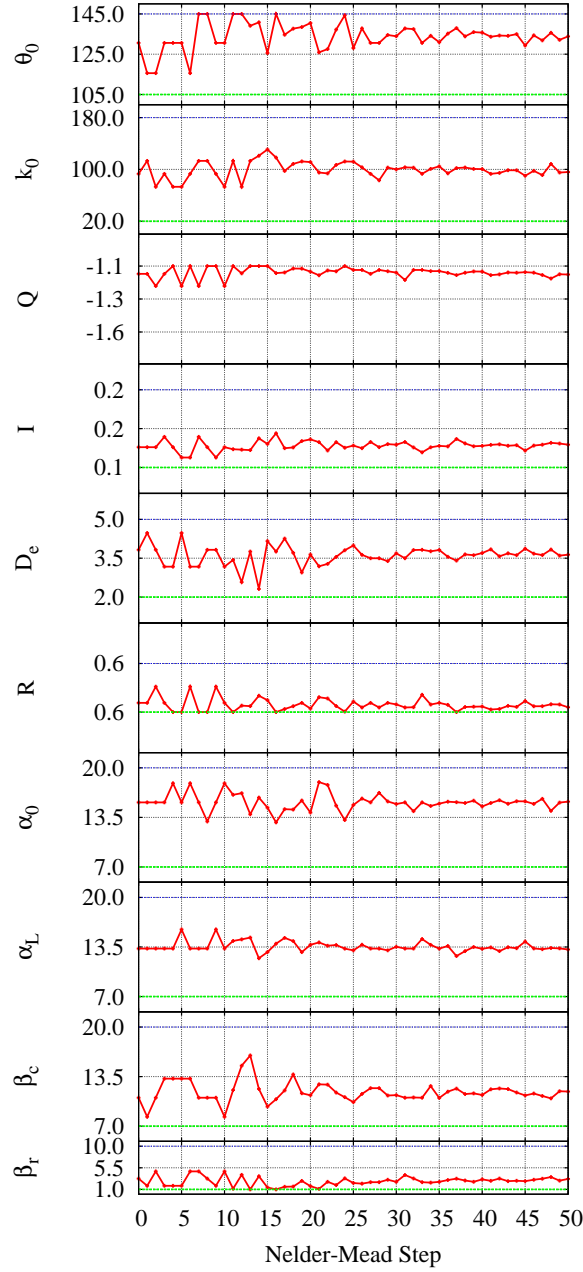


Figure 3.5: Water optimization parameter trajectory. Variation in parameters as function of Nelder-Mead step. Dashed lines indicate minimum and maximum values.

The final set of parameters was chosen as the point in the final converged simplex with lowest target function value (see table 3.1). The decoupling of repulsive and attractive parameters, the added distance dependence of the exponential factors and the added electrostatic interaction have allowed for a decrease in the equilibrium distance, while maintaining the experimental density. The large differences between exponential parameters in the new and original models hint that the

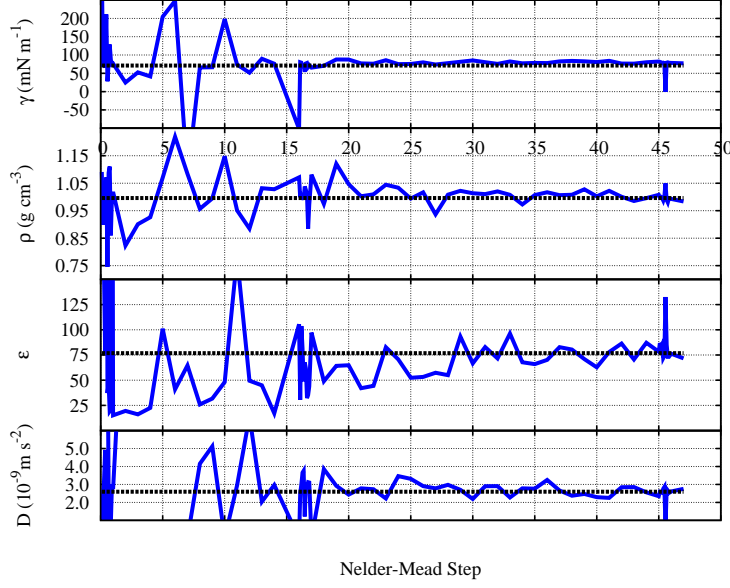


Figure 3.6: Optimization results: Training data (Density (ρ), electric field permittivity (ϵ), diffusion coefficient (D) and surface tension (γ))(solid) as a function of Nelder-Mead step compared with target values (dashed).

Table 3.1: Optimal Parameters: Set of final parameters from Nelder-Mead optimization compared with results from the previous model(CSJ).

| Parameter | CSJ Value ³² | Optimized Value (this work) | Minimum Value | Maximum Value |
|-------------------------------|-------------------------|--------------------------------|---------------|---------------|
| D_e (kJ mol ⁻¹) | 3.4 | 3.742 | 2.0 | 5.0 |
| R (nm) | 0.629 | 0.560 | 0.55 | 0.61 |
| α_0 | 7 | 15.545 | 7.0 | 20.0 |
| α_L | 7 | 13.323 | 7.0 | 20.0 |
| β_c | 7 | 10.976 | 7.0 | 20.0 |
| β_R | 7 | 3.198 | 1.0 | 7.0 |
| θ_0 | — | 131.406 | 105.0 | 145.0 |
| K_0 (kJ mol ⁻¹) | — | 93.918 | 20.0 | 180.0 |
| Q (e) | — | -1.126 | -1.6 | -1.06 |
| I (nm) | — | 0.141 | 0.1 | 0.25 |

extensions have added important flexibility. The simulation results corresponding to the final set of parameters are shown in table 3.2.

3.5.2 Water-Ion

The parameter space for water-ion and ion-ion interaction optimization was bounded by minimum/maximum values as shown in table 3.3. Additionally, parameters $\beta_{R_{ij}}$ were constrained to

Table 3.2: Training data: Results for simulations using the optimized water–water parameters compared with experimental target data.

| Property | Experiment | Training result |
|---|---------------------|-----------------|
| ρ (g cm ⁻³) | 0.996 ⁹³ | 0.993 |
| D (10 ⁻⁹ m s ⁻²) | 2.597 ⁹⁴ | 2.61 |
| ϵ | 76.8 ⁹⁵ | 76.24 |
| γ (mN m ⁻¹) | 71.2 ⁹⁶ | 78.43 |

be less than or equal to the associated value of $\beta_{c_{ij}}$. Optimization results, with mean and rms tar-

Table 3.3: CG water–ion parameter ranges, which in part define the allowable space for optimization.

| Parameter | Minimum value | Maximum value |
|--------------------|---------------|---------------|
| Ion charge (Q) | 0.10 | 1.00 |
| D_e | 0.05 | 5.0 |
| R | 0.3 | 0.9 |
| α_0 | 2.0 | 25.0 |
| α_L | 2.0 | 25.0 |
| β_c | 2.0 | 25.0 |
| β_R | 1.0 | 12.0 |

get function values, and Nelder-Mead transformation type are shown in figure 3.7. The optimized

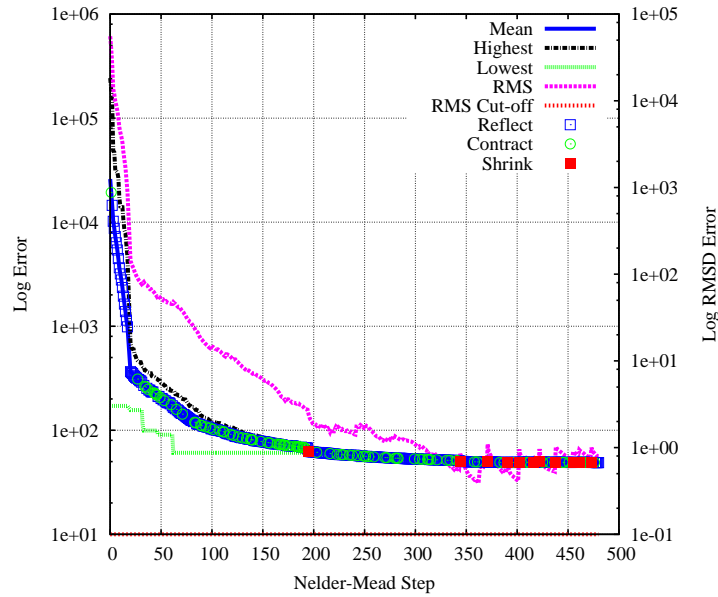


Figure 3.7: Water–Ion Optimization results: Mean error, rms error, and rms cutoff as a function of Nelder-Mead step. Reflection(empty squares), contraction (empty circles) and shrink (filled squares) steps labeled at the corresponding mean error point.

parameters produced by ParOpt are shown in table 3.4. The values from which the target function value associated with this point is determined are shown in table 3.5.

Table 3.4: Optimal Parameters: Set of final parameters from Nelder-Mead optimization of water-ion and ion-ion interactions. The final value for ion charge (Q_{ion}) was 0.998 e.

| Pair | D_e (kJ mol ⁻¹) | R (nm) | α_0 | α_L | β_c | β_R |
|----------|-------------------------------|----------|------------|------------|-----------|-----------|
| Na-Water | 3.6520 | 0.5608 | 22.5169 | 8.8733 | 12.5888 | 7.6724 |
| Cl-Water | 3.1037 | 0.6166 | 9.1807 | 12.4872 | 7.1951 | 5.4333 |
| Na-Na | 2.2902 | 0.5267 | 17.1609 | 9.2439 | 14.4505 | 3.6722 |
| Cl-Cl | 1.6594 | 0.8944 | 21.3582 | 2.3269 | 17.5957 | 8.0573 |
| Na-Cl | 3.2106 | 0.6187 | 20.6154 | 9.5633 | 16.8119 | 7.6055 |

Table 3.5: Training data: Results for the optimized point compared with experimental target data.

| Property | Experiment | Training result |
|--|----------------------|-----------------|
| ρ - 0.1 mol Kg ⁻¹ (g cm ⁻³) | 0.9997 ⁹⁷ | 0.9981 |
| ρ - 0.25 mol Kg ⁻¹ (g cm ⁻³) | 1.0057 ⁹⁷ | 1.0069 |
| ΔG - Na (kJ mol ⁻¹) | 365.0 ⁹⁸ | 341.3 |
| ΔG - Cl (kJ mol ⁻¹) | 340.0 ⁹⁸ | 330.1 |

3.6 Validation

When validating the results of an optimization procedure, the training data cannot be considered alone. The suitability of the force field will depend on matching data that the model was not explicitly trained to reproduce. Therefore, validation simulations were performed to verify the reliability of the optimized parameters. Since validation simulations are not performed iteratively, they are not subject to the same size constraints as during optimization.

3.6.1 Water-Water

All systems for validation of water-water parameters consisted of 110592 CG beads which corresponds to 442368 water molecules. Density, diffusion coefficient, relative permittivity, and surface tension were determined from the larger validation systems using the same methods as during optimization (see sec. 3.5). In addition, bulk thermodynamic quantities were also calculated (see table 3.6):

Coefficient of thermal expansion: Coefficient of thermal expansion, α , was calculated using the finite-difference method.⁹⁹

$$\alpha = \frac{1}{V} \left(\frac{\partial V}{\partial T} \right)_P \approx - \left(\frac{\ln \left(\frac{\rho_2}{\rho_1} \right)}{T_2 - T_1} \right)_P \quad (3.10)$$

Two constant number, pressure and temperature (NPT) ensemble simulations at temperatures $T_1 = 303.15$ K and $T_2 = 308.0$ K were performed. The ensemble averages of the resulting densities were used for ρ_1 and ρ_2 , respectively.

Isothermal compressibility: Isothermal compressibility κ_T was calculated using two methods:¹⁰⁰

(i) a finite difference method for NVT simulations at different densities (ρ_i):

$$\kappa_T = \frac{1}{\rho} \left(\frac{\partial \rho}{\partial P} \right) \approx \frac{\ln \left(\frac{\rho_2}{\rho_1} \right)}{\langle P_2 \rangle - \langle P_1 \rangle} \quad (3.11)$$

and (ii) volume fluctuations in an NPT ensemble simulation.

$$\kappa_T = \frac{\Delta V^2}{\langle V \rangle k_B T} \quad (3.12)$$

where P_i is the pressure on system i and ΔV^2 the variance of the volume. For eq. 3.11, densities used were $\rho_1 = 0.993$ g cm⁻³ and $\rho_2 = 0.963$ g cm⁻³.

Enthalpy of vaporization: Enthalpy of vaporization was calculated from the interaction energy between CG beads (V_{Inter}) and the intermolecular interactions that occur within the bead $V_S = 128.5$ kJ mol⁻¹ determined from quantum mechanical calculations of the water tetramer binding energy.^{32,101}

$$\Delta H_{\text{vap}} = - \frac{V_{\text{Inter}} + V_S}{4} + RT \quad (3.13)$$

where the factor of 4 is due to the level of CG.

The average dipole moment of an individual bead was calculated for the MMCG water model. Since a CG bead represents four water molecules, the total dipole moment for an atomistically detailed SPC/E water molecule along with its three nearest neighbors was computed for comparison. The MMCG model yields an average dipole moment of 4.1 ± 0.25 D, whereas The four water cluster

dipole moment from the SPC/E water model was 5.3 ± 1.0 D, while density function theory (DFT) calculations for an isolated cluster produce no dipole moment.¹⁰² The effective polarizability (α_0) for a CG bead was calculated using eq. 3.14.¹⁰³ The MMCG model yields a value of 63 a.u. compared with a value of 39.33 a.u. given by DFT for an isolated 4-water cluster.¹⁰⁴ The disparity in these values is most likely a result of differences between clusters in the bulk and in isolation.

$$\alpha_0 = \frac{(Q/2)^2 I^2}{K_0} \quad (3.14)$$

Table 3.6: Validation data: Comparing Experimental data, an existing CG model (Polarizable MARTINI), an atomistic model (SPC/E) and the previous model (CSJ) to this work (MMCG). Quantities computed are density ρ , diffusion coefficient D , dielectric permittivity ϵ , surface tension γ , coefficient of thermal expansion α , isothermal compressibility κ_T , and enthalpy of vaporization h_{vap} . Values computed both during optimization and for validation differ due to size differences between simulated systems.

| Property | Experiment | Polarizable MARTINI ⁷⁷ | SPC/E | CSJ ³² | MMCG (this work) |
|---|----------------------|--------------------------------------|--------------------------------|-------------------|---------------------|
| ρ (g cm ⁻³) | 0.996 ⁹³ | 1.043 | 0.998 ⁶⁶ | 0.998 | 0.993 |
| D (10 ⁻⁹ m s ⁻²) | 2.597 ⁹⁴ | 2.5 | 2.5 ⁶⁶ | 4.3 | 3.07 |
| ϵ | 76.8 ⁹⁵ | 75.6 | 70.7 (at 298 K) ¹⁰⁵ | | 74.17 |
| γ (mN m ⁻¹) | 71.2 ⁹⁶ | 30.5 | | 71.0 | 78.73 |
| α (10 ⁴ K ⁻¹) | 3.03 ¹⁰⁶ | | | | 3.445 |
| κ_T (10 ⁻⁶ bar ⁻¹)(NPT) | 44.75 ¹⁰⁷ | | 34.07 ¹⁰⁰ | 170.0 | 68.95 |
| κ_T (10 ⁻⁶ bar ⁻¹)(NVT) | 44.75 ¹⁰⁷ | | 41.41 ¹⁰⁰ | | 57.16 |
| h_{vap} (kJ mol ⁻¹) | 44.0 ¹⁰⁸ | | | 38.4 | 48.52 |

3.6.2 Water-Ion

For the validation of water-ion and ion-ion parameters, systems of differing NaCl pair concentrations were simulated using both the MMCG model and an atomistic model, with the SPC/E water model and the ion parameters from Joung et al..⁹¹ Concentrations simulated were 0.1, 0.25, 0.5, 0.75, 1.0, 2.0, 3.0, 4.0 and 5.0 mol kg⁻¹. Atomistic systems contained approximately 35,000 molecules and were simulated for 10 ns, with the final 8 ns used for analysis. CG systems consisted of approximately 33,000 beads, equivalent to 132,000 water molecules and were simulated for 20 ns with the final 10 ns used for analysis.

Density/Concentration profile: Coarse-grained models are not expected to be transferable

across wide ranges of temperatures, as the atomistic structures modeled change with temperature. Therefore, the density concentration/profile, shown in figure 3.8, instead of density/temperature profile was generated as part of the validation of the model's accuracy.

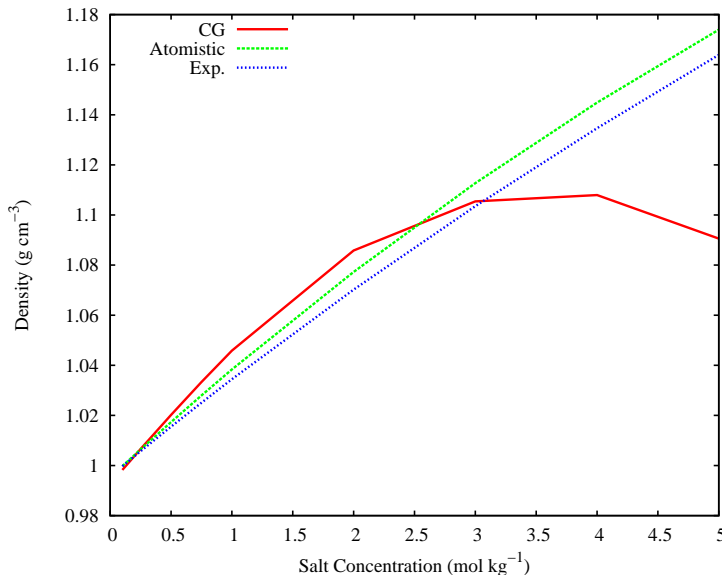


Figure 3.8: Density as a function of concentration for the MMCG model, atomistic model, and experimental.

RDF/Concentration: Along with densities at multiple concentrations, RDFs were also computed. These are compared to the corresponding atomistic RDFs. Figure 3.9 through 3.11 show the RDF for each pair of ion types at several concentrations.

Coordination: The coordination as a function of ion-ion distance ($G(r)$ in equation (3.15)) is determined from the cumulative pair-pair distribution function $N(r)$. The function $N(r)$ is determined from the number density in a shell of thickness dr at distance r . Figure 3.9 through 3.11 show the coordination function for each pair of ion types at several concentrations.

$$G(r) = \int_0^r N(r') dr' \quad (3.15)$$

Poisson-Boltzmann: In the case of counter ions interacting with a charge surface, with out the

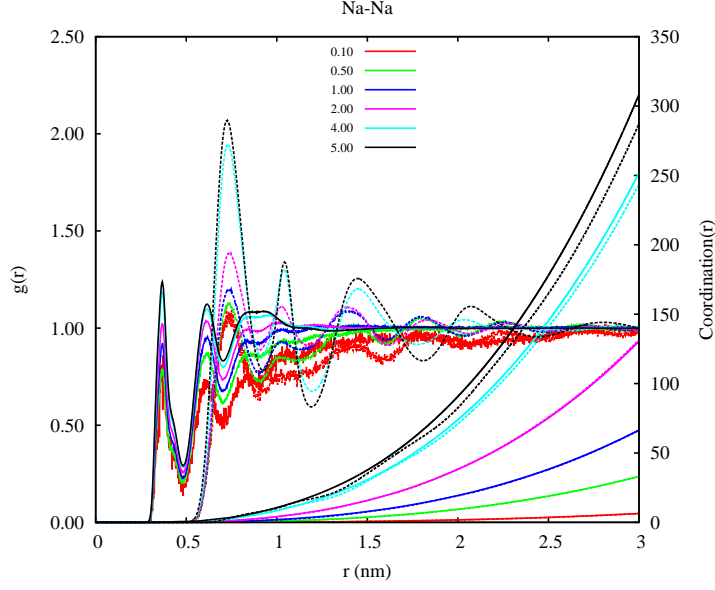


Figure 3.9: Na-Na coordination and radial distribution functions for NaCl concentrations 0.10, 0.50, 1.00, 2.00, 4.00, and 5.00 mol kg⁻¹, with the color indicating the concentration. Atomistic MD simulation results are shown with solid lines, MMCG simulation results are shown with dashed lines.

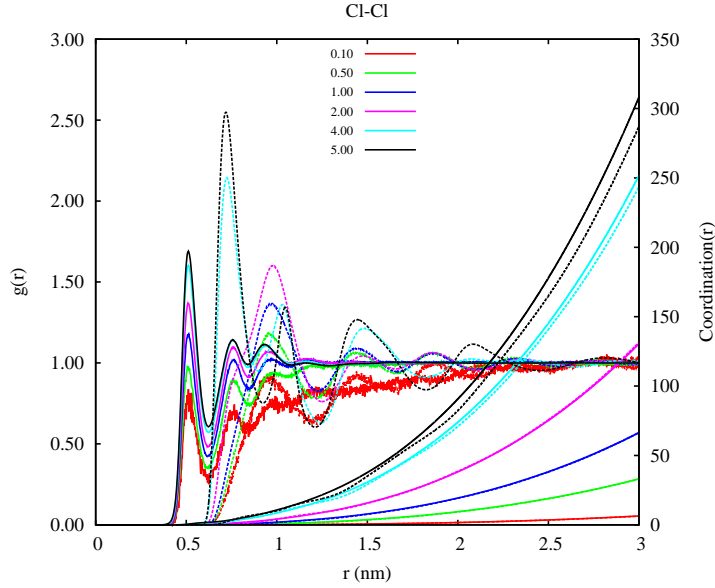


Figure 3.10: Cl-Cl coordination and radial distribution functions for NaCl concentrations 0.10, 0.50, 1.00, 2.00, 4.00, and 5.00 mol kg⁻¹, with the color indicating the concentration. Atomistic MD simulation results are shown with solid lines, MMCG simulation results are shown with dashed lines.

presence of an electrolyte, the distribution of ions is given by

$$\rho(z) = \frac{\rho(0)}{\cos^2(Kz)} \quad (3.16)$$

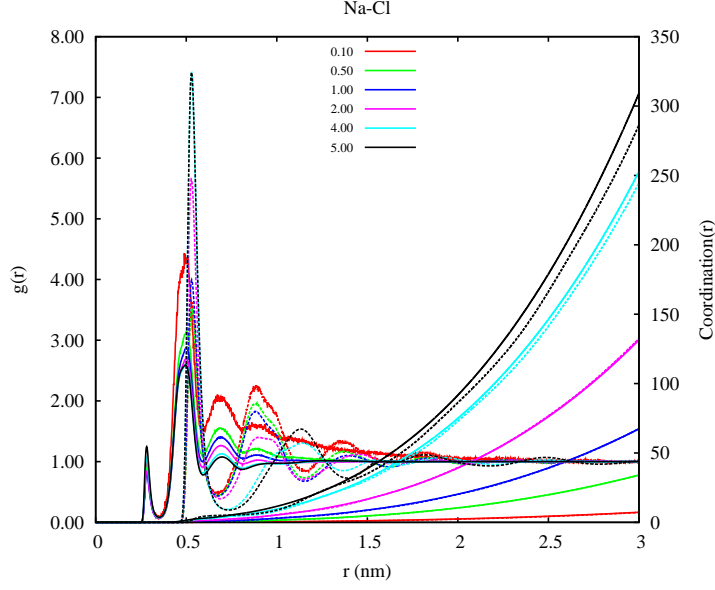


Figure 3.11: Na-Cl coordination and radial distribution functions for NaCl concentrations 0.10, 0.50, 1.00, 2.00, 4.00, and 5.00 mol kg⁻¹, with the color indicating the concentration. Atomistic MD simulation results are shown with solid lines, MMCG simulation results are shown with dashed lines.

where K is the characteristic length of the ion in the solvent and $\rho(z)$ is the ionic concentration as a function of distance from the center of the surfaces (z).¹⁷ Kalcher et al. performed a similar analysis in the case of atomistic MD simulation.⁶⁹ The characteristic length predicted by the Poisson-Boltzmann equation for a monovalent ion is given by

$$K^2 = \frac{e^2 \rho(0)}{2\epsilon\epsilon_0 k_B T} \quad (3.17)$$

where ϵ is the permittivity of the solvent, ϵ_0 is the permittivity of free space.¹⁷ For both CG and atomistic ions, systems were constructed with a single ion type in water surrounded by charged plates and vacuum. The concentration profile was then determined and compared to the profile predicted by the Poisson-Boltzmann equation. Concentration as a function of distance from the center of the interfaces is shown for the atomistic system (figure 3.12) and the CG system (figure 3.13) along with the fitted function (equation (3.16)). The Poisson-Boltzmann predicted value for both systems is $K_{CG} = 0.178 \text{ nm}^{-1}$. Values determined by fitting equation (3.16) were $K_{CG} = 0.171 \text{ nm}^{-1}$ and $K_{atomistic} = 0.187 \text{ nm}^{-1}$.

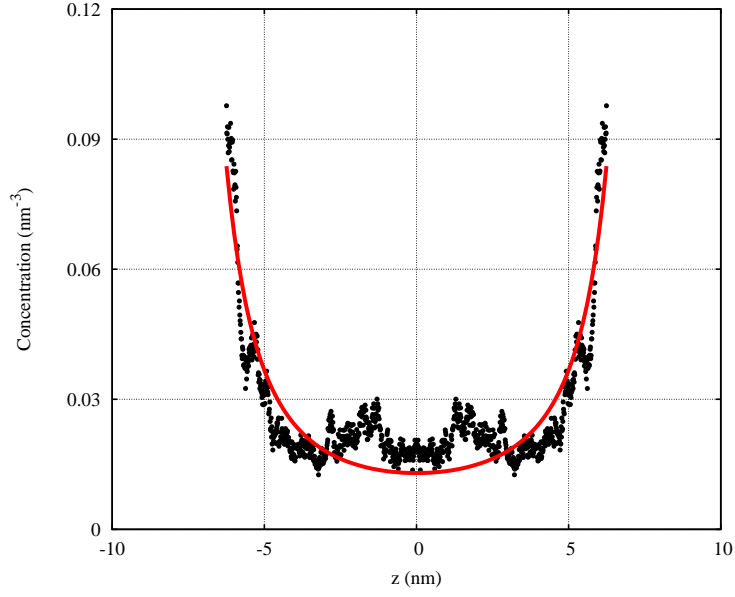


Figure 3.12: The concentration of sodium ions as a function of distance from the center of oppositely charged surfaces for an all atom simulation. A fit to the Poisson-Boltzmann equation predicted distribution (equation (3.16)) is also shown.

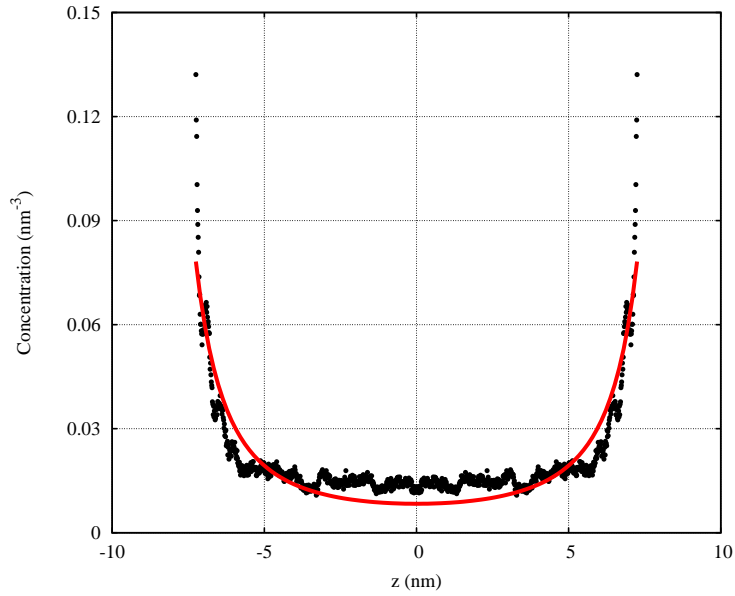


Figure 3.13: The concentration of sodium ions as a function of distance from the center of oppositely charged surfaces for the MMCG model simulation. A fit to the Poisson-Boltzmann equation predicted distribution (equation (3.16)) is also shown.

3.7 Conclusion

The general ParOpt framework has been applied to the optimization of a polarizable CG water-ion model. The non-bonded, non-Coulombic interactions in the model are given by a modified Morse potential. Parameters were optimized via the Nelder-Mead algorithm. The model has more flexibility in the functional form than most CG models, allowing it to match a wide range of experimental properties bulk properties of water (see table 3.6), as well as important characteristics of ion behavior. There are fundamental limitations to the explanatory power of CG water models.

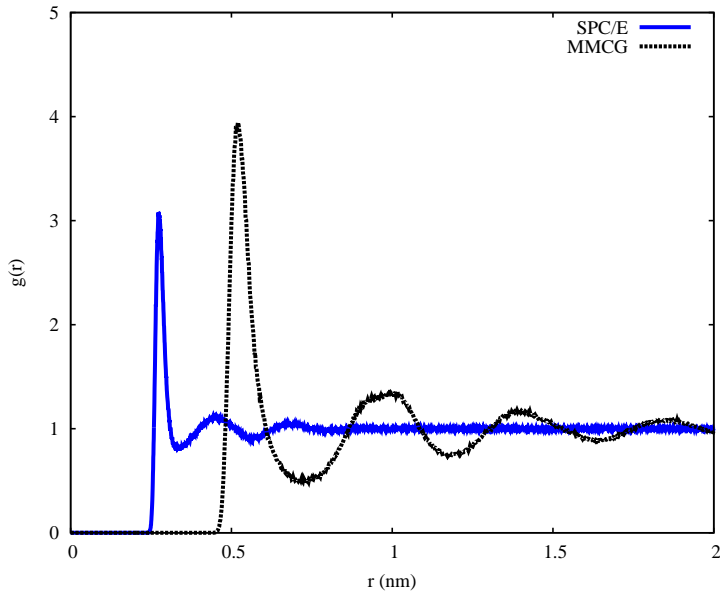


Figure 3.14: RDF: Comparison between $g(r)$ for atomistic SPC/E water model (blue solid line) and the MMCG water model (black dashed line). The CG model has higher long range coordination compared to the atomistic model.

RDFs are notoriously difficult for to match. An artifact common to most CG models is the long-range correlation between CG water beads not seen in experiment or atomistic models. Figure 3.14 shows that the current model suffers from that same limitation. Likely as a result of the high degree of inter-bead correlation at long distances, the model has the propensity to spontaneously solidify at room temperature, a weakness found in other CG water models.³⁰ Spontaneous freezing can be avoided by periodically assigning random velocities chosen from a Maxwell distribution, a method similar to one suggested by Harvey et al. to avoid the over population of low frequency modes of a molecular dynamics system.¹⁰⁹ Since dynamic and structural properties of water clusters vary with

temperature, the internal states of CG beads would as well. Thus, CG models cannot necessarily be assumed to transfer to different temperatures. We believe that the interaction parameters for CG models would be more accurately treated as temperature dependent. The set of potential parameters has been tuned for one temperature, and is not expected to accurately simulate behavior far from the target temperature or in thermodynamic ensembles without constant temperature control.

As figure 3.8 demonstrates, the model is accurate for low concentrations, but at higher concentrations, the ions become over-coordinated, resulting in an overly dense packing of beads. Usage of the model therefore, should be restricted to physiological ion concentrations.

The predictions of the model agree well with experiment for structural, dynamic, and bulk properties. The diffusion constant shows that the model has accurate dynamics. The dielectric constant indicates validity of the parameters that determine electrostatics and polarizability. Density ensures the correct spatial scale of the CG beads. Accurate surface tension indicates an accurate representation of the strength of the interaction between beads. Ion solvation free energies indicate the accuracy of the energies between ions and bulk water. Coordination numbers and RDFs indicate that the model is accurate at long distances, but that short range accuracy is limited by the scale of the coarse-graining, as is expected. The thermodynamic quantities predicted by the model are in good agreement with those of experiment, validating the fitness of the parameters.

Chapter 4

Lipid Bilayer Structure Determination

4.1 A Note to the Reader

This chapter contains previously published material, with permission, from J. C. Fogarty, M. Arjunwadkar, S. A. Pandit, and J. Pan. “Atomically Detailed Lipid Bilayer Models for the Interpretation of Small Angle Neutron and X-ray Scattering Data”. In press. 2014

4.2 Introduction

Cell membranes participate in a host of vital biological functions. In order to understand the role of lipid membrane components in these functions, it is necessary to resolve membrane structures formed by single lipid species. This is achieved by studying model lipid bilayers with simple compositions. Thermal fluctuations of disordered lipid bilayers pose significant difficulty in determining atomic positions. Scattering techniques such as small angle X-ray (SAXS) and neutron (SANS) scattering have been traditionally used to explore structures of lipid bilayers. The utility of the SAXS and SANS methods is primarily due to their sensitivity to heterogeneous electron and neutron scattering length distributions. In particular, X-ray is scattered most strongly by electron dense moieties. Therefore, it is capable of localizing phospholipid phosphate groups. Due to the remarkable difference of neutron scattering length between hydrogen and other atoms including deuterium, neutron scattering is most suited for localization of the hydrogen deficient glycerol/carbonyl backbone. Since X-ray and neutron scattering profiles are dominated by different molecular features,

more detailed information can be inferred by using their combination.^{111–119} This hybrid approach (i.e. the simultaneous fitting of SAXS and SANS data from equivalent lipid bilayers) along with the scattering density profile (SDP) model has been applied to successfully extract structure of many commonly seen phosphatidylcholine (PC), phosphatidylglycerol (PG) and phosphatidylserine (PS) lipid bilayers.^{114–118} The essence of the SDP model is to partition a lipid bilayer into several components each of whose volume probabilities and electron and neutron scattering length densities are described by one mathematical function (e.g., a Gaussian or an error function). The SDP model relies on proper grouping of lipid atoms, and the dimensionality that the model can handle is limited by the fitting procedure. Furthermore, many parameters need to be constrained to avoid instabilities in fitting. For example, in constraint-free analysis, the width of the error function describing hydrocarbon chains, and thus the hydrocarbon core thickness, tends to bloat. Finally, the SDP model relies on nonlinear least squares fitting methods, such as Levenberg-Marquardt. Levenberg-Marquardt can have slow convergence for problems with large residuals, which can result from noisy data.³⁶ Additionally, nonlinear least squares methods may be unsuitable for higher dimensions due to the growth of computational time with the size of the parameter space.¹²⁰ In the interest of dimensional reduction, the SDP model combines groups of atoms. The increased dimensionality for bilayer mixtures poses a non-trivial and difficult challenge for the SDP model. A more recent usage of the SDP model performed parameter optimization via a genetic algorithm which offers advantages (primarily the avoidance of local minima) over the more commonly used Levenberg-Marquardt.¹¹⁹

In general, scattering techniques provide reciprocal space structure, necessitating a transformation into real space which cannot be performed directly (e.g., via an inverse Fourier transformation) because of the unavailability of the full spectrum and phase information. The framework of inverse problems provides a systematic methodology for determining the underlying physical properties of a system by constructing and fine-tuning a mathematical model. Obtaining structural information from scattering data requires solving the inverse problem posed by

$$G(m) = d \tag{4.1}$$

where d is a set of experimentally observable data and m is the underlying physical model.¹²¹ The

operator G performs a transformation which takes a model as input and yields a predicted set of data (d') equivalent to the experimental data d . The statement of the inverse problem is to determine the model (m_s), via an optimization procedure, such that $\|d' - d\|$ (where $\|\cdot\|$ indicates a norm defined on the data space) is minimized. In the case of scattering, G is the Fourier transform of the density profiles associated with the model, d is the set of form factors determined from scattering intensity measurements, and m is a model of the underlying atomic positions. Solving the inverse problem can be complicated by several factors. The solution (m_s) may not exist, i.e. the model may be unable to physically represent the data. Conversely, m_s may be an infinite family of acceptable solutions to the problem. This is the case where the number of degrees of freedom of the model exceeds that of the data set. It may also be the case that the problem suffers from instability, i.e. the solution m_s is extremely sensitive to small changes in d . Since d often has a noise component, this poses a significant challenge to determining m_s . These difficulties can be reduced via regularization procedures which rely on additional physical knowledge of the system (independent of the target data) to reduce the complexity of the problem.

In this chapter, a new atomic density profile (ADP) model for the determination of lipid bilayer structures is presented. The ADP model, which makes minimal assumptions regarding the atomic arrangement within a lipid molecule, is a significant generalization of the SDP model. The SDP model treats the hydrocarbon core with a error function, and uses a handful of Gaussian functions to describe the headgroup and backbone. The ADP model treats all atoms (with hydrogens summed into heavy atoms) as independent Gaussian functions. Sophisticated data analysis methodology and fitting procedures are used to systematically assess detailed lipid bilayer structures embedded in SAXS and SANS data. Specifically, an ensemble of optimal ADP models for each lipid bilayer is generated by solving the inverse problem using a simplex based optimization procedure. Ensemble averaging inspired by a Bayesian formulation of the problem yields detailed bilayer structures.

4.3 Model and Methods

Models for different lipid bilayer compositions are constructed as follows. Each atom in a lipid molecule, with hydrogens summed into their bonded heavy atoms, is assigned a probability density function. This function represents the distribution of the atom in the one dimensional projection of

the bilayer in a leaflet. Since the system is assumed to be homogeneous in the plane of the bilayer, the problem is effectively one dimensional (along bilayer normal). Additionally, the bilayers are assumed to be symmetric with respect to their centers, though the model can be easily extended to asymmetric bilayers. While only pure bilayers are considered, bilayers with arbitrary mixtures of lipids can be treated with the current method. Experimental data (d in equation (4.1)) are SAXS and SANS form factors along with the total lipid molecular volume. Models are regularized using known molecular topology and hydrophobic interactions. A fitness penalty (additional term in $\|d' - d\|$) for water probability within the hydrocarbon core takes the hydrophobic effect into account. A additional penalty on unphysical probability densities is also employed. Linear constraints on mean atomic positions eliminate models with unphysical bond lengths.

4.3.1 The Model (m)

The probability density is approximated of each atom by one normalized Gaussian function of the form:

$$g_i(z) = \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp \left[\frac{-(z - \mu_i)^2}{2\sigma_i^2} \right] \quad (4.2)$$

where i is the index of the atom, z the distance from bilayer center, and μ_i and σ_i^2 the mean and variance of the distribution respectively.

Both the electron and neutron scattering length densities ($\rho(z)$) for a hydrated lipid bilayer are then calculated by expressing the total density as the sum of lipid, including counter-ions when present ($\rho_{\text{lipid}}(z)$), and water ($\rho_{\text{water}}(z)$) contributions.

$$\rho(z) = \rho_{\text{lipid}}(z) + \rho_{\text{water}}(z) \quad (4.3)$$

The density profile for the lipid is determined from the probability densities of the atoms ($g_i(z)$).

$$\rho_{\text{lipid}}(z) = \sum_i w_{\alpha_i} \frac{g_i(z) dz}{V_{\text{slice}}} \quad (4.4)$$

where atom i is of type α_i (e.g., C, O, P or N), V_{slice} is the volume of a slice of thickness dz along the bilayer normal. The weight on atom i (w_{α_i}) is either the number of electrons or the neutron scattering length, depending on the desired density. The contribution from water molecules is

written as

$$\rho_{\text{water}}(z) = w_w \frac{p_w(z)dz}{V_{\text{slice}}} \quad (4.5)$$

where w_w is the corresponding weight for water and $p_w(z)dz$ is the number of water molecules between z and $z + dz$, which is unknown. Therefore, from equation (4.3) the density is given by

$$\rho(z) = \sum_i \left[g_i(z; \mu_i, \sigma_i) dz \frac{w_{\alpha_i}}{V_{\text{slice}}} \right] + p_w(z) dz \frac{w_w}{V_{\text{slice}}}. \quad (4.6)$$

The reference point is then set by subtracting the corresponding density for bulk water:

$$\rho^*(z) = \rho(z) - \rho_{\text{bulk}}(z) \quad (4.7)$$

$$= \sum_i \left[g_i(z; \mu_i, \sigma_i) dz \frac{w_{\alpha_i}}{V_{\text{slice}}} \right] + (p_w(z) dz - P_w) \frac{w_w}{V_{\text{slice}}} \quad (4.8)$$

where P_w is the number of water molecules in a slice containing only bulk water. This is given by $P_w = V_{\text{slice}}/v_{\text{bulk}}$ where v_{bulk} is the partial molecular volume of bulk water. The value for v_{bulk} was assumed to be 30.0 \AA^3 based on the density of water at physiologic temperature. Therefore,

$$\rho^*(z) = \sum_i \left[g_i(z; \mu_i, \sigma_i) dz \frac{w_{\alpha_i}}{V_{\text{slice}}} \right] + \left(p_w(z) dz - \frac{V_{\text{slice}}}{v_{\text{bulk}}} \right) \frac{w_w}{V_{\text{slice}}}. \quad (4.9)$$

The number of water molecules in a slice is chosen, (similar to the SDP model¹¹⁴), such that the total molecular volume in a slab equals the slab volume:

$$V_{\text{water}}(z) = V_{\text{slice}} - V_{\text{lipid}}(z) \quad (4.10)$$

$$V_{\text{water}}(z) = v_w p_w(z) dz \quad (4.11)$$

$$V_{\text{lipid}}(z) = \sum_i v_{\alpha_i} g_i(z; \mu_i, \sigma_i) dz \quad (4.12)$$

$$p_w(z) dz = \frac{V_{\text{slice}}}{v_w} - \sum_i \frac{v_{\alpha_i}}{v_w} g_i(z; \mu_i, \sigma_i) dz, \quad (4.13)$$

where v_{α_i} is the partial volume for species i . Substituting equation (4.13) into equation (4.9) yields

$$\rho^*(z) = \sum_i \left[g_i(z; \mu_i, \sigma_i) dz \left(\frac{w_{\alpha_i}}{V_{\text{slice}}} - \frac{w_w v_{\alpha_i}}{v_w V_{\text{slice}}} \right) \right] + \frac{w_w}{v_w} - \frac{w_w}{v_{\text{bulk}}}. \quad (4.14)$$

Making the approximation that the molecular volume of interfacial water is similar to that of the bulk, i.e. $v_w \approx v_{\text{bulk}}$, leads to:

$$\rho^*(z) = \frac{dz}{v_w V_{\text{slice}}} \sum_i (v_w w_{\alpha_i} - v_{\alpha_i} w_w) g_i(z; \mu_i, \sigma_i) \quad (4.15)$$

Equation (4.15) yields $\rho^*(z)$ as the electron or neutron scattering length density, depending on the choice of weights. Weights for each species are given in table 4.1. Because of the arbitrary scale

Table 4.1: Weights for calculation of electron and neutron scattering length densities. Taking hydrogen-deuterium exchange into account, hydroxyl groups in PG and the amine groups in PS were assumed to have hydrogen to deuterium ratios equal to that of the solvent. All other groups are assumed to have neutron scattering cross sections that are not sensitive to D₂O concentration.

| Species | Electron Count | Neutron Scattering Cross-Section (fm) | | | |
|------------------------------|----------------|---------------------------------------|---------------------|---------------------|---------------------|
| | | 100%D ₂ O | 75%D ₂ O | 70%D ₂ O | 50%D ₂ O |
| C | 6 | 6.65 | — | — | — |
| Ch | 7 | 2.91 | — | — | — |
| Ch ₂ | 8 | -0.83 | — | — | — |
| Ch ₃ | 9 | -4.57 | — | — | — |
| N ⁺ | 6 | 9.40 | — | — | — |
| Nh ₃ ⁺ | 9 | 29.41 | 21.603 | 20.04 | 13.795 |
| Na ⁺ | 10 | 3.63 | — | — | — |
| P | 15 | 5.17 | — | — | — |
| O | 8 | 5.8 | — | — | — |
| O ⁻ | 9 | 5.8 | — | — | — |
| Oh | 9 | 12.47 | 9.868 | 9.347 | 7.265 |
| Water | 10 | 19.24 | 14.035 | 12.994 | 8.83 |

of the scattering intensity and the linearity of the cosine transform, the factor of $dz/(v_w V_{\text{slice}})$ is omitted from equation (4.15) during optimization. The densities are computed for one leaflet of the lipid bilayer, then symmetrized to produce even functions. All functions were treated as discrete, with $dz = 0.05$ Å as the granularity for computation of the density functions $\rho^*(z)$.

The set of parameters $\{\mu_i, \sigma_i, v_{\alpha_i}\}$ determines the space for optimization. With three parameters per atom, the dimension of the parameter space is ~ 160 . In order to reduce the size of the parameter space, head group atoms (including counter-ions when present) were assumed to have identical partial volumes. Thus, the set of partial atomic volumes for tuning was $\{v_{\text{Ch}_3}, v_{\text{Ch}_2}, v_{\text{Ch}}, v_{\text{head}}\}$. Additionally, carbon chain atoms of the same type were assumed to have identical σ values ($\sigma_{\text{Ch}_3}, \sigma_{\text{Ch}_2}, \sigma_{\text{Ch}}$). Such approximations reduce parameter set sizes from ~ 160 to ~ 115 .

4.3.2 The Data (d)

Different contrast SAXS and SANS form factors for phosphatidylcholine (PC), phosphatidylglycerol (PG), and phosphatidylserine (PS) were obtained from previous works.^{114–116,118} Data points which exhibit no discernable scattering signal compared to background noise were truncated. Due to the unavailability of experimental data uncertainties, error bars ($\Delta F_e(q_i)$ in equation (4.17)) were assigned based on the magnitude of scattering signal, as a percentage of the maximum intensity. For the SAXS data, the first lobe is assigned the smallest uncertainty(2%), followed by the second (4%) and third (8%) lobes, while the fourth lobe, when present, is assigned the largest uncertainty (16%). Similar assignments were applied to the SANS data, with two regions defined(4% and 16%). It has been pointed out that the amino NH_3^+ group in PS is capable of fast hydrogen-deuterium exchange with the surrounding water.¹¹⁸ Thus, the NH_3^+ hydrogens solvated by water with x mole fraction of D_2O are replaced by $3x$ deuteriums and $3(1-x)$ hydrogens. This has a significant effect on the neutron scattering length profile. Similar exchange also occurs in PG lipids which possess two hydroxyl groups. Although it was not considered in the published PG lipid bilayers using the SDP model,¹¹⁶ the hydrogen-deuterium exchange of PG hydroxyls is taken into account in the ADP model. Lipid volumes, determined using density measurements, were also included in the data sets for optimization. Lipid volumes for parameter tuning were 1303 \AA^3 for DOPC,¹¹⁴ 1256 \AA^3 for POPC,¹¹⁵ 1228.5 \AA^3 for DPPC,¹¹⁴ 1208.7 \AA^3 for POPG,¹¹⁶ and 1198.5 \AA^3 for POPS.¹¹⁸

4.3.3 The Transformation (G) and the Comparison ($\|d' - d\|$)

To compute the model predicted form factor $F_m(q)$, assuming a symmetric bilayer, the cosine transform was performed.

$$F_m(q) = \int_{-\infty}^{\infty} \rho(z) \cos(qz) dz \quad (4.16)$$

The degree of agreement between $F_m(q)$ and the experimental form factor ($F_e(q)$) was calculated using

$$\chi = \sqrt{\frac{\sum_{i=1}^{N_q} \left(\frac{k_e F_e(q_i) - F_m(q_i)}{k_e \Delta F_e(q_i)} \right)^2}{N_q - 1}}. \quad (4.17)$$

where N_q is the number of q values considered and $\Delta F_e(q)$ is the error bar on the experimental value.¹¹⁶ The q -samples for the model generated data were chosen to match the corresponding

Table 4.2: Constraints on partial atomic volumes. The approximation is that atoms within each group (head, tail Ch, tail Ch₂, and tail Ch₃) have equal volume.

| Volume | Mini. Value (Å ³) | Max. Value (Å ³) |
|-------------------|-------------------------------|------------------------------|
| v_{head} | 12.0 | 21.0 |
| v_{Ch} | 19.0 | 24.0 |
| v_{Ch_2} | 25.0 | 30.0 |
| v_{Ch_3} | 51.0 | 56.0 |

experimental samples. The experimental form factors must be scaled in order to be comparable with the model data. The factor k_e ensures this appropriate scaling.¹¹⁶

$$k_e = \frac{\sum_i \frac{F_e(q_i)F_m(q_i)}{[\Delta F_e(q_i)]^2}}{\sum_i \frac{[F_e(q_i)]^2}{[\Delta F_e(q_i)]^2}} \quad (4.18)$$

The measure of difference (χ) between the model predicted and the experimental form factors was calculated for each set of X-ray data (χ_{elec1} and χ_{elec2} when two data sets were available) and neutron data (χ_{neut100} , χ_{neut75} , χ_{neut70} , χ_{neut50} when sets at 100, 75, 70, and 50 % D₂O concentrations were available).

Total lipid molecular volume was included as an additional contribution to the comparison between model and experiment via

$$\chi_{\text{volume}} = \frac{|(V_{\text{exp}} - \sum_i v_{\alpha_i})|}{V_{\text{exp}}}, \quad (4.19)$$

where V_{exp} are the experimental volumes.

4.3.4 The Regularization

Regularization was introduced in part via linear constraints on the parameter space. Based on molecular dynamics simulation, values for σ_i were restricted to the range [2.25, 3.75] Å.^{122–126} Partial volumes were constrained as listed in table 4.2.

Connectivity information was taken into account during optimization by applying relative constraints to all values of μ_i as follows. Atoms within the $sn-1$ chain were constrained to be within one bond length away from the preceding bonded atom equation (4.20), beginning with the terminal

methyl group ($i = 0$).

$$\mu_{i-1} \geq \mu_i \geq \mu_{i-1} + b \quad (4.20)$$

Atoms in the backbone and head group were constrained to be within one bond length of bonded atoms equation (4.21).

$$|\mu_i - \mu_k| \leq b \quad (4.21)$$

Atoms in the sn -2 chain were constrained similarly to the sn -1 chain equation (4.22), but relative to the backbone carbon ($i = 0$).

$$\mu_{i-1} - b \geq \mu_i \geq \mu_{i-1} \quad (4.22)$$

For optimization the bond length was set to $b = 2.0 \text{ \AA}$. Sodium counter ions were constrained to the range $[10, 35]$. In order to define a coordinate system, one atom must be chosen as the origin without loss of generality. Since the terminal methyl group of the sn -1 chain bilayer can be approximated as bilayer center, the corresponding value for μ was fixed at $z = 0$. The sn -2 terminal methyl group was not similarly restrained. The associated value for σ was also variable parameter in the optimization.

Regularization methods were also applied via the addition of penalties on model configurations which have unphysical properties. Models with a probability of finding any lipid atom (including counter-ions) in a slice between z and $z + dz$ ($p_{\text{lipid}}(z)$) exceeding unity were penalized by the introduction of the term

$$\chi_{\text{reg1}} = \frac{\int_{-\infty}^{\infty} H \{p_{\text{lipid}}(z) - 1\} [p_{\text{lipid}}(z) - 1] dz}{\int_{-\infty}^{\infty} H \{p_{\text{lipid}}(z) - 1\} dz}, \quad (4.23)$$

where $H(x)$ is the Heaviside step function and $p_{\text{lipid}}(0) = 1$ defines the reference point. This term (equation (4.23)) penalizes trial models which have regions of negative water probability, which is unphysical. The presence of water in the hydrocarbon core is unfavorable due to hydrophobic effects. This physical information is incorporated by introducing the term

$$\chi_{\text{reg2}} = \frac{1}{2\mu_{\text{cut}}} \left[\int_{-\mu_{\text{cut}}}^{\mu_{\text{cut}}} (1 - p_{\text{lipid}}(z))^2 dz \right]^{1/2} \quad (4.24)$$

where μ_{cut} is defined as the μ value for the seventh CH_2 relative to the carbonyl carbon on the $sn-1$ chain. This regularization term imposes a soft cutoff on the penetration of water into the hydrocarbon core, and is similar in principle to constraints on the hydrocarbon error function parameters in the SDP model.¹¹⁴ The penalty term in equation (4.24) puts a penalty on water inside of μ_{cut} , but does not impact water probability above the cutoff. The choice of carbon for μ_{cut} is set conservatively, such that small changes in its location have little effect on the optimization results. The cutoff location may need to be adjusted for different lipids and different phases.

4.4 Validation using Molecular Dynamics

As a test case of the ADP model and the optimization methods presented here, similar analysis with lipid bilayer form factors determined from molecular dynamics (MD) simulation as the target has been performed. In the case of MD, both the model m and the data d are known prior to optimization, allowing the direct comparison of ADP model results and the underlying configuration which generated the data. Therefore, solutions inverse problem (equation (4.1)) can be compared with the actual model (m) which gave rise to the data d . Therefore, tuning the ADP model to reproduce MD data provides a test of the reproducibility of G given m and d .

Molecular dynamics simulations were performed on a system consisting of 128 POPC molecules and 100 water molecules per lipid using GROMACS version 4.0.¹²⁷ The system was run under a constant temperature and pressure ensemble with a temperature of 303 K and a pressure of 1 bar for 200 ns with timestep of 2 fs. Long-range electrostatic interactions (beyond 0.95 nm) were computed using the Particle Mesh Ewald method. Lennard-Jones interactions were computed for all distances up to 1.8 nm. Interaction parameters were used from ffG43A1-S3.¹²⁸ Time averaged electron and neutron scattering length densities were extracted from the final 10 ns of the simulation trajectory using the GROMACS included analysis utilities. A cosine transformation (equation (4.16)) was performed to generate scattering form factors. Though only H_2O was used in the simulations, neutron scattering lengths corresponding to solvent D_2O concentrations of 100%, 70% and 50% were used in order to generate data similar to the experimental case. These form factors determined from MD simulation, along with the experimental value for total lipid volume, and regularization terms χ_{reg1} (equation (4.23)) and χ_{reg2} (equation (4.24)) were used to optimize the ADP model.

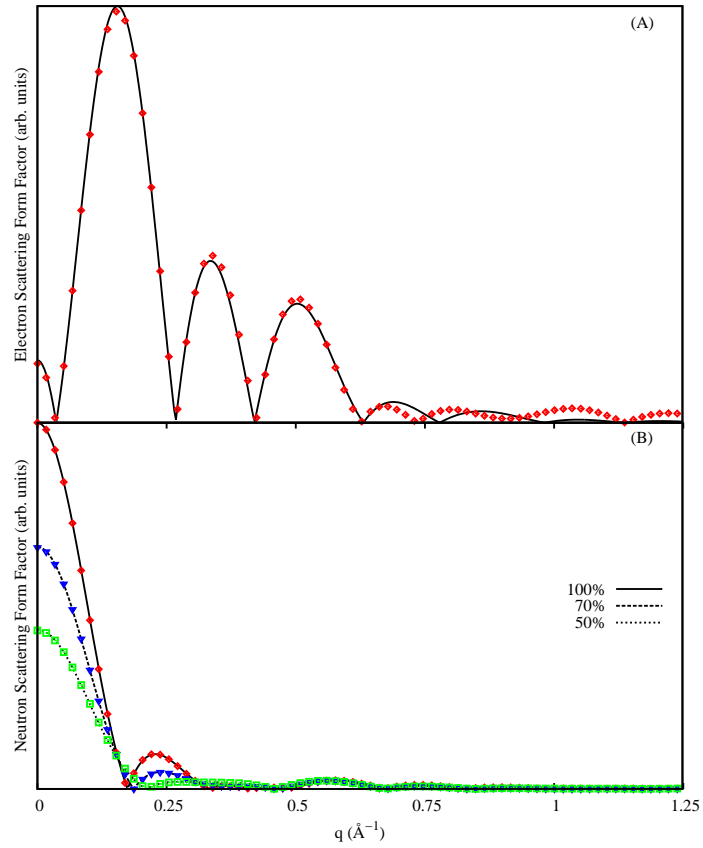


Figure 4.1: Electron (A) and neutron (B) scattering form factors determined from molecular dynamics simulation (points) and optimization of the ADP model (lines) with the MD form factors as targets.

Figure 4.1, which shows optimized ADP model form factors compared with the target MD form factors for one optimization. The ADP model is able to very accurately reproduce the scattering form factors even for large values of q despite all the simplifying assumptions. The ADP model and the optimization technique is therefore well suited to fit the scattering data and extract the structural properties. If the target data are generated with low error and high q values the model reproduction is near perfect (see figure 4.1 and figure 4.2). For real experimental systems where high

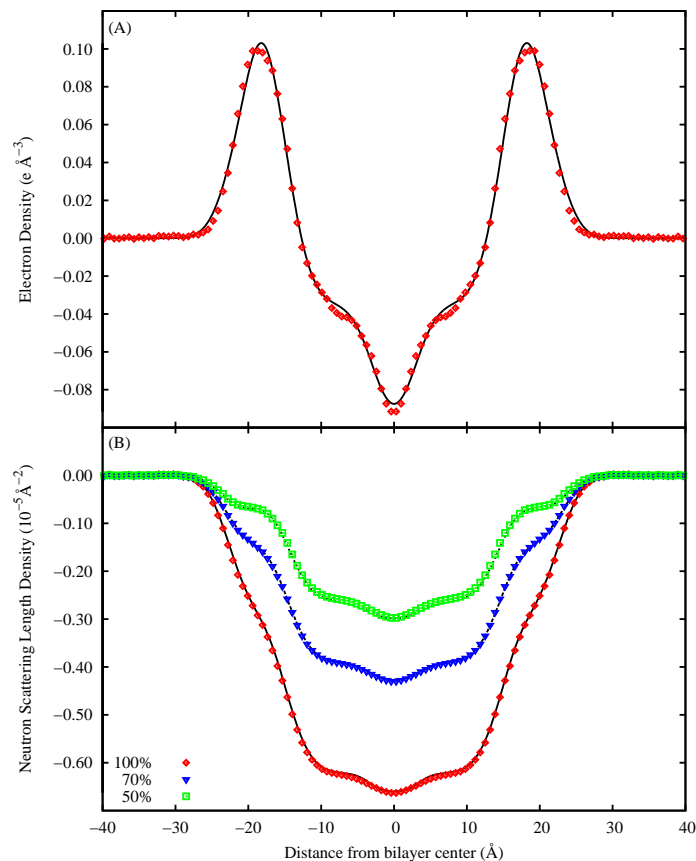


Figure 4.2: Electron (A) and neutron scattering length (B) relative to bulk water densities for MD (points) and the ADP model fit (lines). Since a constant scalar multiple has been removed during calculation of form factors, the ADP results are re-scaled.

errors are inevitable and the range of q is truncated due to instrumental limitations, sophisticated statistical methods were used to choose the model. Figure 4.2 compares the optimized ADP model real space structure (lines) with the molecular dynamics analysis (points). The important structural characteristics of the lipid bilayer are recovered from the MD data by the ADP model, despite several simplifying assumptions. Differences in real space density scale between the MD and ADP

results can result from difference in total lipid volume differences between MD and the experimental value used for ADP optimization and the insensitivity of the transformations to scale. Figure 4.3

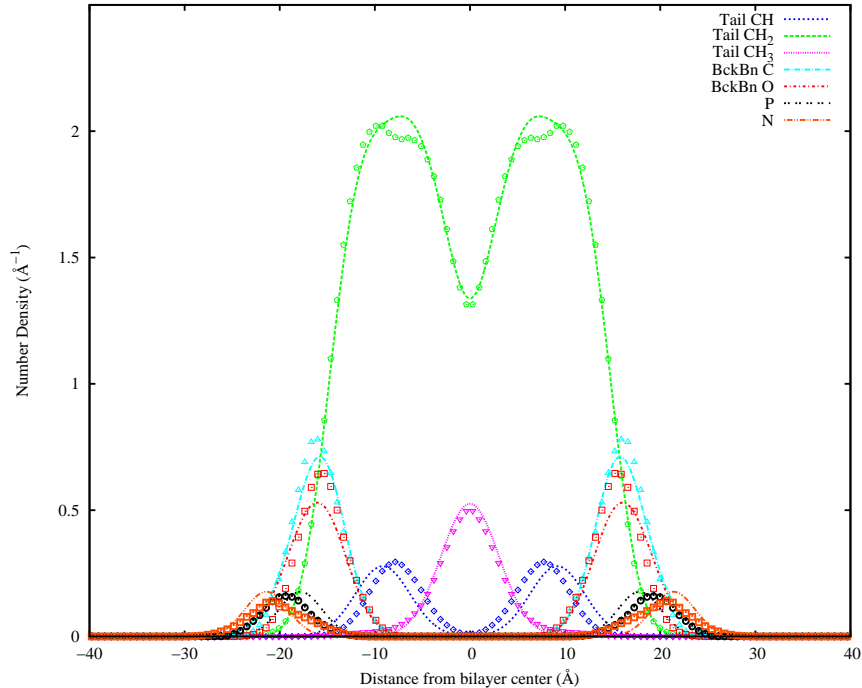


Figure 4.3: Number densities for groups of atoms determined from MD simulation (points) compared with the fitted ADP model (lines)

shows the number densities of selected groups of atom types for MD (points) and ADP (lines). The ADP model and optimization have recovered the main features of atomic distributions. The ADP model, despite simplifying assumptions, was able to accurately fit the low noise, high angle range MD data.

4.4.1 Parameter Optimization

Optimization of model parameters $\{\mu_i, \sigma_i, v_{\alpha_i}\}$ was performed using the software package ParOpt.²⁰⁷ The Nelder-Mead method employed for optimization is well suited to the specific challenges of this problem, i.e the large dimensionality (~ 115) of the space considered and the instability in the calculation of $\|d' - d\|$. The parameter space is the set of parameters $\{\mu_i, \sigma_i, v_{\alpha_i}\}$ bounded by the constraints. The target function for optimization is

$$\mathcal{F}(\{\mu_i, \sigma_i, v_{\alpha_i}\}) = \sum_j W_j \chi_j(\{\mu_i, \sigma_i, v_{\alpha_i}\}) \quad (4.25)$$

where W_j are the weight factors for the associated components (χ_j) of the target function and the index $j \in \{\text{elec1, elec2, neut100, neut75, neut70, neut50, volume, reg1, reg2}\}$. Weights were chosen heuristically to ensure a similar order of magnitude in target function contributions so that all components impact the entire optimization procedure. Weights chosen for optimization were $W_{\text{volume}} = 1000, W_{\text{neut}} = 10, W_{\text{elec}} = 30, W_{\text{reg1}} = 100$, and $W_{\text{reg2}} = 1000$. In the case of DOPC and DPPC, where two sets of experimental X-ray scattering data were included, a value of $W_{\text{elec}} = 15$ was used for each set. A value of $W_{\text{neut}} = 15$ was used for DPPC, since only two D₂O concentrations were available. Nelder-Mead then systematically reduces the value for $\mathcal{F}(\{\mu_i, \sigma_i, v_{\alpha_i}\})$ by evolving a simplex over the parameter space $\{\mu_i, \sigma_i, v_{\alpha_i}\}$.

4.5 Results

A consequence of the atomistic nature of the ADP model is that the number of effectively independent data points is smaller than the number of free parameters in the model, despite the regularization procedure described above. This issue can be only resolved with either further regularization or inclusion of more observations (e.g., neutron scattering of selectively deuterated lipids^{113,114}). More regularization is not feasible due to the lack of additional physical information. Thus, there are many points in the parameter space (i.e. models for the bilayer configuration) that match the data equivalently well within the margins of experimental error. In other words, the condition of uniqueness is not met, and multiple minima are expected in the target function \mathcal{F} in equation (4.25). Each instance of parameter set optimization by the Nelder-Mead method began with random initial configurations. Since the optimization is underdetermined, the converged value depends on the initial configuration, leading to many different optimal models. Since these different models represent the data similarly well, they must all be considered as plausible physical configurations for the bilayer. Therefore, many such models are considered (200 converged points for each lipid), and employ a weighted averaging procedure to obtain structural information, as outlined below.

The weighted average of quantity A is

$$\bar{A} = \frac{\sum_k \omega_k A_k}{\sum_k \omega_k} \quad (4.26)$$

where for the k^{th} optimal model, ω_k is the (non-negative) weight and A_k is the value of A . A measure of variability of \bar{A} is

$$\sigma = \sqrt{\frac{\sum_k \omega_k (A_k - \bar{A})^2}{\sum_k \omega_k}}. \quad (4.27)$$

In order to assign greater importance to models that match the data more closely, for the k^{th} optimal model with target function value \mathcal{F}_k , weights (ω_k) are chosen to be proportional to $\exp(-\mathcal{F}_k)$. To ensure numerical stability, define

$$\omega_k = \exp[-(\mathcal{F}_k - \mathcal{F}_1)] \quad (4.28)$$

where \mathcal{F}_1 is the lowest value for the target function over the set of optimal models considered for averaging.

The exponential form of the weights ω_k in equation (4.28) is motivated by analogy with a Bayesian¹²⁹ statistical formulation of the problem. Though the target function has the form $\mathcal{F} = \chi + \text{physically-based regularization terms}$ (equation (4.25)), a strict Bayesian formulation of the same problem under the assumption of Gaussian noise with known variance structure would lead to a target function of the form $\mathcal{F}' = \chi^2 + \text{physically-based regularization terms}$. In such a Bayesian formulation, the regularization terms together represent the negative logarithm of the prior distribution, and χ^2 is the negative logarithm of the likelihood function. The posterior distribution, which is the prime inferential object of the Bayesian machinery, has the form $\exp(-\mathcal{F}')$. Further, it is a probability (density) function that weighs all possible bilayer models with respect to the data and the prior information. Therefore, it provides weight according to the relative fitness of model. A superficially similar exponential form for averaging weights can also be found in the context of multimodel inference based on information criteria such as Akaike or Bayesian information criteria (BIC).¹³⁰ The use of \mathcal{F} instead of \mathcal{F}' is motivated by historical precedent. The usage of χ has been preferred in previous works, especially in applications of the SDP model.^{114–118} A rigorous statistical formulation, Bayesian or otherwise, would require careful modeling of the noise in the In practice, given the uncertainties in the data,^{131,132} serious discrepancy are not expected in the predictions based on \mathcal{F} and \mathcal{F}' .

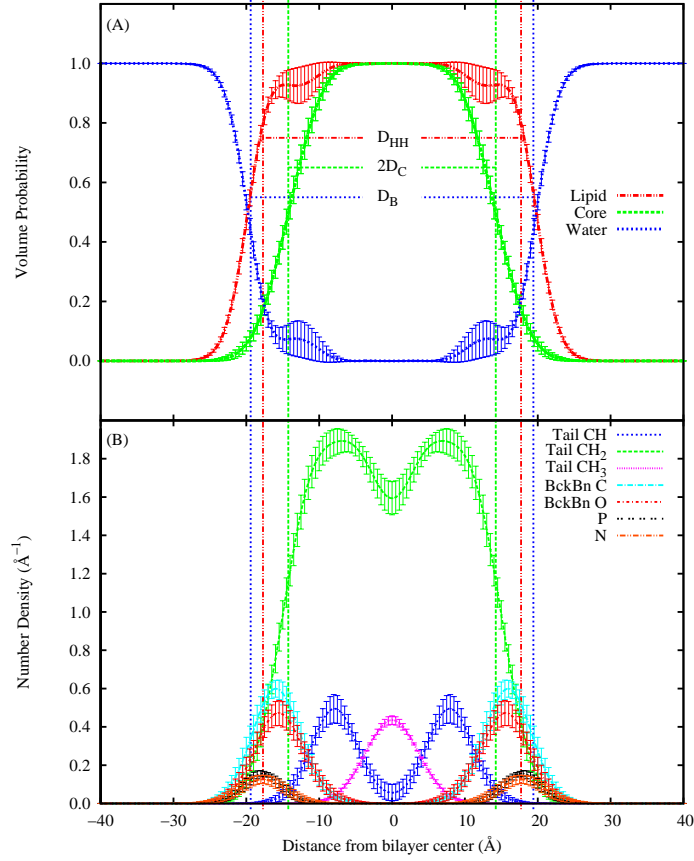


Figure 4.4: Probability density plots (A) for DOPC included for total lipid, hydrocarbon core, and water. Number densities (B) for hydrocarbon chain groups of the same type (Tail Ch, Tail Ch₂, and Tail Ch₃), backbone groups (including the carbonyl atoms) of the same type (BckBn O and BckBn C), and head group atoms (P and N). Profiles and structural property locations D_{hh} , $2D_C$, D_B are taken from the weighted average of those determined from converged parameter sets. Error bars are determined from the standard deviations of the weighted averages, included only at 0.5 Å intervals for clarity.

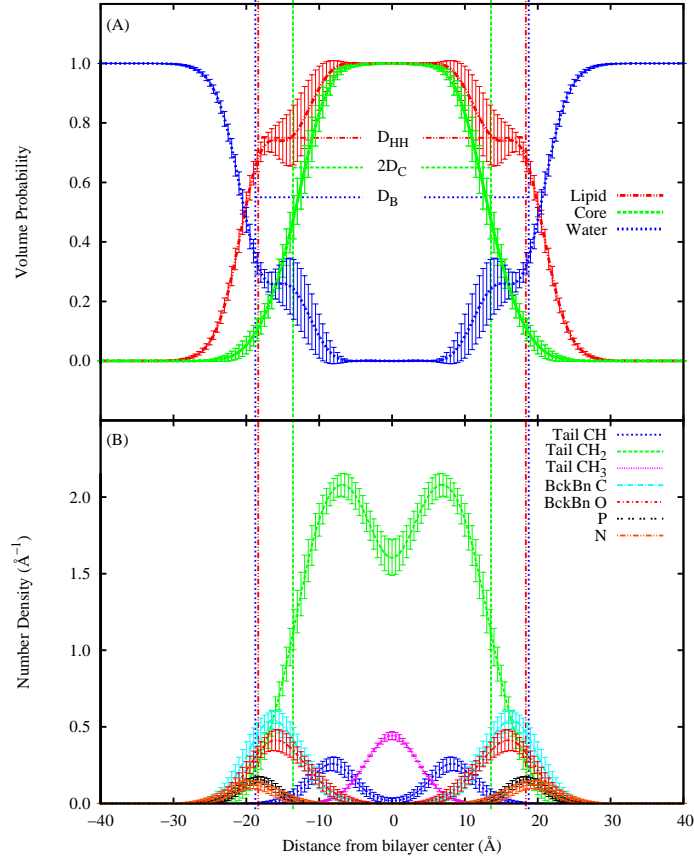


Figure 4.5: Probability density plots (A) for POPC included for total lipid, hydrocarbon core, and water. Number densities (B) for hydrocarbon chain groups of the same type (Tail Ch, Tail Ch₂, and Tail Ch₃), backbone groups (including the carbonyl atoms) of the same type (BckBn O and BckBn C), and head group atoms (P and N). Profiles and structural property locations D_{hh} , $2D_C$, D_B are taken from the weighted average of those determined from converged parameter sets. Error bars are determined from the standard deviations of the weighted averages, included only at 0.5 Å intervals for clarity.

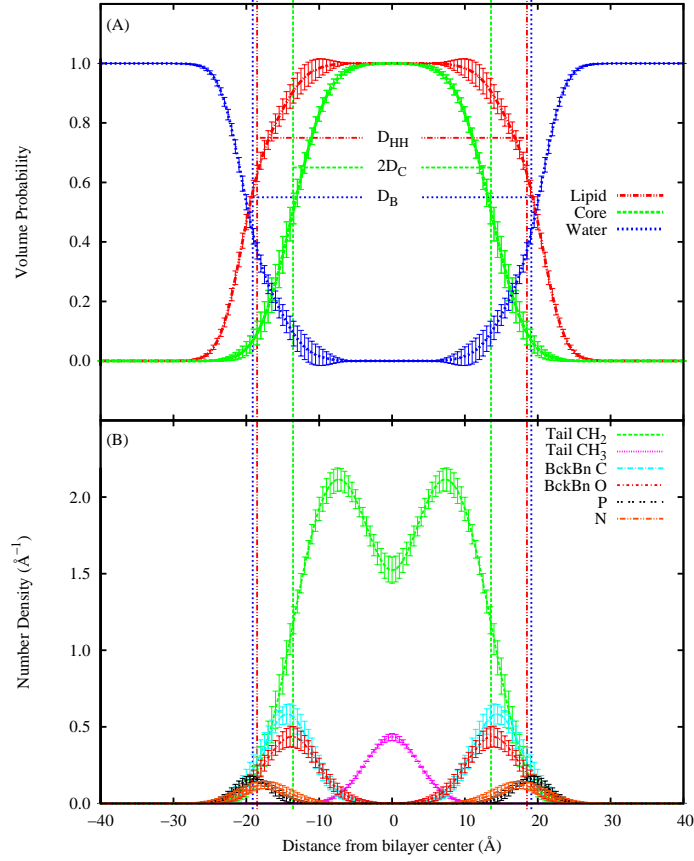


Figure 4.6: Probability density plots (A) for DPPC included for total lipid, hydrocarbon core, and water. Number densities (B) for hydrocarbon chain groups of the same type (Tail CH₂ and Tail CH₃), backbone groups (including the carbonyl atoms) of the same type (BckBn O and BckBn C), and head group atoms (P and N). Profiles and structural property locations D_{hh} , $2D_C$, D_B are taken from the weighted average of those determined from converged parameter sets. Error bars are determined from the standard deviations of the weighted averages, included only at 0.5 Å intervals for clarity.

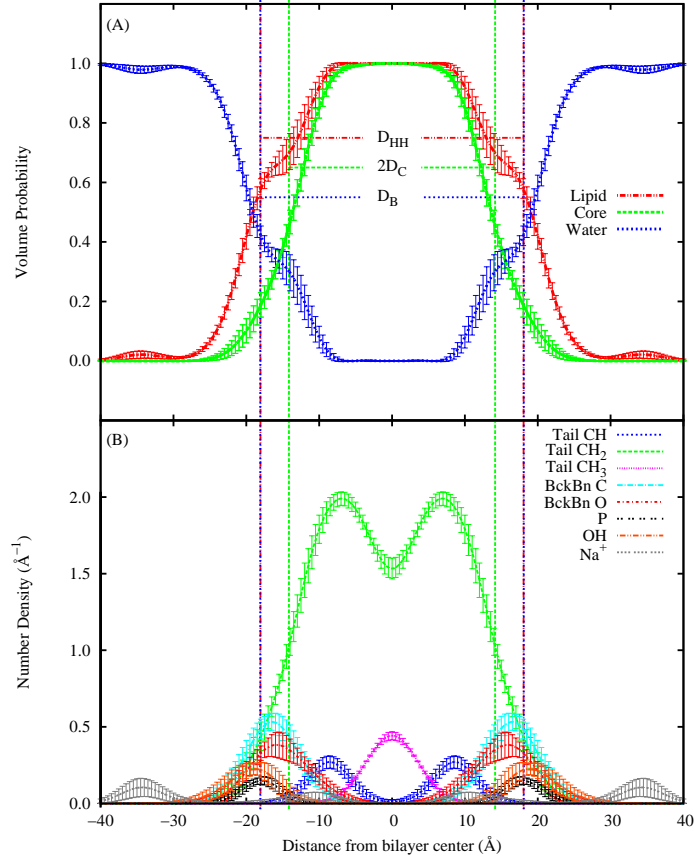


Figure 4.7: Probability density plots (A) for POPG included for total lipid, hydrocarbon core, and water. Number densities (B) for hydrocarbon chain groups of the same type (Tail Ch, Tail Ch₂, and Tail Ch₃), backbone groups (including the carbonyl atoms) of the same type (BckBn O and BckBn C), head group atoms (P and Oh), and counter ion (Na⁺). Profiles and structural property locations D_{hh} , $2D_C$, D_B are taken from the weighted average of those determined from converged parameter sets. Error bars are determined from the standard deviations of the weighted averages, included only at 0.5 Å intervals for clarity.

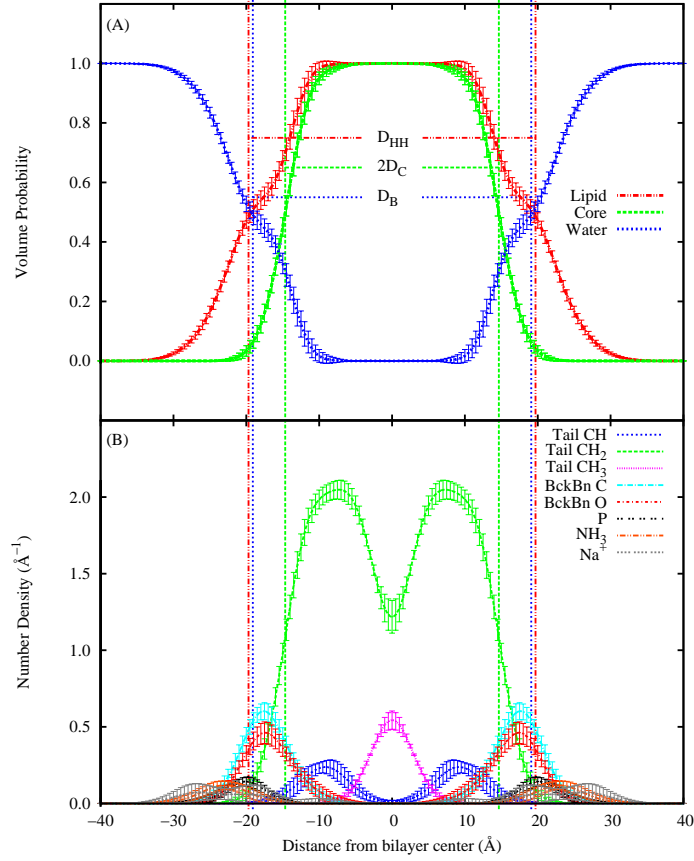


Figure 4.8: Probability density plots (A) for POPS included for total lipid, hydrocarbon core, and water. Number densities (B) for hydrocarbon chain groups of the same type (Tail Ch, Tail Ch₂, and Tail Ch₃), backbone groups (including the carbonyl atoms) of the same type (BckBn O and BckBn C), head group atoms (P and NH₃), and counter ion (Na⁺). Profiles and structural property locations D_{hh} , $2D_C$, D_B are taken from the weighted average of those determined from converged parameter sets. Error bars are determined from the standard deviations of the weighted averages, included only at 0.5 Å intervals for clarity.

4.5.1 Structural Properties

Subfigure (A) of figure 4.4 to 4.8 shows volume probabilities for the hydrocarbon core, total lipid, and water. Average structural characteristics peak-to-peak distance (D_{hh}), overall bilayer thickness (D_B), and hydrocarbon core thickness (D_C) are also indicated. Regions of increased uncertainty in the water and lipid density profiles are found within the headgroup region for all lipids. The magnitude of this increase in uncertainty depends on the lipid, being most extreme in DOPC (figure 4.4) and POPC (figure 4.5). These regions indicate the possible presence of a hydration layer within the backbone region, in contrast to a smooth drop from interfacial to bulk water. Areas of water concentration within the head group have been proposed by both theory¹³³ and experiment.¹³⁴ The counter ion distribution for the two acidic lipids (POPG and POPS) differ substantially. The sodium ion in POPS is more strongly associated with the head group than that of POPG. Subfigure (B) of figure 4.4 to 4.8 shows the number density for different subgroups within a lipid molecule. Tail groups include atoms from both hydrocarbon chains of the same type (Tail Ch, Tail CH_2 , and Tail CH_3). The groups BckBn C and BckBn O include atoms of the respective type in both the glycerol backbone and carbonyl groups.

Figure 4.9 to 4.13 show weighted averages of the electron and neutron scattering length densities for each lipid. Insets contain example fits chosen from the converged point with the lowest value for the target function(\mathcal{F}_l), along with experimental data and the experimental error bars.

Table 4.3: Lipid structural properties determined from model averages. Hydrogen-deuterium exchange is taken into account for hydroxyl groups in POPG and the amine group in POPS. Values shown are weighted averages using equation (4.26) and standard deviations using equation (4.27) of 200 optimized models. Weights were determined using equation (4.28). Head includes headgroup, glycerol backbone, and the carbonyl groups. Core group fatty acid tails, excluding the carbonyl groups.

| | DOPC | POPC | DPPC | POPG | POPS |
|--------------------------|------------------|------------------|------------------|------------------|-----------------|
| $V_{head}(\text{\AA}^3)$ | 346.7 ± 16.4 | 346.7 ± 17.8 | 353.3 ± 24.7 | 263.0 ± 15.6 | 278.7 ± 9.1 |
| $V_{core}(\text{\AA}^3)$ | 956.3 ± 16.4 | 909.3 ± 17.8 | 866.2 ± 26.9 | 945.7 ± 15.6 | 919.8 ± 9.1 |
| $v_{Ch}(\text{\AA}^3)$ | 22.6 ± 1.3 | 22.1 ± 1.7 | — | 21.9 ± 1.8 | 21.4 ± 1.9 |
| $v_{CH_2}(\text{\AA}^3)$ | 27.2 ± 0.7 | 27.2 ± 0.6 | 27.2 ± 0.9 | 28.5 ± 0.5 | 27.5 ± 0.4 |
| $v_{CH_3}(\text{\AA}^3)$ | 51.9 ± 1.1 | 52.2 ± 1.4 | 51.9 ± 1.1 | 52.4 ± 1.5 | 54.2 ± 1.5 |
| $D_{hh}(\text{\AA})$ | 35.4 ± 0.1 | 36.7 ± 0.2 | 37.0 ± 0.2 | 36.1 ± 0.3 | 39.4 ± 0.3 |
| $D_B(\text{\AA})$ | 38.8 ± 0.5 | 37.5 ± 1.0 | 38.2 ± 0.4 | 36.2 ± 0.4 | 38.2 ± 0.3 |
| $2D_C(\text{\AA})$ | 28.5 ± 0.6 | 27.2 ± 1.0 | 27.2 ± 0.7 | 28.3 ± 0.5 | 29.3 ± 0.3 |
| $A(\text{\AA}^2)$ | 67.2 ± 0.9 | 67.0 ± 1.8 | 63.8 ± 0.7 | 66.9 ± 0.7 | 62.7 ± 0.5 |

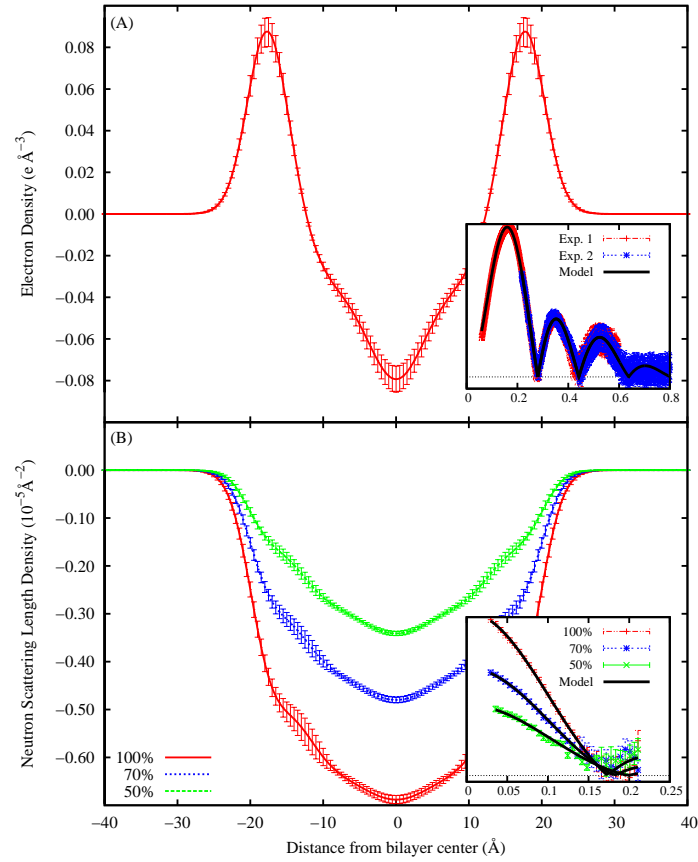


Figure 4.9: Electron (A) and neutron scattering length (B) densities, relative to bulk water, shown for DOPC. Example fits of the lowest target function value point, with model and measured form factors (arbitrary units) as a function of scattering vector q (\AA^{-1}), are included as insets. Percentages indicate D_2O concentration.

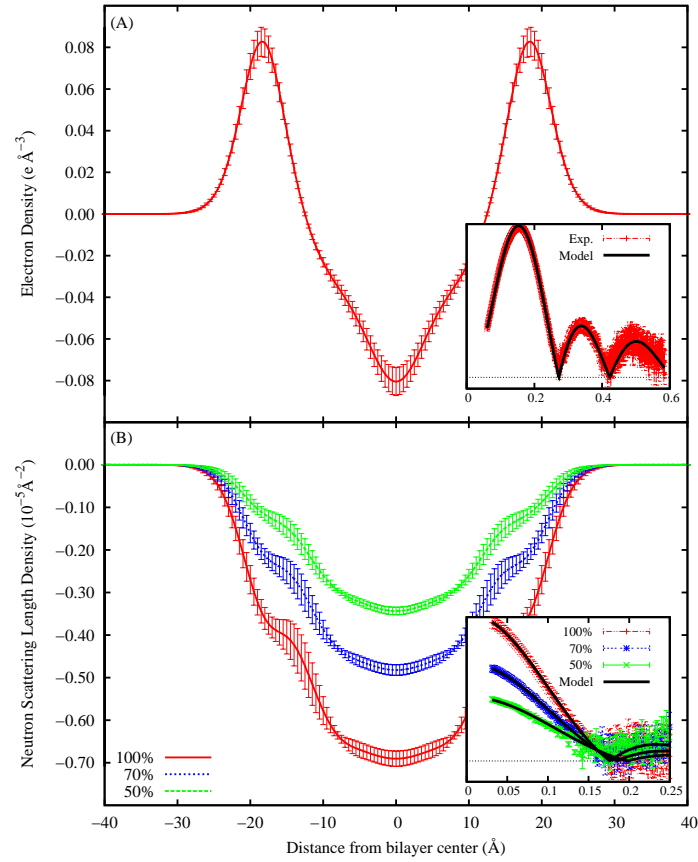


Figure 4.10: Electron (A) and neutron scattering length (B) densities, relative to bulk water, shown for POPC. Example fits of the lowest target function value point, with model and measured form factors (arbitrary units) as a function of scattering vector q (\AA^{-1}), are included as insets. Percentages indicate D_2O concentration.

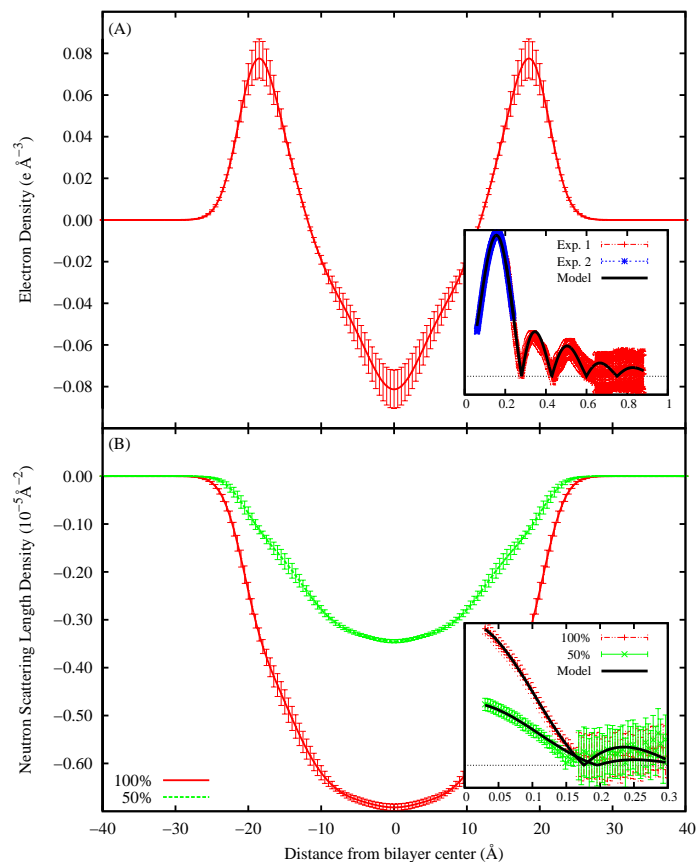


Figure 4.11: Electron (A) and neutron scattering length (B) densities, relative to bulk water, shown for DPPC. Example fits of the lowest target function value point, with model and measured form factors (arbitrary units) as a function of scattering vector q (\AA^{-1}), are included as insets. Percentages indicate D_2O concentration.

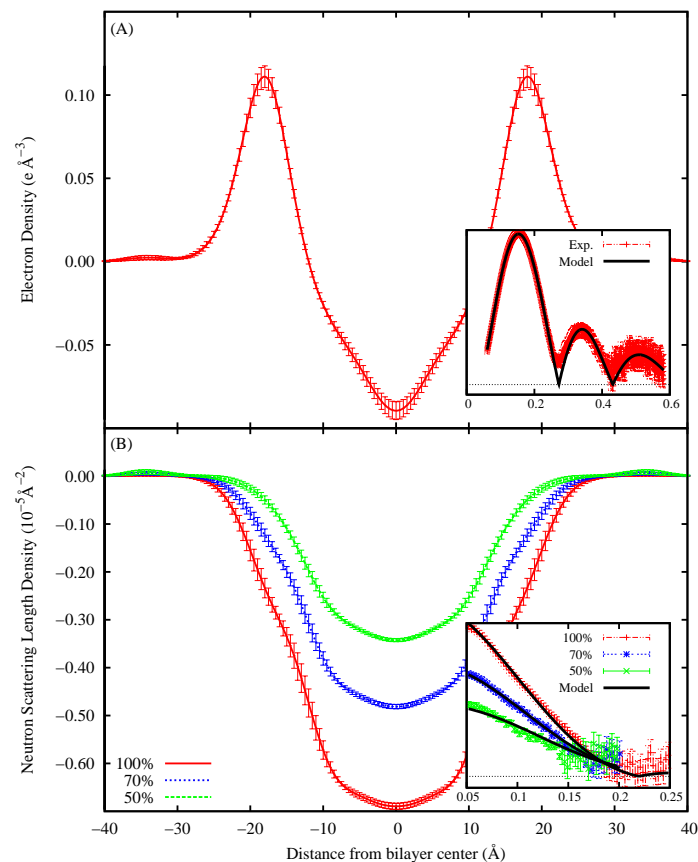


Figure 4.12: Electron (A) and neutron scattering length (B) densities, relative to bulk water, shown for POPG. Example fits of the lowest target function value point, with model and measured form factors (arbitrary units) as a function of scattering vector q (\AA^{-1}), are included as insets. Percentages indicate D_2O concentration.

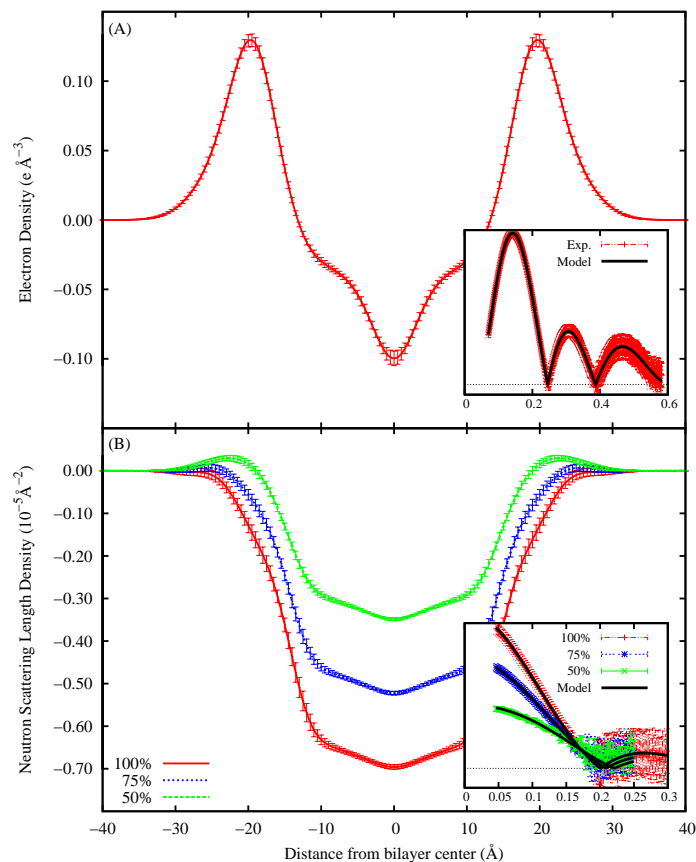


Figure 4.13: Electron (A) and neutron scattering length (B) densities, relative to bulk water, shown for POPS. Example fits of the lowest target function value point, with model and measured form factors (arbitrary units) as a function of scattering vector q (\AA^{-1}), are included as insets. Percentages indicate D_2O concentration.

Table 4.3 lists the weighted averages of several important structural properties for lipid bilayers composed of DOPC, POPC, DPPC, POPG, and POPS determined using the ADP model. In general, the ADP model gives rise to lipid bilayer structures that are consistent with those obtained from the SDP model.^{114–118} The partial component volumes (V_{head} , V_{core} , v_{Ch} , v_{Ch_2} , and v_{Ch_3}) are similar to those reported by the SDP model. Peak-to-peak distance (D_{hh}) is defined as twice the distance from the bilayer center to the maximum value in the electron density profile. Overall bilayer thickness (D_{B}) was calculated by solving the following equation for D_{B} :^{135,136}

$$\int_0^{D_{\text{B}}/2} (1 - p_{\text{lipid}}(z)) dz = \int_{D_{\text{B}}/2}^{\infty} p_{\text{lipid}}(z) dz. \quad (4.29)$$

The thickness of the hydrocarbon core, excluding the chain carbonyl groups, ($2D_{\text{C}}$) was similarly determined by solving:

$$\int_0^{D_{\text{C}}} (1 - p_{\text{core}}(z)) dz = \int_{D_{\text{C}}}^{\infty} p_{\text{core}}(z) dz. \quad (4.30)$$

The overall bilayer thickness D_{B} and hydrocarbon chain thickness $2D_{\text{C}}$ obtained from the ADP model conform to those reported by SDP model.^{114–118} Area per lipid (A) which reflects the lipid bilayer lateral packing property, was calculated from

$$A = \frac{V_{\text{lipid}}}{D_{\text{B}}/2}. \quad (4.31)$$

Results for area are also consistent with those of the SDP model which has a lipid area uncertainty of $1 - 2 \text{ \AA}^2$.¹¹⁴ For the three PC lipids, the ADP model predicts 63.8, 67.0 and 67.2 \AA^2 for DPPC, POPC and DOPC, respectively, while the corresponding lipid areas based on SDP model are 63.1, 64.3, and 67.4 \AA^2 .^{114,115} The lipid area of POPG determined using the ADP model is 66.9 \AA^2 . This value is in good agreement with the reported 66.1 \AA^2 using SDP model which did not model hydrogen-deuterium exchange.¹¹⁶ For the apoptosis related POPS lipid, the ADP model yields a lipid area of 62.7 \AA^2 , which is in also in excellent agreement with the value 62.7 \AA^2 reported using the SDP model.¹¹⁸

For the DPPC bilayer, unlike other lipids, the sum of lipid component volumes V_{head} and V_{core} is not equal to the target value for total lipid volume (V_{exp}). The model structures which best fit

the scattering data are consistent with a total lipid volume of $1219.5 \pm 2.4 \text{ \AA}^3$ which is substantially lower than the target value of 1228.5 \AA^3 . Since the precision of the total volume is higher than the difference between the target and model predicted values, this discrepancy is an important result of the fitting.

Despite the similar structural parameters resulting from the ADP and SDP models, the method and model presented here offer several advantages: (1) the ADP model yields atomic distributions, while only group information can be obtained from the SDP model, (2) the ADP model requires minimal constraints on parameters aside from regularization considerations such as physical bond length, while the SDP model demands many more soft constrained parameters in order to obtain stable fitting (e.g., the error function width), (3) variance in model prediction indicates the precision with which scattering data can predict structural properties, and (4) the ADP model can be easily extended to complicated systems such as lipid mixtures and transmembrane proteins.

4.6 Conclusion

A general method using the ADP model for the determination of lipid membrane structures via SAXS and SANS data has been developed. The method optimizes the ADP model which makes minimal assumptions on the underlying atomic structure. The paucity of experimental data and the overabundance of free parameters are overcome by applying regularization methods, i.e. parameters are constrained to reflect molecular topology and penalty terms are added to take hydrophobic interactions into account. Despite these reductions in redundancy, the model remains underdetermined. Since all solutions to the problem are equally valid, structure predictions from all optimal models are averaged, using a weight factor inspired by Bayesian information criterion. This allows the determination of bilayer structures using atomic models, a level of detail that was not achievable in previously published methods. The greater flexibility of the ADP model and the more rigorous fitting procedure described above has yielded bilayer structure comparable to existing work. The increased level of detail in the ADP model produces bilayer structures with the possibility of a distinct hydration layer within the interfacial region. In future work, the ADP model will be applied to lipid bilayer systems that are too complex for simpler models.

Chapter 5

Reactive Water

5.1 Introduction

Many vital biological processes have important stages with changing chemical compositions. Metabolic pathways involve the enzymatic synthesis or decomposition of energy carrying molecules.¹³⁷ With a reactive classical model of biological systems, phases of these pathways could be studied using powerful statistical mechanical tools. Many of the byproducts of metabolic processes are chemically reactive oxidative species (ROS).¹³⁸ These species are produced as leakage of important biological functions and are found in high concentrations in the mitochondria where many important metabolic reactions occur. Oxidative damage to cellular components, due in part to ROS, is an important factor in many diseases, including many cancers and age related disease.¹³⁹ An accurate reactive biological forcefield would have great explanatory power in the study of oxidative damage to the structure of lipids, proteins and the resulting impairment of function. The accuracy of the water model is of fundamental importance when simulating a biological system.²⁴ The interactions among water molecules and between water molecules and macromolecules is the fundamental driving force behind the hydrophobic effect.²⁵ This effect plays an important role in many biological processes. Therefore, the first step in the development of any biologically relevant force field is in the construction of an accurate representation of water. Even in the case of pure water, bond order potentials may model phenomena outside the scope of conventional classical molecular dynamics. Dissociation of water molecules is an important factor in proton transport within water, via the Grotthus mechanism.¹⁴⁰ The pH of aqueous solutions is an important factor in many biological

processes. The activity of the hydrogen ion could in principle be studied directly via reactive molecular dynamics.¹⁴¹ In previous works (see appendix B) a chemically reactive interface between silica and water was simulated. The water model presented in that work is an accurate reflection of the behavior of water in a constant volume ensemble, but it does not accurately model water at a constant pressure. In simulations of biological relevance, the isothermal isobaric ensemble is of vital importance.

Conventional molecular dynamics makes the assumption of static covalent bonding, assigning harmonic (or harmonic-like) interaction potentials to bonded interactions. The energy of these interactions diverges as the distance separating atoms increases. As a result of this assumption and these interactions, conventional classical molecular dynamics is unable to simulate systems with chemical reactions. Bond order potentials allow chemical reactions by introducing a bond order term which is a measure of the degree of covalent bonding between two atoms. These potentials rely on bond order, and in the case of ReaxFF,¹⁴² the Pauling relationship

$$bo_{ij} \propto \exp \left[\frac{\alpha_{ij}}{r_{ij}} \right] \quad (5.1)$$

where bo_{ij} is the bond order between atoms i and j , α_{ij} is a coefficient dependent on atoms i and j , and r_{ij} is the interatomic separation.

5.2 Model

In the ReaxFF model, the bond order is defined for all pairs of atoms in the system. The uncorrected bond order (BO'_{ij}) is calculated for all neighboring pairs in the system, where the superscripts σ , π , and $\pi\pi$ refer to the different types of covalent bonding. Corrected bond orders are calculated to reflect atomic valencies based on the sums of uncorrected bond order on each atom. This problem is in principle a many-body interaction.

All bonded interaction terms (bond, angle, dihedral) are dependent on bond order such that as the distance between two atoms increases, the bond order and all bonded interaction energies go smoothly to 0. The covalent bond energy term is attractive only. Non-bonded interactions include a repulsive van der Waals energy, which acts on all pairs within a cutoff. Coulombic interactions

are included for all pairs separated by a distance less than a user defined cutoff. Since changing bond connectivity implies a changing electronic structure, ReaxFF employs a charge equilibration method (QEq) which recalculates the partial atomic charge on each atom in order to minimize the electrostatic potential, with the constraint of total system charge neutrality.

The complete ReaxFF potential, as developed by van Duin et al.^{142–144} and implemented in Purdue Reactive Molecular Dynamics (PuReMD)^{145,208} is listed below. For the sake of notational simplicity, subscript indexes have been omitted from parameter values.

5.2.1 Bond Orders

All bonded interactions depend on bond order, therefore the calculation of interactions in the ReaxFF system involves first the determination of bond orders for all pairs. The first phase in the bond order calculation is the assignment of uncorrected bond orders (BO'_{ij}).

$$\begin{aligned}\text{BO}'_{ij} &= \text{BO}'_{ij}{}^{\sigma} + \text{BO}'_{ij}{}^{\pi} + \text{BO}'_{ij}{}^{\pi\pi} \\ \text{BO}'_{ij}{}^{\sigma} &= (1 + bo_{cut}) \exp(p_{bo1}(\frac{r_{ij}}{r_0^{\sigma}})^{p_{bo2}}) - bo_{cut}\end{aligned}\tag{5.2}$$

$$\text{BO}'_{ij}{}^{\pi} = \exp(p_{bo3}(\frac{r_{ij}}{r_0^{\pi}})^{p_{bo4}})\tag{5.3}$$

$$\text{BO}'_{ij}{}^{\pi\pi} = \exp(p_{bo5}(\frac{r_{ij}}{r_0^{\pi\pi}})^{p_{bo6}})\tag{5.4}$$

These values are then corrected for valencies, to give the final bond orders (BO_{ij}) for all neighboring pairs.

$$\Delta'_i = -\text{Val}_i + \sum_{j=1}^{\text{nbr}(i)} \text{BO}'_{ij} \quad (5.5)$$

$$\Delta_i'^{\text{boc}} = -\text{Val}_i^{\text{boc}} + \sum_{j=1}^{\text{nbr}(i)} \text{BO}'_{ij} \quad (5.6)$$

$$\text{BO}_{ij} = \text{BO}'_{ij} \text{f1}(\Delta'_i, \Delta'_j) \text{f4}(\Delta'_i, \text{BO}'_{ij}) \text{f5}(\Delta'_j, \text{BO}'_{ij}) \quad (5.7)$$

$$\text{BO}_{ij}^{\pi} = \text{BO}'_{ij} \text{f1}(\Delta'_i, \Delta'_j) \text{f1}(\Delta'_i, \Delta'_j) \text{f4}(\Delta'_i, \text{BO}'_{ij}) \text{f4}(\Delta'_j, \text{BO}'_{ij}) \quad (5.8)$$

$$\text{BO}_{ij}^{\pi\pi} = \text{BO}'_{ij} \text{f1}(\Delta'_i, \Delta'_j) \text{f1}(\Delta'_i, \Delta'_j) \text{f4}(\Delta'_i, \text{BO}'_{ij}) \text{f4}(\Delta'_j, \text{BO}'_{ij}) \quad (5.9)$$

$$\text{BO}_{ij}^{\sigma} = \text{BO}_{ij} - \text{BO}_{ij}^{\pi} - \text{BO}_{ij}^{\pi\pi} \quad (5.10)$$

$$(5.11)$$

The functions f1, f2, f3, f4, and f5 are calculated as follows.

$$\text{f1}(\Delta'_i, \Delta'_j) = 0.5 \left(\frac{\text{Val}_i + \text{f2}(\Delta'_i, \Delta'_j)}{\text{Val}_i + \text{f2}(\Delta'_i, \Delta'_j) + \text{f3}(\Delta'_i, \Delta'_j)} + \frac{\text{Val}_j + \text{f2}(\Delta'_i, \Delta'_j)}{\text{Val}_j + \text{f2}(\Delta'_i, \Delta'_j) + \text{f3}(\Delta'_i, \Delta'_j)} \right) \quad (5.12)$$

$$\text{f2}(\Delta'_i, \Delta'_j) = \exp(-p_{\text{boc1}} \Delta'_i) + \exp(-p_{\text{boc1}} * \Delta'_j) \quad (5.13)$$

$$\text{f3}(\Delta'_i, \Delta'_j) = \frac{-1}{p_{\text{boc2}}} \log(0.5(\exp(-p_{\text{boc2}} \Delta'_i) + \exp(-p_{\text{boc2}} \Delta'_j))) \quad (5.14)$$

$$\text{f4}(\Delta'_i, \text{BO}'_{ij}) = (1 + \exp(-p_{\text{boc3}}(p_{\text{boc4}} \text{BO}'_{ij} \text{BO}'_{ij} - \Delta_i'^{\text{boc}}) + p_{\text{boc5}}))^{-1} \quad (5.15)$$

$$\text{f5}(\Delta'_j, \text{BO}'_{ij}) = (1 + \exp(-p_{\text{boc3}}(p_{\text{boc4}} \text{BO}'_{ij} \text{BO}'_{ij} - \Delta_j'^{\text{boc}}) + p_{\text{boc5}}))^{-1} \quad (5.16)$$

Parameters p_{boc3} , p_{boc4} , and p_{boc5} are calculated based on atomic parameters.

$$p_{\text{boc3}} = (\text{bo}_i^{132} \cdot \text{bo}_j^{132})^2 \quad (5.17)$$

$$p_{\text{boc4}} = (\text{bo}_i^{131} \cdot \text{bo}_j^{131})^2 \quad (5.18)$$

$$p_{\text{boc5}} = (\text{bo}_i^{133} \cdot \text{bo}_j^{133})^2 \quad (5.19)$$

The difference between an atom's valency and the total corrected bond order plays a role in energy calculations for undercoordination and overcoordination.

$$\Delta_i = -Val_i + \sum_{j=1}^{nbr(i)} BO_{ij} \quad (5.20)$$

$$\Delta_i^{boc} = -Val_i^{boc} + \sum_{j=1}^{nbr(i)} BO_{ij} \quad (5.21)$$

5.2.2 Bond Energy

Total bond energy between two atoms has contributions from σ , π , and $\pi\pi$ bonds.

$$E_{\text{bond}} = -D_e^\sigma BO_{ij}^\sigma \exp(p_{be1}(1 - (BO_{ij}^\sigma)^{p_{be2}})) - D_e^\pi BO_{ij}^\pi - D_e^{\pi\pi} BO_{ij}^{\pi\pi} \quad (5.22)$$

5.2.3 Lone Pair Energy

Lone pair energies are calculated, depending on the total bond order as follows.

$$E_{lp} = \frac{p_{lp2} \Delta_i^{lp}}{1 + \exp(-75 \Delta_i^{lp})} \quad (5.23)$$

$$\Delta_i^e = -Val_i^e + \sum_{j=1}^{n(i)} BO_{ij} \quad (5.24)$$

$$n_{lp,i} = \begin{cases} \exp(-p_{lp1}(2 + \Delta_i^e - 2\text{int}(\frac{\Delta_i^e}{2}))^2) - \text{int}(\frac{\Delta_i^e}{2}) & \text{if mass} \leq 21.0 \\ \frac{1}{2}(Val_1^e - Val_i) & \text{if mass} > 21.0 \end{cases} \quad (5.25)$$

$$\Delta_i^{lp} = n_{lp,\text{opt}} - n_{lp,i} \quad (5.26)$$

$$n_{lp,\text{opt}} = \frac{1}{2}(Val_i^e - Val_i) \quad (5.27)$$

5.2.4 C₂ Correction

With an additional correction for carbon-carbon bonds

$$vov3 = BO_{ij} - \Delta_i - 0.04\Delta_i^4 \quad (5.28)$$

if $vov3 > 3$,

$$E_{lph} = p_{lp3}(vov3 - 3)^2 \quad (5.29)$$

5.2.5 Triple Bond Stabilization

For carbon-oxygen bonds a further addition is made.

$$E_{tbs} = p_{gp10} \cdot \exp[-p_{gp7}(BO_{ij} - 2.5)] \frac{1}{1 + 25 \exp[p_{gp4}(\Delta_i + \Delta_j)]} \cdot [\exp(-p_{gp3}(TBO_i - BO_{ij})) + \exp(-p_{gp3}(TBO_j - BO_{ij}))] \quad (5.30)$$

5.2.6 Overcoordination Energy

An energy penalty for atoms with bond order exceeding the valency is calculated.

$$E_{over} = \frac{\sum_j^{n(i)} p_{ovun1} D_e^\sigma BO_{ij}}{\Delta_i^{lpcorr} + Val_i} \Delta_i^{lpcorr} (1 + \exp(p_{ovun2} \Delta_i^{lpcorr}))^{-1} \quad (5.31)$$

$$\Delta_i^{lpcorr} = \Delta_i - \Delta_i^{lp} (1 + p_{ovun3} \exp(p_{ovun4} \sum_j^{n(i)} (\Delta_j - \Delta_j^{lp})(BO_{ij}^\pi + BO_{ij}^{\pi\pi})))^{-1} \quad (5.32)$$

5.2.7 Undercoordination Energy

Similarly, an energy for atoms with bond orders less than valency is also calculated.

$$E_{under} = -p_{ovun5} \frac{1 - \exp(p_{ovun6} \Delta_i^{lpcorr})}{1 + \exp(-p_{ovun2} \Delta_i^{lpcorr})} (1 + p_{ovun7} \exp(p_{ovun8} \sum_j^{n(i)} (\Delta_j - \Delta_j^{lp})(BO_{ij}^\pi + BO_{ij}^{\pi\pi})))^{-1} \quad (5.33)$$

5.2.8 Valence Angle Energy

Three body angle energies are calculated as follows.

$$E_{val} = f7(BO_{ij})f7(BO_{jk})f8(\Delta_j)p_{val1}(1 - \exp(-p_{val2}(\Theta_0 - \Theta_{ijk})^2)) \quad (5.34)$$

$$f7(BO_{ij}) = 1 - \exp(-p_{val3}BO_{ij}^{p_{val4}}) \quad (5.35)$$

$$f8(\Delta_j) = p_{val5} - (p_{val5} - 1) \frac{2 + \exp(p_{val6}\Delta_j^{angle})}{1 + \exp(p_{val6}\Delta_j^{angle}) + \exp(-p_{val7}\Delta_j^{angle})} \quad (5.36)$$

$$\Delta_i^{angle} = -Val_i^{val} + \sum_j^{n(i)} BO_{ij} \quad (5.37)$$

Where the equilibrium angle Θ_0 is determined as follows.

$$\Theta_0 = \pi - \Theta_{0,0}\{1 - \exp(-p_{val10}(2 - SBO2))\} \quad (5.38)$$

$$SBO = \sum_n^{n(j)} (BO_{jn}^\pi + BO_{jn}^{\pi\pi}) + \left[1 - \prod_n^{n(j)} \exp(-BO_{jn}^8) \right] (-\Delta_j^{angle} - p_{val8}n_{lp,i}) \quad (5.39)$$

$$SBO2 = \begin{cases} 0 & \text{if } SBO \leq 0 \\ SBO^{p_{val9}} & \text{if } 0 < SBO < 1 \\ 2 - (2 - SBO)^{p_{val9}} & \text{if } 1 < SBO < 2 \\ 2 & \text{if } SBO \geq 2 \end{cases} \quad (5.40)$$

5.2.9 Penalty Energy

For systems with one atom containing two double bonds, an additional energy penalty is added.

$$E_{pen} = p_{pen1}f9(\Delta_j) \exp[-p_{pen2}(BO_{ij} - 2)^2] \exp[-p_{pen2}(BO_{jk} - 2)^2] \quad (5.41)$$

$$f9 = \frac{2 + \exp(-p_{pen3}\Delta_j)}{1 + \exp(-p_{pen3}\Delta_j) + \exp(p_{pen4}\Delta_j)} \quad (5.42)$$

5.2.10 Three-body Conjugation Energy

For conjugated molecules, an additional three-body energy is added.

$$\begin{aligned}
E_{coa} = & p_{coa1} \frac{1}{1 + \exp(p_{coa2} \Delta_j^{val})} \cdot \\
& \exp \left[-p_{coa3} \left(-\text{BO}_{ij} + \sum_n^{nbr(i)} \text{BO}_{in} \right)^2 \right] \cdot \\
& \exp \left[-p_{coa3} \left(-\text{BO}_{jk} + \sum_n^{nbr(k)} \text{BO}_{kn} \right)^2 \right] \cdot \\
& \exp \left[-p_{coa4} (\text{BO}_{ij} - 1.5)^2 \right] \exp \left[-p_{coa4} (\text{BO}_{jk} - 1.5)^2 \right]
\end{aligned} \tag{5.43}$$

5.2.11 Torsion Angle Energy

A four-body torsion angle energy is also calculated.

$$\begin{aligned}
E_{tor} = & \frac{1}{2} \text{f10}(\text{BO}_{ij}, \text{BO}_{jk}, \text{BO}_{kl}) \sin(\Theta_{ijk}) \sin(\Theta_{jkl}) \cdot \\
& [V_1(1 + \cos(\omega_{ijkl})) + \\
& V_2 \exp \left(p_{tor1} (2 - \text{BO}_{jk}^\pi - \text{f11}(\Delta_j, \Delta_k))^2 \right) (1 - \cos(2\omega_{ijkl})) + \\
& V_3(1 + \cos(3\omega_{ijkl}))]
\end{aligned} \tag{5.44}$$

$$\begin{aligned}
\text{f10}(\text{BO}_{ij}, \text{BO}_{jk}, \text{BO}_{kl}) = & [1 - \exp(-p_{tor2} \text{BO}_{ij})] \cdot \\
& [1 - \exp(-p_{tor2} \text{BO}_{jk})] [1 - \exp(-p_{tor2} \text{BO}_{kl})]
\end{aligned} \tag{5.45}$$

$$\text{f11}(\Delta_j, \Delta_k) = \frac{2 + \exp \left[-p_{tor3} (\Delta_j^{angle} + \Delta_k^{angle}) \right]}{1 + \exp \left[-p_{tor3} (\Delta_j^{angle} + \Delta_k^{angle}) \right] + \exp \left[p_{tor4} (\Delta_j^{angle} + \Delta_k^{angle}) \right]} \tag{5.46}$$

5.2.12 Four Body Conjugation Energy

A four-body term is applied to conjugated molecules, similarly to equation (5.43).

$$E_{conj} = p_{cot1} \text{f12}(\text{BO}_{ij}, \text{BO}_{jk}, \text{BO}_{kl}) \left[1 + (\cos^2 \omega_{ijkl} - 1) \sin \Theta_{ijk} \sin \Theta_{jkl} \right] \tag{5.47}$$

$$\text{f12} = \exp \left[-p_{cot2} (\text{BO}_{ij} - \frac{3}{2})^2 \right] \exp \left[-p_{cot2} (\text{BO}_{jk} - \frac{3}{2})^2 \right] \exp \left[-p_{cot2} (\text{BO}_{kl} - \frac{3}{2})^2 \right] \tag{5.48}$$

5.2.13 Hydrogen Bond Energy

Hydrogen bonds are explicitly modeled in the ReaxFF model.

$$E_{hbond} = p_{hb1} [1 - \exp(-p_{hb2} \text{BO}_{xh})] \exp \left[-p_{hb3} \left(\frac{r_0^{hb}}{r_{hZ}} + \frac{r_{hZ}}{r_0^{hb}} - 2 \right) \right] \sin^8 \left(\frac{\Theta_{XhZ}}{2} \right) \quad (5.49)$$

5.2.14 van der Waals Energy

For the repulsive only van der Waals interaction, two contributions are possible. An inner wall model, when parameters `rcore2` and `acore2` are greater than zero and a shielding option, when the parameter γ_{vdw} is greater than 0.5. If the interaction is shielded:

$$E_{vdw} = \text{Tap}(r_{ij}) \epsilon_{vdw} \left\{ \exp \left[\alpha_{vdw} \left(1 - \frac{\text{f13}(r_{ij})}{r_{vdw}} \right) \right] - 2 \exp \left[\frac{\alpha_{vdw}}{2} \left(1 - \frac{\text{f13}(r_{ij})}{r_{vdw}} \right) \right] \right\} \quad (5.50)$$

$$\text{f13}(r_{ij}) = \left[r_{ij}^{p_{vdw1}} + \left(\frac{1}{\gamma_{vdw}} \right)^{p_{vdw1}} \right]^{1/p_{vdw1}} \quad (5.51)$$

$$(5.52)$$

If the interaction is unshielded:

$$E_{vdw} = \text{Tap}(r_{ij}) \epsilon_{vdw} \left\{ \exp \left[\alpha_{vdw} \left(1 - \frac{r_{ij}}{r_{vdw}} \right) \right] - 2 \exp \left[\frac{\alpha_{vdw}}{2} \left(1 - \frac{r_{ij}}{r_{vdw}} \right) \right] \right\} \quad (5.53)$$

If the inner wall method is included, an additional term is added.

$$E_{vdw-inner} = \text{Tap}(r_{ij}) \cdot \text{ecore2} \cdot \exp \left[\text{acore2} \left(1.0 - \left(\frac{r_{ij}}{\text{rcore2}} \right) \right) \right] \quad (5.54)$$

Where the taper ($\text{Tap}(r_{ij})$) function ensures that the interactions go smoothly to zero as the cutoff separation is approached.

$$\text{Tap}(r_{ij}) = \sum_{n=0}^7 \text{Tap}_n r_{ij}^n \text{ for } R_{low} = 0 \quad (5.55)$$

$$\text{Tap}_7 = 20/R_{\text{cut}}^7, \text{Tap}_6 = -70/R_{\text{cut}}^6, \text{Tap}_5 = 84/R_{\text{cut}}^5,$$

$$\text{Tap}_4 = -35/R_{\text{cut}}^4, \text{Tap}_3 = 0, \text{Tap}_2 = 0, \text{Tap}_1 = 0, \text{Tap}_0 = 1$$

5.2.15 Coulomb

The Coulombic interaction is shielded, and also includes the taper function.

$$E_{coulomb} = \text{Tap}(r_{ij}) \frac{q_i q_j}{\{r_{ij}^3 + \gamma_{qe qij}^{-3}\}^{1/3}} \quad (5.56)$$

5.2.16 QEq

Charge equilibration (QEq)¹⁴⁷ is a method of determining partial atomic charges, based on ionization potentials and electronegativities. Charges are adjusted to minimized the total electrostatic potential of the system

$$E_{es}(\{q_i\}) = \sum_i \chi q_i + \sum_{ij} q_i H_{ij} q_j \quad (5.57)$$

where q_i is the charge on atom i , $\chi_i = 1/2(IP + EA)$, IP is the ionization potential, and EA is the electron affinity. The term H_{ij} is given by

$$H_{ij} = \eta \delta_{ij} + \text{Tap}(r_{ij}) \frac{1}{\left(r_{ij}^3 + (\gamma_{qe q})^{-3}\right)^{1/3}} (1 - \delta_{ij}) \quad (5.58)$$

where η is the Coulomb repulsion between two electrons in the same orbital. QEq parameters in the ReaxFF forcefield are χ , η , and $\gamma_{qe q}$.

5.2.17 Combination Rules

Combination rules are applied to determine interaction parameters between differing atomic types, unless a specific value is specified in table 5.4. The arithmetic mean ($p_{ij} = (p_i + p_j)/2$) is used for For the bond radii (r_0^σ , r_0^π , $r_0^{\pi\pi}$), The geometric mean is used for ϵ_{vdw} , α_{vdw} , r_{vdw} , γ_{vdw} , and $\gamma_{qe q}$.

5.3 Parameters

Tables 5.1 through 5.7 list the interaction parameters which govern the behavior of the ReaxFF model. Included in these tables are the parameter name, a brief description, the address used for ParOpt optimization (with “x” where more than one address is required), the value determined after all optimization procedures (see section 5.5), and the equation where the parameter appears.

Table 5.1: ReaxFF global parameters. Global parameters were left constant during all optimization stages

| Name | Description | Address | Value | Eq.# |
|-------------|--|---------|----------|--------------|
| p_{boc1} | Overcoordination parameter | 1:1:0 | 50.000 | (5.13) |
| p_{boc2} | Overcoordination parameter | 1:2:0 | 9.5469 | (5.14) |
| p_{coa2} | Valency angle conjugation parameter | 1:3:0 | 1.6725 | (5.43) |
| p_{gp3} | Triple bond stabilisation parameter | 1:4:0 | 1.7224 | (5.30) |
| p_{gp4} | Triple bond stabilisation parameter | 1:5:0 | 6.8702 | (5.30) |
| p_{lp3} | C2-correction | 1:6:0 | 60.4850 | (5.29) |
| p_{ovun6} | Undercoordination parameter | 1:7:0 | 1.0588 | (5.33) |
| p_{gp7} | Triple bond stabilisation parameter | 1:8:0 | 4.6000 | (5.30) |
| p_{ovun7} | Undercoordination parameter | 1:9:0 | 12.1176 | (5.33) |
| p_{ovun8} | Undercoordination parameter | 1:10:0 | 13.3056 | (5.33) |
| p_{gp10} | Triple bond stabilisation energy | 1:11:0 | -40.0000 | (5.30) |
| R_{low} | Lower Taper-radius | 1:12:0 | 0.0000 | (5.55)(5.55) |
| R_{cut} | Upper Taper-radius | 1:13:0 | 10.0000 | (5.55)(5.55) |
| N/A | Not used | 1:14:0 | 2.8793 | — |
| p_{val6} | Valency undercoordination | 1:15:0 | 33.8667 | (5.36) |
| p_{lp1} | Valency angle/lone pair parameter | 1:16:0 | 6.0891 | (5.25) |
| p_{val9} | Valency angle | 1:17:0 | 1.0563 | (5.40) |
| p_{val10} | Valency angle parameter | 1:18:0 | 2.0384 | (5.38) |
| N/A | Not used | 1:19:0 | 6.1431 | — |
| p_{pen2} | Double bond/angle parameter | 1:20:0 | 6.9290 | (5.41) |
| p_{pen3} | Double bond/angle parameter: overcoord | 1:21:0 | 0.3989 | (5.42) |
| p_{pen4} | Double bond/angle parameter: overcoord | 1:22:0 | 3.9954 | (5.42) |
| N/A | Not used | 1:23:0 | -2.4837 | — |
| p_{tor2} | Torsion/BO parameter | 1:24:0 | 5.7796 | (5.45) |
| p_{tor3} | Torsion overcoordination | 1:25:0 | 10.0000 | (5.46) |
| p_{tor4} | Torsion overcoordination | 1:26:0 | 1.9487 | (5.46) |
| N/A | Conjugation 0 (not used) | 1:27:0 | -1.2327 | — |
| p_{cot2} | Conjugation | 1:28:0 | 2.1645 | (5.48) |
| p_{vdw1} | vdWaals shielding | 1:29:0 | 1.5591 | (5.51) |
| bo_{cut} | Cutoff for bond order (*100) | 1:30:0 | 0.1000 | (5.2) |
| p_{coa4} | Valency angle conjugation parameter | 1:31:0 | 1.7602 | (5.43) |
| p_{ovun4} | Overcoordination parameter | 1:32:0 | 0.6991 | (5.26) |
| p_{ovun3} | Overcoordination parameter | 1:33:0 | 50.0000 | (5.26) |
| p_{val8} | Valency/lone pair parameter | 1:34:0 | 1.8512 | (5.39) |
| N/A | Not used | 1:35:0 | 0.5000 | — |
| N/A | Not used | 1:36:0 | 20.0000 | — |
| N/A | Molecular energy (not used) | 1:37:0 | 5.0000 | — |
| p_{gp37} | Molecular energy (not used) | 1:38:0 | 0.0000 | (5.30) |
| p_{coa3} | Valency angle conjugation parameter | 1:39:0 | 2.6962 | (5.43) |

Table 5.2: ReaxFF single body parameters.

| Name | Description | Address | Values | | Eq.# |
|------------------|---------------------------------|---------|----------------------|-----------------------|---------------------|
| i | Atom Name | 2:x:0 | O | H | — |
| r_s | Sigma bond radius | 2:x:1 | 1.2599 [†] | 0.9033 [†] | (5.2) |
| Val | Valency | 2:x:2 | 2.0000 | 1.0000 | (5.5) (5.12) (5.27) |
| mass | Atomic mass | 2:x:3 | 15.9990 | 1.0080 | |
| r_{vdw} | van der Waals radius | 2:x:4 | XXX ^{†‡} | XXX ^{†‡} | (5.50) |
| ϵ_{vdw} | van der Waals energy scale | 2:x:5 | 0.0957 [†] | 0.1021 [†] | (5.50) |
| γ_{qeq} | gamma (QEq) | 2:x:6 | 1.3781* | 0.9930* | (5.56) (5.57) |
| r_0^π | Pi bond radius | 2:x:7 | 1.0737 [†] | -0.1000 | (5.3) |
| Val^e | Number of valence electrons | 2:x:8 | 6.0000 | 1.0000 | (5.27) |
| α_{vdw} | vdW parameter | 2:x:9 | XXX [‡] | XXX [‡] | (5.50) |
| γ_{vdw} | vdW shielding parameter | 2:x:10 | XXX ^{†‡} | XXX ^{†‡} | (5.51) |
| Val^{boc} | Valency | 2:x:11 | 4.0000 | 1.0000 | (5.6) |
| p_{ovun5} | Undercoordination energy scale | 2:x:12 | 37.5000 | 0.0000 | (5.33) |
| N/A | | 2:x:13 | 116.0768 | 121.1250 | — |
| χ | chi (QEq) | 2:x:14 | 11.7629* | 4.8078* | (5.57) |
| η | eta (QEq) | 2:x:15 | 9.7940* | 11.0610* | (5.57) |
| p_{hbond} | Hydrogen bond flag | 2:x:16 | 2.0000 | 1.0000 | |
| $r_0^{\pi\pi}$ | Pi Pi bond radius | 2:x:17 | 0.9065 [†] | -0.1000 | (5.4) |
| p_{lp2} | Lone-pair energy scale | 2:x:18 | 0.0055 [†] | 0.0000 | (5.23) |
| N/A | | 2:x:19 | 68.0152 | 55.1878 | — |
| bo^{131} | Bond order correction parameter | 2:x:20 | 3.4617 [†] | 2.8793 [†] | (5.18) |
| bo^{132} | Bond order correction parameter | 2:x:21 | 0.8433 [†] | 2.5070 [†] | (5.17) |
| bo^{133} | Bond order correction parameter | 2:x:22 | 0.0026 [†] | 0.0002 [†] | (5.19) |
| N/A | | 2:x:23 | 0.9745 | 1.0698 | — |
| N/A | | 2:x:24 | 0.0000 | 0.0000 | — |
| p_{ovun2} | Under / Overcoordination param | 2:x:25 | -3.6848 [†] | -20.5546 [†] | (5.31) |
| p_{val3} | Valence angle parameter | 2:x:26 | 2.7914 [†] | 4.2367 [†] | (5.35) |
| N/A | | 2:x:27 | 1.0493 | 1.0338 | — |
| Val^{val} | Valence angle valency | 2:x:28 | 4.0000 | 1.0000 | (5.37) |
| p_{val5} | Valence angle parameter | 2:x:29 | 2.9225 | 2.8793 | (5.36) |
| rcore2 | vdW shielding parameter | 2:x:30 | 0.0000 | 0.0000 | (5.54) |
| ecore2 | vdW shielding parameter | 2:x:31 | 0.0000 | 0.0000 | (5.54) |
| acore2 | vdW shielding parameter | 2:x:32 | 0.0000 | 0.0000 | (5.54) |

Table 5.3: ReaxFF bond parameters.

| Name | Description | Address | Values | | | Eq. |
|----------------|----------------------------------|---------|-----------------------|-----------------------|-----------------------|---------------------|
| i | Atom index 1 | 3:x:0 | O | O | H | |
| j | Atom index 2 | 3:x:1 | H | O | H | |
| D_e^σ | Sigma bond energy scale | 3:x:2 | 138.1391 [†] | 132.7956 [†] | 145.4878 [†] | (5.22) |
| D_e^π | Pi bond energy scale | 3:x:3 | 0.0000 | 155.6151 [†] | 0.0000 | (5.22) |
| $D_e^{\pi\pi}$ | Pi-Pi bond energy scale | 3:x:4 | 0.0000 | 57.4716 [†] | 0.0000 | (5.22) |
| p_{be1} | Bond energy parameter | 3:x:5 | -0.4068 [†] | 0.2872 [†] | -0.3877 [†] | (5.22) |
| p_{bo5} | Bond energy parameter | 3:x:6 | 0.0000 | -0.2502 [†] | 0.0000 | (5.4) |
| v13cor | 1-3 bo correction param | 3:x:7 | 1.0000 | 1.0000 | 1.0000 | (5.15)(5.16) |
| p_{bo6} | Uncorrected bo parameter | 3:x:8 | 6.0000 | 34.9854 [†] | 6.0000 | (5.4) |
| p_{ovun1} | Overcoordination energy param | 3:x:9 | 0.5587 [†] | 1.6931 [†] | 0.7021 [†] | (5.31) |
| p_{be2} | Bond energy parameter | 3:x:10 | 3.9899 [†] | 0.7157 [†] | 7.8569 [†] | (5.22) |
| p_{bo3} | Uncorrected bond order parameter | 3:x:11 | 1.0000 | -0.9751 [†] | 1.0000 | (5.3) |
| p_{bo4} | Uncorrected bond order parameter | 3:x:12 | 0.0000 | 9.5955 [†] | 0.0000 | (5.3) |
| N/A | | 3:x:13 | 0.0000 | 1.0000 | 1.0000 | — |
| p_{bo1} | Uncorrected bond order parameter | 3:x:14 | -0.0940 [†] | -0.9577 [†] | -0.0724 [†] | (5.2) |
| p_{bo2} | Uncorrected bond order parameter | 3:x:15 | 4.6604 [†] | 5.2705 [†] | 5.6139 [†] | (5.2) |
| ovc | Overcoordination parameter | 3:x:16 | 0.0000 | 1.0000 | 0.0000 | (5.12) (5.13)(5.14) |
| N/A | | 3:x:17 | 0.0000 | 0.0000 | 0.0000 | — |

Table 5.4: ReaxFF off-diagonal parameters. These parameters are used to replace the combination rules for the mixing of single body parameters with an explicit value.

| Name | Description | Address | Value | Eq.# |
|------------------|-------------------------|---------|---------------------|--------|
| i | Atom index 1 | 4:1:0 | O | — |
| j | Atom index 2 | 4:1:1 | H | — |
| ϵ_{vdw} | van der Waals energy | 4:1:2 | 0.0329 [†] | (5.50) |
| r_{vdw} | van der Waals radius | 4:1:3 | XXX ^{†‡} | (5.50) |
| α_{vdw} | van der Waals parameter | 4:1:4 | XXX ^{†‡} | (5.50) |
| r_0^σ | Sigma bond radius | 4:1:5 | 0.9447 [†] | (5.2) |
| r_0^π | Pi bond radius | 4:1:6 | -1.0000 | (5.3) |
| $r_0^{\pi\pi}$ | Pi-Pi bond radius | 4:1:7 | -1.0000 | (5.4) |

Table 5.5: ReaxFF three body parameters.

| Name | Description | Address | Values | | | | | | Eq.# |
|----------------|----------------------------------|---------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|--------|
| i | Atom index 1 | 5:x:0 | H | H | O | O | H | H | — |
| j | Atom index 2 | 5:x:1 | O | O | O | H | H | H | — |
| k | Atom index 2 | 5:x:2 | H | O | O | O | O | O | — |
| $\theta_{0,0}$ | Equilibrium angle | 5:x:3 | 88.3772 [†] | 74.0264 [†] | 90.8579 [†] | 13.6786 [†] | 18.6704 [†] | 18.6704 [†] | (5.38) |
| p_{val1} | Valence angle parameter | 5:x:4 | 8.5670 [†] | 47.4820 [†] | 16.7630 [†] | 15.4878 [†] | 10.7582 [†] | 10.7582 [†] | (5.34) |
| p_{val2} | Valence angle parameter | 5:x:5 | 2.6020 [†] | 2.0055 [†] | 1.8881 [†] | 2.8988 [†] | 2.0606 [†] | 2.0606 [†] | (5.34) |
| p_{coa1} | Three-body conjugation parameter | 5:x:6 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | (5.43) |
| p_{val7} | Valence angle parameter | 5:x:7 | 2.7637 [†] | 1.0000 | 3.0664 [†] | 0.0000 | 0.0000 | 0.0000 | (5.36) |
| p_{pen1} | Energy penalty parameter | 5:x:8 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | (5.41) |
| p_{val4} | Valence angle parameter | 5:x:9 | 1.5938 [†] | 1.2072 [†] | 1.2721 [†] | 2.7272 [†] | 0.5567 [†] | 0.5567 [†] | (5.35) |

Table 5.6: ReaxFF four body parameters.

| Name | Description | Address | Values | | | | | | | | Eq.# |
|------------|---------------------------------|---------|-----------------------|----------------------|----------------------|----------|--------|--------|---------|---------|--------|
| j | Atom index 1 | 6:x:0 | H | H | x | O | x | x | x | x | — |
| k | Atom index 2 | 6:x:1 | O | O | O | O | H | H | H | H | — |
| m | Atom index 3 | 6:x:2 | O | O | O | O | H | H | O | O | — |
| n | Atom index 4 | 6:x:3 | H | O | x | O | x | x | x | x | — |
| V_1 | Four body torsion parameter | 6:x:4 | -2.1511 [†] | -2.3776 [†] | 0.4910 [†] | -2.5000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | (5.44) |
| V_2 | Four body torsion parameter | 6:x:5 | -21.8023 [†] | 43.3875 [†] | 23.4223 [†] | -25.0000 | 0.0000 | 0.0000 | 0.1000 | 0.1000 | (5.44) |
| V_3 | Four body torsion parameter | 6:x:6 | -1.2196 [†] | -0.6786 [†] | 1.1104 [†] | 1.0000 | 0.0000 | 0.0000 | 0.0200 | 0.0200 | (5.44) |
| p_{tor1} | Four body torsion parameter | 6:x:7 | -2.3032 [†] | -6.7705 [†] | -4.9934 [†] | -2.5000 | 0.0000 | 0.0000 | -2.5415 | -2.5415 | (5.44) |
| p_{cot1} | Four body conjugation parameter | 6:x:8 | -0.3749 [†] | -1.3268 [†] | -1.0863 [†] | -0.9000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | (5.47) |
| N/A | | 6:x:9 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | — |
| N/A | | 6:x:10 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | — |

Table 5.7: ReaxFF hydrogen bond parameters.

| Hydrogen Bond Parameters | | | | |
|--------------------------|-------------------------|---------|----------------------|--------|
| Name | Description | Address | Value | Eq.# |
| i | Atom index 1 | 7:1:0 | O | — |
| j | Atom index 2 | 7:1:1 | H | — |
| k | Atom index 2 | 7:1:2 | O | — |
| r_0^{hb} | Hydrogen bond radius | 7:1:3 | 2.1715 [†] | (5.49) |
| p_{hb1} | Hydrogen bond energy | 7:1:4 | -3.9033 [†] | (5.49) |
| p_{hb2} | Hydrogen bond parameter | 7:1:5 | 1.2684 [†] | (5.49) |
| p_{hb3} | Hydrogen bond parameter | 7:1:6 | 16.5854 [†] | (5.49) |

5.4 Target Data

In order to reproduce water at an accurate level, systems for the calculation of quantum energies were produced using DFT and the B3LYP method with the 6-311++g** basis set. Systems produced included a profile of the ground state water dimer as a function of the oxygen-oxygen distance (*dimer_profile*), from 2.0 Å to 4.0 Å. Hydrogen peroxide (*peroxide_profile*) with perturbations about the ground state in the oxygen-oxygen and oxygen-hydrogen bond, bond angles, and dihedral angle was also performed. A profile of a single water molecule (*water_profile*) with bond distances varied symmetrically between 0.9 Å and 1.0 Å and angle between 100° and 108° was analyzed. For these three sets of geometries, only single point calculations were performed. Systems with optimized targets were geometry optimized in the Serial-Reax software using a steepest-descent algorithm.¹⁴⁵ In addition to the profile systems, systems consisting of several molecules were geometry optimized. Molecular hydrogen with 1, 2 and 4 molecules (*hydrogen_clust*), hydrogen peroxide from 1 to 6 molecules (*peroxide_clust*), molecular oxygen with 1, 2, 4, and 7 molecules (*oxygen_clust*), and water with 1 to 34 molecules (*water_clust*) were analyzed. For the calculation of partial charges from the full electronic wavefunction, the CHELPG method was used. The method assigns partial charges in order to fit the electrostatic potential, while also reproducing the same total dipole moment of the system. All *ab initio* quantum calculations for this chapter were performed using Gaussian 09.¹⁴⁸

In addition to quantum calculations performed for this work, data which was generated for the optimization of the iAMOEBA model was also used. The iAMOEBA model is a simplification of the AMOEBA model with the additional assumption of direct polarization.¹⁴⁹ The *ab initio* data used for iAMOEBA optimization which was also used in this work include both gas phase and liquid phase data. The liquid phase data consists of approximately 42,000 systems with water clusters

ranging in size from 2 to 22 molecules which were extracted from AMOEBA simulations ranging from 249.15 K to 373.15 K. These systems were analyzed using dual basis RI-MP2/heavy-aug-cc-pVTZ level of theory.¹⁴⁹ Gas phase clusters calculated using the Møller-Plesset perturbation theory and the coupled cluster (CCSDT) method.¹⁵⁰

5.4.1 Partial Atomic Charges

Since the QEq parameters χ , η , and γ , are independent of all other parameters, these were optimized separately. Since the relative differences between atomic species in parameter values, not their absolute values, determine the partial charges calculated from the QEq method, systems with a reference atom whose parameters are not varied during optimization were added to the training set. Systems with up to four methane molecules, and systems combining methane and methanol were included. Additionally, systems considered for the charge optimization phase were *dimer_profile*, *peroxide_profile*, *water_profile*, *peroxide_clust*, and *water_clust*. Since no energy term were being optimized, only single point calculations of atomic charges using the serial version of the PuReMD software¹⁴⁵ were performed during this optimization procedure. The QEq method is unable to assign partial charges to systems with only one atom type, and therefore cannot polarize molecules such as ozone. Since no charge would be distributed in systems with only one atom type, systems from the groups *oxygen_clust* and *hydrogen_clust* were not used in the charge optimization phase.

5.4.2 *ab initio* Energies

For the optimization of the large parameter set against quantum calculations, systems which were optimized via *ab initio* calculation were also optimized using the serial PuReMD software.¹⁴⁵ These systems include *peroxide_clust*, and *water_clust*, *oxygen_clust*, and *hydrogen_clust* and systems included from the iAMOEBA training set. For the non-energy minimized systems, only single point calculation were performed. These systems include *dimer_profile*, *water_profile*, and *peroxide_profile* and the majority of the the iAMOEBA training set.

5.4.3 Empirical Data

To accurately reproduce macroscale water behavior, ReaxFF parameters were optimized to match experimental data, including densities, diffusion coefficients, correlation functions, bond structure,

and average electrostatic properties. In order to ensure that the optimized model reproduces accurate oxygen and hydrogen behavior in differing chemical environments, system consisting of peroxide solute in water solvent of varying concentration (0%, 1.91%, 24.58%, and 47.01% weight percent) were performed. Systems were simulated for 20 ps in the initial stages of optimization. As the optimization proceed, systems appeared to require more time for equilibration, so run times were extended to 25 ps. A timestep of 0.25 fs was used throughout. Charges were recalculated using the QEq procedure at every step. Systems were maintained at a constant temperature (298.15 K) and pressure (1 atm) ensemble using the Berendsen thermostat and barostat respectively. All systems were simulated using the parallel ReaxFF implementation PuReMD.²⁰⁸ This implementation of ReaxFF has excellent scaling behavior and employs many advanced numerical and computation techniques to achieve very impressive performance.

Densities: Densities for all four systems were calculated from time averages over the final half of the simulation (either 10 ps or 25 ps depending on the total simulation length). Errors in each density were calculated from

$$f_{\rho} = \frac{(\rho_{exp.} - \rho_{reax})^2}{\rho_{exp}} \quad (5.59)$$

Diffusion Coefficient: The coefficient of diffusion for water was calculated by determining the mean-squared displacement as a function of time for oxygen atoms which were part of intact water molecules.

Radial Distribution Function: The radial distribution function (RDF) of two species in a system is calculated from

$$g(r) = \frac{N(r)}{4\pi r^2 \rho dr}, \quad (5.60)$$

where $N(r)$ is the unnormalized count of atoms as a function of distance and ρ is density over the entire system. The RDF is a ratio of the density in a shell of thickness dr and radius r to the homogeneous average density and indicates areas of increased probability to find species. Experimental target data were determined using empirical structure refinement (EPSR) to refine model of liquid water to reproduce high accuracy scattering measurements.¹⁵¹ Com-

parison of RDFs was performed using a linear weight function as follows

$$f_{rdf} = \frac{1}{r_{max}} \int_0^{r_{max}} \frac{(g_{exp}(r) - g_{reax}(r))^2}{g_{exp}(r)} W(r) dr \quad (5.61)$$

$$W(r) = m * x + b \quad (5.62)$$

$$(5.63)$$

where $m = -9.32$ and $b = 50$ are coefficients for the linear weighting function which was chosen to give higher importance to the short range correlations in the system.

Dipole Moment: Dipole moment was calculated as a time average over intact water molecules using

$$\mu = \sum_i q_i (x_i - x_{ref}) \quad (5.64)$$

where the sum is over all bonded atoms (i) in a molecule, x_i and q_i are the position vector and charge of the atom respectively, and x_{ref} is a reference vector. For a neutral molecule, the choice of reference vector is arbitrary. Since the constraint on QEq in the ReaxFF is on total system neutrality and not on molecular neutrality, this assumption does not always hold. Therefore, the reference vector was chosen as the location of the oxygen atom to minimized artifacts. In addition, only molecules with a net charge $Q = \sum_i q_i < 0.001e$ were considered in the average. This averaged value was compared to the experimental value in the same fashion as in equation (5.59).

Bond Distance: Due to the lack of the bonded peak in the experimental RDF, the experimental value for the oxygen–hydrogen bond distance was added to the parameter set. A comparison of the same form as equation (5.59) was used for the target function value component.

5.5 Optimization

An optimization of all parameters against all data would be prohibitively expensive. To simplify the optimization procedure, parameters and data were first split into non overlapping groups. Atomic charges determined from a single point calculation of QEq, depend only on the QEq specific

parameters (η , γ_{qeq} , and χ). Due to this separability, charge parameter sets were first independently optimized with only partial charge calculated from *ab initio* methods as the target. Once optimal values for these parameters were determined, parameters governing bond order, bonded energies, and non-bonded energies were optimized to reproduce *ab initio* single point and geometry optimized energy calculations. This is the largest set of parameters used for optimization. Due to the long time requirements for the simulation of systems from which ensemble averages can be taken, only the non-bonding van der Waals terms were optimized to match experimental observations.

All optimization was divided into three parts: QEq (*) partial charges, single point energies (†), and empirical macroscale properties(‡). Tables 5.1 through 5.7 list the parameters used for each phase.

Figure 5.1 shows the full optimization of the QEq parameters. Figure 5.2 shows the full optimization of the large parameter set to reproduce the *ab initio* energy data. The optimization of the resulting parameter sets after the first two phases to match experimental properties is shown in figure 5.3.

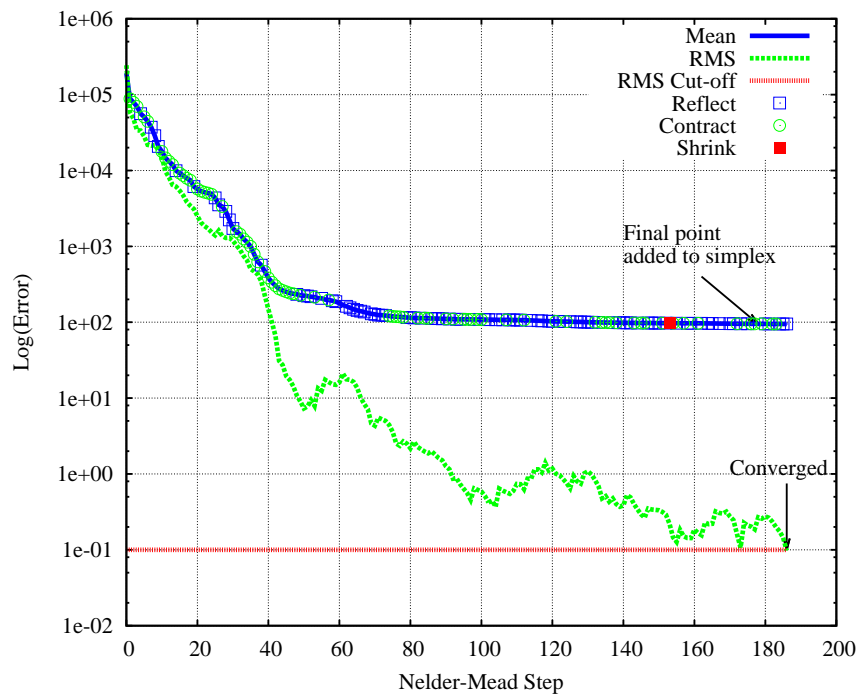


Figure 5.1: Optimization of charge equilibration (QEq) parameters to reproduce *ab initio* charges.

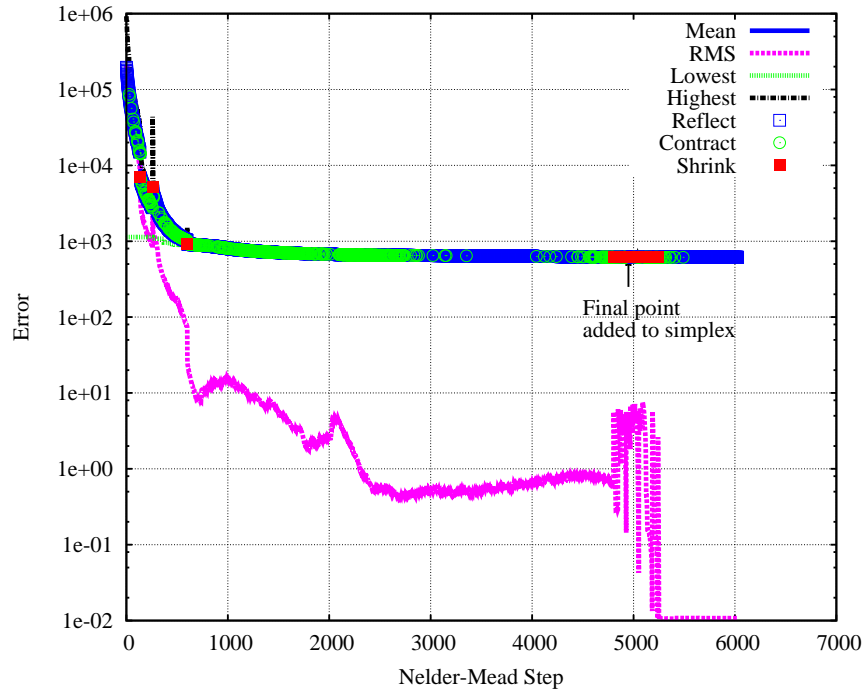


Figure 5.2: Optimization of bond order, bonded energy, and non-bonded energy terms to reproduce single point *ab initio* energy calculations.

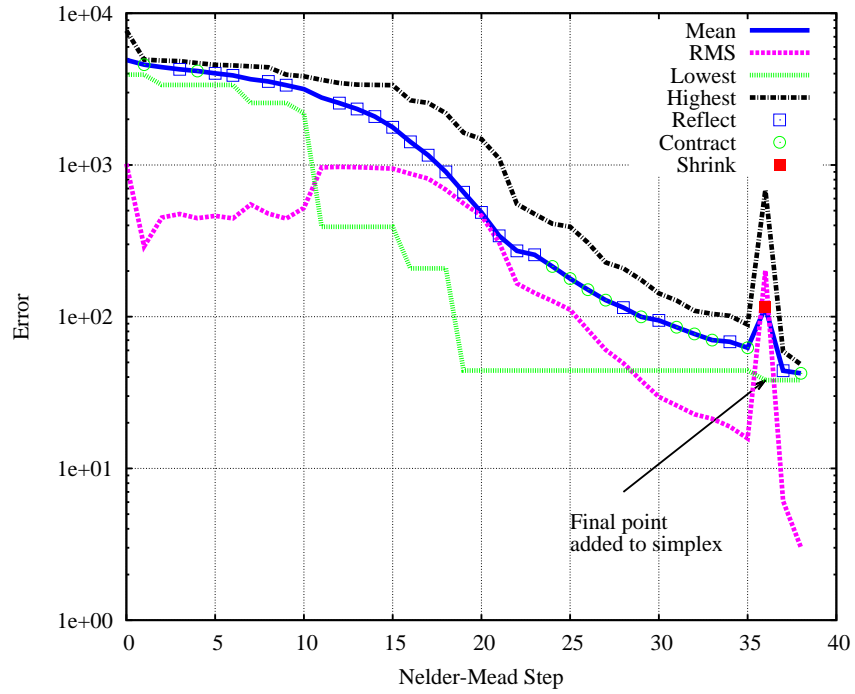


Figure 5.3: Optimization of van der Waals parameters to reproduce macroscopic experimental observables in different mixtures of hydrogen peroxide and water.

5.6 Conclusion

A multistage optimization of the ReaxFF model for oxygen and hydrogen interactions has been performed in order to accurately model water and hydrogen peroxide under physiological conditions. The first stage of this procedure was the tuning of charge equilibration parameters in order to match *ab initio* derived partial atomic charges. Following this, an iterative procedure was undertaken, whereby the model was adjusted to match quantum mechanical data and empirical observations in a consistent manner. To this end, parameters were optimized to match *ab initio* data. The local area of the resulting optimal parameter set was then explored to match experimental measurements. This procedure was then iterated, ensuring that both theoretical and experimental predictions are reproduced by the model.

Chapter 6

Conclusion

In order to solve the optimization problem for a general target function defined on parameter space, the software package ParOpt was developed. This software provides a flexible framework for approaching high dimensional non-linear optimizations. In particular, development of ParOpt has focused on the optimization of molecular models. These models generally have computationally costly target functions, which can also suffer from numerical instability, stochasticity, and a lack of unique solution. Because of these function properties, the Nelder-Mead simplex method implemented within the ParOpt framework has been utilized. This method is well suited to the problem of molecular model optimization due to its ability to greatly reduce target function values in initial steps, its lack of dependence on the target function gradient, and its performance on stochastic problems. Applications of software package ParOpt and the method Nelder-Mead were presented, in the order of the size of the parameter space.

Optimization of a coarse-grained model for water and ions was presented. The model employs a modified Morse potential which is more computationally expensive than the conventionally used Lennard-Jones potential. The function form of these interactions gives the model six parameters for each interaction, where the Lennard-Jones potential has only two. In effect, the usage of this potential trades simulation degrees of freedom (positions and velocities of atoms) for optimization degrees of freedom (free parameters for tuning). Thus, the optimization problem, which is only solved once, has increased complexity, but the resulting simulations, which will be used in many different instances, have reduced complexity. These models were tuned primarily against experimental data, including densities, diffusion coefficients, surface tension, and free energies of solvation. Thus,

small scale interactions are adjusted to reproduce statistical properties of the system.

Determination of lipid bilayer structure from phase space scattering data requires solving the inverse problem $G(m) = d$, where the scattering data d , is known. A model m is generated which represents the underlying, unknown, state of the system. The transformation (G) of this model must match the original observed data. This problem can also be posed as an optimization problem, where the parameters of the model are the variables to be tuned, and the difference between the model predication and the observations determines the target function. This approach was applied to single component lipid bilayers of various lipid types in chapter 4. A new atomic density profile (ADP) model was developed which makes fewer assumptions regarding the underlying atomic structure. Since the the optimization procedure is underdetermined due to a lack of data and an overabundance of model parameters, a statistical sampling of converged models was analyzed. This approach yielded not only the bilayer structure, but also a metric on how precisely bilayer features are resolved by the data. In future work, the model and methods can be applied to more complex systems than have been possible with existing approaches.

A reactive potential such as ReaxFF for the simulation of biological systems using classical molecular dynamics would provide a powerful tool. A reactive model of biologically relevant atomic interactions could be applied to the statistical analysis of many important systems, such as oxidative damage of cellular components by the leakage of metabolic processes. Toward that goal, a reactive model of water has been optimized. The ReaxFF model was tuned in a multiphase procedure. First the charge equilibration (QEq) parameters were optimized to reproduce partial charges derived from *ab initio* calculations. Parameters governing bond order, bonded interactions, and non-bonded van der Waals interactions were then tuned to reproduce a large set of energies computed via quantum chemical methods. This model was further refined in order to reproduce large scale experimentally determined properties of water and hydrogen peroxide systems. These final two steps are performed iteratively, ensuring the simultaneous fitting two both *ab initio* calculations and experimental large scale properties.

The ParOpt software has proven to be a useful tool in the development of molecular models on several different time and length scales. Each of these models and the associated target functions and parameter spaces present different challenges. Some target function (see chapter 3 and chapter 5) involved long time simulation in order to compare macroscopic model prediction with

experimnt. Other target functions (see chapter 4 and chapter 5) invovled faster single point calculations. The characteristics of these target functions also varied. Parameter space sizes also varied, from less than 10 parameters to nearly 100. ParOpt, from a software design perspective, has been constructed with enough generality to allow adjustment to solve each problem.

Bibliography

- [1] P. Allen and D. Tildesley. *Computer Simulation of Liquids*. Clarendon Press, 1989.
- [2] D. Frenkel and B. Smit. *Understanding Molecular Simulation*. Second. Academic Press, 2002.
- [3] D. Rapaport. *The Art of Molecular Dynamics Simulation*. Cambridge University Press, 2004.
- [4] A. Szabo and N. Ostlund. *Modern Quantum Chemistry: Introduction to Advanced Electronic Structure Theory*. Dover Publications, 1996.
- [5] S. Koneshan, J. C. Rasaiah, R. M. Lynden-Bell, and S. H. Lee. “Solvent Structure, Dynamics, and Ion Mobility in Aqueous Solutions at 25 °C”. *J. Phys. Chem. B* 102 (1998), 4193–4204.
- [6] C. L. Pierri, A. De Grassi, and A. Turi. “Lattices for ab initio protein structure prediction”. 73 (2008), 351–361.
- [7] J.-W. Shen, C. Li, N. F. van der Vegt, and C. Peter. “Transferability of Coarse Grained Potentials: Implicit Solvent Models for Hydrated Ions”. *J. Chem. Theory Comput.* 7 (2011), 1916–1927.
- [8] E. Tekin and I. Sabuncuoglu. “Simulation Optimization: A Comprehensive Review on Theory and Applications”. *IEEE Trans.* 36 (2004), 1067–1081.
- [9] T. Terwilliger and J. Berendzen. “Automated MAD and MIR structure solution”. *Acta Crystallogr. Sect. D Biol. Crystallogr.* 55 (1999), 849–861.
- [10] I. Lyubimov and M. G. Guenza. “First-principle approach to rescale the dynamics of simulated coarse-grained macromolecular liquids”. *Phys. Rev. E* 84 (3 2011), 031801.
- [11] W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey, and M. L. Klein. “Comparison of Simple Potential Functions for Simulating Liquid Water”. *J. Chem. Phys.* 79 (1983), 926–935.
- [12] G. C. Cawley and N. L. C. Talbot. “Preventing Over-Fitting During Model Selection via Bayesian Regularisation of the Hyper-Parameters”. *J. Mach. Learn. Res.* 8 (2007), 841–861.
- [13] R. B. Edmonson, H. L. Olsen, and E. L. Gayhart. “Application of Ideal Gas Theory to the Gaseous Expansion from an Electric Spark”. 25 (1954), 1008–1013.
- [14] D. Chandler, J. D. Weeks, and H. C. Andersen. “Van Der Waals Picture of Liquids, Solids, and Phase Transformations”. *Science* 220 (1983), pages.
- [15] J.-P. Hansen and I. McDonald. *Theory of Simple Liquids*. Elsevier Science, 2006.
- [16] M. D. Rintoul and S. Torquato. “Metastability and Crystallization in Hard-Sphere Systems”. *Phys. Rev. Lett.* 77 (1996), 4198–4201.
- [17] J. Israelachvili. *Intermolecular & Surface Forces*. Second. Academic Press Inc., 1992.
- [18] R. K. Pathria. *Statistical mechanics*. Butterworth-Heinemann, 1996.

- [19] H. Longuet-Higgins and B. Wisdom. “A rigid sphere model for the melting of argon”. *Mol. Phys.* 8 (1964), 549–556.
- [20] R. Pengyu, J. Chun, D. G. Thomas, M. J. Schnieders, M. Marucho, J. Zhang, and N. A. Baker. “Biomolecular electrostatics and solvation : a computational perspective”. *Q. Rev. Biophys.* 45 (4 2012), 427–491.
- [21] J. E. Jones. “On the Determination of Molecular Fields. II. From the Equation of State of a Gas”. *Proc. R. Soc. A* 106 (1924), 463–477.
- [22] P. M. Morse. “Diatomic Molecules According to the Wave Mechanics. II. Vibrational Levels”. *Phys. Rev.* 34 (1 1929), 57–64.
- [207] J. C. Fogarty, S.-W. Chiu, P. Kirby, E. Jakobsson, and S. A. Pandit. “Automated Optimization of Water–Water Interaction Parameters for a Coarse-Grained Model”. *J. Phys. Chem. B* 118 (2014), 1603–1611.
- [24] P. Ball. “Water - An Enduring Mystery”. *Nature* 452 (2008), 291–292.
- [25] K. A. Dill, T. M. Truskett, V. Vlachy, and B. Hribar-Lee. “Modeling Water, the Hydrophobic Effect, and Ion Solvation”. *Annu. Rev. Biophys. Biomol. Struct.* 34 (2005), 173–199.
- [26] A. Onufriev. “Implicit Solvent Models in Molecular Dynamics Simulations: A Brief Overview”. In: *Annual Reports in Computational Chemistry, Vol. 4*. Ed. by R. A. Wheeler and D. C. Spellmeyer. Vol. 4. Elsevier, 2008, 125–137.
- [27] V. Molinero and E. B. Moore. “Water Modeled As an Intermediate Element between Carbon and Silicon”. *J. Phys. Chem. B* 113 (2009), 4008–4016.
- [28] J. C. Shelley, M. Y. Shelley, R. C. Reeder, S. Bandyopadhyay, and M. L. Klein. “A Coarse Grain Model for Phospholipid Simulations”. *J. Phys. Chem. B* 105 (2001), 4464–4470.
- [29] S. Marrink, A. de Vries, and A. Mark. “Coarse Grained Model for Semiquantitative Lipid Simulations”. *J. Phys. Chem. B* 108 (2004), 750–760.
- [30] S. J. Marrink, H. J. Risselada, S. Yefimov, D. P. Tieleman, and A. H. de Vries. “The MARTINI Force Field: Coarse Grained Model for Biomolecular Simulations”. *J. Phys. Chem. B* 111 (2007), 7812–7824.
- [31] W. Shinoda, R. DeVane, and M. L. Klein. “Multi-property Fitting and Parameterization of a Coarse Grained Model for Aqueous Surfactants”. *Mol. Simulat.* 33 (2007), 27–36.
- [32] S.-W. Chiu, H. L. Scott, and E. Jakobsson. “A Coarse-Grained Model Based on Morse Potential for Water and n-Alkanes”. *J. Chem. Theory Comput.* 6 (2010), 851–863.
- [33] K. R. Hadley and C. McCabe. “On the Investigation of Coarse-Grained Models for Water: Balancing Computational Efficiency and the Retention of Structural Properties”. *J. Phys. Chem. B* 114 (2010), 4590–4599.
- [34] J. D. Bernal and R. H. Fowler. “A Theory of Water and Ionic Solution, with Particular Reference to Hydrogen and Hydroxyl Ions”. *J. Chem. Phys.* 1 (1933), 515–548.
- [35] M. W. Mahoney and W. L. Jorgensen. “A Five-site Model for Liquid Water and the Reproduction of the Density Anomaly by Rigid, Nonpolarizable Potential Functions”. *J. Chem. Phys.* 112 (2000), 8910–8922.
- [36] J. Nocedal and S. Wright. *Numerical Optimization*. Springer, 2006.
- [37] M. Di Pierro and R. Elber. “Automated Optimization of Potential Parameters”. *J. Chem. Theory Comput.* 9 (2013), 3311–3320.

- [38] C. Wang, N. Kuzu, and Y. Tamai. “Molecular dynamics study on surface structure of α -SiO₂ by charge equilibration method”. *J. Non-Cryst. Solids* 218 (2003), 131–141.
- [39] R. Fletcher. *Practical methods of optimization*. Wiley, 1987.
- [40] A. P. Gurson. “Simplex Search Behavior in Nonlinear Optimization”. PhD thesis. College of William & Mary, 2000.
- [41] S. Sivanandam and S. Deepa. *Genetic Algorithm Optimization Problems*. Springer, 2008.
- [42] E. Elbeltagi, T. Hegazy, and D. Grierson. “Comparison among five evolutionary-based optimization algorithms”. *Adv. Eng. Inform.* 19 (2005), 43–53.
- [43] S. H. Brooks and M. R. Mickey. “Optimum Estimation of Gradient Direction in Steepest Ascent Experiments”. *Biometrics* 17 (1961), pages.
- [44] M. Wetter et al. “GenOpt-A generic optimization program”. In: *Seventh International IBPSA Conference, Rio de Janeiro*. 2001, 601–608.
- [45] J. A. Nelder and R. Mead. “A Simplex Method for Function Minimization”. *Comput. J.* 7 (1965), 308–313.
- [46] A. C. T. van Duin, J. M. A. Baas, and B. van de Graaf. “Delft molecular mechanics: a new approach to hydrocarbon force fields. Inclusion of a geometry-dependent charge calculation”. *J. Chem. Soc., Faraday Trans.* 90 (19 1994), 2881–2895.
- [47] V. J. Torczon. “Multi-Directional Search: A Direct Search Algorithm for Parallel Machines”. PhD thesis. Rice University, 1989.
- [48] D. J. Woods. “An Interactive Approach for Solving Multi-objective Optimization Problems”. PhD thesis. Rice University, 1985.
- [49] K. McKinnion. “Convergence of the Nelder-Mead simplex method to a nonstationary point”. *SIAM J. Sci. Comput.* 9 (1998), 148–158.
- [50] J. Lagarias, J. Reeds, M. Wright, and P. Wright. “Convergence Properties of the Nelder–Mead Simplex Method in Low Dimensions”. *SIAM J. Sci. Comput.* 9 (1998), 112–147.
- [51] B. Guillot. “A Reappraisal of What we Have Learnt During Three Decades of Computer Simulations on Water”. *J. Mol. Liq.* 101 (2002), 219–260.
- [52] L. Darré, M. R. Machado, and S. Pantano. “Coarse-grained Models of Water”. *WIREs Computational Molecular Science* 2 (2012), 921–930.
- [53] F. Sedlmeier, J. Janecek, C. Sendner, L. Bocquet, R. Netz, and D. Horinek. “Water at Polar and Nonpolar Solid Walls (Review)”. *Biointerphases* 3 (2008), FC23–FC39.
- [54] M. L. Berkowitz, D. L. Bostick, and S. Pandit. “Aqueous Solutions next to Phospholipid Membrane Surfaces: Insights from Simulations”. *Chem. Rev.* 106 (2006), 1527–39.
- [55] R. O. Dror, R. M. Dirks, J. P. Grossman, H. Xu, and D. E. Shaw. “Biomolecular Simulation: A Computational Microscope for Molecular Biology”. *Annu. Rev. Biophys.* 41 (2012), 429–52.
- [56] F. Carrión, A. D. L. Maza, and J. Parra. “The Influence of Ionic Strength and Lipid Bilayer Charge on the Stability of Liposomes”. *J. Colloid Interface Sci.* 164 (1994), 78–87.
- [57] S. Garcia-Manyes, G. Oncins, and F. Sanz. “Effect of Ion-Binding and Chemical Phospholipid Structure on the Nanomechanics of Lipid Bilayers Studied by Force Spectroscopy”. *Biophys. J.* 89 (2005), 1812–1826.

- [58] A. J. Schoeffler, C. R. Ruiz, A. M. Joubert, X. Yang, and L. V. J. “Salt Modulates the Stability and Lipid Binding Affinity of the Adipocyte Lipid-binding Proteins”. *J. Biol. Chem.* 35 (2003), 33268–33275.
- [59] D. J. Klein, P. B. Moore, and T. A. Steitz. “The contribution of metal ions to the structural stability of the large ribosomal subunit”. *RNA* 10 (2004), 1366–1379.
- [60] K. Szalewicz, C. Leforestier, and A. van der Avoird. “Towards the Complete Understanding of Water by a First-principles Computational Approach”. *Chem. Phys. Lett.* 482 (2009), 1–14.
- [61] I.-F. W. Kuo, C. J. Mundy, M. J. McGrath, J. I. Siepmann, J. VandeVondele, M. Sprik, J. Hutter, B. Chen, M. L. Klein, F. Mohamed, M. Krack, and M. Parrinello. “Liquid Water from First Principles Investigation of Different Sampling Approaches”. *J. Phys. Chem. B* 108 (2004), 12990–12998.
- [62] D. Feller, E. D. Glendening, D. E. Woon, and M. W. Feyereisen. “An extended basis set ab initio study of alkali metal cation–water clusters”. *J. Chem. Phys.* 103 (1995), 3526–3542.
- [63] S. S. Xantheas. “Quantitative Description of Hydrogen Bonding in Chloride Water Clusters”. *J. Phys. Chem.* 100 (1996), 9703–9713.
- [64] W. L. Jorgensen. “Revised TIPS for Simulations of Liquid Water and Aqueous Solutions”. *J. Chem. Phys.* 77 (1982), 4156–4163.
- [65] H. Berendsen, J. Postma, W. Gunsteren, and J. Hermans. “Interaction Models for Water in Relation to Protein Hydration”. In: *Intermolecular Forces*. Ed. by B. Pullman. Vol. 14. Springer Netherlands, 1981, 331–342.
- [66] H. J. C. Berendsen, J. R. Grigera, and T. P. Straatsma. “The Missing Term in Effective Pair Potentials”. *J. Phys. Chem.* 91 (1987), 6269–6271.
- [67] W. Jorgensen. “Transferable Intermolecular Potential Functions for Water, Alcohols, and Ethers. Application to Liquid Water”. *J. Am. Chem. Soc.* 103 (1981), 335–340.
- [68] B. Schropp and P. Tavan. “The Polarizability of Point-Polarizable Water Models: Density Functional Theory/Molecular Mechanics Results”. *J. Phys. Chem. B* 112 (2008), 6233–6240.
- [69] I. Kalcher, D. Horinek, R. Netz, and J. Dzubiella. “Ion specific correlations in bulk and at biointerfaces”. *J. Phys.: Condens. Matter* (2009), 424108.
- [70] A. Grossfield, R. Pengyu, and J. W. Ponder. “Ion Solvation Thermodynamics from Simulation with a Polarizable Force Field”. *J. Am. Chem. Soc.* 125 (2003), 15671–15682.
- [71] T. W. Whitfield, S. Varma, E. Harder, G. Lamoureux, S. B. Rempe, and B. Roux. “Theoretical Study of Aqueous Solvation of K⁺ Comparing ab Initio, Polarizable, and Fixed-Charge Models”. *J. Chem. Theory Comput.* 3 (2007), 2068–2082.
- [72] M. L. Klein and W. Shinoda. “Large-Scale Molecular Dynamics Simulations of Self-Assembling Systems”. *Science* 321 (2008), 798–800.
- [73] T. Murtola, A. Bunker, I. Vattulainen, M. Deserno, and M. Karttunen. “Multiscale Modeling of Emergent Materials: Biological and Soft Matter”. *Phys. Chem. Chem. Phys.* 11 (12 2009), 1869–1892.
- [74] G. Voth, ed. *Coarse-graining of Condensed Phase and Biomolecular Systems*. CRC Taylor & Francis Group, 2009.
- [75] M. E. Johnson, T. Head-Gordon, and A. A. Louis. “Representability Problems for Coarse-grained Water Potentials”. *J. Chem. Phys.* 126 (2007), 144509.

- [76] T. Ha-Duong, N. Basdevant, and D. Borgis. “A polarizable Coarse-grained Water Model for Coarse-grained Proteins Simulations”. *Chem. Phys. Lett.* 468 (2009), 79–82.
- [77] S. O. Yesylevskyy, L. V. Schafer, D. Sengupta, and S. J. Marrink. “Polarizable Water Model for the Coarse-Grained MARTINI Force Field”. *PLoS Comput. Biol.* 6 (2010), e1000810.
- [78] Z. Wu, Q. Cui, and A. Yethiraj. “A New Coarse-Grained Model for Water: The Importance of Electrostatic Interactions”. *J. Phys. Chem. B* 114 (2010), 10524–10529.
- [79] L. Darré, M. R. Machado, P. D. Dans, F. E. Herrera, and S. Pantano. “Another Coarse Grain Model for Aqueous Solvation: WAT FOUR?” *J. Chem. Theory Comput.* 6 (2010), 3793–3807.
- [80] S. Riniker and W. F. van Gunsteren. “A simple, Efficient Polarizable Coarse-grained Water Model for Molecular Dynamics Simulations”. *J. Chem. Phys.* 134 (2011), 084110.
- [81] R. C. DeMille and V. Molinero. “Coarse-grained ions without charges: Reproducing the solvation structure of NaCl in water using short-ranged potentials”. *J. Chem. Phys.* 131 (2009), 034107.
- [82] S. Izvekov, J. M. J. Swanson, and G. A. Voth. “Coarse-Graining in Interaction Space: A Systematic Approach for Replacing Long-Range Electrostatics with Short-Range Potentials”. *J. Phys. Chem. B* 112 (2008). PMID: 18366209, 4711–4724.
- [83] M. Praprotnik, L. D. Site, and K. Kremer. “Adaptive Resolution Molecular-Dynamics Simulation: Changing the Degrees of Freedom on the Fly”. *J. Chem. Phys.* 123 (2005), 224106.
- [84] B. Ensing, S. O. Nielsen, P. B. Moore, M. L. Klein, and M. Parrinello. “Energy Conservation in Adaptive Hybrid Atomistic/Coarse-Grain Molecular Dynamics”. *J. Chem. Theory Comput.* 3 (2007), 1100–1105.
- [85] S. O. Nielsen, R. E. Buló, P. B. Moore, and B. Ensing. “Recent Progress in Adaptive Multiscale Molecular Dynamics Simulations of Soft Matter”. *Phys. Chem. Chem. Phys.* 12 (39 2010), 12401–12414.
- [86] A. Savelyev and G. Papoian. “Molecular Renormalization Group Coarse-Graining of Electrolyte Solutions: Application to Aqueous NaCl and KCl”. *J. Phys. Chem. B* (2009), 7785–7793.
- [87] R. D. Groot and K. L. Rabone. “Mesoscopic Simulation of Cell Membrane Damage, Morphology Change and Rupture by Nonionic Surfactants”. *Biophys. J.* (2001), 725–736.
- [88] M. Neumann. “Dipole Moment Fluctuation Formulas in Computer Simulations of Polar Systems”. *Mol. Phys.* 50 (1983), 841–858.
- [89] M. D. Tissandier, K. A. Cowen, W. Y. Feng, E. Gundlach, M. H. Cohen, A. D. Earhart, J. V. Coe, and T. R. Tuttle. “The Proton’s Absolute Aqueous Enthalpy and Gibbs Free Energy of Solvation from Cluster-Ion Solvation Data”. *J. Phys. Chem. A* 102 (1998), 7787–7794.
- [90] D. van der Spoel, E. Lindahl, B. Hess, A. R. van Buuren, E. Apol, P. J. Meulenhoff, D. P. Tieleman, A. L. T. M. Sijbers, K. A. Feenstra, R. van Drunen, and H. J. C. Berendsen. *Gromacs User Manual version 4.5.6*. www.gromacs.org. 2010.
- [91] I. S. Joung and T. E. I. Cheatham. “Determination of Alkali and Halide Monovalent Ion Parameters for Use in Explicitly Solvated Biomolecular Simulations”. *J. Phys. Chem. B* 112 (2008), 9020–9041.
- [92] J. Alejandre, D. J. Tildesley, and G. A. Chapela. “Molecular Dynamics Simulation of the Orthobaric Densities and Surface Tension of Water.” *J. Chem. Phys.* 102 (1995), 4574.

- [93] R. Weast. *Handbook of Chemistry and Physics 53rd Edition*. Chemical Rubber Pub., 1972.
- [94] M. Holz, S. Heil, and A. Sacco. “Temperature-dependent Self-diffusion Coefficients of Water and Six Selected Molecular Liquids for Calibration in Accurate H-1 NMR PFG Measurements”. *Phys. Chem. Chem. Phys.* 2 (2000), 4740–4742.
- [95] G. Kaye and T. Laby. *Tables of Physical and Chemical Constants*. Longman, 1995.
- [96] N. B. Vargaftik, B. N. Volkov, and L. D. Voljak. “International Tables of the Surface Tension of Water”. *J. Phys. Chem. Ref. Data* 12 (1983), 817–820.
- [97] W. Haynes. *CRC Handbook of Chemistry and Physics, 93rd Edition*. Taylor & Francis, 2012.
- [98] Y. Marcus. “Thermodynamics of solvation of ions. Part 5.-Gibbs free energy of hydration at 298.15 K”. *J. Chem. Soc., Faraday Trans.* 87 (18 1991), 2995–2999.
- [99] I. G. Tironi and W. F. Van Gunsteren. “A Molecular Dynamics Simulation Study of Chloroform”. *Mol. Phys.* 83 (1994), 381–403.
- [100] K. A. Motakabbir and M. Berkowitz. “Isothermal Compressibility of SPC/E Water”. *J. Phys. Chem.* 94 (1990), 8359–8362.
- [101] J. F. Pérez, C. Z. Hadad, and A. Restrepo. “Structural Studies of the Water Tetramer”. *Int. J. Quantum Chem* 108 (2008), 1653–1659.
- [102] M. Yang, P. Senet, and C. Van Alsenoy. “DFT Study of Polarizabilities and Dipole Moments of Water Clusters.” *Int. J. Quantum Chem* 101 (2005), 535–542.
- [103] A. J. Rusnak, E. R. Pinnick, C. E. Calderon, and F. Wang. “Static Dielectric Constants and Molecular Dipole Distributions of Liquid Water and Ice-Ih Investigated by the PAW-PBE Exchange-correlation Functional”. *J. Chem. Phys.* 137 (2012).
- [104] T. K. Ghanty and S. K. Ghosh. “Polarizability of Water Clusters: An ab initio Investigation”. *J. Chem. Phys.* 118 (2003), 8547–8550.
- [105] M. Reddy and M. Berkowitz. “The Dielectric Constant of SPC/E Water”. *Chem. Phys. Lett.* 155 (1989), 173–176.
- [106] G. S. Kell. “Density, Thermal Expansivity, and Compressibility of Liquid Water From 0.deg. to 150.deg.. Correlations and Tables for Atmospheric Pressure and Saturation Reviewed and Expressed on 1968 Temperature Scale”. *J. Chem. Eng. Data* 20 (1975), 97–105.
- [107] G. S. Kell. “Precise Representation of Volume Properties of Water at One Atmosphere”. *J. Chem. Eng. Data* 12 (1967), 66–69.
- [108] D. Lide, ed. *Handbook of Chemistry and Physics*. CRC, 1992.
- [109] S. Harvey, R. Tan, and T. Cheatham. “The Flying Ice Cube: Velocity Rescaling in Molecular Dynamics Leads to Violation of Energy Equipartition”. *J. Comput. Chem.* 19 (1998), 726–740.
- [204] J. C. Fogarty, M. Arjunwadkar, S. A. Pandit, and J. Pan. “Atomically Detailed Lipid Bilayer Models for the Interpretation of Small Angle Neutron and X-ray Scattering Data”. In press. 2014.
- [111] N. Franks and W. Lieb. “The structure of lipid bilayers and the effects of general anaesthetics: An X-ray and neutron diffraction study”. *J. Mol. Biol.* 133 (1979), 469–500.
- [112] M. Wiener and S. White. “Fluid Bilayer Structure Determination by the Combined Use of X-Ray and Neutron Diffraction. I. Fluid Bilayer Models and the Limits of Resolution”. *Biophys. J.* 59 (1991), 162–173.

- [113] M. Wiener and S. White. “Structure of a fluid dioleoylphosphatidylcholine bilayer determined by joint refinement of x-ray and neutron diffraction data. III. Complete structure”. *Biophys. J.* 61 (1992), 434–447.
- [114] N. Kuerka, J. F. Nagle, J. N. Sachs, S. E. Feller, J. Pencer, A. Jackson, and J. Katsaras. “Lipid Bilayer Structure Determined by the Simultaneous Analysis of Neutron and X-Ray Scattering Data”. *Biophys. J.* 95 (2008), 2356–2367.
- [115] N. Kuerka, M.-P. Nieh, and J. Katsaras. “Fluid phase lipid areas and bilayer thicknesses of commonly used phosphatidylcholines as a function of temperature”. *Biochim. Biophys. Acta Biomembr.* 1808 (2011), 2761–2771.
- [116] N. Kuerka, B. W. Holland, C. G. Gray, B. Tomberli, and J. Katsaras. “Scattering Density Profile Model of POPG Bilayers As Determined by Molecular Dynamics Simulations and Small-Angle Neutron and X-ray Scattering Experiments”. *J. Phys. Chem. B* 116 (2012), 232–239.
- [117] J. Pan, F. A. Heberle, S. Tristram-Nagle, M. Szymanski, M. Koepfinger, J. Katsaras, and N. Kuerka. “Molecular Structures of Fluid Phase Phosphatidylglycerol Bilayers as Determined by Small Angle Neutron and X-ray Scattering”. *Biochim. Biophys. Acta Biomembr.* 1818 (2012), 2135–2148.
- [118] J. Pan, X. Cheng, L. Monticelli, F. A. Heberle, N. Kucerka, D. P. Tieleman, and J. Katsaras. “The molecular structure of a phosphatidylserine bilayer determined by scattering and molecular dynamics simulations”. *Soft Matter* 10 (21 2014), 3716–3725.
- [119] P. Heftberger, B. Kollmitzer, F. A. Heberle, J. Pan, M. Rappolt, H. Amenitsch, N. Kuerka, J. Katsaras, and G. Pabst. “Global small-angle X-ray scattering data analysis for multilamellar vesicles: the evolution of the scattering density profile model”. *J. Appl. Crystallogr.* 47 (2014), 173–180.
- [120] N. D. Pham and B. M. Wilamowski. “Improved Nelder Mead’s Simplex Method and Applications”. *J. Comput.* 3 (2011).
- [121] R. C. Aster, B. Borchers, and C. H. Thurber. *Parameter Estimation and Inverse Problems*. San Diego : Elsevier, 2004., 2004.
- [122] S. A. Pandit, S.-W. Chiu, E. Jakobsson, A. Grama, and H. L. Scott. “Cholesterol Packing around Lipids with Saturated and Unsaturated Chains: A Simulation Study”. *Langmuir* 24 (2008), 6858–6865.
- [123] J. B. Klauda, N. Kucerka, B. R. Brooks, R. W. Pastor, and J. F. Nagle. “Simulation-based Methods for Interpreting X-ray Data from Lipid Bilayers.” *Biophys. J.* 90 (2006), 2796–2807.
- [124] J. B. Klauda, R. M. Venable, J. A. Freites, J. W. O’Connor, D. J. Tobias, C. Mondragon-Ramirez, I. Vorobyov, A. D. MacKerell, and R. W. Pastor. “Update of the CHARMM All-Atom Additive Force Field for Lipids: Validation on Six Lipid Types”. *J. Phys. Chem. B* 114 (2010), 7830–7843.
- [125] O. Berger, O. Edholm, and F. Jähnig. “Molecular dynamics simulations of a fluid bilayer of dipalmitoylphosphatidylcholine at full hydration, constant pressure, and constant temperature”. *Biophys. J.* 72 (1997), 2002–2013.
- [126] K. Tu, D. Tobias, and M. Klein. “Constant pressure and temperature molecular dynamics simulation of a fully hydrated liquid crystal phase dipalmitoylphosphatidylcholine bilayer”. *Biophys. J.* 69 (1995), 2558–2562.

- [127] B. Hess, C. Kutzner, D. van der Spoel, and E. Lindahl. “GROMACS 4: Algorithms for Highly Efficient, Load-balanced, and Scalable Molecular Simulation”. *J. Chem. Theory Comput.* 4 (2008), 435–447.
- [128] S.-W. Chiu, S. A. Pandit, H. L. Scott, and E. Jakobsson. “An Improved United Atom Force Field for Simulation of Mixed Lipid Bilayers”. *J. Phys. Chem. B* 113 (2009). PMID: 19708111, 2748–2763.
- [129] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin. *Bayesian Data Analysis*. Third. Chapman and Hall/CRC, 2013.
- [130] K. Burnham and D. Anderson. *Model Selection and Multimodel Inference*. New York: Springer, 2002.
- [131] B. R. Pauw. “Everything SAXS: small-angle scattering pattern collection and correction”. *J. Phys.: Condens. Matter* 25 (2013), 383201.
- [132] B. Pauw. “Corrigendum: Everything SAXS: small-angle scattering pattern collection and correction (2013 J. Phys.: Condens. Matter 25 383201)”. *J. Phys.: Condens. Matter* 26 (2014), 239501.
- [133] J. Das, E. Flenner, and I. Kosztin. “Anomalous diffusion of water molecules in hydrated lipid bilayers”. *J. Chem. Phys.* 139 (2013), 065102.
- [134] J. A. Mondal, S. Nihonyanagi, S. Yamaguchi, and T. Tahara. “Three Distinct Water Structures at a Zwitterionic Lipid/Water Interface Revealed by Heterodyne-Detected Vibrational Sum Frequency Generation”. *J. Am. Chem. Soc.* 134 (2012), 7842–7850.
- [135] J. Nagle and S. Tristram-Nagle. “Structure of Lipid Bilayers”. *Biochim. Biophys. Acta* 1469 (2000), 159–195.
- [136] V. Parsegian and R. Rand. “Interaction in membrane assemblies”. In: *Structure and Dynamics of Membranes*. Ed. by R. Lipowsky and E. Sackmann. Vol. 1. North-Holland, 1995, 643–690.
- [137] D. Voet and J. Voet. *Biochemistry*. Second. Wiley, 1995.
- [138] R. Kohen and A. Nyska. “OXidation of Biological Systems: Oxidative Stress Phenomena, Antioxidants, Redox Reactions, and Methods for their Quantification”. *Toxicol Pathol.* 30 (2002), 620–50.
- [139] H. Huang and K. G. Manton. “The Role of Oxidative Damage in Mitochondria During Aging: A Review”. *Front Biosci.* 9 (2004), 1100–1117.
- [140] S. Cukierman. “Et tu, Grotthuss! and other unfinished stories”. *Biochim. Biophys. Acta Bioenergetics* 1757 (2006). Proton Transfer Reactions in Biological Systems, 876–885.
- [141] A. C. T. van Duin, C. Zou, K. Joshi, V. Bryantsev, and W. A. Goddard. “CHAPTER 6 A Reaxff Reactive Force-field for Proton Transfer Reactions in Bulk Water and its Applications to Heterogeneous Catalysis”. In: *Computational Catalysis*. The Royal Society of Chemistry, 2014, 223–243.
- [142] A. C. T. van Duin, S. Dasgupta, F. Lorant, and W. A. Goddard. “ReaxFF: A Reactive Force Field for Hydrocarbons”. *J. Phys. Chem. A* 105 (2001), 9396–9409.
- [143] A. C. T. van Duin, A. Strachan, S. Stewman, Q. Zhang, X. Xu, and W. Goddard. “ReaxFF(SiO) reactive force field for silicon and silicon oxide systems”. *J. Phys. Chem. A* 107 (2003), 3803–3811.

- [144] J. E. Mueller, A. C. T. van Duin, and W. A. Goddard. "Development and Validation of ReaxFF Reactive Force Field for Hydrocarbon Chemistry Catalyzed by Nickel". *J. Phys. Chem. C* 114 (2010), 4939–4949.
- [145] H. M. Aktulga, S. Pandit, A. C. van Duin, and A. Grama. "Reactive molecular dynamics: Numerical methods and algorithmic techniques". *SIAM J. Sci. Comput.* 34 (2010), C1–C23.
- [208] H. M. Aktulga, J. C. Fogarty, S. A. Pandit, and A. Y. Grama. "Parallel reactive molecular dynamics: Numerical methods and algorithmic techniques". *Parallel Comput.* 38 (2012), 245–259.
- [147] A. K. Rappé and W. A. I. Goddard. "Charge Equilibration for Molecular Dynamics Simulations". *J. Phys. Chem.* 95 (1991), 3358–3363.
- [148] M. J. Frisch, G. W. Trucks, H. B. Schlegel, G. E. Scuseria, M. A. Robb, J. R. Cheeseman, G. Scalmani, V. Barone, B. Mennucci, G. A. Petersson, H. Nakatsuji, M. Caricato, X. Li, H. P. Hratchian, A. F. Izmaylov, J. Bloino, G. Zheng, J. L. Sonnenberg, M. Hada, M. Ehara, K. Toyota, R. Fukuda, J. Hasegawa, M. Ishida, T. Nakajima, Y. Honda, O. Kitao, H. Nakai, T. Vreven, J. A. Montgomery Jr., J. E. Peralta, F. Ogliaro, M. Bearpark, J. J. Heyd, E. Brothers, K. N. Kudin, V. N. Staroverov, R. Kobayashi, J. Normand, K. Raghavachari, A. Rendell, J. C. Burant, S. S. Iyengar, J. Tomasi, M. Cossi, N. Rega, J. M. Millam, M. Klene, J. E. Knox, J. B. Cross, V. Bakken, C. Adamo, J. Jaramillo, R. Gomperts, R. E. Stratmann, O. Yazyev, A. J. Austin, R. Cammi, C. Pomelli, J. W. Ochterski, R. L. Martin, K. Morokuma, V. G. Zakrzewski, G. A. Voth, P. Salvador, J. J. Dannenberg, S. Dapprich, A. D. Daniels, Ö. Farkas, J. B. Foresman, J. V. Ortiz, J. Cioslowski, and D. J. Fox. *Gaussian09 Revision D.01*. Gaussian Inc. Wallingford CT 2009. 2009.
- [149] L.-P. Wang, T. Head-Gordon, J. W. Ponder, P. Ren, J. D. Chodera, P. K. Eastman, T. J. Martinez, and V. S. Pande. "Systematic Improvement of a Classical Molecular Model of Water". *J. Phys. Chem. B* 117 (2013), 9956–9972.
- [150] P. Atkins and R. Friedman. *Molecular Quantum Mechanics*. Oxford University Press, 2005.
- [151] A. Soper. "The Radial Distribution Functions of Water as Derived from Radiation Total Scattering Experiments: Is There Anything We Can Say for Sure?" *ISRN Phys. Chem.* 2013 (2013), 279463.
- [152] F. Gao and L. Han. "Implementing the Nelder-Mead simplex algorithm with adaptive parameters". *Comp. Optim. Appl.* 51 (2012), 259–277.
- [210] J. C. Fogarty, H. M. Aktulga, A. Y. Grama, A. C. van Duin, and S. A. Pandit. "A reactive molecular dynamics simulation of the silica-water interface". *J. Chem. Phys.* 132 (2010), 174704.
- [154] M. Qhobosheane, S. Santra, P. Zhang, and W. Tan. "Biochemically functionalized silica nanoparticles". *Analyst* 126 (2001), 1274–1278.
- [155] W. Tan, K. Wang, X. He, X. Zhao, T. Drake, L. Wang, and R. Bagwe. "Bionanotechnology based on silica nanoparticles". *Med. Res. Rev.* 24 (2004), 621–638.
- [156] R. S. McDonald. "Surface Functionality of Amorphous Silica by Infrared Spectroscopy". *J. Phys. Chem.* 62 (1958), 1168–1178.
- [157] K. Davis and M. Tomozawa. "An infrared spectroscopic study of water-related species in silica glasses". *J. Non-Cryst. Solids* 201 (1996), 117–198.
- [158] R. Sato and M. Tomozawa. "Interaction of water diffusion and hydrogen diffusion in silica glass at high temperatures". *J. Non-Cryst. Solids* 343 (2004), 26–32.

- [159] R. Mozzi and B. Warren. “The Structure of Vitreous Silica”. 2 (1969), 164.
- [160] Q. Mei, C. Benmore, S. Sen, R. Sharma, and J. Yarger. “Intermediate range order in vitreous silica from a partial structure factor analysis”. *Phys. Rev. B* 78 (2008), 144204.
- [161] T. Charpentier, P. Kroll, and F. Mauri. “First-Principles Nuclear Magnetic Resonance Structural Analysis of Vitreous Silica”. *J. Phys. Chem. C* 113 (2009), 7917–7929.
- [162] A. Pedone, G. Malavasi, M. C. Menziani, U. Segre, F. Musso, M. Corno, B. Civalleri, and P. Ugliengo. “FFSiOH: a New Force Field for Silica Polymorphs and Their Hydroxylated Surfaces Based on Periodic B3LYP Calculations”. *Chem. Mater.* 20 (2008), 2522–2531.
- [163] C. Mischler, W. Kob, and K. Binder. “Classical and ab-initio molecular dynamic simulation of an amorphous silica surface”. 147 (2002), 222–225.
- [164] E. R. Cruz-Chu, A. Aksimentiev, and K. Schulten. “Water-silica force field for simulating nanodevices”. *J. Phys. Chem. B* 110 (2006), 21497–21508.
- [165] J. Du and A. N. Cormack. “Molecular Dynamics Simulation of the Structure and Hydroxylation of Silica Glass Surfaces”. *J. Am. Chem. Soc.* 88 (2005), 2532–2539.
- [166] T. Mahadevan and S. Garofalini. “Dissociative Chemisorption of Water onto Silica Surfaces and Formation of Hydronium Ions”. *J. Phys. Chem. C* 112 (2008), 1507–1515.
- [167] M.-H. Du, A. Kolchin, and H.-P. Cheng. “Hydrolysis of a two-membered silica ring on the amorphous silica surface”. *J. Chem. Phys.* 120 (2004), 1044.
- [168] A. A. Hassanali and S. J. Singer. “Model for the Water-Amorphous Silica Interface: The Undissociated Surface”. *J. Phys. Chem. B* 111 (2007), 11181–11193.
- [169] M. Du, A. Kolchin, and H. Cheng. “Water-silica surface interactions: A combined quantum-classical molecular dynamic study of energetics and reaction pathways”. *J. Chem. Phys.* 119 (2003), 6418–6422.
- [170] J. Yu, S. B. Sinnott, and S. R. Phillpot. “Charge optimized many-body potential for the Si/SiO₂ system”. *Phys. Rev. B* 75 (2007), 085311.
- [171] Y. Saad and M. Schultz. “GMRES - A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear-systems”. *SIAM J. Sci. Comput.* 7 (1986), 856–869.
- [172] K. Chenoweth, A. C. T. van Duin, and W. A. Goddard. “ReaxFF Reactive Force Field for Molecular Dynamics Simulations of Hydrocarbon Oxidation”. *J. Phys. Chem. A* 112 (2008), 1040–1053.
- [173] S. Nosé. “A unified formulation of the constant temperature molecular dynamics methods”. *J. Chem. Phys.* 81 (1984), 511–519.
- [174] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola, and J. R. Haak. “Molecular dynamics with coupling to an external bath”. *J. Chem. Phys.* 81 (1984), 3684–3690.
- [175] G. J. Martyna, D. J. Tobias, and M. Klein. “Constant pressure molecular dynamics algorithms”. *J. Chem. Phys.* 101 (1994), 4177.
- [176] K. Vollmayr, W. Kob, and K. Binder. “Cooling-rate effects in amorphous silica: A computer-simulation study”. *Phys. Rev. B* 54 (1996), 15808.
- [177] N. T. Huff, E. Demiralp, C. Tahir, and W. A. I. Goddard. “Factors affecting molecular dynamics simulated vitreous silica structures”. *J. Non-Cryst. Solids* 253 (1999), 133–142.

- [178] M. Tarini, P. Cignoni, and C. Montani. “Ambient Occlusion and Edge Cueing for Enhancing Real Time Molecular Visualization”. 12 (2006), 1237–1244.
- [179] T. T. Trinh, A. P. J. Jansen, and R. A. van Santen. “Mechanism of Oligomerization Reactions of Silica”. *J. Phys. Chem. B* 110 (2006), 23099–23106.
- [180] T. A. Halgren and W. Damm. “Polarizable force fields”. 11 (2001), 236–242.
- [181] M. Sprik, J. Hutter, and M. Parrinello. “Ab initio molecular dynamics simulation of liquid water: Comparison of three gradient-corrected density functionals”. *J. Chem. Phys.* 105 (1996), 1142.
- [182] W. S. Benedict, N. Gailar, and E. K. Plyler. “Rotation-Vibration Spectra of Deuterated Water Vapor”. *J. Chem. Phys.* 24 (1956), 1139–1165.
- [183] S. A. Clough, Y. Beers, G. P. Klein, and L. S. Rothman. “Dipole moment of water from Stark measurements of H₂O, HDO, and D₂O”. *J. Chem. Phys.* 59 (1973), 2254–2259.
- [184] Y. S. Badyal, M.-L. Saboungi, D. L. Price, S. D. Shastri, D. R. Haefner, and A. K. Soper. “Electron distribution in water”. *J. Chem. Phys.* 112 (2000), 9206–9208.
- [185] R. Bentwood, A. Barnes, and W. Orville-Thomas. “Studies of intermolecular interactions by matrix isolation vibrational spectroscopy Self-association of water”. *J. Mol. Spectrosc.* 84 (1980), 391–404.
- [186] K. Kuchitsu and Y. Morino. “Estimation of Anharmonic Potential Constants. I. Linear XY₂ Molecules”. 38 (1965), 805–813.
- [187] A. Soper. “Orientational correlation function for molecular liquids: The case of liquid water”. *J. Chem. Phys.* 101 (1994), 6888–6901.
- [188] K. Krynicki, C. D. Green, and D. W. Sawyer. “Pressure and temperature dependence of self-diffusion in water”. 66 (1978), 199–208.
- [189] D. A. Kemp and M. S. Gordon. “An interpretation of the enhancement of the water dipole moment due to the presence of other water molecules”. *J. Phys. Chem. A* 112 (2008), 4885–4894.
- [190] P. Chaikin and T. Lubensky. *Principles of condensed matter physics*. The Edinburgh Building, Cambridge CB2 2RU, UK: Cambridge University Press, 1995.
- [191] A. Takada, P. Richet, C. Catlow, and G. Price. “Molecular dynamics simulations of vitreous silica structures”. *J. Non-Cryst. Solids* 345 (2004), 224–229.
- [192] P. Vashishta, R. K. Kalia, and J. P. Rino. “Interaction potential for SiO₂: A molecular-dynamics study of structural correlations”. *Phys. Rev. B* 41 (1989), 12197.
- [193] D. I. Grimley and A. C. Wright. “Neutron Scattering from Vitreous Silica”. *J. Non-Cryst. Solids* 119 (1990), 49.
- [194] J. Da Silva, D. Pinatti, C. Anderson, and M. Rudee. “A refinement of the structure of vitreous silica”. *Philos. Mag.* 31 (1975), 713.
- [195] V. Sokhan and D. Tildesley. “The free surface of water: molecular orientation, surface potential and nonlinear susceptibility”. *Mol. Phys.* 92 (1997), 625–640.
- [196] T. Bakos, S. Rashkeev, and S. Pantelides. “Reactions and Diffusion of Water and Oxygen Molecules in Amorphous SiO₂”. *Phys. Rev. Lett.* 88 (2002), 055508.
- [197] E. Papirer. *Adsorption On Silica Surfaces*. Vol. 90. New York Marcel Dekker, Inc., 2000.

- [198] L. Zhuravlev. “The surface chemistry of amorphous silica. Zhuravlev model”. 173 (2000), 1–38.
- [199] S. A. Pandit. *Soft Matter Volume 4 Lipid Bilayers and Red Blood Cells*. Vol. 4. Weinheim: Wiley-VCH Verlag GmbH & Co. KGaA, 2008.
- [200] S. Ong, X. Zhao, and K. Eissenthal. “Polarization of water molecules at a charged interface: second harmonic studies of the silica/water interface”. 191 (1992), 327–335.
- [209] A. Y. Grama, J. C. Fogarty, H. M. Aktulga, and S. A. Pandit. “N-Body Computational Methods”. In: *Encyclopedia of Parallel Computing*. Ed. by D. A. Padua. Springer, 2011, 1259–1268.
- [205] J. C. Fogarty, A. Y. Grama, A. C. van Duin, and S. A. Pandit. “Optimization of a Reactive Molecular Dynamics Model for Water”. In preparation. 2014.
- [206] J. C. Fogarty, S.-W. Chiu, E. Jakobsson, and S. A. Pandit. “Extension of a Coarse-Grained Model to Water-Ion Interactions”. In preparation. 2014.

Appendix A

ParOpt Manual

A.1 Introduction

ParOpt is an automated parameter optimization software framework. The user specifies a parameter space, a target function defined on the space, and options governing the optimization procedure. ParOpt produces a local minima in the parameter space, along with various output during the procedure. The method of optimization is a choice of the user. The parameter space is defined by specifying parameters to be varied and their allowable ranges (sec A.3.3). The target function is separated into a logical procedure which consists of three phases: execution, (sec A.3.1), analysis (sec. A.3.4), and comparison(sec. A.3.4). Since it is parallelized, the execution, or simulation phase, should perform the most computationally costly part of the target function evaluation. For example, the execution phase can consist of several molecular dynamics simulations, or long quantum chemical calculations. The analysis phase of the target function should involve extraction from the simulation results of data which is easily comparable to the known target data. The comparison phase then is application some metric defined between the target data and the observed data. The specifics of each of these phases is left the user to define.

A.2 Usage

```
./paropt control_file(s)
```

A.3 Input

ParOpt requires at least one argument: an input file referred to as the control file. Multiple files can be included, in order to run iterative optimizations. In the case of multiple control files, ParOpt will find the optimum value of the first parameter space, and then apply these parameters to the subsequent set (the user must ensure that parameter addressing is consistent across files). This procedure is then repeated for all control files and iterated over. A control file has 4 sections: Global Options, Force Field Constants, Force Field Parameters, and Data. The global section must precede all other sections. Sections are determined by a line containing one of the following:

```
global:
constants:
parameters:
data:
```

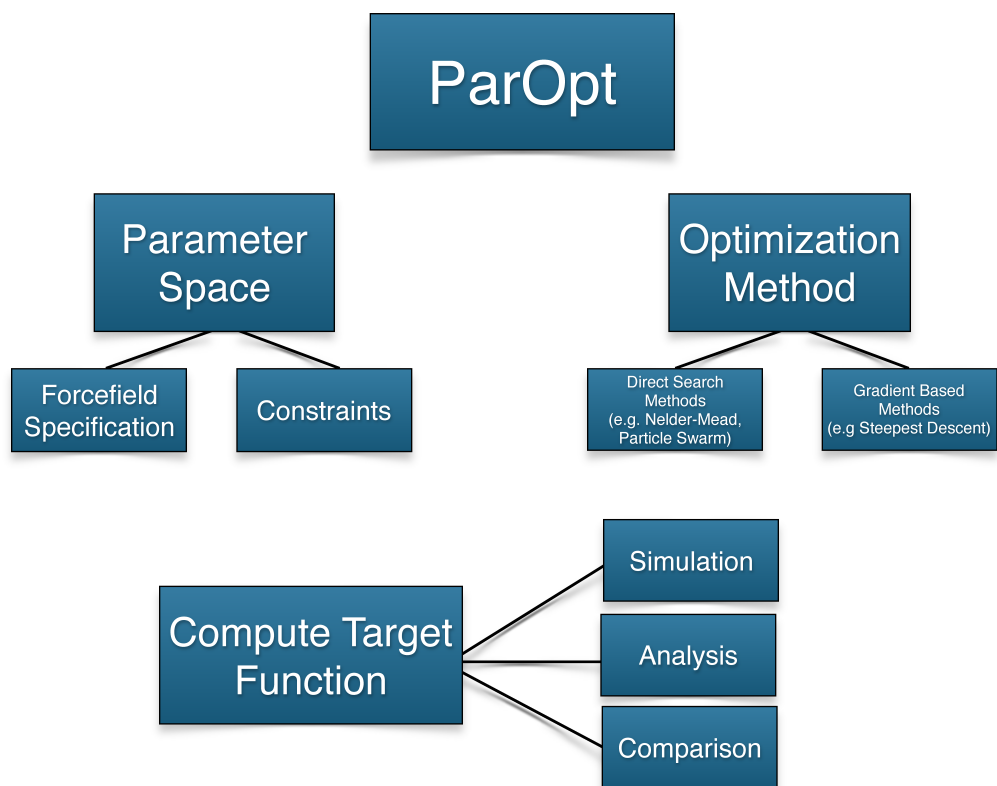


Figure A.1: Illustration of the design and structure of the ParOpt optimization software.

constraints:

CPP is executed on the input file, allowing **#include**, **#define**, **ifdef**, etc. Comments can occur anywhere on a line and are represented by an exclamation point (!). Comparisons of string literals in the control file are not case sensitive. Colons are required wherever present. Files are read in with blank space delimiters, no fixed widths are assumed.

The global section must appear first. This section contains the executable commands, names of forcefield files, specification of force field file writing methods, specification of optimization methods, and general options. The other sections can be in any order, and multiple sections of one type are allowed. Most global options may be redefined. The primary method by which ParOpt interfaces with external processes is by writing force field parameters. Any values (real numbers, strings etc.) to be written to the force field file, whether varied for optimization or not, must be listed in the input file for ParOpt. Parameters to be varied are placed in the params section, those held constant are placed in the constants section. Constraints on minimum and maximum allowable ranges for parameters are specified in the parameters section. Any other constraints are placed in the constraints section. Executable commands are written in the global section, while analysis and comparison are in the data section. Note: all values listed in this manual are the default values, unless marked as an example.

A.3.1 Global

Global parameters are read from the beginning of the control file. Global parameters include:

Simulation Type

simulation_type: **Generic**

This specifies the type of simulation to be run. As of 4.2 available options are:

Generic Generic force field writing

Reax Reactive Molecular Dynamics

GROMACS GRONingen MAchine for Chemical Simulation

With careful setup, any plain text file may be written using the Generic forcefield file writing. The only difference between simulation types is in the force field writing. See section A.4 for detail on the differences between the methods.

Manager Type

manager_type: **nelder-mead**

The type of optimization manager to run. Options are:

nelder-mead Iterative Nelder-Mead optimization

sequential One parameter at a time sequential parabolic fitting

multi-sequential Loop of the sequential manager

priority

sensitivity

steepest Simple steepest descent

scattering Perform a one-dimensional optimization of atomic density profiles in order to match scattering (e.g. X-ray and/or neutron) data. For this manager, all calculations are performed internal to ParOpt. See section A.5 for details.

swarm Use the particle swarm optimization method.

Force Field File(s)

```
(example)
ffield:                1 ffield.water
```

The force field files required to run simulations, preceded by the number of files. These files may or may not be used to run simulations, depending on input to the exec lines. These are the temporary files that ParOpt will write during optimization, once for each target function evaluation. One set of forcefield files corresponds to one point in parameter space. For the Reax and GROMACS managers, only one force field file is allowed. For generic, any number of files may be used. Once optimization is complete (for manager_type **nelder-mead** every time the Nelder Mead method converges) a final set of files is written, with the suffix **_final_N** where N is the index of the final forcefield file set, i.e. how times the optimization has converged.

Executable(s)

```
(example):
exec:  2
      9 grompp_mpi -f control.mdp -c geo.gro\\
        -o test.tpr -p ff_out.top
      6 mpirun -np NP mdrun_mpi -s test.tpr
exec:  1
      1 run_simulation.sh
```

The first line specifies the number of execution steps in running the simulation. Each simulation to be run requires its own executable line. Each subsequent line has the number of arguments followed by the arguments, with the first argument is the executable itself. This is followed by its arguments. In this example, the force field file name **ff_out.top** should be specified in the **ffield:** line.

Number of Processors

```
num_proc:  1
```

Specifies the number of simultaneous processes that will be executed. Lines within the same exec block will be executed sequentially. Jobs are started in batches, so to achieve the best efficiency, jobs of similar lengths should be run together. For example, if **num_proc** is set to 4, the optimizer will start the first four jobs. Once all four jobs have terminated, the next set of four jobs will be started.

Children's Working Directory

```
child_wd:  ./
```

All child process started by ParOpt will have this directory as their working directory. Make sure to write the force field files correctly, keeping this in mind. No ParOpt file output (including force field file writing) explicitly uses this directory. This option is useful when running a multi-phase optimization. Then control files can be maintained in one directory, but specific running for each phase will take place in different directories.

Seed

seed: -1

Seed for random selection of initial parameters. The random function is seeded before generation of the random parameters for each control set. A value of `-1` sets the seed based on the time.

Comment

comment_character:

Reads in the character to be used to write a one line header comment in the force field file writing. For GROMACS, it is automatically set to `;` unless overridden. Default is no comment. If no characters are present after **comment_character**, the comment line is not written.

Verbosity

verbosity: warn

Takes as argument a list of flags which specify how wordy ParOpt should be. Options are: silent, input, warn, calc, info, exec, and all. Each controls whether to write to stderr in certain circumstances:

silent turns all flags off.

input prints information about reading input files.

warn prints all warnings (enabled by default).

calc prints scores for data lines as they are calculated.

exec prints all commands executed by the managers.

info prints miscellaneous information.

scattering prints information for the scattering manager.

all turns all flags on.

Quit

quit: never

Determines at which stage(s) non-zero exit codes from child processes will cause the target function evaluation to exit. Options are: **never**, **analysis**, **simulation**, and **always**. **Never** will never exit the target function evaluation, while **always** will quit whenever getting a non-zero exit code.

Particle Swarm Options

```
num_swarms:          50          !size of the swarm
swarm_trajectory:    "swarm.xyz"  !swarm trajectory output
swarm_rmsd_cut:      1.0
swarm_velocity_scale: 1.0
swarm_omega:         1.0
swarm_phi_g:         0.01
swarm_phi_p:         0.01
```

Nelder-Mead Options

```
max_nm_steps:        1000        !max steps to converge
max_nm_stuck_steps:   10         !max steps w/ identical error
nm_iterations:        5000       !max times to converge
nm_recompute_lowest:  0          !recompute every this many steps
nm_initial_point:     0          !add initial point to simplex
recalculate_errors:   15         !retry initial simplex errors
nm_constraints:       boundary   !constraint method
nm_a:                 1.0        !alpha
nm_b:                 0.5        !beta
nm_g:                 2.0        !gamma
nm_d:                 0.5        !delta
nm_verbosity:        all        !verbosity
nm_adaptive:          0          !adaptive parameters?
nm_rmsd_cut:          0.001      !nm convergence cutoff
nm_restart:           0          !restart from this file
```

Options for the Nelder-Mead Manager. Nelder-Mead simplex transformation parameters α , β , γ , and δ are specified by `nm_a`, `nm_b`, `nm_g`, and `nm_d`. The option `nm_adaptive` scales the Nelder-Mead simplex parameters based on the dimension of the space.¹ This can speed up optimization, especially in the case of high dimension ($\gg 10$). A Nelder-Mead iteration is a run of Nelder-Mead from a starting set of points, resulting in convergence (rmsd below `nm_rmsd_cut`) or quitting (reaching `max_nm_steps`, or `max_nm_stuck_steps`). A stuck step is defined as a step in which the mean changed by 0 (i.e. no points changed). The option `nm_iterations` specifies the number of iterations of Nelder-Mead to run. The method of constraints for Nelder-Mead is specified via the parameter `nm_constraints`. A value of `boundary` places a point violating a constraint on the boundary (in the case of minimum and maximum value constraints) or at the location specified in the constraint line (see sec. A.3.5). A value of `inf_err` assigns infinite error to any point violating any constraint. The option `nm_verbosity` takes a list of flags (similar to the `verbosity:` option). Possible flags are: `silent`, `error`, `simplex`, `constraint`, and `all`.

Initial Points Generation `random_about_point:` 1 !Flag for initial simplex method
(example)
`full_random:` 0.75 !Scale range by (1-value)

¹F. Gao and L. Han. "Implementing the Nelder-Mead simplex algorithm with adaptive parameters". English. *Comp. Optim. Appl.* 51.1 (2012), 259–277

If `random_about_point` is specified, the initial parameter values for Nelder-Mead and particle swarm are chosen with values a step up, a step down, or no step from the initial value. If `full_random` is specified, points are chosen on the min-max range of that parameter, with the range scaled by the number specified (p):

$$\{min, max\} \rightarrow \{min + p * (max - min)/2, max - p * (max - min)/2\} \quad (A.1)$$

A value of $p = 0$ gives the full range. Whichever option of `full_random` or `random_about_point` is specified last is used.

Initialization script

```
(default)
initialize:          0
(example)
initialize:          1 init.sh
```

Command and any arguments to be run (in the order they appear in this line) before any optimization starts. Scripts don't work with all manager types. Scripts work with Nelder-Mead.

Preparation script

```
(default)
cleanup:             0
(example)
prepare:             1 prep.sh
```

Command with arguments to be run (in the order they appear in this line) before simulations are run, but after the force field file is written. This meaning depends on the manager. This script is useful to make modifications to other simulation related files which depend on the force field parameters, but don't explicitly appear in the parameter set. An example is define a charge which is opposite to a charge included in the parameter space. Scripts don't work with all manager types. Scripts work with Nelder-Mead.

Cleanup Script

```
(default)
cleanup:             0
(example)
cleanup:             2 clean.sh up
```

Command with arguments to be run (in the order they appear in this line) after simulations are run, but before any analysis or comparison is performed. The is useful if backups of simulations are desired, and ensuring that files are in the desired location for analysis. Scripts don't work with all manager types. Scripts work with Nelder-Mead.

Finalization Script


```

(default)
finalize:                0
(example)
finalize:                1 final.sh

```

Command with arguments to be run (in the order they appear in this line) after optimization is complete. This meaning depends on the manager. Scripts don't work with all manager types. Scripts work with Nelder-Mead

System Error

```

print_system_error:      1

```

Final Error

```

print_final_error:       1

```

Control File

```

print_control_file:      1

```

Minimum Maximum parameter constraints

```

add_param_range:         0      !default
add_param_range:         top    !example

```

This option selects where (top of the list, bottom of list) to add minimum and maximum parameter values to the list of constraints. If no other constraints are being applied, the location is not important. If other constraints are being used, then the order of constraints may be meaningful. A value of 0 does not add parameter minimum and maximum values to the constraints, and these values do not effect the parameter space during optimization. See section A.3.5 for more information on the constraints.

A.3.2 Constant Parameters

Constants are force field parameters that will not be varied during optimization, but will be written into each new force field. A line consists of a name, a colon, an address, a format specification for printf (also determines the type of the data) and the value. Refer to the addressing section (sec. A.4) for the addressing scheme for each simulation method.

```

constant_name:  address format_specification value
(example)
cons1:  7:10:1 "%-15.8lf" 115.0
cons2:  1:12:1 "%14d"      12

```

A.3.3 Variable Parameters

Parameters that will be varied in order to produce a more accurate force are contained in this section. As parameters and constants will both be used in the same force field file, there should be no overlap. The number of parameters here will determine the size of parameter space to be explored. The line contains the name, address, format, initial value, minimum value and maximum value, and step size. Initial value and step size are used to generate initial sets of parameter when the option `random_about_point` is used. The values for minimum and maximum are use when the option `full_random` is set. Minimum and maximum values may be omitted. In this case, the range is set to $-\infty, \infty$.

```
parameter_name:  address format_specification init_val\\
                  min_val max_val step_val
```

(example)

```
param1:  2:11:1 "%12.4e" 70.0 90.5 0.05
```

```
param2:  2:11:2 "%14.5f" 1.1124
```

A.3.4 Data

The error function is computed as a sum of all the error function components:

$$\mathcal{F} = \sum_i w_i f_i(p) \quad (\text{A.2})$$

where \mathcal{F} is the target function value for optimization, the index i specifies a data line (see below) w_i is the weight associated with that line, p is a point in parameter space (a specific combination of parameter values), and $f_i(p)$ is the value returned by the comparator associated with the specific data line. Each data point, for each i , to be added to a total error is specified of a single line with the fields:

```
data_name: weight tolerance analysis_executable num_args args...\\
            comparator_executable num_args args....
```

(example)

```
energy: 1.0  1.0 Energy_Pot 3 paropt_data merged-0.pot reax_data\\
            comparator 2 reax_data paropt_data
```

data_name: A unique name for each data line.

weight Multiplicative factor for determining total weight.

tolerance Not yet used.

analysis The analysis executable, the number of arguments, and arguments list

comparison The comparison executable, its number of arguments, and the arguments list

Executable names for either analysis or comparison can be set to NULL to skip execution of that phase of target function evaluation.

A.3.5 Constraints

Complex boundaries of the allowable parameter space are defined using this section. Constraints are applied to the Nelder-Mead optimization method and initial simplex (for Nelder-Mead) and set of particles (for particle swarm) generation. The particle swarm optimization is not constrained. Each line in the constraints section specifies one constraint. Individual constraints cannot be split onto multiple lines. The constraint lines follow prefix notation, with `()` grouping operators and their arguments.

`(operator arguments...)`

Allowable operators are

if The conditional takes three arguments: a boolean (e.g. the value returned by `>` or `=`), a real value to be returned if the boolean is true, and a real value to be returned if the boolean is false.

assign The assignment operator takes a parameter name (the name specified in the **params:** section of the control file) and a real value to assign. It returns the value assigned. Constants cannot be modified via constraints.

abs A unary operator returning the absolute value of the real value argument.

max A binary operator returning the maximum of two real argument values.

min A binary operator returning the minimum of two real argument values.

< A binary operator returning boolean true if the first argument is less than the second argument and false otherwise.

> A binary operator returning boolean true if the first argument is less than the second argument and false otherwise.

= A binary operator returning boolean true if the first argument is equal to than the second argument and false otherwise.

& A binary logical operator taking two boolean arguments and returning the logical “and”.

| A binary logical operator taking two boolean arguments and returning the logical “or”.

! A unary logical operator returning the logical “not” of the boolean argument.

+ A binary real operator returning the sum of the two real arguments.

- A binary real operator returning the value of the second argument subtracted from the first.

***** A binary real operator returning the product of the two real arguments.

/ A binary real operator returning the division of the first argument by the second.

sqr A unary real operator returning the square root of the argument.

sqr A unary real operator returning the square of the argument.

sin A unary real operator returning the sine of the argument which must be an angle in radians.

cos A unary real operator returning the cosine of the argument which must be an angle in radians.

tan A unary real operator returning the tangent of the argument which must be an angle in radians.

asin A unary real operator returning the principle value, in radians, of the inverse sine.

acos A unary real operator returning the principle value, in radians, of the inverse cosine.

atan A unary real operator returning the principle value, in radians, of the inverse tangent.

exp A unary real operator returning the exponential of the argument.

log A unary real operator returning the natural log of the argument. Should be paired with a requirement that the argument be larger than 0.

mod A binary real operator returning the remainder after division of the first argument by the second.

The global parameter **add_param_range** specifies that the minimum and maximum values assigned in the parameters section be included in the constraints section. Only variable parameters listed in the **params:** section may be used. It is suggested to use **#define** within the **constants:** and **constraints** section to define constraints relative to constant parameters. Examples of constraint lines are:

```
A maximum value
    (assign parameter1 (if (> parameter1 MAX_VALUE) MAX_VALUE parameter1))

Alternative max. value
    (assign parameter1 (min parameter1 MAX_VALUE))

A minimum value
    (assign parameter1 (if (< parameter1 MIN_VALUE) MIN_VALUE parameter1))

Alternative min. value
    (assign parameter1 (max parameter1 MIN_VALUE))

A parameter as a maximum value
    (assign parameter1 (if (> parameter1 parameter2) parameter2 parameter1))
```

A.4 Addressing

Reax, GROMACS, and Generic have a 3 index addressing scheme.

A.4.1 Reax

A reax address looks like <section>:<entry>:<data> sections are:

global 0

atom 1 - each entry is an atom type

bond 2 - each entry is an atom pair

off-diagonal 3 - each entry is an atom pair

angle 4 - each entry is an atom triplet

torsion 5 - each entry is an atom 4-tuple (with x as wildcard)

Reax force field writing only uses the first force field file in the global section line(**ffield:**)

A.4.2 GROMACS

GROMACS: <section>:<line>:<entry>. The GROMACS force field addressing is similar, but not so much as rigid as Reax. GROMACS only uses the first force field file in the global section line(**ffield:**). The file is written section by section, line by line, and word by word in the format specified by the associated constant or parameter line.

A.4.3 Generic

Generic addressing is similar to GROMACS writing: <file>:<line>:<entry>, with the first address value specifying which of the files in **ffield** to write to.

A.5 Scattering

The scattering manager type will run an optimization without using any external commands. This removes the file i/o and fork/exec/wait overhead associated with the normal method of executing a ParOpt optimization. The scattering manager utilizes the same Nelder-Mead implementation as the Nelder-Mead manager, so all related Nelder-Mead parameters apply.

In the global section, field files should still be included, in order to determine the writing of final force field files. No executable lines are required. The differences in input files for the scattering manager are primarily in the **data:** section. Data lines are in the same form (see section A.3.4, but **analysis_executable** specifies an internal method of ParOpt. Comparison and analysis are wrapped into one function call, specified by the **analysis_executable** method. Entries for **comparator_executable** and its arguments are ignored. Total error is calculated in the usual fashion, by a weighted sum of each individual contribution. Available methods for defining the target function in the scattering manager are as follows ².

A.5.1 Scattering Data Lines

ChiScattering

(example)

```
electron_density: 3.0 0 ChiScattering 2 electron.field exp_data
```

This analysis method computes the underlying density associated with the **field_file** according to equation (A.3), symmetrizes the density, computes a cosine transformation at the **q** values in the **exp_data** file, and compares the values according to equation (A.4). The file **exp_data** should contain two column data, in plain text, and white space separated, with the first column having **q** values and the second column **F(q)**.

$$\rho^*(z) = \sum_i \left[g_i(z; \mu_i, \sigma_i) dz \frac{w_{\alpha_i}}{V_{\text{slice}}} \right] + \left(p_w(z) dz - \frac{V_{\text{slice}}}{v_{\text{bulk}}} \right) \frac{w_w}{V_{\text{slice}}}. \quad (\text{A.3})$$

²J. C. Fogarty, M. Arjunwadkar, S. A. Pandit, and J. Pan. “Atomically Detailed Lipid Bilayer Models for the Interpretation of Small Angle Neutron and X-ray Scattering Data”. In press. 2014

$$\chi = \sqrt{\frac{\sum_{i=1}^{N_q} \left(\frac{k_e F_e(q_i) - F_m(q_i)}{k_e \Delta F_e(q_i)} \right)^2}{N_q - 1}}. \quad (\text{A.4})$$

The scale factor for comparing experimental and model data is given by k_e :

$$k_e = \frac{\sum_i \frac{F_e(q_i) F_m(q_i)}{[\Delta F_e(q_i)]^2}}{\sum_i \frac{[F_e(q_i)]^2}{[\Delta F_e(q_i)]^2}} \quad (\text{A.5})$$

FixCenter

(example)
`fix_center: 1.0 0 FixCenter 1 volume.field`

This target function contribution penalizes total lipid volume probability greater than unity, which implies an unphysical negative value for water probability (see equation (A.6)) The file `volume.field` is analogous to the field file in **ChiScattering**, but with volume weights instead of electron numbers or neutron scattering lengths.

$$\chi_{\text{reg1}} = \frac{\int_{-\infty}^{\infty} H \{p_{\text{lipid}}(z) - 1\} [p_{\text{lipid}}(z) - 1] dz}{\int_{-\infty}^{\infty} H \{p_{\text{lipid}}(z) - 1\} dz}, \quad (\text{A.6})$$

FixWaters

(example)
`fix_waters: 100.0 0 FixWaters 2 volume.field ATOM_NAME`

This term incorporates the hydrophobicity of hydrocarbon chains into the target function. The position of the cutoff in equation (A.7) is set as the μ value for the atom with name `ATOM_NAME`.

$$\chi_{\text{reg2}} = \frac{1}{2\mu_{\text{cut}}} \left[\int_{-\mu_{\text{cut}}}^{\mu_{\text{cut}}} (1 - p_{\text{lipid}}(z))^2 dz \right]^{1/2} \quad (\text{A.7})$$

TotalVolume

(example)
`total_volume: 100.0 0 TotalVolume 2 volume.field VOLUME`

This adds in a comparison of the total molecular volume of the lipid, compared with a target value (`VOLUME`) according to equation (A.8).

$$\chi_{\text{volume}} = \frac{|(V_{\text{exp}} - \sum_i v_{\alpha_i})|}{V_{\text{exp}}}, \quad (\text{A.8})$$

A.5.2 Scattering Field Files

Scattering field files are required for the analysis routines associated with the scattering manager (section A.5.1). These files are plain text, white-space delimited and contain five columns:

```
name mu_name sigma_name volume_name weight
```

The name column is used for reference. All other columns must be parameter or constant line names (see section A.3.3 and section A.3.2), including the weight column. The first line of this file must contain information for water, where values for μ and σ are set to zero:

```
water ZERO ZERO      water_vol water_electrons
N      N_mu N_sigma head_vol  N_electrons
P      P_mu P_sigma head_vol  P_electrons
:
```

In this example using the approach in the ADP model ³, the following values would appear as constants: `water_electrons`, `N_electrons`, `P_electrons`, `water_vol`, and `ZERO`. Values which would appear in the parameter lines would be `N_mu`, `N_sigma`, `P_mu`, `P_sigma`, and `head_vol`. The fourth column is used to specify which density is being calculated (electron, neutron scattering length, or volume).

A.6 Output

All logging information indicating the status of a ParOpt run is written to stderr. What to write is specified by the option `verbosity` (section A.3.1). Additionally, ParOpt writes the following files, which are overwritten if they are present in the working directory:

`ffcontrol_N`: Each of the N control files specified on input is cpp'd and written into `ffcontrol_N`. This file is then read by ParOpt.

`ffcontrol_N.out`: After reading the `ffcontrol_N` file, all parameters (global, param, cons, and data) are written to the `.out` file. Defaults are also written. `ffcontrol_N.out` files are valid input files for ParOpt.

`comp_error`: ParOpt doesn't directly write to this file, but many tools do.

`final_error.out`: Errors are written at each convergence.

`times.out`: Section-wise information on the time elapsed while running.

`system_error.out` (sequential managers only):

`error.out` (sequential managers only):

`simplex_file` (Nelder-Mead, Scattering only) Prints a the simplex and its properties at each Nelder-Mead iteration. Each simplex iteration begin with a line with: "step", Nelder-Mead iteration index, move type (Initial, Reset, Reflect, Contract, or Expand), mean, rmsd, rmsd cutoff, simplex volume, highest value index, highest value, lowest value index, lowest value. For example:

³J. C. Fogarty, M. Arjunwadkar, S. A. Pandit, and J. Pan. "Atomically Detailed Lipid Bilayer Models for the Interpretation of Small Angle Neutron and X-ray Scattering Data". In press. 2014

```
step 0 Initial 7.165050e+04 2.828772e+05 1.000000e-01\\
2.478123e+01 19 1.643016e+06 27 1.756095e+02
```

A.7 Additional Tools

A.7.1 Comparison Tool

A general comparison tools is included:

```
ffcomp [-ImNoSst] --target_file (-t) targetfile
        --observed_file (-o) observedfile
        --index (-I) index_columns:data_columns
        --method (-m) method --single (-s)
        --seperator (-S) seperator_character
        --target_value (-v) value
```

with options:

- N:
 Normalize (off by default)
- target_file or -t targetfile:
 target data file
- observed_file or -o observedfile:
 observed data file
- index or -I index_columns:data_columns:
 format for data file(s) - each line will have index_columns integer indexing columns, follwed
 by data_columns number of columns.
- method or -m method:
 allowed methods (with abbreviations) are:
 - square_difference (squ or sqr):
 - chi_squared (chi):
 - percentage (per):
 - nothing:
- target_value or -v value:
 value to use instead of reading from file
- single or -s:
- seperator (-S) seperator_character:

A.7.2 Forcefield Extraction

```
ffextract -c control_file
```

This command expects one line of parameter values, with white space separation, on stdin writes force field file to stdout. The control file, which should be the post-preprocessed file `ffcontrol_N` (see section A.6 is required).

A.7.3 Parameter Extraction

```
param_extract -c control_file [-p]
```

This method reads a control file specified by the `-c` flag (required) and a set of force field files written as specified in the control file and writes updated parameter and constants lines to stdout. The control file should be the post-preprocessed file `ffcontrol_N`. The `-p` flag (optional) prints one line parameter values to stdout.

A.7.4 Reax Profiles

```
profile [-2 type1 type2 ] [-3 type1 type2 type3]\\  
        [-4 type1 type2 type3 type4] ffield_file
```

Types, for two- (-2), three- (-3), and four- (-4) body interactions, are specified by the index (beginning from 0) of the type in the single body parameters section of the force field file. Terms for output, which are printed in the order they appear on the command line, are specified by flags:

- `-t` Total energy
- `-be` Total bond energy
- `-be_s` σ bond energy
- `-be_p` π bond energy
- `-be_pp` $\pi\pi$ bond energy
- `-bo` Total bond order (uncorrected)
- `-bo_s` σ bond order (uncorrected)
- `-bo_p` π bond order (uncorrected)
- `-bo_pp` $\pi\pi$ bond order (uncorrected)
- `-vdw` van der Waals interaction energy
- `-dvdw` gradient of van der Waals interaction energy
- `-c q1 q2` Coulombic interaction, with charges specified by `q1` and `q2`

By default, total energy (`-t`) only is printed.

The range for output is specified with `-r_min`, `-r_max`, and `-dr`, each of which takes a single real valued argument. Parameters can be modified before output using the flag `mod` has the form:

```
-mod name order (1,2,3,4) type1, [type2,type3,type4] new_value
```

Valid **names** along with their associated **order** are: `bo_cut(0)`, `valency(1)`, `r_s(1)`, `r_pi(1)`, `r_pi_pi(1)`, `r_vdw(1)`, `gamma_vdw(1)`, `alpha_vdw(1)`, `p_bo1(2)`, `p_bo3(2)`, `p_bo5(2)`, `r_s(2)`, `r_p(2)`, `r_pp(2)`, `r_vdw(2)`, `gamma_vdw(2)`, `alpha_vdw(2)`

A.7.5 Luzzati Thickness

`luzzati ffcontrol field`

This method calculates the luzzati thickness for a scattering manager optimization. The control file should be the post-preprocessed file `ffcontrol_N`. The file `field` determines the density for which the thickness is calculated (see section A.5.2).

The Luzzati thickness (D), is by solving:

$$\int_0^D (1 - p_{\text{lipid}}(z)) dz = \int_D^\infty p_{\text{lipid}}(z) dz \quad (\text{A.9})$$

A.7.6 Partial Volume

`partial_volume ffcontrol field`

The control file should be the post-preprocessed file `ffcontrol_N`. The file `field` determines which atoms are included in the sum, and should have atomic volumes as the fourth column (see section A.5.2).

$$V = \sum_i v_{\alpha_i} \quad (\text{A.10})$$

A.7.7 Scattering

`run_scattering ffcontrol`

This computes all the target function component values for the initial point specified in `ffcontrol`.

A.7.8 Scattering Components

`scattering_component ffcontrol data_line`

This method takes a control file and a full data line, as specified in section A.3.4 (name: weight tolerance function argc-1 args...), and computes the target function component value associated with the line.

A.8 Getting Help

Read the manual (again)
or email: jcfogart@mail.usf.edu

Appendix B

Silica-Water Interface

B.1 A Note to the Reader

This chapter contains material from the manuscript:

J. C. Fogarty, H. M. Aktulga, A. Y. Grama, A. C. van Duin, and S. A. Pandit. “A reactive molecular dynamics simulation of the silica-water interface”. *J. Chem. Phys.* 132.17 (2010), 174704

B.2 Introduction

Amorphous Silica (a-SiO₂), its surface properties, and hydrolysis have been topics of research in diverse application domains, ranging from geo-sciences to nanoelectronics. The high dielectric constant and selectivity for chemical modification makes silica among the most widely used substrates in the design of nanoelectronic devices. Recent advances in molecular biology have demonstrated its use in devices capable of performing *in-vivo* screening of biomolecules and biomolecular processes, along with other applications in biotechnology that rely on surface modification of silica. Silica can serve as a substrate for biosensors, electronic components, and enzymes.^{154,155} Since these devices are often required to function in inhospitable aqueous cellular environments, a detailed understanding of their interactions with water is crucial.

Experimental methods such as infrared spectroscopy,^{156–158} x-ray crystallography,^{159,160} Nuclear Magnetic Resonance (NMR)¹⁶¹ and electron microscopy do not sufficiently describe the processes by which silica is corroded by water at an atomic scale.¹⁶² Molecular dynamics (MD) tech-

niques, due to their ability to probe nanoscale spatio-temporal processes, can provide valuable insights into this problem. Conventional classical MD has the ability to simulate bulk properties of α -SiO₂.^{38,163–165} It lacks, however, the ability to model chemical reactions, specifically the dissociation of water and the resulting recombination of O and Oh units with the silica surface. Consequently, while conventional MD simulations can yield reliable data on bulk silica, they do not sufficiently describe interfacial properties of interest.¹⁶³ In a significant effort, Garofalini et al. have simulated water-silica interaction using molecular dynamics with a dissociative water model.¹⁶⁶ Several new observations are made using a more general reactive potential that incorporates variable charges. The work also confirms a number of observations of Garofalini et al. A reactive potential allows changes in bond order which coincide with changes in electron densities, thereby implying modification of partial charges. The current work also extends the scale of reactive simulations.

Quantum mechanical *ab-initio* methods have been used to simulate chemical reactions at the silica–water interface.^{163,167} These simulations are typically limited to sub-nanometer length and picosecond time scales. For this reason, *ab-initio* approaches are unable to simultaneously describe bulk systems and interfaces. Due to limitations on scalability, surface characteristics of silica in *ab-initio* simulations are artificially constructed, as opposed to being derived from an annealing process.¹⁶⁸ These *ab-initio* simulations generally also ignore the interaction of the interfacial section with the silica bulk. Attempts have been made to bridge this gap using hybrid simulation techniques, whereby the surface sites are simulated using quantum calculations and bulk sections are simulated using classical MD.¹⁶⁹ This approach has potential drawbacks due to the interface between the *ab-initio* and MD regions of the system. Classical force fields must be tuned not only to fit experimental results, but also to interface with the *ab-initio* calculations. Inconsistencies between MD force fields and quantum calculations can result in unwanted changes in the structure of the system.¹⁶⁷

In this work, use a novel molecular dynamics forcefield, ReaxFF, developed by van Duin et al.¹⁴² is used. This method relies on the development of empirical force fields that mimic the quantum mechanical variation of bond order. ReaxFF replaces the harmonic bonds of conventional MD with bond orders and energies that depend on inter-atomic distances. Valencies, explicitly satisfied in MD simulation, necessitate many-body calculations in ReaxFF. The approach allows bond order and all bonded interactions to decay smoothly to zero, allowing chemical reactions within

a molecular dynamics framework. Consequently, ReaxFF can overcome many of the limitations inherent to non-reactive MD, while retaining, in large part, the desirable scalability. The applicability of ReaxFF to large scale silica systems has been questioned on the basis of its computational cost.¹⁷⁰ State-of-the-art implementations (SerialReax), however, have demonstrated excellent computational efficiencies, thereby greatly alleviating scalability concerns.

Results of a reactive molecular dynamics simulation of a water–silica system are presented. It is shown that ReaxFF is able to reproduce bulk properties of silica and water to a high degree of accuracy, silica surface properties in agreement with that of *ab-initio* calculations, and predict chemical reactions at the interface. At the time of the writing, no other computational approach satisfactorily addresses all three aspects of this problem simultaneously.

B.3 Methods

All of the molecular dynamics simulations in this chapter are performed using SerialReax molecular simulation software.¹⁴⁵ ReaxFF differs from a classical MD approach in several fundamental ways:

- Although no statistical mechanical approach would ever formally assume distinguishability, indistinguishability is completely required in the analysis of reactive simulations. ReaxFF allows atoms to move from one chemical species to another. Consequently, interpretation of ReaxFF data must assume the indistinguishability of particles, inherent to any quantum mechanical system, even though the assumption is not explicit in the simulation technique itself. Specifically, for the current system, oxygen atoms may transition between water and silica, and hence cannot be labeled as members of specific chemical species.
- The bond order term and its corrections force the ReaxFF potential to be inherently many body. Consequently, all force field terms dependent on bond order become many body interactions. This makes the calculation of a reactive potential more computationally expensive than a classical MD approach.
- The SerialReax model incorporates a charge equilibration technique (QEq) introduced by Rappé and Goddard.¹⁴⁷ This approach seeks to minimize electrostatic energy by assigning partial charges based on ionization potential, electron affinities, and atomic radii. The total

electrostatic energy and atomic chemical potential are given by:

$$E_Q(Q_1...Q_N) = \sum_A (E_{A0} + \chi_A^0 Q_A) + 1/2 \sum_{A,B} Q_A Q_B J_{AB} \quad (\text{B.1})$$

$$\chi_A(Q_1...Q_N) = \chi_A^0 + J_{AA}^0 Q_A + \sum_{B \neq A} J_{AB} Q_B. \quad (\text{B.2})$$

Total energy is then minimized with the constraints that total charge remains constant and that all atomic chemical potentials remain equal. Implementation of QEq requires the solution of a large system of linear equations with constraints. In SerialReax, this large system of equations is solved at every step using an efficient preconditioned linear solver (GMRES¹⁷¹).

- Reordering of chemical species within ReaxFF requires dynamic neighbor lists, even for bonded interactions, such as bond, angle, and torsion. This requires careful design and orchestration of dynamic data structures within SerialReax to minimize computational overhead w.r.t. classical MD.

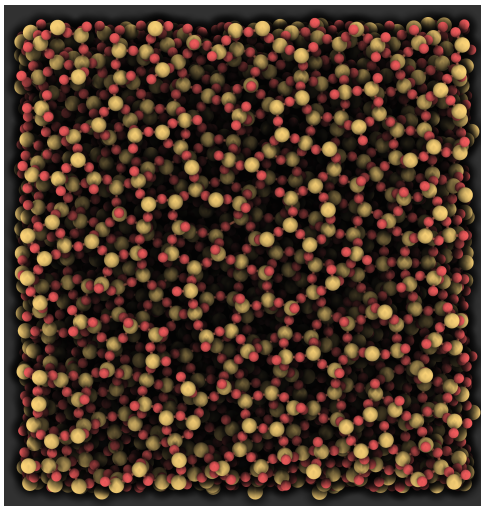
These differences add significant complexity to ReaxFF implementations, when compared to a classical MD approach. SerialReax relies on a range of sophisticated algorithms, data structures, and numerical techniques to minimize the cost of these computations. As a result, ReaxFF retains much of desirable scalability of classical MD, but adds considerable simulation power.

B.3.1 System preparation

Since ReaxFF updates bond order and bond order dependent quantities at every time-step, it requires a smaller time-step than conventional MD approaches.¹⁷² Chemical reactions occur over sub-picosecond time scales. Thus, all simulations were performed with a 0.5 fs time-step. All simulations, unless otherwise noted, were performed at a temperature of 300 K. A Nosé–hoover thermostat was used in each case to couple the system to a heat reservoir.¹⁷³ Whenever a constant pressure and temperature (NPT) ensemble was utilized, a Berendsen barostat and thermostat were used to couple to a bath.^{174,175} A Berendsen barostat was preferred over a Parrinello-Rahman barostat to avoid large pressure fluctuations, which can lead to unrealistic chemical reactions.

The amorphous silica system was constructed by initially placing 2000 silica (SiO_2) molecules randomly in a $67.4 \text{ \AA} \times 67.4 \text{ \AA} \times 20.0 \text{ \AA}$ box, resulting in a silica system with an initial density of 2.2 g/cm^3 . To eliminate atomic overlaps and bad contacts the system was energy-minimized in a microcanonical (NVE) ensemble for 50 ps. The bulk and surface properties of a silica system are highly dependent on the annealing procedure.¹⁷⁶ Simulation of an amorphous silica slab with the correct structural properties requires very high temperature annealing.¹⁷⁷ Hence, the system was annealed twice from 4000 K to 300 K. In the first annealing simulation, the system was heated to 4000 K and gradually cooled to 300 K with steps of 100 K per 4 ps using NVT ensemble. The system was again heated to 4000 K using an NPT ensemble, for ~ 75 ps, until the system completely melted and formed a uniform block with dimensions $52.110 \text{ \AA} \times 50.174 \text{ \AA} \times 36.477 \text{ \AA}$. The system was again cooled, still in an NPT ensemble, to 300 K in steps of 100 K per 4 ps. All NPT simulations were conducted at a constant pressure of 1 atm. The final annealed system was used to perform a continuous simulation of 185 ps for validation of silica properties. The same system was also used for the silica-water interface simulation. Fig. B.1 shows a freshly cut surface of annealed silica.

Figure B.1: Surface of annealed silica. Undercoordinated silicon atoms are bound to only three oxygen atoms. Undercoordinated oxygen atoms are bound to only 1 silicon atom. (Image generated with Qutemol¹⁷⁸)

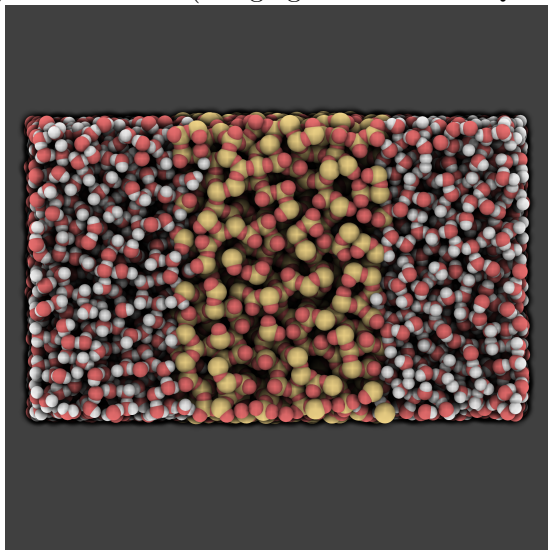


The water system was prepared by filling a box of size $51.800 \text{ \AA} \times 49.900 \text{ \AA} \times 23.600 \text{ \AA}$ (chosen to fit the silica system) with 2025 water molecules. This resulted in a density of 0.99 g/cm^3 . Water molecules were added with random alignment and location (avoiding overlap). The system was

then thermalized under an NVT ensemble for 664 ps. An NPT ensemble was not used due to the need to fit the water box to the silica slab. Although semi-isotropic pressure coupling would have allowed for an NPT ensemble, it was not available in the SerialReax version used in this work. This system was used both for the interface simulation and the validation of water properties. Position, bond, and angle data were output every 500 steps for water verification analysis.

The water and silica systems described above were combined by positioning copies of the water system normal to the z-axis and adjacent to the silica system, resulting in total system dimensions of $52.110 \text{ \AA} \times 50.174 \text{ \AA} \times 83.700 \text{ \AA}$. The system was then simulated under the NVT ensemble. NVT was used instead of NPT because the difference in compressibility between a-SiO₂ and water would cause unrealistic pressure effects. After 70 ps, the simulation was then restarted with velocities randomly generated to fit the set temperature. Velocities were reset in order to remove any artifacts from the system construction. The simulation was then run for a total of 580 ps. The system maintained thermal equilibrium with the heat bath for the final 150 ps. Therefore, these steps were used for structural and electrostatic analysis. As the reactions between water and the silica slab took place in the first 150 ps of the run, chemical analysis was performed on these initial steps. For the final 370 ps of simulation, position, velocity, bond, and angle data were written every 250 steps. Fig. B.2 shows a snapshot of the simulated system.

Figure B.2: Silica-water system snapshot: z-axis shown as horizontal. Oxygen (red), hydrogen (white) and silicon (yellow) atoms shown.(Image generated with Qutemol¹⁷⁸)



B.3.2 Force field parameters

To obtain a ReaxFF description capable of describing the reactions at the SiO_2 /water interface the ReaxFF Si/O/H parameters described by van Duin et al.¹⁴³ were modified. While these parameters could describe Si/SiO₂ interfaces, they were solely based on QM-data describing radical reactions, and were thus unable to describe the energetics related to proton-transfer reactions at the water/silica interface. To extend ReaxFFSiO(2003) to these reactions, first the O/H parameters were replaced with a set of ReaxFF O/H parameters fitted against water-clusters and proton-transfer reactions in $\text{H}_3\text{O}^+[\text{H}_2\text{O}]_n$ and $\text{OH}^-[\text{H}_2\text{O}]_n$ systems.¹⁴¹ Keeping the O/H parameters fixed, the Si/O, Si/Si and Si/H bond and angle parameters were subsequently re-fitted against the QM-based training set data used to fit ReaxFFSiO(2003). These data included bond dissociation curves for all Si/O/H bond combinations, angle distortion energies for all Si/O/H angle combinations and equations of state for bulk-Si and bulk-SiO₂-data. To augment this training set for water/silica cases, two additional sets of QM-based data (at the DFT/B3LYP/6-311G**++ level of theory) were added to the training set. These sets described (i) the binding and dissociation of a single water molecule from a $\text{Si}(\text{OH})_4$ molecules and (ii) reaction energies for the $\text{Si}(\text{OH})_4$ polymerization.¹⁷⁹ Fig. B.3 and Tab. B.1 compare the ReaxFF results to the QM-data for these two cases, indicating that ReaxFF can successfully describe both the non-reactive interaction of a water molecule with a hydroxylated silica surface and the reaction energies associated with silica formation.

Figure B.3: Dissociation of an oxygen group from silicon. Reax force field validation.

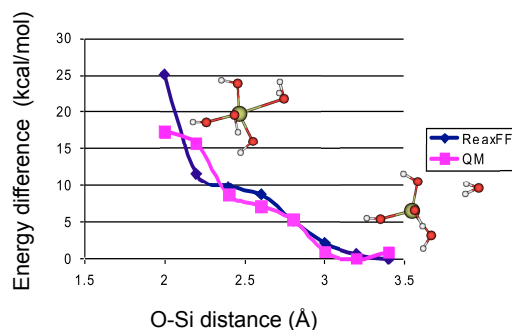


Table B.1: Free energy differences for force field validation.

| Reaction | ReaxFF | QM ¹⁷⁹ |
|---|--------|-------------------|
| Si(OH) ₄ +Si(OH) ₃ O ⁻ → (OH) ₃ Si-O-Si(OH) ₂ O ⁻ + H ₂ O (dimer anion) + H ₂ O | -23.0 | -20.8 |
| (OH) ₃ Si-O-Si(OH) ₂ O ⁻ + Si(OH) ₄ → (OH) ₃ Si-O-Si(OH) ₂ -O-Si(OH) ₂ O ⁻ + H ₂ O (trimer anion + H ₂ O) | -18.9 | -14.3 |
| (OH) ₃ Si-O-Si(OH) ₂ -O-Si(OH) ₂ O ⁻ + Si(OH) ₄ → branched quadrimer anion + H ₂ O | -24.0 | -28.9 |

B.4 Results

B.4.1 Validation of models

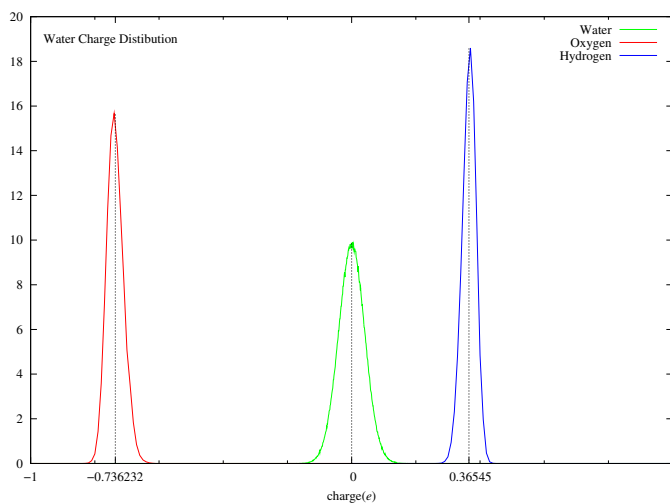
Validation of the water model involves computation of the average structural properties of a single water molecule and the static and dynamic properties of bulk water. Tab. B.2 shows a comparison of water properties computed using the ReaxFF model with those of *ab-initio* simulations, and experiments. In spite of several approximating assumptions, the current ReaxFF model reproduces several key properties of water. The oxygen–hydrogen bond length (d_{OH}) in the first line is the average distance between oxygen and hydrogen atoms that share a bond order $\geq \frac{1}{2}$. Of note is that the model reproduces experimental data for bond length and angle within the standard deviations. The partial charges (lines 3 and 4) on oxygen and hydrogen atoms (q_{O} and q_{H}) result from the QEq process. These charges (please see Fig. B.4) are lower than the popular fixed charge water models, such as SPC and SPC/E. Lower charges are expected since fixed charge models rely on a mean field approximation of large partial charges to reproduce bulk properties of liquid water.¹⁸⁰

Coordinate independence of the dipole moment vector requires molecular charge neutrality. While the QEq method implemented in SerialReax software imposes a constant charge constraint on the entire system, it does not require molecules to be charge neutral. hence, computation of the electrical dipole moment of water in ReaxFF is non-trivial. To enforce neutrality, half of the oxygen charge was assigned to each hydrogen atom, ignoring the hydrogen charge computed by QEq. Fig. B.4 shows the distribution of charge on water molecules. The standard deviation of water charge about a neutral molecule is $4.1 \times 10^{-2} e$, which indicates that a substantial number of water molecules are not neutral. however, the distribution suggests that the molecular charge constraint is only marginally violated by the QEq procedure. Consequently, approximations in-

Table B.2: Water Data. Reax results compared with *ab-initio*, SPC/E, and experimental. Water bond length is given as d_{OH} . RDF minimum and maximum values are local min and max. Diffusion is atomic self diffusion.

| Property | ReaxFF | SPC/E ⁶⁶ | <i>ab-initio</i> BLYP ¹⁸¹ | Experimental |
|-------------------------------------|------------------|---------------------|--------------------------------------|---------------|
| d_{OH} (Å) | 0.98 ± 0.04 | 1.000 | 0.973 | 0.957^{182} |
| HOH angle(°) | 104 ± 4 | 109.47 | 104.4 | 104.5^{183} |
| q_{O} (e) | -0.73 ± 0.03 | -0.8476 | — | — |
| q_{H} (e) | 0.36 ± 0.03 | 0.4238 | — | — |
| μ (D) | 2.1 ± 0.2 | 2.35 | 1.81 | 2.9^{184} |
| OO distance (Å) | 2.88 ± 0.2 | — | 2.95 | 2.98^{185} |
| OHO angle (°) | 168 ± 6 | — | 173 | 174^{186} |
| OO rdf first max. (Å) | 2.77 | — | 2.80 | 2.82^{187} |
| OO rdf first min. (Å) | 3.35 | — | 3.35 | 3.51^{187} |
| D_{O} (Å ² /ps) | 0.29 | 0.249 | 0.13 | 0.24^{188} |
| D_{H} (Å ² /ps) | 0.29 | — | — | — |

Figure B.4: Distribution of charges in water. Water molecule charge is distributed about neutral, with oxygen and hydrogen charges distributed about -0.736 and 0.365 e respectively.



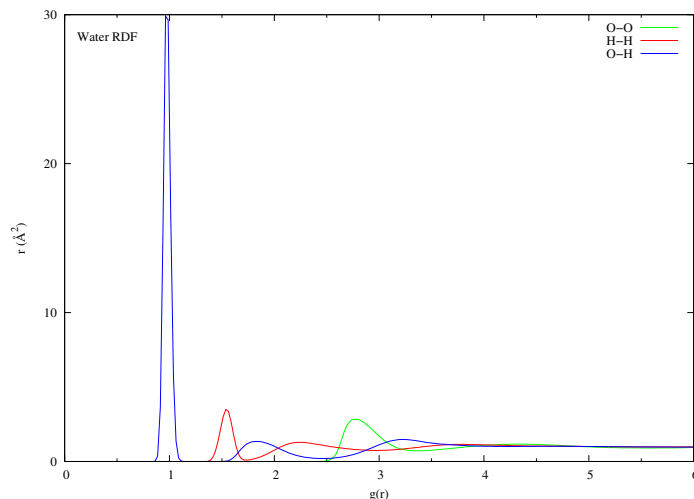
volved in computation of dipole moment introduce negligible error. The average dipole moment in a water cluster has been established by both experiment¹⁸⁴ and theory¹⁸⁹ to be larger than that of an isolated molecule. The dipole moment reported for *ab-initio* simulation is for an isolated molecule, while the value presented here is for bulk water.

The structure factor describes the scattering interaction between incident particles and the form of the scattering medium. Structure factor is not dependent on the nature of the interaction itself, but rather on the geometry of the system alone. A Fourier transform on the density function produces the structure factor.¹⁹⁰ Since the radial distribution function (RDF) is dependent on the density function, it can be used to predict the results of scattering experiments. The RDF between two atomic species is defined as

$$g(r) = \frac{N(r)}{4\pi r^2 \rho \delta r}, \quad (\text{B.3})$$

where $N(r)$ is the number of type 2 atoms in the shell between r and $r + \delta r$ around the type 1 atoms and ρ is the number density of type 2 atoms, taken as the ratio of the number of atoms to the volume of the simulation cell. Tab. B.2 (rows 8 and 9) list the properties derived from O–O RDF. The RDF for water is presented in Fig. B.5.

Figure B.5: Water Radial Distribution Function: RDF between oxygen oxygen (O–O), hydrogen–hydrogen (H–H) and oxygen–hydrogen (O–H) shown for bulk water.



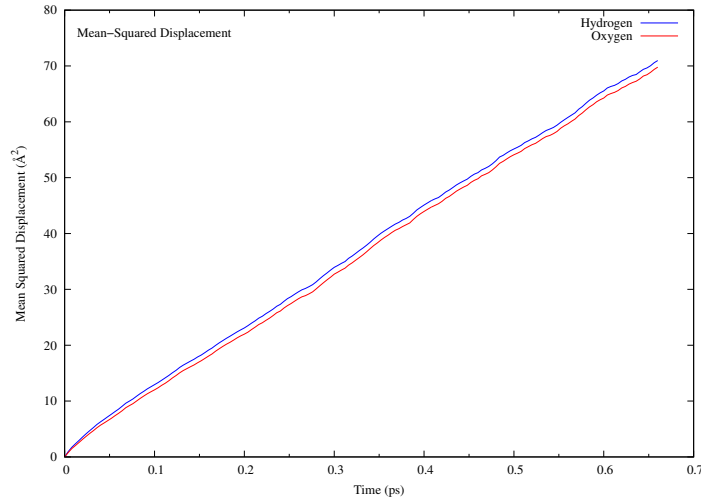
The diffusion coefficient (row 10 of Tab. B.2) of oxygen in water is representative of its transla-

tional mobility. It can be calculated from the long time behavior of the mean square displacement (MSD) of the atom using the Einstein relation:

$$D = \lim_{t \rightarrow \infty} \frac{\langle |\vec{r}(t) - \vec{r}(t_0)|^2 \rangle}{6(t - t_0)}, \quad (\text{B.4})$$

where \vec{r} is the position of the atom. The MSD of each atom was calculated over 166 ps intervals in the 664 ps production run. A least squares straight-line fit of the trajectory averaged MSD was then performed over the sub-interval from 40 ps to 166 ps within the 166 ps interval. Fig. B.6 shows the MSD of water oxygen and hydrogen as a function of time. The diffusion coefficients of these two atomic species do not show significant difference. This indicates that water diffuses in bulk water mostly as entire molecules without dissociation.

Figure B.6: Water Mean Squared Displacement: MSD shown for oxygen and hydrogen atoms within a pure water system.



The validation of the amorphous silica model involves computation of structural properties including mass density, bond lengths, bond angles, and radial distribution functions. Tab. B.3 reports these properties for the simulated model and a comparison with *ab-initio*, and experimental values.

Though the reported density for Silica in ReaxFF is lower than that from classical MD and experiments, the specifics of the annealing process have an effect on the structural properties of the final system. The silicon–oxygen bond length (d_{SiO}), calculated as the average distance between

Table B.3: Silica Data. Pure silica in Reax computed with Molecular dynamics and experiment. Silicon Oxygen bond is given as d_{SiO} . Max and min RDF values are local min and max values. Coordination is calculated as the average number with the first coordination shell. Full width at half maximum given in parentheses.

| Property | ReaxFF | MD | Experiment |
|-----------------------------------|--------------|-----------------------------|---|
| Density(g/cm ³) | 2.14 | 2.23 ¹⁹¹ | 2.20 ¹⁵⁹ |
| d_{SiO} (Å) | 1.59 ± 0.07 | 1.62 ± 0.05 ¹⁹² | 1.608 ± .004 ¹⁹³ |
| Si-Si dist.(Å) | 3.0 ± 0.2 | — | — |
| O-O dist.(Å) | 2.7 ± 0.3 | — | 2.65 ¹⁵⁹ |
| Si-O-Si angle (°) | 150 (21.5) | 152 (35.7) ¹⁷⁶ | 144 (38), ¹⁵⁹ 153 ¹⁹⁴ |
| O-Si-O angle (°) | 109.2 (20.9) | 108.3 (12.8) ¹⁷⁶ | 109.4, ¹⁹⁴ 109.5 ¹⁵⁹ |
| Si-O RDF 1 st max. (Å) | 1.56 | 1.595 ¹⁷⁶ | 1.608, ¹⁹³ 1.620 ¹⁵⁹ |
| Si-O RDF 1 st min. | 2.5 | — | — |
| Si-O RDF 2 nd max. | 3.90 | 4.12 ¹⁷⁶ | 4.15 ¹⁵⁹ |
| O-O RDF 1 st max. | 2.53 | 2.590 ¹⁷⁶ | 2.626, ¹⁹³ 2.65 ¹⁵⁹ |
| Si-Si RDF 1 st max. | 3.06 | 3.155 ¹⁷⁶ | 3.077, 3.12 ¹⁵⁹ |
| Si Coordination | 3.98 | — | — |
| O Coordination | 1.99 | — | — |

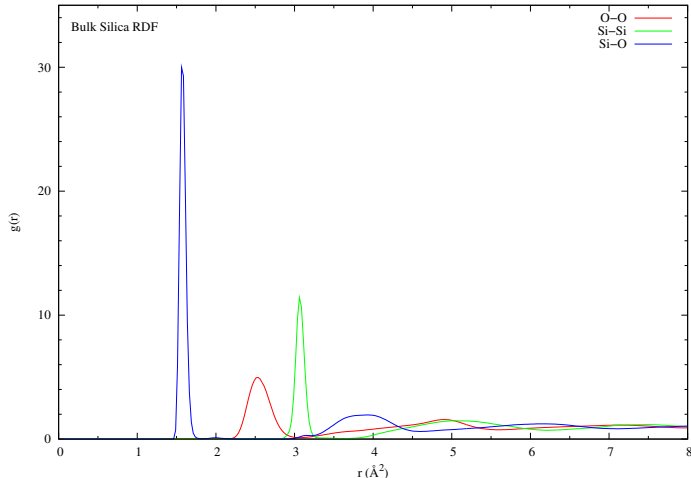
bonded silicon and oxygen atoms is in agreement with both experiment and MD. Angles within a tetrahedra (O-Si-O angles) and between two tetrahedra (Si-O-Si angles) are given, along with full width at half maximum (in parentheses). There is a wide range of Si-O-Si angles from 120-180, which is a fundamental difference between crystalline silica and a-SiO₂.¹⁵⁹

Important aspects of the RDF for a-SiO₂ are also reported in Tab. B.3. The maximum values in the RDF (RDF max.) show the most likely distance between the two atomic species in a specific coordination shell. A minimum value indicates the radius of the shell. The average coordination numbers for silicon and oxygen (Si and O coordination respectively), which are determined by counting the number of atoms within the first coordination shell, reflect the high level of coordination in bulk silica. The RDFs are presented in Fig. B.7.

B.4.2 Silica–water interface

ReaxFF allows the application of a broad range of analytical techniques. This approach yields information about the change in concentration of chemical species, which otherwise would not be available to MD simulation. ReaxFF is capable of probing length and time scales that permit utilization of thermodynamic and statistical tools. As a result, the silica–water interface described produces a large amount of information. The data can be interpreted to determine the structural

Figure B.7: Silica Radial Distribution Function: RDFs in pure silica shown for oxygen–oxygen (O–O), silicon–silicon (Si–Si), and silicon–oxygen (Si–O)

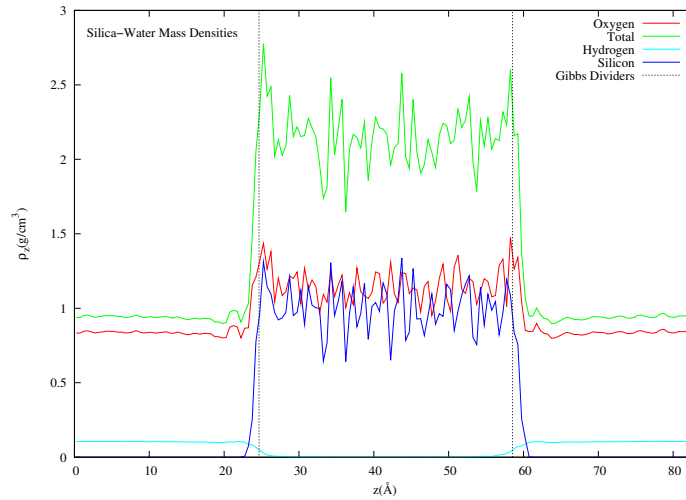


properties of the system, i.e., system geometry, bond characteristics, and molecular orientations. The scope of this analysis cannot be reached by *ab-initio* calculations. Unlike other classical MD approaches, ReaxFF generates data regarding the chemical composition of the system. Starting with pure water and pure silica, the final system models a hydroxylated silica surface covered by silanol (Si-OH) groups. The orientation of the water dipole moment gives rise to electric polarization along silica-water surface. This charge distribution gives rise to a measurable potential difference between the water bulk and the silica bulk. ReaxFF facilitates predictions for new experimental techniques to investigate this.

Since ReaxFF is able to simulate chemical reactions, the approach yields statistically valid samples of reactants produced by the silica-water interface, mostly notably silanol. The bond lengths and angles in SiOH groups are also able to be determined.

Fig. B.8 shows the mass density for the interfacial system, as well as its constituent atoms. The system was cut into 0.5 Å slabs along the z-axis, and a mass was calculated for each. The large variations seen in the silica bulk result from the lack of fluidity at room temperature in a-SiO₂, which prevents the averaging out of local density extrema. A peak in the total mass density near the surface results from an increase in the oxygen mass density. This local maximum is a result of the strong alignment of water dipole moment with the silica surface. As is clearly illustrated in Fig. B.8, a sharp boundary between water and silica does not exist.

Figure B.8: Mass Densities: Mass densities for the interfacial system, illustrating the locations of the Gibbs Dividers (dashed). Included is density of oxygen (red), hydrogen (teal), silicon (blue), and the total (green).



Thus, in order to clearly describe the interfacial area, a Gibbs dividing surface (z_G) was defined on each water-silica interfacial area. The dividing surface is chosen such that:¹⁹⁵

$$\int_{-\infty}^{z_G} (n(z) - n_1) dz = - \int_{z_G}^{\infty} (n(z) - n_2) dz \quad (\text{B.5})$$

where n is the number density of silicon atoms and the bounds at $-\infty$ and ∞ are defined as the points at which n is randomly distributed about n_1 or n_2 respectively. Silicon atoms located outside of these boundaries are defined as being located in the interface, while hydrogen atoms located within these boundaries are defined as being part of the silica bulk. The silica system had a width of 36.5 Å before the addition of water. An identical analysis was performed using hydrogen number density, which yielded results that differed by less than 0.5 Å. The width of bulk silica, defined as the distance separating the Gibbs dividers, is 33.9 Å.

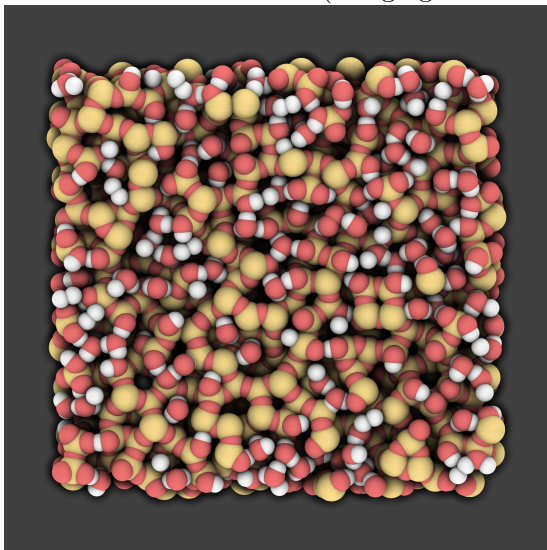
In the reaction with water and a freshly cut silica surface, hydrogen penetration into the silica bulk has been observed as evidenced by Fig. B.8. Tab. B.4 shows atomic diffusion in the silica bulk (within the Gibbs dividers) for hydrogen. The diffusion constant was calculated in the same manner as for bulk water (see above). MSD was calculated for the atoms beginning within the Gibbs dividers, regardless of later location. Straight line fitting was performed for trajectories of ~ 200 ps. It is proposed that water is able to diffuse through a thin film of silica via hydrogen

Table B.4: Silica-water Interface Data. $\text{Si}(\text{OH})_2$ is geminal silanol. Full width at half maximum given in parentheses.

| Property | Reax | Exp. |
|--|------------------|--|
| SiOH Conc. (nm^{-2}) | 5.4 | 4.3-5.2, ¹⁹⁷ 4.9 ¹⁹⁸ |
| $\text{Si}(\text{OH})_2$ to SiOH Ratio | 0.158 | 0.09-0.2 ¹⁹⁷ |
| D_h Silica Bulk ($\text{\AA}^2/\text{ps}$) | 0.0168 | — |
| Si Coordination | 3.96 | — |
| Si-O-h Bond Angle | $113(8.3) \pm 7$ | — |

hopping, i.e., rather than diffusing as whole units, water molecules dissociate at the surface, and hydrogens diffuse through, combining with other dissociated water molecules at the other surface. This mode of diffusion has been suggested by Bakos et al.¹⁹⁶ to occur when a water molecule encounters a narrow ring. The energy barrier for this reaction may be lower due to the presence of coordination defects on the surface and within the bulk of the silica slab and could therefore be a major path of diffusion through a thin silica film containing undercoordinated silicon atoms and NBO sites. Though hydrogen diffusion was observed in bulk silica, only a handful were able to cross the entire span. The diffusion constant shown here also yields estimates of the rate of this process. Order of magnitude estimates show that $\sim 10^9$ years would be required for a mole of water to diffuse through a thin (1 mm) silica container. In continuing investigations, a potential or chemical potential gradient could be used to increase this rate of crossing.

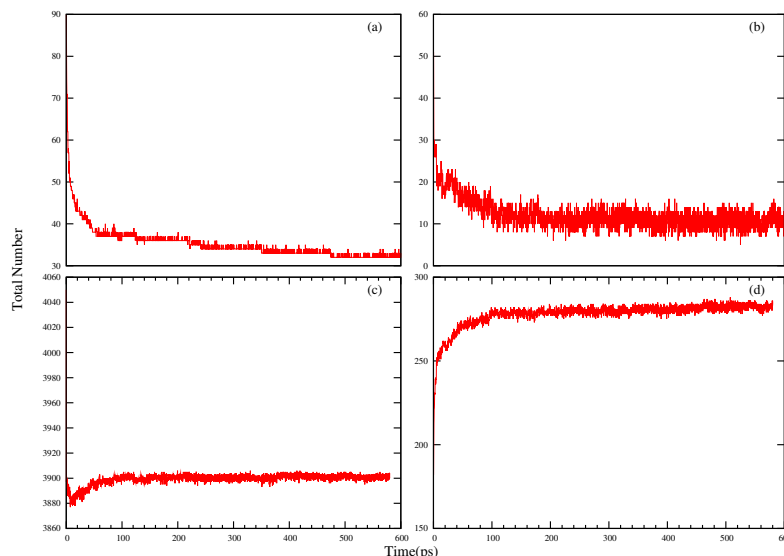
Figure B.9: Silica-Water interfacial area. (Image generated with Qutemol¹⁷⁸)



Since ReaxFF is able to simulate chemical reactions (the breaking and forming of bonds, which

produces changes in chemical species throughout a simulation), it enables examination of the stoichiometry of the silica-water reaction. Fig. B.10 shows the time evolution of total number of four types of chemical species in the silica-water system. Fig. B.10(a) shows the total number of undercoordinated silicon atoms, while Fig. B.10(b) shows the total number of nonbridging oxygen (NBO) atoms. NBOs are defined as oxygen atoms having a bond with a silicon atom and no other bonds. Both undercoordinated silicon and oxygen atoms are clearly consumed in the reaction, approaching an equilibrium value lower than the initial value. Silanol units (Fig. B.10(d)) are products of this reaction, increasing from an assumed initial value of 0. The water curve misleadingly indicates a production of water. The total number of water molecules started at 4050. The equilibrium value reached in the reaction is much lower than initial, indicating that water molecules have been consumed, despite the early transient behavior. Fig. B.10(c) also indicates that water dissociated within the first few picoseconds. It can be concluded that the time for the silica/water reaction to reach equilibrium is on the order of 1/2 ns.

Figure B.10: Silica-water species counts: The number of water molecules(c), silanol molecules(d), NBO atoms(b), and undercoordinated silicon atoms(a).



It is clearly observed that the main product of this reaction is silanol (SiOH), in its various forms. Based on Fig. B.10, primary reactants for this process are 3-coordinated Silicon atoms,

1-Coordinated Oxygen atoms, and dissociated water. A possible equation for this reaction is:

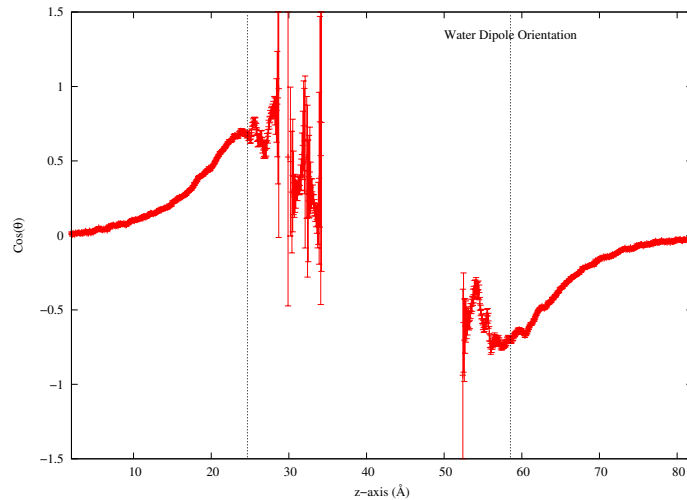


where Si and O are representative of under-coordinated (unfilled valency) silicon and oxygen atoms either on the silica surface or in the first few angstroms of the bulk.

Once the reaction has equilibrated, the equilibrium concentration of the main reactant and product, H_2O and SiOH , respectively are determined. It is found that 149 ± 2 water molecules have dissociated while 284 ± 2 silanols have formed. Since the ratio of water dissociated to silanol formed is approximately 1:2, as each water molecule dissociates at the silica surface, the free hydrogen atom bonds to an undercoordinated oxygen on the surface, and the OH group bonds to an undercoordinated silicon atom. The equilibrium for this reaction is reached when the surface is completely hydroxylated.

The electric dipole ($\vec{\mu}$) of water molecules in the silica-water interfacial system was computed using the same method described in section B.4.1. A plot of $\langle \cos(\theta) \rangle (z)$ (Fig. B.11), where θ is the angle between the z-axis and $\vec{\mu}$, yields a polarized alignment of $\vec{\mu}$ with respect to the silica surface.

Figure B.11: Water Dipole Moment: Orientation of water molecules given as Cosine of the angle between dipole moment and surface normal (z-axis). Error bars are given as inverse square-root of number of water molecules. Gibbs Dividers(dashed line) shown for reference.



This alignment of charges along the silica-water interface produces a potential difference between

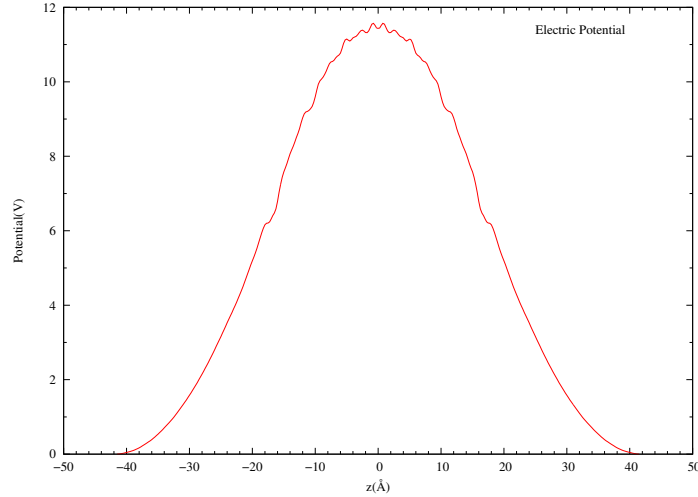
the silica bulk and the water (taken to be ground). Charge density was calculated by dividing the system into planes along z with width 0.5 \AA and taking the sum of charges enclosed. Electric potential was calculated by integrating charge density as a function of z :¹⁹⁹

$$\phi(z) - \phi(z_0) = \frac{-1}{\epsilon_0} \int_{z_0}^z \int_{z_0}^{z'} \rho(z'') dz'' dz' \quad (\text{B.7})$$

where z_0 is in the center of the bulk water, ϵ_0 is the permittivity of free space, and ρ is the charge density.

The method relies on the uniform distribution of charge within each plane along the z -axis. Since the system has periodic dimensions in x and y , each plane is mostly homogeneous. Therefore, this assumption does not introduce significant error. The system was divided in half and averaged in order to eliminate the large fluctuations in charge density that arise due to the lack of mobility in the solid silica atoms. The integrals were then performed from the outside-in (i.e., from water to the center of the silica slab) to yield the potential (Fig. B.12).

Figure B.12: Electric potential across the silica layer: averaged over both halves of the interface. Distance (z) shown is distance from the center of the silica slab.



Taking the water as ground, a potential difference of 11.5 V is measured between the center of the silica slab to the water, with the silica at a higher potential. This large difference results from the tight alignment of water dipoles with the interface surface normal. Water molecules at the interface are nearly 75% aligned, which leads to a large charge distribution at the surface. Ong

et al.,²⁰⁰ using second harmonic generation, determine the interfacial surface potential to be higher than that of bulk water. A direct comparison between their results and values reported here is not possible due to ambiguity in identification of the interfacial surface. Also of note is that the potential reported here, which includes contributions from surface and dipole potentials, has the same sign and similar order of magnitude, provided the interfacial surface is appropriately chosen. Since choice of this surface is to some extent arbitrary, a quantitative comparison of these potential values is imprudent.

B.5 Conclusion

A large silica water interface is studied using reactive molecular dynamics. This is the first simulation of its kind that achieves the length and time scales required to investigate various properties of such a system. A system of this size is necessary in order to produce a sufficient number of silanol units on the surface to allow for statistical analysis. A simulation time on the order of a half nanosecond is required to allow the system to reach equilibrium.

The molecular dynamics approach is based on a force field proposed by van Duin et al.¹⁴³ The specific Reax implementation (SerialReax) and force fields are validated by verifying structural properties of pure silica and water systems. Water properties such as bond angle and distance, charges, pair correlation data, and self diffusion constants were used to compare with experimental and *ab-initio* data. For pure silica, bond angles and distances, coordination, and pair correlations were used to validate the model.

The chemical reactions between reactive water and dangling bonds on a freshly cut silica surface are analyzed. These reactions were analyzed by observing the increasing concentration of silanol units and the decreasing number of dangling bonds on the silica surface. In these simulations, reactions involving silanol groups reached chemical equilibrium in ~ 250 ps. It is observed that water molecules penetrate a silica film through a process similar to the Grotthuss mechanism, called hydrogen hopping. In this process, hydrogen atoms pass through the film by associating and dissociating with oxygen atoms within the bulk silica as opposed to through diffusion of intact water molecules.

A large silica-water interface has been successfully simulated. The structural, chemical, and

electrical properties of the interface are in excellent agreement with experimental and quantum chemical data available today.

Appendix C

Publications

- J. C. Fogarty, M. Arjunwadkar, S. A. Pandit, and J. Pan. “Atomically Detailed Lipid Bilayer Models for the Interpretation of Small Angle Neutron and X-ray Scattering Data”. In press. 2014.
- J. C. Fogarty, A. Y. Grama, A. C. van Duin, and S. A. Pandit. “Optimization of a Reactive Molecular Dynamics Model for Water”. In preparation. 2014.
- J. C. Fogarty, S.-W. Chiu, E. Jakobsson, and S. A. Pandit. “Extension of a Coarse-Grained Model to Water–Ion Interactions”. In preparation. 2014.
- J. C. Fogarty, S.-W. Chiu, P. Kirby, E. Jakobsson, and S. A. Pandit. “Automated Optimization of Water–Water Interaction Parameters for a Coarse-Grained Model”. *J. Phys. Chem. B* 118 (2014), 1603–1611.
- H. M. Aktulga, J. C. Fogarty, S. A. Pandit, and A. Y. Grama. “Parallel reactive molecular dynamics: Numerical methods and algorithmic techniques”. *Parallel Comput.* 38 (2012), 245–259.
- A. Y. Grama, J. C. Fogarty, H. M. Aktulga, and S. A. Pandit. “N–Body Computational Methods”. In: *Encyclopedia of Parallel Computing*. Ed. by D. A. Padua. Springer, 2011, 1259–1268.
- J. C. Fogarty, H. M. Aktulga, A. Y. Grama, A. C. van Duin, and S. A. Pandit. “A reactive molecular dynamics simulation of the silica-water interface”. *J. Chem. Phys.* 132 (2010), 174704.

Appendix D

Copyright Permissions



Joseph Fogarty <jcfogart@mail.usf.edu>

Inclusion of article in Dissertation

EIC Office Journal of Physical Chemistry <eic@jpc.acs.org>
To: Joseph Fogarty <jcfogart@mail.usf.edu>

Tue, Oct 14, 2014 at 11:37 AM

Greetings;

Permission is granted with the understanding that the appropriate citations of the published work must be made.

Sincerely,

Davine for

George C. Schatz

Editor-in-Chief

From: Joseph Fogarty [mailto:jcfogart@mail.usf.edu]
Sent: Tuesday, October 14, 2014 9:43 AM
To: EIC Office Journal of Physical Chemistry
Subject: Inclusion of article in Dissertation

Hello,

I am preparing my dissertation for a PhD degree.

I have reviewed the document at

<http://pubs.acs.org/userimages/ContentEditor/1218205107465/dissertation.pdf>

which seems to give contradictory instructions for inclusion of published material.

I'm hoping to include substantial portions of an article published in JCPb, of which I am the first author:

<http://pubs.acs.org/doi/abs/10.1021/jp409545x>

I think all I need to do is include the ACS copyright create line. What would be the appropriate pages, all pages which

include material from the JCPb article?

Additionally, the dissertation will be publish electronically, so what link should be included to the JCPb article? Where should that be included?

Thank you very much for your time,

Joseph Fogarty

--

Joseph C Fogarty
University of South Florida
Department of Physics
4202 E Fowler Ave., ISA 2019
Tampa, FL 33620
Phone: (813)-974-2027
Fax: (813) 974-5813



Joseph Fogarty <jcfogart@mail.usf.edu>

Reuse of BBA Biomembranes submitted article in PhD Dissertation

Permissions Helpdesk <permissionshelpdesk@elsevier.com>
To: Joseph Fogarty <jcfogart@mail.usf.edu>

Tue, Oct 14, 2014 at 11:31 AM

Dear Joseph,

Thank you for your e-mail.

Permission is covered by the rights you retain as an Elsevier journal author as outlined at <http://www.elsevier.com/journal-authors/author-rights-and-responsibilities>, which include Inclusion in a thesis or dissertation, provided that proper acknowledgement is given to the original source of publication.

Our preferred acknowledgement should read "With permission from (author, article title, journal, volume, page range, Elsevier, copyright year)".

Should you require any further clarification, please let me know.

Best of luck with your dissertation.

Laura

Laura Stingelin
Permissions Helpdesk Associate
Elsevier

1600 John F. Kennedy Boulevard
Suite 1800
Philadelphia, PA 19103-2899
T: (215) 239-3867
F: (215) 239-3805
E: l.stingelin@elsevier.com

Questions about obtaining permission: whom to contact? What rights to request?

When is permission required? Contact the Permissions Helpdesk at:

☎ +1-800-523-4069 x 3808 ✉ permissionshelpdesk@elsevier.com

From: Joseph Fogarty [<mailto:jcfogart@mail.usf.edu>]

Sent: Tuesday, October 14, 2014 10:59 AM

To: Permissions Helpdesk

Subject: Reuse of BBA Biomembranes submitted article in PhD Dissertation

Hello,

I am the first author on an article submitted to BBA Biomembranes (BBAMEM-14-272R1)

I am preparing a dissertation for my PhD.

I would like to include substantial portions of the article in my dissertation.

How do I cite the article? Are there specific releases of copyright that I need?

Can I use the article in part, with modifications?

Thanks for your help and time,
Joseph Fogarty

--

Joseph C Fogarty
University of South Florida
Department of Physics
4202 E Fowler Ave., ISA 2019
Tampa, FL 33620
Phone: [\(813\)-974-2027](tel:(813)974-2027)
Fax: [\(813\) 974-5813](tel:(813)974-5813)



Joseph Fogarty <jcfogart@mail.usf.edu>

Use of manuscript in dissertation

AIPRights Permissions <Rights@aip.org>
To: Joseph Fogarty <jcfogart@mail.usf.edu>

Fri, Nov 7, 2014 at 9:47 AM

Dear Dr. Fogarty:

Thank you for requesting permission to reproduce material from AIP Publishing LLC publications.

Material to be reproduced:

"A reactive molecular dynamics simulation of the silica-water interface",

Joseph C. Fogarty, Hasan Metin Aktulga, Ananth Y. Grama, Adri C. T. van Duin and Sagar A. Pandit

J. Chem. Phys. 132 , 174704 (2010)

For use in the following manner:

Reproduced in your PhD dissertation.

Permission is granted subject to these conditions:

1. AIP Publishing LLC grants you non-exclusive world rights in all languages and media. This permission extends to all subsequent and future editions of the new work.

2. The following copyright notice must appear with the material (please fill in the information indicated by capital letters):

"Reproduced with permission from [FULL CITATION]. Copyright [PUBLICATION YEAR], AIP Publishing LLC."

When reusing a full article, the copyright notice must be printed on the first page of the reprinted article or book chapter. When reusing figures, photographs, covers, or tables, the notice may appear in the caption, in a footnote, or in the reference list.

In cases where the new publication is licensed under a Creative Commons license, the full copyright notice as stated above must appear with the reproduced material.

3. If the material is published in electronic format, we ask that a link be created pointing back to the abstract of the article on the journal website. This can be accomplished through the use of the article's DOI.

4. This permission does not apply to any materials credited to sources other than the copyright holder.

Please let us know if you have any questions.

Sincerely,

Susann Brailey

Manager, Rights and Permissions

AIP Publishing LLC

1305 Walt Whitman Road

Suite 300

Melville, NY 11747-4300



Tel. +1 516-576-2268

rights@aip.org

From: Joseph Fogarty [mailto:jcfogart@mail.usf.edu]

Sent: Monday, November 3, 2014 1:42 PM

To: AIPRights Permissions

Subject: Use of manuscript in dissertation

[Quoted text hidden]