Predicting Thermodynamic Properties over Combinatorially Large Chemical Spaces

---

A Dissertation

Presented to

the Faculty of the School of Engineering and Applied Science

University of Virginia

In Partial Fulfillment

of the requirements for the Degree

Doctor of Philosophy (Chemical Engineering)

by

Levi N. Naden

August 2016

APPROVAL SHEET

The dissertation

is submitted in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

AUTHOR

The dissertation has been read and approved by the examining committee:

Michael R. Shirts

Advisor

David L. Green

Roseanne M. Ford

Geoffrey M. Geise

Kateri H. DuBay

Accepted for the School of Engineering and Applied Science:

Craig H. Benson, Dean, School of Engineering and Applied Science

August

2016

Abstract

Current computational property prediction methods are limited in the number of molecules they can test at once. To predict properties for thousands or millions of molecules at once, new techniques must be developed with efficient computational scaling in the number of molecules simultaneously tested. In this dissertation, I develop a general approach to carry out computational alchemical free energy calculations using a variance minimized linear basis function approach. This approach provides a means to collect data for statistical free energy estimates that scales efficiently with the number of thermodynamic states or tested molecules. I achieve efficient scaling by splitting the potential energy function into a sum of pairs of basis functions and alchemical switches, so that energy is computed through matrix multiplication instead of simulation force code. The basis function approach allows construction of optimized, minimal variance alchemical switches from a single simulation, entirely in postprocessing, removing the need to optimize through iterative simulations. This is possible because each set of alchemical switches only changes the distribution of samples over the sampled thermodynamic space. I used this novel technique to find the variance minimized alchemical pathway to be: coupling Weeks-Chandler-Andersen decomposed Lennard-Jones forces with a capped repulsive nonbonded basis function, removing the fully coupled cap once the probability of observing atoms within the capped region is zero, and then coupling electrostatics through linear scaling. I show this pathway is just as statistically efficient as common soft core alchemical pathways

on small organic solutes in water.

I extend the basis function approach to look at atomic parameter searches in multiple parameter dimensions. The relative solvation free energy differences are computed for over 130,000 nonbonded parameter combinations of an ion. This system provides a simple problem where only one particle is alchemically modified to better focus on development of multidimensional sampling techniques. The computational effort of generating energies needed for free energy analysis drops from over a thousand CPU years to tens of CPU seconds because of my basis function approach. I compute free energies, entropy, enthalpy, and radial distribution functions of arbitrary parameter combinations using only the data from 203 sampled states. This work also creates an adaptive sampling process to generate mutual phase space overlap. The phase space overlap of sampled states is monitored alongside the mean and maximum uncertainty to determine convergence in a multidimensional space.

I develop a method to predict solvation properties of a combinatorial number of molecules simultaneously from a single simulation by combining the computational efficiency of the basis function approach with the multidimensional free energy convergence techniques. I estimate solvation free energies of 10^3 molecules combinatorially constructed by independently mutating 30 R-groups on a benzene core with separate basis function sets, creating 30 alchemical dimensions to sample. This is a practical system where the multi-atom R-groups are alchemically changed at different rates, creating complex interactions between R-groups and the solvent. I sample the large chemical space through Hybrid Monte Carlo (MC) and λ-dynamics to avoid pre-populating MC moves in 30D space, and to avoid numerical instabilities associated with λ-dynamics. The basis function analysis provides up to 145,000x speed-up over relying on simulation force code to compute energies required for free energy estimation.

Contents

Chapter 1

Introduction

Computers aid us in nearly every modern engineering design problem from airplanes such as designing, assembling, and simulating the flight of the entire Boeing 777 [1]; to space shuttle flight maneuvers and computational fluid dynamics simulations of re-entry [2], to placing over half a billion transistors on computer processors in a way that efficiently dissipates heat [3], or to laying out chemical plant piping [4–6]. So why do we not hear about "computer designed molecules?"

Several areas of chemical engineering have been advanced by computers. Engineers are modeling gas diffusivities and solubilities in polymeric membranes more frequently in computers [7, 8]. Simulations allow mechanical property estimation and prediction of nanoparticle coatings [9, 10]. The drug design field has been developing a wide array of computational methods such as free energy perturbation, docking, structure-based design, and pharmacophore identification to name a few [11–14]. Despite all these advancements, there are many challenges to large-scale computational chemical design.

A major challenge in chemical design is the immense number of molecules that could be created. For instance, there are over 10^{60} possible small organic molecules weighing less than 500 Daltons [15]. Methods like high-throughput screening can analyze a combinatorially large set of molecules to find new chemicals and materials for given applications [16, 17]. However, these screening methods become too costly as the chemical complexity increases, and successful chemicals from these methods have been relatively rare and still mostly serendipitous [18]. That same amount of money could be put towards computing time on readily available computers. Thus as computing becomes cheaper, computational prediction techniques will become cost competitive with, or reduce the cost of, leading experimental techniques. For instance, the United States Government reports that computational techniques have reduced the cost of drug development by $130 million on top of reducing the average research time by one year [19, 20].

There have been a number of hardware developments in the last decade or so

which have vastly improved our predictive capabilities. The proliferation of multi-core CPUs on desktop computers and clusters have greatly increased serial and parallel computational power. Specialized ANTON processors allow us to simulate milliseconds of simulations within two months [21]. The advent of GPUs have added an order of magnitude of speed to simulations, modeling, visualizations and more due to the GPU's ability to handle massively parallel processing [22–25]. Distributed and cloud solutions such as Folding@Home [26], Amazon AWS [27], and Google Exacycle [28] has made cost-effective resources available to everyone which used to take dedicated, self-maintained hardware.

Molecular simulation software developers have taken advantage of these hardware advancements to develop more powerful chemical simulation software. More complex chemical interactions are being included in models such as polarization in AMOEBA [29–32], and hydrocarbon and transition metal catalyzed reactions in ReaxFF [33, 34]. We are also seeing nearly every classical force field relying on quantum mechanical calculations to improve their accuracy. Molecular simulation software is being made available to the public such as GROMACS [35–38], OpenMM [32, 39], HOOMD-blue [40–42], and TINKER [29, 31, 43]. We are also seeing releases of commercial software for industrial computational drug design in FEP+ [44]. These advances have given us a strong platform with which to accurately and straightforwardly simulate a given molecule. Now, the bottleneck is no longer the hardware and software. Instead, the new bottleneck is efficiently estimating properties on many different molecules to determine which ones to synthesize.

The few exiting techniques for looking at chemical properties in large chemical spaces do not scale well beyond very simple changes in molecular structure. Virtually all publications for chemical property estimation look at the properties of a single molecule type in a single surrounding environment per simulation. Simultaneously gathering data from multiple chemically distinct molecules currently requires running

independent parallel simulations. Outside of the drug design field, only data mining techniques have met with moderate success for simultaneous multi-molecule type property predictions [45–47]. They unfortunately require substantial reference data and are very prone to over fitting as small changes in molecular structure can result in large changes to thermodynamic properties, even for approaches which couple data mining with experimentation to fill in the gaps [48]. Within drug design, the methods of simultaneously estimating properties in chemical spaces are limited to either compounds which roughly share the same volume [49, 50], or can only scale to a few molecular structure changes before computational time becomes a limiting factor [51, 52]. There is substantial room for improvement in developing new methods for estimating properties across large chemical spaces. The hardware and software advances over the last decade have reach the point that those methodological advances can now be developed.

My dissertation work develops a method of estimating thermodynamic properties in a combinatorially large chemical space by taking advantage of powerful statistical tools alongside GPU accelerated molecular simulations. This method allows one to compute the properties of thousands of molecules simultaneously from a single simulation, and could be straightforwardly extended to millions or more. This approach bridges the current gap in computational property prediction by designing a statistically efficient simulation, sampling multiple molecules simultaneously, and analyzing those samples in computationally efficient ways. This dissertation focuses on classical molecular simulations as they are better suited to sampling many configurations the molecules can adopt orders of magnitude faster relative to quantum simulations [53]. The application of this approach to chemical property prediction required the development of several pre-requisite methods, covered in the upcoming chapters.

This work specifically focuses on predicting the relative free energy of solvation for small, organic molecules. However, the methods are relatively general and would work

for infinite dilution scenarios in any solvent. A large number of other thermodynamic properties can be derived from various derivatives in free energy, and a free energy difference itself provides information about binding affinity, equilibrium partitioning, and activity (from excess free energy). I choose to look at the solvation free energy difference as it can be a more difficult thermodynamic property to experimentally measure, but easier to estimate in simulation with the right statistical tools.

This dissertation is organized into the following way. The remainder of this chapter establishes the statistical mechanics and current computational techniques to predict molecular properties in classical condensed phase simulations. Chapter 2 designs a basis function method to simulate molecules in a statistically efficient way which minimizes the number of required samples, and in a computationally efficient way to reduce the computational time needed to estimate thermodynamic properties. Chapter 3 explores different designs and practical sampling options for the basis function method to identify the optimal parameters for simulating with real molecules. Chapter 4 applies the basis function method to the problem of determining optimal ion parameters and shows how the methods can compute thermodynamic properties for hundreds of thousands of parameter combinations in a computationally efficient manner. Chapters 2-4 are my current publications modified slightly to fit in as a chapter [54–56]. Chapter 5 applies the basis function method to the solvation properties of multiple molecules built combinatorially from a common structure. This last chapter covers implementing the basis functions in a simulation package, applying the basis functions to sample combinatorial chemical space from a single simulation, analyzing the simulation to compute thermodynamic properties at multiple chemicals, and the current implementation limitations as well as how improvements could be made in the future.

1.1 Estimating Properties by Computer Simulation

Computer simulations provide an attractive supplement or precursor to laboratory experiments. Simulations allow prediction of physical, thermodynamic, transport and other properties for chemicals or materials of arbitrary complexity, including mixtures, with moderate accuracy [57–63]. Data from simulations can help estimate molecular properties without a chemical laboratory space or requiring synthesized chemicals. Simulations cannot completely replace laboratory testing, but instead provide better starting points for experimental efforts. The drug discovery field, for instance, uses simulations to identify sets of favorable leads and pharmacophores which can then be synthesized and tested in a laboratory, ruling out many structures along the way [12, 13, 53, 64]. Chemists can generate hundreds to thousands of molecules from these leads, speeding up the design process.

Physics or statistics based molecular simulations can provide property estimates when knowledge based or semi-empirical methods cannot. Group contribution methods such as UNIQUAC [65] and UNIFAC [66] cannot capture all the subtle interactions between the different molecules such as multi-residue proteins interacting with complex drug molecules [67]. The quantitative structure-activity relationships (QSAR) method helps predict properties for biological molecules similar to group contribution methods. QSAR methods examine the response that molecular fragments induce in a biological system and determine the entire molecule's interaction by the sum of its parts [68–70]. However, QSAR requires a large number of synthesized molecules which can be screened in a laboratory to determine the response in a biological system, limiting effectiveness in searching novel, unsynthesized chemicals. Molecular dynamics (MD) or Monte Carlo (MC) simulations capture all of the detailed interactions by calculating the interatomic forces of each particle on every other particle. Simulations can be run on

both synthesized, and unsynthesized molecules, providing a way to test any conceivable chemical. The force field describing the interatomic forces is frequently applicable to a specific class of molecules, such as organic and pharmaceutical molecules [71–76]. Even the quality of property estimates of simple systems which are generally well described by group contribution methods can be enhanced by simulations [77, 78].

Estimating molecular properties requires varying degrees of sampling per molecular system. For instance, computing the heat capacity of a system will take several orders of magnitude more samples than computing the internal energy to the same precision [79]. The number of required samples also depends on which molecule is being studied. Computing accurate protein folding kinetics can require nanoseconds (ns) to milliseconds (ms) and higher of simulation, depending on the protein [80].

1.1.1 Computational Free Energy Methods

Estimating thermodynamic properties by simulation requires defining the physical interactions in the molecular system. A simulation must define a set of atoms, the Hamiltonian, the statistical ensemble being sampled, and all ensemble constant properties such as pressure, temperature, and simulation volume. The Hamiltonian consists of the all atomic interactions defined by the force field and external forces acting on the atoms such as position restraints or magnetic fields. For example, consider simulating a single methane in a box of water under a constant number of particles (N), pressure (P), and temperature (T); also known as the isobaric-isothermal ensemble (NPT). The defined physical interactions are all of the atoms for the water and methane, all bonded and nonbonded forces, and finally, the pressure and temperature the simulation should be kept at. We refer to this collection of physical interactions as "the thermodynamic state" which differs slightly from the standard macroscopic thermodynamic state since we must specify interatomic interactions as part of our model.

We can compute useful thermodynamic properties through derivatives in the free energy or free energy differences. The free energy alone does not provide useful property estimates as it can only be computed up to an additive constant in simulation. If the potential energy of a system was shifted by a constant amount, the free energy would also be shifted by that amount. Fortunately, only free energy differences or derivatives provide useful thermodynamic estimates, so this additive constant cancels out when computing thermodynamic properties. Computing accurate free energies between two states requires generating samples which connect the states in a statistically low error manner.

Statistical mechanics provides formulas to compute thermodynamic properties from molecular models. Virtually all of these formulas depend on statistical averages estimated from the phase space of the thermodynamic state. The phase space of any thermodynamic state is the complete set of position and momentum, or "configurations," that the atoms in that state can access. The phase space of all configurations can be counted in continuous space and related to the partition function, Z, in a simplified form as

$$Z = \int_{\Gamma} \exp\left(-u(\mathbf{x})\right) \, \mathrm{d}\mathbf{x} \tag{1.1}$$

u is the reduced potential energy function, $\mathbf{x}$ is a given configuration, and $\int_{\Gamma}$ is an integral over the phase space volume, which is over all positions and momentum of all particles in the system.

The reduced potential provides a convenient function to generalize the definition of partition function and thermodynamic property estimates. I define the reduced potential energy function in Eq. (1.1) as

$$u(\mathbf{x}) = \beta \left[U(\mathbf{x}) + PV(\mathbf{x}) + \mu\mathbf{N}(\mathbf{x})\right] \tag{1.2}$$

where $\beta = 1/(k_B T)$, U is the potential energy function, V is the volume, and μ is

the chemical potential. The exact form of Eq. (1.2) used in Eq. (1.1) depends on the sampled statistical ensemble. For instance, in the canonical ensemble, held at constant T, constant V and constant N (NVT) only the $U(\mathbf{x})$ term appears in the reduced potential, but in the NPT ensemble, the $PV(\mathbf{x})$ term will also appear. Eq. (1.1) assumes the particle masses are constant, allowing an analytical solution to the kinetic energy contribution to the partition function. This simplification generates a leading constant which has been omitted for simplicity. The reduced potential can be used to estimate either the Helmholtz (A) or Gibbs (G) free energy depending on the sampled ensemble. I generalize this choice by defining a general free energy, F, and computing it from the partition function as

$$F = -k_B T \ln Z, \tag{1.3}$$

with a free energy difference computed as

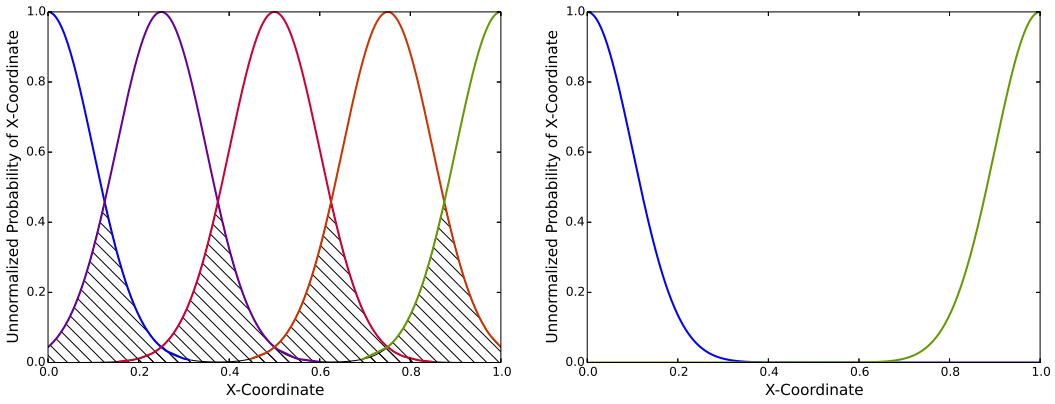
$$F_j - F_i = (-k_B T \ln Z_j) - (-k_B T \ln Z_i) \tag{1.4}$$

$$\Delta F_{ij} = -k_B T \ln \left(\frac{Z_j}{Z_i} \right). \tag{1.5}$$

Generally, we cannot directly compute the partition function, but we can compute a ratio of partition functions [81]. However, estimating this ratio requires statistical information about the relation between the two states, and this requires phase space overlap.

Accurate free energy estimates require phase space overlap between states of interest. The phase space overlap is a measure of how many configurations from a given state are observable in another state. This measure provides a relationship between two states, allowing more accurate estimate of the partition function ratio. If there is little to no phase space overlap between the states, we cannot compute accurate free energy differences [81]. We can sample at multiple states along a thermodynamic

path connecting the target states to generate phase space overlap through a series of overlapping states. As an example, Fig. 1.1 shows a case where I draw samples from harmonic oscillators with different centers. Little to no phase space overlap exists when I only draw samples from the end states as in Fig. 1.1b. If I sample along multiple intermediates, Fig. 1.1a, I create a path of good phase space overlap, and thus make it possible to calculate a better estimate of the free energy difference.



(a) (b)

Figure 1.1: Sampling multiple intermediate states creates phase space overlap between the end states. Shown is the unnormalized probability of generating a given x-coordinate for a set of harmonic oscillators whose centers are distributed along the x-axis, shown as different colors. The harmonic oscillators in (a) create a path of overlapping phase space, shown by hatches. No such phase space overlap exists in (b), so estimates of the free energy difference will be poor.

Since free energy is a state function, the free energy difference between two target states can be estimated as the sum between any series of states, i.e.

$$F_1 - F_N = -k_B T \ln \left[\left(\frac{Z_1}{Z_2} \right) \left(\frac{Z_2}{Z_3} \right) \cdots \left(\frac{Z_{N-1}}{Z_N} \right) \right]. \tag{1.6}$$

Those running free energy simulations frequently choose computationally efficient thermodynamic paths to lower the computational costs of the free energy calculations. The thermodynamic path we choose does not have to be physically re-creatable in a laboratory, since free energy is a state function. A commonly chosen non-physical path

is to change the atomic interaction parameters such as bond lengths, Lennard-Jones parameters, and charge so at one end of the path state A is represented, and state B is represented at other end. Linearly scaling the potential energy along the path provides this transformation and can be represented with

$$u(r, \lambda) = (1 - \lambda)u_A(r) + \lambda u_B(r) \tag{1.7}$$

where r is the interatomic distance, and λ is the transformation variable. For example, consider the relative solvation free energy between benzene (state A) and phenol (state B). At state A, the OH group does not interact with its surroundings so u_B does not contribute to the energy and is effectively not present in the system. As λ goes from 0→1, one of benzene's hydrogens contributes less to the potential and the OH group contributes more until only the OH group is present. Researchers in molecular simulation have designed many thermodynamic paths between molecules or for inserting molecules into dense fluids [14, 57, 81–89], each with their own strengths and weaknesses [90–92]. There has also been limited effort in designing paths to directly maximize phase space overlap [54, 55, 84, 93–97]. Estimating the free energy from these paths is also done with any number of methods [81, 93, 98–101], where each method also has its own strengths and weaknesses [60, 102]. Although significant effort has been put into computationally efficient paths, these only make it faster to carry out estimates of single free energy differences. But making single calculations faster does not reduce the scaling of the exponentially large problems of searching through chemical spaces.

1.1.2 Current Limitations in Simulation-Based Estimates of the Free Energy

Although computer power and accuracy is rapidly increasing, we mostly only estimate free energy for a limited number of molecules per simulation. One of the major limiting factors to estimating multiple molecules' properties is representing them in simulations. Consider again the benzene to phenol example, there is a clear set of atoms whose interactions change along a thermodynamic path. However, how would we transform between all the different amino acids in a single simulation? A few researchers have designed thermodynamic paths connecting multiple molecules through multidimensional paths [51, 52], shared volume methods [49, 50], and iteratively updating the thermodynamic path [91, 103, 104]. The researchers implementing these methods have been able to generate free energy differences for 2 to 10^3 structurally similar molecules, with more theoretically possible [50]. As the number of target molecules, and thus thermodynamic states, increases, the computational cost of generating the statistical information required for free energy estimates also increases.

Free energies estimators require the potential energy of each configuration evaluated at multiple thermodynamic states. The number of potential energies required for free energy estimates depends on which estimator is chosen. Sophisticated free energy estimators such as the Weighted Histogram Analysis Method (WHAM) [99, 100] or the Multistate Bennett Acceptance Ratio (MBAR) [101] operate by analyzing the mutual phase space overlap between all states simultaneously. Analyzing all mutual phase space overlap provides higher accuracy estimates for the free energy than methods which only estimate free energy differences from two states such as the Bennett Acceptance Ratio (BAR) [105] or from a single state such as Exponential Averaging (EXP) [98]. The free energy estimated by MBAR, the lowest bias estimator [101], is

computed by

$$F_i = -\ln \sum_{j=1}^{K} \sum_{n=1}^{N_j} \frac{\exp\left(-u_i(\mathbf{x}_{jn})\right)}{\sum_{k=1}^{K} N_k \exp\left(F_k - u_k(\mathbf{x}_{jn})\right)} \tag{1.8}$$

where k loops over all sampled states, j loops over all states (sampled and unsampled), n loops over all N_j samples in state j, and u_i indicates the reduced potential energy from state i. $u_i(\mathbf{x}_{jn})$ and $u_k(\mathbf{x}_{jn})$ indicates that the potential energy of every collected sample must be re-evaluated at every thermodynamic state, not just the state the configuration was drawn from.

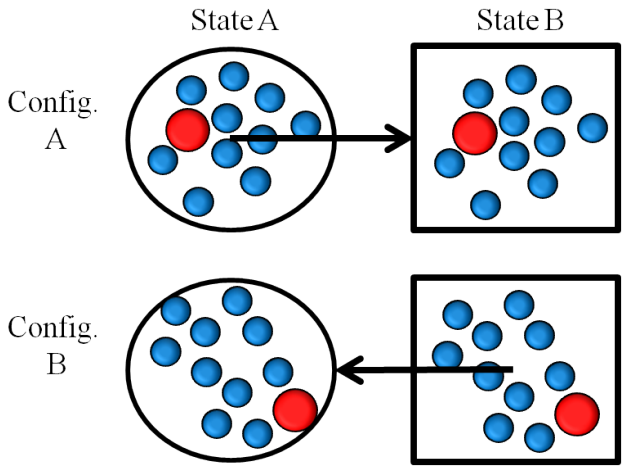


Figure 1.2: Reweighting methods determine the probability of observing any configuration in any state. Configurations (colored circles) are generated in two separate states (outline shapes) and the potential energy for each configuration is evaluated in thermodynamic state A and thermodynamic state B. These potentials provide the statistical information necessary to estimate free energies.

The process of re-evaluating potential energy is visually represented in Fig. 1.2. The figure shows two states and two configurations we want the potential energies of. We evaluate the potential energy for each configuration, generated from its own state, at both states. All the free energy estimators mentioned previously require us to carry out the re-evaluation process multiple times. Re-evaluating potentials for BAR, WHAM, and MBAR helps estimate the statistical weight each sample has to adjacent (BAR) or all other (WHAM, MBAR) states, this process is known as reweighting. Once samples are reweighted, the free energy can be estimated from the desired method.

The next chapters look at my designs for thermodynamic paths which apply these fundamental statistical mechanics to minimize the computational cost of the energy re-evaluation and minimizing the samples required to reach a target statistical error.

Chapter 2

Designing Basis Functions to Compute Thermodynamic Properties

2.1 Introduction

This chapter has previously been published [54] as: L. N. Naden, T. T. Pham, and M. R. Shirts. Linear Basis Function Approach to Efficient Alchemical Free Energy Calculations. 1. Removal of Uncharged Atomic Sites. *J. Chem. Theory Comput.*, 10:1128-1149, 2014.

Designing new molecules with desired thermodynamic properties requires efficient prediction methods to explore even a small fraction of combinatorial chemistry space. [15] Accurate simulation-based methods allow us to predict thermodynamic properties sensitive to small molecular changes, making them highly valuable for large chemical space screening. Two thermodynamic properties that have been studied most intently by simulation methods are the free energy of solvation and ligand binding free energies, as these quantities can provide insight in to the efficacy of drug candidates. [106] There are also many other properties of physical interest that can be related to free energies, including solubilities [107] and membrane permeabilities. [108]

"Alchemical" transformations are common computational methods for calculating free energy differences. In these methods, the free energy is estimated along a computationally efficient, often nonphysical thermodynamic path connecting the end states of interest. Estimating the free energy along such a path can be performed with several methods each having their own advantages and disadvantages. [60, 102] Choosing a statistically efficient pathway, however, is a non-trivial issue. [84–88, 95–97] In virtually all methods, we must simulate intermediate states along the coupling pathway to obtain a free energy with acceptable statistical error. Some methods require the potential of a configuration to be evaluated at neighboring states along the pathway, such as the Bennett acceptance ratio (BAR) [105] or free energy perturbation. Multistate methods require the potential of a configuration from one state to be evaluated at every other state, such as the Weighted Histogram Analysis Method (WHAM) [99, 100] or multistate BAR (MBAR). [101] Thermodynamic integration

(TI) requires derivatives along the coupling path to be computed at the simulated state.

The challenge for rapid calculation of free energies is that the cost of generating enough samples to obtain sufficiently accurate estimates can become prohibitively large if there are numerous intermediate states along the path or if the systems have long correlation times at some or all of the intermediate states. To help overcome this problem, we can select coupling pathways that have high overlap in configuration space among neighboring states along the path to reduce the amount of simulation required for a given statistical error tolerance. [84, 85, 95] The number of intermediate states required along a pathway will depend on the specific molecular transformation, the statistical tolerance for the problem, and the computational budget.

Increasing the overlap in configurational phase space along a path is equivalent to minimizing the statistical variance of free energy calculations along the pathway using TI. More precisely, the error in the free energy estimation using TI is proportional to the square root of the variance over the number of samples taken. [84, 97] The variance is an intrinsic property of the path along which the transformation is performed, independent of the number of samples, and determined by the thermal fluctuations around equilibrium at states along the pathway. [96] The variance of each point along the pathway can also be thought of as the square of the Riemannian metric used to measure thermodynamic length along that pathway. [93, 94, 96] Minimizing the thermodynamic length between states will maximize their phase space overlap, and reduce statistical error in free energy calculations. This relationship is exact for TI, but can be shown to be approximately true for other free energy methods. [84]

A common approach which produces a fairly statistically efficient alchemical path for the disappearance or appearance of molecules with Lennard-Jones site potentials

is the "soft core" potential, with the most general form [82–84]

$$U(r,\lambda) = 4\epsilon_{ij}\lambda^a \left[\left(\frac{1}{\alpha(1-\lambda)^b + (r/\sigma_{ij})^c} \right)^{12/c} - \left(\frac{1}{\alpha(1-\lambda)^b + (r/\sigma_{ij})^c} \right)^{6/c} \right] \quad (2.1)$$

where $U(r,\lambda)$ is the pairwise nonbonded potential depending on radial distance r between two atoms and the alchemical coupling variable λ. a, b, and c are positive, usually integer, constants, α is a positive free parameter that can be optimized, and ϵ_{ij} and σ_{ij} are the Lennard-Jones (LJ) parameters. Potentials of this type provide a relatively good phase space overlap between neighboring states which in turn reduces the variance. [82–84, 105] As λ goes from fully coupled to decoupled, soft core potentials cause the excluded volume region in the pair interaction to gradually decrease in energy, allowing solvent molecules to "leak" into the excluded volume of the solute, until the solute excluded volume disappears completely. The correct choice of constants can reduce the total variance summed over all intermediate states for this class of potentials. The original version of soft core potentials had $a = 1$ or $a = 4$, $b = 2$, $c = 6$, and $\alpha = 0.5$. The choice of $a = 1$, $b = 1$, $c = 6$, and $\alpha = 0.5$ is about 30% more efficient; [58, 84] we will refer to this choice as the "1-1-6" potential. A soft core potential which is near minimal over all possible pair potentials of r and λ has recently been developed with parameters $a = 1$, $b = 1$, $c = 48$ and $\alpha \approx 0.0025$, which we will refer to as the "1-1-48" potential. [84, 97]

Soft core potentials increase the difficulty of implementing free energy methods on new, highly parallelized architectures. Developing new, generalized algorithms on architectures such as GPUs or Many Integrated Core (MIC) platforms has been of large interest to the molecular simulation community. Coding generalized algorithms on these platforms is a non-trivial task and soft core potentials add complexity as configuration and alchemical information must both be passed to the platform calculating the inner force loop due to the nonlinear coupling of the two variables.

For free energy methods requiring derivatives, the derivative would also need to be calculated on the platform, further increasing difficulty in quickly writing new code. Eliminating the need to code alchemical transformation-specific information in the accelerated inner loops would reduce the time needed to implement free energy methods on new platforms, which has lagged compared to the implementation of other molecular mechanics functions on these platforms.

We can reduce the difficulty in coding free energy on new platforms and remove the free energy calculations from the inner loops if we write the potential energy as

$$U(r,\lambda) = \sum_{i=1}^{N} h_i(\lambda) u_i(r) + u_{\text{unaffected}}(r) \tag{2.2}$$

where $h_i(\lambda)$ are alchemical switching functions depending only on λ, and $u_i(r)$ are pairwise potential basis functions depending only on r. In this equation, N is the total number of basis function and alchemical switch combinations, and $u_{\text{unaffected}}$ is the portion of the potential energy of the system not dependent on the coupling parameter. For example, to insert a single solute site from a box of solvent, we could write

$$U(r,\lambda) = h_1(\lambda) \sum_{j=1}^{N_{\text{solvent atoms}}} U_{\text{LJ}}(r) + h_2(\lambda) \sum_{j=1}^{N_{\text{solvent atoms}}} U_{\text{electrostatic}}(r) + U_{\text{solvent-solvent}}(r) \tag{2.3}$$

with $h_1(\lambda) = (2\lambda)^4$ for $0 < \lambda < 1/2$ and 1 for $1/2 < \lambda < 1$, and $h_2(\lambda) = 0$ for $0 < \lambda < 1/2$ and $2\lambda - 1$ for $1/2 < \lambda < 1$. This would be equivalent for a workable, though not particularly statistically efficient pathway of first turning on the Lennard-Jones interaction with a λ^4 dependence, and then turning on the electrostatic interaction with linear dependence. We can certainly construct a more efficient path, as we will show, but this provides an example how existing approaches fit into this general formalism.

Using this method, the terms needed for different free energy methods can then be calculated outside of the inner loop. We only need to compute the potential energy of the basis functions $u_i(r)$ once for all intermediate states, with the rest of the required calculations involving manipulations of the coupling functions $h_i(\lambda)$ outside the force calculation of the inner loop. Additionally, no special code for the inner loop is required to compute $\partial\mathcal{H}/\partial\lambda$, the derivative of the Hamiltonian, as it can be calculated directly from the basis function terms with minor post-processing.

Some instances of such linear potentials have been studied previously, so the functional form itself is not entirely novel. For example, initial free energy applications often multiplied the entire potential energy of the molecule disappearing with a single linear scaling term, of the form $U(\lambda, r) = (1 - \lambda)U_{\text{initial}}(r) + \lambda U_{\text{final}}(r)$. However, these simplest linear combination potentials are generally no longer used for problems involving changing numbers of atomic sites because they lead to highly inaccurate, numerically diverging free energies. [82, 83, 85, 95, 109] This occurs because r^{-n} terms common in many intermolecular potentials create an infinite potential, or singularity, at $r = 0$ for all λ except $\lambda = 0$. The sudden jump of the potential from infinity to zero when approaching $\lambda = 0$ causes the thermodynamic expectation of $\langle\partial u/\partial\lambda\rangle$ to diverge, giving large variances and uncertainties. [95] This problem is sometimes called the "endpoint catastrophe." [85, 109] As discussed, soft core potentials avoid this problem as the singularity is replaced by a truncated core potential at $r < \sigma$ that gradually disappears as the coupling parameter reaches the decoupled state.

Many simulation packages have moved away from linear interpolation and implement some form of soft core method for free energy calculations. For example, GROMACS, [110] GROMOS, [86, 87] CHARMM, [111] and AMBER [112] all implement a variant of soft core potentials to carry out free energy calculations with an option to do simple linear interpolation. Some of the packages allow interpolation with a power such that $U(\lambda, r) = (1 - \lambda^n)U_{\text{initial}}(r) + \lambda^n U_{\text{final}}(r)$, which removes the

endpoint catastrophe for $n \geq 4$, but can still have simulation stability issues for some systems and time steps. [85] GROMOS allows defining products of an arbitrary function of λ, similar to our $h_i(\lambda)$, times the soft core equation (Eq. 2.1). However, these functions still involve the soft core functional form, leaving r and λ coupled. In all packages, the choice of "best" path is left to the user. The basis function approach we present here differs from all soft core methods in that it requires that r and λ be decoupled from each other in separate functions and it differs from traditional linear interpolation since more complex switch functions than λ^n are permitted. Despite this limitation on the basis functions, we will show that a near optimal path can be found to minimize statistical error with this basis function approach.

Recent work has tried to identify new potential formalisms that can avoid the endpoint catastrophe while still keeping the advantages of a linear basis potential, such as the form described by Buelens and Grubmüller. [88] This approach can be written as single basis function from Eq. (2.2) as $U(r, \lambda) = \lambda \sum_{\text{affected pairs}} u(r)$ but avoids the infinite potential at $r = 0$ by setting a maximum finite value for the basis function $u(r)$ and creating a polynomial switch from the LJ function to this cap. However, it is unclear what the statistical efficiency of this approach is, as it was not directly tested, and the fact that significantly more intermediates are required near $\lambda = 0$ than near $\lambda = 1$ indicates it is not a particularly statistically efficient path. [88] Additionally, the approach requires conditionals to be evaluated with every pair potential to check for the needed for caps, which could prove problematic in some highly vectorized implementations.

The goal of this work is to present a formalism for developing low-variance pathways for molecular transformations based on linear combinations of more than one coordinate basis set. This chapter combines the implementation simplicity of the linear basis potential approach and the statistical efficiency of optimized soft core potentials. We will follow this approach to propose statistically efficient linear basis pathways that

work in general cases of removing molecules from dense fluids. We can minimize the variance of families of linear basis potentials with the methods we have previously developed, [84, 97] resulting in a family of highly statistically efficient alchemical paths that also reduce the cost of re-evaluating potentials and can simplify implementing free energy code.

2.2 Theoretical

We can write the most general version of the linear basis potential as:

$$U(r,\lambda) = \mathbf{h}(\lambda)\mathbf{u}^T(r) + u_{\text{unaffected}}(r) \tag{2.4}$$

which is a vectorized version of Eq. (2.2). Here, the vector functions $\mathbf{h}(\lambda) = [h_1(\lambda), h_2(\lambda), \ldots, h_N(\lambda)]$, and $\mathbf{u}(r) = [u_1(r), u_2(r), \ldots, u_N(r)]$ replace the individual components in the previous equation. λ can vary from 0 to 1 and each of the N $h_i(\lambda)$ is a monotonic function of λ which has endpoints $h_i(0) = 0$, resulting in a fully noninteracting $u_i(r)$ term, and $h_i(1) = 1$ which results in a fully interacting $u_i(r)$ term. The only requirements we place on the $h_i(\lambda)$ that they are continuous and that $dh_i(\lambda)/d\lambda \geq 0$ i.e. it is monotonic, although we allow some of $h_i(\lambda)$ to be fixed while the others change. This formalism means that λ can be thought of as a curve through the N-dimensional alchemical space mapping the single $[0,1]$ domain to the N $h_i(\lambda)$ which span a $N \times [0,1]$ range. A key to our linear basis potential approach is that there are *multiple* $h_i(\lambda)$ and *multiple* basis functions which together can produce a broad range of functions that can be adjusted to create a highly statistically efficient pathway with minimum variance. We will occasionally leave off the explicit dependence of $h_i(\lambda)$ on λ and write h_i in some equations for simplicity.

Computing the potential energies at different values of λ becomes trivial if the basis function energies are tabulated, and can be done either in code (but outside the

inner loop) or in post-processing, as long as the energies of each basis function are stored at each configuration of interest. The derivative $\partial u/\partial\lambda$ can also be computed trivially using the basis function energies, because:

$$\frac{\partial U(r,\lambda)}{\partial\lambda} = \sum_i^N \frac{\partial U(r,\lambda)}{\partial h_i}\frac{\partial h_i}{\partial\lambda} = \sum_i^N \frac{\partial h_i}{\partial\lambda}u_i(r) = \mathbf{h}'(\lambda)\mathbf{u}^T(r) \tag{2.5}$$

where $\mathbf{h}'(\lambda) = [\partial h_1/\partial\lambda, \partial h_2/\partial\lambda, \dots]$.

There are two main choices that must be made to design a linear basis potential representation of some pathway through alchemical space.

1. Choosing the basis functions, $\mathbf{u}(r)$

2. Choosing the alchemical switching functions, $\mathbf{h}(\lambda)$

The basis functions must be chosen to match the potential function at the end states as well as avoid the endpoint catastrophe. Beyond that, there is significant freedom in the choice of basis functions. Given a set of basis functions satisfying the conditions at the endpoints, the alchemical switching functions can be chosen to minimize the variance of the path along λ. The statistical uncertainty of a simulation performed along this path is a function of both the choice of basis and the choice of switches. However, given a choice of basis, there exists a single set of switches that minimizes the variance of the path.

There is one complication we encounter while determining the alchemical switches given a set of basis function. As we will see, approaches to minimize the variance over all the switches simultaneously break down when one or more of the $h_i(\lambda)$ functions are zero while the others change. Unfortunately, many of the pathways most straightforward to implement may have this feature; for example, in Eq. (2.3), we change the Lennard-Jones and electrostatic energy terms sequentially. We will therefore separately consider the "alchemical schedule," the choice of the ranges of

λ where each $h_i(\lambda)$ is changing or constant, and treat this as a third choice we must make to determine the pathway.

To help visualize what we mean by a schedule, Fig. 2.1 shows a few sample schedules that could be taken connecting initial and final states. All schedules will by definition give the same free energy between end states, but some alchemical schedules are larger variance pathways than others, and some may give completely divergent answers. For example, turning on attractive electrostatic interactions before turning on any repulsive Lennard-Jones interactions will lead to negative infinite potential energies and crashing simulations.

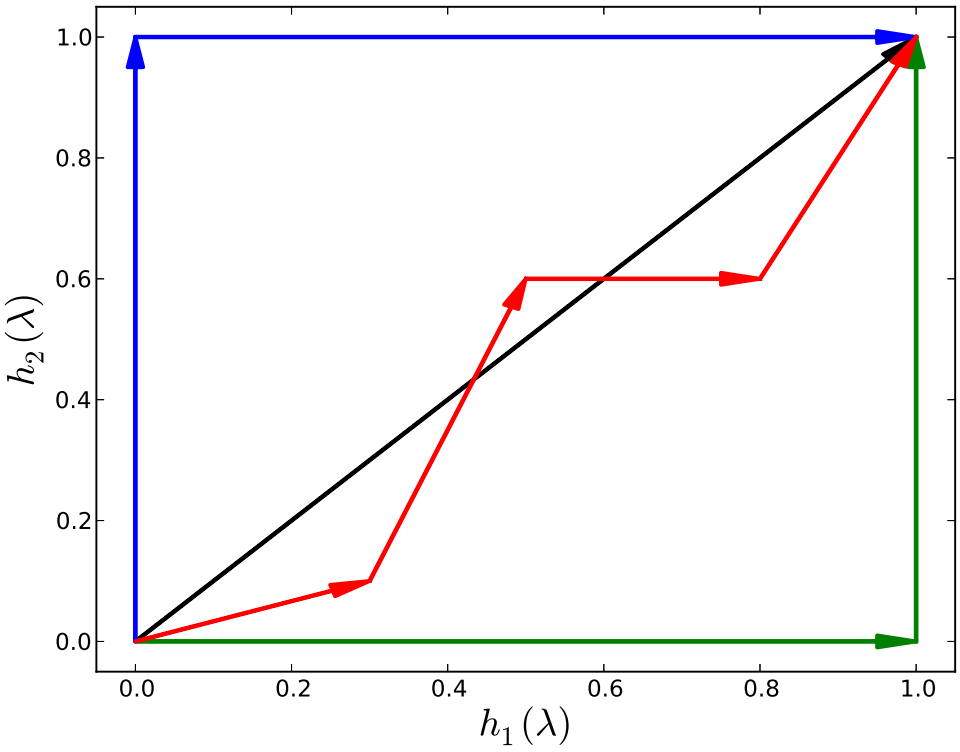


Figure 2.1: **Different alchemical schedules can be used to carry out the same free energy transformation, each with different variances**. An example of different choices of the alchemical schedule is shown for when only two $h_i(\lambda)$ are present. Both h_i's could change simultaneously (black) as with attractive and repulsive components of soft core potentials. Each h_i could be changed one at a time (blue or green) such as the common case when decoupling electrostatics before other forces. Alternately, a custom schedule (red) could be chosen to improve sampling at the corresponding states. All pathways yield the same end states and thus the same values for state functions. However, some paths may be more statistically efficient than others.

Due to the complexity in general design of a linear basis set, we will use a single simple schedule throughout this chapter and focus on the logic of the other design choices. We will leave the selection of best schedule for a specified choice of basis functions to Chapter 3. We will also focus primarily on determining statistically efficient paths for solute-solvent nonbonded interactions. Solute-solute interactions, both bonded and nonbonded, will behave differently because of the strong couplings in intramolecular degrees of freedom.

2.2.1 Fixing the Alchemical Schedule

The three main nonbonded interactions used by most pairwise force fields are repulsive, van der Waals dispersion, and electrostatic forces. In this chapter, we assume the electrostatic interactions are fully decoupled using the linear alchemical switch defined by $h_{\text{electrostatic}}(\lambda) \propto \lambda$ before any alchemical modifications of the attractive or the repulsive interactions. This linear electrostatic switch is a common choice that has been found to be quite statistically efficient and often considered best practice. [58, 85, 113, 114] The correct amount of simulation effort to devote to turning off electrostatic interactions will depend on the contribution to the statistical error of the electrostatic versus Lennard-Jones terms, which will depend on the magnitude of the partial charges. We will leave the further optimization of this choice for Chapter 3, and instead concentrate on the more challenging removal of Lennard-Jones terms. After the electrostatics are decoupled, $h_{\text{electrostatic}}(\lambda)$ will be 0 for the remainder of the transformation, e.g. we would set $h_2(\lambda) = 0$ for all λ shown in our example Eq. (2.3).

For this chapter, we also choose for both the repulsive and the attractive forces to be modified simultaneously with alchemical switches $h_{\text{repulsive}}(\lambda)$ and $h_{\text{attractive}}(\lambda)$ respectively. These switches will change together only after $h_{\text{electrostatic}} = 0$. Because the electrostatics are assumed to already be off, any solvation free energy differences presented in this chapter will be of the uncharged molecules. This allows us to directly compare to the magnitude of statistical error computed using soft core simulations. [84, 97] For simplicity, we will also rescale λ to represent only this non-electrostatic part of our schedule such that $h_{\text{repulsive}}(\lambda = 0) = h_{\text{attractive}}(\lambda = 0) = 0$ and $h_{\text{repulsive}}(\lambda = 1) = h_{\text{attractive}}(\lambda = 1) = 1$.

This particular choice of schedule and electrostatic alchemical switch is reasonable because the disappearance of Lennard-Jones terms is usually the largest contribution to the statistical uncertainty of free energy calculations. [84] Further optimizing the components of the linear combination potential for molecules with significant

electrostatic interaction, as well as examining more general choices for alchemical schedules will be addressed in a Chapter 3.

2.2.2 Selecting Accurate Basis Functions without Singularities

The singularity in the basis functions can be avoided by setting a maximum finite potential referred to here as a "cap." If the cap on the potential is large enough, then the probability of observing atoms within the capped region of the potential energy surface will be effectively zero. This capped potential method was also explored by Buelens and Grubmüller [88] who examined a linear coupling parameter, but we wish to generalize the potential in our more general linear basis potential formalism.

Using a capped repulsive potential function as one of our basis functions for a minimum variance path is very strongly suggested by inspecting the shape of the optimized soft core potential, as shown in Fig. 9 of Pham and Shirts. [84] In the 1-1-48 form of the soft core potential, at intermediate states, for $r < 0.8\sigma_{ij}$, the potential becomes almost completely flat with respect to r, looking very much like a capped potential in r with a capping distance independent of λ. This capped portion of the potential is then scaled to zero as $\lambda \to 0$.

If the capped potential is used as the end state of the simulation, then the capped potential at the fully coupled state must be high enough such that the probability of observing a pair of atoms in the capped region is statistically zero on the time scales of any simulation. We wish to quantify the deviation that would result from replacing the full repulsive potential with a cap. Because the largest contribution to the variance in introducing an atomic interaction site comes from two-particle repulsive interactions, [84] we can approximate the radial distribution function $g(r, \lambda)$

when two particles are very close by the zeroth-order approximation

$$g(r, \lambda) \approx e^{-\beta U(r,\lambda)} \tag{2.6}$$

where $\beta = 1/k_B T$ and T is the temperature. If this RDF approximation for the capped Lennard-Jones potential and the uncapped potential are sufficiently well matched, the basis functions should be statistically identical to other potentials at the fully coupled end state. We will eventually find that there are problems with using an approximate capped end state. However, we will also show that any deviations caused by the capping procedure can be addressed with minimal additional complexity.

To keep with the basis function formalism, we treat the short-range repulsive forces and the long-range attractive forces separately by applying the Weeks-Chandler-Andersen (WCA) decomposition [115] to our capped potential. The decomposition provides us two differentiable potential basis functions and is written as

$$U_{\text{wca}}(r) = u_R(r) + u_A(r) \tag{2.7}$$

where the subscripts R and A denote the repulsive and attractive terms respectively. Denoting $u_{\text{LJ}}(r)$ as the full Lennard-Jones potential, the individual terms of the WCA potential can be written as

$$\begin{aligned} u_R(r) &= u_{\text{LJ}}(r) + \epsilon_{ij} && \text{for } r < 2^{1/6}\sigma_{ij} \\ &= 0 && \text{for } r \geq 2^{1/6}\sigma_{ij} \end{aligned} \tag{2.8}$$

$$\begin{aligned} u_A(r) &= -\epsilon_{ij} && \text{for } r < 2^{1/6}\sigma_{ij} \\ &= u_{\text{LJ}}(r) && \text{for } r \geq 2^{1/6}\sigma_{ij}. \end{aligned} \tag{2.9}$$

The WCA decomposition avoids the unrealistic negative singularity from the attractive van der Waals forces brought on from capping only repulsive interactions. Setting

an arbitrary negative cap is an alternative, [88] however, this adds extra parameters which we wish to avoid for simplicity.

To make the function continuous, the repulsive potential must be capped smoothly. We choose a smooth polynomial function defined over the transition region. This polynomial should be monotonic to avoid creating artificial minima, as well as not imposing an excessive deviation in the force from the normal Lennard-Jones potential. We avoid these problems by setting a quartic polynomial to match the potential energy values and derivatives at either end of the transition region, and the second derivative of the potential at the higher r endpoint. The second derivative condition on the quartic polynomial was chosen to keep the polynomial monotonic over the transition. Cubic polynomials sometimes fail to be monotonic over the transitions we tested with different Lennard-Jones parameters. Applying these changes to the repulsive basis function gives

$$\begin{aligned} u_{R,\mathrm{cap}} &= u_{\mathrm{cap}} && \text{for } r < f_{\mathrm{cap}}\sigma_{ij} \\ &= Ar^4 + Br^3 + Cr^2 + Dr + E && \text{for } f_{\mathrm{cap}}\sigma_{ij} \le r < f_{\mathrm{switch}}\sigma_{ij} \\ &= u_{\mathrm{LJ}}(r) + \epsilon_{ij} && \text{for } f_{\mathrm{switch}}\sigma_{ij} \le r < 2^{1/6}\sigma_{ij} \\ &= 0 && \text{for } r \ge 2^{1/6}\sigma_{ij} \end{aligned} \tag{2.10}$$

where f_{cap} is the fraction of σ_{ij} over which the capping region extends and where the switch starts and f_{switch} is the fraction of σ_{ij} at which the switch ends and the normal Lennard-Jones function resumes. $u_{\mathrm{cap}} = u_R(f_{\mathrm{cap}}\sigma_{ij})$ is the capped, constant potential, the constants A-E are fit to meet the conditions at either end of the transition region, and the choice of the constants of f_{cap} and f_{switch} will be discussed in the next section. The two basis functions we use in the vector $\mathbf{u}(r)$ are thus Eq. (2.10) for the repulsive term and Eq. (2.9) for the attractive term.

The final potential energy term, with alchemical switches is then

$$u_{\mathrm{WCA}}(r,\lambda) = h_R(\lambda)u_R(r) + h_A(\lambda)u_A(r) + h_C(\lambda)u_C(r). \tag{2.11}$$

Deviation of the Capped Potential from a Fully Coupled Lennard-Jones Potential

The thermodynamic difference between the capped WCA potential and a normal Lennard-Jones potential will depend on how large the cap is. The value of the cap will depend on the choice of the transition region between f_{cap} and f_{switch} in Eq. (2.10). If the cap is high enough, i.e. small f_{cap}, then there will be no statistical difference between the capped potential and the Lennard-Jones potential, as the energy barrier will prevent pairs of atoms from being closer than $f_{\mathrm{cap}}\sigma_{ij}$. However, large caps will be less statistically efficient for decoupling, as disappearing a large barrier will have larger variances. Capped potentials with smaller barriers will be more statistically efficient to decouple, but will be thermodynamically different than the Lennard-Jones potential.

We can add a step into the alchemical schedule to account for any thermodynamic difference between the capped and the Lennard-Jones potentials. This modifies our alchemical schedule to become a three-step process in turning on the interactions between a molecule and its surroundings. First, we turn on the capped repulsive and van der Waals attractive forces. Second, we linearly transition between the capped repulsive potential and the original repulsive term of the WCA decomposition. [115] Finally, we turn on the electrostatics of the solute. This schedule still fits perfectly within the context of the linear basis function approach and the basis function representation for this additional step is

$$U_{\mathrm{capping}}(r,\lambda) = \lambda\left(u_R(r) - u_{R,\mathrm{cap}}(r)\right) + u_{R,\mathrm{cap}}(r). \tag{2.12}$$

The value of f_{cap} will determine magnitude of the cap and the statistical efficiency of this step. If f_{cap} is too large, the variance for disappearing the infinite potential will be large since the low barrier does not prevent pairs of atoms from 'leaking' into the core.

After some experimentation, we chose a transition region where $f_{\text{cap}} = 0.8$ and $f_{\text{switch}} = 0.9$ for this chapter. Examining the 1-1-48 potential with several combinations of ϵ_{ij} and σ_{ij}, we found that the transition from a scaled Lennard-Jones potential to the capped potential generally takes place in this region. One might expect naively that the ideal height would correspond to a potential energy of several times k_BT, since the probability of single pairwise interactions become negligible with only a few k_BT. However, this neglects the fact that shorter contact distances can easily be overcome by multiple positive interactions in the rest of the system. This means that to be completely thermodynamically indistinguishable from a Lennard-Jones particle, this cap must be higher, between 20 and $50k_BT$ for solute-solvent interactions, as will be detailed below.

The thermodynamic difference between the capped WCA-decomposed potential and a normal Lennard-Jones potential should be small to prevent large variances when switching between the full r^{-12} repulsive term and the capped potential. Fig. 2.2 shows the zeroth-order RDF for the Lennard-Jones potential and the capped potential. The two curves are identical for $r \geq 0.9\sigma_{ij}$ by design and only notably differ in the transition region between $0.8\sigma_{ij}$ and $0.9\sigma_{ij}$. Because this potential energy difference is small, the Boltzmann weights of the two curves will also be small and the curves will have good phase space overlap. Later in this chapter, we will examine variance of transition regions larger and smaller than the $0.8\sigma_{ij}$-$0.9\sigma_{ij}$ range, and will demonstrate problems with using both of these ranges, though we must first introduce additional tools to diagnose this. The remainder of the chapter will assume this transition region between $0.8\sigma_{ij}$ and $0.9\sigma_{ij}$ unless otherwise noted.

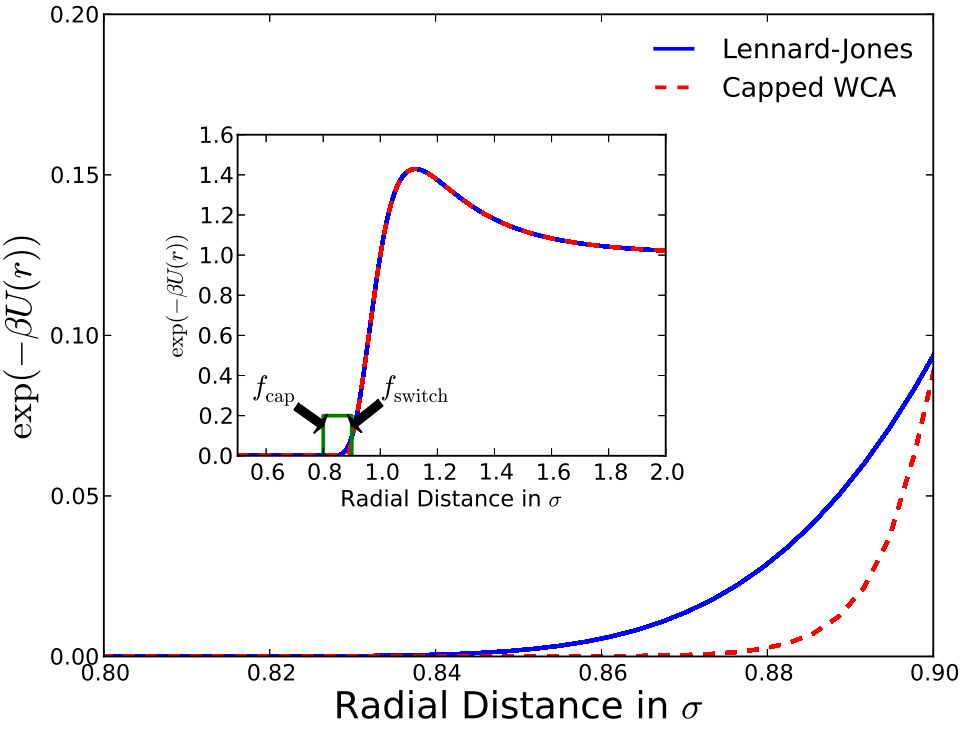


Figure 2.2: **A capped potential should be statistically similar to an uncapped potential at the fully coupled state**. The zeroth-order RDF comparison of a Lennard-Jones potential and the capped Weeks-Chandler-Andersen potential for UA Methane interacting with a TIP3P water are shown. Green box in inset highlights the enlarged region. The region of interest is $r = 0.8\sigma_{ij}$ to $r = 0.9\sigma_{ij}$ where the two potentials differ. The capped potential is large enough to be indistinguishable from a normal Lennard-Jones potential below $0.8\sigma_{ij}$ and only slightly different in the switch region. This difference has low enough Boltzmann weight that it should have minimal affect on the simulation.

2.2.3 Designing Low Variance Alchemical Switching Functions

We next must choose the alchemical switches corresponding to the repulsive and attractive basis functions, with the goal to minimize the total variance of the transformation. To minimize the variance along the path, we take the approach presented by Pham and Shirts. [84, 97] We wish to minimize the objective function of the total variance, which can first be estimated by evaluating a zeroth-order approximation to

the RDF as:

$$\mathrm{Var}\left(\Delta F\right) \approx \frac{4\pi\rho}{\beta} \int_0^1 \int_0^\infty \left(\frac{\partial U(r,\lambda)}{\partial \lambda}\right)^2 \exp(-\beta U(r,\lambda)) r^2 \,\mathrm{d}r \,\mathrm{d}\lambda \tag{2.13}$$

where ρ is the solvent number density, and F is either the Helmholtz or Gibbs free energy, depending on the ensemble sampled, with an additional PV term in the Boltzmann weight if examining the Gibbs free energy. Although this approach assumes thermodynamic integration (TI), previous research has shown that the optimal TI pathway is also (to within statistical error) the minimum variance pathway for BAR and MBAR, [84] so additional theories for those methods need not be developed.

We know that the soft core 1-1-48 potential has near minimum variance among the family of all pairwise paths. [84, 97] If the RDF of our linear basis potential pathway closely matches the RDF for 1-1-48 in both λ and r space, then the variances must also closely match as well. The zeroth-order RDF is a suitable alternate to directly matching the shape of $U(r,\lambda)$ because it is fully described by the pairwise potential and finite everywhere. This approximated RDF also allows comparison of capped potentials like our WCA-decomposed potential and soft core 1-1-48.

Eq. (2.13) can be generalized to a set of $h_i(\lambda)$ components by rewriting it as a path integral

$$\mathrm{Var}\left(\Delta F\right) \approx 4\pi\rho \int_0^1 \int_0^\infty \sum_{i,j} \left(\frac{\partial U}{\partial h_i}\frac{\partial U}{\partial h_j} \exp\left[-\beta U(r,\lambda)\right] r^2\right) \frac{\partial h_i}{\partial \lambda}\frac{\partial h_j}{\partial \lambda} \,\mathrm{d}r \,\mathrm{d}\lambda. \tag{2.14}$$

Up until now, we have used a zeroth-order approximation to the RDF. If the potential can be decomposed into basis functions and alchemical switches, the variance can be derived exactly without any assumptions about the RDF in terms of the potential

energy by starting from the equation for TI as:

$$\Delta F = \int_0^1 \sum_{i=1} \left\langle \frac{\partial \mathcal{H}}{\partial h_i} \right\rangle \frac{\partial h_i}{\partial \lambda} \, \mathrm{d}\lambda \tag{2.15}$$

$$\mathrm{Var}\left(\Delta F\right) = \int_0^1 \sum_{i,j} \mathrm{Cov}\left(\frac{\partial U}{\partial h_i}, \frac{\partial U}{\partial h_j}\right) \frac{\partial h_i}{\partial \lambda} \frac{\partial h_j}{\partial \lambda} \, \mathrm{d}\lambda. \tag{2.16}$$

Defining the $N \times N$ covariance matrix of $\mathbf{u}$ as $\mathrm{Cov}\left(\mathbf{u}, \mathbf{u}\right)$, we can rewrite the variance at a specific λ as

$$\mathrm{Var}\left(\Delta F\right)\left(\lambda\right) = \mathbf{h}'(\lambda) \cdot \mathrm{Cov}\left(\mathbf{u}, \mathbf{u}\right)\left(\lambda\right) \cdot \mathbf{h}'^{T}(\lambda). \tag{2.17}$$

The total variance over the entire transformation is then the integral:

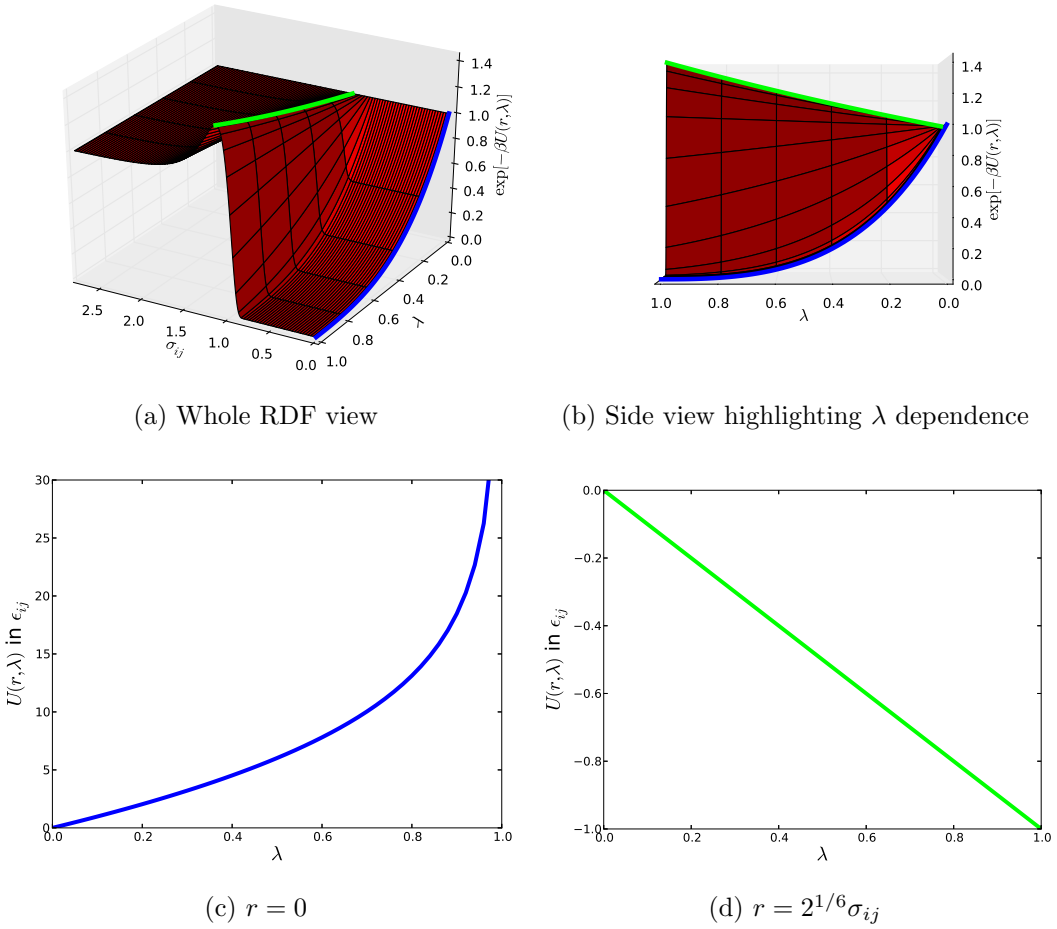
$$\mathrm{Var}\left(\Delta F\right) = \int_0^1 \mathbf{h}'(\lambda) \cdot \mathrm{Cov}\left(\mathbf{u}, \mathbf{u}\right) \cdot \mathbf{h}'^{T}(\lambda) \, \mathrm{d}\lambda. \tag{2.18}$$

These equations are derived in Appendix A.1. For the derivation of initial alchemical switches, we will focus on the zeroth-order RDF approximation, Eq. (2.14), as no simulations are required and optimizations can be carried out in seconds. Once we have an initial estimate, the zeroth-order RDF approximation will not be needed since we will have full potentials and can calculate the variance directly from Eq. (2.18).

To obtain the variance at any arbitrary point in state space given simulation data at one or multiple states, we use Eq. (2.18), with the covariance matrix estimated using MBAR [101]. The total potential and Boltzmann weights of any point in state space are straightforward to evaluate using Eq. (2.4). To compute the covariance matrix with reasonable accuracy, we must have some phase space overlap between the state of interest and the simulated state. For a single type of schedule, we can reparameterize pathways to maximize phase space overlap, as will be shown below, essentially always guaranteeing good phase space overlap as long as our initial simulations also had good

phase space overlap.

We know that the soft core 1-1-48 potential has near optimal variance in the space of all pairwise potentials for the disappearance of Lennard-Jones particles, [97] and that the two-body interactions that create excluded volume contribute the most to the variance of particle insertion. We therefore examine Eq. (2.1) and its zeroth-order RDF in Fig. 2.3 to see if we can identify basis functions and alchemical switches that are near optimal.



(a) Whole RDF view

(b) Side view highlighting λ dependence

(c) $r = 0$

(d) $r = 2^{1/6}\sigma_{ij}$

Figure 2.3: **There are two distinct dependencies on λ for soft core 1-1-48 which suggest two basis functions and switches describing low variance paths for removing or inserting Lennard-Jones interactions**. Shown are the predicted zeroth-order radial distribution function (RDF, (a) and (b)) and potential energy at constant r ((c) and (d)) along the soft core "1-1-48" pathway for alchemically appearing a united atom (UA) methane site in TIP3P water. The colored lines highlight λ dependence at $r = 2^{1/6}\sigma_{ij}$ for the green line and $r = 0$ for the blue line. In a region where $r/\sigma_{ij} < 1$, (c), the potential increases rapidly approaching infinity since soft core models converge to Lennard-Jones at $\lambda = 1$. When $r/\sigma_{ij} > 1$, (d), there is a linear dependence on λ.

Fig. 2.3 shows that the potential energy has two very distinct types of dependence on λ depending on the distance between particles. From the minimum of the potential and larger r, we observe that the potential is linear in λ, but at short range we observe a more complicated relation. This decomposition is also clear when directly examining

Eq. (2.1) and the ratio $(r/\sigma_{ij})^c$. When this ratio is greater than 1, the λ dependence is only significant in the prefactor. When r/σ_{ij} is less than 1, the denominator is controlled by λ resulting in a more complex equation. We also observe that when $r/\sigma_{ij} < 1$, the RDF and the potential of 1-1-48 are mostly insensitive to r.

The two distinct regions of λ dependence suggest two alchemical switch and basis function pairs, one for $r < \sigma_{ij}$ and another for $r > \sigma_{ij}$. The two basis functions naturally come from the WCA decomposition, as this decomposition closely matches the behavior in the two regions seen in the 1-1-48 potential. We would prefer simpler alchemical switches for these two basis functions to simplify the overall calculation. Since the long range dependence seen in Fig. 2.3 is linear in the soft core 1-1-48 potential, a natural choice for the basis derived for the attractive component of the WCA decomposition is simply $h_i(\lambda) = \lambda$.

The repulsive basis must be treated differently, as the behavior of the plateau region in the 1-1-48 potential, shown in Fig. 2.3 as a blue line, is a more complicated function of λ. Additionally, the potential goes to infinity in the limit of $\lambda \to 1$, so we cannot match this potential with a simple alchemical switch as we did the linearly scaled attractive interactions. Considering the physical constraints of the problem we can derive an approximate formula for the dependence on λ of a low variance pathway as:

$$h_i(\lambda) = \frac{K^\lambda - 1}{K - 1} \tag{2.19}$$

where K is a positive optimization constant greater than 1 (see Appendix A.2 for derivation). Additional trial and error exploration of the inequalities derived in Appendix A.2 yields a second possible alchemical switch:

$$h_i(\lambda) = \lambda \left[p + (1-p) \exp\left(- \left(\frac{1-\lambda}{s} \right)^2 \right) \right] \tag{2.20}$$

where $0 \leq p \leq 1$, s is positive, and both are parameters that can be adjusted to

minimize the variance. This switch was designed to approximate the short range 1-1-48 dependence on λ by being predominately linear in λ at small values, and transition to a mostly Gaussian shape at larger λ. The first switch, Eq. (2.19), will be referred to as "Switch A" and the second switch, Eq. (2.20), will be labeled "Switch B." Many other equations are possible, but having near optimal switches allows us to start to explore the linear basis pathways. We emphasize that in the end, we will be able to derive the optimal choice of switching function, without the RDF or any approximation to it, and these functions are simply starting points in the process.

2.3 Simulation Methods

Molecular dynamics simulations of united atom (UA) methane, UA anthracene, and all-atom (AA) 3-methylindole were carried out with YANK [116, 117] which was built on GPU accelerated OpenMM v4.1.1 [25, 39, 118, 119] in explicit TIP3P water. OpenMM allows for rapid and simple deployment of arbitrary nonbonded potentials, making it straightforward to implement arbitrary basis functions. YANK provides the capability to do alchemical solvation free energies as well as Hamiltonian replica exchange to improve sampling [120] with modifications to improve computational efficiency. [121, 122]

Molecular input files were constructed using AMBERTOOLS's LEaP [112] with OPLS-AA force field parameters for all atoms except UA anthracene, where parameters were taken from Pitera and van Gunsteren. [90] Starting molecular geometries were acquired from the supplementary material from Mobley et al. [89] and the molecules were solvated in a periodic cubic box of TIP3P water with boundaries 1.2 nm from the solute in keeping with the source's setup. This lead to 620, 874, and 961 water molecules for UA methane, UA anthracene, and AA 3-methylindole respectively.

Two additional molecules which were not part of Pham's test set [84] were also

tested. AA n-decane and a Lennard-Jones (LJ) sphere with a larger radius than Pham evaluated were tested. These two molecules allow validating the basis function approach on more asymmetric as well as larger molecular shapes. The n-decane was generated using the same steps as AA 3-methylindole. The large LJ sphere's σ_i was chosen to be 5.23 Å, the radius of buckminsterfullerene (C_{60}); ϵ_{ii} was chosen the same as UA methane, with the mass for molecular dynamics set to that of C_{60}. Mixing rules were kept the same as the other molecules in AMBERTOOLS such that $\sigma_{ij} = 0.5(\sigma_{ii} + \sigma_{jj})$ and $\epsilon_{ij} = (\epsilon_{ii}\epsilon_{jj})^{1/2}$. Because no previous data are available for variances of solvating molecules with these parameters we can only compare variances between switches and not to soft core pathways.

All molecules in all pathways except 1-1-48 decoupled their repulsive and attractive forces together with sampling at YANK's default values of $\lambda_{\text{repulsive}} = \lambda_{\text{attractive}} = \{1.0, 0.95, 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1, 0.0\}$. $(r/\sigma_{ij})^{48}$ of Eq. (2.1) exceeds the numerical precision of the OpenMM kernel which handles arbitrary potentials, so the variances were taken from earlier studies. [84] Because of this, direct comparisons of free energy difference and error in free energy between the 1-1-48 potential and the others will not be possible. However, variance can still be compared as explained below.

Simulations were carried out under isothermal-isobaric (NPT) conditions at 298 K and 1 atm. Every state was simulated for 2 ns with a 2 fs time step, samples collected every 1 ps, and Hamiltonian replica exchange between all states attempted every 1 ps. Although replica exchange is not required for these systems, there is negligible penalty for performing with it in YANK. The potential energy of every state was collected as well as the potential evaluated for every state's configuration re-evaluated to all other state's potentials; this information was analyzed by MBAR [101] for evaluating free energy and expectation values. Errors in variance were obtained using 200 bootstrap samples. [123] 3-methylindole's and n-decane's bonded hydrogens were

constrained by the SHAKE algorithm [124] and water was constrained by the SETTLE algorithm. [125] Pressure control was handled by a Monte Carlo barostat [126, 127] and temperature control through Langevin dynamics. The nonbonded cutoff was 9 Å and interactions outside this cutoff were handled by PME with an error tolerance of 5×10^{-4}. Dispersion corrections were not calculated [57, 128] however, they will only shift the free energy differences of a given molecule by a constant amount for all configurations at each value of λ, thus not affecting comparison of variance pathways.

2.4 Results

We are interested in answering five main questions while examining these linear basis function potentials:

- Is the implementation performed correctly, and do the free energy differences of the linear basis potentials from simulations converge to the same value as using the 1-1-6 pathway?
- What are the variance minimizing parameters for Switch A and Switch B derived from the zeroth-order RDF?
- Do these new linear basis potentials have comparable or lower variance than the standard 1-1-6 and optimized 1-1-48 soft core potentials?
- Can we identify the set of alchemical switches with the lowest possible variance over all sets of alchemical switches given these basis functions?
- Are these low variance switches specific to the molecules they are developed with, or are they generalizable to other molecular transformations involving removal of Lennard-Jones sites?

2.4.1 Estimated Variances and Optimal Basis Function Parameters

UA methane is the simplest place to start for estimating optimal function parameters. These parameters can be tested with other molecules to determine if these parameters are sufficiently general. The zeroth-order RDF approximation is the most accurate in the case of UA methane, since there is a single interaction site and the molecule is isotropic. A comparison of the variance and $\langle \partial u / \partial \lambda \rangle$ using the zeroth-order RDF between Switch A, Switch B, soft core 1-1-6, soft core 1-1-48, and a pure linear decoupling of the UA methane from solvent is shown in Fig. 2.4. The area under the curves gives total variance, the objective function we wish to minimize. The curves shown in Fig. 2.4 were created with the optimized parameters for our proposed switches, and a list of the parameters with the total variances is shown in Table 2.1. The alchemical switches $h_i(\lambda)$ used in generating Fig. 2.4 are shown in Fig. 2.5 and compared to a scaled 1-1-48 at $r = 0$ to show similarities in shape.

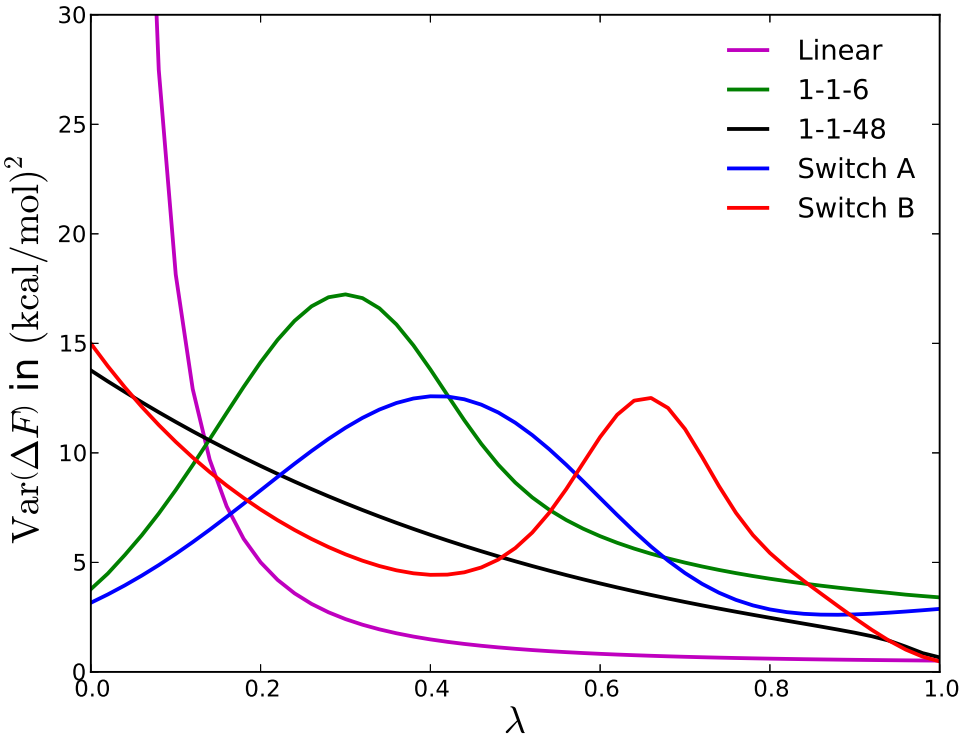


Figure 2.4: **Predicted variances using a zeroth-order RDF approximation for insertion using linear basis function paths are lower than the common soft core 1-1-6 potential, and near the minimal variance soft core 1-1-48 potential**. This optimization was carried out for inserting a united atom methane site in TIP3P water for two linear basis potentials, the two soft core potentials, and a purely linear transformation pathway. Optimized parameter(s) for Switch A are $K = 35$ and for Switch B are $p = 0.22$ and $s = 0.284$. This path only shows Lennard-Jones types interactions and assumes repulsive and attractive forces were appearing together. Total variance is found by integrating under these curves and variance for both switches falls between 1-1-6 and 1-1-48 soft core potentials.

Table 2.1: Optimal parameters and variances from zeroth-order RDF predictions for basis functions, soft core potentials, and linear alchemical transformations. Variances are in units of $(\text{kcal/mol})^2$. Soft core potential parameters follow convention in Eq. (2.1), Switch A's parameter is for Eq. (2.19), and Switch B's parameters are for Eq. (2.20). The linear transformation has no parameterization and its variance was large enough to be considered not converged (NC) by numeric integration. Transformation is for appearing a single united atom methane site in TIP3P water. Simulations of corresponding switch or soft core potentials were run with these parameters.

Alchemical Path	Parameters	Variance
Linear	—	NC
Soft Core 1-1-6	$a = 1$, $b = 1$, $c = 6$, $\alpha = 0.5$	8.52
Soft Core 1-1-48	$a = 1$, $b = 1$, $c = 48$, $\alpha = 0.0025$	5.84
Switch A	$K = 35$	7.03
Switch B	$p = 0.22$, $s = 0.284$	7.04

Fig. 2.5 suggests that nearly matching the 1-1-48 potential form at $\lambda < 0.5$ is a key factor in minimizing the variance. The predicted parameters were found to be insensitive to ϵ_{ij} and σ_{ij} for both switches, meaning the switches are likely to be applicable to a wide range of atom types.

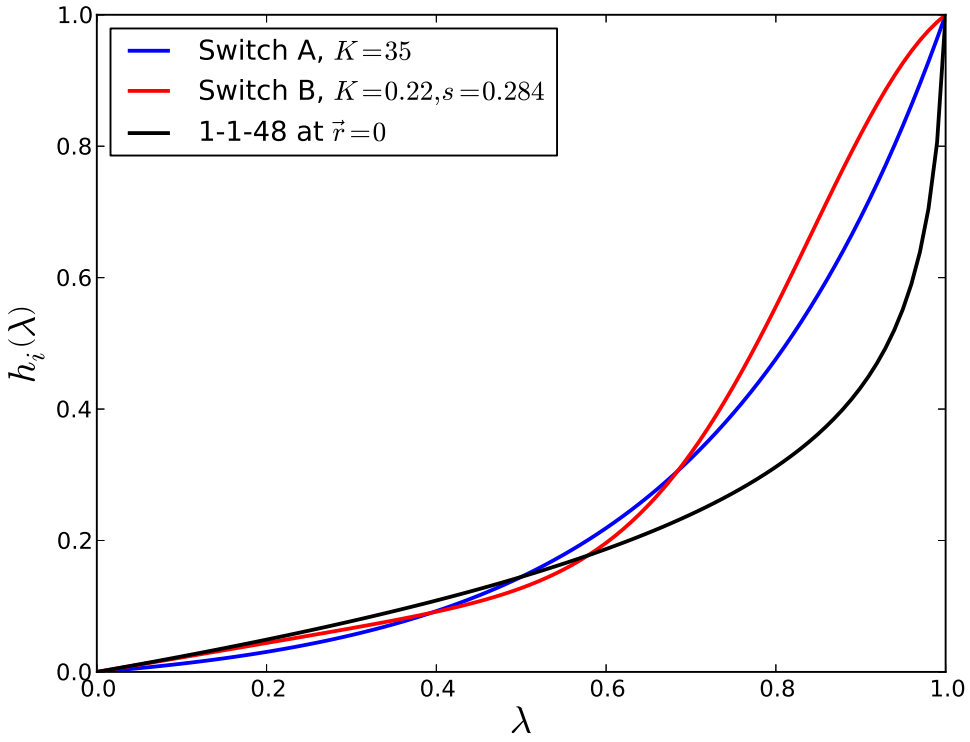


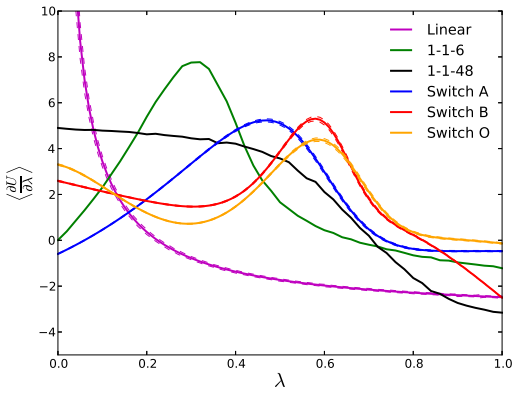
Figure 2.5: **Alchemical switches matching soft core 1-1-48 for $\lambda < 0.5$ is an important key to minimizing total variance**. Switch A and Switch B compared to the soft core 1-1-48 pathway at $r = 0$. The potential for 1-1-48 was normalized to its value at $r = 0$ and $\lambda = 0.99$ to compare the shape of 1-1-48 with the switches. All the curves have a near linear dependence at low λ and a more complex relation at larger λ. Constants for the curves came from minimizing the variance of the zeroth-order RDF approximation for a single UA methane interacting with TIP3P water on a schedule where repulsive and attractive interactions change together.

Ideally, the basis functions should have lower overall statistical variance than the soft core 1-1-6 potential to be worth using. Previous work has shown [84] that the ordering of total variances of paths estimated with the zeroth-order RDF approximation is almost always the same as the ordering of total variance in the actual simulations carried out without approximation. According to the prediction, Switch A and Switch B should have lower variances (and therefore lower statistical error in free energy calculations) than soft core 1-1-6 but not quite as low as 1-1-48.

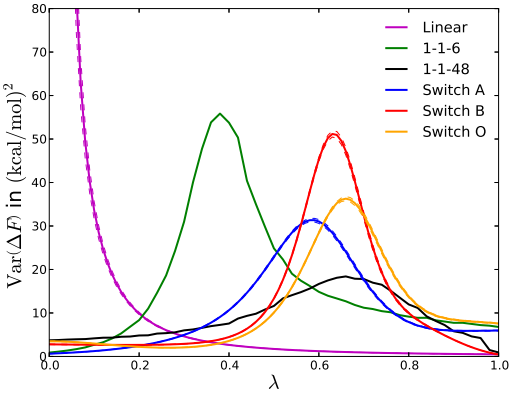
2.4.2 Variances and Free Energies from Simulations

Validating against Previously Studied Molecules

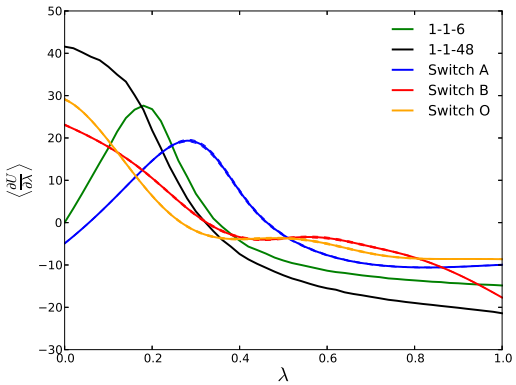
In Fig. 2.6 we show the $\langle \partial u/\partial \lambda \rangle$'s and variances as a function of λ taken from simulations of the three test molecules for the switches discussed so far ("Switch O" curves in Fig. 2.6 will be discussed later). Free energies for the three molecules are reported in Table 2.2. The soft core variances have been replotted from the figures in Pham [84] as $\partial u/\partial \lambda$ information required for soft core variance is not collected by OpenMM and cannot be evaluated by Eq. (2.18). Both Pham's variance and the basis function variance are computed directly from the simulated data. The derivation of both approaches is shown in Appendix A.1.



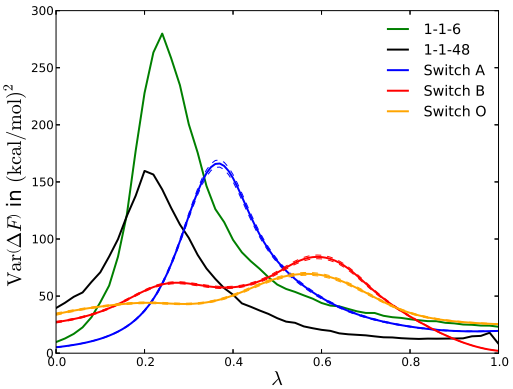
(a) $\left\langle \frac{\partial u}{\partial \lambda} \right\rangle$ UA methane



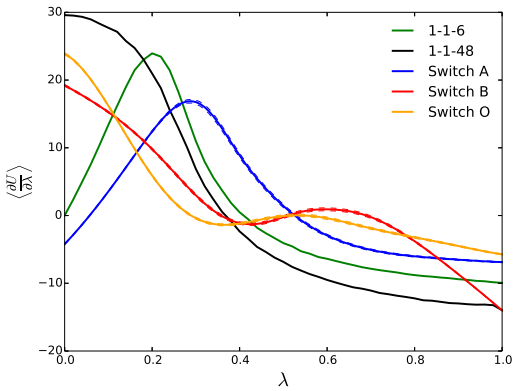
(b) Variance UA methane



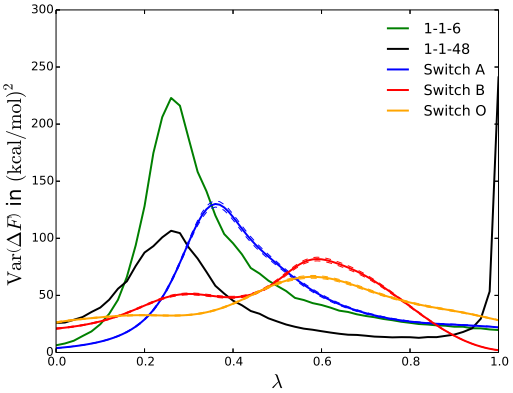
(c) $\left\langle \frac{\partial u}{\partial \lambda} \right\rangle$ UA anthracene



(d) Variance UA anthracene



(e) $\left\langle \frac{\partial u}{\partial \lambda} \right\rangle$ AA 3-methylindole



(f) Variance AA 3-methylindole

Figure 2.6: Caption on following page

Figure 2.6: **The linear basis function approach has variances lower than common soft core potentials and approach the lower limit among all pairwise potentials**. This is demonstrated for multiple molecules by the $\langle \partial u / \partial \lambda \rangle$ and variance of alchemical solvation from simulation in TIP3P water for a series of different coupling pathways for Lennard-Jones decoupling. Errors are shown with dashed lines around the curves and were estimated by 200 bootstrap samples; error is often less than thickness of the curve. Soft core data taken from Pham. [84] Switch A and Switch B have comparable or lower statistical efficiency than soft core approaches, with the optimized Switch O even lower. Reweighting using the linear basis approach allows the entire smooth curve to be calculated from only 10 to 12 sampled states. Note a strong qualitative similarity to predicted curves in Fig. 2.4.

Table 2.2: Simulated free energy of solvation and variance in TIP3P water. Free energies are in kcal/mol and variances are in $(\text{kcal/mol})^2$. Free energies for 1-1-48 path not shown as data are taken from Pham [84] who ran with a Hamiltonian including a λ dependent constant offset resulting in a different overall free energy. However, the variance of the paths can be compared and serves as a target variance. Error estimates for free energy estimated by MBAR [101] and for variance by 200 bootstrap samples. "NC" denotes that the variance did not converge. † indicates data taken from Pham. [84] Statistical error of variance for † involved more samples, but it is the value of variance, not its uncertainty that determines the statistical error of simulations. "LJ to Capped" is the free energy of changing from capped to uncapped potential and *not* included in individual free energy of switches.

Alchemical Pathway	Solvation Free Energy	Variance
UA Methane		
Linear	1.761 ± 0.114	NC
Soft Core 1-1-6	1.727 ± 0.028	$16.93 \pm 0.09^\dagger$
Soft Core 1-1-48	—	$9.03 \pm 0.09^\dagger$
Switch A	1.764 ± 0.027	10.80 ± 0.17
Switch B	1.742 ± 0.026	11.96 ± 0.19
Switch O	1.748 ± 0.026	10.97 ± 0.16
LJ to Capped	$-0.002 \pm 4 \times 10^{-4}$	$0.001 \pm 4 \times 10^{-4}$
UA Anthracene		
Soft Core 1-1-6	-0.277 ± 0.067	$78.49 \pm 0.53^\dagger$
Soft Core 1-1-48	—	$49.29 \pm 0.53^\dagger$
Switch A	-0.275 ± 0.060	53.95 ± 1.02
Switch B	-0.397 ± 0.056	48.57 ± 0.84
Switch O	-0.334 ± 0.054	45.95 ± 0.81
LJ to Capped	-0.021 ± 0.001	$0.008 \pm 8 \times 10^{-4}$
AA 3-methylindole		
Soft Core 1-1-6	1.790 ± 0.065	$63.79 \pm 0.40^\dagger$
Soft Core 1-1-48	—	$40.25 \pm 0.40^\dagger$
Switch A	1.751 ± 0.061	48.09 ± 0.99
Switch B	1.771 ± 0.055	44.66 ± 0.81
Switch O	1.930 ± 0.050	42.92 ± 0.70
LJ to Capped	-0.050 ± 0.004	0.26 ± 0.06

The calculated variance is statistically identical between our data collected with OpenMM and Pham's data previously collected with GROMACS. [38, 84, 110] Differences in the $\langle \partial u/\partial \lambda \rangle$ curves in Fig. 2.6a, c, and e are entirely due to the dispersion correction. The variances in Fig. 2.6b, d, and f are independent of differences in the applied dispersion correction in the two simulations. [57, 128]

The linear basis potentials and the linearly scaled variances were only sampled at twelve λ values. However, the properties were then estimated at 101 total λ points to construct the curves, producing smooth curves with low uncertainty. This immediately demonstrates another advantage of the linear basis potential approach: once we collect the values of the basis functions at some values of λ, we can estimate the derivatives and variance at *any* point along the curve without needing to recalculate any additional energy terms. We can make this calculation because we only require the basis function potential energies to calculate $\langle \partial u/\partial \lambda \rangle$ at any state, which can be done using Eq. (2.5) as long as the values of the basis functions at each sampled state are recorded. The Boltzmann factor of the configurations changes at each state, but depends only on the total potential at that state and can be easily calculated from the computed basis functions outside of any inner loops. MBAR [101] was used to estimate the expectation at each state reweighted to the correct ensemble. The statistical error of these expectation calculations will depend on how close in phase space our sampled states are to the states we estimate the observables at, but we will see that assuming we are comparing using a single schedule, lack of overlap is never a problem.

The linear basis function implementation gives free energies statistically consistent compared with the 1-1-6 pathway implemented in OpenMM. The ranking of the total variance of the different pathways (linear, 1-1-6, 1-1-48, Switch A and Switch B) using the estimated zeroth-order RDF predictions compared to actual simulations are indeed the same for UA methane, though switches A and B are reversed for UA anthracene and AA 3-methylindole, larger molecules for which the approximation to the radial

distribution function approach starts to break down. The zeroth-order RDF thus aids the design process of low variance initial optimal switches which can later be refined with actual simulation data. Importantly, when performing the actual simulations, the proposed linear basis potential approaches still have lower statistical error and variance than 1-1-6, though not as low as 1-1-48 as predicted by the zeroth-order RDF. The more precise results using simulation confirm that these alchemical switches can be just as statistically efficient as soft core methods.

The purely linear scaling path for the Lennard-Jones terms ($h(\lambda) = \lambda$) has large, unconverged variances and the largest relative free energy error, as anticipated. The linear transformation was simulated only for UA methane to validate the zeroth-order RDF prediction. Since the linear transformation will virtually always have higher errors and variances for appearing atomic sites in dense fluid, [82, 83, 85] it was not considered for the other molecules.

Validating against Large and Asymmetric Molecules

The simulated variances and free energies for n-decane and the large LJ sphere are shown in Table 2.3. Since no previous data are available for these molecules, the free energies and variances are shown for the basis function approaches, but only the free energy is shown for the soft core 1-1-6 pathway because the full $\langle \partial u / \partial \lambda \rangle$ curves cannot be generated.

Table 2.3: Simulated free energy of solvation and variance for n-decane and a large Lennard-Jones (LJ) sphere in TIP3P water to validate the approach in the cases of a long, non-hydrogen bonding, nonrigid molecule and a very large molecule. Free energies are in kcal/mol and variances are in $(\text{kcal/mol})^2$. Error in variance was found by 200 bootstrap samples. The large LJ sphere was sampled with three extra intermediate states and every 4 ps instead of every 1 ps due to long correlation times at intermediate λ; ‡ was *not* sampled at the three extra states to show how lower variance can reduce required samples to achieve a target statistical precision. — indicates that no data are available. "LJ to Capped" is the free energy of changing from capped to uncapped potential and *not* included in individual free energy of switches.

Alchemical Pathway	Solvation Free Energy	Variance
AA n-decane		
Soft Core 1-1-6	4.409 ± 0.072	—
Switch A	4.473 ± 0.068	68.56 ± 1.35
Switch B	4.368 ± 0.069	63.84 ± 1.26
Switch O	4.435 ± 0.060	60.94 ± 1.14
LJ to Capped	-0.110 ± 0.005	0.560 ± 0.09
Large LJ Sphere		
Soft Core 1-1-6	16.127 ± 0.116	—
Switch A	15.935 ± 0.103	65.70 ± 1.56
Switch B	16.300 ± 0.102	64.53 ± 1.51
Switch O	15.956 ± 0.089	59.80 ± 1.41
Re-optimized Switch O	$16.043 \pm 0.090^{\ddagger}$	$45.74 \pm 1.40^{\ddagger}$
LJ to Capped	-0.006 ± 0.001	$0.007 \pm 3 \times 10^{-4}$

The large LJ sphere was sampled with additional states at $\lambda = \{0.55, 0.45, 0.35\}$ due to large changes in the variance in that region and with samples taken every 4 ps instead of every 1 ps because correlation times were found to be greater than the original sampling rate of 1 ps. Because we are sampling less frequently to obtain uncorrelated samples, we have larger statistical error in the results, but all the free energies are statistically consistent.

How Significantly Do Capped Potentials Affect the Thermodynamics of the End States?

We must examine whether we can use our fully coupled capped end states with transition window between $0.8\sigma_{ij}$ to $0.9\sigma_{ij}$ as the end state of the transformation, or if there are remaining errors. For some molecules, the free energy difference is negligible, less than 0.01 kcal/mol, but for other larger molecules, it is as large as 0.1 kcal/mol. For improved robustness, we thus also recommend including a last step of changing from the capped potential to the full repulsive potential.

Calculating the free energy difference between our initial choice of the capped potential and uncapped potential is statistically efficient. The variance and free energy of turning on the cap are reported in Table 2.2 and Table 2.3 in the "LJ to Capped" entry with a switching region of $0.8\sigma_{ij}$ to $0.9\sigma_{ij}$. Because this transition window is defined in terms of σ_{ij}, the width and height of the cap will depend on the molecular parameters. We thus must compare transition windows by examining the range of cap magnitudes. Interactions with small atoms like hydrogen will have small caps while larger atoms such as carbon and oxygen will have larger caps based on ϵ_{ij}. This transition window provided caps between $3.5k_BT$ and $8.8k_BT$ for our systems with an average of $6.2k_BT$. We find that the variance of this final transformation step contributes a negligible amount to the total uncertainty over all molecules, as it is at least two orders of magnitude smaller than the insertion. The most efficient pathway guaranteeing correct thermodynamics involves a basis function with a capped potential with switch between $0.8\sigma_{ij}$ and $0.9\sigma_{ij}$ and then changing linearly to the full Lennard-Jones potential.

Entirely eliminating the extra step of switching from a full potential to a capped function would require a repulsive basis potential with a larger cap, which requires a transition starting at $r < 0.8\sigma_{ij}$. For results that are truly independent of the cap, we would like the free energy difference between the capped potential and uncapped

potential at least one order of magnitude less than the statistical error of the full solvation free energy. The higher value the cap, the closer the capped WCA potential comes to the unmodified potential, preventing solute atoms from "leaking" into the capped region of the potential. For the molecules studied here, the transition region which provided a statistically identical free energy was between $0.70\sigma_{ij}$ and $0.75\sigma_{ij}$. This provided a cap between 20 and $50k_BT$ with an average of $36.7k_BT$ over all atom types for our systems.

There is a moderate loss in statistical efficiency when using a harder cap over the entire transformation. Simulating with parameters for Switch A of $K = 165.5$ and Switch B of $p = 0.064$, $s = 0.31$ which have been reoptimized for this harder cap, we found that using this cap increased statistical uncertainty in free energy estimates by 3% for UA methane, but up to 41% for the large LJ sphere. Essentially, disappearing an average cap of $\approx 35k_BT$ is too similar to a linear decoupling of an uncapped Lennard-Jones potential.

If harder caps than our initial guess yield larger variance, could we use softer caps than our initial guess as long as the last step coupling to the full potential is retained? Softer caps here are defined by a transition region that starts at $r > 0.8\sigma_{ij}$. However, we find that using a smaller potential energy cap can introduce large errors as the cap is insufficiently repulsive. Fig. 2.7 shows several choices of soft transition regions for 3-methylindole. Very large variances near $\lambda = 0$ start to appear with a very small adjustment to the window spacing, in many cases becoming a significant percentage of the total statistical uncertainty. The average magnitude of the cap at $0.85\sigma_{ij}$ is $2.5k_BT$ and is low enough to have very little phase space overlap with a fully coupled Lennard-Jones potential, which causes the large increase in variance. Softer caps should therefore be avoided as such low caps can cause diverging variances, making any optimization of switch severely error prone. We would need an *additional* capping basis function to bridge this phase space uncertainty, which adds far too much

complexity given that we are already close to the provably minimum variance pathway for most molecules without this additional basis function.

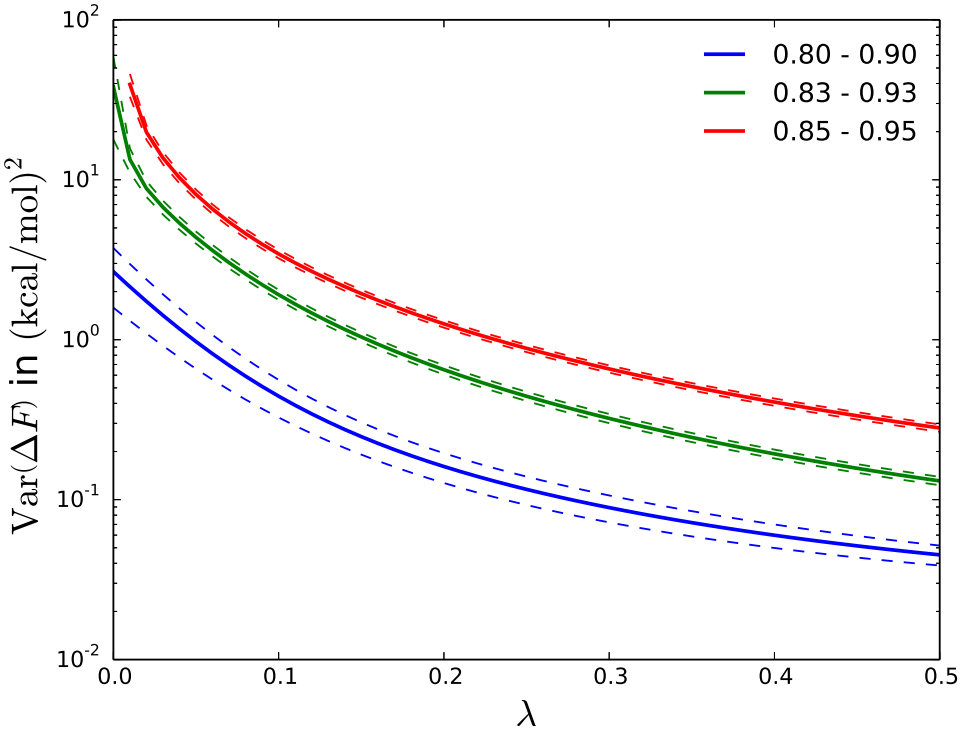


Figure 2.7: **Variances and uncertainty are large if the cap on a potential function is too soft**. This is shown here with the variance of linearly transforming from capped to uncapped potential with different transition regions for 3-methylindole. Curves labeled based on fraction of σ_{ij} of the transition region in a capped WCA potential. The variance is shown on a logarithmic scale to better visualize the rapid increase in variance from small changes in the transition region. The variance of the transition region between "0.85-0.95" (red) did not converge for $\lambda < 0.01$ and is truncated. Removing the singularity at $\lambda = 0$ results in more statistical error as the cap becomes softer. Errors estimated by 200 bootstrap samples.

We conclude that the error introduced from disappearing the positive singularity from the fully coupled state must be balanced between bringing the infinite potential to a capped one, and decoupling the capped potential from the system. We find that this balance is achieved with a transition region of $0.8\sigma_{ij}$ to $0.9\sigma_{ij}$ with the mean cap of $6.2k_BT$ with respect to atomic pairs in the systems. This choice of capped potential we use for the reminder of the chapters.

2.4.3 Variances and $\langle \partial u / \partial \lambda \rangle$ Predicted over Alternate Paths from a Single Simulation

It is possible to predict the expectations along any pathway, including the total statistical variance, with any arbitrary set of switches, $\mathbf{g}(\lambda)$, given only the data from a single simulation with a known set of monotonic switches, $\mathbf{h}(\lambda)$. A single simulation can therefore be used to find the optimal alchemical switch over the space of all possible $\mathbf{h}(\lambda)$ functions using data from only a single initial simulation. In fact, we did not necessarily need to identify two putative low variance switching functions. Once we performed simulations with a single linear basis potential, we can recalculate all observables of interest with any other alchemical switching function using the same basis functions in post-processing. If the simulated and predicted simulations involve different alchemical schedules, then our ability to predict the variance of the proposed pathway will depend on the phase space overlap. If the trial simulation involved the same alchemical schedule as the initial simulation, however, then we can always guarantee that we will have good sampling for the trial simulation if the initial simulation also had good sampling. This optimization is possible because all of the thermodynamic information for any set of switches is contained in every other set of switches. We examine two example switches in Fig. 2.8 to demonstrate this logic.

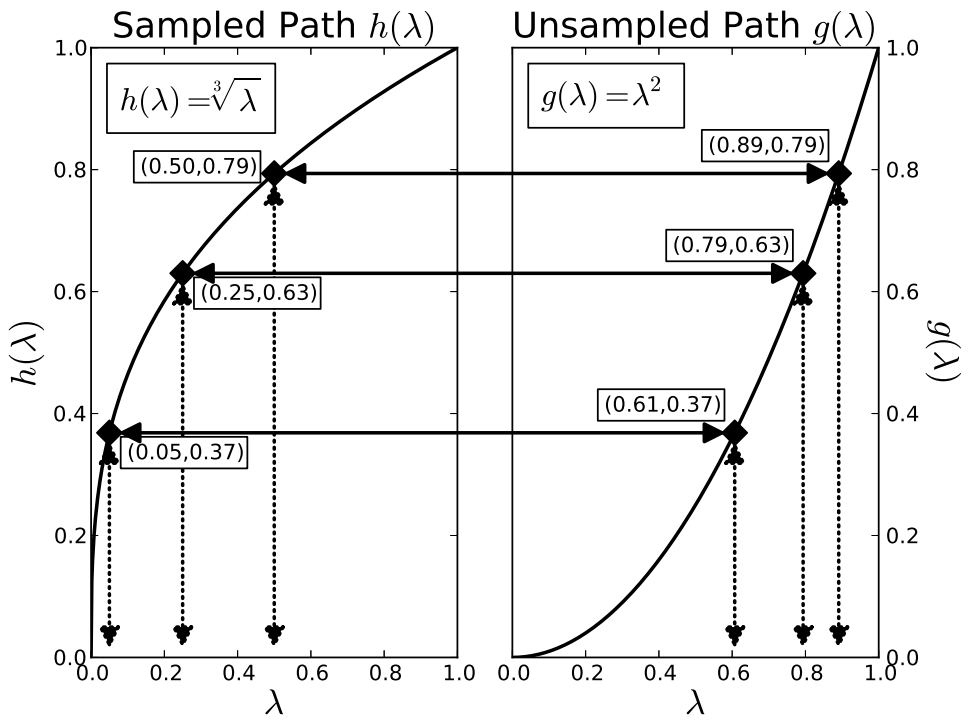


Figure 2.8: **Thermodynamic information about a pathway using an arbitrary alchemical switch is contained within any sampled switch**. An example of how to map from a sampled pathway (left) to an unsampled pathway (right) is shown. Monotonic switches both sample the $[0, 1]$ range and scale the potential energy over the $\lambda = [0, 1]$ domain. As an example, any thermodynamic property, X, of the unsampled path, X_g at the state $\lambda = 0.61$ can be found by evaluating the sampled pathway's X_h at its $\lambda = 0.05$ state; similarly, $X_g\left(g(\lambda = 0.79)\right) = X_h\left(h(\lambda = 0.25)\right)$ and $X_g\left(g(\lambda = 0.89)\right) = X_h\left(h(\lambda = 0.50)\right)$. This mapping is not affected by properties which require derivatives since the derivative with respect to λ do not participate in evaluating expectation values such as the variance. The explicit mapping here is $h = g^6$, but a similar invertible mapping exists for other paths even if not analytically calculable.

Consider a single basis function simulated with switch $h(\lambda_h)$ at a set of λ values λ_h.; also consider an unsampled switch $g(\lambda_g)$ with a different set of λ values λ_g. Because the basis function is independent of the alchemical switch, the potential of a fixed configuration is identical when $h(\lambda_h) = g(\lambda_g)$ as seen in the tie lines of Fig. 2.8. Generalizing the equality to multiple basis functions and alchemical switches, potentials will be the same for fixed basis functions when $\mathbf{h}(\lambda_h) = \mathbf{g}(\lambda_g)$.

Examining Eq. (2.17), we must have a map of the $\mathrm{Cov}\left(\mathbf{u}, \mathbf{u}\right)$ of the basis potentials

sampled from $\mathbf{h}$ to the covariance computed at the unsampled states $\mathbf{g}$. Expectations and covariances will be dependent explicitly on $\mathbf{h}$ and only implicitly on λ, so the covariance can be denoted as a function of $\mathbf{h}$ as $\mathrm{Cov}\left(\mathbf{u},\mathbf{u}\right)\left(\mathbf{h}\right)$. The shared domain of $\mathbf{h}$ and $\mathbf{g}$ means

$$\mathrm{Cov}\left(\mathbf{u},\mathbf{u}\right)\left(\mathbf{h}\right) = \mathrm{Cov}\left(\mathbf{u},\mathbf{u}\right)\left(\mathbf{g}\right) \quad \text{if } \mathbf{h} = \mathbf{g} \tag{2.21}$$

which is the required map. We extended this map to be explicitly defined by λ_h and λ_g since sampling is generally done by explicit statement of λ_h rather than $\mathbf{h}$. The inverse of the components of $\mathbf{h}$ must be evaluated to determine the λ_h corresponding to the state defined by the arbitrary λ_g. This inverse is *not* a matrix inverse, but a component-wise functional inverse, and will be denoted as $\mathbf{h}^{-1}(\lambda) = \left[h_1^{-1}(\lambda), h_2^{-1}(\lambda), \ldots, h_n^{-1}(\lambda)\right]$; we will assume that $\mathbf{h}^{-1}(\cdot)$ is a complete set of λ instead of writing $\lambda_h = \mathbf{h}^{-1}(\cdot)$ to conserve space in the equations. We can now write the variance of the unsampled path as

$$\begin{aligned}\mathrm{Var}\left(\Delta F\right)\left(\lambda_g\right) &= \mathbf{g}'(\lambda_g) \cdot \mathrm{Cov}\left(\mathbf{u},\mathbf{u}\right)\left(\lambda_g\right) \cdot \mathbf{g}'^{T}(\lambda_g) \\ &= \mathbf{g}'(\lambda_g) \cdot \mathrm{Cov}\left(\mathbf{u},\mathbf{u}\right)\left(\mathbf{h}^{-1}\left(\mathbf{g}\left(\lambda_g\right)\right)\right) \cdot \mathbf{g}'^{T}(\lambda_g).\end{aligned} \tag{2.22}$$

where $\mathrm{Cov}\left(\mathbf{u},\mathbf{u}\right)\left(\lambda_g\right)$ is the covariance matrix of the unsampled path, and $\mathrm{Cov}\left(\mathbf{u},\mathbf{u}\right)\left(\mathbf{h}^{-1}(\cdot)\right)$ is the covariance matrix of the sampled path.

Eq. (2.22) is an important result of this chapter as an optimization routine can be written around it that can run entirely in post-processing and only needs a single simulation's worth of data, not a series of iterative simulations with changing parameters.

A consequence of the linear combination is that the covariance is only explicitly dependent on $\mathbf{h}(\lambda_h)$ and not on $\mathbf{h}'(\lambda_h)$. If the covariance depended explicitly on $\mathbf{h}'(\lambda_h)$ instead, then no map could be made since the domain of $\mathbf{h}'(\lambda_h)$ is not necessarily that of $\mathbf{g}'(\lambda_g)$ and no one-to-one map may exist. $\langle \partial u/\partial \lambda \rangle$ along the alchemical path

defined by the set of switches $\mathbf{g}$ can be predicted from Eq. (2.5) to get

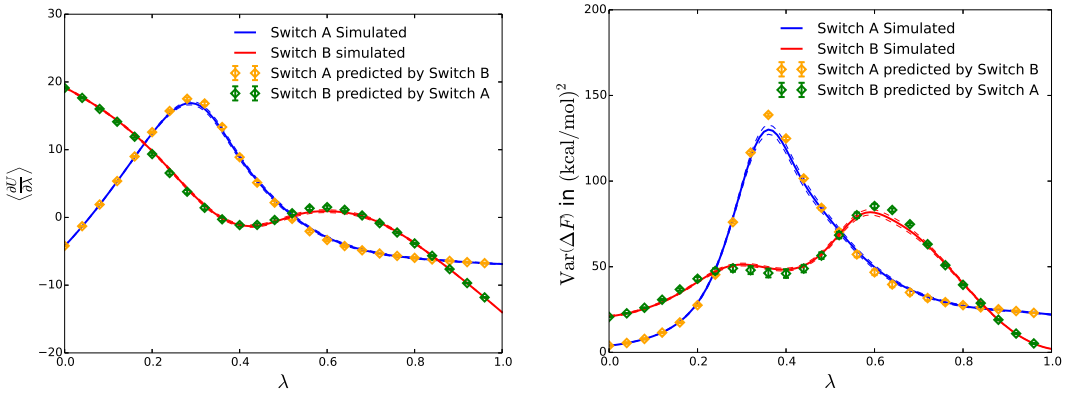
$$\left\langle \frac{\mathrm{d}U}{\mathrm{d}\lambda} \right\rangle = \mathbf{g}'_\lambda \cdot \left\langle \mathbf{u}^T \right\rangle \left(\mathbf{h}^{-1} \left(\mathbf{g} \left(\lambda_g \right) \right) \right). \tag{2.23}$$

Predicting the variance of unsampled alchemical switches has some complications. The first inherent problem with this prediction is that the λ_h found by the inverse may not be exactly any of the sampled λ_h. However, the expectation value of the potential energy at this unsampled λ_h can easily be estimated as was done to estimate the intermediate λ points of Fig. 2.6. Assuming that there was reasonable overlap between the sampled λ_h states, then any interpolated values between these will also have good overlap. The second limitation is that the basis functions in $\mathbf{u}(r)$ must be known. One simulation must therefore be run for each unique molecular system before predictions can be made about other alchemical switches; we cannot make exact estimates from simulations of a different molecule. This prediction method clearly takes advantage of the properties of linear basis potentials, meaning variances from a soft core potential like 1-1-48 at arbitrary points cannot be predicted by this method. Soft core potentials instead require separate evaluations of potential energies or derivatives of the potential at each individual configuration sampled.

The predicted covariance matrix will depend on each alchemical switch described by the alchemical path. The total potential affects the Boltzmann weight, which affects the covariance of predicted path. Each switch must be evaluated at the predicted states, even if it is not being varied. Fixing the alchemical schedule will help determine how unmodified switches are changing with respect to the switch being predicted.

As a demonstration of the power of this approach, we can see that both the variances and $\langle \partial u / \partial \lambda \rangle$ of Switch A can be predicted with high accuracy using data from the Switch B simulation and vice versa along the entire range of λ, shown in Fig. 2.9. Although Switch B does not have an analytical inverse, it is monotonic on

the domain so its inverse was found numerically. Clearly, all deviations between the predictions and the validation simulations are very small and are within statistical noise.



(a) $\left\langle \frac{\partial u}{\partial \lambda} \right\rangle$ (b) Variance

Figure 2.9: $\langle \partial u / \partial \lambda \rangle$ **and variance of an alchemical switch can be accurately predicted from data collected exclusively from another switch**. In this case, the properties of Switch A and Switch B for solvating 3-methylindole are estimated from data taken using the other switch. Error bars for simulation shown as dashed lines around the solid, error for prediction shown as capped vertical bars from the diamond; error is often less than thickness of the curve or diamond. The figures show that variance and $\langle \partial u / \partial \lambda \rangle$ can be predicted with extremely high accuracy from only one set of simulation data, implying that an optimal switch can easily be designed in post-processing. Only a limited number of prediction values are shown, but they can be generated at any intermediate, just as simulated values can be.

We can perform simple optimizations on the parameters of Switch A and Switch B as an intermediate step in the process of optimizing potentials over all possible alchemical switching functions. We found that the lowest variance of "Switch A form" potentials is obtained with $K = 10.8$ when predicting with data from Switch A's simulation, which was performed with the optimized value from the zero-RDF approximation, $K \approx 35$. When attempting to predict the minimum variance parameters for Switch A using a simulation performed with Switch B, we obtain minimum variance for Switch A with $K \approx 10.9$, validating the optimization of one transformation pathway using data collected from another pathway.

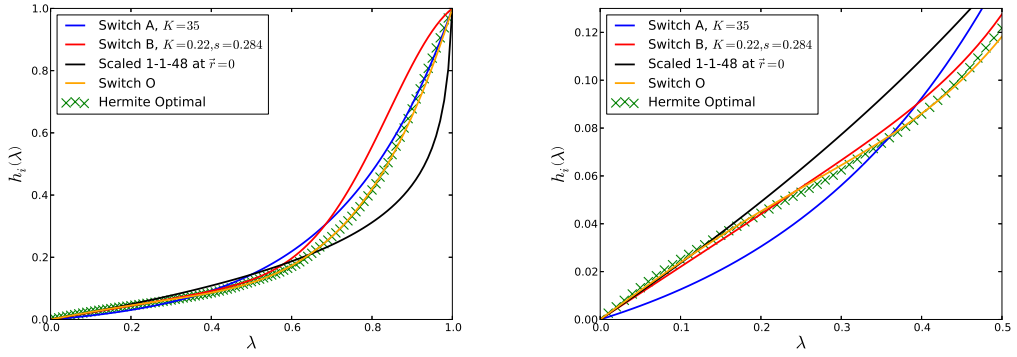
Selecting the Optimal Alchemical Switch

We next examine the case of the lowest variance set of alchemical switch functions **h** for a given set of basis functions. Such functions must satisfy $h_i(0) = 0$, $h_i(1) = 1$, and be monotonic, but may be of arbitrary mathematically complexity, rather than a pre-defined form like Eq. (2.19) or Eq. (2.20). For the purpose of optimization, we represent the family of possible switches with a monotonic spline.

An optimal $h_i(\lambda_i)$ for the repulsive basis function was created with a constrained optimization using the BY linear approximation (COBYLA) [129] routine to enforce monotonic spline knots using SciPy's [130] optimization module. Monotonic, cubic Hermite splines [131, 132] were generated to enforce the monotonic interpolating splines between these knots. After this optimization, we then approximate the splined curve with a single best fit polynomial for implementation simplicity. We select the lowest order polynomial possible, provided that the variance between the spline and the polynomial differed by less than 0.5% and was monotonic on the $[0, 1]$ domain. We are able to accurately fit to our splined minimum variance switch with a quartic polynomial of the form

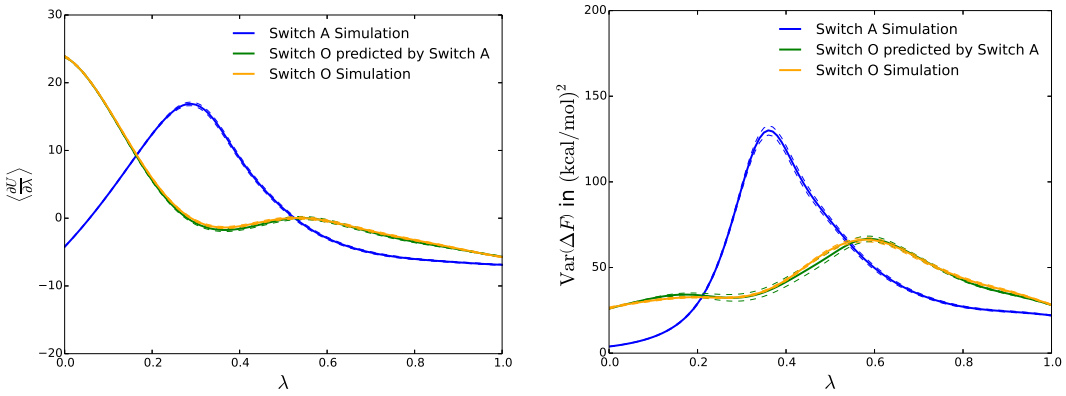
$$h_i(\lambda) = A\lambda^4 + B\lambda^3 + C\lambda^2 + (1 - A - B - C)\lambda \tag{2.24}$$

with only three fitting terms because we enforce the conditions $h_i(0) = 0$ and $h_i(1) = 1$. The parameters for this optimized switch function, which we will call "Switch O," are $A = 1.62$, $B = -0.889$, and $C = 0.0255$. For comparison, all three switches (A, B, and O) and the optimized Hermite spline are shown in Fig. 2.10. These Switch O parameters were found by taking the data from 3-methylindole's simulated Switch A with $K = 35$ and optimizing to find the minimum variance switch, which is labeled in Fig. 2.11 as "Switch O predicted by Switch A."



(a) Across entire λ range (b) Zoomed view of small λ range

Figure 2.10: **The largest impact to the overall variance is determined by the slope of the alchemical switch when** $\lambda < 0.5$. The alchemical switching functions and the optimal Hermite spline compared to soft core 1-1-48 (normalized by the potential at $r = 0$ and $\lambda = 0.99$) are shown. Zoomed in view in (b) highlights how the curves differ at small λ. This optimized switch is the best possible given the set of basis functions and fixed alchemical schedule. Different schedules with other basis functions would be required to lower the variance further.



(a) $\left\langle \frac{\partial u}{\partial \lambda} \right\rangle$ (b) Variance

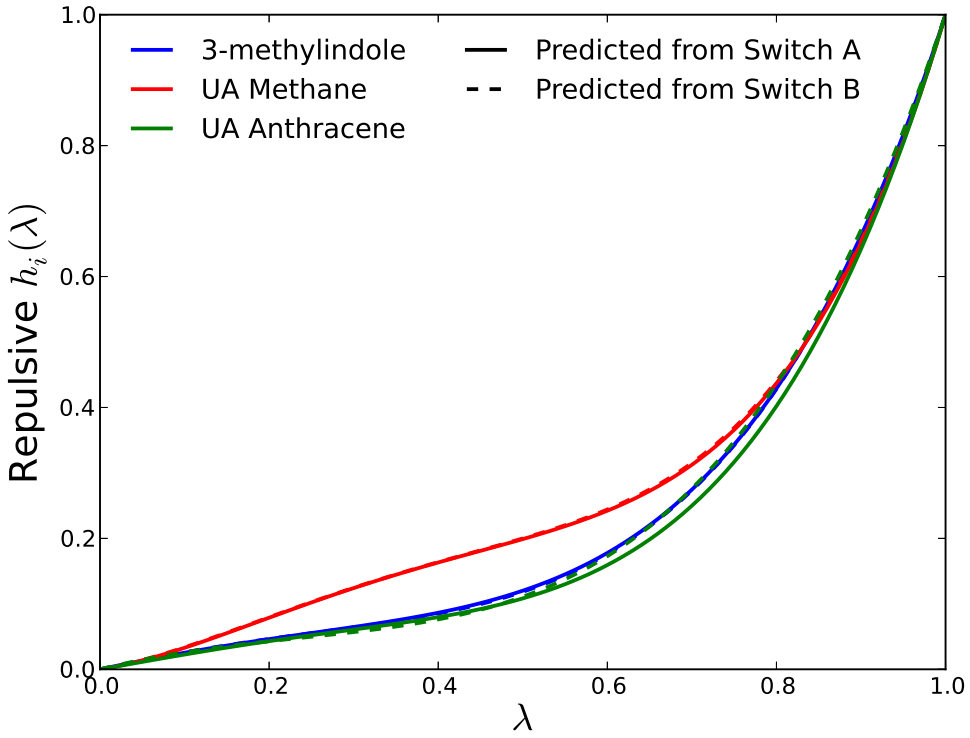
Figure 2.11: **The thermodynamic properties of the optimal switch can be predicted with high accuracy as seen with the minimized variance switch, Switch O, for 3-methylindole compared to Switch A**. Switch O was fit to a constrained quartic polynomial and found to have lower variance than other simulated alchemical switches. Error for each curve was found by 200 bootstrap samples and shown as dashed lines around the solid curve; error is often less than thickness of the curve. The predicted and simulated results are on top of each other, supporting the theory that an optimal switch can be developed from a single simulation.

Results from this predicted Switch O (using the Switch A simulation) are in excellent agreement compared to observables computed from simulations performed using this Switch O with 3-methylindole, labeled "Switch O Simulation" in Fig. 2.11. The variance of the prediction is 42.53 ± 1.20 $(\text{kcal/mol})^2$ and the simulated variance of 42.92 ± 0.70 $(\text{kcal/mol})^2$. The free energy of solvation for this transformation using the optimized switch at twelve intermediate points is 1.930 ± 0.050 kcal/mol, which has the lowest statistical uncertainty of all the reported methods in the table, demonstrating it is indeed an optimal path. Variances and $\langle \partial u / \partial \lambda \rangle$ for Switch O are plotted in Fig. 2.6 and all free energies are shown in Table 2.2.

We can probe the shape of this alchemical pathway to understand more about which regions contribute most to the variance. We find that the region of $0 \leq \lambda \leq 0.5$ shown in Fig. 2.10b, specifically the slope of the function in this region, has the largest impact on total variance. If the slope in this region is too large, then the variance will be significantly increased at that point in the path. If the slope is too low in this region, then a large slope is needed at $\lambda \geq 0.5$ to reach the $h_i(1) = 1$ condition, also resulting in a large overall variance. We find that the exact polynomial constants found from Eq. (2.24) are somewhat sensitive to the data set used to perform the optimization, but the shape of the switch itself is much more robust.

We can also test the robustness of the optimized alchemical switch with respect to change of molecule. Fig. 2.12 shows the polynomial fits to the optimized curves found from data generated with Switch A and Switch B for all molecules in this chapter. 3-methylindole and UA anthracene converge to approximately the same optimized curve with data sets from both Switch A and B. However, UA methane converges to a slightly different curve. The polynomial fit parameters shown in Fig. 2.12 vary considerably with data set and initial guess points of the optimization, however as can be seen in the figure, the curves remain essentially the same for 3-methylindole and UA anthracene. The optimization also converges to nearly the same curve for the same

molecule, regardless of starting point. The RMSD between Switch A and Switch B for UA methane, UA anthracene, and 3-methylindole in Fig. 2.12 is 0.00097, 0.017, and 0.0055 respectively, and can be qualitatively seen in the overlapping dashed and solid lines for each molecule. Although the optimal alchemical switch for UA methane is different than the switches for the other molecules, the variance-optimized curves derived for the other molecules still give free energies with lower variance than either Switch A or B for UA methane as seen in Table 2.2. Additionally, we note in Fig. 2.6 that all switches behave more robustly than the 1-1-48 switch for 3-methylindole, as there is no large change in the variance near $\lambda = 1$. Keeping the variance consistent across the entire range of molecular shapes is useful for purposes such as Hamiltonian replica exchange that strive to keep the exchange rate roughly constant across the transformation. [94] Indeed, in Table 2.2, we see that for UA anthracene, the total variance is slightly less than the 1-1-48 curve, which is possible since the 1-1-48 curve is minimal variance only for UA methane.



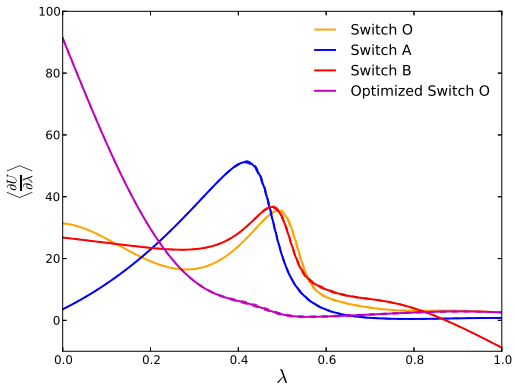
(a) Comparison of optimized alchemical switch found from different molecules and starting paths

Molecule	Predicting Switch	Coef. A	Coef. B	Coef. C	RMSD
UA Methane	Switch A	3.43	−4.35	1.72	0.0503
	Switch B	3.34	−4.19	1.64	0.0502
UA Anthracene	Switch A	1.86	−1.21	0.128	0.0107
	Switch B	0.796	0.562	−0.679	0.0092
AA 3-Methylindole	Switch A	1.62	−0.889	−0.0255	0.0000
	Switch B	1.28	−0.298	−0.275	0.0055

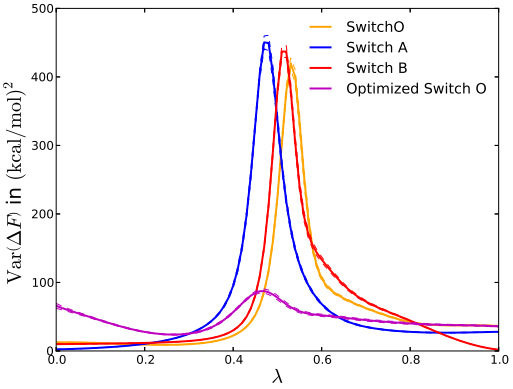
(b) Switch O coefficients and deviation from simulated Switch O for each optimization

Figure 2.12: **The optimal switch's shape for different molecules is robust, even when the polynomial coefficients are not**. A comparison of optimal fitting curves found starting from Switch A and Switch B for several test molecules. The root mean square deviation (RMSD) in dimensionless units is between each switch and the optimized Switch O generated from the parameters from 3-methylindole and Switch A, over 101 uniformly distributed points on each curve. Switches for the same molecules optimized from differing starting points are almost identical, and switches optimized from different molecules are very similar. Even though UA methane does not have the same optimal alchemical switch, there is little improvement in the variance when using this different curve.

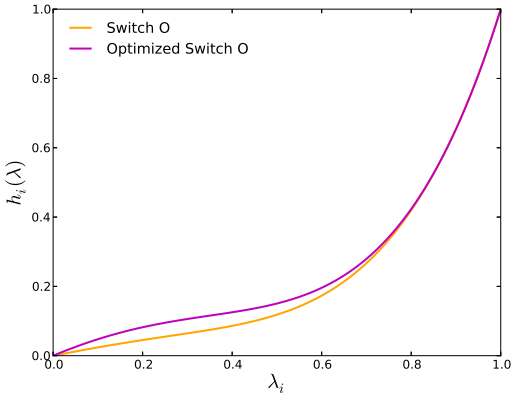
For n-decane, Switch O from the 3-methylindole data appears to be near optimal, though further optimization reduces the variance by about 7% from Switch O. However, for the large LJ sphere (with buckyball radius) we find that Switch O has a very large increase in the variance for λ between 0.4 and 0.6 (Fig. 2.13), requiring additional sampling between at $\lambda_i = \{0.55, 0.45, 0.35\}$ to obtain an accurate free energy. We applied the optimization routine to the large LJ sphere using the Switch A simulation to obtain a re-optimized switch which significantly reduced the peak in and the total variance (also Fig. 2.13). The re-optimized switch was then simulated *without* the extra sampling at $\lambda_i = \{0.55, 0.45, 0.35\}$ and gave a simulated free energy of 16.043 ± 0.090 kcal/mol and a variance of 45.74 ± 1.40 $(\text{kcal/mol})^2$; this is approximately 25% lower variance compared to other results in Table 2.3. This re-optimized alchemical switch is flatter in the region $\lambda = 0.4$ to $\lambda = 0.6$ as seen in Fig. 2.13c, meaning the flat cap in the large excluded volume region changes more slowly to maximize the phase space overlap between neighboring states. This extreme case of a C_{60}-sized sphere is a practical example of both how a single optimized switch (Switch O) derived from a single example molecule is better than standard simulation methods and near optimal, but also how easily even this pathway can be improved with a small amount of post-processing.



(a) $\left\langle \frac{\partial u}{\partial \lambda} \right\rangle$ Large LJ sphere



(b) Variance Large LJ sphere



(c) Comparison of original Switch O and re-optimized switch for large LJ sphere

Figure 2.13: **Extreme peaks in variance can be reduced by a simple post-processing re-optimization procedure.** Here is shown $\langle \partial u / \partial \lambda \rangle$, variance, and the switches for a large Lennard-Jones sphere for Switch A, Switch B, Switch O, and a re-optimized Switch O specifically for this molecule. The re-optimized switch was created by predicting the optimal switch from sampled data of Switch A. Having a nearly flat $h_i(\lambda)$ in the regions of large variance can drastically reduce the total variance as seen in (c). Errors are shown with dashed lines around the curves and were estimated by 200 bootstrap samples; error is often less than thickness of the curve.

In summary, although slightly improved paths can be determined for particularly extreme cases (such as very large or very small molecules), a single optimized switch for the repulsive core has nearly minimal variance for all molecules, with a range of sizes from methane to a buckyball, and asymmetries as large as n-decane. If required, optimization for individual molecules requires only a single initial simulation (using,

for example, the general Switch O) and then a simple reoptimization using only this simulation data.

2.5 Discussion

Our optimized linear basis potentials for all tested molecules have total variances, and thus statistical efficiencies between the total variances of the 1-1-6 and 1-1-48 soft core potentials, even for the initial guesses, Switch A and Switch B, as anticipated.

The most efficient possible alchemical switches are somewhat dependent on the molecular identity, though the switches optimized from single molecule are better for all tested molecules than the initial guessed switches or the 1-1-6 soft core potential. If the calculation is performed for a ligand-protein binding process, instead of solvation, the magnitude of the variance will change, but the relative efficiency of variance pathways will not change significantly. The ranges of λ with large variance will not significantly change along the same path since adding a protein to the solvent will not significantly change the density of the fluid around the ligand. We have illustrated this in Appendix A.3 as an example, though an in-depth comparison of variances in both protein binding and solvation is beyond the scope of this chapter.

Our variance prediction equation, Eq. (2.22), is a powerful tool for optimizing paths of lower variances. Switch O has lower variance with all tested molecules compared to the soft core 1-1-6 pathway, although a re-optimized switch was needed to obtain the lowest possible variance for the large LJ sphere. The extra sampled states and longer correlation times for the large LJ sphere are not unexpected due to the excluded volume being more than twenty times larger than UA methane. Although Switch O is slightly different than the minimum variance path in this extreme case, the simplicity of our variance minimization procedure allows a more optimal switch to be easily obtained. In this case, increasing the number of basis functions may allow an even

lower variance path for turning off such a molecule and their optimal switches could easily be generated with the methods described in this chapter.

For general molecular transformations, we recommend a capped potential with a transition region of $0.8\sigma_{ij}$ to $0.9\sigma_{ij}$ and transforming from an uncapped, full Lennard-Jones potential to a capped potential in a single step. This cap with a typical value of $6.2k_BT$ provides a statistically efficient balance of contributions to the variance from these two steps. We find that harder and softer caps do not reduce statistical error over the entire transformation and should be avoided. Application-specific considerations, such as high temperature systems, may need different transition windows to adjust the caps for lower variance in those situations.

Expanding the variance optimization routine to a high number of basis functions simultaneously is a non-trivial matter. The number of terms in the covariance matrix quickly increases with multiple basis functions. This can cause the optimization routine to fail or be very prone to initial conditions as the covariance becomes less well behaved. These problems will be compounded if multiple switch optimizations are attempted at once. For this reason, we do not recommend optimizing more than one or two switches at a time, depending on how many total basis functions there are, and explore more general optimizations in the next chapter.

2.5.1 Improvements and Implementation of the Basis Function Approach

Our method requires that alchemical switches be monotonic to make it possible to invert them. However, this is not a particularly limiting requirement. When considered as a path, the variance of an alchemical switch is directly proportional to the curvature of the switch and its thermodynamic length [93, 94, 96] as is derived in Appendix A.1. Monotonic switches will have less curvature than an otherwise similar non-monotonic switch, so intuitively should have lower variance. Additionally,

any sequence of sampled λ states which would require a non-monotonic switch can be reordered to be monotonic. For example, if samples are taken in the sequence $h_i(\lambda) = \{0.4, 0.5, 0.3, 0.6\}$, one could just as easily sample the sequence $h_i(\lambda) = \{0.3, 0.4, 0.5, 0.6\}$ without loss of information; any repeated h_i value is equivalent to more samples at that state. For these reasons, assuming monotonic switches is much easier and is not a practical limiting factor in the approach.

Although the curves match very well with previous results of Pham and Shirts, [84] the total error in our variance estimations is somewhat larger, as can be seen in Table 2.2. This is because we sampled with many fewer states using shorter simulations, and would need about 2 to 6 times more sampling to reach the level of uncertainty obtained in the previous study. However, the uncertainties in our estimates of total variance are still small enough to reach the conclusions in this chapter.

Specific applications may require changing the basis functions themselves or simulation along other alchemical schedules. Such changes could involve different basis functions instead of the capped WCA decomposition, adding one or more additional basis function terms in Eq. (2.4), or using a different schedule with repulsive and attractive terms turned off separately. For example, one could add a second capped repulsive term with a cap at $r > 0.8\sigma_{ij}$ and add a step to transform between different capped basis functions. This may improve phase space overlap as the core is softened, but may also require more intermediate states. However, increasing the number of basis functions could lead to overfitting and increased complexity. Any increased complexity in the potential energy function may not be worthwhile to implement if there are not sufficient gains in statistical efficiency. In all of the wide variety of molecular shapes and sizes examined, the procedure presented here gave variances at most 25% larger of the previously found 1-1-48 minimum variance pathway, [84, 97] and is lower in one case. It is thus unlikely that the increase in complexity would be worth the marginal improvement.

Removing the free energy calculations from the force evaluation loop can make the simulation more easily adaptable to new software or computing paradigms. This linear basis approach makes this removal much easier. Free energy calculations require either the potential energy of configurations evaluated at other thermodynamic states, or require an analytic computation of the derivative of the Hamiltonian with respect to the coupling parameter. Traditional soft core potentials such as Eq. (2.1) require modifying the λ value and evaluating the potential and/or its derivative for each configuration in the inner loop. With the basis function approach, one would only have to evaluate each basis function, $u_i(r)$ in Eq. (2.4), once for a configuration and store them temporarily in memory. Evaluating the configuration's potential at other thermodynamic states can then be done by calculating all the $h_i(\lambda)$ of that state, then evaluate the products with the corresponding stored basis function and summing over all terms. If only the total potential is stored, the value of the basis functions can be solved through linear algebra by evaluating configurations at multiple states. $\partial u/\partial\lambda$ can also be computed external to the inner loop using the basis function energies along with knowledge of $\mathbf{h}'(\lambda)$.

2.6 Conclusions

This chapter develops an approach for designing pathways for statistically efficient free energy calculations involving removing or inserting molecules into dense fluids. We seek to retain functions which are easier to parallelize to new architectures than soft core potentials while still keeping the low variance of these approaches. We can achieve both of these objectives using the formalism of linear basis function potentials, consisting of linear combinations of basis functions with alchemical switches to represent the nonbonded potential energy function. The linear basis potential form is shown to greatly simplify the math needed to do potential energy re-evaluation or calculating

derivatives of the potential, making it simple to estimate the potential energy at other states. This simplified math allows for potential energy re-evaluation to be carried out entirely in post-processing. One generally applicable set of basis functions and three sets of alchemical switches, one of which is provably optimal for the choice of basis set, were presented as examples and were all shown to have lower total variance, and therefore lower statistical uncertainty, than the soft core 1-1-6 alchemical pathway for a diverse molecule set.

The simplicity and power of predicting variance and $\langle \partial u/\partial \lambda \rangle$ from the basis potential energies makes it possible to easily find an optimal alchemical switch like Switch O that is simple enough for general usage in other molecular simulation packages. The fact that only one simulation is needed for a given system to find the optimal switch is a large improvement over optimizing pathway parameters requiring multiple, iterative simulations. Indeed, the entire $\langle \partial u/\partial \lambda \rangle$ curve can be generated for arbitrary points along any other alchemical path as long as the original path is sampled sufficiently finely for the initial simulation, with 10-12 states generally enough.

From this chapter, we can recommend a four basis function approach for decoupling small molecules from dense fluids like water. The basis consists of electrostatic, capping, attractive, and repulsive basis functions. The electrostatic basis function is simply the standard electrostatic term, and is turned off before the other three basis functions using a simple linear coupling, although the optimal switch for this transformation was not examined in this chapter. The potential is then transformed in a single step to a capped WCA-decomposed potential composed of the attractive and repulsive basis functions to avoid quickly disappearing a singularity at $r = 0$. These two terms are turned off simultaneously using a linear term for the attractive function, and the three-parameter quartic polynomial of Switch O for the repulsive term, or a similarly shaped alchemical switch. Further improved pathways may be possible, but the procedure described in this chapter gives more efficient pathways compared to

standard approaches currently used. Further analysis of the alchemical schedule and alternate ways to include electrostatics will be the focus of Chapter 3.

Chapter 3

Maximizing Statistical and Computational Efficiency Sampling with Basis Functions

3.1 Introduction

This chapter has previously been published [55] as: Naden, L. N.; Shirts, M. R. Linear Basis Function Approach to Efficient Alchemical Free Energy Calculations. 2. Inserting and Deleting Particles with Coulombic Interactions, *J. Chem. Theory Comput.*, 11:2536-2549, 2015.

Calculations of free energy differences by computer simulation can provide valuable insight into molecular thermodynamics without experimental measurements. Rigorous statistical methods allow computer simulations to estimate free energy differences through clearly defined thermodynamic states and sampling schemes. [133] Alchemical transformations are a common computational technique which provide the difference of free energies at two end states along a thermodynamic path connecting them. Sampling this path provides the information needed for statistical analysis and estimation of the free energy difference. A popular application of alchemical methods is the computation of drug binding in order to understand the molecular details of drug action or to reduce the number of molecules required to be synthesized. [12, 106, 109, 134–136] But there are a wide range of other applications for alchemical methods where knowledge of the chemical potential of a small molecule in some environment may be useful, such as estimating membrane permeability or solubility.

There can be significant computational costs associated with alchemical free energy calculations due to the number of intermediate states along the thermodynamic pathway that must be sampled to achieve sufficiently high precision. Intermediate states are chosen to provide larger phase space overlap between adjacent states which reduces the uncertainty in evaluating free energies and thermodynamic expectation values. A range of methods can be used to estimate free energies including exponential averaging (EXP), [98] thermodynamic integration (TI), the Weighted Histogram Analysis Method (WHAM), [99, 100] the Bennett acceptance ratio (BAR), [105] and its multistate version (MBAR). [101] Each of these methods has their own advantages

and disadvantages. [60, 102] Estimating free energies of solvation and absolute binding affinities typically require numerous intermediate states because the phase space overlap is low between the state where the entire, large solute is fully interacting with its environment and the state where the solute is fully non-interacting.

A way to reduce this computational cost is to design thermodynamic paths which improve the phase space overlap between intermediate states. [84, 85, 94, 95, 97] The "soft core" approach is one way to define a pathway of potential energy functions from a fully interacting state to a fully non-interacting state. [82, 83] The pathways in potential energy space that can be described by this approach provide decent phase space overlap between neighboring states, which reduces the statistical uncertainty of free energy calculations performed along path and thus the number of samples needed to obtain a given level of precision. [105] Minimizing the statistical uncertainty of calculations performed along an alchemical path is equivalent to minimizing the thermodynamic length of the path. [54, 84, 93, 94, 96, 97] The path which minimizes the total uncertainty for the transformation is one which has an equal contribution to the uncertainty across every point along the path. However, the minimum not necessarily achievable via pairwise potentials [97].

To clarify the objectives of the chapter, we must introduce some statistical definitions. Using thermodynamic integration (TI) to calculate free energy differences involves calculating the average of $\partial u/\partial\lambda$ at each value of λ, and then integrating this average numerically from $\lambda = 0$ to $\lambda = 1$ to obtain the total free energy of the transformation. In the standard case when simulations at each state are performed independently, the variances in the mean of $\partial u/\partial\lambda$, which are statistically independent, weighted by the scalar factors used in the numerical integration method, will add in the square to produce the variance of the total calculation of the free energy difference. The square root of this variance is the standard deviation of the overall calculation. By adjusting the pathway of potential energy functions that connect the end states,

we can lower the variance of the overall calculation by increasing the overlap between individual Hamiltonians simulated.

When comparing pathways, the most statistically efficient path is roughly independent of the method performing the analysis such as TI, BAR, or MBAR [54, 97] Since the variance of the mean in $\partial u/\partial\lambda$ at any simulated state will be proportional to the number of samples at that state, the overall variance of the entire process will also be proportional to the number of samples used for the entire process at all states, assuming fixed proportion of samples per state and samples that are statistically decorrelated. We can thus normalize this variance in the calculation by the total number of decorrelated samples used in the calculation to obtain a measure of the statistical efficiency of the pathway that is independent of the length of the simulations. We will call this measure the variance of the pathway (or variance of the path).

Choosing the most statistically efficient pathway (i.e. the path with lowest variance of the pathway) through alchemical space is non-trivial. A number of efforts have been made to minimize the variance of the soft core path by changing the parameters in the potential energy function. [58, 84–88, 95, 96] and an efficient formalism has been developed for calculating minimal variance paths within a given function space. [54, 84, 97]

The functional form of the soft core potentials adds significant complexity when implementing free energy methods on new, highly parallelized architectures such as GPU's or Many Integrated Core (MIC) platforms. As described in Chapter 2 and our previous study, [54] soft core methods require writing special code to handle the alchemical modifications to atoms and implementing these modifications on new platforms have lagged behind the implementation of other molecular mechanics methods. If the alchemical-specific code can be removed from the inner force loop of molecular simulations, the time needed to implement free energy methods on these new platforms could be drastically reduced.

As shown in Chapter 2, representing the thermodynamic path with a linear combination of basis functions makes it possible to remove alchemical specific code from inner force loops. [54] The basis function approach splits the potential energy into basis functions involving only the spatial coordinates of a system, and alchemical switches depending only on the transformation coordinate. We also showed how this basis function approach made it possible to find a near statistically optimal thermodynamic paths to be computed from a single trial simulation entirely in post-processing. This optimized linear basis function method was just as statistically efficient as soft core methods for uncharged Lennard-Jones particles.

Chapter 2 looked only at the introduction and removal of Lennard-Jones interactions. This chapter examines the most statistically efficient way to couple electrostatics. It also identifies the lowest uncertainty sequence in which all nonbonded forces can be coupled to the rest of the system, as this order becomes an additional variable when there are multiple basis functions. To complete the analysis of the transformation of alchemical nonbonded components, the following design questions must be addressed:

1. What combination of basis functions and alchemical switches produces the minimum variance pathway to modify the charge of a molecule, including reducing the charge to zero?

2. In which sequence should the entire set of nonbonded interactions be alchemically coupled to the environment to give minimum variance pathways for insertion or deletion of a charged molecule?

The answers to these questions extend the low variance linear basis function approach to nearly all nonbonded alchemical changes. This chapter looks at the statistically efficient way to alchemically couple electrostatics to the environment by analyzing the charging of ions with a fully repulsive core, inversion of charge on a dipole, and turning on the charges in ethanol and 1,4-butanediol molecules in water.

This chapter also answer the question of which is the most statistically efficient order in which the nonbonded forces are coupled. This order, or "schedule" as referred to here, becomes variable in the process when forces can be alchemically changed separately and can be successfully implemented in many separate ways. [85, 92, 109, 114, 137, 138] This chapter identifies the most statistically efficient schedules across different sets of basis functions and compares them to a soft core scheme in which all forces are coupled together using the tryptophan side chain analog, 3-methylindole. This chapter focuses on inserting and deleting entire atomic sites as this class of alchemical transformation is the most challenging. [82, 83, 85, 95, 109]

Results for optimal charging pathways are valid for both absolute and relative free energy calculations. Although we only benchmark the approaches for introduction of deletion of sites with absolute free energy calculations in water, the same pathways will also be efficient for insertion or deletion into other dense fluids. The same principles will apply for relative free energy calculations in which multiple heavy atoms are inserted, as the fluid density around these sites is mostly independent of the fluid density around the unchanging parts of the molecule. The proposed transformations may not be maximally efficient for relative free energy calculations with small changes, but such calculations require many fewer intermediate states and thus require much less optimization.

3.2 Theory

The notation is identical in this chapter and the previous chapter, [54] with only the addition of notation for electrostatic terms. σ_{ij} and ϵ_{ij} are the Lennard-Jones parameters between sites i and j, and q_i and q_j are the charges of particle i and j respectively. ϵ_0 is the permittivity of vacuum. Subscripts E, R, A and C denote electrostatic, Lennard-Jones repulsive, Lennard-Jones attractive, and capping potential

energy terms respectively, e.g. $u_E(r)$ is the electrostatic basis function. The capping potential is the basis function which transitions from uncapped repulsive potential to a capped repulsive near $r = 0$.

The theory presented here relies on the variance minimization theory laid out in sections 2.2 and 2.4.1.

3.2.1 Selecting a Statistically Efficient Alchemical Schedule

An alchemical "schedule" is defined by what order the nonbonded forces are coupled to the environment. Chapter 2 assumed a schedule where the Lennard-Jones repulsive and attractive interactions changed simultaneously, so $h_R(\lambda)$ varied with $h_A(\lambda)$ along the entire $0 \leq \lambda \leq 1$ interval in Eq. (2.17), followed by a small step where $h_C(\lambda)$ was fully coupled to complete the process. Changing interactions independent of each other causes each $h_i(\lambda)$ to change at different rates with λ, such as when decoupling electrostatics before decoupling Lennard-Jones. Examples of two different schedules are illustrated in Fig. 3.1. The upper schedule shows attractive (h_A) and repulsive (h_R) switches being fully coupled before electrostatics (h_E), and the lower schedule shows repulsive force coupling continuously as electrostatic and attractive are alternating coupled. The height of each line shows the value of the respective $h_i(\lambda)$. Some interactions may become fully coupled before $\lambda = 1$ as is the case for h_R in the upper schedule and h_A in both schedules.

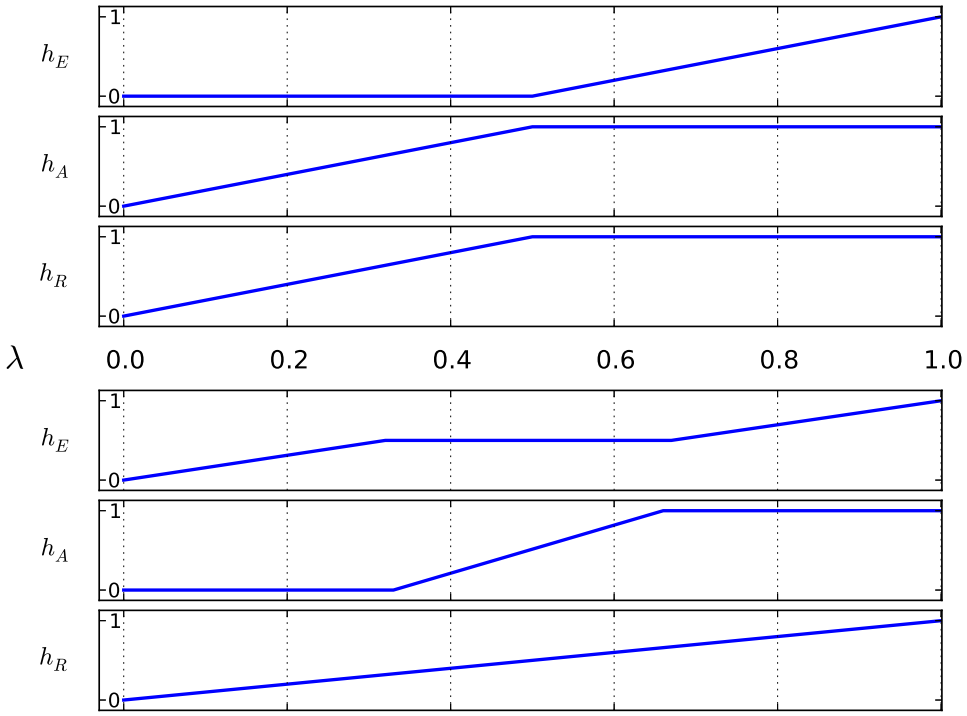


Figure 3.1: Two examples of alchemical switches with different schedules. Switch labels are shown to the left of each plot. Each switch is allowed to change over different ranges of the λ parameter. Height of each line is a relative value of $h_i(\lambda)$. The upper schedule shows full coupling of repulsive and attractive forces before electrostatics. The lower schedule has the repulsive potential continuously increasing but electrostatics and attractive switches alternating.

The variance between separate alchemical schedules cannot be mapped by Eq. (2.22) as the phase space is not shared between schedules. Consider trying to map an electrostatic switch $h_E(\lambda_h)$ from a sampled λ_h where the repulsive interaction is present to a λ_g where the repulsive force is removed. There will be little to no phase space overlap between these two systems. $h_E(\lambda_h)$ will therefore not contain all the thermodynamic information about $h_E(\lambda_g)$, making the mapping process impossible to use. Once an alchemical schedule is chosen, then a minimal variance pathway can be computed.

Choosing Reasonable Alchemical Schedules

To simplify the analysis, we first identify schedules which provide converged numerical answers and are reasonable to implement. As a first simplification, the alchemical schedules assume that each type of interaction is fully coupled or decoupled over a single, continuous space of λ monotonically. This assumption ignores some less likely possibilities; e.g. removing half of the charge, removing the attractive term, then removing the other half of the charge; or removing the attractive term, and then decreasing the charge while increasing the attractive term. Section 3.2.1 revisits these non-monotonic pathways and argues that they cannot substantially improve the statistical error associated with each pathway.

Examining the possible orders to couple the three types of interactions (repulsive, attractive, and electrostatic), either sequentially or simultaneously, leads to 24 possible permutations of alchemical schedules. As an example of these permutations, for the upper schedule of Fig. 3.1, the first set of switches is $\{h_R,h_A\}$ followed by $\{h_E\}$. Other alternatives include changing the switches simultaneously, $\{h_R,h_A,h_E\}$, or each switch could be changed one at a time, with six possible orderings.

This chapter distinguishes the separate schedules with a shorthand notation. A sequence of letters denotes the schedule along which force is coupled to the environment, and stages in the schedule are separated by a slash (/). Electrostatic, repulsive, and the attractive switches are denoted by capital E, R, and A respectively. Additionally, each schedule must have a step to remove the cap on the repulsive potentials imposed by the linear basis approach, denoted by C. As the soft core methods do not have a cap stage, they will not have a C in their schedule. Although the schedules, derivations, and discussions shown here are for coupling, the process of decoupling is equivalent and would simply be the reverse ordering of the schedules. For example, the upper schedule in Fig. 3.1 would be a soft core schedule and denote AR/E since there is no explicit removal of the cap. All three forces in soft core changing together would be

denoted by AER which is equivalent to RAE, EAR, and so forth.

We can quickly eliminate many of the 24 possible permutations. Basis function schedules must have a separate stage for coupling C, so AER would be invalid for the basis function approach. As an example linear basis function approach with a cap: coupling electrostatics, then attractive, then the capped repulsive, and lastly removing the cap would be denoted E/A/R/C. The repulsive force is always capped when it is coupled, then uncapped in the next step, i.e. C will always be in the stage after R. Such an ordering of R and C is required to have well-converged results. Adding these requirements reduce the 24 permutations to 13 unique ones. These unique permutations are summarized in Table 3.1 and grouped into classes discussed in the Appendix B.1.

Table 3.1: Only a finite number of alchemical schedules are reasonable to use. Permutations of coupling electrostatics (E), attractive (A), and repulsive (R) forces with a cap (C) and logical arguments allow reducing the 13 unique permutations down to 4 reasonable ones. This table shows the unique permutations and categorizes them. The schedules can be **Reasonable**, **Attractive Core** where a negative infinite potential can be generated at $r = 0$, or **Large Coupling Variance** where the cap required on the repulsive interaction [54] is large enough to generate a high variance to decouple it. Schedules are shown by order of coupling e.g. R/C/EA fully couples the repulsive interaction, then removes the cap on the repulsive force, then fully couples the electrostatic and attractive interactions together. * indicates that this schedule is only reasonable using the WCA decomposition.

Reasonable	**Attractive Core**	**Large Coupling Variance**
AR/C/E*	A/E/R/C	ARE/C
R/C/AE	A/ER/C	E/AR/C
R/C/A/E	A/R/C/E	ER/C/A
R/C/E/A	AE/R/C	E/R/C/A
	E/A/R/C	

Further analysis shows that there are only 4 of the 13 unique schedules that need to be examined in detail to identify the lowest variance pathways, which are R/C/A/E,

R/C/E/A, R/C/AE and AR/C/E. These are all paths in which the capped repulsive basis is first turned on, then the cap is removed, then the electrostatics can be turned on. The only difference is the timing of the addition of the Lennard-Jones attractive term. It could be turned on between the capping term and electrostatics term, after the electrostatics term, or during either the capped repulsive basis or the electrostatics term. The details of the logic restricting our search to these four sequences is given in Appendix B.1.

Non-monotonic or non-sequential pathways will have larger variances than monotonic and sequential pathways

We argue that we need only consider monotonic changes of λ. The large curvature of non-monotonic switches increases the variance of any pathway, independent of alchemical schedule. The variance of each schedule is directly proportional to the curvature in $\partial u/\partial\lambda$, [54, 93, 94, 96] which itself is a function of $\partial^2 h_i/\partial\lambda^2$. Any force controlled by a non-monotonic switch would naturally have more curvature, and thereby pathway variance, than any monotonic switches. The 13 unique schedules in Table 3.1 assume that each force is fully coupled to the system in one stage then remains fully coupled, and the alchemical switch controlling that force is monotonic.

We also must consider whether it might make sense to couple a single force over multiple stages. The lower schedule in Fig. 3.1 shows a process where the electrostatics are coupled only halfway in the first stage of the transformation, held constant in the second stage, and then coupled the remainder of the way in the last stage. This is in contrast to all of the suggested schedules where each force is coupled in a single stage. We can remove the need to test schedules where individual forces are coupled in multiple stages with two logical arguments.

- **The repulsive force must be fully coupled in the first stage.** The previous section and Appendix B.1 argued that electrostatics should be coupled

only after the repulsive force is fully coupled. The attractive force can only be coupled with the repulsive force if the potential has a WCA decomposition, but the repulsive force must still be the first stage to prevent an attractive core from forming.

- **Identifying non-intuitive covariance terms would be the only way to lower the variance below fully coupling a force in a single stage.** Since electrostatics and repulsive must be coupled separately from each other, the only force which could be coupled in multiple stages is the attractive force. Any reduction in variance from coupling the attractive force in multiple stages would come from the covariance terms in Eq. (2.17) The attractive component can only be coupled with the repulsive force in the WCA decomposition, and we will show contributes a small amount to the variance in this case. As such, any $\mathrm{Cov}\left(u_R, u_A\right)$ terms will only provide small reductions in the total variance of the pathway. Since electrostatics can have both attractive and repulsive interactions, $\mathrm{Cov}\left(u_E, u_A\right)$ is system dependent so no general trend can be established for changing $h_A(\lambda)$ in non-monotonic ways.

Because coupling the attractive force across multiple stages will only provide marginal reduction in the variance at best, and because repulsive and electrostatic forces must be coupled separately, we believe there is not enough value in designing schedules where any force is coupled over more than one stage.

3.2.2 Basis Functions for Electrostatics and Alternate Lennard-Jones Basis Functions

The standard form of a pairwise electrostatic interaction for point charges is the basis function studied here. Since this chapter is not examining schedules which require capped electrostatics, there is no need to create a cap as is done for the WCA

decomposed Lennard-Jones basis functions. [54] This makes the electrostatic basis

$$u_E(r) = \frac{q_i q_j}{4\pi\epsilon_0 r}. \tag{3.1}$$

The corresponding, minimal pathway variance alchemical switch, $h_E(\lambda)$, still needs designed, but this can be done with a trial switch, Eq. (2.22), and the method summarized in Section 2.4.1.

Decomposing the electrostatics requires implementation based considerations. If the electrostatics are decomposed into short- and long-range contributions such as reaction field or particle mesh Ewald (PME), [139] then additional basis functions may be required. Specifically for PME, there is a term in the potential energy which scales as $h_E^2(\lambda)$ which cannot be neglected due to long-range interactions of each alchemical atom with other alchemical atoms in periodic copies. Because the alchemical implementation may be different based on software, a general solution cannot be provided here, but this chapter's specific implementation is discussed in Appendix B.3.

Exploring an alternate set of basis functions to the WCA decomposition of Lennard-Jones interactions can help generalize the basis function method. In Chapter 2 and Section 3.2.1, a capped WCA decomposition was assumed which takes the form of Eqs. (2.17) -(2.8). An alternate set of Lennard-Jones functions can be defined which splits the repulsive and attractive interactions based on their respective exponent of r^{-n}. These 12-6 basis functions can be written as

$$u_{12\text{-}6}(r) = u_{12}(r) + u_6(r) \tag{3.2}$$

where the subscripts 12 and 6 denote the repulsive and attractive terms respectively. The 12-6 basis functions are simply

$$u_{12}(r) = \frac{4\epsilon_{ij}\sigma_{ij}^{12}}{r^{12}} \tag{3.3}$$

$$u_{6}(r) = \frac{-4\epsilon_{ij}\sigma_{ij}^{6}}{r^{6}}. \tag{3.4}$$

The u_{12} basis will need to be capped in the same way as the WCA decomposed basis functions and is defined as

$$\begin{aligned} u_{12,\text{cap}} &= u_{\text{cap}} && \text{for } r < f_{\text{cap}}\sigma_{ij} \\ &= Ar^4 + Br^3 + Cr^2 + Dr + E && \text{for } f_{\text{cap}}\sigma_{ij} \leq r < f_{\text{switch}}\sigma_{ij} \\ &= u_{12}(r) && \text{for } r \geq f_{\text{switch}}\sigma_{ij} \end{aligned} \tag{3.5}$$

where $u_{\text{cap}} = u_{12}(f_{\text{cap}}\sigma_{ij})$, $f_{\text{cap}} = 0.82$, and $f_{\text{switch}} = 0.92$ to keep the cap height between $3.5k_BT$ and $8.8k_BT$ and minimize the variance in the pathway from switching between an infinite potential and the capped potential. [54] The constants A–E are determined by ϵ_{ij}, σ_{ij}, f_{cap}, and f_{switch} in the same way as the constants of Eq. (2.10). This set of basis functions is numerically stable for all schedules in Table 3.1 except AR/C/E as denoted since the attractive interaction is not capped. Each schedule will be affixed with either "-WCA" or "-12-6" in its name to distinguish between the WCA and 12-6 and basis functions respectively.

3.3 Experimental Design

Molecular dynamics for solvation of molecules with electrostatic interactions were carried out with YANK [116, 117] which was built on GPU accelerated OpenMM v4.1.1 [25, 39, 118, 119, 122] in explicit TIP3P water. This includes the charging of positive and negative ions, dipole inversion, charging of all-atom (AA) ethanol,

charging of AA 1,4-butanediol, and insertion of the AA side-chain analog to tryptophan, 3-methylindole. OpenMM provides a platform to rapidly implement arbitrary basis functions and alchemical switches. However, it does not directly compute derivatives of the energy with respect to the coupling parameter, which are required to calculate variance in the mean at each point. In the case of a basis function pathway, these derivatives can easily be recomputed from the potential energy differences. This is not possible for the soft core approach and so soft core alchemical simulations with all forces changing at once were carried out with GROMACS v4.6.4. [35, 38] The validation that the thermodynamic properties being computed are comparable between the two simulation packages is discussed below.

Solvation free energy simulations and charging-only transformations were tested with a variety of different molecules. Charging of a molecule was carried out on Na^+, Cl^-, a dipole made from united atom (UA) ethane with +0.5/-0.5 charge, AA ethanol, and AA 1,4-butanediol. Full insertion of a molecule with partial charges was tested with AA 3-methylindole. The charged species were chosen to test if opposite signs have different optimized paths in a polar solvent such as water. Ethanol was chosen to see if asymmetric charge density changes the optimized switch. 1,4-butanediol was selected as a molecule with strong propensity to internally hydrogen bond in vacuum, which would be disrupted by the introduction of polar solvent. 3-methylindole was selected as it is the small molecule analog of the largest protein side chain.

All molecules were initially constructed using AMBERTOOLS's LEaP with OPLS-AA force field parameters for all molecules except the dipole. The dipole's starting geometries were from taken from Paliwal and Shirts [60]. 1,4-butanediol structure was imported from PubChem (CID 8064). 3-methylindole's starting molecular geometry was acquired from the supplementary material from Mobley et al. [89]. All molecules were inserted in a periodic cubic box of TIP3P water with boundaries 1.2 nm from the solute. The number of waters in ion systems were 633 and 707 for Na^+ and Cl^-

respectively. The UA ethane dipole had 1405 waters, the ethanol system had 769 waters, the 1,4-butanediol system had 872 waters, and the 3-methylindole system had 961 waters.

YANK simulations were carried out under isothermal-isobaric (NPT) conditions at 298 K and 1 atm. Each state was simulated for 2 ns with a 2 fs time step, samples collected every 1 ps, and Hamiltonian replica exchange [120] between all states attempted every 1 ps with Gibbs sampling to improve replica mixing efficiency. [121]. 3-methylindole's bonded hydrogens were constrained by the SHAKE algorithm [124] and water was constrained by the SETTLE algorithm. [125] Pressure control was handled by a Monte Carlo barostat [126, 127] and temperature control through Langevin dynamics. The nonbonded cutoff was 0.9 nm and interactions outside this cutoff were handled by PME with a relative error tolerance of $5 \cdot 10^{-4}$. Although Hamiltonian replica exchange was not required for these systems, there was negligible computational effort to running with it in YANK.

GROMACS simulations were run with binaries compiled in double precision. NVT equilibration was carried out for 100 ps followed by NPT equilibration for 500 ps before NPT production simulations of 6 ns per simulated state. Temperature was held at 290 k and maintained through Langevin dynamics. Pressure (for NPT) was maintained at 1 atm with a Parrinello-Rahman barostat, [140, 141] a time constant of 5 ps, and a compressibility of $4.5 \cdot 10^{-5}$ bar^{-1}. Alchemical sampling for the soft core path was done at 21 evenly spaced λ values from $\lambda = 0$ to $\lambda = 1$. Replica exchange was not done with GROMACS, however, only decorrelated samples were examined in both simulations which should provide the same thermodynamic results.

The two simulation packages were validated to ensure the thermodynamic properties being computed are directly comparable. The potential energy from an NVE trajectory in OpenMM was computed by both GROMACS and OpenMM and was found to be identical to machine precision with the simulation settings used in this chapter.

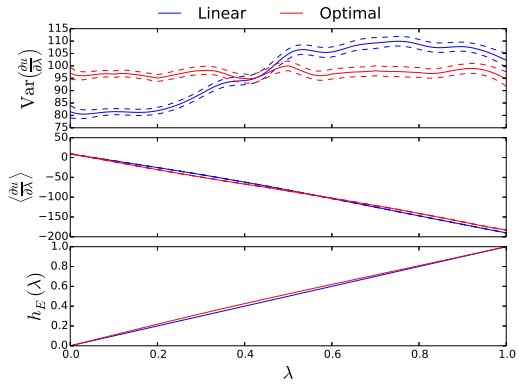
Although thermostats and barostat were different, only algorithms preserving the correct energy distributions were used [142] so that ensembles were directly comparable. Different thermostats do affect the correlation time of samples, so all trajectories were decorrelated before comparison. [143]

The analysis code and our implementation of the linear basis function method for OpenMM and YANK can be found on GitHub [144].

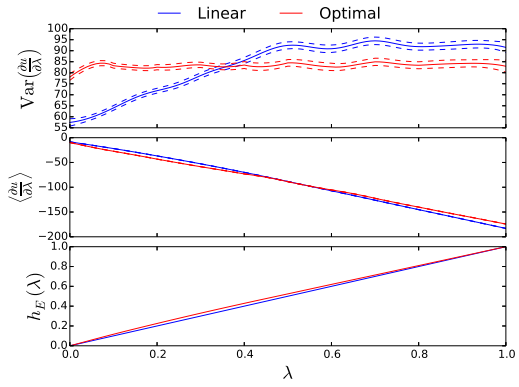
3.4 Results and Discussion

3.4.1 Coupling Electrostatics

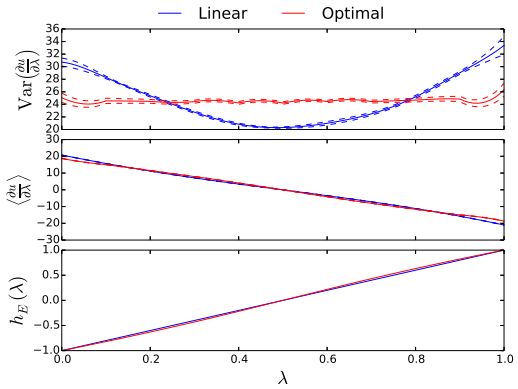
Linear coupling of electrostatics is almost exactly as statistically efficient as the minimum variance optimal path. Fig. 3.2a and Fig. 3.2b shows the sample variance of $\partial u/\partial\lambda$, $\langle\partial u/\partial\lambda\rangle$, and optimized alchemical switches for coupling the electrostatics of the Na^+ and Cl^- ions. The optimized switch is recalculated in post-processing with the technique from Section 2.4.1 and detailed in the Chapter 2. There is no significant difference in the shape between the optimized switches of the two ions, and virtually no difference between either optimized alchemical switch and the simplest linear switch $h(\lambda) = \lambda$. The variances of the pathway using these optimized switches are nearly flat and approach the theoretical minimum of a perfectly flat variance of the path over the transformation. [84, 97] Total variance for the optimized path is less than 0.5% lower than linearly coupling the electrostatics Thus, there is no reason to use anything other than linear coupling of electrostatics for ions given how little reduction there is in the variance of the pathway, especially given the fact that the optimized switch would require implementing some more complicated functional form.



(a) Charging a sodium ion, Na^+



(b) Charging a chlorine ion, Cl^-



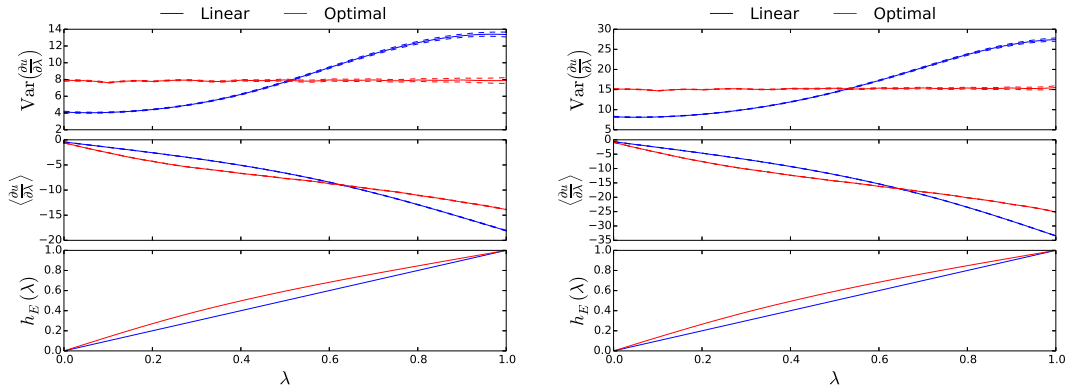
(c) Dipole inversion

Figure 3.2: A linear alchemical switch is nearly optimal for coupling charged particles. The variance, $\langle \partial u/\partial \lambda \rangle$, and electrostatic alchemical switch for charging ions and inverting a dipole are shown. Simulations were run with a linear switch (blue curves) and the minimized variance curve computed in post-processing (red curves). In all three cases, the minimal variance was nearly flat, indicating an approach to minimum variance over all possible (even non-pairwise) potentials. The optimized switches did not change significantly from a linear switch. Total variance reduction using an optimized path is less than 0.5%. This implies a linear switch is sufficiently nearly optimal for simple particles. Units are $(\text{kcal/mol})^2$ for variance and kcal/mol for $\partial u/\partial \lambda$; error is shown as dashed lines around curves.

The dipole inversion behaves similarly to the ion solvation in that the linear coupling path has nearly the same pathway variance in the calculation as the minimal variance path. The sample variance in $\partial u/\partial \lambda$, $\langle \partial u/\partial \lambda \rangle$, and electrostatic switch for the dipole inversion are shown in Fig. 3.2c. This transformation is symmetric around

$\lambda = 0.5$ as required by the symmetry of the system. The total variance along the optimized path is only marginally better than the linear transformation, reducing the total variance by only 0.5% with respect to the linear path. We again conclude that alchemically changing the charge with a linear switch will be suitable for most applications involving changes in charge of simple particles.

The optimized charging path for molecules with a more complicated charge distribution again results in only a small reduction in variance of the pathway compared to the linear path. Unlike the ions or the dipole, the optimized path for charging ethanol and 1,4-butanediol is not as close to linear as shown in Fig. 3.3a and Fig. 3.3b. However, the minimum variance switch still reduces the variance less than 5% from the linear switch in both cases. Implementing this optimized switch could probably be done with some sort of low dimensional polynomial, but it is also not clear how this polynomial might change on a system-by-system basis. We note that in the cases of ethanol and 1,4-butanediol, the total variance in $\partial u/\partial\lambda$ associated with the charging calculation is rather small compared to the variance of charging a large dipole, and even smaller than the removal of a repulsive site. [54] This is likely because any nonlinear terms relate to the higher moments of the charge distribution, which will contribute less energy than the monopole and dipole components. As such, the rest of the analysis will be carried out assuming a linear switch for electrostatic coupling, as the additional marginal efficiency gained will never be worth the complication.



(a) Charging ethanol

(b) Charging 1,4-butanediol

Figure 3.3: A linear switch is near minimal variance, even for molecules which strongly interact with the solute and itself. The variance in $\partial u/\partial\lambda$, $\langle \partial u/\partial\lambda \rangle$, and electrostatic alchemical switch for charging ethanol and 1,4-butanediol are shown. The linear switch (blue curves) is not as close to the minimal variance switches (red curves) as is the case in the charging ions and dipole inversion. The theoretical minimal variance is approached as seen from the flat variance, however, the improvement in variance is less than 5% for the ethanol and less than 4.4% for the 1,4-butanediol. Since this is not a significant reduction in variance, the best general purpose alchemical switch is a linear one when considering that linear switches are also nearly optimized for particles carrying full charge. Units are $(\text{kcal/mol})^2$ for variance of $\partial u/\partial\lambda$ and kcal/mol for $\langle \partial u/\partial\lambda \rangle$; error is shown as dashed lines around curves.

3.4.2 Identifying the Optimal Alchemical Schedule

The variance of pathway to calculate the free energy difference is not simply the sum of the sample variances in $\partial u/\partial\lambda$ at each point. Because the sample variance in $\partial u/\partial\lambda$ and free energy are estimated along a TI path, the variance of the calculation of free energy is computed by adding variances in the mean of $\partial u/\partial\lambda$, weighted by the scalar factors used in the numerical integration. Up to this point, the variances of $\partial u/\partial\lambda$ discussed have simply been the sample variances. For example the sample variances in $\partial u/\partial\lambda$ computed for the AR/C/E-WCA path are shown in Fig. 3.4.

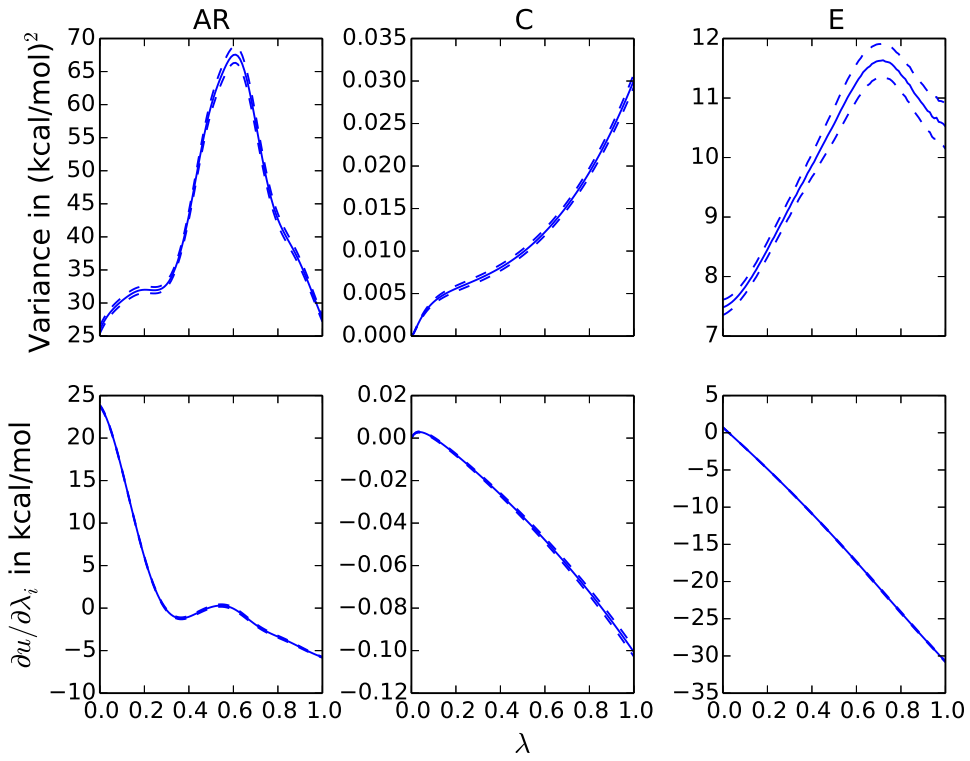


Figure 3.4: The sample variance of $\partial u/\partial\lambda$ in each stage of the transformation is not additive on its own. The variance in $\partial u/\partial\lambda$ and $\langle\partial u/\partial\lambda\rangle$ (lower) for the AR/C/E-WCA basis are shown. Only by dividing by the number of samples drawn from each can the variances shown be added. Each stage in the transformation is given by its shorthand letter above the column described in the text. The capping stage (C) contributes very little to the total variance, even without dividing by the number of samples drawn. Error is shown as dashed lines around the figure and often smaller than the thickness of the line.

The integral of such a variance curve is related to the thermodynamic length, or Riemannian length ($\mathcal{L}$), [54, 84, 93, 94, 96, 97] between the states. The thermodynamic length can be computed by

$$\mathcal{L} = \int_0^1 \sqrt{\mathrm{Var}\left(\frac{\mathrm{d}u}{\mathrm{d}\lambda}\right)}. \tag{3.6}$$

The variance minimization efforts up to this point have focused around minimizing this length, which happens when the sample variance in $\partial u/\partial\lambda$ is flat. However, as the number of decorrelated samples from each schedule is different, and the distribution of

states is different between the basis functions and the soft core path, we cannot compare the sample variances in $\partial u/\partial\lambda$ alone to claim which path is the most statistically efficient.

A fair comparison of thermodynamic pathways requires distributing discrete samples in a statistically optimal manner for that path. If sampling is done at discrete states, the variance of the total calculation of the free energy, $\mathrm{Var}\left(\Delta F\right)$ is related to sample variance in $\partial u/\partial\lambda$, s^2, as

$$\mathrm{Var}\left(\Delta F\right) \geq \sum_{i=1}^{N_{\mathrm{stage}}} \sum_{j=1}^{K_i} \frac{w_{i,j}^2 s_{i,j}^2}{n_{s,i}} \tag{3.7}$$

where i loops over each of the N_{stage} stages in the transformation, j loops over each of the discrete K_i states in stage i, and $s_{i,j}^2$ is the sample variance from state j in stage i. $n_{i,j}$ is the number of samples drawn from state j and stage i. $w_{i,j}$ are the weights of the quadrature method used in the TI integration to calculate the free energy. [145] By choosing a consistent distribution of states and samples to each state, a fair comparison between each schedule can be made by comparing the variance of the overall free energy calculation from Eq. (3.7).

There are two limiting cases to consider when distributing samples between states for a fair comparison of the alchemical methods. In the maximally efficient case, sampling is done proportional to the anticipated sample standard deviation of $\partial u/\partial\lambda$. [94] This choice will maximize statistical efficiency of the entire calculation. However, this requires *a priori* knowledge of the sample variance, which is generally only partially known by analogy to other similar systems at the time of the simulation. This method is nearly mathematically equivalent to adjusting the spacing between states along the path. As $\Delta\lambda$ between states is reduced, additional samples are placed in nearby state space, ideally in the regions of high variance. The switch optimization procedure adjusting $h_i(\lambda)$ is also an equivalent process to adjusting the spacing between states

since flattening $h_i(\lambda)$ results in additional sampling at the same intermediate states, also effectively increasing sampling at states with high sample variance. Given that the switch optimization procedure effectively adjusts the spacing between states, we will only look at uniformly spaced states along λ and instead distribute discrete samples to each state.

The other extreme case is simply performing uniform sampling at all states, which is a fairly common approach as it requires no additional knowledge beforehand. The realistic case is between these two situations, as some sense of the λ with large sample variance is often qualitatively known.

Finally, a fair comparison of statistical efficiency requires methods eliminating bias from free energy differences and analyzing only uncorrelated samples. Since each alchemical method and schedule are estimating the same transformation, the free energy differences should be within error of each other. If the free energy differences are not within error of each other, it indicates insufficient sampling and poor estimates at best, or fundamental implementation issues at worst. The value of the free energy differences in this chapter are correctly within statistical error of each other and tabulated in Table B.1 of Appendix B.2. This statistical efficiency comparison relies only on uncorrelated samples, so each trajectory is subsampled to extract only uncorrelated samples using the timeseries functions of the *pymbar* Python module [101]. The computational cost to draw uncorrelated samples from the different states will be addressed later in this chapter.

Comparison of Statistical Efficiency from Different Sampling Schemes

The maximum number of samples to distribute among alchemical states will be fixed for each schedule and alchemical method. The different sampling schemes will assume 21000 uncorrelated samples are distributed to each schedule for both limiting cases. This number was chosen so that 1000 samples can be uniformly distributed to the 21

sampled states in the soft core electrostatics case. Samples are distributed to a fixed number of states in each schedule. The basis function path has 101 evenly spaced states at each stage since we can estimate the variance at arbitrary λ values. [54] There is an overlapping state between each stage of a schedule so care must be taken not to double count these states, e.g. in the AR/C/E schedule, the thermodynamic state defined by $h_R(\lambda) = h_A(\lambda) = 1$ is identical to the state $h_C(\lambda) = 0$. The soft core electrostatics pathway will only be estimated at the 21 sampled states as sample variance data is only available through direct simulation at each state. Eq. (3.7) is valid even if the spacing is not uniform, though it will be held uniform in this chapter.

Both the AR/C/E-WCA and the R/C/AE-WCA paths have nearly the same statistical efficiency as the soft core electrostatics path within error when sampled proportionally with the sample standard deviation. Fig. 3.5 shows the variance of the calculation of free energy for each pathway for both uniform sampling (green, left most bars), and proportional sampling with the sample standard deviation (orange, right most bars) along with total error in the estimate of the variance of the free energy shown with the error bar at the top. The middle bars (purple) are a hybrid sampling scheme and will be discussed below. As expected, proportional sampling is more statistically efficient than uniform sampling, seeming to imply the sample variance must be known ahead of time for the basis function approach to be as efficient as soft core.

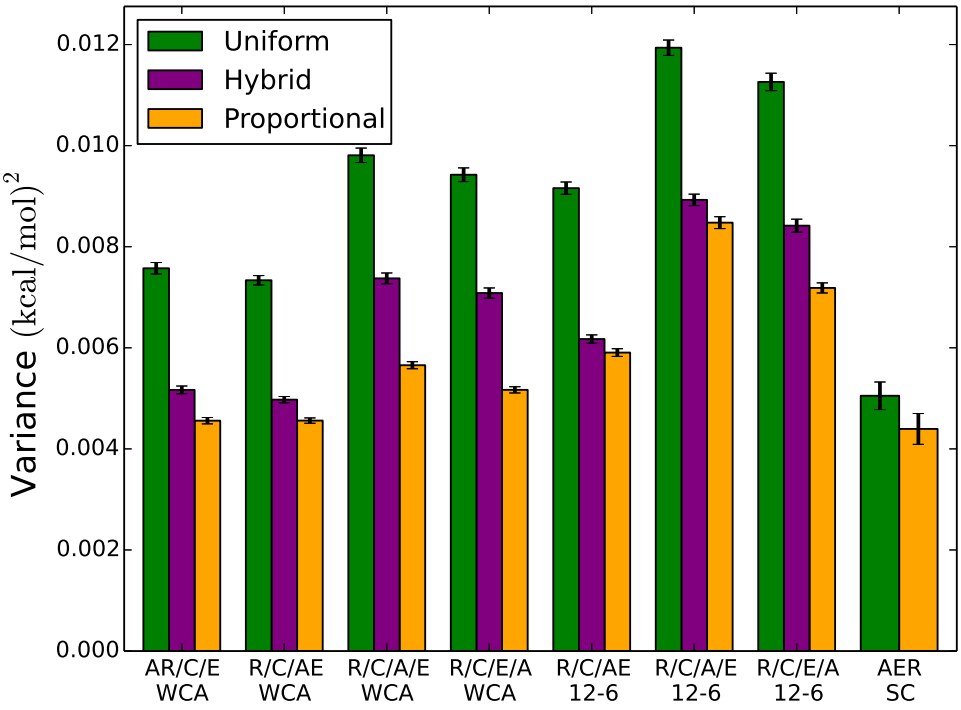


Figure 3.5: A hybrid sampling scheme with 3-step, WCA basis functions is as statistically efficient as soft core methods. The variance of the calculation for free energy is shown for three sampling schemes applied to all tested basis function schedules and the soft core path. Samples were distributed either uniformly, proportional to the sample standard deviation, or in a hybrid approach which is uniform except for the capping stage, where only endpoints are sampled. The hybrid scheme provides comparable variance of the calculation to uniformly sampled soft core methods. "WCA" and "12-6" labels distinguish the Lennard-Jones basis functions and "SC" soft core method applied to all forces with 1-1-6 parameterization.

However, most of this reduction in uncertainty can be gained by simply moving samples out of the capping stage to other stages. Because sampling at intermediates states of the cap contributes near negligible uncertainty as seen in Fig. 3.4, sampling only the endpoints allows distributing more samples into the repulsive and electrostatics stages. These endpoints also overlap with the AR, R, E, and/or AE stages and must be sampled anyways. In this "hybrid" sampling, the C step is only sampled at the endpoints, but other stages are uniformly sampled. This hybrid pathway lowers the variance of the calculation in free energy for the AR/C/E-WCA and R/C/AE-WCA pathways to nearly the same statistical value as the soft core electrostatics (shown

in Fig. 3.5) but without the need for *a priori* knowledge of all the sample variance curves. Sampling proportional to the standard deviation still provides lower variance of the calculation, but at most by 13% for the AR/C/E-WCA and R/C/AE-WCA pathways. Neglecting sampling the C stage does introduce a small amount of numerical integration error. However, the numerical bias introduced is less than 3% of the total variance of the free energy calculation, so makes statistically no change in the results.

The 12-6 and the four-step basis function schedules have significantly larger variances in the free energy than the three-step WCA basis function schedules. Because of this, there appears to be little benefit to splitting up the Lennard-Jones forces in the 12-6 basis unless the contribution from the individual dispersion and repulsion components is of interest to the study.

The larger variance of free energy calculation on the 12-6 schedules is due primarily to the uncapped attractive component, which contributes significantly more to the variance than the capped WCA attractive component. The attractive component in the 12-6 pathways will contribute more to the numerator in Eq. (3.7) for the uniform case, and reduce the number of samples available to the repulsive step, decreasing the denominator of Eq. (3.7) in the proportional case. Any of the four-step processes also suffer from having fewer samples available for the denominator in the repulsive step, which causes an increase in variance of the calculation as well.

The contribution of each stage to the variance of the calculation in Fig. 3.5 can be visualized in Appendix B.2 in Fig. B.8. The tabulated variance at each stage, free energy, sampled uncertainty, and numeric values for variance of the calculation from each sampling scheme are also included in Appendix B.2 in Table B.1 and Table B.2.

In summary, there are four heuristics we can make about sampling with the basis function approach. From these heuristics, a schedule and sampling scheme can be chosen without explicit knowledge of the basis function sample variances.

1. Couple attractive Lennard-Jones forces simultaneously with either the repulsive or electrostatic term.

2. A WCA decomposition of Lennard-Jones terms is somewhat more efficient than a 12-6 decomposition.

3. Only the end states of the capping stage need to be sampled.

4. Uniform distribution of states to every stage and uniform sampling of each state for all stages except for the capping stage will provide a low statistical error, approximately as low as a soft core pathway where all forces (including electrostatics) are coupled simultaneously.

3.4.3 Decorrelation times along the soft core pathway are somewhat longer than the basis function pathway

The cost of generating uncorrelated samples is an additional constraint that we must also consider when looking at the efficiency of a schedule. Some states might have intrinsically slower configurational sampling. An extreme case was noted by Pham et al., [84] where the provably lowest variance path actually resulted in extremely slow kinetics for transitioning between two dominant sets of allowed configurations, requiring tens or hundreds of nanoseconds. This resulted in a recommendation to sample a somewhat higher variance path with much faster conformational sampling between the resulting configurations, as the computational efficiency of this approach outweighed the statistical inefficiency.

The two limiting cases in the comparison of schedules in the previous section assumed that a fixed number of uncorrelated samples were available to distribute across states, but the simulation time required to generate those uncorrelated samples will be different depending on the conformational dynamics created by the potential function. To explore how the correlation time of the sampling affects the efficiency

of these schedules, we will look at the recommended AR/C/E-WCA and the soft core electrostatics paths, using ideas from the difficulty index method introduced by Schultz and Kofke for comparing methods accounting for statistical uncertainty and computational time. [79]

The difficulty index method proposes a difficulty, D, to compute a given property as

$$D = t^{1/2}\sigma \tag{3.8}$$

where t is a specified required time to compute a property, and σ is the uncertainty in that property, after time t. The difficulty index is the logarithm of the difficulty, scaled by a normalizing factor to compare the net computational effort required to compute different properties, accounting for both statistical efficiency and time to calculate of the property. As the same property is being compared here, only the difficulty, not the difficulty index, will be examined. σ in this case will be the standard deviation of the free energy, computed as the square root of the values in Fig. 3.5, tabulated in Appendix B.2 as Table B.2. Assuming that identical hardware and software is used to generate these samples, the only difference between time to collect an uncorrelated sample is the correlation time. This removes any differences in CPU-clock time from different software, enhanced sampling method such as replica exchange, and hardware such as different processors or GPU vs. CPU. For this chapter, the units of difficulty will be $\text{ps}^{1/2}\,\text{kcal/mol}$.

Correlation times were calculated by running a NVE simulation of both the AR/C/E-WCA basis function pathway, and the soft core electrostatics pathway. Using NVE simulation removes any effect temperature or pressure coupling has on the correlation time, [146] which is particularly important since GROMACS and OpenMM are being compared, and have somewhat different implementations of the thermostats used for these simulations. Since energies computed by both simulation packages is identical to machine precision in NVE, the differences between correlation times will

only be due to the different pathways in an NVE simulation.

The AR/C/E-WCA simulation, selected as one of the two best basis function-based paths and simulated in OpenMM, was run without replica exchange to match the soft core electrostatic simulations run in GROMACS. The autocorrelation time [143] of the $\partial u/\partial\lambda$ time series was computed for each method. The autocorrelation time of the $u(r,\lambda)$ time series in the fully coupled and fully decoupled states should be the same between both OpenMM and GROMACS simulations, and is path independent, so it was used as a validation check. Because sampling was only done at a limited number of states, the correlation for the basis function time series is shown as a total progress of the entire coupling process, and not separated by individual stages. Because there will be two values of $\partial u/\partial\lambda$ at a state where basis function stages overlap, the maximum autocorrelation time of the two was chosen to be conservative.

More sampling is required near the decoupled state of the soft core path relative to the basis function path for a practical implementation as shown by Fig. 3.6. The correlation times at the end states for the $u(r,\lambda)$ (bottom) time series agree within 10% showing the two simulation packages are generating comparable dynamics, as they should. At most intermediates, the correlation times in the two pathways are similar. However, the correlation time of $\partial u/\partial\lambda$ (top) for the soft core electrostatic path is almost four times that of the basis function path at the decoupled state. Since the remainder of the correlation times are comparable, this state at $\lambda = 0$ will be used to compute difficulty. From Eq. (3.8), the difficulty of the soft core path near the decoupled state is 0.38 whereas the basis function difficulty is 0.12. Although the decoupled state may take 3.5 times additional CPU effort for the soft core method, the remainder of the sampled states will take roughly equal CPU effort between the soft core and linear basis function approaches. This increased computational effort at the end state will result in a moderate overall increase in simulation time for simulating with the soft core approach instead of the basis function approach.

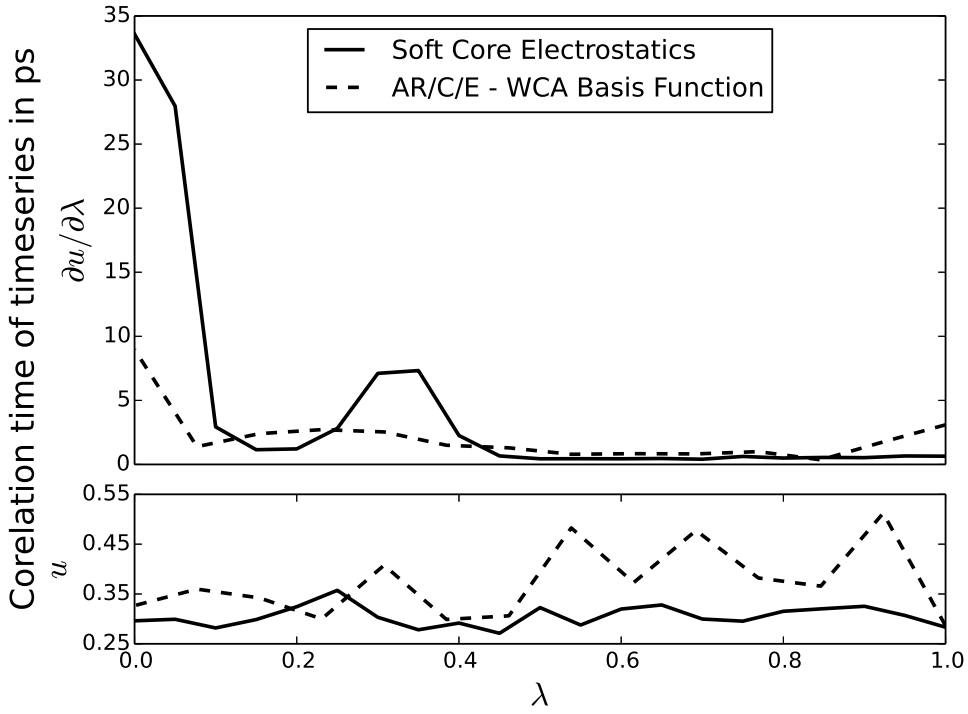


Figure 3.6: The basis function method has lower correlation times than soft core electrostatics. Correlation times of soft core electrostatics (solid lines) and a AR/C/E-WCA schedule (dashed lines) sampled from NVE simulations are shown. The basis function curve shows all three stages along a single λ parameter. Top image show the correlation time for $\partial u/\partial\lambda$ and bottom show the correlation time for $u(r, \lambda)$, which allow us to validate that the simulation packages are generating the same dynamics at the coupled and decoupled states. The decoupled state for the soft core electrostatics would require nearly four times as many samples relative to the basis functions to generate the same number of uncorrelated samples. Correlation times for $u(r, \lambda)$ and $\partial u/\partial\lambda$ are roughly equal for $\lambda > 0.5$ and are shown as separate plots for clarity. Lines are drawn between discretely sampled states and serve to guide the eye only.

Finally, and most importantly, we note that the correlation time for calculations of complex systems, such as protein-ligand binding, is dominated by the large scale motions of the system, which are independent of the much shorter timescale of these alchemical changes. Reducing the variance of the path reduces the number of uncorrelated data points that must be collected for a given accuracy of calculation, where the correlation times are those of the longest timescale motions of interest in the system, which are usually not the ones involving the alchemical transformation.

3.5 Conclusions

We have incorporated and tested electrostatics into our linear basis function formalism. Alchemical electrostatic methods in their current form are easily included in the basis function formalism when the charge is coupled after Lennard-Jones type interactions. We have shown in this chapter the most statistically efficient switch for charging is nearly linear, and provides too marginal (5% maximum) a reduction in variance for the pathway (and therefore statistical uncertainty) too small to justify the complexity of implementing improved switches. Simply turning on charges linearly appears to be the most effective option in essentially all cases.

We recommend either the AR/C/E-WCA or R/C/AE-WCA basis functions with linear charging be considered for future implementations of alchemical transformations. We have carefully examined multiple basis function coupling methods and compared them to soft core electrostatics in terms of statistical and computational efficiency. Several heuristics that do not rely on explicit knowledge of the variance were provided which should help those wishing to implement the basis function method. The overall statistical efficiency for the basis function with the heuristics is comparable to traditional soft core methods, however, the basis functions are slightly more computationally efficient. The basis functions have the added benefit of being able to remove the alchemical forces from the inner force loop of the simulation and quickly compute energies at unsampled states, as well as avoiding code to explicitly compute the analytical derivative with respect to the coupling parameter.

Chapter 4

Applying the Basis Functions to Sample a Large Chemical Space

4.1 Introduction

This chapter has previously been published [54] as: Naden, L. N.; Shirts, M. R. Rapid Computation of Thermodynamic Properties over Multidimensional Nonbonded Parameter Spaces Using Adaptive Multistate Reweighting, *J. Chem. Theory Comput.*, 12:1806-1823, 2016.

Many applications of molecular simulations require searching over large parameter spaces to predict or match physical observables. Molecular simulation parameters such as charges, Lennard-Jones dispersion and repulsion parameters, as well as bonds, angles, and torsion force constants determine the energies and probabilities of configurations in simulations, and thus in turn, determine what thermodynamic properties will be observed. The ability to accurately estimate thermodynamic properties without the need for laboratory experiments has the potential to save both time and resources in fields such as polymer [147] and solvent [148] design as well as drug discovery. [12–14] This time and cost savings is important both in the design of new molecules, where properties are unknown, and 'reverse property prediction' where a model or molecule is designed to match specific experimental targets, such as designing metal-organic frameworks (MOFs) with specific gas loadings. [149]

Assigning parameters in a molecular simulation to fit experimental data becomes more difficult as the number of free parameters increases. Some experimental parameters, such as bond lengths and angles, are relatively easy to estimate using small-molecule crystallographic structures and quantum chemistry. However, nonbonded parameters, such as partial charges and dispersion terms, are much more difficult to assign as they are model parameters that do not directly correspond to laboratory observables. Instead, possible nonbonded parameters are constrained by sets of experimental observables such as transfer free energies, heats of vaporization, densities, and heat capacities. Assigning nonbonded parameters is even more difficult for coarse grain models since no single atoms correspond to the coarse grain beads,

making parameters much less transferable.

Identifying nonbonded model parameters consistent with a set of experimental thermodynamic values requires expensive iterative, self-consistent simulations, gradient optimizations, [150] or quantum mechanical simulations. [71, 151–153] Most condensed phase force fields [71–73, 75, 76, 154, 155] are parameterized by iterative fitting to a training set of experimental thermodynamic data for a small set of molecules chosen to represent a broader spectrum of similar molecules. [156, 157] Accurate fits are required to predict properties of biological systems or complex mixtures where group contribution methods such as UNIQUAC [65] and UNIFAC [66] are inadequate.

Some of the most computationally expensive properties to estimate involve the free energy differences between two states, such as the solvation free energy, which is the free energy difference of two systems as one solute molecule moves from solution to vapor, or activity coefficients, which measure the deviation of the chemical potential of a species from ideality. Accurately computing free energy differences (or equivalently, chemical potential differences) also provide a way to compute many other thermodynamic properties as they can be derived from the derivatives of the free energy with respect to temperature (T), pressure (P), volume (V), and number of particles (N_i).

Estimating the free energy difference between two thermodynamic states accurately requires designing a thermodynamic path between the states. Thermodynamic paths connect two or more states through a series of smaller steps along a parametric path whose free energy can be calculated more easily. [95, 158]

Paths that are both computationally and statistically efficient, especially for full deletion or insertion of a molecule into a dense fluid, are nontrivial to design and must often include a number of non-obvious, nonphysical intermediates. [54, 55, 57, 81, 84–90, 92, 95–97] The end states and multiple intermediate states along the thermodynamic path must be sampled to create good configuration space overlap between the end states, which is required to accurately estimate free energy differences

between the endpoints. [14, 81, 105, 159–161] Technically, we require good overlap in the full phase space, both configurations and velocity, but because velocities are thermalized in essentially all systems of thermodynamic interest, it is the configuration space that we must generally worry about when connecting states together.

If one wishes to compute free energies of solvation for many different parameterizations of the same molecule, the direct approach of estimating properties by sampling along a thermodynamic pathway connecting all parameter choices is extremely expensive for highly accurate calculations. Accurately calculating the differences in free energies due to a small change of parameters is particularly difficult because we must take the differences between two similar numbers with independent statistical error.

Reweighting methods can help solve both the problem of expense and the problem of cancellations of errors. In a recent study, our group showed how multistate reweighting can directly calculate the $\Delta\Delta G$ between two different long-range interaction approaches, with very small uncertainties for relatively low computational cost. [162] The expense is lowered by constructing thermodynamic cycles directly connecting Hamiltonians with similar parameters and thus significantly overlapping configurational spaces. These small uncertainties are possible because multistate reweighting methods such as the Multistate Bennett Acceptance Ratio (MBAR) [101] can directly calculate the covariances between the two free energies through analyzing all potential energy differences, rather than only uncorrelated calculations. This same approach can be applied to small changes in parameters.

Estimating properties with reweighting methods requires constructing a thermodynamic path between the different parameterizations. It also requires potentially significant computational resources to perform simulations with parameters that have properly overlapping configurations. The combination of these two requirements adds theoretical and practical limitations to simultaneously searching large, multidimensional nonbonded parameter spaces.

1. The space of nonbonded parameters is often at least multiple dimensions *per particle* or particle type. For example, the nonbonded parameters of charge (q) and at least two Lennard-Jones-like terms (ϵ_{ij}, σ_{ij}) can result in at least three parameter dimensions per particle type.

2. There is no obvious way to define computationally efficient thermodynamic paths between any two points in these multidimensional spaces or select *a prior* simulation points in this space that give rise to low error estimates of thermodynamic properties across the entire space.

3. Reweighting methods requires computing energies from the sampled configurations to other sampled states, and any unsampled state of interest. This re-computation typically requires re-running the simulation force loops over all generated configurations for each combination of parameters of interest. The computational cost to search such a multidimensional space of nonbonded parameters scales, at best, linearly with the number of samples, and at worst quadratically with the number of parameter combinations, since data may be collected with simulations at each parameter combination. [54]

Designing efficient thermodynamic paths through arbitrary thermodynamic states is a challenging task, [54, 55, 84] but designing paths in multidimensional parameter spaces adds additional complexities. An example in Fig. 4.1 demonstrates the challenges of identifying low-uncertainty paths in multiple dimensions. This figure shows two arbitrarily defined thermodynamic states in a two-dimensional parameter space and attempts to draw pathways between them with high mutual configuration space overlap, providing low error estimates. However, the choice of which path to sample is not immediately obvious. The shortest Euclidean path in parameter space has large uncertainty, but two alternative paths have low uncertainty. This sort of multidimensional space raises questions with no obvious answers: How can we *a priori*

identify which paths have more mutual configuration space overlap, and consequently result in simulations with lower uncertainty, without exhaustively sampling the system? Could samples drawn from both paths but with different proportions provide lower uncertainty than sampling either path by itself?

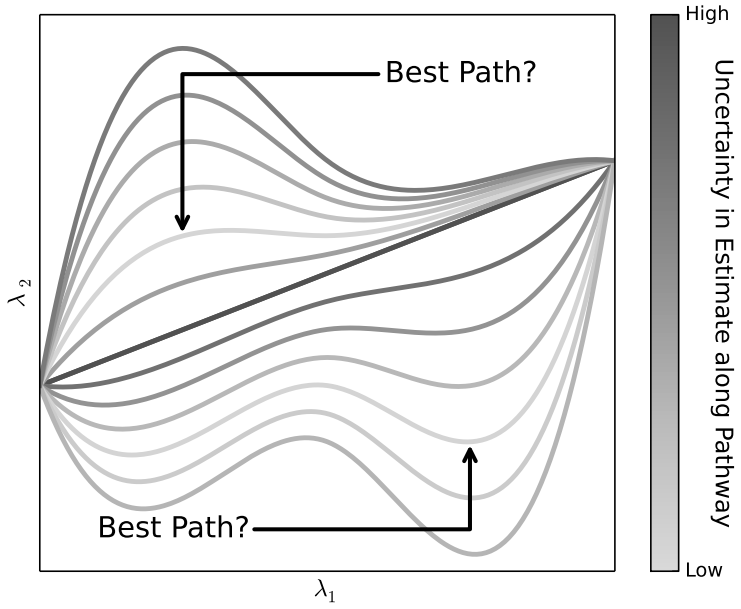


Figure 4.1: **Defining the "best" thermodynamic path between arbitrary states is a nontrivial problem.** This figure shows several multidimensional thermodynamic paths connecting two states, and the relative uncertainty of each pathway shown by a gray scale gradient of each curve. The path with shortest Euclidean distance in parameter space (λ_1, λ_2) has a large uncertainty, and at least two of the other paths have lower uncertainty. However, it is unclear if sampling a single "best path," or a combination of multiple low variance paths will have the highest computational efficiency for a target statistical error. The most computationally efficient sampling scheme may not be along any single path at all, and instead may be achieved by sampling non-parameterized states in a multidimensional parameter space.

Previous research on identifying low-variance paths along a single coupling parameter used local minimization of total variance along the path. [54, 55, 84] However, in multiple dimensions, since multiple potential low-variance paths could exist, local optimization is unlikely to identify the most efficient path between arbitrary states, nor whether it would be more optimal to sample multiple paths. The identification of

an optimal path choice is made more complicated if we are interested in all the mutual free energy differences between multiple states, since we must identify a path or a network of paths connecting all of the states, with samples collected along each path. One would ideally want ad hoc rules estimating the computational efficiency of these paths, determined *a priori* to avoid unnecessary sampling. Defining such rules for a diverse set of chemical systems becomes increasingly complex as the dimensionality and the number of states increases. Removing the need to define these difficult multidimensional paths significantly lowers these barriers to searching through large parameter spaces for optimal parameters.

We can remove the need to explicitly define or sample along thermodynamic paths between states in multiple dimensions with *multistate* reweighting methods. These methods, such as the MBAR [101] and the Weighted Histogram Analysis Method (WHAM), [99, 100] allow samples taken from anywhere in the parameter space to contribute to the estimate of properties anywhere else in the space, without the need to define which thermodynamic states are adjacent as is needed for the original Bennett Acceptance Ratio (BAR). [105] MBAR has the important advantage over WHAM in that sampled configurations do not need to be binned along a pre-defined thermodynamic path for the analysis which is particularly important in multidimensional spaces, where it becomes increasingly difficult to populate histogram bins. Any given combination of nonbonded parameters will share a high degree of configuration space overlap with slightly perturbed parameters, though it is not clear *a priori* how large the perturbation can be. MBAR uses the probability of observing samples drawn from one state in any other state to estimate the effective ratio of partition functions between any two states, with or without samples, to provide free energy differences with uncertainty estimates. Therefore, with moderate sampling at the right parameter combinations, we can estimate thermodynamic properties at a large number of parameter combinations, even without explicit sampling of every

parameter combination, so long as for each choice of parameters there is some degree of configuration space overlap with some combination of sampled parameters.

With the limitation of explicitly defining a thermodynamic path removed, we can focus on decreasing the computational cost of computing the energy of every drawn sample in every sampled state and unsampled state we care about. These energies are required to compute the probability of every sample in every state. We define "state of interest" as a shorthand for any thermodynamic state that we want to estimate thermodynamic properties at. Each thermodynamic state is defined here by the parameter combination and thermodynamic ensemble. Reweighting methods require the energy of each sampled configuration to be known at each state of interest to compute free energy differences and other properties between states. Computing a configuration's energy multiple times with different energy functions is a time limiting step for calculating thermodynamic properties across large parameter spaces. Collecting the energy of each configuration at each state usually requires running the simulation code, or at least the inner force loop of simulation code, multiple times on each configuration. If we only are interested in reweighting the properties from a single sampled state, [98] we would only have to carry out this computation once per configuration (N) drawn at a sampled state (K_s), per unsampled state of interest (K_u). This results in energy calculations that scale as $\mathcal{O}\left(N(K_u + K_s)\right)$, assuming equal sampling per sampled state. However, such estimates are known to be statistically inefficient and prone to substantial bias when configuration overlap is not substantial. [60, 102, 163] Multistate methods, like MBAR, [101] require calculating the energy for each configuration at each sampled state as well as each state of interest, so the scaling is quadratic in the number of sampled states as $\mathcal{O}\left(N(K_u + K_s^2)\right)$. This is not a burdensome task for a small number of states along a single thermodynamic path, making it well worth using multistate simulations regardless of how expensive recalculating the energies of the configurations are. However, as tens or hundreds of

thousands of states of interest and hundreds of sampled states are considered, this scaling becomes a computational bottleneck that must be overcome.

We can reduce the cost to compute energies at thermodynamic states to computationally trivial vector multiplication by defining the energies using linear basis functions. [54, 55, 88] Energies calculated using a basis function approach can be most generally written as

$$u(r, \lambda) = \sum_{i}^{n} h_i(\lambda) u_i(r) + u_{\text{unaffected}}(r) \tag{4.1}$$

where $u(r, \lambda) = \beta U(r, \lambda)$ is the reduced energy (the energy scaled by β) as a function of both the configuration r and some (possibly multidimensional) alchemical coupling parameter λ; $h_i(\lambda)$ are a set of nonphysical, alchemical switches that are independent of configuration; $u_i(r)$ are the basis functions; n is the total number of basis function and alchemical switch pairs; and $u_{\text{unaffected}}(r)$ is the system's potential energy not dependent on the alchemical variables. This approach computes the energy of a configuration at any thermodynamic state by scalar multiplication of the configuration dependent basis functions, which only have to be computed once per configuration. The vector multiplication can eliminate the need to run the inner force loop on a configuration more than once, reducing the computational cost of evaluating energies from $\mathcal{O}\left(N(K_u + K_s^2)\right)$, to $\mathcal{O}\left(NK_s\right)$, which is simply the total number of sampled configurations. The alchemical switches can take any form, so long as r remains part of the basis functions and not the switches. This form of the potential energy is in contrast to forms such as the soft core form of Lennard-Jones interactions [82, 83], which cannot be represented as sums of separable combinations of r and λ.

In this chapter, we combine multistate reweighting methods with a linear basis function approach to compute thermodynamic properties over a large nonbonded parameter space. To demonstrate the process, we look at the Lennard-Jones parameters

ϵ_{ii} and σ_{ii}, and partial charge, q_i, for a single particle in explicit solvent. This approach could can aid in future large parameter space searches to quickly find a range of nonbonded parameters and fine tune a fitting or optimization procedure. The relative free energy, enthalpy, and entropy of solvation are explored as these are some of the most computationally expensive properties to estimate. We also estimate the Born solvation free energy of charging, compare our results to specific ion free energies computed from others, and compute radial distribution functions. The techniques shown here are generalizable to other thermodynamic properties. Single particle solvation is intended as a demonstration of the approach. The same approach could be used for problems as general as exploring the nonbonded parameters of multiple atomic sites for improved force field parameterization or exploring coarse-grained potentials so long as the potential energy can be represented as Eq. (4.1).

The techniques presented provide a novel alternative to standard methods for searching parameter space. If one merely requires a single optimal set of parameters under some criteria, the approach here will not be as useful as a standard optimization method. However, this approach allows researchers to estimate properties over an entire parameter space at once, eliminating the need to perform additional simulations to compute the free energy or other thermodynamic properties at different parameter choices. This approach makes it easy and efficient to probe theories which make predictions over a wide parameter range, and are thus not easily amenable to brute force calculations. We demonstrate this efficiency in a comparison of Born solvation theory to explicit solvent simulation.

We explore both a two-dimensional and a three-dimensional parameter space. We first estimate free energies in a 2-D parameter space of ϵ_{ii} and σ_{ii} to demonstrate the ability to estimate properties in a wide parameter space, easily visualize the results, and identify how reducing uncertainty in the property estimates is non-trivial. Secondly, we estimate multiple properties in a larger, 3-D parameter space in ϵ_{ij},

σ_{ii}, and q_i and iteratively generate new simulations to reduce the statistical error in the estimate of solvation properties across the entire range of parameters. This 3-D parameter search looks at a more practical parameterization with particle charge, but focuses on the issues of reducing uncertainty in the entire parameter space through adaptively drawing samples requiring minimal user input.

4.2 Theory

The notation in this chapter is as follows. σ_{ii} and ϵ_{ii} are the Lennard-Jones parameters, and q_i is the charge of a particle. ϵ_0 is the permittivity of vacuum. $u(r)$ are basis functions of the potential energy function and $h(\lambda)$ are the alchemical switches. Subscripts E, R, and A denote an electrostatic, Lennard-Jones repulsive, and Lennard-Jones attractive term respectively, e.g. $u_E(r)$ is the electrostatic basis function. The subscripts i, j, and k on nonbonded parameters denote arbitrary atoms, and subscripts X, Y, and Z denote an explicit set of parameters with fixed values which define a thermodynamic state, but their values not explicitly defined to be general. Subscript S denotes a solvent particle. The subscript ℓ will be used for summation indices, and C represents a collection of constants.

4.2.1 Representing nonbonded parameter space with basis functions

We generalize the potential energy to simplify writing the energy of any potential in multidimensional space. The three nonbonded parameters explored here lead to a pairwise nonbonded potential energy between two point particles a distance r apart

$$u(r) = \beta \left[\frac{4\epsilon_{ij}\sigma_{ij}^{12}}{r^{12}} + \frac{-4\epsilon_{ij}\sigma_{ij}^{6}}{r^{6}} + \frac{q_i q_j}{4\pi\epsilon_0 r} \right]. \tag{4.2}$$

Eq. (4.2) can be more generally written as

$$u(r) = \frac{C_{12}}{r^{12}} + \frac{C_6}{r^6} + \frac{C_1}{r} \tag{4.3}$$

$$= \sum_{\ell} \left(\frac{C_n}{r^n} \right)_{\ell} \tag{4.4}$$

where n takes discrete values of 12, 6, or 1 depending on the index of ℓ and each C_n corresponds to the power of r^{-n}. The energy of a configuration at any point in parameter space is found by adding an alchemical switch, $h_n(\lambda_n)$, to each term of Eq. (4.3) and then adding the remainder of the pairwise interactions not affected by the alchemical changes. Each λ_n can vary independently from each other, allowing a multidimensional representation of the energy in terms of the parameters. The alchemical switches scale each of the 12, 6, and 1 terms to produce each of the target thermodynamic states. The total potential is then

$$u(r, \lambda) = u_{\text{unaffected}}(r) + \sum_{\ell} \left(\frac{h_n(\lambda_n) C_n}{r^n} \right)_{\ell}. \tag{4.5}$$

Computing the basis functions can be done either directly in code or in post-processing with fixed reference states. The most computationally efficient way to compute the basis functions would be to have the simulation package provide them at run time. However, most simulation packages will not allow the user direct access to the basis function values without heavily modifying its code, since usually only the total potential energy or the total λ-dependent energy is required. An alternative solution which avoids any code modification is to choose two fixed reference states and compute the basis functions as a difference in energy, as was done in this chapter and explored below. This alternative approach means we must run the force calculations at least three times for any sampled configuration: once while the samples are generated, and once for each reference state.

The potential can be represented as linear combination of alchemical perturbations

around a fixed reference particle at state X with respect to a second reference particle at state Y as

$$\begin{aligned} u(r,\lambda) &= u_{\text{unaffected}}(r) + \sum_{\ell} \left[\frac{(1-h_n(\lambda_n))C_{n,X}(r) + h_n(\lambda_n)C_{n,Y}(r)}{r^n} \right]_\ell \\ &= u_{\text{unaffected}}(r) + u_X(r) + \sum_{\ell} \left[\frac{h_n(\lambda_n)\Delta C_{n,XY}(r)}{r^n} \right]_\ell \end{aligned} \tag{4.6}$$

where $\Delta C_{n,XY}(r) = C_{n,Y}(r) - C_{n,X}(r)$ and $u_X(r)$ is the complete nonbonded pairwise potential for particle X alone. The computed basis functions are then calculated as the energy difference between the two reference particles, and the unmodified potential energy of particle X becomes part of $u_{\text{unaffected}}(r)$. The potential energy at arbitrary state Z can now be computed using this perturbation. This reference state approach makes computing the basis functions possible without major simulation code changes. The numerical error should be monitored for any round off error since two similar energies are subtracted, as we discuss in Section 4.4.5.

Unlike with standard alchemical transformations between $\lambda = 0$ and $\lambda = 1$, the accessible parameter space is not bounded by the reference states. Consider an arbitrary state, Z, with parameters outside the range of the parameters $C_{n,X}$ and $C_{n,Y}$. The values of alchemical switches defining Z would then fall outside the standard $[0, 1]$ domain. States which fall outside this domain still have physical meaning in this context, unlike states with λ outside $[0, 1]$ have no meaning for particle insertion or deletion simulations. For example, the expanded domains in Chapters 2 and 3 served no practical purpose since $h_n(\lambda_n) < 0$ represented a state where particles had an attractive atomic center, and $h_n(\lambda_n) > 1$ represented a state "more than fully coupled."

The number of terms in Eq. (4.6) will increase quadratically as the number of interaction sites in the solute increase, increasing the number of ϵ_{ij} and σ_{ij} terms. However, geometric mixing rules can avoid such a large increase in terms. Details of

how geometric mixing rules allow $h_n(\lambda_n)$ terms to be solvent-independent are given in supplementary material in section C.1.

The perturbed energy representation of the thermodynamic space is more computationally efficient than re-running simulation force loops. The optimal computational cost for this perturbed energy function is only $\mathcal{O}(3NK_s)$, which scales much better than $\mathcal{O}(N(K_u + K_s^2))$ as K_u and K_s increase. We note that the $\mathcal{O}(3NK_s)$ computational effort is only optimal if one can separate the individual terms in the potential. This is sometimes possible if, for example, the energy terms are one nonbonded term and one bonded term. Simulation packages like GROMACS [35, 38] return the bonded and nonbonded contributions separately. In our work here, our true computational effort is $\mathcal{O}((1+2B)K_sN) = \mathcal{O}(7K_sN)$ where B is the number of basis function terms since we must solve for each basis function through linear algebra. We also perturb only one atom type, keeping computational cost low. The computational costs increases roughly proportionally with the number of perturbed atom types. We emphasize that the $\mathcal{O}(7K_sN)$ scaling, even when considering the number of perturbed atom types, is still less computational effort than $\mathcal{O}(N(K_u + K_s^2))$ as we are designing for hundreds of thousands of unsampled states.

The methods outlined here should be applicable for more generalized parameter dimensions and thermodynamic spaces so long as the energy can be written as a linear combination of basis functions as in Eq. (4.1). The case of appearance and disappearance of multiple atoms or molecules in a system traditionally is done with soft core energy functions. [82, 83] However, we detail in our previous work basis functions for particle removal which are as statistically efficient as soft core paths. [54, 55] Since we are not appearing or disappearing atoms in the dense fluid, we do not need the soft core basis functions and can use the perturbed Lennard-Jones and Coulombic terms of Eq. (4.6).

4.3 Experimental Design

Molecular dynamics (MD) simulations of a single particle in 1195 TIP3P water molecules were carried out with GROMACS 4.6.5 [35, 38] compiled in double precision. NVT equilibration was carried out for 100 ps in a 36.238 nm^3, followed by NPT equilibration for 500 ps, followed by NPT production simulations of 6 ns per simulated parameter combination. Temperature was held at 298 K and coupled through Langevin dynamics with a time constant of 5 ps. Pressure (for NPT simulations) was held at 1 atm and coupled with a Parrinello-Rahman barostat, [140, 141] with a time constant of 5 ps, and a compressibility of $4.5 \cdot 10^{-5}$ bar^{-1}.

Solvation properties were estimated over a grid of nonbonded parameters for the particle. For the 2-D case, the parameter ranges are $0.0239\,\text{kcal/mol} \leq \epsilon_{ii} \leq 0.8604\,\text{kcal/mol}$ ($0.1\,\text{kJ/mol} \leq \epsilon_{ii} \leq 3.6\,\text{kJ/mol}$) and $0.25\,\text{nm} \leq \sigma_{ii} \leq 1.2\,\text{nm}$. This range was chosen to include the largest possible particles in the OPLS-AA force field. [71, 72] with additional parameters to test the limits of the reweighting methods. Solvation properties were calculated on a square grid of ϵ_{ii} and σ_{ii} with 151 grid points in each dimension for 22,801 total parameter combinations. Grid points and initially sampled states were distributed uniformly in ϵ_{ii} and uniformly in σ_{ii}^3 so that sampling was done approximately proportional to the free energy of cavitation. [164, 165] The eleven initial sampled states were at $\sigma_{ii} = \{0.250, 0.573, 0.712, 0.811, 0.891, 0.958, 1.017, 1.070, 1.118, 1.162, 1.200\}$ nm with $\epsilon_{ii} = \{0.0239, 0.0502, 0.0765, 0.1028, 0.1291, 0.1554, 0.1816, 0.2079, 0.2342, 0.2605, 0.2868\}$ kcal/mol. Relative solvation properties were computed from the reference parameters $\epsilon_{ii} = 0.1816\,\text{kcal/mol}$, $\sigma_{ii} = 1.0170\,\text{nm}$ so the reference was roughly in the middle of the σ_{ii}^3 space. An additional state was drawn to reduce overall uncertainty at $\sigma_{ii} = 0.300$ nm and $\epsilon_{ii} = 0.1921$ kcal/mol.

For the 3-D case, the parameter ranges are $0.0239\,\text{kcal/mol} \leq \epsilon_{ii} \leq 0.8604\,\text{kcal/mol}$, $0.25\,\text{nm} \leq \sigma_{ii} \leq 0.958\,\text{nm}$, and $-2.0 \leq q_i \leq +2.0$ in units of elementary charge with each dimension having 51 points for 51^3 grid points in the parameter space and a

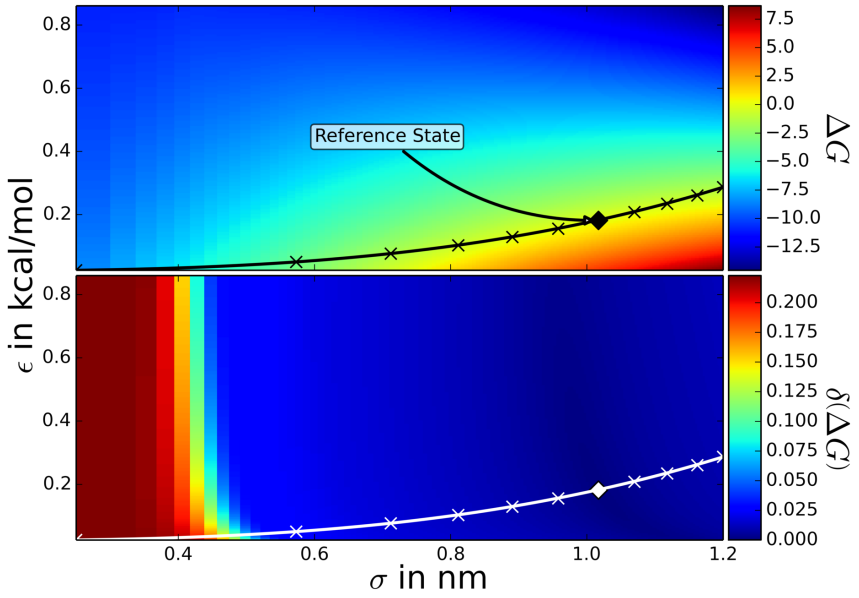
total of 132,651 parameter combinations. To improve resolution in some of the images, 101 uniformly spaced σ_{ii} states were estimated for $101 \cdot 51^2$ grid points for 262,701 combinations. The reference state chosen for this set test was $\epsilon_{ii} = 0.0502\,\text{kcal/mol}$, $\sigma_{ii} = 0.5732\,\text{nm}$, and $q_i = 0.0$. This set covers particles in the OPLS-AA force field from hydrogen (bound to a carbon), through the largest ions. The reference state was chosen to show how properties with low uncertainty can be estimated to particles of very different sizes and charges through iteratively selecting new parameter combinations to simulate. The spacing for initial sampling for ϵ_{ii} and σ_{ii} remains unchanged from the 2-D case and the initial sampled states at $q = 0$ were the first six points from the 2-D case plus the additional state to reduce uncertainty. These initial points were chosen for the $q = 0$ plane as we knew these initial points would provide low uncertainty estimates for the uncharged particles. Additional sampling in q_i was done proportional to $q_i^{1/2}$ in keeping with Born theory for the free energy of solvation of charged spheres. This choices resulted in initial sampling at charges ± 2.0000, ± 1.8516, ± 1.6903, ± 1.5119, ± 1.3093, ± 1.0690, ± 0.7559, and 0.0000, all with the reference state choices of ϵ_{ii} and σ_{ii}. Starting molecular geometries were generated with AMBERTOOLS's LEaP [112] and initial equilibration was carried out with the reference state parameters. All other solutes started their equilibration process from the final frame of the reference ion's NPT equilibration step.

Details about specific algorithm to compute basis functions with GROMACS and all input files are included in Appendix C and the online supplementary information for this article. [56] The analysis code can be found on GitHub. [166]

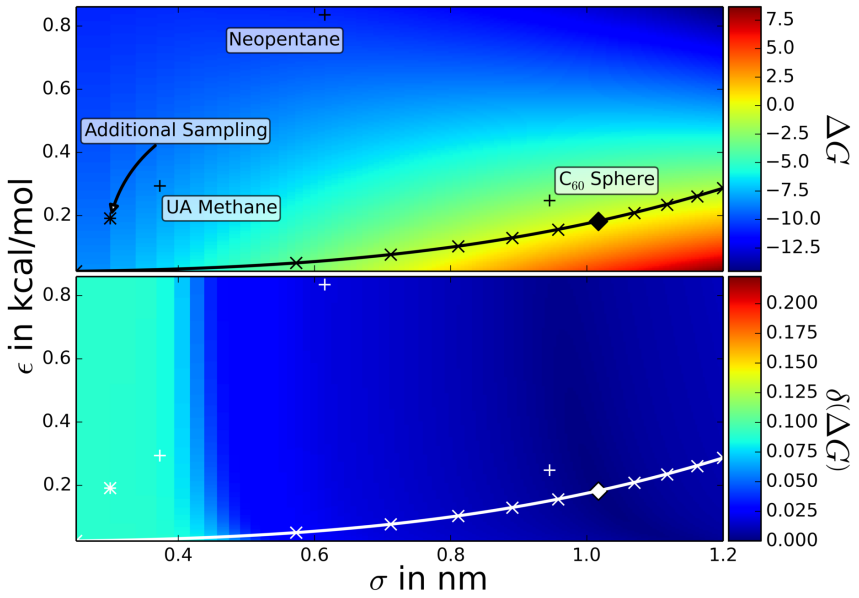
4.4 Results and Discussion

4.4.1 Solvation properties over a 2-D parameter space

With the combination of methods described above, we can efficiently and accurately calculate the free energy of solvation and other thermodynamic properties over multidimensional parameter spaces. Figure 4.2 shows the free energy, and error in free energy of uncharged Lennard-Jones spheres evaluated at 151^2 combinations of ϵ_{ii} and σ_{ii}. The free energy differences were estimated using MBAR implemented in the `pymbar` package. [101] One of the main keys to making this calculation feasible is that the linear basis function approach allows rapid calculation of potential energies in post-processing. Reconstructing the potential energies required for free energy estimates through vector operations takes only seconds on a single core of a desktop computer's CPU. The same evaluation of energies would have to be run through the inner force loops at all 151^2 states without the linear basis function method, scaling as $\mathcal{O}\left(N(K_u + K_s^2)\right)$. We ran each sampled state's trajectory through single point energy calculations with GROMACS estimating the potential under every other sampled state thermodynamic conditions to quantify the computational cost. Each simulation of 30000 samples took over 1500 CPU seconds to re-evaluate the energies of the given trajectory. For reference, the average time to run the simulation on the same hardware was 25 CPU hours. If we make a conservative calculation and assume each re-run of the inner force loop took the minimum 1500 CPU seconds, the 12 sampled states would have taken 13 CPU years to run each configuration through the 151^2 parameter combinations. This computational cost illustrates the primary speed improvement over re-running the inner force loop code, since time required to collect samples and estimate free energies is not affected by how the potential energies are computed in post-processing.



(a) Original 11 sampled states



(b) Naive additional sampling at 12 states and spot checks of Lennard-Jones spheres

Figure 4.2: Caption on following page

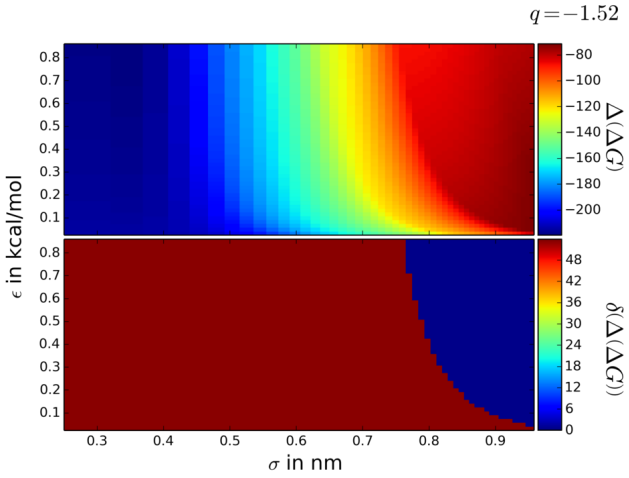
Figure 4.2: **The free energies at any combination of nonbonded parameters can be predicted in post-processing and regions of large uncertainty can be quickly found**. Shown is the free energy (top panels) and statistical error in the free energy (bottom panels) for 151^2 parameter combinations of ϵ_{ii} and σ_{ii} with no charge. Samples were drawn at locations shown by an X and the drawn line is to guide the eye. The free energies are all relative to the reference state shown by the diamond. (a) shows the estimates drawing samples from 11 states only. A large region of uncertainty is seen at small σ_{ii}. (b) A 12^{th} state was sampled in the region of high uncertainty, reducing the uncertainty in the whole region. The relative free energy of solvation was compared to chemically realistic spheres through soft core particle insertion simulations and is within error of the $\delta\Delta G$. Free energy is shown in units of kcal/mol.

Estimating properties in the 2-D parameter space, regions of large uncertainty can quickly be identified by visual inspection, and we can draw additional samples to reduce uncertainty. Fig. 4.2a shows the estimates of the solvation free energy when sampling from only 11 equivolume spaced states. We can see in the figure that parameter combinations with roughly $\sigma_{ii} < 0.5\,\text{nm}$ have high estimated uncertainty with respect to the reference state. We can naively sample by a single additional state in this region which drastically reduces the error in our estimation across this range as shown in Fig. 4.2b. The error is a much steeper function of σ_{ii} than ϵ_{ii} in these ranges since large particles share virtually no configuration space overlap with small particles in a dense fluid due to changes in the packing of solvent particles around small solutes.

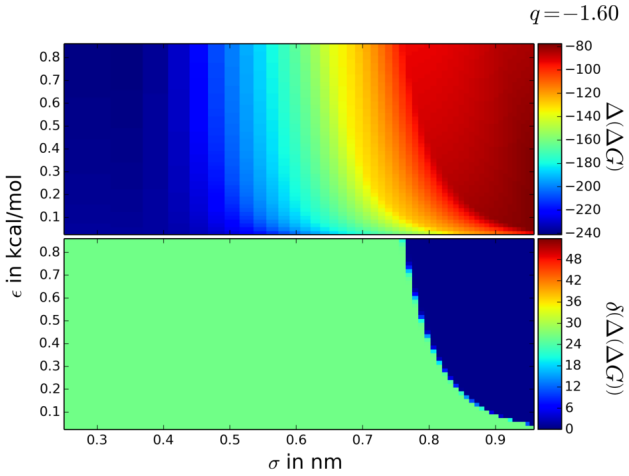
The linear basis functions approach reproduces the results from direct fixed-parameter solvation simulations. Fig. 4.2b is annotated to show where several chemically realistic Lennard-Jones spheres fall in the parameter space. Soft core solvation simulations were run for the parameters of united atom (UA) methane, [60, 72, 163] neopentane [167], and a sphere roughly the size of a C_{60} molecule. [54] The relative free energies are statistically indistinguishable between the direct solvation simulations and those computed in Fig. 4.2b. The exact numbers and methods for the solvation simulations are shown in the supplementary material in Section C.2.

4.4.2 Solvation properties over a 3-D parameter space

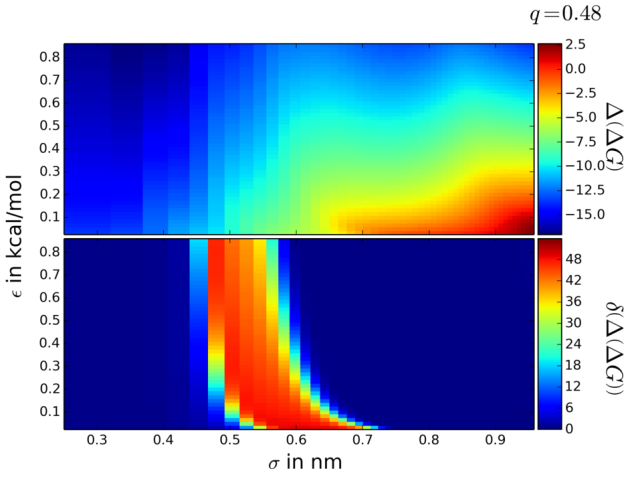
Even visualizing the thermodynamic properties and their uncertainties in 3-D space is a nontrivial task. Iterative determination of optimal states to sample is significantly harder in this higher dimension space. Fig. 4.3 shows the relative solvation free energy in the 3-D parameter space for three slices of fixed q_i, and samples drawn from the initial 21 states. Because the reference state is an uncharged particle, there is large uncertainty in the relative solvation free energy to charged particles, as seen by Fig. 4.3a and Fig. 4.3c. We show the entire 3-D space of solvation free energy as a function of the three force fields parameters in an animated movie provided in the supplementary material. [168]



(a)



(b)



(c)

Figure 4.3: Caption on following page

Figure 4.3: **The free energies in a multidimensional nonbonded parameter space can be estimated and visualized rapidly.** Shown are slices of the 3-D parameter space cube at fixed q_i with the initial sampling of 21 states. Because the reference state is an uncharged particle, the error to charged parameters is large. The lack of configuration space overlap causes the uncertainty in the error estimate to be large and unconverged. (a) and (c) show samples of the free energy on either side of $q = 0$. (b) shows a discontinuous jump in uncertainty between the nearby charge $q = -1.52$ in (a), an artifact of poor configuration space overlap. Animated movies showing the full free energy and uncertainty across the whole parameter space for initial and final samplings are included in the supplementary materials. [168] Free energy is shown in units of kcal/mol.

Regions of poor configuration space overlap and large uncertainty can be visually identified in the initial simulations. Fig. 4.3a and Fig. 4.3b are taken from two nearby values of charge. One would expect the uncertainty to change smoothly with the partial charges differing by only 0.08, however the magnitude of the uncertainty changes by a factor of nearly two, causing a visual artifact. These sorts of artifact indicates that there is little configuration space overlap, and thus both the free energies themselves as well as the estimate of the uncertainties have not converged. In order to improve our estimates, additional samples must be drawn at new parameter combinations to improve the configuration space overlap between our reference point and parameter regions such as the ones in Fig. 4.3b. However, deciding exactly where to put these new samples cannot be easily done by visual inspection as in the 2-D case. We must identify an algorithmic ways of placing these new points that can be easily automated, rather than having to do time-consuming manual trial-and-error simulations.

4.4.3 Adaptive sampling in 3-D parameter space and improving configuration space overlap

We need to create an algorithmic way to draw a minimal number of total samples from multiple states in thermodynamic space and create low error estimates of the free energy and other thermodynamic properties. We use as a metric of convergence the

extent of overlap of configurations between every state in the thermodynamic space since high overlap provides low uncertainty estimates. We start by defining the concept of the network of thermodynamic paths in multidimensional space, followed by defining the configuration overlap along such a network. Our algorithm detects when there is insufficient configuration space overlap between sampled states, then adaptively choose new states to sample to maximize the overlap. We briefly summarize the algorithm here, with additional details of the algorithm covered in the supplementary material [168] in section C.3. The implementation used is available online. [166]

Explicit thermodynamic paths need not be directly defined by the user if property estimates are made by reweighting samples spanning the configuration space of all parameters being searched. A multistate statistical analysis method such as MBAR [101] takes into account the configuration space overlap from all samples relative to the state of interest. The concept of a single "path" is now obsolete since any two states are now connected through a network of configuration space overlap and connected states. In order to have low statistical error estimates, we need to ensure our algorithm generates sufficient configuration space overlap through the network of sampled states to connect the two states we want free energy estimates between.

We define two types of configurational space overlap that help us better describe the lack of overlap our algorithm detects. We define the *local configuration space overlap* as the set of configurations shared between simulations performed at an arbitrary parameter combination, and other nearby parameter combinations up to some finite δp away in parameter space, where p is one of the parameter we are searching through. We also define *global configuration space overlap* as the extent to which all states of interest in the parameter space are connected to all other states in this parameter space through a connected network of regions with local configuration space overlap. We will need the algorithm to detect boundaries of local configuration space overlap and extend them until global configuration space overlap is created.

A key issue in sampling multidimensional parameter spaces is that the most naive adaptive sampling will lead to improving local configuration space overlap, but will not extend the boundaries around sampled states, and thus will not create global configuration space overlap. The intuitive place to put additional samples to improve uncertainties is the parameter combinations where uncertainties are the largest. However, sampling only the largest uncertainty point may improve the local configuration space overlap for states near this sampled state, but does not necessarily create networks of local configuration overlap boundary yielding global configuration space overlap. In this case, the uncertainty in relative solvation free energy differences between states inside a region of local configuration space overlap becomes smaller, but uncertainty to the reference state, or any other state outside of the local configuration space overlap, will still be large. The presence of local overlap but not global overlap is easy to identify when sampling along a 1-D path, since it is easy to tell where along the path there are insufficient samples. Global overlap is harder to identify or create in higher dimensional space since it is much less obvious where new samples should be placed by visual inspection. We need not place samples in all places where there are no samples; we instead must design our algorithm to extend local configuration overlap boundaries and automatically construct a network of overlapping states.

We now present our algorithm to adaptively choose new states to sample. The algorithm is designed to minimize the number of samples needed to create a network of local configuration space overlap connecting to the reference state. Further details of the algorithm can be found in the supporting information. [168]

We first identify regions of local configuration overlap with a clustering algorithm and image processing tools. Local configuration space overlap is identified in the 51^3 grid by clustering adjacent points having nearly identical statistical uncertainty relative to the reference state with a density-based clustering algorithm, DBSCAN. [169] Lack of configuration space overlap between two clusters results in a nearly constant, large

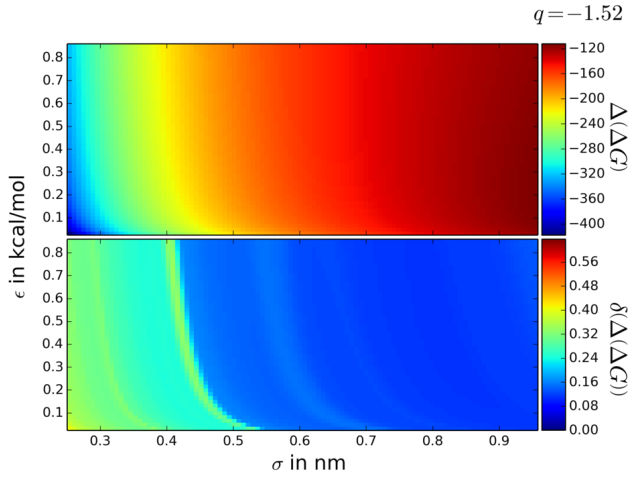
uncertainty estimate between any two points in either cluster. Therefore, as we examine the free energy difference as a function of parameter, the estimate of the statistical uncertainty to the reference state changes discontinuously at the cluster edge. We then treat each cluster as a volume-occupying shape in the parameter space using SciPy's [130] multi-dimensional image processing module `ndimage`.

We then enlarge each local configuration space overlap cluster to better connect with its neighbors. We first choose a new state to sample at random inside each local configurational cluster as we may have no samples in this cluster of local configuration overlap. We then generate a complete, weighted graph where the state chosen inside each cluster's volume are vertices, and edges of the graph are the lines connecting these states and the reference state.

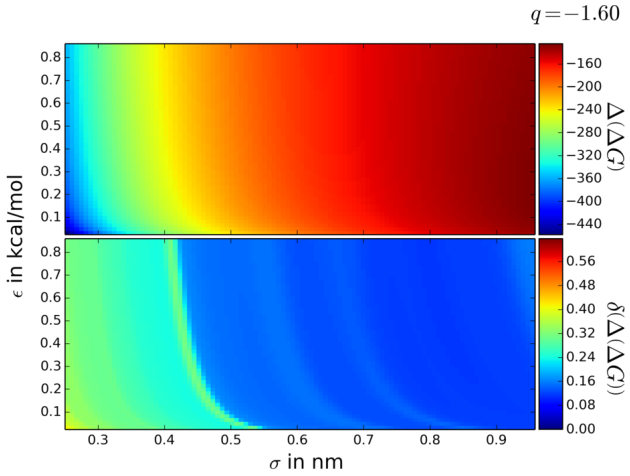
The weight of each edge is computed by numerically integrating the uncertainty at an equal number of uniformly spaced points along each edge. The algorithm estimates the uncertainty of each integration point by multidimensional interpolation from nearby grid points since many integration points are not on the 51^3 grid. A minimum spanning tree (MST) is created from this complete weighted graph using Kruskal's algorithm [170] implemented in SciPy's sparse graph routines, with the weight used as the distance between clusters. The MST provides the directions along which each cluster's volume is expanded. The intersection of the edge with the cluster's boundary is detected by the Sobel boundary detection algorithm [171] implemented in SciPy since the uncertainty changes discontinuously at the boundary. We then draw new samples at the randomly chosen vertex states, and states defined by the parameters at the intersection of the each graph's edge with the cluster boundaries. This graph-theoretic approach scaling to arbitrary N-dimensions, as direct visualization of higher dimensional spaces becomes increasingly difficult.

Statistical uncertainties in the free energy differences between states are reduced by two orders of magnitude even though the amount of sampling is only increased by

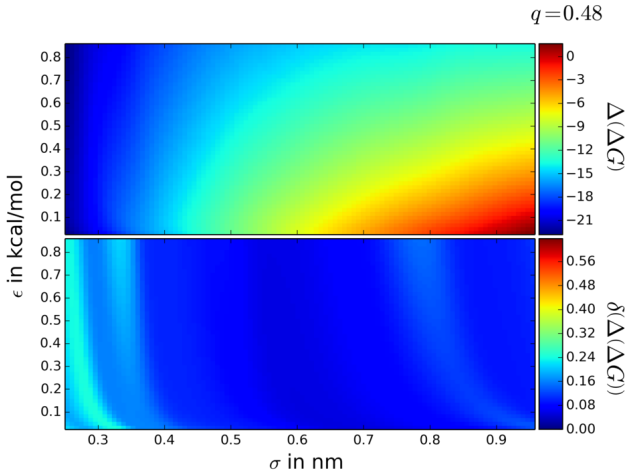
nine times (21 to 204 states), because this adaptive algorithm generates good global configuration space overlap. Fig. 4.4 shows the same three slices of the 3-D parameter space as Fig. 4.3, now with 203 sampled parameter combinations, all adaptively chosen except for the initial 21 combinations. We estimated properties at 101 σ_{ii} points for the figure to improve image quality. All time comparisons are made assuming 51^3 parameter combinations. During this process, the maximum error in relative solvation free energy differences was reduced from 53.405 kcal/mol to 0.631 kcal/mol and the mean error was reduced from 16.162 kcal/mol to 0.118 kcal/mol. However, the initial uncertainty is a misleading underestimate as much of the parameter space had no global configuration space overlap with the reference state, meaning the error estimates are unconverged.



(a)



(b)



(c)

Figure 4.4: Caption on following page

Figure 4.4: **Adaptive sampling allows reduction of the uncertainty in the whole multidimensional nonbonded parameter space.** Shown are slices of the 3-D parameter space at the same fixed q_i as in Fig. 4.3. The total uncertainty has been reduced by more than an order of magnitude with only a few adaptive iterations and a total of 203 sampled states. (a) and (c) have significantly reduced error relative to their counterparts in Fig. 4.3. (b) no longer has the discontinuous uncertainty as it did in Fig. 4.3. Animated movies showing the full free energy and uncertainty across the whole parameter space are included in the supplementary materials. [168] Free energy is shown in units of kcal/mol.

The consequences of the poor configuration space overlap can be seen in Fig. 4.5 where the maximum and mean uncertainty jumps at certain iterations. The jumps in uncertainty indicate that a new region of poor configuration space overlap has been identified and partially sampled. Ways to monitor when global configuration space overlap has been reached are discussed in Section 4.4.6.

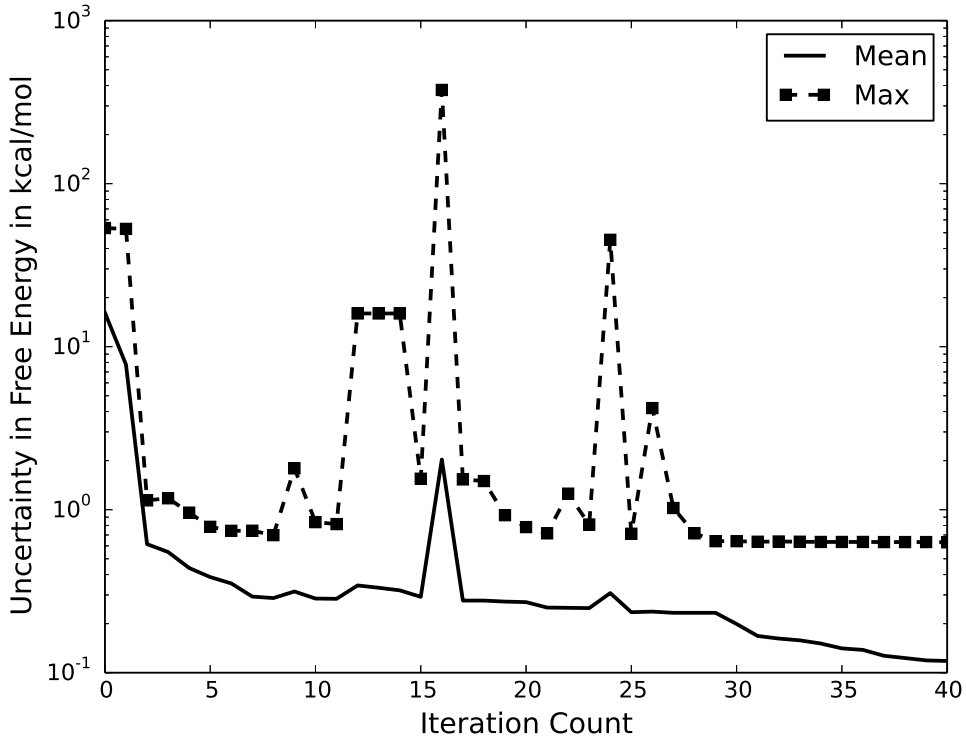


Figure 4.5: **Discovery of regions of poor configuration space overlap appear as sudden jumps in the maximum uncertainty from between two iterations, but these regions are joined by sampling adaptively.** The mean (solid) and maximum (dashed) uncertainty in the free energy for the 3-D nonbonded parameter combinations are plotted as function of iteration of the algorithm. The algorithm reduces the uncertainty of the largest areas of uncertainty before moving to others, where it can find regions of no configuration space overlap. Once some configuration space overlap is found through adaptive sampling, the uncertainty jumps as a more converged estimate can be made. The iterative process improves configuration space overlap and lowers overall uncertainty. Uncertainty in free energy is shown in units of kcal/mol and shown using a logarithmic scale to show changes at both large and small maximum uncertainty.

The adaptive sampling algorithm correctly places samples to reduce regions of poor configuration space overlap. Fig. 4.6 shows all of the sampled states in a scatter plot of the 3-D parameter space. Subsequent adaptive iterations are shown in color scale ranging from blue for the initial iterations to red in the final iteration. The large clustering of points is expected at small σ_{ii}, small ϵ_{ii}, and large q_i, because the TIP3P water's hydrogens can tightly rearrange around the particle due to very large Coulombic interactions and nearly no Lennard-Jones repulsion against the water's

oxygen. The tightly packed water arrangements share little to no configuration space overlap with any other parameter combinations, so many samples at these states are needed to accurately estimate properties.

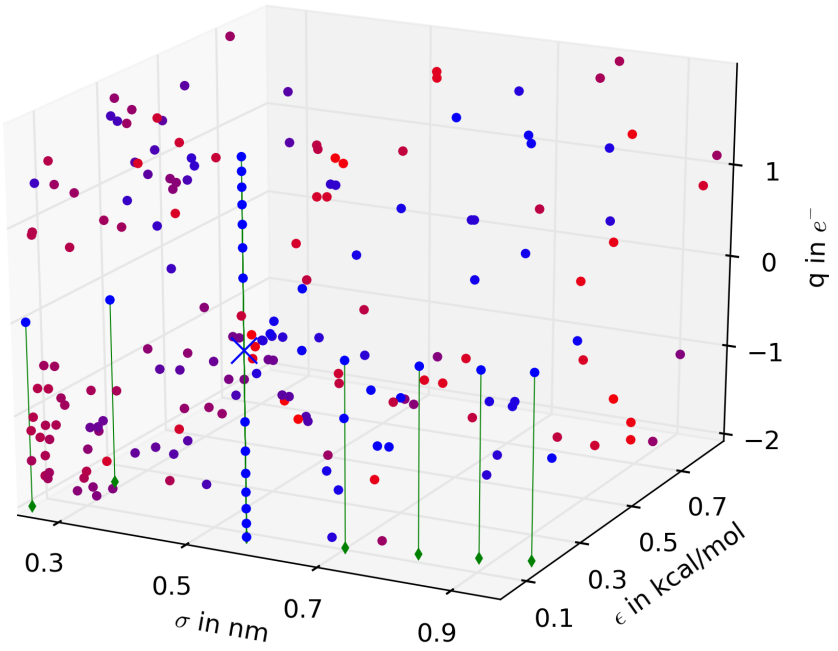


Figure 4.6: **The adaptive sampling algorithm samples the entire parameter space.** The sampled states are shown as a scatter plot with successive iterations moving from initial (blue) to final (red) as a function of the 3-D parameter space. The initial sampled points are projected onto the $\sigma_{ii} - \epsilon_{ii}$ plane (green) connected with a green vertical line to help distinguish them from points chosen adaptively. Many new states are selected at small σ_{ii} as large uncertainty in the region is identified from successive iterations. Fewer samples states are needed at intermediate σ_{ii} as the configuration space overlap to states already sampled is high. The reference state is shown as an X.

The bottleneck of computing the energies is completely removed with the linear basis function approach. There are more than five times more states in the 3-D parameter (132,651) space than the 2-D space (22,801). Despite this, computational cost to compute the energies in 3-D space only increases to less than 30 CPU seconds for 21 sampled states, and just over 50 CPU seconds for 203 sampled states. Taking again

the conservative average of 1500 CPU seconds needed to evaluate each trajectory's configuration energies at a new state, energy evaluations at the 132,651 parameter combinations would have taken over 132 and 1280 CPU years for 21 and 203 sampled states, respectively. Optimized code to explicitly calculate the basis functions at the time of the force calculation would allow even faster vectorized calculations than the post-processing used here, allowing this method to scale to even larger multidimensional spaces. After removing this bottleneck, the main cost is in performing the 203 simulations at sampled states and estimating properties with MBAR. Our simulations took an average of 25 CPU hours per simulation to run, and MBAR calculations took 108 CPU hours to compute properties. We would need to invest the time to run simulations and compute properties with MBAR, independent of how the energies were computed.

The algorithm could be further optimized depending on the relative cost of the simulations and of MBAR over very large numbers of states. In this case, simulations were relatively cheap compared to MBAR. If the simulations were more expensive, then shorter simulations could be run between iterations. Additionally, more proposed states for simulation could be generated at each step to reduce wall time. For example, instead of 10 new states run for 10 ns, 20 new states could be run for 5 ns each. In general, shorter cycles of simulation plus analysis are expected to improve performance within well-sampled regions, the error in free energy estimates scales approximately as $N^{-1/2}$. Adding significant configuration space overlap between regions that are not connected will scale significantly better.

Shorter cycles can reduce the uncertainty more quickly than simply drawing more samples at fewer states. To show this, we truncated the data from 203 states after 3 ns of simulation and performed the analysis again. This truncation gives roughly the same number of samples as the 103 state iteration, though ignores the fact that the predicted location of new sampled states may be slightly different with half as much

data collected at each iteration. The mean uncertainty between the truncated 203 states of data and 103 states of data was 0.157 and 0.251 kcal/mol and the maximum uncertainty was 0.787 and 0.716 kcal/mol, respectively. The fact the mean uncertainty is lower for the truncated set at 203 states but the maximum was lower for the 103 states can be resolved by looking at the convergence of the algorithm by direct measurement of the configuration space overlap. We show that the truncated 203 states has better configuration space overlap and thus connectivity than the equal-sample 103 sate case, and is therefore a more reliable estimate, in our convergence discussion in section 4.4.6.

Non-adaptive sampling of the parameter space would be vanishingly unlikely to have provided accurate estimates with reasonable statistical uncertainty. Figure 4.6 shows that significantly more samples were needed at smaller σ_{ii} in order to yield low statistical error in this region, which would not have happened with either random or uniform sampling without significantly more sampled states. Additionally, to yield reasonable answers throughout space, any sampling scheme requires a connected network of global configuration space overlap. It is unlikely that the correct bridging states connecting sampled regions of space could be placed randomly, and these bridging states do not exist along any rectangular grid. Given these considerations, simpler non-adaptive sampling schemes would almost always have higher uncertainty estimates for an equal number of samples, and were not directly tested in this work.

We want to emphasize that searching this parameter space is theoretically possible with pre-defined paths, but computationally impractical. In Chapters 2 and 3, optimizing the alchemical switches to identify the most statistically efficient pathways worked well to optimize alchemical paths in one dimension. Analyzing the sample variance in thermodynamic integration (TI) along paths and optimizing the $h_n(\lambda_n)$ to reduced this variance identified the lowest error pathway. However, since we need a network of connected states in multidimensional space, optimizing TI would require

mapping the multidimensional space to 1-D and connecting each state along a fixed path, Since there are many possible ways to connect states in multidimensional space, optimizing TI in this multidimensional space would require connecting each state to every other state along a fixed path, resulting in $(51^3)!$, more than 10^{10^5}, possible overlapping paths, assuming the paths were restricted to visiting each state only once. We can instead use the adaptive method outlined here to sample discrete states in the multidimensional space that creates a global configuration space overlap network over the entire space, then analyzing the configuration space overlap between pairs of states with MBAR. [101]

4.4.4 Computing Other Thermodynamic Properties and Comparing to Reported Results

Estimating properties over the entire multidimensional parameter space at once can provide thermodynamic information which would otherwise require extensive simulations to compute at each thermodynamic state. This section looks at estimating five properties where significant simulation is required when sampling each state individually: the relative solvation entropy and enthalpy, the absolute solvation free energy of ions, the radial distribution function (RDF) focusing on the first hydration shell, and the difference between the Born solvation free energy and the simulation estimate in the free energy of charging a particle. In this section, we show how we can compute the enthalpy and entropy from the same collected data as used for free energies, compare ion parameter free energies to show accuracy and limitations in our approach, estimate RDFs without generating trajectories at the target parameters, and identify trends in the deviation of solvation free energies from the Born solvation approximation.

We can estimate thermodynamic observables at any parameter combination so long as the observable is an equilibrium property. Observables of equilibrium properties

are estimated by the statistical expectation value of the observable. Computing an equilibrium expectation value of some observable, A, with MBAR [101] is

$$\langle A \rangle_a = \sum_{n=1}^{N} W_{na} A(\mathbf{x}_n) \tag{4.7}$$

where the summation runs over all samples in all states, W_{na} are the statistical weights from reweighting each nth drawn sample in the ath state, the $A(\mathbf{x}_n)$ are the observed values from sample n and are functions of the configuration, $\mathbf{x}$. The weight MBAR assigns to each sample n is [101]:

$$W_{na} = \frac{e^{f_a - u_a(\mathbf{x}_n)}}{\sum_{k=1}^{K} N_k e^{f_k - u_k(\mathbf{x}_n)}} \tag{4.8}$$

where f_a is the reduced free energy (βA or βG) of state a, $u_a(\mathbf{x}_n)$ is the reduced internal energy (βU) of the configuration $\mathbf{x}_n$ in state a, and N_k is the number of samples collected from state k. These equations are what allow us to estimate any equilibrium thermodynamic property at any state from our collected samples.

Relative Solvation Entropy and Enthalpy

We can estimate the relative solvation entropy and enthalpy alongside the relative solvation free energy without additional sampling. The relative solvation enthalpy is the difference in solvation enthalpy, ΔH_{solv}, between the reference state X and any other state j in the multidimensional parameter space. We compute the relative enthalpy difference, $\Delta\left(\Delta H_{Xj}\right)$, as

$$\Delta\left(\Delta H_{Xj}\right) = \langle \Delta U_{Xj} \rangle \tag{4.9}$$

since ΔU_{Xj} contains the configuration potential energy and the PV contribution. Similarly the relative solvation entropy, $T\Delta\left(\Delta S_{Xj}\right)$, is computed as

$$-T\Delta\left(\Delta S_{Xj}\right) = \Delta\left(\Delta G_{Xj}\right) - \Delta\left(\Delta H_{Xj}\right) \tag{4.10}$$

where $\Delta\left(\Delta G_{Xj}\right)$ is the relative free energy of solvation. In all cases, we report the difference in thermodynamic properties with respect to the reference state, subscript X. These properties we estimate have such large uncertainty from only the initial 21 states due to poor sampling that any number we report is essentially meaningless. Computing relative enthalpies and entropies generally requires significantly more samples to compute than relative free energies, [162, 172, 173] as only samples with local configuration space overlap to the states of interest significantly contribute to the precision of expectations of observables such as the enthalpy.

Additional sampling reduces the uncertainty in the estimates of relative solvation entropy and enthalpy by orders of magnitude, but not to the same extent as the uncertainty in the free energy. This is because whether or not there is good global configuration space overlap does not ensure that a given state has good local configuration space overlap with its neighbors. Fig. 4.7 shows the relative solvation entropy and enthalpy estimations, along with uncertainty at 203 sampled states. The uncertainty smoothly transitions between adjacent states, suggesting the estimates are numerically converged. However, the maximum uncertainty is several orders of magnitude larger than the relative solvation free energy. The maximum uncertainty in relative solvation entropy changes from an uncertain estimate to $0.193\,\mathrm{kcal}/\left(\mathrm{mol}\cdot K\right)$, and the relative solvation enthalpy's maximum uncertainty drops to $57.5\,\mathrm{kcal/mol}$. The mean uncertainty for relative solvation entropy and enthalpy fall to $0.0147\,\mathrm{kcal}/\left(\mathrm{mol}\cdot K\right)$ and $4.37\,\mathrm{kcal/mol}$, respectively. Although these error estimates are still too large to make practical predictions of solvation entropies, the estimates of the errors from the 203

states are well-defined, which is a marked improvement over the initial sampled states. The whole configuration space has been sampled, and all states now have at least moderate local configuration space overlap with its neighbors. Once decent estimates of properties are found, we can run additional simulations on states that have the most desirable preliminary estimates of properties.

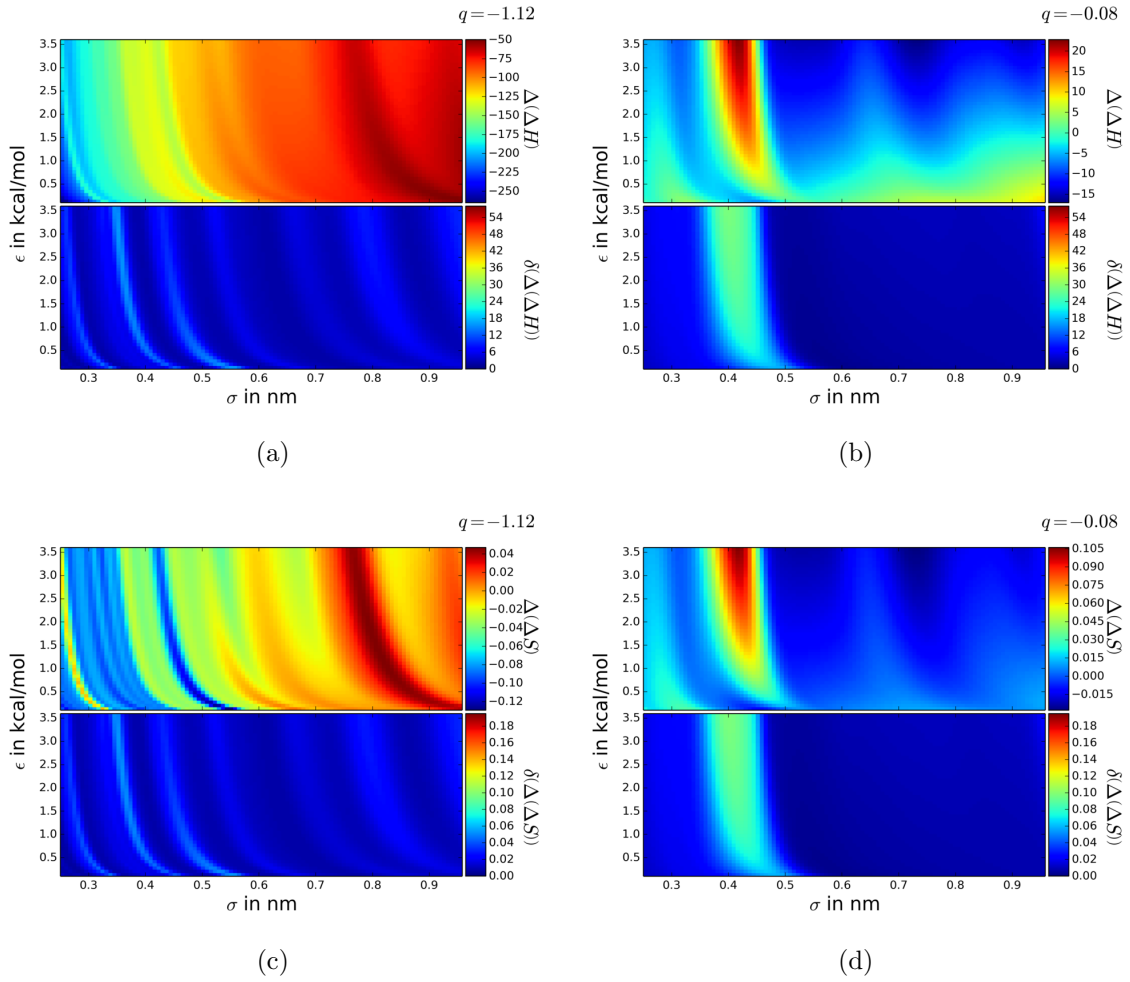


Figure 4.7: **Uncertainty in entropy and enthalpy are also reduced with the uncertainty in the free energy.** The enthalpy H, ((a) and (b)) and entropy S, ((c) and (d)) are computed from 203 total sampled states and reported in kcal/mol and kcal/ $(\text{mol} \cdot K)$ respectively. The uncertainty in both properties now smoothly transitions between adjacent states. The uncertainty is still significantly larger than that of the free energy, which is expected as these properties require more sampling to compute accurately than the free energy. Additional samples or other means of computing these properties would be required to reduce error further.

Ion Solvation Free Energies

We compare the results from our analysis to a detailed ion parameter study by Joung and Cheatham. [74] Their study parameterized Li^+, Na^+, K^+, Rb^+, Cs^+, F^-, Cl^-, Br^-, and I^- in several water models including TIP3P and compared them to experimental and computational studies from others. They parameterized the ions based on several experimental observables, including free energy of solvation. We compared their absolute solvation free energies to absolute solvation free energies we computed from our method in Table 4.1. The table shows the ion parameters and free energy of solvation, ΔG, estimated from their work and our approach. The table also shows the first hydration shell (FHS) location which will be discussed in the next section. We compute absolute solvation free energies for our work by adding the free energies from the relative free energy evaluations to those from a single set of solvation simulations of the reference particle along a soft core potential [82, 83] and a 1-1-6 parameterization. [54, 84] These simulations were run with the same conditions as described in section 4.3 at 11 states uniformly distributed along λ from 0 to 1 of the soft core path. We note that these particular choices of ion parameters we directly compare to represent only the smaller particle / higher charge density fraction of the parameter space. However, we showed in the 2D search that the large, uncharged particles were correctly described by rapid parameter scans. The fact that the results for these extreme cases match direct simulation strongly suggest other cases, such as large ions with correspondingly lower charge density, will also match more direct calculations of the free energies.

Table 4.1: Absolute solvation free energies and first hydration shell (FHS) locations compared from this work and from Joung and Cheatham. [74] Ion σ_{ii} were back calculated from Lorentz-Berthelot to geometric mixing rules for ion/oxygen interactions. ϵ_{ii} and ΔG are in kcal/mol; σ_{ii} and FHS location are in nm. Error in this work's FHS computed by 200 bootstrap samples [123] with a discretization of ± 0.0075 nm.

Ion	σ_{ii}	ϵ_{ii}	ΔG		FHS	
			Joung and Cheatham	This Work	Joung and Cheatham	This Work
Li+	0.1965	0.0280	-115.6	-105.26 ± 0.54	0.196	0.211 ± 0.016
Na+	0.2479	0.0874	-90.6	-90.63 ± 0.44	0.238	0.234 ± 0.015
K+	0.3039	0.1937	-72.6	-72.42 ± 0.36	0.275	0.279 ± 0.016
Rb+	0.3231	0.3278	-67.6	-67.31 ± 0.35	0.292	0.294 ± 0.030
Cs+	0.3532	0.4065	-62.5	-62.13 ± 0.35	0.311	0.309 ± 0.021
F-	0.4176	0.003364	-121.6	-120.91 ± 0.45	0.263	0.257 ± 0.016
Cl-	0.4617	0.0356	-91.5	-90.99 ± 0.36	0.313	0.309 ± 0.028
Br-	0.4825	0.0587	-84.8	-84.48 ± 0.36	0.329	0.333 ± 0.015
I-	0.5396	0.0537	-75.9	-75.89 ± 0.34	0.351	0.347 ± 0.017

We re-computed parameter values and adjusted the reported free energies to make the values from Joung and Cheatham [74] comparable to this work. Their ion σ_{ii} was calculated for Lorentz-Berthelot mixing rules. We back calculated the σ_{ii} for geometric mixing rules in Table 4.1 by setting the σ_{ij} between the ion and the oxygen in water equal in both mixing rules, giving the relation

$$\sigma_{\text{ion-ion, geo}} = \frac{\left(\sigma_{\text{ion-ion, LB}} + \sigma_{\text{OW-OW}}\right)^2}{4\sigma_{\text{OW-OW}}} \tag{4.11}$$

where $\sigma_{\text{ion-ion, geo}}$ is the reported ion σ_{ii} in Table 4.1, $\sigma_{\text{ion-ion, LB}}$ is σ_{ii} for the Lorentz-Berthelot mixing rule reported by Joung and Cheatham, and $\sigma_{\text{OW-OW}}$ is the TIP3P oxygen-oxygen σ_{jj} which we assumed was constant between the mixing rules. The solvation free energies from Joung and Cheatham were adjusted by -1.9 kcal/mol to remove the correction they added for ideal gas expansion when comparing simulations, carried out with gas-phase standard states at 1 M, to experimental results, where gas-phase standard states are typically 1 atm. [74, 174]

The solvation free energies from our work and Joung and Cheatham's work are within statistical error. The free energies from this chapter appearing in Table 4.1 are within two standard deviations of Joung and Cheatham [74] for all ions except Li^+, which is outside the parameter range studied. The comparable accuracy of our results to those of Joung and Cheatham provide validation for the free energies we report. Our method has the added benefit of computing the solvation free energy for arbitrary parameter combinations. The ability to compute properties for arbitrary parameter combinations comes from sampling only 203 states plus 11 for the absolute solvation free energies. Joung and Cheatham carried out 12-13 simulations for each of the 9 ions, resulting in 108-117 simulations for comparison. We were able to compute properties at roughly 14,000 times the number of parameter combinations, for only double the simulation cost.

Our method breaks down if the parameter combination falls outside the defined range. The parameters for the Li^+ ion in Table 4.1 fall significantly outside the range we searched, namely, the σ_{ii} is less than the 0.25 nm minimum. The estimated free energy for this parameter combination is not within statistical error for that of Joung and Cheatham. [74] The estimated value for any thermodynamic property at this ion will likely be inaccurate as the estimation is now an extrapolation instead of a thermodynamically-consistent interpolation between sampled states. Estimates on parameter combinations falling just outside the searched range, such as the Na^+ ion, appear to still be accurate, so the range of convergence of these calculations outside of sampled parameters is not zero.

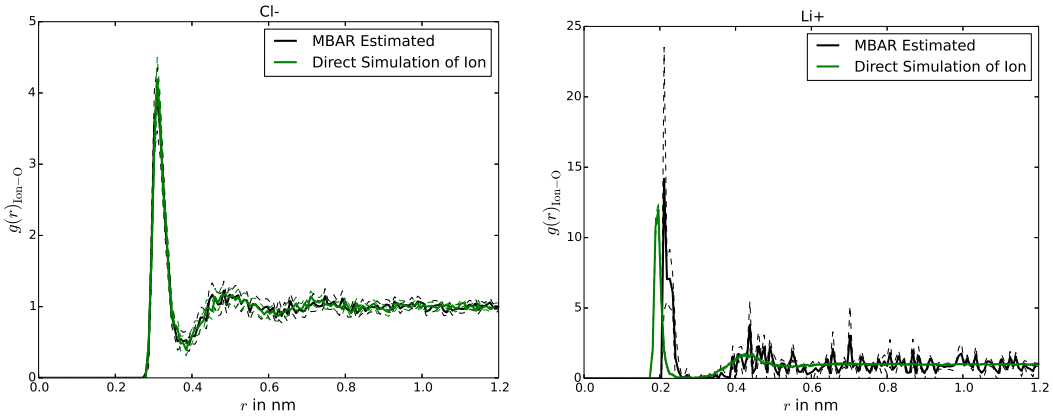
Estimating Radial Distribution Functions

We can estimate the radial distribution function (RDF or $g(r)$) of a specified parameter combination without explicitly sampling that combination. The first hydration shell and the water RDF are properties that many have tried to compute accurately and compare to experiment. [74, 175–182] Traditionally, a RDF is generated by measuring the distances between two specified atomic groups (e.g. ion-water, water oxygen-water oxygen, etc.) generated over a trajectory, counting the number of pairs that are within a shell of size $r + \delta r$, then average over the shell volume and whole trajectory. The fact that the RDF is an average property and dependent only on the configuration implies it is a thermodynamic equilibrium property which can be computed as a statistical observable. The observable for computing the RDF is the discrete count of pairs within a specified $r + \delta r$ shell normalized by the shell radius and the number volume $\rho = N_{\text{particles}}/V$. We must estimate the RDF at multiple shell volumes in order to generate a complete RDF curve.

We expect the RDFs we estimate to have noticeably more noise than an RDF computed by direct simulation at a given parameter combination. Samples with

local configuration space overlap to the target parameter combination contribute substantially more to the precision of expectation values than samples from further away. Because we sampled a wide breadth of parameter space, many of the samples are very far from the ion parameter combination, so the effective number of samples available to estimate an RDF is much lower than would be available from direct simulation. This lower number of effective samples will add noise to the RDF which is why we have chosen to focus on the more strongly defined first hydration shell. We will show however, that we can still qualitatively recover the remaining RDF features.

The first hydration shells (FHS) are accurately predicted by the RDF estimation. Fig. 4.8 shows the RDF estimated for the Li^+ and the Cl^- ions from Joung and Cheatham. [74] The RDF is estimated at 160 discrete bins (0.0075 nm spacing) from $r = 0$ to $r = 1.2$ nm along with the error in that estimate. The black curves in Fig. 4.8 are the estimates from MBAR computed purely from data collected at our 203 states, and not from any data drawn at the ion's parameters. Error in the MBAR estimate is shown as dashed lines and is two standard deviations of the uncertainty in the RDF, also computed by MBAR. To validate the MBAR results, the green curves in Fig. 4.8 show the RDF computed from simulations at the given ion's parameters. The data from these direct simulations of the ions were not used in the MBAR estimate. The error in the green curves is taken from 200 bootstrap samples [123] of the RDF for each ion. The Li^+ ion RDF is shown in Fig. 4.8b to again emphasize that estimates made outside the parameter range tend to break down, evidenced by the erratic behavior in the RDF.



(a) Cl^- (b) Li^+

Figure 4.8: **Radial distribution functions (RDF) can be estimated at any parameter combination inside the parameter search range.** The ion-oxygen RDF in TIP3P water is shown for Cl^- and Li^+ with Joung and Cheatham parameters. [74] The RDFs are estimated without sampling the explicit parameter combinations using 160 discrete bins. Estimates made from parameters inside the parameter range, Cl^- (a) are accurate as the first hydration shell is predicted within error to Joung and Cheatham. [74] in Table 4.1. Estimating parameters which fall outside the searched parameter range, Li^+ (b), are inaccurate. Error is shown dashed lines of two standard deviations, computed by MBAR for the black curves and 200 bootstrap samples for the green curves.

The Cl^- ion in Fig. 4.8a shows an example where we can estimate solvation structure and improve future simulations. We further validated the RDF calculation by determining the peak of the first hydration shell for every ion from Joung and Cheatham as the bin with highest occupancy [74] and compared our results to theirs in Table 4.1. The peaks we estimate and those from Joung and Cheatham are in agreement with each other, within the error of our bin width of 0.0075 nm. The RDF curves generated using reweighting are not as smooth as the RDF curves from other studies, and we do not expect them to be smooth with the range of parameters we searched. However, all features are well preserved. This approach can be used to broadly search parameter space and generate approximate RDF's, until those that replicate the RDF or hydration shell properties of interest are identified. Further simulations could then be run around the sets of properties which gave the RDF

replicating the target properties to make more accurate estimates, resulting in searches over a much narrower parameter space than examined through here. A complete set of the RDFs estimated via reweighting and direct simulation for every ion is included in the supplementary material. [168]

Born Approximation to Solvation Free Energy

The Born approximation to solvation free energy measures the effort to transfer a charged particle between two dielectrics. The free energy differences for this approximation of transferring a hard sphere particle between vacuum and a fluid is

$$\Delta G = \frac{q^2}{4\pi\epsilon_0 R_{ij}} \left(\frac{1}{\epsilon_d} - 1 \right) \tag{4.12}$$

where $\epsilon_d = 92$ is the estimated dielectric constant of our fluid, TIP3P water [183], and R_{ij} is the Born radius.

We can estimate the Born radius of any particle in our search space from our sampled states. Choosing the correct Born radius, or effective hard sphere (EHS) radius is a nontrivial task. However, we can estimate the EHS radius with our RDF calculation. We first compute the RDF for a given parameter combination, $g(r)$, then compute the EHS radius by determining an r_0 where the following conditions for the oxygen-ion g(r) are met:

$$g(r_0 - \delta r) = 0 \tag{4.13}$$

$$g(r_0) = 0 \tag{4.14}$$

$$g(r_0 + \delta r) > 0 \tag{4.15}$$

to a tolerance of 10^{-5}. r_0 can be interpreted as the point on $g(r)$ where the probability of finding a particle changes from zero to nonzero. We set $R_{ij} = r_0$ for the Born solvation calculations.

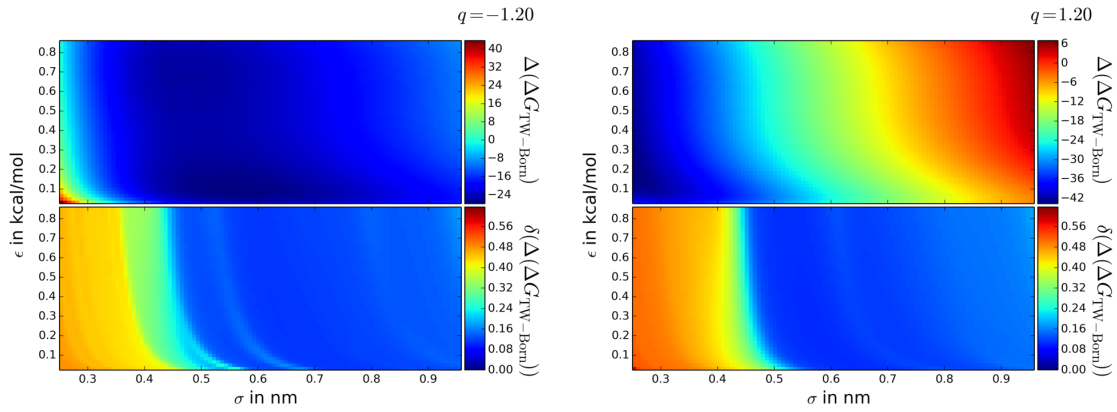
We applied correction terms to our estimated free energies to remove errors introduced by our choice of simulation settings and water model. These corrections allow a comparison of free energy between different methods without having a methodological dependence. These corrections required us to additionally compute the average box length of the simulations, found by estimating mean volume, $\langle V \rangle$, at each parameter combination through Eq. (4.7) and Eq. (4.8). All of the corrections we applied are detailed in Hünenberger and Reif [184] and we go through explicit detail of which corrections we applied and why in the supplementary material [168] in section C.6.

There are a number of reasons the Born approximation is not perfect for our particle-water system. These imperfections come from not simulating a truly infinite medium, the water model having an asymmetric charge distribution, Lennard-Jones terms affecting the free energy of transfer, and the fact that R_{ij} may change on charging since we have soft particles. We are interested in identifying deviations from the Born approximation with our method, given that we fully expect deviations from these imperfections.

The Born approximation to the solvation free energy is the free energy of transferring a charged hard sphere with radius R_{ij}, not for the free energy of solvating the uncharged particle. To remove this dependence on the cavitation free energy, [184] we estimate the R_{ij} from the RDF as described above for each uncharged combination of σ_{ij} and ϵ_{ij}, then calculate the free energy difference to the same values of σ_{ij} and ϵ_{ij} but with a charge. This allows us to compare our free energy of charging to the Born approximation to the free energy of charging, and identify deviations between the model and our simulation. We do not recalculate R_{ij} at the end state.

Both trends and failures of the Born approximation can be easily visualized for the entire parameter space. Fig. 4.9 shows the difference between the Born approximation (ΔG_{Born}) and this work's estimate for the charging free energy (ΔG_{TW}). Any deviation from $\Delta\Delta G_{\text{TW-Born}} = 0$ indicates nonidealities relative to the Born approximation to

the free energy of charging. There are several deviations which can be seen in the figure. The first deviation is the Born free energy generally predicts less favorable solvation free energy for both signs on the charged particles ($\Delta G_{\mathrm{TW}} < \Delta G_{\mathrm{Born}} < 0$). However, this deviation is asymmetric as the deviation from Born theory of the positively charged particle at $q = +2$ is up to $-152.1\,\mathrm{kcal/mol}$, but the negatively charged particle only deviates up to $-78.5\,\mathrm{kcal/mol}$ at $q = -2$. The charging free energy more strongly depends on σ_{ii} for the positive particle as the $\Delta\left(\Delta G\right)$ in Fig. 4.9a spans 13.7 kcal/mol from $\sigma_{ii} \approx 0.5$ to $\sigma_{ii} \approx 0.95$ on average, whereas in Fig. 4.9b the span 27.6 kcal/mol on average in the same range of σ_{ii}. We can also observe the opposite case where the Born estimate is more favorable than the observed estimate ($\Delta G_{\mathrm{Born}} < \Delta G_{\mathrm{TW}} < 0$) in Fig. 4.9a where $\Delta\left(\Delta G\right) > 0$ at small σ_{ii} and ϵ_{ii}. This opposite case occurs because the negatively charged particle attracts the TIP3P water's hydrogens, which do not have Lennard-Jones interactions, and would be able to approach much more closely than the Born model predicts with its hard sphere approximation. Simply estimating free energy at a few states in the parameter space would have been insufficient to observe these broad trends as a significant degree of interpolation, or worse extrapolation, would have been required.



(a) (b)

Figure 4.9: **Trends and failures in approximations can be visualized over wide parameter space.** The deviation of the Born hydration free energy from the computed solvation free energy is shown for two fixed slices of q_i. Explicitly shown is the Born free energy minus this work's free energy estimates: $\Delta G_{\text{Born}} - \Delta G_{\text{TW}}$. Free energy difference estimates for each combination of σ_{ii} and ϵ_{ij} are relative to the same combination at $q_i = 0$ as to approximate only the contribution of charging a given sphere in solvent. (a) and (b) show the deviations with the solute carrying a $\pm q$ charge. The Born model generally predicts a more favorable interaction relative to the simulation. An exception to this trend is at very small σ_{ii} and ϵ_{ii} for a negatively charged particle where it predicts a less favorable interaction than the simulation as the TIP3P water hydrogens can tightly pack around the particle. Animated movies showing the full free energy and uncertainty across the whole parameter space are included in the supplementary materials. [168] Free energy is shown in units of kcal/mol.

A full animation showing $\Delta\Delta G_{\text{TW-Born}}$ at combination in the parameter space is included in the supplementary material. [168]

4.4.5 Monitoring numerical bias

The numerical bias caused by the process of calculating energies from perturbations of reference states can be minimized. Eq. (4.6) and in the supplementary material Eq. (C.5) involve the addition and subtraction of many small numbers, depending on reference states X and Y. This can lead to rounding errors which may propagate to the simulation package and be made worse by the software's precision. Choosing reference

states with larger $\Delta C_{n,XY}$ can help reduce accumulation of error. If the software natively allows access to the basis function values, then this source of numerical error is eliminated. However, we must quantify our numerical error here since the perturbation approach was chosen.

We find that rounding errors do not propagate from the perturbed basis function representation to the thermodynamic properties for these calculations. The rounding errors for this system were checked by evaluating the energy of each sampled configuration at every sampled state as though we did not use the basis function approach. The energies of every configuration evaluated at every sampled state computed from directly from GROMACS were compared to the energies computed from the basis functions and any deviation was a result of numerical bias. The energies computed from basis function calculations and the energies directly from GROMACS reruns differed by less than 0.002%. This very small relative error does not assure that errors themselves are negligible, since large energies have large absolute errors. The largest absolute error in all 203 simulations was 11.804 kcal/mol, which at first appears is likely to have a significant effect in the final answers. However, these large absolute errors do not affect any of the property estimates. This is because these large rounding errors occur when the trajectory from a particle with small σ_{ii} is evaluated in a force field for a particle with large σ_{ii}. This often resulted in the oxygen of TIP3P water being within the large particle's excluded volume, resulting in a highly repulsive interaction. Every configuration with rounding error in this chapter had a Boltzmann weight, $\exp\left(-\beta U(r,\lambda)\right)$, indistinguishable from zero at machine precision, and thus these errors do not contribute to any of the properties of interest. The energies calculated using reference states thus give results that are sufficiently close to those from direct evaluation of the energies for all uses.

4.4.6 Convergence and alternate algorithm conditions

Examining the uncertainty estimate from the reference state alone is insufficient to determine convergence of the calculations. The multidimensional space initially has almost no global configuration space overlap, which causes unconverged estimates of the properties and their uncertainties. Poor overlap implies that the mean and the maximum uncertainty alone are not appropriate gauges for convergence since the error does not consistently decrease with number of samples as observed in Fig. 4.5. A network of overlapping configuration space between all states required for accurate estimate of properties and quantification of the uncertainties in these properties, and we need a way to diagnose whether this network has been created at a given stage of our adaptive algorithm.

The configuration space overlap can be analyzed through a multidimensional extension of the Overlapping Distribution Method. [81, 185] This method can quantify the overlap between states by considering the probability of each sample occurring in every state. The unnormalized probability of a sample can be computed from its Boltzmann weight. Just as each sampled configuration carries a Boltzmann weight for the state in which it was drawn, the configurations can be reweighted to all other states to determine what the relative Boltzmann weights are in the other sampled states. [99–101, 105, 186] MBAR [101] stores each sample's weights as a matrix $\mathbf{W}$ whose entries are the W_{na} of Eq. (4.8). The pairwise probabilities can be assembled in the "overlap matrix," constructed from the matrix of weights. This multidimensional overlap matrix is calculated from the weights as

$$\mathbf{O} = \mathbf{W^T W N} \tag{4.16}$$

where $\mathbf{N}$ is a diagonal matrix with each ith entry equal to the number of samples from the ith state. [101, 187] The individual elements of the overlap matrix, O_{ij}, can

be read as the probability of a sample generated in state j being observed in state i. Since $\mathbf{O}$ is a Markov matrix, it can be shown that $\mathbf{O}$ will have at least one eigenvalue of 1, which is also the maximum over all eigenvalues. All other eigenvalues will be real and positive. [187] However, multiple eigenvalues of 1 in $\mathbf{O}$ indicate that there are discontinuous regions of sampled configuration space and $\mathbf{O}$ can be rearranged to a block diagonal matrix that illustrates which states are and are not connected. Thermodynamic property measurements made between the discontinuous configuration spaces will have undefined uncertainty, which numerically can show up as either NaN or very large numbers that change dramatically with small changes in sampling. [187]

Monitoring both the eigenvalues of the overlap matrix and maximum uncertainty can provide good guidance as to when converged property estimates have been reached. Monitoring exclusively the eigenvalues of $\mathbf{O}$ is insufficient to determine convergence over the entire parameter space since $\mathbf{O}$ only involves the sampled states. However, if the sampled states are well-enough dispersed such that the estimated uncertainties of all the unsampled states are low, and simultaneously all the sampled states are connected as demonstrated by having a single eigenvalue with value 1 for $\mathbf{O}$, we can have high confidence that the uncertainty estimates are reliable.

We therefore defined our property estimates as converged once there were no repeated eigenvalues of 1 in $\mathbf{O}$, and once no further clusters of uncertainties in the relative free energy above the target threshold are found, i.e. the clustering algorithm can not find new adjacent grid points with large uncertainty. If desired, the uncertainty can be iteratively reduced by lowering the error threshold of the algorithm. The deviation of the five largest eigenvalues from 1 are shown for each iteration in the supplementary material in Table C.2.

We have given a practical example in our data where the eigenvalues of the overlap matrix provide more information than the mean and max uncertainty alone. We showed an example in section 4.4.3 where we compared truncated data from 203

sampled states to the roughly equal number of total samples from the 103 sampled states. The mean uncertainty was lower in the truncated 203 sampled state case, but the max uncertainty was lower in the 103 sampled states case. If we look at the second eigenvalue of the overlap matrix from each case (since the first eigenvalue is 1), we can determine how connected the samples are. The second eigenvalues are 0.9998 and 1.000 for the truncated 203 and 103 states respectively. The fact that the 103 states' eigenvalue is 1 to machine precision and the truncated 203 states is not indicates that there are still regions of discontinuous local configuration space overlap in the 103 sampled state and that the algorithm is further from convergence. This further indicates that our estimate for the maximum uncertainty in the 103 sampled states is less reliable than the estimate in the truncated 203 states. This practical example shows why the mean and max uncertainty alone are insufficient metrics of convergence.

The proposed adaptive sampling algorithm could be improved by changes to this algorithm. For example, the algorithm discussed here and detailed in the supplementary information [168] identified clusters of high relative uncertainty by counting the number of grid points in the cluster. This resulted in many states being chosen adaptive at larger σ_{ii} due to the density of grid points, despite the fact that most of the regions with no configuration space overlap were at $\sigma_{ii} < 0.35\,\text{nm}$. Although states were eventually placed at small σ_{ii}, one improvement could be to place points in regions with the largest integrated uncertainty, using the uncertainty as a weighting on the overall number of grid points. This improvement would still favor the larger clusters, but to a lesser extent as new pockets of poor configuration space overlap are identified and the uncertainty jumps back up as in Fig. 4.5.

This chapter was used to determine high accuracy free energies over entire large parameter range, and this chapter's range of nonbonded parameters will likely exceed many practical applications. However, this parameter search method could be

adaptively shrunk to hone in on specific property estimates. Instead of determining the thermodynamic properties for a large set of starting parameters, a desired thermodynamic property could be provided and a set of parameters which generate this property are searched for, as is the case for reverse property prediction. In this case, the initial grid spacing could be larger, and a rough estimate of the property surface can be acquired, spending less simulation time per iteration. Each subsequent iteration would then narrow the search area and reduce the grid spacing, seeking the target value. States from previous iterations outside the narrowed search space, can still be included in the analysis, preventing discarded information. Alternatively, computational time can be saved by excluding these outlier states if analysis of $\mathbf{O}$ shows that these states are not actually connected to the states ultimately of interest.

Thermodynamic property estimates are not limited to relative solvation free energies, entropies, and enthalpies. Once the simulations are converged, further thermodynamic properties can be derived from derivatives and fluctuations with respect to V, P, and T, [188–190] as well as any other property computed from statistical expectation values. [101]

4.5 Conclusion

We have shown how one can rapidly estimate thermodynamic properties in a multidimensional nonbonded parameter space by combining two time saving advantages. Computing the energies required for estimating thermodynamic properties can be accelerated with linear combinations of basis functions instead of re-running simulation force loops. Estimating the thermodynamic properties in the multidimensional parameter space is possible with the binless, multidimensional, path-free statistical method, MBAR. With these methods, properties are estimated at all states of interest simultaneously without needing to define how any state is connected to any others

beforehand.

Converged results can be acquired by adaptively sampling the multidimensional parameter space, creating a network of globally-connected configuration space overlap between all states. Simply adding samples in regions of large uncertainty does not necessarily create configuration space overlap to all states, as the uncertainty of differences to other states is not reduced. Regions of poor configuration space overlap can be identified by examining the overlap matrix between all sampled states, and the maximum uncertainty in the unsampled states. The parameter space is then adaptively sampled until configuration space overlap is created between the reference state and all other states of interest, and uncertainties are pushed sufficiently low for the purpose at hand.

The methods shown here can help speed up future thermodynamic property searches in multidimensional parameter space. So long as the energy functions can be computed with vector operations, and do not require re-running the simulation force loops, these methods can scale to even higher dimensionality and extend to other thermodynamic properties. Re-writing the simulation code to directly provide the required basis functions would allow even faster energy evaluation, potentially removing the need to ever compute the energy of a configuration more than once.

Chapter 5

Sampling and Estimating Chemical Properties in a Combinatorial Chemical Space

5.1 Introduction

This chapter describes an approach to estimate thermodynamic properties of a combinatorial set of molecules from a single simulation in a computationally efficient way. We extend the basis function method described in Chapters 2 and 3 to modify the interactions of fragments of a molecule along multidimensional alchemical pathways, as opposed to alchemically modifying the interactions of a single, complete molecule along a single alchemical pathway. We apply the multidimensional convergence tests from Chapter 4 to check our confidence in the estimates of thermodynamic properties. The majority of the effort is in designing a simulation approach which correctly simulates a molecule representing a combinatorial chemical space. We also design the simulation to collect the potential energies of each molecular fragment as opposed to just total potential energy of all molecules.

We can reduce molecular design costs by reducing the number of chemicals we have to synthesize and test in a laboratory setting. The cost to purchase chemical building blocks can range from $1/gram for simple chloroformates to more than $1000/gram for complex primary amines, and that is with optimized synthesis procedures [191, 192]. The actual cost to create a new molecule will then require time to develop a new synthesis procedure, wet lab operational costs, and finally the cost to measure thermodynamic properties, all on top of raw material costs. We want to reduce the costs of designing molecules by reducing the size of the chemical space we have to test in a wet lab. We can use computational methods to estimate thermodynamic properties without synthesis, and filter out chemicals before we ever have to physically make them. Computational methods can only help reduce the number of molecules that need to be synthesized, and will never replace the need for a laboratory. Free energy differences are a particularly helpful property to compute as they provide information such as chemical partitioning, activity coefficients, binding affinities and other properties from their first and second derivatives. If we can predict free energies

over large chemical spaces, we can filter out molecules with undesirable thermodynamic properties, reducing the number of molecules that have to be synthesized.

Previous attempts to estimate properties in large chemical space are limited in the diversity of their chemicals or must make a number of assumptions to obtain accurate result. Semi-empirical methods like UNIFAC and UNIQUAC provide estimates for many non-ideal liquids, however, they are based on empirical observations with limited statistical mechanics support, making them unable to provide quantitative estimates for complex biological systems such as membranes and drug binding [65, 66, 77, 78]. The more computationally expensive, but more statistical mechanically exact simulation methods, can predict properties in more complex molecular systems, but do not scale well to large chemical spaces. Shared volume methods and iterative thermodynamic paths updates are limited by the diversity of molecules they can explore [49, 50, 91, 103, 104]. The most recent advances in multidimensional thermodynamic paths have to carefully choose parameters to avoid simulation instability and make approximations at the paths' end states [51, 52]. These statistical mechanics simulation methods share a fundamental issue in that their computational effort scales at an inefficient rate proportional to the number of thermodynamic states, which increase with the size of the chemical space.

The current statistical mechanical simulation free energy estimation methods described above are computationally demanding and thus require prohibitive effort to predict accurate thermodynamic properties over a large chemical space. These estimate the free energy differences between thermodynamic states by estimating the ratio of partition functions from drawn samples. The accuracy of the estimation, previously detailed in section 1.1.1, depends on how much phase space overlap exists between states. Chapter 4 shows the difficulty in generating phase space overlap for large chemical spaces, but we recap here to emphasize the computational effort required to estimate properties in such spaces. The phase space overlap is generated

by sampling along thermodynamic paths connecting the end states. The MBAR [101] free energy estimator allow us to estimate properties without explicitly defining thermodynamic paths, but requires the energy of every sampled configuration we collect to be computed at every sampled state, and every end state. The most traditional way to get all these energy evaluations is to run every sampled atomic configuration through a simulation software's force code to evaluate the energy at every state. These evaluations scale computationally as $\mathcal{O}\left(N(K_u + K_s^2)\right)$ where N are the number of samples, K_s is the number of sampled states, and K_u are the number of unsampled states. The previously mentioned methods for estimating free energies over larger chemical spaces are prohibitively slow at this scaling for thousands to millions of end states. Example timing and further details of this are covered in Chapter 4.

This chapter presents our method to estimate the solvation free energy difference over a combinatorially large chemical space in a computationally and statistically efficient way. We reduce the computational scaling to estimate energies at all states through the basis function method we developed in Chapters 2 and 3. This lowers the energy evaluation computational scaling down to roughly $\mathcal{O}\left(NK_s\right)$ as energy re-evaluation is handled through matrix multiplication instead of relying on simulation force evaluations. The basis function method also maximizes the phase space overlap along a single path to maximize statistical efficiency. Here, we extend the basis function method to multidimensional thermodynamic paths to sample combinations of end states at the same time. Chapter 4 looked at how to explore large multidimensional chemical space and use MBAR in tandem with the basis functions to show multidimensional property prediction could be done with computational efficiency.

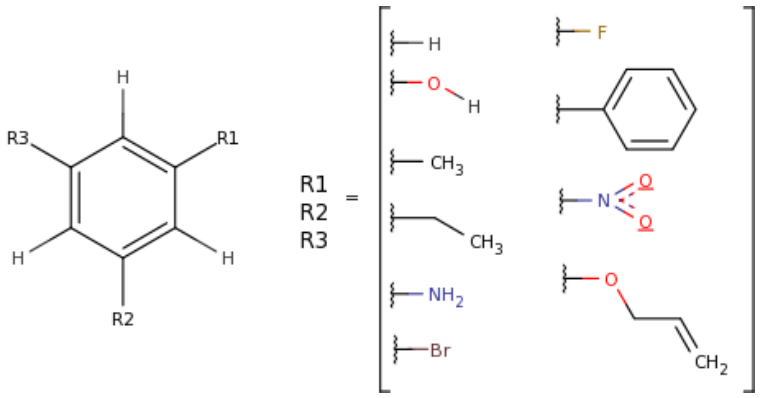
This work specifically combines our basis function method with λ-dynamics [51, 52] and Hybrid MC [193, 194] to preserve the correct thermodynamic ensemble, run stable simulations, and explore a multidimensional chemical space. In this implementation, we only look at 10^3 end state molecules, but this method could scale to systems with

orders of magnitude more end states.

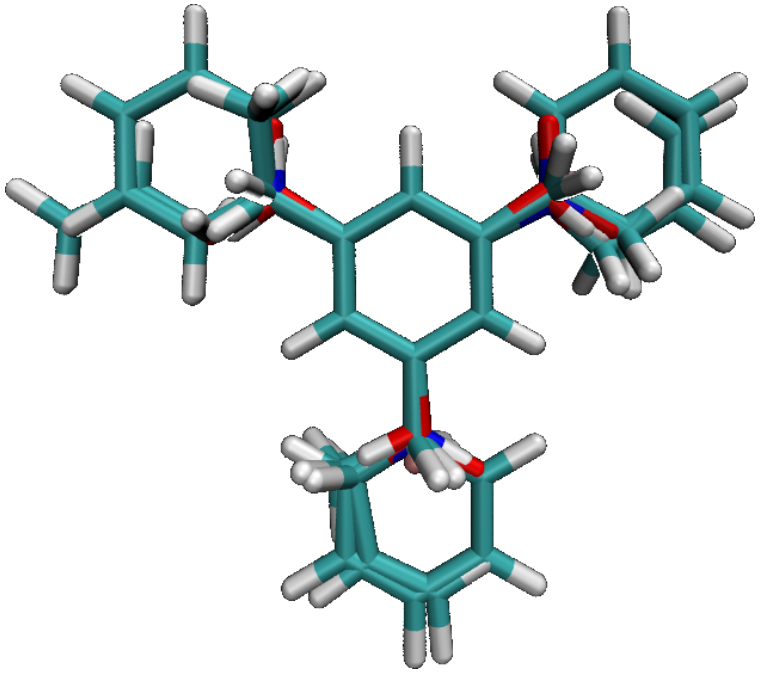
5.2 Theory

5.2.1 Defining the Chemical Space

We describe how to construct a simulation object that can physically model all molecules of interest. The chemical space we examine is a series of molecules constructed combinatorially from a common core. We start with the highly symmetric benzene ring as our core and mutate the R-groups at the 2, 4, and 6 carbons while fixing a hydrogen at the 1, 3, and 5 carbons. Changing which R-groups are interacting with the surrounding atoms at each site allows us to sample a combinatorially large set of molecules, while having a means to self-consistently validate our results as many of the combinations will be chemically identical to other combinations. We will have every R-group present in our simulation at the same time, but the extent to which all the atoms interact will be controlled by a dual topology approach [12]. The atoms on the same core carbon site but different R-groups will never interact to avoid steric collisions and simplify the chemical space we sample. Further details of atomic interactions are discussed in section 5.2.2. A model of our molecule is shown in Figure 5.1 in a 2D model, and a 3D representation as it will appear in a simulation. We call this molecular construct "the examol" as the size of the chemical space we would eventually like to explore is on the "exa-" (10^{18}) scale.



(a) Sketch version of examol



(b) Sketch version of examol

Figure 5.1: The examol molecule has all possible combinations of substituents present at the same time. The core structure, an aromatic ring, has multiple R-groups at each mutating site. (a) shows a sketch of each R-group and where it attaches. (b) shows all the R-groups attached as it would be seen in simulation. The simulation adjusts the extent to which each R-group interacts with its surroundings to tune exactly which state, and corresponding chemical, we sample at any given time.

For this test R-groups are chosen to occupy different volumes, have varying degrees of flexibility, and have wide ranging water solubilities. Figure 5.1a shows the core benzene ring and the 10 substituents: hydro-, hydroxyl-, methyl-, ethyl-, amino-,

bromo-, fluoro-, phenyl-, nitro-, and allyl ether. The bulky groups, like phenyl-, will be difficult to solvate in free energy simulations due to the size of the group [12]. Contrast this to the hydroxyl- group which has a small excluded volume in comparison, but has hydrogen bonding ability creating favorable interactions with a water solvent. Other substituents like the methyl- and amino- groups make for common R-groups in medicinal chemistry of drugs. This set of R-groups will provide a practical challenge to this method to test its viability on a diverse set of molecules. In practical application, the R-groups do not have to be as diverse as we are testing here. Due to the symmetry of the benzene core, with these 10 R-groups on the three sites, there are 220 unique molecules although there are 1000 combinations.

We represent the chemical space with a unique alchemical dimension for each R-group that we can sample independently. We first index each common core site with i for $N_i = 3$ total sites. We then index each R-group on each site with j for $N_j = 10$ R-groups. The jth index for any R-group type is the same, e.g. the hydro- group is always $j = 1$, the hydroxyl- group is always $j = 2$, etc. We will alchemically couple each R-group along its own independent λ variable, giving $N_i \cdot N_j = 30$ alchemical dimensions. This means that only a subset of atoms on the examol are alchemically coupled with any one λ. Contrast this to what we did in Chapters 2-3 where a complete molecule was alchemically coupled [54, 55] and Chapter 4 where we alchemically changed a single type of nonbonded interaction [56]. We will apply the basis function method from Chapters 2-3 to maximize statistical efficiency [54, 55]. Each individual alchemical dimension is represented by a 1-D λ variable which is also indexed by i and j. The collective symbol for all these variables is $\boldsymbol{\lambda}$.

The examol represents a realistic molecule when a single $\lambda_{i,j} = 1$ on every ith site, where all other $\lambda_{i,k\neq j} = 0$. For instance, if the methyl- is $j = 3$ and the hydro- is $j = 1$, the case where $\lambda_{1,3} = 1$, $\lambda_{2,1} = 1$, $\lambda_{3,1} = 1$, and all other $\lambda = 0$ is toluene.

5.2.2 Atomic Interactions

The examol requires a complex set of rules governing which atomic interactions are active. These rules are significantly more complicated than a standard alchemical free energy simulation that only controls what forces are coupled in what order. We detail the general rules here with specifics in Appendix D.

We exclude atomic interactions between atoms on different R-groups on the same ith carbon site. These interactions will cause steric collisions and having two or more R-groups fully coupled on the same core carbon site is a non-physical molecule which is not part of our chemically realistic end states. The pairs of excluded atoms ignore all nonbonded, angular, and torsional interactions between each other. We discuss how to avoid needlessly sampling states where these interactions would occur if we allowed them in section 5.2.4.

The common core benzene structure is not alchemically decoupled from the solvent like the R-groups are. This includes all six carbons and the three hydrogens at the 1, 3, and 5, positions. Only the Lennard-Jones repulsion and dispersion interactions are not alchemically changed as the partial charges depend on the R-groups. How the partial charges are assigned is discussed below.

We classify each pair of interacting atoms into one of three categories. Each category is treated differently to ensure the end states correctly represent real molecules while keeping the implementation as simple as possible.

- Alchemical/non-alchemical
- Alchemical/alchemical
- Non-alchemical/non-alchemical

We note that all harmonic covalent bonded terms are left fully coupled, regardless of interaction category. The persistent harmonic bonds ensure the examol does not

drift apart in the decoupled state, and the free energy differences from having excess harmonic energies can be corrected using vacuum simulations [195].

Alchemical/non-alchemical interactions are controlled through our minimal variance basis function approach. The alchemical/non-alchemical interactions include R-group interactions with the solvent and R-group interactions with the common core. It should be noted that intra-R-group interactions fall under the non-alchemical/non-alchemical category as we decouple the R-groups from their surroundings as opposed to annihilating all of the R-group's interactions. The nonbonded interactions are controlled by the R/C/AE-WCA basis function pathway [54, 55]. Just as we did in Chapter 3, each λ_{ij} has a $[0, 1]$ domain and acts as a 1-1 map to each of the individual forces along the basis function path. These forces are the Weeks-Chandler-Andersen decomposed Lennard-Jones repulsive and attractive forces [115], our statistically efficient capped repulsive Lennard-Jones force, and the electrostatic force. Angular and torsional interactions between atoms on an R-group and common core atoms are controlled through linear scaling the force with λ_{ij}, which also fits a generalized basis function form, but does not cause simulation instability as linearly changing nonbonded terms might [54, 55, 82, 83, 85, 95, 109, 195].

Alchemical/alchemical interactions are controlled through linear scaling of all forces simultaneously. This class of interactions are cross-R-group nonbonded interactions and electrostatics of the common core as the partial charges of the core atoms is a function of all R-groups. We theorize that linear scaling will be acceptably efficient for the cross alchemical terms as the atoms will infrequently occupy the same excluded volume, thus avoiding singularities in the energy. The purpose of the capped basis functions and the soft core potentials is to avoid such singularities when appearing or disappearing atoms in dense fluid [54, 55, 82, 83, 85, 95, 109, 195]. However, these atoms on separate R-groups will not reside on top of each other very often and will have to overcome the barriers of the solvent and all the interactions of other R-groups

from the same site repelling the R-groups on other sites. We have also found through short simulations that the linear coupling path does not cause simulation instability, and reduces the computational effort due to having fewer basis functions to compute. It should be noted we could control this class of interactions through a capped basis function pathway, which may have been required if all six core benzene sites could be mutated. For these interactions, the effective λ parameter, λ_{eff}, which is passed to the linear alchemical switch is

$$\lambda_{\text{eff}} = \lambda_{i,j} \cdot \lambda_{k,l} \text{ for } i \neq k \tag{5.1}$$

where $\lambda_{i,j}$ and $\lambda_{k,j}$ are the two alchemical parameters assigned to the R-groups interacting. $i \neq k$ is required since R-groups on the same site do not interact. However, j and k can take any of the N_j values, including $j = k$.

The non-alchemical/non-alchemical interactions have the simplest rules and are identical to non-alchemical simulations. The atoms in this set are the solvent/solvent interactions, the intra-R-group interactions, and the common core Lennard-Jones interactions with solvent and itself. These are controlled through standard bonded, 12-6 Lennard-Jones, and point-charge electrostatic forces. No modification to these are needed as these interactions are not alchemically controlled.

The partial charge on the core changes with the thermodynamic state of the R-groups. To determine the partial charge of the core, we approximate the charge as a linear combination of the partial charges if each R-group was present by itself. We start with the core where we have neutralized at all but one mutation sites. The other mutation sites are treated as a united atom carbon site. We then attach a single R-group at the mutation site and assign charges with the AM1-BCC [196] method. We then repeated this process for every R-group at every site. The partial charge of the core at any point in the simulation is then the sum of each step of this process, scaled

by $h_E(\lambda_{i,j|E})$ which is a map of the 1-D variable $\lambda_{i,j}$ to the electrostatic switch. This keeps a net neutral molecule which retains neutrality as each λ changes independently.

5.2.3 Chemical Sampling with λ-dynamics

We sample the examol with multiple-time step hybrid MC/λ-dynamics simulations; for shorthand, we call it "λ-dynamics for Examol," or "λDX." This carries out dynamics in both chemical and Cartesian space at the same time. The energy is described by both the coordinates of the atoms, and the current chemical state. Proposed Monte Carlo (MC) moves would therefore depend on spaces. Chemical space MC moves must be small (only fractions of the path between molecular end states) if they are accepted with reasonable probability [61, 121, 197–201]. Given the coupled nature of Cartesian and chemical spaces, and the difficulty of changing chemical state by itself, we choose to propose moves in both simultaneously.

We propose moves through λ-dynamics [51, 52, 202] where both Cartesian coordinates, and chemical coordinates are updated at the same time. Each λ variable is assigned a fictitious mass and is subject to a Hamiltonian of

$$\mathcal{H}(\boldsymbol{\lambda}|r) = \sum_i \sum_j \frac{m_{i,j}}{2} \lambda_{i,j}^2 + U(r, \boldsymbol{\lambda}) \tag{5.2}$$

where $m_{i,j}$ is the fictitious mass assigned to each λ variable and $U(r, \boldsymbol{\lambda})$ is the potential energy. We note that $\mathcal{H}(\boldsymbol{\lambda}|r)$ shows this is the Hamiltonian acting only in the chemical space, and that the full Hamiltonian acts in both Cartesian and chemical space. Because the λ variables have a mass, they can be treated just like another particle in the system: they experience force, have a velocity, and have a position. This allows the λ variables to be updated alongside Cartesian variables. We constrain our λ-dynamics positions by reflective boundary conditions with hard walls at $\lambda_i = [0, 1]$. This keeps the λ coordinates within their domains without us having to define an oscillating

function such as $\lambda_i = f(\theta_i)$, $-\infty < \theta < \infty$ which would bias values away from a uniform distribution of λ sampling.

We wrap a MC move around the λ-dynamics simulations to improve simulation stability. Hybrid MC (HMC) is a technique where the proposed MC move is taken after a short series of molecular dynamics (MD) time steps [193, 194, 203]. Rejections in HMC reject the entire MD series and reset positions to before the MD steps were taken. HMC helps reduce numerical instability from finite MD step while also preserving the correct thermodynamic ensemble, regardless of step size. λ-dynamics has known simulation instability and error in free energy estimates based on the choice of the fictitious alchemical mass [51, 52]. HMC allows us to overcome both of these issues and the alchemical mass now becomes a HMC acceptance rate tuning parameter. A larger mass causes slower changes in chemical state, and thus slower energy changes, increasing acceptance rate. Smaller masses allow faster energy changes, and thus lower acceptance rates. We tune the alchemical mass to try and approach the optimal HMC acceptance rate of 65.1% [204].

We accelerate our simulation by simulating with a multiple time step MC. λ-dynamics adds additional computational overhead in that we have to compute a force with respect to each alchemical parameter, $\partial u/\partial\lambda$, on top of computing a force in the Cartesian directions. One way to accelerate this process is to evaluate alchemical force on a smaller subset of atomic interactions. We choose to sample from an 'approximate' potential [205] to carry out MD with fewer atomic interactions, then analytically correct for the approximation with an additional MC step. This approximate MC sampling is evaluated in the following way: The initial non-approximate energy is evaluated, E_O. We then evaluate the same configuration under the approximate energy, which we label with a primed notation ($'$), E'_O. We carry out any type of MC sampling as an inner loop under the approximate potential and generate a new state with energy E'_N. The inner MC moves are accepted with the standard Metropolis

criteria of

$$\alpha'_{O\rightarrow N} = \min\left(\exp\left[-\beta(E'_N - E'_O)\right], 1\right) = \min\left(\exp\left[-\beta\Delta E'\right], 1\right) \tag{5.3}$$

where $\beta = (k_B T)^{-1}$. We can then ensure the correct thermodynamic ensemble under the full energy is preserved by evaluating the full energy after the inner moves, E_N, and accepting the whole procedure with the probability

$$\alpha_{O\rightarrow N} = \min\left(\exp\left[-\beta\Delta E\right]\exp\left[+\beta\Delta E'\right], 1\right) = \min\left(\exp\left[-\beta(\Delta E - \Delta E')\right], 1\right). \tag{5.4}$$

The derivation for Eq. 5.4 is general for any type of inner MC, including HMC [205], so this approach will work for our Hybrid MC/λ-dynamics sampling. We also never have to evaluate $\partial u/\partial\lambda$ for the full energy with this scheme since the outer MC evaluation only needs the total energy, not the force.

The approximate MC sampling becomes multi-time step sampling with the correct choice of approximate potential. If we choose our approximate potential to be only alchemical/non-alchemical and non-alchemical/non-alchemical type interactions, we only have to compute λ-dynamics forces on the alchemical/non-alchemical interactions. This means that we do not have to worry about computing the chain rule derivative from the alchemical/alchemical interactions due to Eq. (5.1). We theorize that alchemical/alchemical interactions are small in magnitude and do not frequently interact strongly, meaning they vary more slowly than the alchemical/non-alchemical interactions. If this theory is correct, the approximate MC sampling effectively turns this process into a multiple time step procedure.

Our λDX method overcomes the limitations with each of the individual sampling techniques that construct it. Sampling in chemical and Cartesian space simultaneously with λ-dynamics lets us avoid pre-populating MC moves in chemical space. The HMC move turns the λ-dynamics of the fictitious mass in chemical space into an acceptance

rate tuning parameter. Finally, the multiple-time step sampling of the approximate MC sampling lets us reduce computational overhead from computing derivatives with respect to λ while still sampling the correct ensemble.

5.2.4 Biasing Potential

The multidimensional nature of the examol implies sampling by diffusing through chemical space alone will take excessive computational time. If we assume that each λ can only take 11 uniformly spaced values on a $[0, 1]$ domain, a random walk simulation will take an expected $> 5 \cdot 10^{32}$ steps to visit each state in our 30 dimensional space once [206, 207]. This number of steps assumes equal probability of visiting each state, and does not account for natural kinetic barriers which appear in simulations of real molecules. We do not want to sample the chemical space where $\lambda_{i,j}$ and $\lambda_{i,k}$ are both large, meaning two R-groups on the same site are coupled. We propose two types of biases: a free energy bias to overcome the free energy barriers, and a fixed flat-bottom multidimensional harmonic bias to reduce the chemical space we need to sample.

A free energy bias along each R-group enhances sampling by reducing the time we spend kinetically trapped in low free energy states. We estimate the free energy at regular intervals during the simulation and approximate the bias on a cubic spline in each λ variable by itself. We then estimate the bias anywhere in the chemical space as a linear combination of each spline. Our free energy bias is then

$$B_F(\boldsymbol{\lambda}) = -\sum_{i}^{N_i}\sum_{j}^{N_j} F_{i,j}(\lambda_{i,j})|_{\lambda=0 \forall \lambda \neq \lambda_{i,j}} \tag{5.5}$$

That is the free energy bias, $B_F(\boldsymbol{\lambda})$, is the sum of free energies over all $\lambda_{i,j}$ when all other $\lambda = 0$. We show this free energy bias in Fig. 5.2 where our approximation is the middle pane. This approximation is a compromise between the two extremes of free energy bias. If we wanted to be exact, we would estimate the free energy at every

combination of $\boldsymbol{\lambda}$, as shown on the right pane of Fig. 5.2. However, this would be computationally taxing as this problem scales with the curse of dimensionality and we do not know the exact values of $\boldsymbol{\lambda}$ that λ-dynamics will sample *a priori*. The other extreme, as was implemented by the original multisite λ-dynamics publications [51, 52] and shown on the left pane, linearly scales the free energy at the end state of each λ so the bias is $\lambda_{i,j}F(1)$. Although this is computationally efficient, it can be woefully inaccurate as shown in our example figure. We feel our choice is a decent compromise since we do not need to bias with the exact free energy to escape kinetic traps, but we do still want to capture some features of the free energy surface.

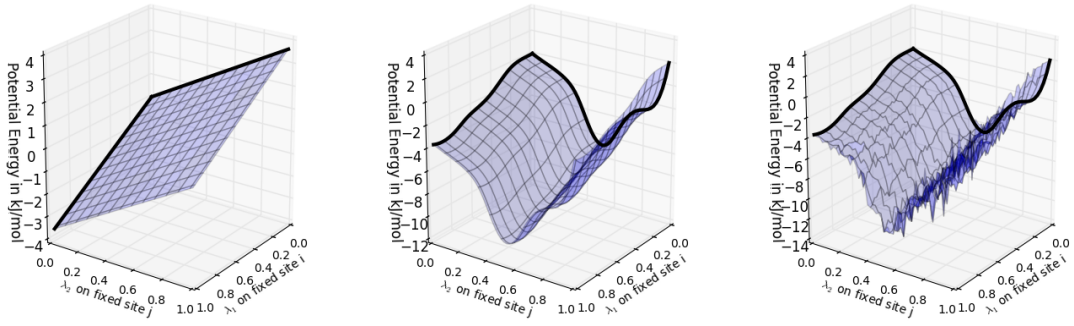


Figure 5.2: The bias of any point in chemical space is the linear combination of the free energy along each axis in chemical space. Shown are several ways to handle the bias for two λ. The left pane shows an extreme approximation where the bias is a linear scaling of fully coupled free energy, $\lambda_{i,j} \cdot F(1)$. The right is an exact calculation where the free energy surface is computed at every combination of $\lambda_{i,j}$ and $\lambda_{k,l}$. The middle is a compromise where the free energy is computed along each λ assuming all other $\lambda = 0$, then the free energy in the middle is the linear combination of the two, as in Eq. (5.5). We compute our free energy bias with the middle plot's method as a trade off between accuracy and computational effort.

We do not need to sample all of the possible chemical space to estimate free energies at the real molecule end states. The free energy bias lets the simulation escape kinetic wells, but we also want to limit what regions of chemical space to sample. Simply excluding the interactions between R-groups on the same common carbon site does not prevent the λ-dynamics from sampling a state where multiple λ on the same site are

large. If anything, excluding the interactions reduces the kinetic barriers to sampling those states since steric collisions no longer occur.

We do not want to sample states where multiple R-groups are fully coupled on one site as doing so generates configurations which share little phase space overlap with the end states. When multiple R-groups are coupled on the same site, there are additional angular and torsional forces the core experiences. We observed that these extra forces overcome the rigidness of the planar aromatic ring and cause ring puckering. This puckered benzene ring would virtually never be observed with any end state, and thus would share little phase space overlap. Multiple coupled R-groups also cause a concentration of sites where nonbonded interactions occur. These sites cause excessive solvent packing (in the case of favorable interactions), or solvent depletion around (in the case of unfavorable interactions). As we observed in Chapter 4, tight packing or depletion of solvent drastically reduces the phase space overlap to a reference state and takes significant more samples in these outlier states to converge simulations. Lastly, the basis functions were designed to be minimum variance moving from the decoupled state to the fully coupled state, so we theorize that moving through the decoupled state to chemically realistic molecules would take the most advantage of that design.

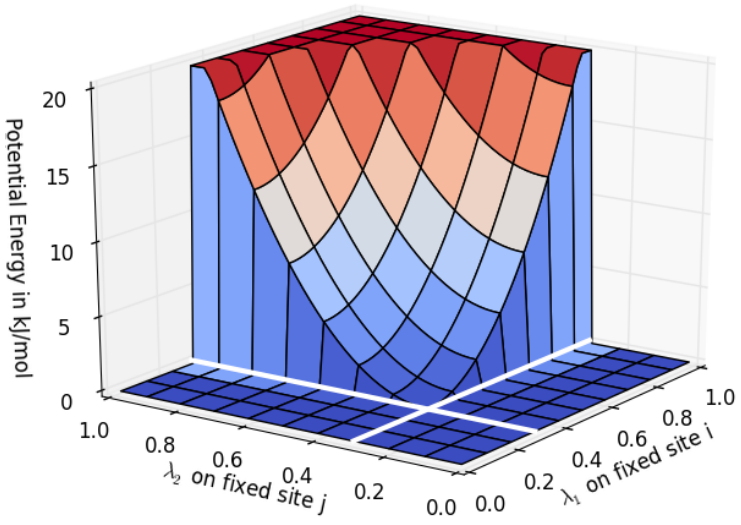
We prevent our simulation from sampling states where multiple R-groups are fully coupled on a single site by a flat-bottom harmonic restraint. Our bias is 0 near the decoupled states so we can better sample the decoupled region of chemical space. To sample the chemically realistic end states, we apply a harmonic bias when two $\lambda_{i,j}$ on the same site become large to improve sampling when one λ is large, but not multiple. We propose two forms of the flat-bottom harmonic bias equations, each with their own strengths and weaknesses:

$$B_{H,1}(\boldsymbol{\lambda}) = \sum_{i}^{N_i} \sum_{j}^{N_j} H\left(\lambda_{i,j} - \lambda_{\min 1}\right) \sum_{k \neq j}^{N_j} KH\left(\lambda_{i,k} - \lambda_{\min 2}\right) \left(\lambda_{i,k} - \lambda_{\min 2}\right)^2 \tag{5.6}$$

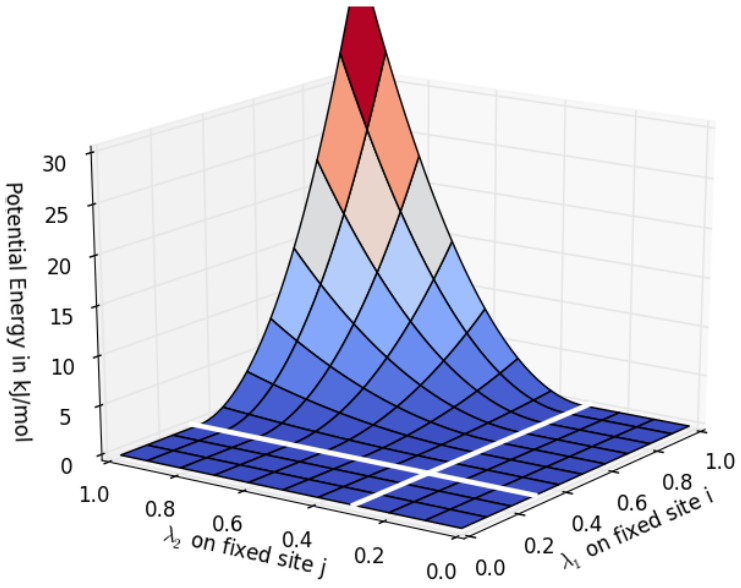
and

$$B_{H,2}(\boldsymbol{\lambda}) = \sum_{i}^{N_i} \sum_{j}^{N_j} H\left(\lambda_{i,j} - \lambda_{\text{min1}}\right) \left(\lambda_{i,j} - \lambda_{\text{min1}}\right)^2 \times \sum_{k \neq j}^{N_j} KH\left(\lambda_{i,k} - \lambda_{\text{min2}}\right) \left(\lambda_{i,k} - \lambda_{\text{min2}}\right)^2 . \tag{5.7}$$

H is the Heaviside step function, K is a constant bias in units of energy per λ^2 (or per λ^4 for $B_{H,2}$). If one R-group is near the fully coupled state ($\lambda_{i,j} \geq \lambda_{\text{min1}}$), other R-groups on the same site approaching the fully coupled state become disfavored ($\lambda_{i,k} \geq \lambda_{\text{min2}}$), driving the system away from the fully coupled state of multiple R-groups. We show an example of both biases in Fig. 5.3. This figure shows how the bias drives the λ-dynamics walkers towards either the decoupled state, or to the chemically realistic end states.



(a) Harmonic Bias 1



(b) Harmonic Bias 2

Figure 5.3: The flat-bottom bias keeps the simulation near realistic chemical species and along the pathways the linear basis functions were derived from. However, the bias only applies to λ on the same core site. The white lines show where the flat-bottom thresholds are located. (a) is a bias which creates hard walls at the thresholds to ensure no sampling is done above the two thresholds. (b) is a bias with steady, first derivative continuous walls above the thresholds, but requires a large force constant to avoid sampling just above the thresholds.

The first bias shown in Fig. 5.3a, $B_{H,1}$, creates strong walls at the thresholds and ensures minimal to no sampling is done anywhere above the thresholds. However, this bias is not first derivative continuous and MC moves which propose a state above the thresholds may have large rejection rates due to the sudden jump in energy. The second bias shown in Fig. 5.3b, $B_{H,2}$, smoothly increases above the threshold so small MC proposals above the thresholds will be accepted more often than $B_{H,1}$. The second harmonic bias will sample a larger chemical space near the decoupled regions as can be seen near the intersection of the white lines in Fig. 5.3b. In both cases, K should be chosen such that the free energy bias cannot overcome the flat-bottom harmonic bias so we retain control over the chemical space we sample. Note the actual value of K used to create Fig. 5.3b was chosen to better see the shape of the bias and lower than what would be done in simulations. Both the flat-bottom harmonic biases and the free energy bias have analytical derivatives in $\partial u/\partial\lambda$ so we can evaluate the force in each λ direction.

We run our simulations with the first harmonic bias, $B_{H,1}$, to ensure we sample the minimal chemical space. If the MC rejection rate caused by $B_{H,1}$ is too large, we theorize we can restart simulations from the decoupled state to sample near the decoupled state and possibly new end states from our λDX walkers. The increase in chemical space that $B_{H,2}$ introduces would be too large for practical sampling. Consider the case of $\lambda_{\text{min1}} = \lambda_{\text{min2}} = 0.2$ of Eq. (5.7), then increasing the threshold by 0.1. The accessible volume which could be sampled would increase by a factor of $(0.3/0.2)^{30} > 1.9 \cdot 10^5$. We feel the trade off of more MC rejections is better than the increased accessible volume.

5.2.5 Free Energy Calculation

Free energy differences are evaluated in a separate process from the λDX simulation. The simulations generate samples which we analyze in MBAR [101] to compute the

free energy. The potential energy of every sample is evaluated at every sampled state through linear algebra under our basis function approach [54, 55].

The free energy analysis provides both the adaptive bias and the chemically realistic end state free energies. Once MBAR is solved, we can generate the $N_j^{N_i}$ combinatorial end state free energy differences. These free energy differences (and their uncertainty) provide insight into the convergence of our simulation. Beyond ensuring we have global phase space overlap of our sampled states (see Chapter 4) [56], we can ensure we have sufficient sampling once chemically identical molecules have the same free energy differences when constructed from the same R-groups on different core carbon sites. We also can solve for the free energy biases needed for Eq. 5.5 which we can write to file, and have the running λDX simulation update its internal biases on the fly.

5.3 Simulation Setup

The exact implementation details of our λDX simulation can be found in the appendices. Here we cover the general details, omitting exact code modifications.

The λDX simulation was implemented in a modified version of OpenMM 7.0 [32, 39]. The examol was constructed by building a single R-group on a core benzene ring inside Maestro from the Schrödinger company. Partial charges were assigned by AM1-BCC [196]. The common cores of each R-group were aligned and the R-groups were then attached to a single common core inside OpenMM. All atomic parameters were taken from the GAFF force field [73] except for partial charges. The examol was solvated in TIP3P water with built-in OpenMM methods so there was at least 1.2 nm of water between any edge of the examol and the simulation box edge.

Simulations were carried out in the NVT ensemble at 298 K. The MD steps of the inner HMC loop were carried out under NVE ensemble, then velocities were drawn

from the Maxwell-Boltzmann distribution to preserve temperature after an outer MC evaluation, regardless of acceptance. This process differs from the massive Andersen thermostat [126] in that velocities were also drawn for the alchemical particles for λ-dynamics. A custom time integration algorithm was written to carry out the λ-dynamics.

The simulation was carried out with a mixture of PME and reaction field electrostatics. The water electrostatics were handled by PME with a relative error tolerance of $5 \cdot 10^{-4}$. Electrostatics of the examol were handled with a reaction field electrostatics [208] with a dielectric of 78.3. Due to implementation restrictions, the contribution of the PME method's long range, reciprocal space electrostatic interactions of the examol with periodic copies could not be isolated for basis function analysis, so reaction field electrostatics were employed instead. We would have been able to separate out the basis function reciprocal space electrostatics for a single R-group as we had done previously [54, 55] but not for all the different alchemical/alchemical interactions in this case. The cutoff for all nonbonded interactions electrostatics was 9Å. The water hydrogens were constrained by the SETTLE algorithm [125] and all other bonded hydrogens were constrained by the SHAKE algorithm [124].

Short simulations were run to determine the parameters which maximize the mean $\Delta\lambda$ constrained by the 65% optimal HMC acceptance rate [204]. The alchemical mass was set to $50\,\mathrm{amu} \cdot \mathrm{Å}^2$ for each parameter. Proportional distribution of mass was found to have a smaller $\Delta\lambda$. The harmonic bias force values were $K = 50\,\mathrm{kJ/mol}\ \lambda^2$ and $\lambda_{\mathrm{min1}} = \lambda_{\mathrm{min2}} = 0.3$. 10 MD steps were taken per MC evaluation for the inner HMC steps, and 1 inner HMC evaluation was evaluated per outer MC evaluation of the full potential. MD time steps were 1.25 fs and 500 total MD time steps were drawn between each recorded sample to better capture decorrelated samples [143].

Initial positions of the λDX simulation were taken from an equilibration simulation at fixed $\boldsymbol{\lambda}$. An equilibration simulation was run with only MD to generate a relaxed

starting conformation. $\lambda = 0.5$ was set and fixed for all λ in this simulation and 2 ns of the equilibration simulation was run. The relaxation period of this simulation was evaluated with the *timeseries* module packaged with MBAR [101, 143] and the configuration which was closet to the mean energy post-relaxation was used as the starting configuration for production simulations.

5.4 Discussion

The basis function method is many orders of magnitude faster at computing energies at all states over traditional simulations. For the purposes of analysis, we report data from $N = 10133$ drawn samples at a total of $K_s = 5896$, we also choose a reference state of all R-groups fully decoupled for measuring free energy differences. We draw the timing for traditional energy evaluation of running simulation code force loops from Chapter 4. This gives a conservative estimate of 30000 energy evaluations in 1500 CPU seconds and equates to 0.05 seconds per energy evaluation. We analyzed our data at $K_u = 10^3$ chemically realistic end states. $N(K_s + K_u)$ energy evaluations under traditional means would take 40.44 CPU days. Our implementation's computational effort scales with the number of basis functions so is a rough function of N_λ. At $N_\lambda = 30$, it took 43.77 CPU seconds to evaluate NK_s energies, at a rate of 0.00742 CPU seconds per state. Making our total $N(K_s + K_u)$ effort 51.17 CPU seconds, or 68,281x speed-up over traditional methods. If we had instead simulated with all six core carbon sites mutable, we get $N_\lambda = 60$ and $K_u = 10^6$. With this scale up traditional simulations would take 16.15 CPU years to evaluate $N(K_s + K_u)$ energies. Our basis function method slows to 0.0309 CPU seconds per state for a total time of 8.63 CPU hours, which is a still 16,396x speed-up over traditional methods. These timings assume that we have not pre-populated the value of the alchemical switches at each state. If we assume that the switches are pre-computed while the simulations

are running, the time to evaluate $N(K_s + K_u)$ energies with our approach decreases to 3.42 CPU seconds (145,924x speed-up) and 3.469 CPU hours (40,564x speed-up) for $N_\lambda = 30$ and $N_\lambda = 60$ respectively. So we are significantly more computationally efficient compared to traditional energy evaluation methods. It is worth noting these speed-ups are independent of the convergence problem as a traditional simulation would still have lacking global phase space overlap with the same samples. If anything, our accelerated analysis means we can update the biases, and thus drive the simulation to escape kinetic barriers to sampling, faster than traditional simulations.

Adding in the time to generate samples, our λDX implementation is still a faster option than traditional simulations for predicting free energies in this large chemical space. Because we are performing λDX simulations, we have extra computational effort to generate a sample compared to traditional MD in the form of HMC calculations and force derivatives in λ dimensions. The mean time to generate a simulate a single time step (and sample) on traditional MD simulations is 0.029 CPU seconds, computed from the raw simulation data of Chapter 4. Our λDX simulations take an average of 0.926 seconds (wall clock, running on GPU) to generate a sample. This means our simulations execute 31.6x slower than traditional simulations. However, including the analysis time, which must be done no matter how the sample was generated, our method is still faster by 53.7x at worse, and 12,623x at best for the numbers stated above. Although comparing GPU wall clock time is not equal to parallelized CPU time, this is the closest comparison we can make as we assume each simulation package's code is optimized for the hardware we simulate on. Recall that traditional methods scale by $\mathcal{O}\left(N(K_u + K_s^2)\right)$ and the basis function method scales optimally as $\mathcal{O}\left(NK_s\right)$. So as more samples are drawn, the more proportional time will be spent on analysis than sample generation. Finally, we desire to approach "exa-" scale in unsampled states, K_u, meaning our method will scale even better to larger systems.

The large chemical space will take more samples than we have currently drawn to

converge. At the time of writing, the simulations had not converged by our metrics. We stress again that choice of λDX over traditional simulations for either simulation or analysis does not effect the convergence assuming the exact same samples were drawn. However, the λDX methods have the advantage in that the samples drawn are from a limited chemical space and the basis functions the λDX simulations are built around help reduce the number of samples that will be needed to converge [54, 55]. The eigenvalues of the overlap matrix for our drawn samples have repeated 1's to machine precision [101, 187], indicating that insufficient samples have been drawn to say we have global phase space overlap and better confidence in property estimations [56]. Fig 5.4 shows the free energy difference between our reference state and every realistic chemical end state that can be made in multiple combinatorial ways. The x-axis is the free energy difference of each unique molecule and the y-axis is the free energy difference of the chemically identical molecules constructed in a different combinatorial way. A converged simulation would have every point along the $y = x$ curve as chemically identical molecules should have the same free energy differences. The average difference between the x and y value of these points is 29.430 kT with an average uncertainty of 4.84 kT, with outliers discarded (and not shown) which would dominate the mean values. We do, however, observe that increasing samples does shrink the uncertainty. Initial estimates of the same plot with 5x fewer samples showed an average free energy difference > 80 kT with uncertainty on the same order of magnitude.

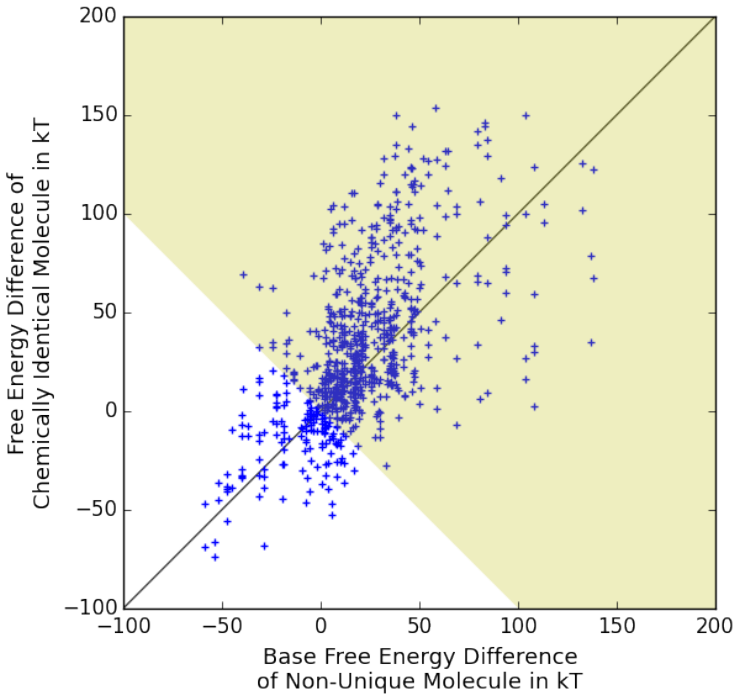
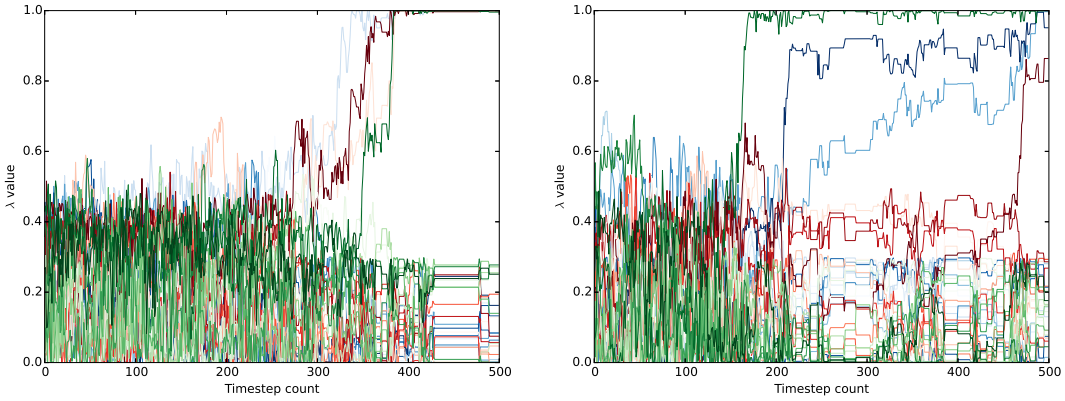


Figure 5.4: Many chemically identical molecules do not have the same free energy differences. This figure shows the free energy difference for each unique molecule combination on the x-axis and compares it to the free energy difference of other chemically identical molecules on the y-axis, made up of the same R-groups on different sites of the core. The reference state is the common benzene core with no coupled R-groups as an example. The shaded area shows molecules who's free energy difference to the reference state is positive for at least one of its combinations. Error bars are not shown for visual clarity.

The biases we apply reduce the chemical space while allowing the simulation to escape kinetic traps. Fig 5.5 shows the λ values from two different simulations. The λ lines are shaded based on what common core site the R-groups they represent are attached to, e.g. every shade of green line is attached to the 4 position on the benzene ring. The figures shows that simulations occasionally get kinetically trapped and all MC moves are rejected, as indicated by flat lines in the graphs. This is especially predominant in simulations without the adaptive free energy bias, an example of which is shown in Fig. 5.5a. The simulation becomes trapped towards the end for a large number of time steps. Contrast this with simulation where we have applied the free energy bias in Fig. 5.5b where the instances of trapped samples are much shorter. We

also see that the harmonic bias keeps walkers below the $\lambda = 0.3$ threshold as another walker on the same site approaches $\lambda = 1$, indicated by the fact that only one or two of the same base color are large at any one time. Once a λ approaches the coupled state though, it tends to remain stuck there. This is due to the HMC moves which move walkers above the harmonic bias threshold causing a sudden jump in energy, and thus are a rejected move. These rejections slow the speed at which we converge by limiting the extent of global phase space overlap we have between the reference state, and the end states. Specifically, the rejections reduce the number of samples we collect from any λ variable between 0.5 and 0.8, when the electrostatics are being coupled. Ways to address this problem are discussed in section 5.4.1.



(a) Early Simulation w/o FE bias (b) Later Simulation w/ FE bias

Figure 5.5: The λ parameters move through a chemical space controlled by the two types of biases. Each figure shows all 30 λ walkers over different simulations. The line colors indicated which common core site the R-group corresponding to each walker is attached to. Red, green, and blue lines show λ for R-group walkers on the 2, 4, and 5 site respectively, with intermediates colored lines between the three shades distinguishing between the R-groups on the same site. (a) shows a simulation with no adaptive free energy biases where the walkers get trapped towards the end of the simulation and all MC moves are rejected (flat line at > 400 samples) as there is no free energy bias to escape the kinetic traps. (b) shows a simulation where adaptive free energy biases are applied and ensure that a larger chemical space is sampled near the decoupled state (indicated by the spread of the walkers), and traps can be escaped more quickly as the kinetic barriers are reduced.

We can analyze the unconverged simulation data for screening purposes and as the

start of an optimization procedure from qualitative determinations. Points inside the shaded region of Fig 5.4 indicate a free energy difference > 0 relative to the reference state for any of the chemical combinations. In the drug design field, chemists typically would not want to synthesize drugs with low binding affinities and would instead focus on drugs with free energy differences < 0, rejecting all the molecules in the shaded region of the picture (within error). Although we only looked at solvation simulations, drug binding simulations would only cost 5-10x the length of simulation time relative to solvation properties [136], and solvation simulations are needed regardless for accurate and computationally efficient binding free energies. The binding simulations would also be run separately and through their own analysis code, so both solvation and binding simulations can run in tandem. Consider also liquid-liquid extraction where one would want positive solvation free energy differences between the carrier and the solvent, but negative solvation free energy differences between the solute and the solvent. A scatter plot like Fig 5.4 could be used to plan out which solvent to run, although the fluid in simulation should be the carrier fluid (instead of water here). The simulations could be modified to bias away from sampling molecules with have unfavored free energy, depending on application. Doing such biasing would shrink the chemical space we desired property estimations over, additionally reducing how much phase space overlap is needed. Our method works as a qualitative screening process so far, and will provide quantitative free energy, and other thermodynamic property estimate, as the property estimates converge.

5.4.1 Improving the simulation and convergence

The first harmonic bias, $B_{H,1}$ of Eq. (5.6), is too strong in the current λDX implementation. Fig 5.5 shows that once a λ walker approaches $\lambda = 1$, it becomes very hard for another λ to approach. The harmonic bias being the cause for this observation is enforced by Fig. 5.3. The sharp jumps in energy at the λ_{min} lines create a barrier

of at least 9.89 kT if one $\lambda = 1$, which causes a large rejection rate in the HMC moves. Although we do avoid sampling the chemical space where multiple $\lambda > \lambda_{\min}$ as seen in Fig. 5.5, we have virtually no samples for $\lambda > 0.5$ in any λ. Our next step will therefore be to apply the second harmonic bias, $B_{H,2}$ of Eq. (5.7). $B_{H,2}$ will allow smoother energy increases as multiple walkers pass over the thresholds, which should improve our MC acceptance rate. We will need to choose a new K constant to minimize the increase of accessible parameter volume.

Additional adaptive bias components would steer the simulations away from undesired molecules. As discussed above, one could add an additional bias potential which reduces the likelihood of sampling molecules with large, positive free energy differences (or negative, depending on application). Even something as simple as a harmonic bias on the end state itself, reducing the available chemical space to sample.

We are working with the OpenMM developers to implement parts of our approach as native functions within the software. The largest of these features is evaluating $\partial u/\partial\lambda$ for each λ variable. The approximate MC sampling [205] procedure is effectively required in our implementation since evaluating $\partial u/\partial\lambda$ for the alchemical/alchemical interactions adds $6.6x$ computational effort per MD time step with unoptimized code. We are working to add the ability to handle these derivatives natively, instead of our current implementation which requires us to effectively halt the simulation and to compute alchemical forces. The additional features will allow us to simplify the code defining the forces themselves, which will reduce the overhead in evaluating Cartesian force as well, further speeding up our simulations. Code simplification will not only speed-up the simulation, but it will also lower the barrier for someone else to implement our method in their own work.

5.5 Conclusions

We have developed a method to sample and estimate thermodynamic properties over a combinatorial chemical space that is both computationally and statistically efficient. This method required developing the basis function method to compute energies at any thermodynamic state through matrix multiplication instead of re-running simulation force code, a development which had been previously discarded by other computational free energy researchers because of low statistical efficiency. We overcame those statistical limitations and made a technique which can sample increasing larger chemical spaces with better computational efficiency than any previous multi-molecule sampling method. We also had to develop a way to represent a combinatorial chemical space inside a simulation with our examol. Multiple R-groups were combined onto a single core structure without unstable steric collisions. Finally, we created a new time integration algorithm to sample the chemical space by combining λ-dynamics, Hybrid Monte Carlo, and multiple time step MC sampling to explore only the subset of chemical space that represented real molecules in a computationally efficient manner. Each of these methods helps overcome the limitations brought on by the others to make stable simulation better suited to exploring large chemical spaces.

Our method has the potential to perform analysis in chemical spaces several orders of magnitude faster over traditional means. We have shown that our basis function method provides several orders of magnitude faster analysis than using traditional simulation methods to estimate energies. We have shown in Chapter 4 that the MBAR statistical estimator is one of the only reliable means of estimating thermodynamic properties over a large chemical space from a limited number of simulations. No matter how samples are drawn, analysis with MBAR requires specific energy evaluations, and our method is far superior at computing those energies. Although one could simply run 10^3 or 10^6 simulations, one for each chemical, our method lets us gather the same information from a single simulation. Furthermore, even though our simulations are

slower, they are still computationally faster than traditional simulations due to the time we save in analysis.

Chapter 6

Concluding Remarks

The basis function approach presented in this dissertation provides a computationally and statistically efficient way to estimate chemical thermodynamic properties in simulations. I have shown in Chapter 2 how the basis function method can provide a computationally efficient way to estimate thermodynamic properties on a single free energy difference calculation by reducing the cost of energy reweighting to simple matrix multiplication. The approach has the added benefit of being on equal statistical efficiency of the current best soft core methods. Chapter 3 generalized the basis function approach to develop the most efficient thermodynamic path for connecting end states, applying the basis functions to all types of nonbonded forces.

We can estimate properties over large parameter spaces with the basis functions and multistate reweighting techniques. By combining the basis function method with MBAR, Chapter 4 showed how to compute thermodynamic properties over vast atomic parameter space. Not only did the combination of the two methods allow efficient property estimates at over 260,000 parameter combinations, but the two methods provided an algorithm that creates global phase space overlap, a necessary component to accurate chemical property estimations. We must have good phase space overlap before we can make any accurate property estimate since even the uncertainty in our estimate is itself an estimate, and affected by the extent of phase space overlap.

Finally, this dissertation looked at extending the basis function approach to begin estimating thermodynamic properties over large chemical spaces. Hardware and software advancements have made estimating properties in chemical space possible, but there have been a lack of methods to do so. Chapter 5 developed a method to efficiently estimate properties over a combinatorial number of chemicals by applying the basis functions to a unique molecular representation. I developed the rules of atomic interactions, designed a special simulation protocol, and created an analysis procedure which takes advantage of the basis function approach to estimate free energies over a large number of molecules faster than any currently existing methods.

Even though additional simulation is required to generate quantitative estimates, I have shown how the basis function approach provides a way to start exploring the large chemical spaces we need to overcome the chemical engineering challenges of our modern world.

Chapter 7

Appendices

A.1 Derivation of linear basis function variance

In this section, we derive the variance of the free energy using the linear basis function approach with thermodynamic integration (TI) in the general case of multiple λ_i and multiple basis functions dependent only on coordinates as discussed in sections 2.2.3 and 2.4.3. This is an extension of the case with a single λ considered in previous work. [84]

Starting from the TI equation for the total free energy, with all λ_i parameterized by λ,

$$\begin{aligned}\Delta F &= \int_0^1 \frac{\mathrm{d}F}{\mathrm{d}\lambda}\,\mathrm{d}\lambda \\ &= \int_0^1 \sum_{i=1} \frac{\partial F}{\partial \lambda_i}\frac{\partial \lambda_i}{\partial \lambda}\,\mathrm{d}\lambda \\ &= \int_0^1 \sum_{i=1} \left\langle \frac{\partial \mathcal{H}}{\partial \lambda_i} \right\rangle \frac{\partial \lambda_i}{\partial \lambda}\,\mathrm{d}\lambda\end{aligned}$$

where $\mathcal{H}$ is the Hamiltonian, and F is either the Gibbs or the Helmholtz free energy, depending on the ensemble sampled. At a single point λ along this path, we can write an estimator for $\mathrm{d}F/\mathrm{d}\lambda$ as

$$\frac{\mathrm{d}F}{\mathrm{d}\lambda} = \sum_{i=1} \left\langle \frac{\partial \mathcal{H}}{\partial \lambda_i} \right\rangle \frac{\partial \lambda_i}{\partial \lambda} = \left\langle \sum_{i=1} \frac{\partial \mathcal{H}}{\partial \lambda_i}\frac{\partial \lambda_i}{\partial \lambda} \right\rangle. \tag{A.1}$$

Since we can write the estimator of $\mathrm{d}F/\mathrm{d}\lambda$ in terms of an ensemble average of some quantity, we can write the variance as

$$\begin{aligned}\mathrm{Var}\left(\frac{\mathrm{d}F}{\mathrm{d}\lambda}\right) &= \left\langle \left(\sum_{i=1}^{N} \frac{\partial \mathcal{H}}{\partial \lambda_i}\frac{\partial \lambda_i}{\partial \lambda} \right)^2 \right\rangle - \left\langle \sum_{i=1} \frac{\partial \mathcal{H}}{\partial \lambda_i}\frac{\partial \lambda_i}{\partial \lambda} \right\rangle^2 \\ &= \sum_{i,j=1} \mathrm{Cov}\left(\frac{\partial \mathcal{H}}{\partial \lambda_i}, \frac{\partial \mathcal{H}}{\partial \lambda_j} \right) \frac{\partial \lambda_i}{\partial \lambda}\frac{\partial \lambda_j}{\partial \lambda}.\end{aligned}$$

This can be compared to Eq. 5 of Crooks, [96] Eq. 2 of Shenfeld et al., [94] and

Eq. 42 of Gelman and Meng [93] to see that this total covariance is the square of the Riemannian metric used to measure thermodynamic length as:

$$\mathcal{L} = \int_0^1 \sqrt{\mathrm{Var}\left(\frac{\mathrm{d}F}{\mathrm{d}\lambda}\right)}.$$

To avoid cross correlation terms from expanding $(\partial\mathcal{H}/\partial\lambda_i)^2$ in terms of pair potentials such as $\sum_{k,l}(\partial u_k/\partial\lambda_i)(\partial u_l/\partial\lambda_i)$, we can rewrite the covariance in terms of pairwise functions. For ease of derivation, we first define the canonical partition function $Q(\lambda) = \int_\Gamma \exp(-\beta\mathcal{H}(\mathbf{x},\lambda))\,\mathrm{d}\mathbf{x}$, where coupling variables λ_i are functions of the single parameter λ and we then note that:

$$\begin{aligned}
\frac{\partial}{\partial\lambda_i}Q^{-1} &= -Q^{-2}\frac{\partial Q(\lambda)}{\partial\lambda_i} \\
&= -Q^{-2}\int_\Gamma \frac{\partial}{\partial\lambda_i}\exp(-\beta\mathcal{H}(\mathbf{x},\lambda))\,\mathrm{d}\mathbf{x} \\
&= -Q^{-2}\int_\Gamma -\beta\frac{\partial\mathcal{H}(\mathbf{x},\lambda)}{\partial\lambda_i}\exp(-\beta\mathcal{H}(\mathbf{x},\lambda))\,\mathrm{d}\mathbf{x} \\
&= \beta Q^{-1}\frac{\int_\Gamma \frac{\mathrm{d}\mathcal{H}(\mathbf{x},\lambda)}{\mathrm{d}\lambda_i}\exp(-\beta\mathcal{H}(\mathbf{x},\lambda))\,\mathrm{d}\mathbf{x}}{Q} \\
&= \beta Q^{-1}\left\langle\frac{\partial\mathcal{H}}{\partial\lambda_i}\right\rangle.
\end{aligned}$$

Although we use the canonical partition function Q, the results are equivalent in NPT or μVT ensembles as long as the paths only change the Hamiltonian, not the external thermodynamic parameters. We then examine the derivative $\frac{\partial}{\partial\lambda_i}\left\langle\frac{\partial\mathcal{H}}{\partial\lambda_j}\right\rangle$ to obtain the

covariance

$$\begin{aligned}
\frac{\partial}{\partial\lambda_i}\left\langle\frac{\partial\mathcal{H}}{\partial\lambda_j}\right\rangle &= \frac{\partial}{\partial\lambda_i}\left(\int_\Gamma \frac{\partial\mathcal{H}(\mathbf{x},\lambda)}{\partial\lambda_j}\exp\left[-\beta\mathcal{H}(\mathbf{x},\lambda)\right]Q^{-1}\,\mathrm{d}\mathbf{x}\right)\\
\frac{\partial}{\partial\lambda_i}\left\langle\frac{\partial\mathcal{H}}{\partial\lambda_j}\right\rangle &= \int_\Gamma \frac{\partial^2\mathcal{H}}{\partial\lambda_i\partial\lambda_j}\exp\left[-\beta\mathcal{H}(\mathbf{x},\lambda)\right]Q^{-1}\,\mathrm{d}\mathbf{x}\\
&\quad+\int_\Gamma -\beta\frac{\partial\mathcal{H}}{\partial\lambda_i}\frac{\partial\mathcal{H}}{\partial\lambda_j}\exp\left[-\beta\mathcal{H}(\mathbf{x},\lambda)\right]Q^{-1}\,\mathrm{d}\mathbf{x}\\
&\quad+\int_\Gamma \beta\frac{\partial\mathcal{H}}{\partial\lambda_j}\left\langle\frac{\partial\mathcal{H}}{\partial\lambda_i}\right\rangle\exp\left[-\beta\mathcal{H}(\mathbf{x},\lambda)\right]Q^{-1}\,\mathrm{d}\mathbf{x}\\
\frac{\partial}{\partial\lambda_i}\left\langle\frac{\partial\mathcal{H}}{\partial\lambda_j}\right\rangle &= \left\langle\frac{\partial^2\mathcal{H}}{\partial\lambda_i\partial\lambda_j}\right\rangle-\beta\left\langle\frac{\partial\mathcal{H}}{\partial\lambda_i}\frac{\partial\mathcal{H}}{\partial\lambda_j}\right\rangle+\beta\left\langle\frac{\partial\mathcal{H}}{\partial\lambda_i}\right\rangle\left\langle\frac{\partial\mathcal{H}}{\partial\lambda_j}\right\rangle.
\end{aligned}$$

This can then be arranged to form:

$$\begin{aligned}
\beta\left\langle\frac{\partial\mathcal{H}}{\partial\lambda_i}\frac{\partial\mathcal{H}}{\partial\lambda_j}\right\rangle-\beta\left\langle\frac{\partial\mathcal{H}}{\partial\lambda_i}\right\rangle\left\langle\frac{\partial\mathcal{H}}{\partial\lambda_j}\right\rangle &= \left\langle\frac{\partial^2\mathcal{H}}{\partial\lambda_i\partial\lambda_j}\right\rangle-\frac{\partial}{\partial\lambda_i}\left\langle\frac{\partial\mathcal{H}}{\partial\lambda_j}\right\rangle\\
\mathrm{Cov}\left(\frac{\partial\mathcal{H}}{\partial\lambda_i},\frac{\partial\mathcal{H}}{\partial\lambda_j}\right) &= \beta^{-1}\left(\left\langle\frac{\partial^2\mathcal{H}}{\partial\lambda_i\partial\lambda_j}\right\rangle-\frac{\partial}{\partial\lambda_i}\left\langle\frac{\partial\mathcal{H}}{\partial\lambda_j}\right\rangle\right).
\end{aligned}$$

Since $\mathrm{Cov}\,(x,y)=\mathrm{Cov}\,(y,x)$, and the partial derivatives of state functions are equal, we must also have

$$\frac{\partial}{\partial\lambda_i}\left\langle\frac{\partial\mathcal{H}}{\partial\lambda_j}\right\rangle=\frac{\partial}{\partial\lambda_j}\left\langle\frac{\partial\mathcal{H}}{\partial\lambda_i}\right\rangle.$$

Therefore, the total variance for the calculation of ΔF, assuming equal sampling at each point along the path, is

$$\mathrm{Var}\,(\Delta F)=\int_0^1\sum_{i,j}\mathrm{Cov}\left(\frac{\partial\mathcal{H}}{\partial\lambda_i},\frac{\partial\mathcal{H}}{\partial\lambda_j}\right)\frac{\partial\lambda_i}{\partial\lambda}\frac{\partial\lambda_j}{\partial\lambda}\,\mathrm{d}\lambda \tag{A.2}$$

$$=\beta^{-1}\int_0^1\sum_{i,j}\left(\left\langle\frac{\partial^2\mathcal{H}}{\partial\lambda_i\partial\lambda_j}\right\rangle-\frac{\partial}{\partial\lambda_i}\left\langle\frac{\partial\mathcal{H}}{\partial\lambda_j}\right\rangle\right)\frac{\partial\lambda_i}{\partial\lambda}\frac{\partial\lambda_j}{\partial\lambda}\,\mathrm{d}\lambda. \tag{A.3}$$

We will assume from here that the masses of all molecules are independent of λ. This makes the Hamiltonian's dependence on λ entirely through its potential energy, U, so $\langle\partial\mathcal{H}/\partial\lambda_i\rangle=\langle\partial U/\partial\lambda_i\rangle$ as the kinetic energy can be exactly accounted

for analytically. The variance can then be rewritten in terms of the complete radial distribution function (RDF) (i.e. not approximated) as

$$\begin{aligned}\mathrm{Var}\left(\Delta F\right) &= \beta^{-1}\int_0^1 \sum_{i,j}\left(\left\langle\frac{\partial^2 U}{\partial\lambda_i\partial\lambda_j}\right\rangle - \frac{\partial}{\partial\lambda_i}\left\langle\frac{\partial U}{\partial\lambda_j}\right\rangle\right)\frac{\partial\lambda_i}{\partial\lambda}\frac{\partial\lambda_j}{\partial\lambda}\,\mathrm{d}\lambda \\ &= 4\pi\rho\beta^{-1}\int_0^1\int_0^\infty \sum_{i,j}\left[\frac{\partial^2 U}{\partial\lambda_i\partial\lambda_j}g(r,\lambda)r^2 - \frac{\partial}{\partial\lambda_i}\left(\frac{\partial U}{\partial\lambda_j}g(r,\lambda)r^2\right)\right]\frac{\partial\lambda_i}{\partial\lambda}\frac{\partial\lambda_j}{\partial\lambda}\,\mathrm{d}r\,\mathrm{d}\lambda\end{aligned} \tag{A.4}$$

where $g(r,\lambda)$ is the RDF and ρ is the solvent number density. Applying the zeroth-order approximation of the RDF of $g(r,\lambda) \approx \exp\left[-\beta U(r,\lambda)\right]$ to the second term gives

$$\begin{aligned}\frac{\partial}{\partial\lambda_i}\left\langle\frac{\partial U}{\partial\lambda_j}\right\rangle &\approx 4\pi\rho\frac{\partial}{\partial\lambda_i}\int_0^\infty \frac{\partial U}{\partial\lambda_j}\exp\left[-\beta U(r,\lambda)\right]r^2\,\mathrm{d}r \\ &= 4\pi\rho\int_0^\infty \frac{\partial^2 U}{\partial\lambda_i\partial\lambda_j}\exp\left[-\beta U(r,\lambda)\right]r^2 - \beta\frac{\partial U}{\partial\lambda_i}\frac{\partial U}{\partial\lambda_j}\exp\left[-\beta U(r,\lambda)\right]r^2\,\mathrm{d}r \\ &= \left\langle\frac{\partial^2 U}{\partial\lambda_i\partial\lambda_j}\right\rangle - \int_0^\infty 4\pi\rho\beta\frac{\partial U}{\partial\lambda_i}\frac{\partial U}{\partial\lambda_j}\exp\left[-\beta U(r,\lambda)\right]r^2\,\mathrm{d}r.\end{aligned} \tag{A.5}$$

Substituting Eq. (A.5) into Eq. (A.4) gives the approximated variance equation of

$$\mathrm{Var}\left(\Delta F\right) \approx 4\pi\rho\int_0^1\int_0^\infty \sum_{i,j}\left(\frac{\partial U}{\partial\lambda_i}\frac{\partial U}{\partial\lambda_j}\exp\left[-\beta U(r,\lambda)\right]r^2\right)\frac{\partial\lambda_i}{\partial\lambda}\frac{\partial\lambda_j}{\partial\lambda}\,\mathrm{d}r\,\mathrm{d}\lambda \tag{A.6}$$

which reduces to Eq. (2.13) for one λ_i.

When the potential is represented with basis functions, we can replace all λ_i directly with h_i. Derivatives in h_i can also be removed from the covariance because they do not participate in the expectation integrals. We define a shorthand such that h_i is equivalent to $h_i(\lambda)$ and

$$\frac{\partial U(r,\lambda)}{\partial h_i} = \frac{\partial h_i}{\partial h_i}u_i(r) + \sum_{j\neq i}\frac{\partial h_j}{\partial h_i}u_j(r) = u_i(r).$$

Eq. (A.2) then simplifies to

$$\begin{aligned}
\mathrm{Var}\left(\Delta F\right) &= \int_0^1 \sum_{i,j} \mathrm{Cov}\left(\frac{\partial \mathcal{H}}{\partial h_i}, \frac{\partial \mathcal{H}}{\partial h_j}\right) \frac{\partial h_i}{\partial \lambda} \frac{\partial h_j}{\partial \lambda} \,\mathrm{d}\lambda \\
&= \int_0^1 \sum_{i,j} \left(\left\langle \frac{\partial U}{\partial h_i} \frac{\partial U}{\partial h_j} \right\rangle - \left\langle \frac{\partial U}{\partial h_i} \right\rangle \left\langle \frac{\partial U}{\partial h_j} \right\rangle \right) \frac{\partial h_i}{\partial \lambda} \frac{\partial h_j}{\partial \lambda} \,\mathrm{d}\lambda \\
&= \int_0^1 \sum_{i,j} \left\langle \frac{\partial h_i}{\partial h_i} u_i(r) \frac{\partial h_j}{\partial h_j} u_j(r) \right\rangle - \left\langle \frac{\partial h_i}{\partial h_i} u_i(r) \right\rangle \left\langle \frac{\partial h_j}{\partial h_j} u_j(r) \right\rangle \frac{\partial h_i}{\partial \lambda} \frac{\partial h_j}{\partial \lambda} \,\mathrm{d}\lambda \\
&= \int_0^1 \sum_{i,j} \frac{\partial h_i}{\partial \lambda} \frac{\partial h_j}{\partial \lambda} \left[\langle u_i(r) u_j(r) \rangle - \langle u_i(r) \rangle \langle u_j(r) \rangle \right] \mathrm{d}\lambda \\
&= \int_0^1 \sum_{i,j} \frac{\partial h_i}{\partial \lambda} \frac{\partial h_j}{\partial \lambda} \mathrm{Cov}\left(u_i, u_j\right) \,\mathrm{d}\lambda. \qquad \text{(A.7)}
\end{aligned}$$

Defining the covariance matrix for $\mathbf{u}$ as

$$\mathrm{Cov}\left(\mathbf{u}, \mathbf{u}\right) = \begin{bmatrix}
\mathrm{Var}\left(u_1(r)\right) & \mathrm{Cov}\left(u_1(r), u_2(r)\right) & \cdots & \mathrm{Cov}\left(u_1(r), u_n(r)\right) \\
\mathrm{Cov}\left(u_2(r), u_1(r)\right) & \mathrm{Var}\left(u_2(r)\right) & \cdots & \mathrm{Cov}\left(u_2(r), u_n(r)\right) \\
\vdots & \vdots & \ddots & \vdots \\
\mathrm{Cov}\left(u_n(r), u_1(r)\right) & \mathrm{Cov}\left(u_n(r), u_2(r)\right) & \cdots & \mathrm{Var}\left(u_n(r)\right)
\end{bmatrix},$$

allows the variance for basis functions to be written in condensed, matrix form as

$$\mathrm{Var}\left(\Delta F\right) = \int_0^1 \mathbf{h}'(\lambda) \cdot \mathrm{Cov}\left(\mathbf{u}, \mathbf{u}\right) \cdot \mathbf{h}'^{T}(\lambda) \,\mathrm{d}\lambda \qquad \text{(A.8)}$$

where $\mathbf{h}'(\lambda) = [\partial h_1/\partial\lambda, \partial h_2/\partial\lambda, \dots]$. This equation is also Eq. (2.18).

$\langle \partial u/\partial\lambda \rangle$ can also be simplified by applying the basis functions to Eq. (A.1) and assuming the Hamiltonian's dependence on λ entirely through its potential energy as

was done earlier. The result is

$$\begin{aligned}\frac{\mathrm{d}F}{\mathrm{d}\lambda} &= \left\langle \sum_{i=1} \frac{\partial \mathcal{H}}{\partial h_i} \frac{\partial h_i}{\partial \lambda} \right\rangle \\ &= \left\langle \sum_{i=1} \frac{\partial h_i}{\partial h_i} u_i(r) \frac{\partial h_i}{\partial \lambda} \right\rangle \\ &= \left\langle \mathbf{h}'(\lambda) \cdot \mathbf{u}^T \right\rangle \\ &= \mathbf{h}'(\lambda) \cdot \left\langle \mathbf{u}^T \right\rangle .\end{aligned}$$

Extending this equation to predict the expectation of an unsampled switch yields Eq. (2.23).

A.2 Deriving an efficient short range basis function

A rule of thumb for the short-range alchemical switch can be derived from Eq. (2.13) by assuming that when r is small, $u_i(r)$ for repulsive interactions will be a large, finite, and nearly constant value because of the capped potential. As seen in Eq. (2.13), the variance is a function of the product of $(\partial u/\partial \lambda)^2$ and $g(r)$. To minimize this product, we must have, in the two body regime, that:

$$\left(\frac{\partial U(r,\lambda)}{\partial \lambda} \right)^2 \exp\left[-\beta U(r,\lambda) \right] r^2 < V \tag{B.1}$$

Where V is approximately constant. If we assume that the potential and thus the variance do not change greatly with r^2 because of the cap, then this will be satisfied with:

$$\left(\frac{\partial U(r,\lambda)}{\partial \lambda} \right)^2 \exp\left[-\beta U(r,\lambda) \right] < V' \tag{B.2}$$

where V' is a new constant. We assume the energy is dominated by the repulsive term near $r = 0$, allowing us to rewrite the equation in terms of the repulsive basis function, u_R, and repulsive alchemical switch, h_R, as:

$$\left(\frac{\partial h_R(\lambda)}{\partial \lambda}\right)^2 u_R(r)^2 \exp\left[-\beta h_R(\lambda) u_R(r)\right] < V'. \tag{B.3}$$

We can move the exponential to the right hand side and, because the variance must always be positive, take the square root considering only the positive inequality. We then apply a Taylor series expansion to the first order of the zeroth-order RDF term and get

$$\frac{\partial h_R(\lambda)}{\partial \lambda} u_R(r) < V'' \left[1 + \frac{\beta h_R(\lambda) u_R(r)}{2}\right] \tag{B.4}$$

where V'' is another modified constant. Since we assumed $u_R(r)$ is large, $1/u_R(r)$ is small and we divide by $u_R(r)$ to obtain a condition for the $h_R(\lambda)$ of the repulsive term depending only on λ_i:

$$\frac{\partial h_R(\lambda)}{\partial \lambda} < V''' h_R(\lambda) \tag{B.5}$$

with V''' again absorbing other constants. If we make the rule of thumb inequality an equality and apply the boundary conditions on $h_i(\lambda)$, we obtain a solution to Eq. (B.5)

$$h_i(\lambda) = \frac{K^\lambda - 1}{K - 1} \tag{B.6}$$

where K is a positive free parameter that can be optimized. This equation is also Switch A, Eq. (2.19).

A.3 A brief comparison of variances in solvent and complex environment

Low variance paths for alchemical solvation should also be low variance paths for alchemical protein binding. Whether a ligand is surrounded by water or protein, the local density of fluid and therefore the number and size of excluded volume sites is roughly the same. The increased enthalpy of binding configurations would suggest that the variances might be somewhat greater, but the general shape will be determined primarily by the excluded volume and the number density around the solute, which remains roughly independent of the liquid environment.

A full examination of the difference of the variance of pathways in protein and solution does not appear to have been performed. There appears to be a general acceptance that the pathway that is most efficient for one composition will be most efficient for the other. Finding the most statistically efficient pathway to alchemically modify a ligand is frequently done only in one environment [57, 84, 85, 87, 88, 95, 97, 109] and the other environment is not extensively examined. Many simulation packages with soft core alchemical transformations assume a single pathway for soft core transformations, up to user specifications, regardless of the composition of the system. [86, 87, 110–112]

We examine a simple comparison of variance between pure solvent and host-guest systems here. This will not be a full study of the minimal variances between complex and solvent, but simply evidence that the low variance pathways developed here for solvent should be low variance paths in complex. We re-examine data from host-guest systems from Monroe and Shirts [209] where the guest molecule is alchemically solvated and alchemically removed from the host structure in separate simulations. Further experimental details and system information are provide in the reference.

Shown in Fig. A.1 is the variance of the coupling the repulsive and attractive

interactions with a soft core 1-1-6 interaction for several guests to the same host. Lines on the figure are drawn connecting data points, and unlike many of the graphs in this paper using the linear basis approach, should not be read as information about unsampled states.

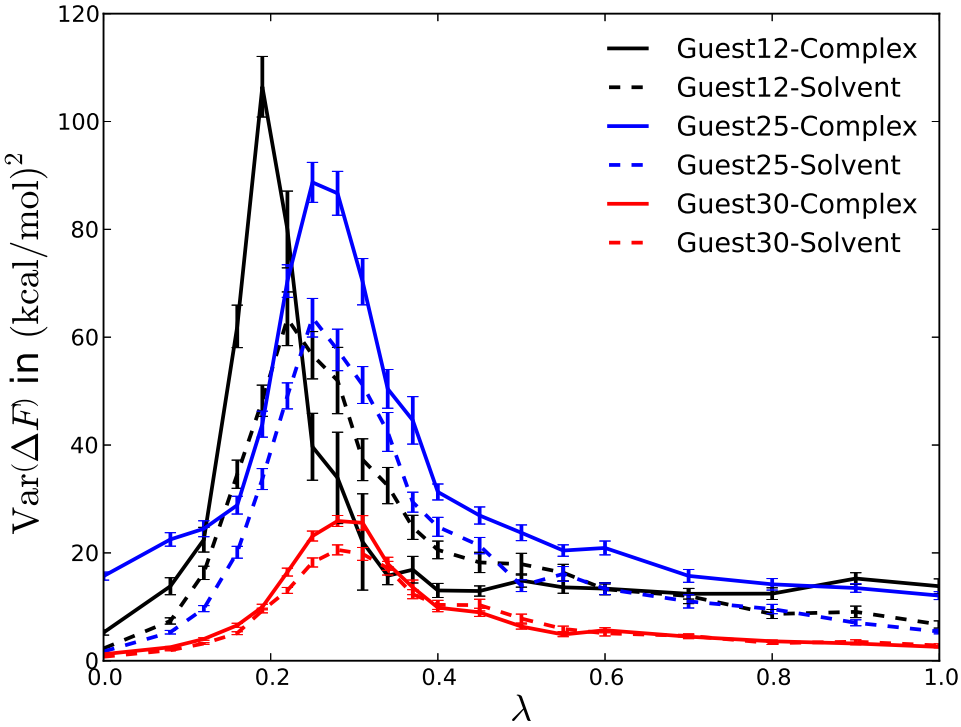


Figure A.1: **Low variance paths will be similar for solvation and protein binding.** The shape of the variance curve will be similar since adding a protein to the solvent will not significantly change the system density or response of fluid to the ligand. Shown here is the variance for solvating and binding three ligands in a host-guest system where the alchemical path and solute are held constant between environments. Lines are drawn to guide the eye and provide a shape of the variance curve but do not represent data between sampled points. Data are only available at points where error is estimated with 200 bootstrap samples. The variance is only shifted slightly in between the environments along λ, so regions of large variance are shared between environments. The magnitude of the variance and number of uncorrelated samples are different in complex than solution but the overlapping peaks means any variance minimization method will minimize the variance in both environments. Data for this figure generated from Monroe and Shirts. [209]

The regions of high variance overlap between solvent and complex environments. In the cases of Guest 25 and Guest 30, the maximum in the variance curve is almost

at the same λ, and for Guest 12, the peak is only shifted by a small $\delta\lambda$. A full proof would show that this very close correspondence between the shape of the variance peak hold for all pathways, not just a single pathway. However, the fact that the shapes are the same up to scaling factor expected because of the larger magnitude of ligand binding energies suggests that the shape of the variance curve for a pure solvent system is the similar to that of the ligand bound system.

The variance minimization procedures outlined in Chapter 2 are equally valid for any environment. If the minimizing the variance in solution environment does not provide the desired statistical efficiency, one can run a short, trial simulation with the system of interest and use Eq. (2.22) to find a low variance path. Attempting to find a true universally low variance path may not be possible, since changes in the environment can shift the locations of high variance as seen in Guest 12 of Fig. A.1. Even so, the flexibility and power of the linear basis method allows low variance switches to be chosen with ease, simplifying the choice of low variance paths for arbitrary systems.

B.1 Narrowing the choices for low variance schedules

B.1.1 The attractive force must be coupled after repulsive to prevent an attractive core

We note that the core of a particle must have a repulsive interaction to ensure an infinite attractive core does not form, which would cause integration instabilities. [82–85, 92, 97, 109] This applies not only at $r = 0$ but also for any off-atom sites (like lone pairs or the oxygen charge in 4 point water models) as they are protected by the repulsive cores of nearby atoms. Any schedule which has the attractive force coupled before the repulsive force need not be considered further, eliminating schedules such as AE/R/C or A/R/C/E. However, schedules where repulsive interactions are coupled simultaneously with attractive interactions (i.e. E/RA/C or RA/C/E) are retained since they can be decomposed by a Weeks-Chandler-Andersen (WCA) [115] decomposition which eliminates the infinite attraction of the r^{-6} term. Schedules which have attractive cores are categorized in Table 3.1 as "Attractive Core" schedules. Removing these reduces the total unique schedules to eight.

B.1.2 The repulsive core should be fully coupled before the electrostatics

The basis functions and schedules must ensure the magnitude of an attractive electrostatic potential never exceeds the magnitude of the Lennard-Jones repulsion at $r = 0$ to avoid an attractive core causing simulation instability. [92] There are two main ways to ensure this: cap the electrostatic interaction at finite values, similar to Eq. (2.10), or only couple electrostatics after the repulsive interactions have been uncapped. Although capped electrostatics can provide numerically stable simulations, [210–212]

they do not provide low variance coupling pathways. The total capped potential must be between $3.5k_BT$ and $8.8k_BT$ at $r = 0$, in order to provide a low variance pathway. [54] As both attractive and repulsive electrostatic interactions are possible, we must design basis functions to handle both, while also keeping the total capped potential inside the target range.

In order to design numerically stable electrostatic basis functions, we must first examine how electrostatics are handled in soft core, where repulsive and attractive interactions are treated identically. Soft core potentials require constrained constants to maintain numerical stability. Soft core methods must be designed to prevent double minimum from forming in the potential energy separated by large energy barriers. The primary cause of this double minimum is the electrostatic cap dominating the repulsive cap near $r = 0$ for cases of a soft core alchemical atom interacting with an atom holding charge, but no Lennard-Jones core, e.g. the hydrogen in TIP3P water. [92] The soft core Lennard-Jones potential takes the general form

$$u_{\mathrm{SC,LJ}}(r,\lambda) = 4\epsilon_{ij}\lambda^a \left[\left(\frac{1}{\alpha(1-\lambda)^b + (r/\sigma_{ij})^c} \right)^{12/c} - \left(\frac{1}{\alpha(1-\lambda)^b + (r/\sigma_{ij})^c} \right)^{6/c} \right]. \tag{B.1}$$

The electrostatic potential has a soft core form of

$$u_{\mathrm{SC,E}}(r,\lambda) = \lambda \frac{q_i q_j}{4\pi\epsilon_0 \left[(1-\lambda)\,\beta + r^m \right]^{1/m}}. \tag{B.2}$$

In Eq. (B.1) and Eq. (B.2), a, b, c, and m are positive, usually integer, constants, and α and β are positive free parameters. Choosing $a = b = 1$ is a statistically efficient choice, [84] leaving the other constants free for variance minimization and numerical stability analysis. In order to prevent a double minimum from forming using both

soft core equations together, the following constraints must be true [92]:

$$c = m \tag{B.3}$$

$$\alpha^{1/c}\sigma_{ij} \leq \beta^{1/m} \tag{B.4}$$

The basis functions must also prevent double minimum potentials, but have arbitrary functional form. Contrary to the soft core potentials, the basis functions do not require any specific functional form. However, a single set of basis functions is better than multiple system dependent sets, for simplicity in both use and implementation. Since a capped electrostatic potential is required to use with the capped repulsive basis function, the functional form for the electrostatics should be similar to Eq. (2.10) and is controlled by the same f_{cap} and f_{switch}, removing the option to define different parameters for different systems. This constraint could be removed, but then separate potentials would need to be defined for attractive and repulsive electrostatic interactions as will be shown.

To determine the necessary parameter values, and therefore the magnitude of the cap, we examine the limiting case of Li^+ and Fl^- ions interacting. These ions have small σ_{ij} terms, but also carry full ± 1 charges. The small σ_{ij} will allow the particles to approach closer, and therefore feel a stronger electrostatic force than typical solute-solvent systems with larger atom cores and partial charges. The required cap height in this extreme case will be more than sufficient for almost all other physically relevant systems.

The parameter f_{cap} must be smaller with an electrostatic cap than with a repulsive cap alone to for the transformation to have low statistical error. The OPLS-AA parameters for Li^+ are $\sigma_{ii} = 2.13$ Å, $\epsilon_{ii} = 0.0183$ kcal/mol, and for Fl^- are $\sigma_{ii} = 2.73$ Å, $\epsilon_{ii} = 0.720$ kcal/mol. [71] Choosing the basis functions for attractive and repulsive Lennard-Jones interactions as Eq. (2.9) and Eq. (2.10) respectively, the potential of

the cap for either the repulsive Lennard-Jones or electrostatic potential is

$$u_{i,\text{cap}} = u_i(f_{\text{cap}}\sigma_{ij}) \tag{B.5}$$

where i is either R or E, u_i is the uncapped basis function for the respective force. The standard point charge interaction formula is chosen for the electrostatic basis such that

$$u_E(r) = \frac{q_i q_j}{4\pi\epsilon_0 r}. \tag{B.6}$$

With these choices, a cap at $f_{\text{cap}} \approx 0.6$ is required to preserve a repulsive core at the capped values. This value ensures the capped repulsive Lennard-Jones interaction dominates the capped attractive electrostatic potential and keep the total potential in the target range at $r = 0$.

Repulsive interactions should be fully coupled with their caps removed before electrostatic interactions are coupled into the system. Although the total cap for an attractive electrostatic interaction is within our low variance target range, a repulsive electrostatic interaction is not. The Li^+/Li^+ interaction has a total cap of $397k_BT$ at $f_{\text{cap}} \approx 0.6$, which can significantly increase the variance of the free energy calculation along such a path. [54] To correct this issue, a separate f_{cap} and f_{switch} would be required to bring the total cap back into the range of $3.5k_BT$ and $8.8k_BT$. As an arbitrary system could have both repulsive and attractive electrostatic interactions, the basis functions would need to be designed and implemented with rules checking each interaction and using the appropriate parameters. This is a less desirable solution since it adds complexity to implementation and usability of the basis function method. It is much simpler if the electrostatics are coupled after the repulsive interactions.

B.2 Variances and Free Energies of Each Alchemical Schedule

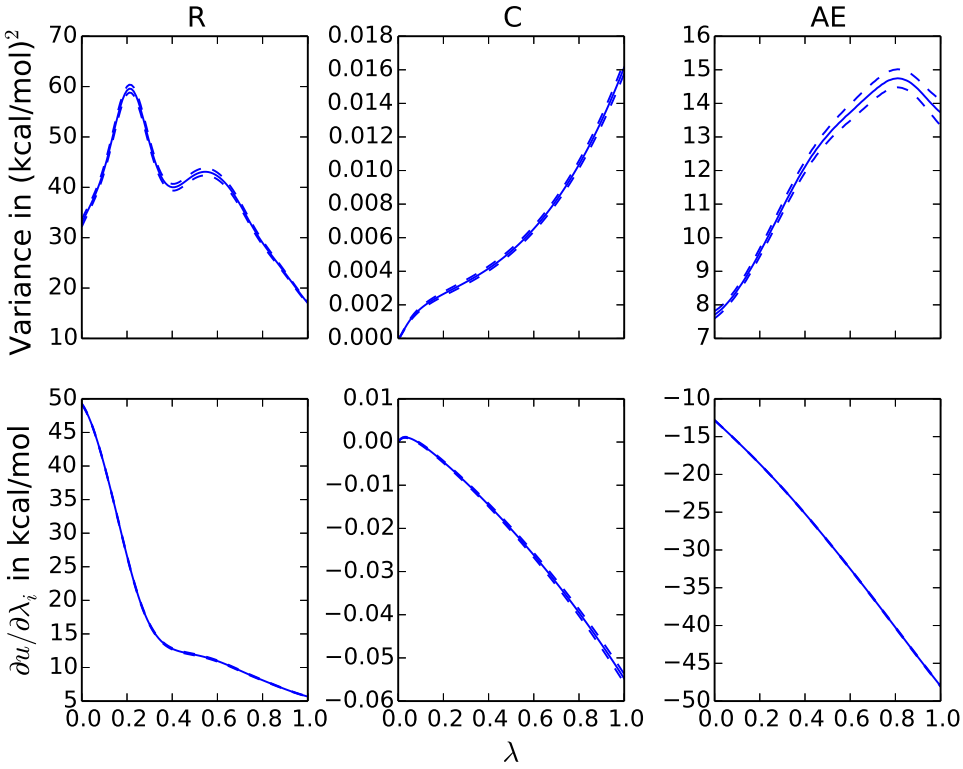


Figure B.1: Variance and $\partial u/\partial\lambda$ for the R/C/AE-WCA basis function schedule. Error is shown as dashed lines around the figure and often smaller than the thickness of the line. Total variance cannot be estimated by adding the integral under each curve as the number of samples from each stage is not taken into account.

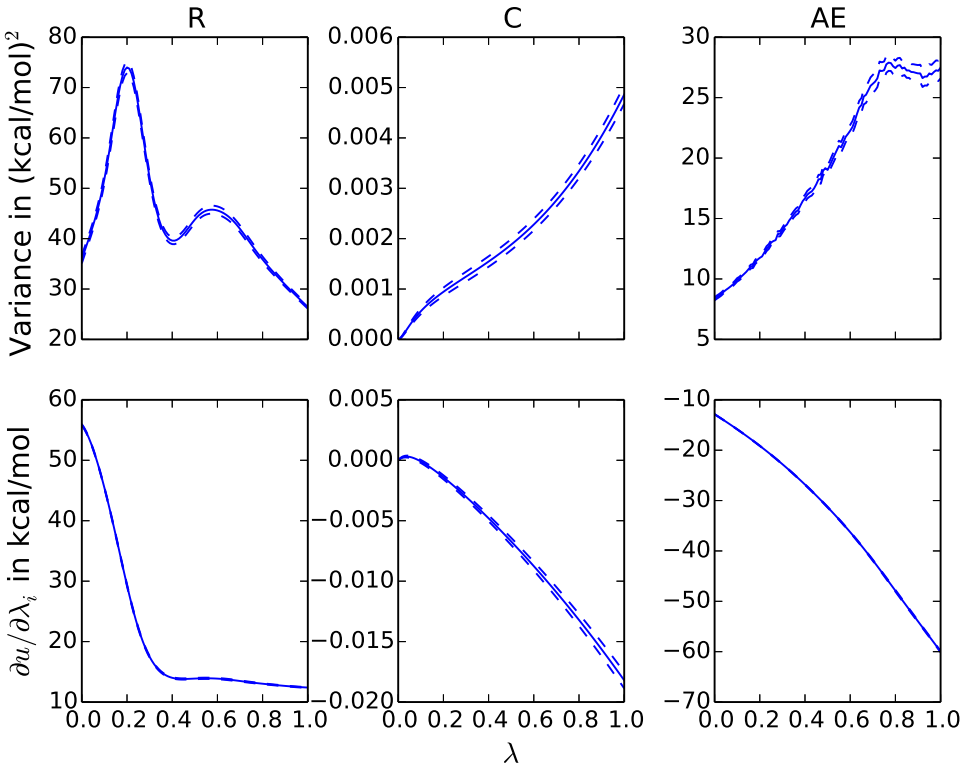


Figure B.2: Variance and $\partial u/\partial\lambda$ for the R/C/AE-12-6 basis function schedule. Error is shown as dashed lines around the figure and often smaller than the thickness of the line. Total variance cannot be estimated by adding the integral under each curve as the number of samples from each stage is not taken into account.

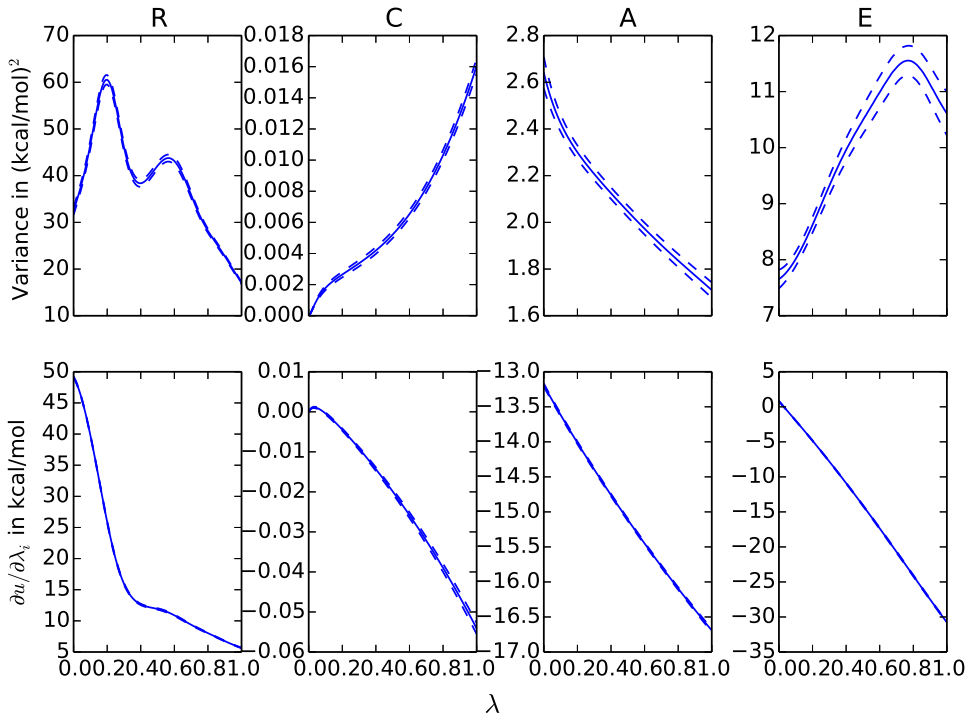


Figure B.3: Variance and $\partial u/\partial\lambda$ for the R/C/A/E-WCA basis function schedule. Error is shown as dashed lines around the figure and often smaller than the thickness of the line. Total variance cannot be estimated by adding the integral under each curve as the number of samples from each stage is not taken into account.

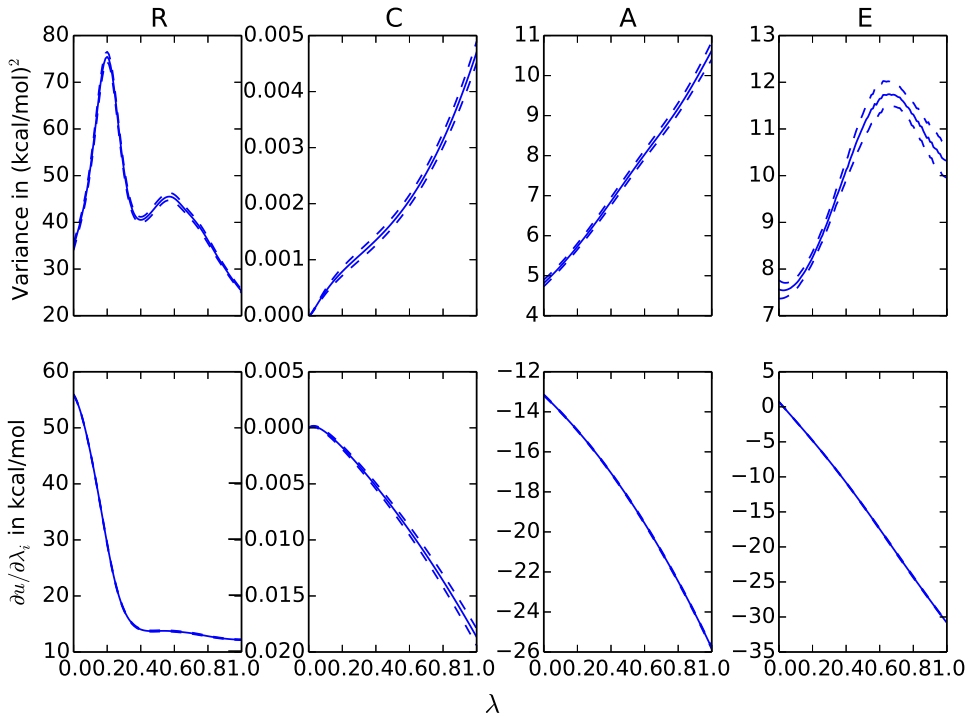


Figure B.4: Variance and $\partial u/\partial\lambda$ for the R/C/A/E-12-6 basis function schedule. Error is shown as dashed lines around the figure and often smaller than the thickness of the line. Total variance cannot be estimated by adding the integral under each curve as the number of samples from each stage is not taken into account.

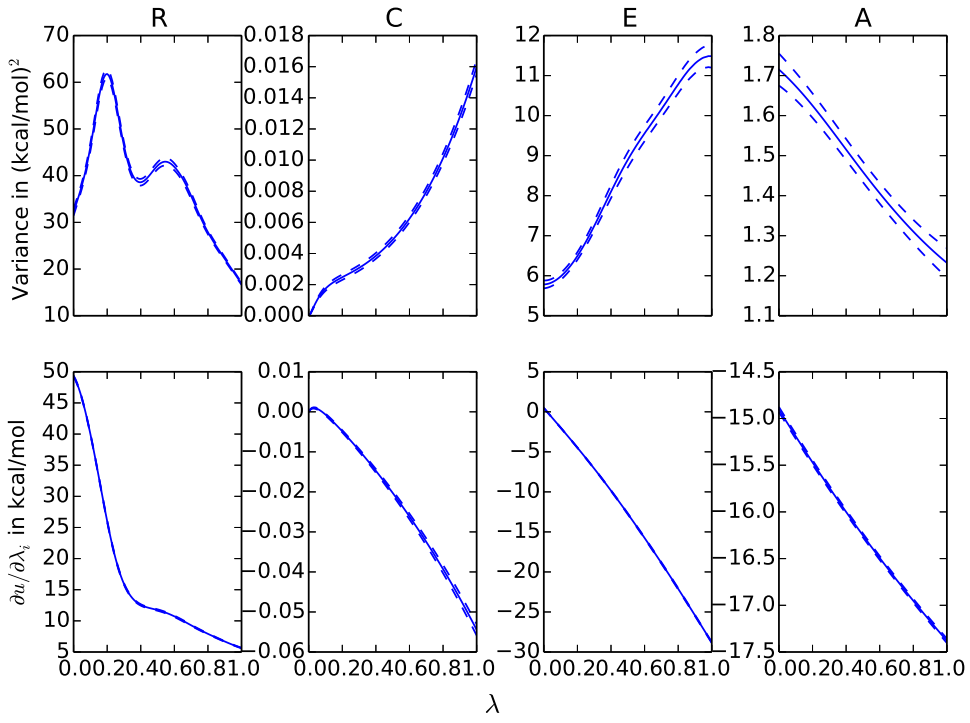


Figure B.5: Variance and $\partial u/\partial\lambda$ for the R/C/E/A-WCA basis function schedule. Error is shown as dashed lines around the figure and often smaller than the thickness of the line. Total variance cannot be estimated by adding the integral under each curve as the number of samples from each stage is not taken into account.

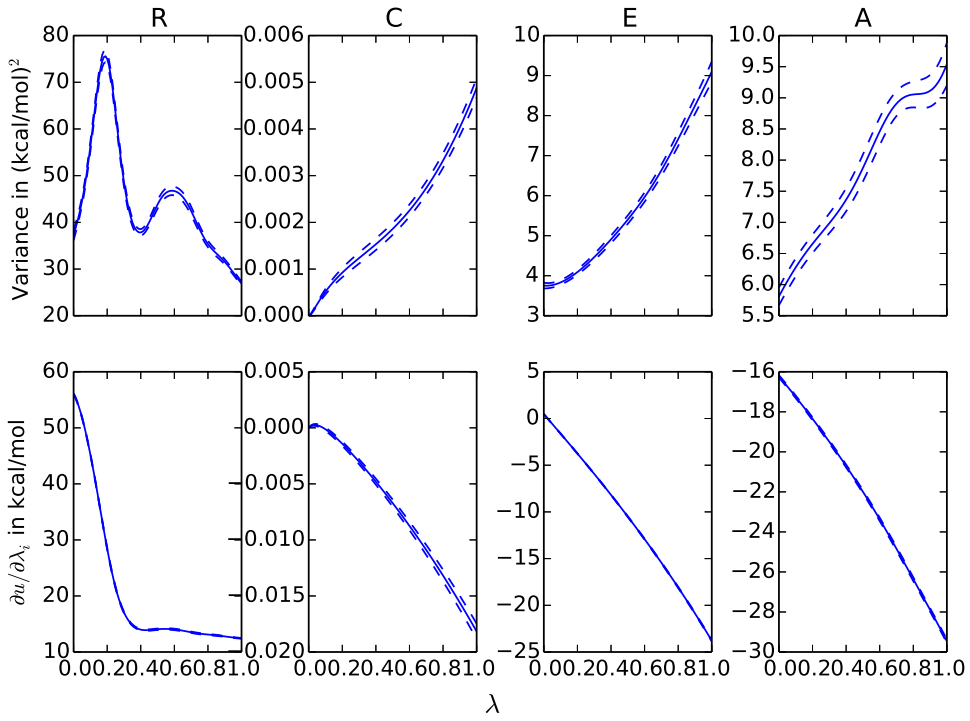


Figure B.6: Variance and $\partial u/\partial\lambda$ for the R/C/E/A-12-6 basis function schedule. Error is shown as dashed lines around the figure and often smaller than the thickness of the line. Total variance cannot be estimated by adding the integral under each curve as the number of samples from each stage is not taken into account.

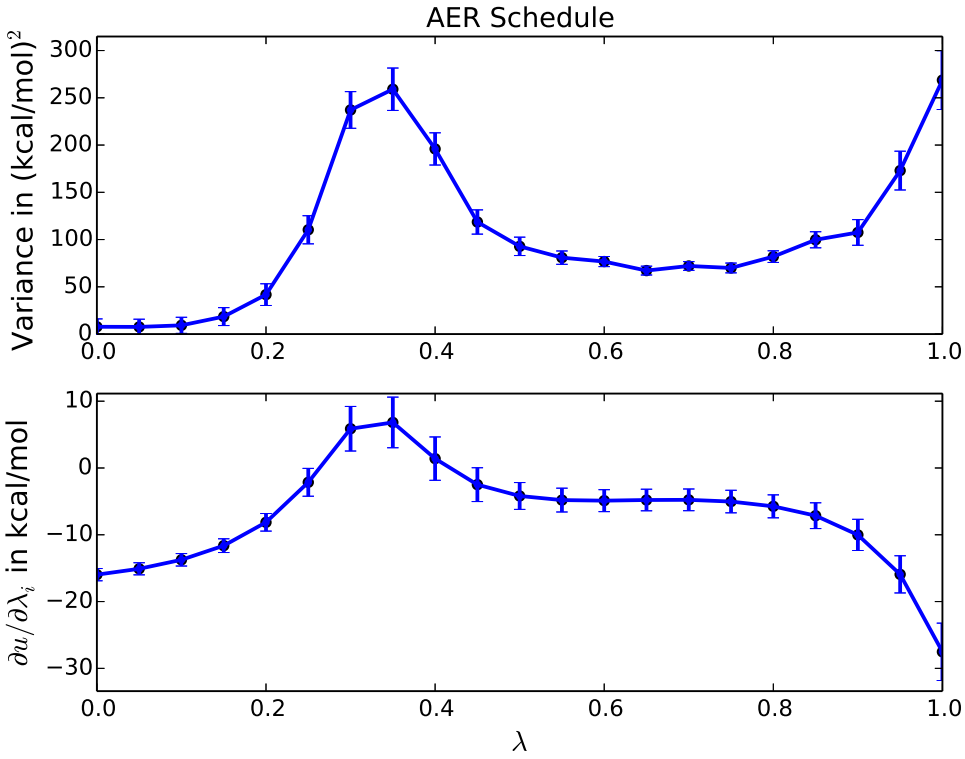


Figure B.7: Variance and $\partial u/\partial\lambda$ for the AER-SC (soft core) alchemical method. Sampling was done at discrete states shown as points, and the line connecting points serves to guide the eye only. Error shown as vertical error bars in both figures and computed by 200 bootstrap samples. Error bars of variance are doubled and error of $\partial u/\partial\lambda$ is increased by factor of 10 to be visible in the figure.

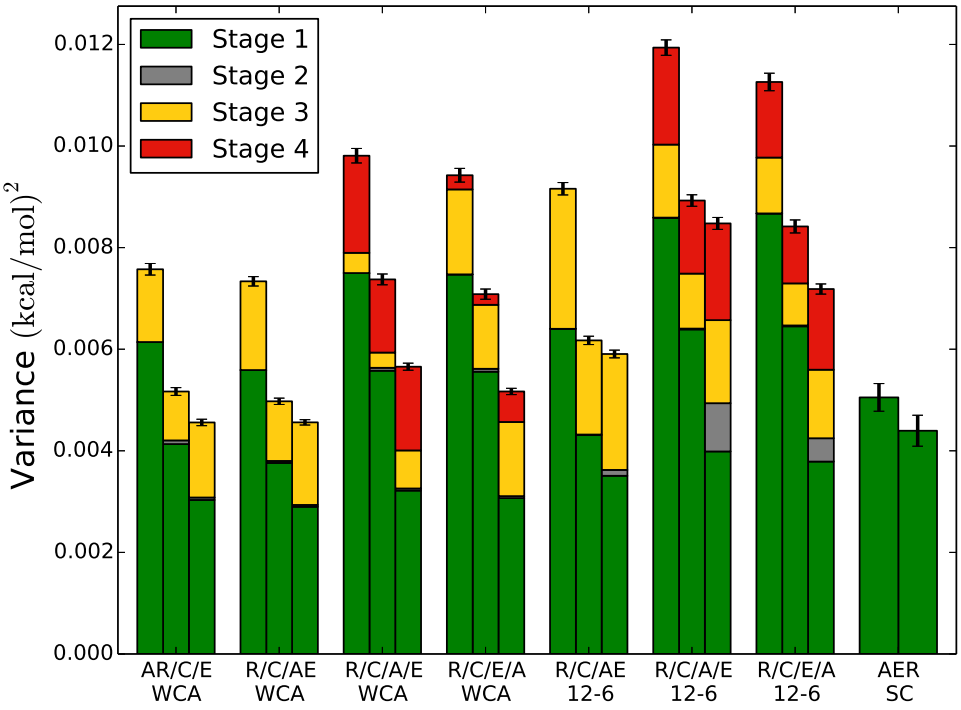


Figure B.8: Decomposition of each stages contribution to the variance of the calculation of free energy for all sampling schemes. Shown is the variance of the free energy for 21000 samples distributed to all seven basis function paths and the soft core electrostatic path. Samples were distributed either uniformly (left side bars), uniformly with capped stages using minimal samples (center bars) or proportional to the sample standard deviation (right side bars) and total error in the variance of the free energy is shown as an error bar at the top of a stack. Individual stages are distinguished by color and stacked in coupling order with green, gray, yellow, and red. E.g. In R/C/AE, Stage 1 is "R" (green), Stage 2 is "C" (gray), and Stage 3 is "AE" (yellow). The capping stage, "C" often contributes very little to the total variance and does not appear in the bar chart at this scale. "WCA" and "12-6" labels distinguish the Lennard-Jones basis functions and "SC" soft core method applied to all forces 1-1-6 parameterization.

Table B.1: Sample variance and free energy of solvation for 3-methylindole between seven different basis function pathways and the soft core electrostatics path. The variances shown here are direct integration of the curves in Fig. 3.4 and Fig. B.1 through Fig. B.6. They are non additive without accounting for the number of samples from each state. Free energies are within two standard deviations. Direct comparison between statistical uncertainties of each path is not a reliable measure of relative computational efficiency as each path has different correlation times and an unequal numbers of samples are drawn between each path. Variance of the calculation of free energy is shown in Fig. 3.5 and Table B.2. Variance is in units of $(\text{kcal/mol})^2$. Free energy is in units of kcal/mol. Stage number in a column header corresponds to stage in the schedule, e.g. in R/C/AE, Stage 1 is R, Stage 2 is C, and Stage 3 is AE. "WCA" and "12-6" labels distinguish the Lennard-Jones basis functions and "SC" single-step soft core method with 1-1-6 parameterization.

Schedule	**Stage 1**	**Stage 2**	**Stage 3**	**Stage 4**	**Free Energy**
AR/C/E-WCA	42.97 ± 0.793	$1.20{\cdot}10^{-2}$ ± $4.04{\cdot}10^{-4}$	10.01 ± 0.230	–	-5.678 ± 0.077
R/C/AE-WCA	39.12 ± 0.614	$6.32{\cdot}10^{-3}$ ± $1.73{\cdot}10^{-4}$	12.21 ± 0.203	–	-5.538 ± 0.061
R/C/A/E-WCA	39.18 ± 0.707	$6.41{\cdot}10^{-3}$ ± $2.48{\cdot}10^{-4}$	2.06 ± 0.030	10.04 ± 0.230	-5.810 ± 0.074
R/C/E/A-WCA	38.85 ± 0.681	$6.10{\cdot}10^{-3}$ ± $2.33{\cdot}10^{-4}$	8.71 ± 0.170	1.47 ± 0.030	-5.854 ± 0.074
R/C/AE-12-6	44.81 ± 0.740	$2.10{\cdot}10^{-3}$ ± $1.08{\cdot}10^{-4}$	19.27 ± 0.423	–	-5.696 ± 0.074
R/C/A/E-12-6	44.67 ± 0.750	$1.92{\cdot}10^{-3}$ ± $0.97{\cdot}10^{-4}$	7.49 ± 0.111	10.05 ± 0.228	-5.719 ± 0.074
R/C/E/A-12-6	45.10 ± 0.867	$2.08{\cdot}10^{-3}$ ± $1.23{\cdot}10^{-4}$	5.73 ± 0.117	7.83 ± 0.197	-5.770 ± 0.090
AER-SC	102.57 ± 5.491	–	–	–	-5.759 ± 0.106

Table B.2: Variance in the calculation for free energy for coupling 3-methylindole between seven different basis function pathways and the soft core electrostatics path. The variances here are shown after distributing 21000 samples to evenly spaced states in each path through two separate means: "uniformly" and "proportional" to the sample standard deviation. These variances are additive and provide the lower bound estimate of the variance of the free energy of solvation. These numbers are visualized in Fig. 3.5 and Fig. B.8. A special case, which is a hybrid of the first two, is considered to the basis function paths where uniform sampling was done, except in the capping stage which was only sampled at the end points. This is denoted by the "Hybrid" label in the "Sampling" column. Variance is in units of $(\text{kcal/mol})^2$. Stage number in a column header corresponds to stage in the schedule. E.g. In R/C/AE, Stage 1 is R, Stage 2 is C, and Stage 3 is AE. "WCA" and "12-6" labels distinguish the Lennard-Jones basis functions and "SC" soft core method with 1-1-6 parameterization. Table appears on next page.

Table B.2: Caption appears on previous page

Schedule	Sampling	Stage 1	Stage 2	Stage 3	Stage 4	Total
AR/C/E-WCA	Uniformly	$6.23 \cdot 10^{-3} \pm 1.11 \cdot 10^{-4}$	$1.75 \cdot 10^{-6} \pm 5.89 \cdot 10^{-8}$	$1.45 \cdot 10^{-3} \pm 3.34 \cdot 10^{-5}$	–	$7.68 \cdot 10^{-3} \pm 1.15 \cdot 10^{-4}$
	Proportional	$3.03 \cdot 10^{-3} \pm 5.35 \cdot 10^{-5}$	$4.84 \cdot 10^{-5} \pm 1.79 \cdot 10^{-6}$	$1.48 \cdot 10^{-3} \pm 3.35 \cdot 10^{-5}$	–	$4.56 \cdot 10^{-3} \pm 6.32 \cdot 10^{-5}$
	Hybrid	$4.17 \cdot 10^{-3} \pm 7.40 \cdot 10^{-5}$	$7.30 \cdot 10^{-5} \pm 1.86 \cdot 10^{-6}$	$9.72 \cdot 10^{-4} \pm 2.23 \cdot 10^{-5}$	–	$5.22 \cdot 10^{-3} \pm 7.74 \cdot 10^{-5}$
R/C/AE-WCA	Uniformly	$5.67 \cdot 10^{-3} \pm 8.91 \cdot 10^{-5}$	$9.24 \cdot 10^{-7} \pm 2.53 \cdot 10^{-8}$	$1.77 \cdot 10^{-3} \pm 2.95 \cdot 10^{-5}$	–	$7.44 \cdot 10^{-3} \pm 9.38 \cdot 10^{-5}$
	Proportional	$2.90 \cdot 10^{-3} \pm 4.54 \cdot 10^{-5}$	$3.70 \cdot 10^{-5} \pm 1.15 \cdot 10^{-6}$	$1.63 \cdot 10^{-3} \pm 2.67 \cdot 10^{-5}$	–	$4.56 \cdot 10^{-3} \pm 5.27 \cdot 10^{-5}$
	Hybrid	$3.80 \cdot 10^{-3} \pm 5.97 \cdot 10^{-5}$	$3.92 \cdot 10^{-5} \pm 8.95 \cdot 10^{-7}$	$1.19 \cdot 10^{-3} \pm 1.97 \cdot 10^{-5}$	–	$5.02 \cdot 10^{-3} \pm 6.28 \cdot 10^{-5}$
R/C/A/E-WCA	Uniformly	$7.65 \cdot 10^{-3} \pm 1.39 \cdot 10^{-4}$	$1.23 \cdot 10^{-6} \pm 4.82 \cdot 10^{-8}$	$4.06 \cdot 10^{-4} \pm 5.82 \cdot 10^{-6}$	$1.95 \cdot 10^{-3} \pm 4.44 \cdot 10^{-5}$	$1.00 \cdot 10^{-2} \pm 1.46 \cdot 10^{-4}$
	Proportional	$3.22 \cdot 10^{-3} \pm 5.80 \cdot 10^{-5}$	$4.16 \cdot 10^{-5} \pm 1.85 \cdot 10^{-6}$	$7.49 \cdot 10^{-4} \pm 1.07 \cdot 10^{-5}$	$1.65 \cdot 10^{-3} \pm 3.72 \cdot 10^{-5}$	$5.66 \cdot 10^{-3} \pm 6.98 \cdot 10^{-5}$
	Hybrid	$5.66 \cdot 10^{-3} \pm 1.03 \cdot 10^{-4}$	$5.82 \cdot 10^{-5} \pm 1.73 \cdot 10^{-6}$	$3.04 \cdot 10^{-4} \pm 4.36 \cdot 10^{-6}$	$1.46 \cdot 10^{-3} \pm 3.34 \cdot 10^{-5}$	$7.48 \cdot 10^{-3} \pm 1.08 \cdot 10^{-4}$
R/C/E/A-WCA	Uniformly	$7.62 \cdot 10^{-3} \pm 1.34 \cdot 10^{-4}$	$1.19 \cdot 10^{-6} \pm 4.53 \cdot 10^{-8}$	$1.71 \cdot 10^{-3} \pm 3.33 \cdot 10^{-5}$	$2.84 \cdot 10^{-4} \pm 5.76 \cdot 10^{-6}$	$9.61 \cdot 10^{-3} \pm 1.38 \cdot 10^{-4}$
	Proportional	$3.07 \cdot 10^{-3} \pm 5.38 \cdot 10^{-5}$	$3.93 \cdot 10^{-5} \pm 1.71 \cdot 10^{-6}$	$1.46 \cdot 10^{-3} \pm 2.81 \cdot 10^{-5}$	$6.01 \cdot 10^{-4} \pm 1.22 \cdot 10^{-5}$	$5.17 \cdot 10^{-3} \pm 6.20 \cdot 10^{-5}$
	Hybrid	$5.64 \cdot 10^{-3} \pm 9.89 \cdot 10^{-5}$	$5.79 \cdot 10^{-5} \pm 1.74 \cdot 10^{-6}$	$1.28 \cdot 10^{-3} \pm 2.49 \cdot 10^{-5}$	$2.13 \cdot 10^{-4} \pm 4.31 \cdot 10^{-6}$	$7.19 \cdot 10^{-3} \pm 1.02 \cdot 10^{-4}$
R/C/AE-12-6	Uniformly	$6.49 \cdot 10^{-3} \pm 1.07 \cdot 10^{-4}$	$3.07 \cdot 10^{-7} \pm 1.58 \cdot 10^{-8}$	$2.80 \cdot 10^{-3} \pm 6.14 \cdot 10^{-5}$	–	$9.29 \cdot 10^{-3} \pm 1.24 \cdot 10^{-4}$
	Proportional	$3.51 \cdot 10^{-3} \pm 5.78 \cdot 10^{-5}$	$1.16 \cdot 10^{-4} \pm 1.00 \cdot 10^{-5}$	$2.28 \cdot 10^{-3} \pm 4.84 \cdot 10^{-5}$	–	$5.91 \cdot 10^{-3} \pm 7.60 \cdot 10^{-5}$
	Hybrid	$4.35 \cdot 10^{-3} \pm 7.19 \cdot 10^{-5}$	$1.18 \cdot 10^{-5} \pm 4.53 \cdot 10^{-7}$	$1.87 \cdot 10^{-3} \pm 4.11 \cdot 10^{-5}$	–	$6.23 \cdot 10^{-3} \pm 8.28 \cdot 10^{-5}$
R/C/A/E-12-6	Uniformly	$8.76 \cdot 10^{-3} \pm 1.47 \cdot 10^{-4}$	$3.73 \cdot 10^{-7} \pm 1.88 \cdot 10^{-8}$	$1.47 \cdot 10^{-3} \pm 2.18 \cdot 10^{-5}$	$1.95 \cdot 10^{-3} \pm 4.41 \cdot 10^{-5}$	$1.22 \cdot 10^{-2} \pm 1.55 \cdot 10^{-4}$
	Proportional	$3.99 \cdot 10^{-3} \pm 6.68 \cdot 10^{-5}$	$9.50 \cdot 10^{-4} \pm 8.32 \cdot 10^{-5}$	$1.64 \cdot 10^{-3} \pm 2.42 \cdot 10^{-5}$	$1.90 \cdot 10^{-3} \pm 4.28 \cdot 10^{-5}$	$8.48 \cdot 10^{-3} \pm 1.17 \cdot 10^{-4}$
	Hybrid	$6.48 \cdot 10^{-3} \pm 1.09 \cdot 10^{-4}$	$1.70 \cdot 10^{-5} \pm 6.95 \cdot 10^{-7}$	$1.10 \cdot 10^{-3} \pm 1.63 \cdot 10^{-5}$	$1.46 \cdot 10^{-3} \pm 3.31 \cdot 10^{-5}$	$9.06 \cdot 10^{-3} \pm 1.15 \cdot 10^{-4}$
R/C/E/A-12-6	Uniformly	$8.84 \cdot 10^{-3} \pm 1.70 \cdot 10^{-4}$	$4.05 \cdot 10^{-7} \pm 2.39 \cdot 10^{-8}$	$1.12 \cdot 10^{-3} \pm 2.32 \cdot 10^{-5}$	$1.52 \cdot 10^{-3} \pm 3.81 \cdot 10^{-5}$	$1.15 \cdot 10^{-2} \pm 1.76 \cdot 10^{-4}$
	Proportional	$3.79 \cdot 10^{-3} \pm 7.30 \cdot 10^{-5}$	$4.59 \cdot 10^{-4} \pm 4.85 \cdot 10^{-5}$	$1.35 \cdot 10^{-3} \pm 2.73 \cdot 10^{-5}$	$1.59 \cdot 10^{-3} \pm 3.97 \cdot 10^{-5}$	$7.18 \cdot 10^{-3} \pm 1.00 \cdot 10^{-4}$
	Hybrid	$6.54 \cdot 10^{-3} \pm 1.26 \cdot 10^{-4}$	$1.77 \cdot 10^{-5} \pm 7.52 \cdot 10^{-7}$	$8.40 \cdot 10^{-4} \pm 1.74 \cdot 10^{-5}$	$1.14 \cdot 10^{-3} \pm 2.85 \cdot 10^{-5}$	$8.54 \cdot 10^{-3} \pm 1.30 \cdot 10^{-4}$
AER-SC	Uniformly	$5.06 \cdot 10^{-3} \pm 2.73 \cdot 10^{-4}$	–	–	–	$5.06 \cdot 10^{-3} \pm 2.73 \cdot 10^{-4}$
	Proportional	$4.39 \cdot 10^{-3} \pm 3.05 \cdot 10^{-4}$	–	–	–	$4.39 \cdot 10^{-3} \pm 3.05 \cdot 10^{-4}$

B.3 Implementing Long-Range Electrostatic Interactions with PME

Electrostatic interactions are frequently decomposed into short-range and long-range interactions such as with the particle mesh Ewald method (PME). [139] Long-range forces are computed in an entirely separate procedure than short-range forces to improve simulation efficiency. There are three components to the PME energy: direct, reciprocal, and self-energy summations; written as [39, 81]

$$u_E(r) = u_{\text{dir}} + u_{\text{rec}} + u_{\text{self}} \tag{B.7}$$

$$u_{\text{dir}} = \frac{1}{2}\sum_{i,j} q_i q_j \frac{\text{erfc}(\alpha r)}{r} \tag{B.8}$$

$$u_{\text{rec}} = \frac{1}{2\pi V}\sum_{i,j} q_i q_j \sum_{\mathbf{k}=0} \frac{\exp(-(\pi\mathbf{k}/\alpha)^2 + 2\pi \text{i}\mathbf{k}\cdot(\mathbf{r}_i - \mathbf{r}_j))}{\mathbf{m}^2} \tag{B.9}$$

$$u_{\text{self}} = -\frac{\alpha}{\sqrt{\pi}}\sum_i q_i^2 \tag{B.10}$$

where α is a function of the error tolerance, $\mathbf{k}$ is a vector which is a function of the lattice vectors in Fourier space, $\mathbf{r}$ is the radial position vector of a particle, and $\mathbf{m}$ is the magnitude of $\mathbf{k}$. The electrostatics are decomposed this way because the long range term converges quickly in Fourier space where as the short range term converges quickly in real space. This is a very efficient algorithm for systems with periodic boundary conditions (PBC).

It is often simpler to modify the charge of an atom than it is to modify the simulation code directly in many simulation packages which support alchemical transformations. [35, 38, 39, 119] For our implementation, the alchemical switch is acting directly on q_i instead of the product of $q_i q_j$. Because of this, there are two types of alchemical interactions which evolve: solute-solvent interactions which scale as $h_i(\lambda)$, and solute-solute interactions which scale as $h_i(\lambda)^2$. The solute-solute interactions

come from non-excluded short-range interactions and long-range interactions with copies of the solute in other periodic boxes. If we do not wish to scale the long-range interactions separately from the short-range interactions, we only need to simulate one extra state to isolate the basis functions scaling as $h_i(\lambda)^2$. If the intramolecular charges are allowed to interact, as was done in our implementation, then the $h_i(\lambda)^2$ terms are only with periodic copies of the atom and contribute negligible amounts to the variance. Because we require that the atomic core be fully repulsive before coupling the electrostatics, there will not be a scenario where a charged, alchemical atom without a repulsive core overlaps with another atom, ensuring the variance in the electrostatics is controlled through linear interactions. The potential energy of the interactions scaling as $h_i(\lambda)^2$ are roughly half of those scaling as $h_i(\lambda)$ in our simulations, and contribute a non-negligible amount to the total potential energy. However, the variance in the potential for the interactions scaling as $h_i(\lambda)^2$ are three to four orders magnitude less than the $h_i(\lambda)$ interactions, so they have negligible affect on the variance calculations. Therefore, the $h_i(\lambda)^2$ must be included in order to accurately compute the energy at unsampled states, but do not significantly change the variance calculations.

It is possible to have separate alchemical switches for long- and short-range interactions. In this instance, we would need to generate enough states to isolate the short-range terms as $h_i(\lambda)$ and $h_i(\lambda)^2$, as well as the long-range terms also scaling as $h_i(\lambda)$ and $h_i(\lambda)^2$. Some simulation packages may allow coding separate alchemical switches for real space and Fourier space, but only OpenMM will be examined here as the alchemical switches were not implemented in other packages.

B.4 OpenMM Implementation of PME-decomposed Alchemical Switches

OpenMM's CustomNonbondedForce class [39, 119] does not allow access to the PME evaluations, but still allows manipulation of the PME switch. The ability to write any custom expression into OpenMM's custom nonbonded engine lets us define a set of equations which calculate the PME components. We separate out the PME interactions by constructing a two-basis CustomNonbondedForce: one basis subtracting OpenMM's direct space calculation, and another to add in in our custom direct space evaluation with a separate switch. Lastly, we set the charge of the atom to modify the Fourier space interactions. This CustomNonbondedForce between atom i and j is written as

$$u_E(r,\lambda) = u_{\text{alch,dir}} - A_{ij}u_{\text{dir}} \tag{B.11}$$

$$u_{\text{dir}} = A_i(h_{\text{PME}})q_iA_j(h_{\text{PME}})q_j\frac{\text{erfc}(\alpha r)}{4\pi\epsilon_0 r} \tag{B.12}$$

$$u_{\text{alch,dir}} = A_i(h_E)q_iA_j(h_E)q_j\frac{\text{erfc}(\alpha r)}{4\pi\epsilon_0 r} \tag{B.13}$$

where

$$A_{ij} = \begin{cases} 1 & \text{if atom } i \text{ or } j \text{ is alchemically modified} \\ 0 & \text{otherwise.} \end{cases} \tag{B.14}$$

and

$$A_i(X) = \begin{cases} X & \text{if atom } i \text{ is alchemically modified} \\ 1 & \text{otherwise.} \end{cases} \tag{B.15}$$

These equations allow OpenMM's normal electrostatic evaluation, but cancel out the direct space evaluation in a controllable method. This allows us to isolate the long-range evaluations from the short-range evaluations and design a minimal variance switch for each decomposed term. When a capped potential is applied, we only cap $u_{\text{alch,dir}}$ since OpenMM's evaluation of u_{dir} is still removed from the system to produce

accurate results.

C.1 Considerations for solvents with multiple unique particles

One complication to the linear basis representation approach is when the solvent is composed of multiple particle types. Situations where this happens includes both solvents with multiple types of molecules, and solvents where multiple atom types make up each solvent molecule, such as water. A different set of scalar alchemical switches, $h_n(\lambda_n)$, for each unique atom type in the solvent is required to accurately compute the energies for arbitrary state Z. This is because we are only changing σ_{ii} of the solute explicitly, but C_{12} and C_6 will scale as non-linear functions of σ_{ij}, which will be different for the interactions with each solvent atom type. One solution is to compute the basis functions for each solvent atom type interacting with the solute. However, this can be avoided with geometric mixing rules for both Lennard-Jones parameters,

$$\epsilon_{ij} = (\epsilon_{ii}\epsilon_{jj})^{1/2}$$
$$\sigma_{ij} = (\sigma_{ii}\sigma_{jj})^{1/2}$$

as opposed to arithmetic mixing rules for σ_{ij}, where $\sigma_{ij} = 0.5(\sigma_{ii}+\sigma_{jj})$. It is important to note that Eq. (4.6) is still valid for arithmetic mixing rules, but requires a separate C_{12} and C_6 term for each atom type in the solvent. Alternatively, the arithmetic mixing rules could undergo binomial expansion as

$$\left[\frac{1}{2}(\sigma_{ii}+\sigma_{jj})\right]^n = \frac{1}{2^n}\sum_{\ell=0}^{n}\binom{n}{\ell}\sigma_{ii}^{n-\ell}\sigma_{ij}^{\ell} \tag{C.1}$$

and new $h_n(\lambda_n)$ functions written which scale each $\sigma_{ii}^{n-\ell}$ term. The potential of any atom can be computed with geometric mixing rules and two reference points. This section will denote each C_n explicitly in terms of the reference particles, X and Y,

and the arbitrary solvent site S, where S can represent any of the solvent particles as they will cancel out of the equations.

We can represent Coulombic and geometric Lennard-Jones mixing rules of the C_n terms by a general form of

$$C_{n,ij} = (C_{n,ii}C_{n,jj})^m \tag{C.2}$$

where m takes discrete values of $1/2$ or or 1 depending on the index of ℓ. For our basis functions, the C_{12} and C_6 terms have $m = 1/2$, whereas the electrostatic basis is $m = 1$ since its mixing term is $q_i q_j$. Representing the mixing rules in this way is beneficial as the following derivation does not depend on the exact value of m. Eq. (4.6) can be expanded as

$$\begin{aligned} u(r,\lambda) &= u_{\text{unaffected}} + \sum_\ell \left[\frac{h_n(\lambda_n)\left(C_{n,YS} - C_{n,XS}\right) + C_{n,XS}}{r^n}\right]_\ell \\ u(r,\lambda) &= u_{\text{unaffected}} \\ &\quad + \sum_\ell \left[\frac{h_n(\lambda_n)\left[\left(C_{n,YY}C_{n,SS}\right)^m - \left(C_{n,XX}C_{n,SS}\right)^m\right] + \left(C_{n,XX}C_{n,SS}\right)^m}{r^n}\right]_\ell . \end{aligned} \tag{C.3}$$

We know that for any arbitrary state Z that

$$\left(C_{n,ZZ}C_{n,SS}\right)^m = h_n(\lambda_n)\left[\left(C_{n,YY}C_{n,SS}\right)^m - \left(C_{n,XX}C_{n,SS}\right)^m\right] + \left(C_{n,XX}C_{n,SS}\right)^m . \tag{C.4}$$

We can determine what value $h_n(\lambda_n)$ should take given a $C_{n,ZZ}$ from Eq. (C.4) by

$$h_n = \frac{C_{n,ZZ}^m - C_{n,XX}^m}{C_{n,YY}^m - C_{n,XX}^m}. \tag{C.5}$$

Because we are using multiplicative mixing rules, Eq. (C.5) does not depend on any solvent parameter or site. The potential of any configuration evaluated at state Z can then be computed from the basis functions [54] and Eq. (4.6).

C.2 Relative free energies for uncharged, chemically realistic Lennard-Jones spheres

Solvation simulations for chemically realistic Lennard-Jones (LJ) spheres were carried out to validate the parameter search approach. The LJ spheres tested were united atom (UA) methane, [60, 72, 163] neopentane [167], and an approximation for a C_{60} molecule. [54] This simulations were carried out under the same conditions as in Section 4.3 with the addition of sampling along a fixed thermodynamic path. Solvation simulations were carried out per sphere along the a soft core coupling path [82, 83] with a 1-1-6 parameterization. [54, 84, 97] λ was along this path was sampled at 11 uniformly placed states from $\lambda = 0$ to $\lambda = 1$.

Table C.1: Relative free energies are statistically indistinguishable between solvation simulations and those computed from the basis function search of nonbonded parameter space. The free energy of solvation each particle was simulated (Direct Solvation) and the relative free energy to the reference state was computed (Relative Solvation). The Relative Solvation was compared to the relative free energy computed from the parameter search with 12 states of collected data (Parameter Search). The values for Relative Solvation are consistent with the statistical uncertainty for the Parameter Search free energy for the tested particles. Free energy is in units of kcal/mol.

Molecule	Direct Solvation	Relative Solvation	Parameter Search
Reference Particle	10.331 ± 0.128	0.000 ± 0.000	0.000 ± 0.000
UA Methane	2.215 ± 0.149	-8.116 ± 0.197	-7.696 ± 0.083
Neopentane	-0.264 ± 0.093	-10.595 ± 0.159	-10.758 ± 0.019
C_{60}	8.175 ± 0.135	-2.157 ± 0.186	-2.318 ± 0.004

C.3 Adaptive sampling algorithm for 3-D parameter search

Identifying multiple regions of phase space which are locally connected, but not globally connected is done by clustering grid points in the multidimensional space based on the relative uncertainty. The algorithm is available on GitHub [166] and details for the algorithm is as follows:

1. Compute the free energy (ΔF) and uncertainty ($\delta\Delta F$) at every grid point in the multidimensional space with MBAR. [101]

2. Choose the subset of grid points, S, where $\delta\Delta F$ is larger than a threshold uncertainty. We chose the grid points in S such that $\delta\Delta F \geq 0.5\,\text{kcal/mol} \in S$.

3. The DBSCAN [169] clustering algorithm is run on each grid point in S with the following neighbor and neighborhood criteria

   - Neighbor grid point is adjacent to current point, including diagonals
   - Relative error in the uncertainty of current point and the neighbor point is $< 10\%$. This must also be true for the relative error in uncertainty of the neighbor point with the current point.
   - A minimum of 5 points is required to defined a neighborhood, N.

4. Define the number of grid points in each N_i neighborhood as C_i

5. Select the "large" neighborhoods, L, where "large" is defined by $C_i / \sum_i C_i > 10\%$. This is done to minimize sampling small clusters which may be eliminated on subsequent iterations due to improved phase space overlap from sampling the large clusters.

   In the event no large clusters are identified, the three largest neighborhoods are selected instead.

In the event that **zero** clusters are identified, reduce the error threshold until 1/3 of the total points are above the threshold. This should allow new clusters to be found while also continuing to lower the uncertainty. Repeat the clustering step.

6. Identify the boundaries in each dimension for each L_j cluster. We choose to use SciPy's [130] multidimensional image analysis module, ndimage. A separate index, j, loops over L to account for the fact that $L \subseteq N$.

7. Select a point inside each L_j to perform additional sampling.

   One option is the center of "mass", where the "mass" is the uncertainty of each grid point in L_j.

   Alternately, a random point can be selected as was done for Chapter 4. Choosing only the center of mass was observed to improve local phase space overlap, but be slow at improving global phase space overlap since the center of mass is often far away from the boundary of L_j.

8. Let each point found in the previous step be a vertex, v_j in a graph. Let the reference state that relative thermodynamic properties are measured with respect to also be a vertex, R. Define the superset of all vertices, V, such that $V = \{v, R\}$.

9. Create a complete graph of V.

10. Find the minimum spanning tree (MST) of the complete graph of V. This was done with Kruskal's algorithm [170] here. Distance is defined by the Euclidean distance in multidimensional space.

    The MST creates a network of edges along where we will expand the local phase space overlap from each L, while also minimizing total number of edges required. So long as there is any path of phase space overlap connecting two

states; converged, low uncertainty estimates of thermodynamic properties can be made.

11. Compute the error along regularly spaced points on each edge of the MST with multidimensional interpolation from nearby grid points. The points along the edge do not have to reside along the regular spaced grid.

12. Run a boundary detection algorithm on each edge in the MST to identify the boundary of the local phase space overlap. We chose the the Sobel boundary detection algorithm. [171]

13. The new states to sample are then each point of the vertices, v, and each boundary found along the edges of the MST.

C.4 Ion Radial Distribution Functions

This section has the estimated radial distribution functions (RDFs, $g(r)$) for each ion from the Joung and Cheatham set in TIP3P water. [74] The RDFs are estimated at 160 evenly spaced bins from $r = 0$ to $r = 12$ nm from 203 sampled states of data. We have provided the Python code used to compute the RDFs on GitHub. [166] The following RDFs are the Ion-Water Oxygen pair distances estimated by MBAR [101] with error shown as dashed lines around the curve black lines. The green lines are the RDFs computed by directly simulating the ion and estimating the RDF from the trajectory. Error in the green lines is shown as dashed lines around the curve and estimated from 200 bootstrap samples of the simulated data. [123] The data from these direct simulations were not used in the MBAR estimate. The Li^+ and Na^+ σ_{ii} are are below the searched parameter space, leading to their RDFs appearing erratic and falling below $g(r) = 0$.

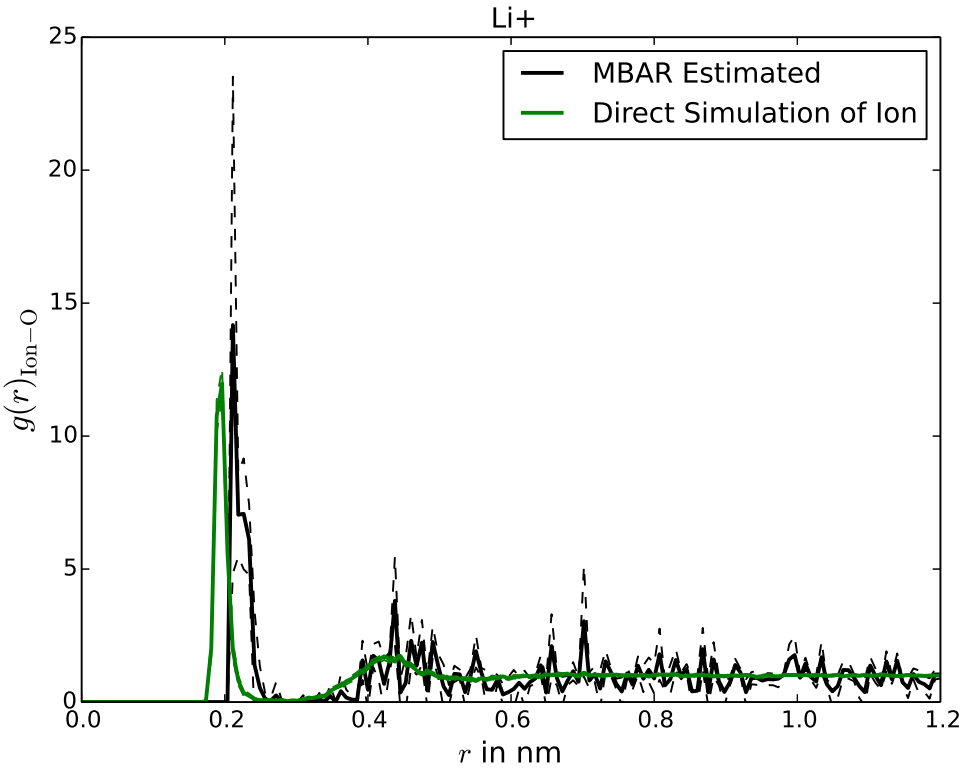


Figure C.1: RDF for Li^+, this is also Fig. 4.8b

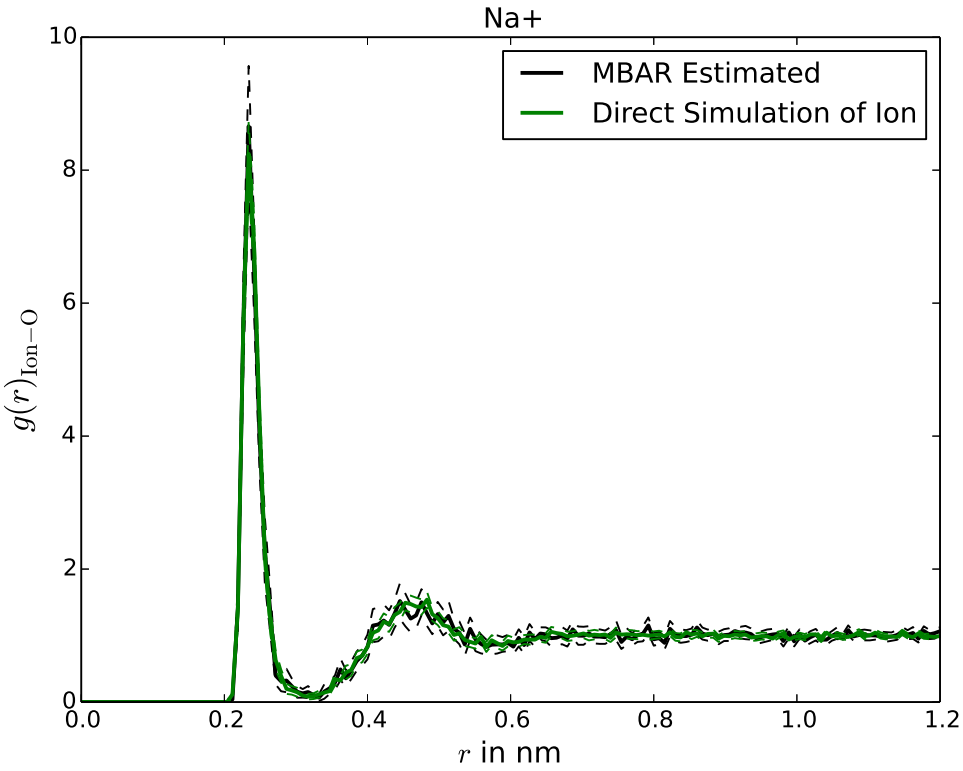


Figure C.2: RDF for Na^+

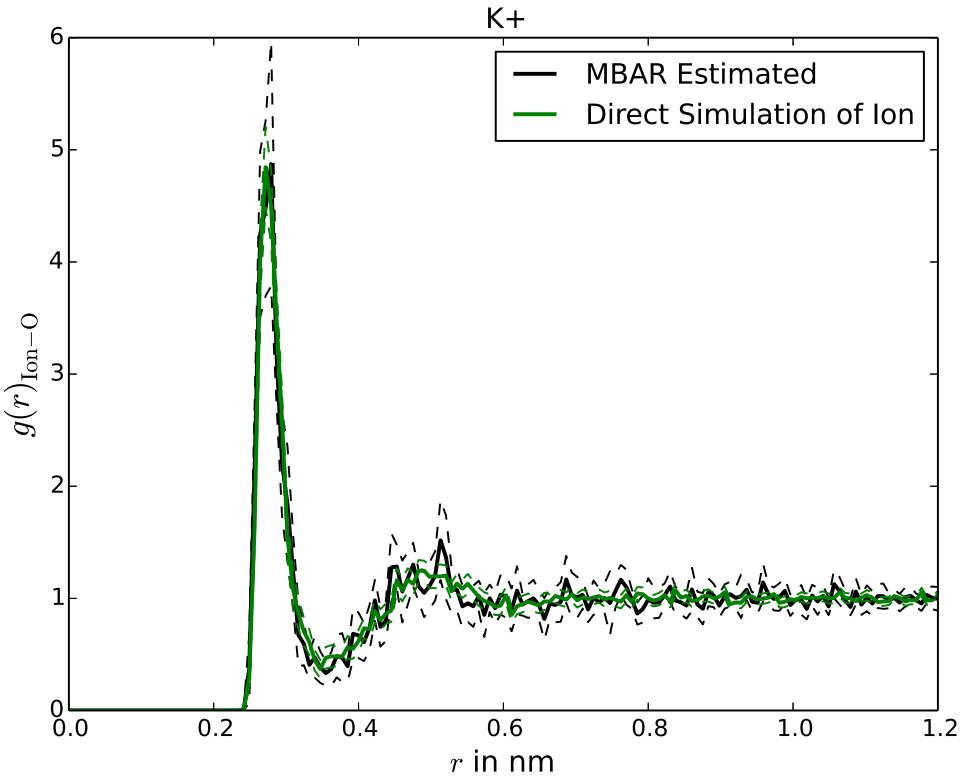


Figure C.3: RDF for K^+

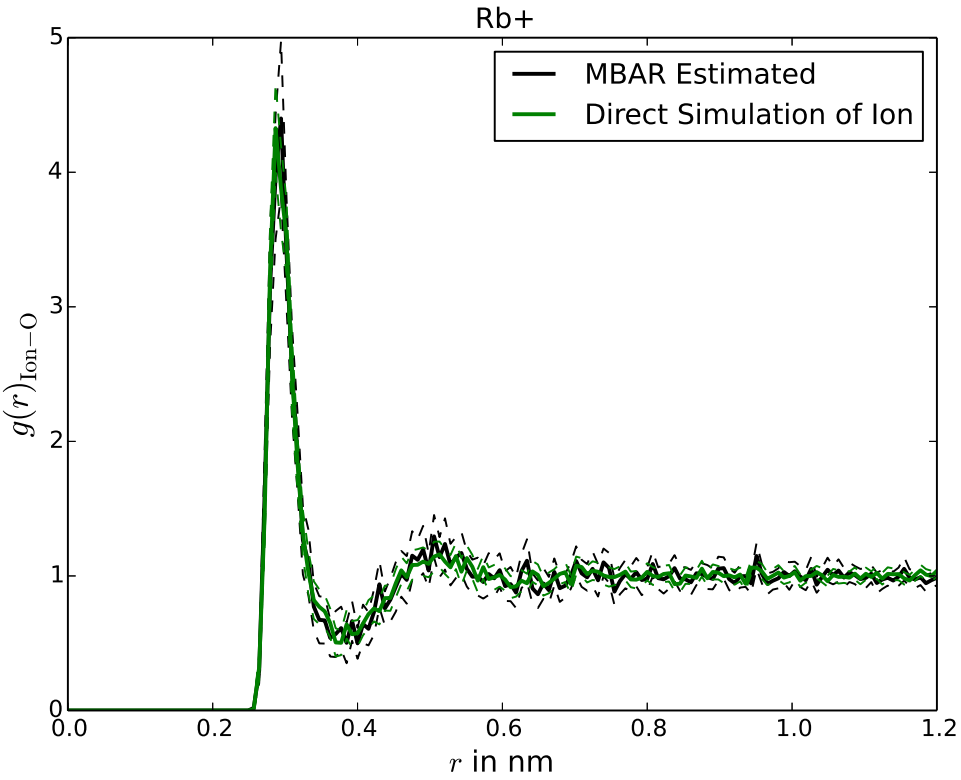


Figure C.4: RDF for Rb^+

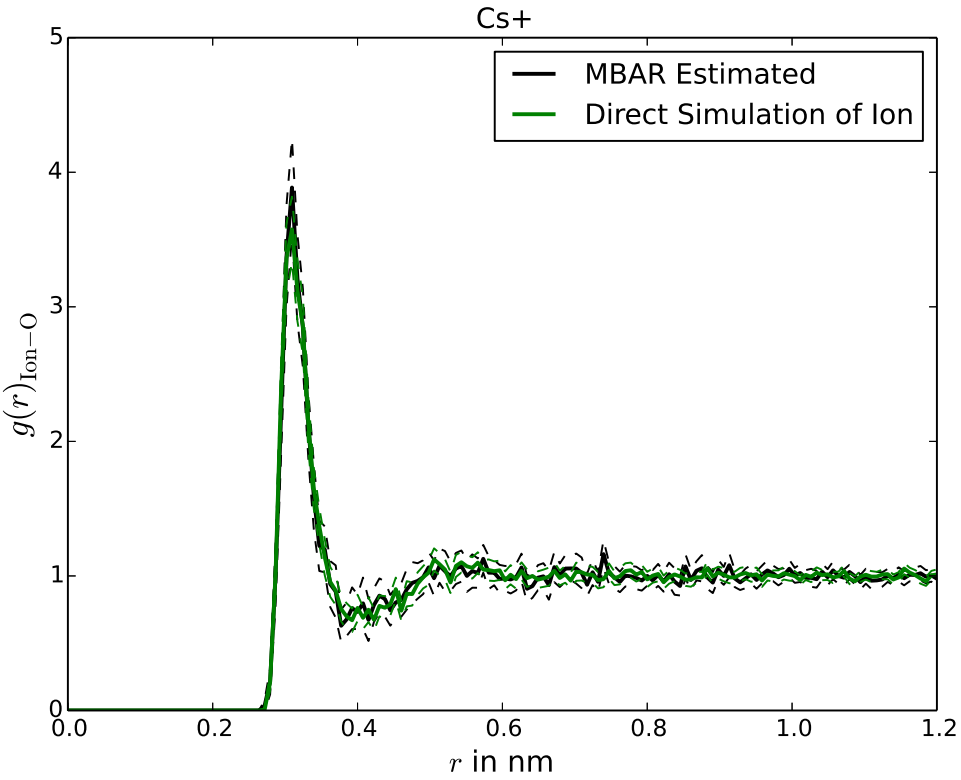


Figure C.5: RDF for Cs^+

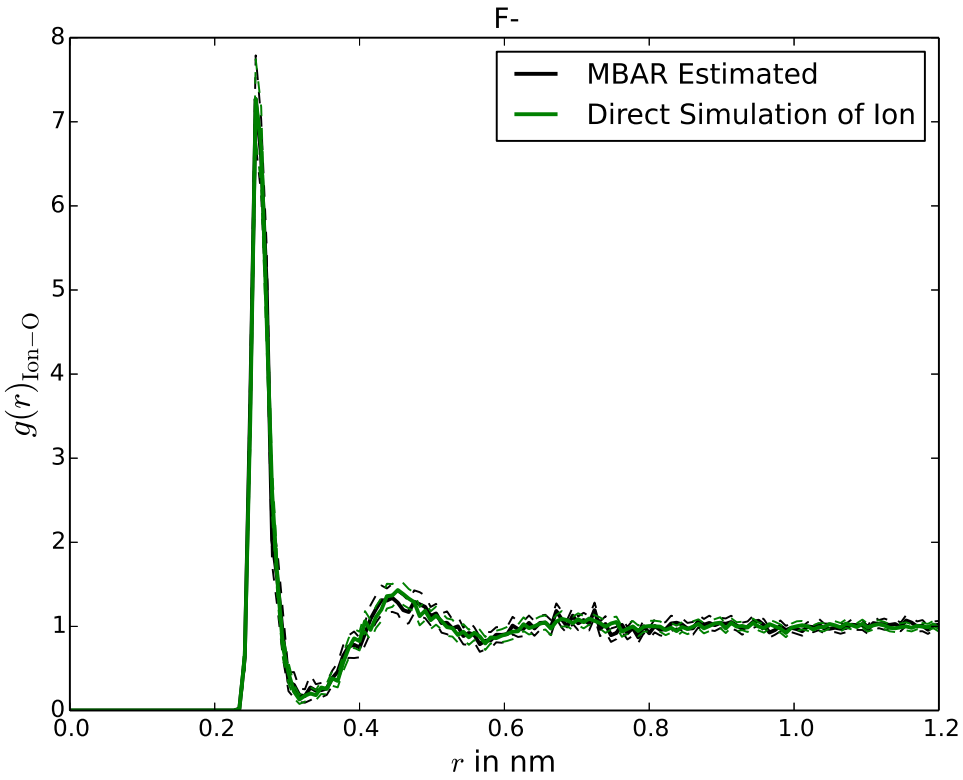


Figure C.6: RDF for F^-

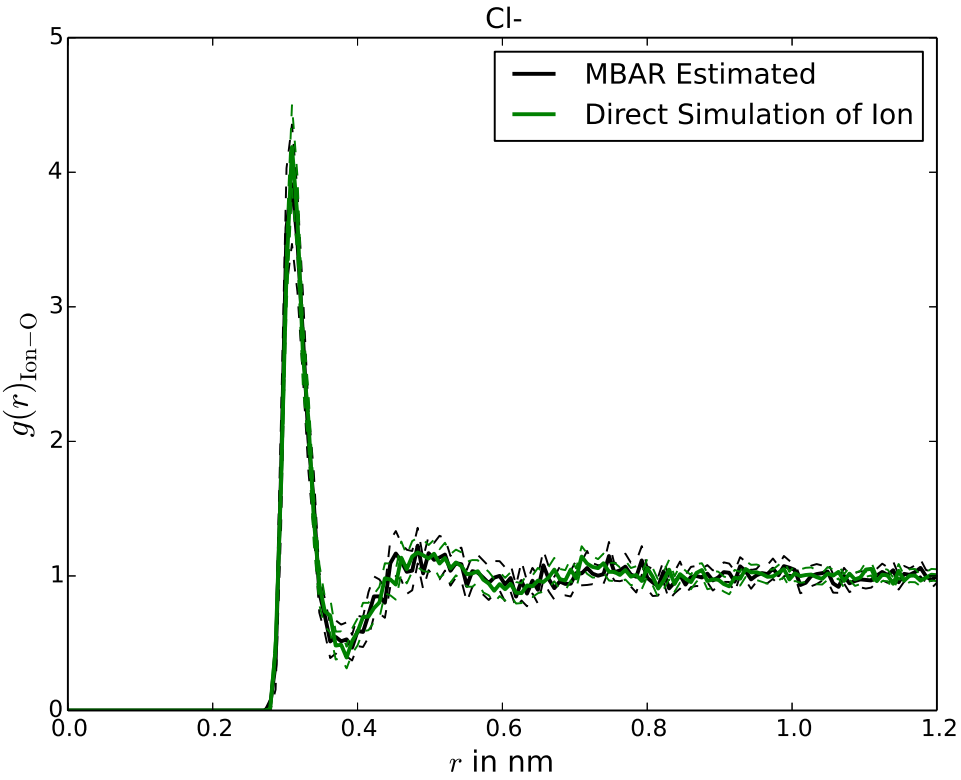


Figure C.7: RDF for Cl^-, this is also Fig. 8a

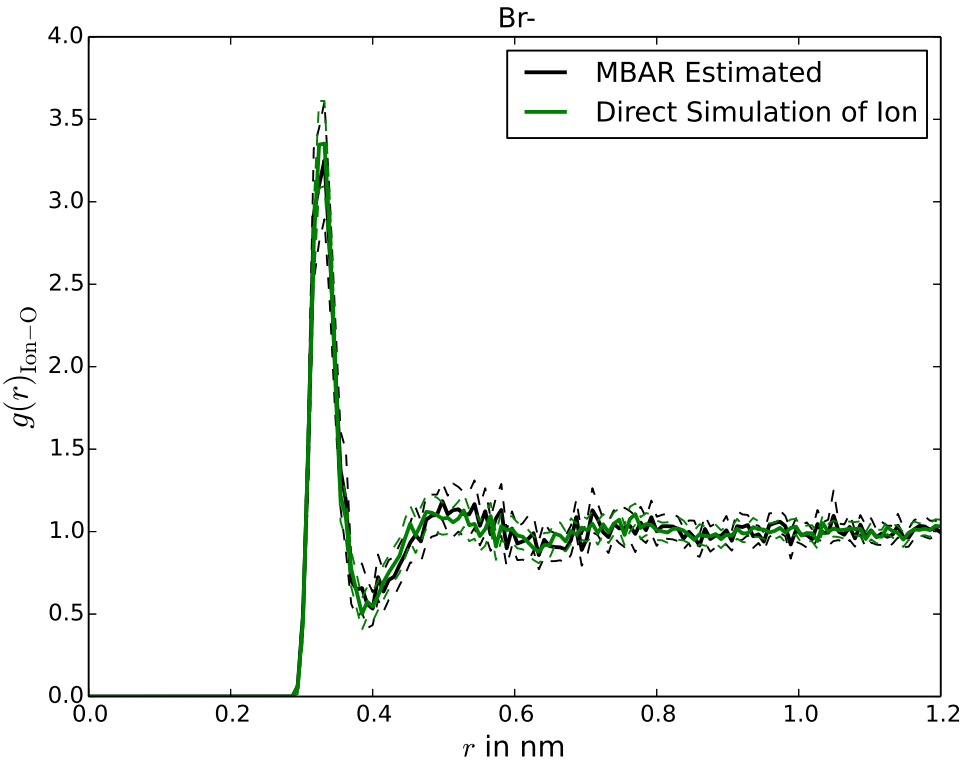


Figure C.8: RDF for Br^-

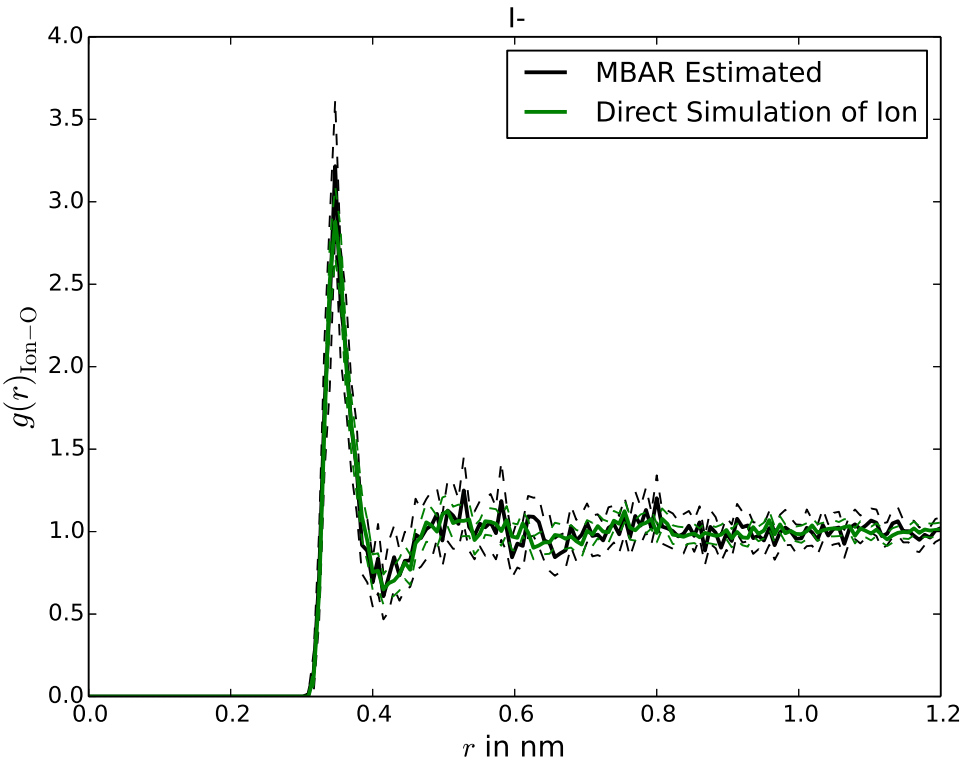


Figure C.9: RDF for I^-

C.5 Sampled nonbonded parameter combination, mean and maximum uncertainty, and eigenvalues per iteration

Table C.2: All sampled parameters combinations, algorithm iteration from Section C.3, and eigenvalues (λ). Also shown is the maximum and mean uncertainty in the free energy per iteration. Algorithm attempts to create global phase space overlap to the reference state and samples regions of large uncertainty adaptively. New regions of low phase space overlap appear as jumps in the mean and maximum uncertainty and as eigenvalues of 1 to machine precision. Eigenvalues of the overlap matrix have a maximum of 1 and there will always be at least a eigenvalue of 1. The eigenvalues are shown as $1 - \lambda$ to show how close each value is to 1, instead of repeated nines in the decimals. As such, the closer $1 - \lambda$ is to zero, the closer λ is to 1, indicating a lack of phase space overlap in sampled states. Uncertainty in free energy is in units of kcal/mol, partial charge (q_i) in units of e^-, ϵ_{ij} in units of kcal/mol, and σ_{ij} in units of nm.

Iteration	**Max. Uncertainty**	**Mean Uncertainty**	**State Number**	q_i	ϵ_{ij}	σ_{ij}	Minimum five $1 - \lambda$ eigenvalues				
			1	0.0000	0.0239	0.2500					
			2	0.0000	0.0501	0.5731					
			3	0.0000	0.0764	0.7120					
			4	0.0000	0.1027	0.8111					
			5	0.0000	0.1290	0.8906					
			6	0.0000	0.1553	0.9579					
			7	0.0000	0.1912	0.3000					
			8	-2.0000	0.0501	0.5731					
			9	-1.8516	0.0501	0.5731					
			10	-1.6903	0.0501	0.5731					
Initial	53.405	16.162	11	-1.5119	0.0501	0.5731	0	0	$7.00{\cdot}10^{-08}$	$2.04{\cdot}10^{-03}$	$1.22{\cdot}10^{-02}$
			12	-1.3093	0.0501	0.5731					
			13	-1.0690	0.0501	0.5731					
			14	-0.7559	0.0501	0.5731					
			15	2.0000	0.0501	0.5731					
			16	1.8516	0.0501	0.5731					
			17	1.6903	0.0501	0.5731					
			18	1.5119	0.0501	0.5731					
			19	1.3093	0.0501	0.5731					
			20	1.0690	0.0501	0.5731					
			21	0.7559	0.0501	0.5731					

Continued on next page

Table C.2: Continued

Iteration	**Max. Uncertainty**	**Mean Uncertainty**	**State Number**	q_i	ϵ_{ij}	σ_{ij}	Minimum five $1-\lambda$ eigenvalues				
1	52.786	7.764	22	-1.6798	0.6820	0.4889	0	$1.90{\cdot}10^{-07}$	$9.88{\cdot}10^{-05}$	$5.07{\cdot}10^{-04}$	$1.06{\cdot}10^{-03}$
			23	-1.4749	0.4091	0.8854					
			24	-0.8544	0.0790	0.7594					
			25	1.6595	0.5888	0.6613					
			26	-0.5981	0.0704	0.7134					
			27	0.4314	0.1902	0.5986					
			28	-1.6634	0.6602	0.5463					
			29	-0.8917	0.0988	0.7682					
2	1.141	0.615	30	-0.7029	0.6529	0.8259	0	$1.18{\cdot}10^{-04}$	$3.30{\cdot}10^{-04}$	$6.78{\cdot}10^{-04}$	$1.10{\cdot}10^{-03}$
			31	1.1703	0.8006	0.6868					
			32	-0.0984	0.1345	0.6221					
			33	1.0954	0.7947	0.6935					
3	1.175	0.550	34	-0.6410	0.4941	0.4167	0	$1.80{\cdot}10^{-04}$	$5.52{\cdot}10^{-04}$	$9.29{\cdot}10^{-04}$	$1.21{\cdot}10^{-03}$
			35	1.2092	0.1541	0.9427					
			36	-0.3846	0.3165	0.4915					
			37	0.9190	0.1291	0.8801					
4	0.959	0.439	38	-1.5202	0.1299	0.6784	0	$3.45{\cdot}10^{-04}$	$8.54{\cdot}10^{-04}$	$1.18{\cdot}10^{-03}$	$1.85{\cdot}10^{-03}$
			39	1.3982	0.1985	0.7431					
			40	-1.3074	0.1187	0.6656					
			41	0.9507	0.1511	0.6975					
5	0.785	0.386	42	-1.3734	0.5789	0.7564	0	$3.22{\cdot}10^{-04}$	$8.33{\cdot}10^{-04}$	$1.33{\cdot}10^{-03}$	$1.70{\cdot}10^{-03}$
			43	1.7199	0.4999	0.9266					
			44	-1.2360	0.5260	0.7419					
			45	-1.3115	0.5773	0.7606					
6	0.742	0.353	46	-0.9298	0.5189	0.3859	0	$3.45{\cdot}10^{-04}$	$8.82{\cdot}10^{-04}$	$1.59{\cdot}10^{-03}$	$1.66{\cdot}10^{-03}$
			47	-1.8873	0.0676	0.6959					
			48	1.9708	0.6180	0.7441					
			49	-0.9112	0.5095	0.3917					
			50	-0.2264	0.0522	0.5907					
			51	-0.8718	0.5209	0.4012					

Continued on next page

Table C.2: Continued

Iteration	**Max. Uncertainty**	**Mean Uncertainty**	**State Number**	q_i	ϵ_{ij}	σ_{ij}	Minimum five $1-\lambda$ eigenvalues				
7	0.743	0.293	52	-1.8748	0.4493	0.7703	0	0	$4.56{\cdot}10^{-04}$	$9.31{\cdot}10^{-04}$	$1.54{\cdot}10^{-03}$
			53	-1.4899	0.8230	0.3421					
			54	0.7295	0.6144	0.8865					
			55	1.5584	0.1116	0.8817					
			56	0.1870	0.0575	0.6282					
			57	-1.5284	0.7857	0.4340					
			58	-1.6665	0.4625	0.7809					
			59	1.5418	0.1217	0.8818					
8	0.698	0.287	60	0.3923	0.7698	0.3943	0	0	$4.59{\cdot}10^{-04}$	$9.64{\cdot}10^{-04}$	$1.31{\cdot}10^{-03}$
			61	1.7228	0.1464	0.4531					
			62	1.6539	0.1425	0.4592					
			63	0.4455	0.7449	0.3970					
9	1.793	0.315	64	1.1971	0.1407	0.3559	0	0	$1.04{\cdot}10^{-04}$	$4.62{\cdot}10^{-04}$	$9.65{\cdot}10^{-04}$
			65	1.0055	0.1262	0.4082					
10	0.839	0.285	66	1.4059	0.0609	0.2523	0	0	0	$4.47{\cdot}10^{-04}$	$9.51{\cdot}10^{-04}$
			67	0.7310	0.0557	0.4622					
11	0.816	0.284	68	1.9054	0.2172	0.2961	0	0	0	0	$4.47{\cdot}10^{-04}$
			69	1.8673	0.2139	0.3079					
12	15.988	0.343	70	-1.8691	0.1965	0.4437	0	0	$1.38{\cdot}10^{-06}$	$2.18{\cdot}10^{-04}$	$6.19{\cdot}10^{-04}$
			71	1.3075	0.3699	0.2895					
			72	-0.5981	0.0970	0.5383					
			73	1.1244	0.3251	0.3615					
13	15.988	0.332	74	-1.9953	0.6150	0.4158	0	0	$1.38{\cdot}10^{-06}$	$4.99{\cdot}10^{-04}$	$8.38{\cdot}10^{-04}$
			75	-1.8357	0.5698	0.4330					
14	15.987	0.320	76	-1.4563	0.1414	0.4190	0	0	$1.38{\cdot}10^{-06}$	$5.30{\cdot}10^{-03}$	$9.56{\cdot}10^{-04}$
			77	1.6046	0.6396	0.3768					
			78	-1.2524	0.1286	0.4475					
			79	1.2836	0.5217	0.4317					

Continued on next page

Table C.2: Continued

Iteration	**Max. Uncertainty**	**Mean Uncertainty**	**State Number**	q_i	ϵ_{ij}	σ_{ij}	Minimum five $1-\lambda$ eigenvalues				
15	1.549	0.292	80	-1.8182	0.6459	0.3714	0	0	$2.79{\cdot}10^{-04}$	$4.74{\cdot}10^{-04}$	$8.07{\cdot}10^{-04}$
			81	-1.0881	0.3692	0.2903					
			82	1.4854	0.2016	0.3396					
			83	1.9247	0.4277	0.3114					
			84	-1.0010	0.3436	0.3349					
			85	1.3963	0.1925	0.3637					
			86	-1.7744	0.6293	0.3675					
			87	1.7226	0.3237	0.3250					
16	375.767	2.032	88	-1.8046	0.2576	0.3012	0	0	0	$5.90{\cdot}10^{-05}$	$2.80{\cdot}10^{-04}$
			89	-1.6602	0.2410	0.3426					
17	1.534	0.277	90	-1.3695	0.1634	0.2763	0	0	$7.98{\cdot}10^{-05}$	$4.21{\cdot}10^{-04}$	$8.07{\cdot}10^{-04}$
			91	-1.2204	0.5042	0.3138					
			92	-1.3421	0.1612	0.2902					
			93	-1.3546	0.1975	0.2805					
18	1.499	0.277	94	-1.7942	0.8037	0.2658	0	0	$6.04{\cdot}10^{-05}$	$4.21{\cdot}10^{-04}$	$8.07{\cdot}10^{-04}$
			95	-1.7583	0.7886	0.2809					
19	0.925	0.273	96	-1.3573	0.7020	0.2538	0	0	$2.25{\cdot}10^{-04}$	$5.76{\cdot}10^{-04}$	$8.07{\cdot}10^{-04}$
			97	-1.3301	0.6890	0.2705					
20	0.782	0.271	98	-1.6690	0.6204	0.2839	0	0	$2.53{\cdot}10^{-04}$	$6.36{\cdot}10^{-04}$	$8.08{\cdot}10^{-04}$
			99	-1.5355	0.5748	0.3306					
21	0.716	0.251	100	-1.9047	0.0793	0.3256	0	0	$2.75{\cdot}10^{-04}$	$7.74{\cdot}10^{-04}$	$8.12{\cdot}10^{-04}$
			101	-0.8341	0.7216	0.9577					
			102	-1.7905	0.0775	0.3523					
			103	-0.3003	0.2919	0.7587					

Continued on next page

Table C.2: Continued

Iteration	**Max. Uncertainty**	**Mean Uncertainty**	**State Number**	q_i	ϵ_{ij}	σ_{ij}	Minimum five $1-\lambda$ eigenvalues				
22	1.254	0.250	104	-1.9806	0.1162	0.2766	0	0	$2.35{\cdot}10^{-04}$	$5.36{\cdot}10^{-04}$	$7.33{\cdot}10^{-04}$
			105	-1.9768	0.7723	0.5198					
			106	-0.7425	0.5554	0.2759					
			107	0.8850	0.4370	0.2808					
			108	-1.9014	0.1136	0.3031					
			109	0.8673	0.4293	0.2942					
			110	-1.9028	0.7593	0.5108					
			111	0.7548	0.4465	0.2804					
23	0.811	0.249	112	-1.7238	0.2068	0.2500	0	0	$2.33{\cdot}10^{-04}$	$6.59{\cdot}10^{-04}$	$7.78{\cdot}10^{-04}$
			113	-0.8728	0.3653	0.2734					
			114	-0.7332	0.3148	0.3616					
			115	-1.5707	0.2353	0.2545					
24	45.245	0.308	116	-1.8840	0.5348	0.2500	0	$3.40{\cdot}10^{-07}$	$2.29{\cdot}10^{-04}$	$6.65{\cdot}10^{-04}$	$7.94{\cdot}10^{-04}$
			117	-2.0000	0.7852	0.8961					
			118	1.8905	0.3243	0.2500					
			119	-1.7333	0.4960	0.3088					
			120	1.8149	0.3133	0.2824					
			121	-1.8864	0.5398	0.3097					
25	0.712	0.235	122	-1.8495	0.0742	0.7687	0	$2.20{\cdot}10^{-04}$	$5.66{\cdot}10^{-04}$	$7.48{\cdot}10^{-04}$	$2.12{\cdot}10^{-03}$
			123	1.2859	0.1263	0.2500					
			124	1.1208	0.8537	0.9557					
			125	1.9858	0.7161	0.2500					
			126	-0.9987	0.7161	0.2500					
			127	1.2345	0.1233	0.2824					
			128	1.5659	0.3622	0.2500					
			129	1.9858	0.7161	0.2500					

Continued on next page

Table C.2: Continued

Iteration	**Max. Uncertainty**	**Mean Uncertainty**	**State Number**	q_i	ϵ_{ij}	σ_{ij}	Minimum five $1-\lambda$ eigenvalues				
26	4.200	0.237	130	-1.7965	0.0761	0.2500	0	$1.76{\cdot}10^{-05}$	$1.98{\cdot}10^{-04}$	$5.19{\cdot}10^{-04}$	$6.48{\cdot}10^{-04}$
			131	-1.1030	0.1372	0.2500					
			132	-0.5576	0.0754	0.2500					
			133	0.6581	0.2799	0.2500					
			134	-0.5464	0.2799	0.2672					
			135	0.6318	0.2707	0.2824					
			136	-1.7410	0.0810	0.2500					
			137	-1.0048	0.1261	0.2500					
27	1.024	0.233	138	-1.5042	0.0268	0.2500	0	$1.68{\cdot}10^{-04}$	$3.27{\cdot}10^{-04}$	$5.22{\cdot}10^{-04}$	$6.89{\cdot}10^{-04}$
			139	-1.1001	0.0268	0.2500					
			140	2.0000	0.8516	0.8046					
			141	-1.0341	0.2125	0.2962					
			142	1.8800	0.8035	0.7942					
			143	-1.3829	0.0856	0.2500					
28	0.716	0.233	144	-1.7489	0.0677	0.2500	0	$1.65{\cdot}10^{-04}$	$4.96{\cdot}10^{-04}$	$6.38{\cdot}10^{-04}$	$1.40{\cdot}10^{-03}$
			145	-1.9845	0.2315	0.3688					
			146	-1.2066	0.0742	0.2500					
			147	-1.1825	0.0737	0.2672					
			148	-1.8101	0.1103	0.2909					
			149	-1.5320	0.0703	0.2500					
29	0.641	0.233	150	-1.9604	0.2006	0.2500	0	$1.49{\cdot}10^{-04}$	$4.66{\cdot}10^{-04}$	$6.41{\cdot}10^{-04}$	$1.66{\cdot}10^{-03}$
			151	-1.1793	0.0288	0.2500					
			152	-0.7656	0.0534	0.2500					
			153	0.8766	0.0515	0.2500					
			154	1.6513	0.0648	0.2728					
			155	-0.7503	0.0533	0.2672					
			156	0.8064	0.0514	0.3088					
			157	-1.3043	0.0563	0.2500					
			158	-1.0469	0.0367	0.2500					
			159	0.9076	0.0520	0.2510					

Continued on next page

Table C.2: Continued

Iteration	**Max. Uncertainty**	**Mean Uncertainty**	**State Number**	q_i	ϵ_{ij}	σ_{ij}	Minimum five $1-\lambda$ eigenvalues				
30	0.641	0.199	160	0.6764	0.8382	0.3336	0	$1.60{\cdot}10^{-04}$	$5.41{\cdot}10^{-04}$	$1.11{\cdot}10^{-03}$	$1.71{\cdot}10^{-03}$
			161	1.4891	0.2481	0.9246					
			162	0.6629	0.8224	0.3424					
			163	0.4765	0.1135	0.7249					
31	0.636	0.168	164	1.8978	0.2442	0.7260	0	$1.79{\cdot}10^{-04}$	$6.66{\cdot}10^{-04}$	$1.49{\cdot}10^{-03}$	$1.85{\cdot}10^{-03}$
			165	-1.6132	0.6120	0.5550					
			166	-1.2083	0.3974	0.8977					
			167	0.6832	0.1200	0.6369					
			168	-1.0002	0.3985	0.5621					
			169	-1.4755	0.5390	0.5621					
32	0.638	0.162	170	1.6020	0.2264	0.4529	0	$7.73{\cdot}10^{-04}$	$1.51{\cdot}10^{-03}$	$1.85{\cdot}10^{-03}$	$1.87{\cdot}10^{-03}$
			171	0.2884	0.0819	0.5552					
33	0.636	0.158	172	-1.3653	0.4527	0.9329	0	$1.84{\cdot}10^{-04}$	$7.97{\cdot}10^{-04}$	$1.57{\cdot}10^{-03}$	$1.93{\cdot}10^{-03}$
			173	1.7108	0.3400	0.3091					
			174	-0.2731	0.1307	0.6790					
			175	1.5397	0.3110	0.6790					
34	0.634	0.151	176	-1.7402	0.1735	0.2968	0	$1.83{\cdot}10^{-04}$	$8.48{\cdot}10^{-04}$	$1.61{\cdot}10^{-03}$	$1.93{\cdot}10^{-03}$
			177	0.7764	0.8525	0.9370					
			178	1.5458	0.1252	0.6505					
			179	-1.1485	0.1316	0.4332					
			180	1.0821	0.1028	0.6292					
			181	1.5304	0.1398	0.6590					
35	0.634	0.141	182	-0.8934	0.6479	0.8374	0	$1.82{\cdot}10^{-04}$	$8.81{\cdot}10^{-04}$	$1.64{\cdot}10^{-03}$	$2.02{\cdot}10^{-03}$
			183	1.6751	0.1045	0.3914					
			184	-0.5360	0.4088	0.7534					
			185	1.2396	0.0904	0.4536					
36	0.634	0.138	186	-1.2996	0.6071	0.6355	0	$1.82{\cdot}10^{-04}$	$8.68{\cdot}10^{-04}$	$1.63{\cdot}10^{-03}$	$1.97{\cdot}10^{-03}$
			187	1.8999	0.7088	0.5807					
			188	-1.1696	0.5514	0.6298					
			189	1.8360	0.7067	0.5819					

Continued on next page

Table C.2: Continued

Iteration	**Max. Uncertainty**	**Mean Uncertainty**	**State Number**	q_i	ϵ_{ij}	σ_{ij}	Minimum five $1-\lambda$ eigenvalues				
37	0.633	0.127	190	-1.7443	0.2920	0.6633	0	$1.85{\cdot}10^{-04}$	$8.80{\cdot}10^{-04}$	$1.67{\cdot}10^{-03}$	$2.17{\cdot}10^{-03}$
			191	0.5927	0.5728	0.9083					
			192	-0.1047	0.0647	0.5794					
			193	0.2188	0.5279	0.8777					
38	0.633	0.123	194	-1.9705	0.7639	0.8696	0	$1.84{\cdot}10^{-04}$	$8.78{\cdot}10^{-04}$	$1.66{\cdot}10^{-03}$	$2.16{\cdot}10^{-03}$
			195	1.7253	0.5893	0.9258					
			196	0.0345	0.0610	0.5852					
			197	-1.7487	0.7534	0.8732					
39	0.633	0.119	198	-1.6028	0.7917	0.8307	0	$1.84{\cdot}10^{-04}$	$9.30{\cdot}10^{-04}$	$1.68{\cdot}10^{-03}$	$2.21{\cdot}10^{-03}$
			199	1.3667	0.4290	0.5441					
			200	0.1093	0.0805	0.5710					
			201	1.3073	0.4362	0.5533					
40	0.631	0.118	202	-1.3977	0.3683	0.5108	0	$1.89{\cdot}10^{-04}$	$9.21{\cdot}10^{-04}$	$1.70{\cdot}10^{-03}$	$2.24{\cdot}10^{-03}$
			203	-1.0064	0.2793	0.5297					

C.6 Corrections to Simulation Free Energies for Comparison to Born Solvation Model

The choice of simulation parameters and settings introduces various errors which will change the free energy estimate in predictable ways. Many of the calculations performed in Chapter 4 were between simulations carried out under the same boundary conditions, ensembles, electrostatic treatment, etc. Removing these errors introduced from the simulation settings would therefore apply to all results, and only shift the answer, but not the difference in properties, such as free energy differences. If we want to compare the free energy estimate between methods, we must correct the simulated results to provide a methodological independent free energy estimate. One example of these corrections is the ideal gas expansion comparison simulations and experimental results that we accounted for in the results from Joung and Cheatham in Section 4.4.4.

Comparing our free energy estimates from simulation to the Born approximation to solvation free energy requires removing the methodological dependence from our simulations. Hünenberger and Reif provide an excellent account of all the neglected physical factors in the Born model, and how to remove the methodological dependence for atomistic simulations. [184] This section details the corrections we applied to our free energy estimates. The result of the these corrections and the comparison of our simulation to the Born approximation are shown in Section 4.4.4. We use the similar terminology and variables as in Hünenberger and Reif, but not exact; as such, we show which equations and tables in the source material these corrections came from inside angle braces: e.g. {Eq. 6.1}. As a reminder, we ran with periodic boundary conditions under a lattice-sum electrostatics scheme.

The free energy of solvation, ΔG is

$$\Delta_s G = \Delta_s G^{raw}_{chg} + \Delta_s G_{cor} + \Delta_s G_{cav} + \Delta_s G_{std} \tag{C.6}$$

where the subscript s stands for simulation, $\Delta_s G^{raw}_{chg}$ is the free energy estimate of charging an ion of fixed size from zero charge, $\Delta_s G_{cor}$ are the correction terms to remove methodological dependence, and $\Delta_s G_{std}$ is the isothermal ideal-gas compression at standard state {Eq. 6.42}. We have changed from the general free energy F to the Gibbs free energy G as we sampled with the NPT ensemble. Our comparison to the Born model was a deviation of the Born model to our results for charging only. This means that we subtracted off the free energy of the uncharged particle of the same size for both models, removing the need to calculate $\Delta_s G_{cav}$ and $\Delta_s G_{std}$ as they would cancel out of the calculation. $\Delta_s G_{cor}$ for periodic boundary conditions with lattice-sum electrostatics is

$$\Delta_s G_{cor} = \Delta_s G_B + \Delta_s G_{C1} + \Delta_s G_{C2} + \Delta_s G_D \tag{C.7}$$

where each term on the right hand side corresponds to a different type of correction {Eq. 6.43} which we detail below.

$\Delta_s G_B$ removes the error in solvent polarization introduced from the finite simulated system size and periodicity. This correction is analytical for the spherical ion surrounded by a periodic lattice-sum solvent and is {Eq. 6.20} of the source material as

$$\begin{aligned}\Delta_s G_B = {} & (8\pi\epsilon_0)^{-1} N_A q^2 \left(1 - \epsilon_d^{-1}\right) L^{-1} \\ & \times \left[\alpha_{LS} + \frac{4\pi}{3}\left(\frac{R_{ij}}{L}\right)^3 - \frac{16\pi^2}{45}\left(\frac{R_{ij}}{L}\right)^5\right]\end{aligned} \tag{C.8}$$

where ϵ_0 is the dielectric of vacuum, N_A is Avogadro's constant, q is the charge of the solute particle in units of elementary charge e, $\epsilon_d = 92$ is the model solvent dielectric for TIP3P water, [183] L is the average box length for the cubic simulation box evaluated at an uncharged particle of the same size, $\alpha_{LS} \approx -2.837297$ is the

lattice-sum self-term constant {Eq. 3.28}, and R_{ij} is the Born radius which we assume is the effective hard sphere (EHS) radius as we computed in Section 4.4.4.

$\Delta_s G_{C1}$ corrects for the error introduced by evaluating the electrostatics of complete molecules up to a cutoff, instead of evaluating up to a spherical cutoff discarding atoms outside the cutoff, but bound to a fragment of the molecule inside the cutoff {Section 3.3.3}. We evaluate this correction by substituting {Eq. 6.27 and Eq. 6.45} into {Eq. 6.30}

$$\Delta_s G_{C1} \approx \frac{-N_A q(N_S+1)}{6\epsilon_0 L^3}\left(1-\frac{4\pi R_{ij}^3}{3L^3}\right)\mathcal{Q} \tag{C.9}$$

where N_S is the number of solvent molecules, and $\mathcal{Q} = 7.64 \cdot 10^{-3} e \cdot nm^2$ is the quadrupole-moment trace of the TIP3P water model {Table 3.1}.

$\Delta_s G_{C2}$ corrects for the vanishing average potential in the lattice-sum model as the potential is evaluated towards the edge of the box. This results in an omitted zero term in the Fourier series and offsets the potential at the solute center. The lattice-sum correction is {Eq. 6.37} and

$$\Delta_s G_{C2} = -N_A q \frac{4\pi R_{ij}^3}{3L^3}\left(\chi_S + \frac{\tilde{\chi}_{S-}}{R_{ij}}\right) \tag{C.10}$$

where $\chi_S = 0.73V$ is the interface potential at a planar air-liquid interface measured in the air-to-liquid direction, and $\tilde{\chi}_{S-} = -0.11V \cdot nm$ is a factor characterizing the near approximate linear dependence of the air-liquid interface potential measured in the same direction. The values used for χ_S and $\tilde{\chi}_{S-}$ are taken from SPC water at 300K as these were the data available, so $\Delta_s G_{C2}$ is an approximation for our purposes.

$\Delta_s G_D$ corrects for the fact that the solvent model relative dielectric, ϵ_d, differs from the experimental relative dielectric, ϵ_e. In the case of water, $\epsilon_e = 78.36$ {Table 1.1}. We compute this correction term as {Eq. 6.41} and

$$\Delta_s G_D = \frac{N_A q^2\left(\epsilon_e^{-1} - \epsilon_d^{-1}\right)}{8\pi\epsilon_0 R_{ij}} \tag{C.11}$$

Finally, combining all of these corrections into our simulated results allows us to make a methodologically independent comparison between our results and the Born approximation to solvation free energy.

D.1 Construction of the Examol

We want as simple a system as we can design while still having enough details to sample a wide range of molecular types. The following molecules will provide a sound benchmark

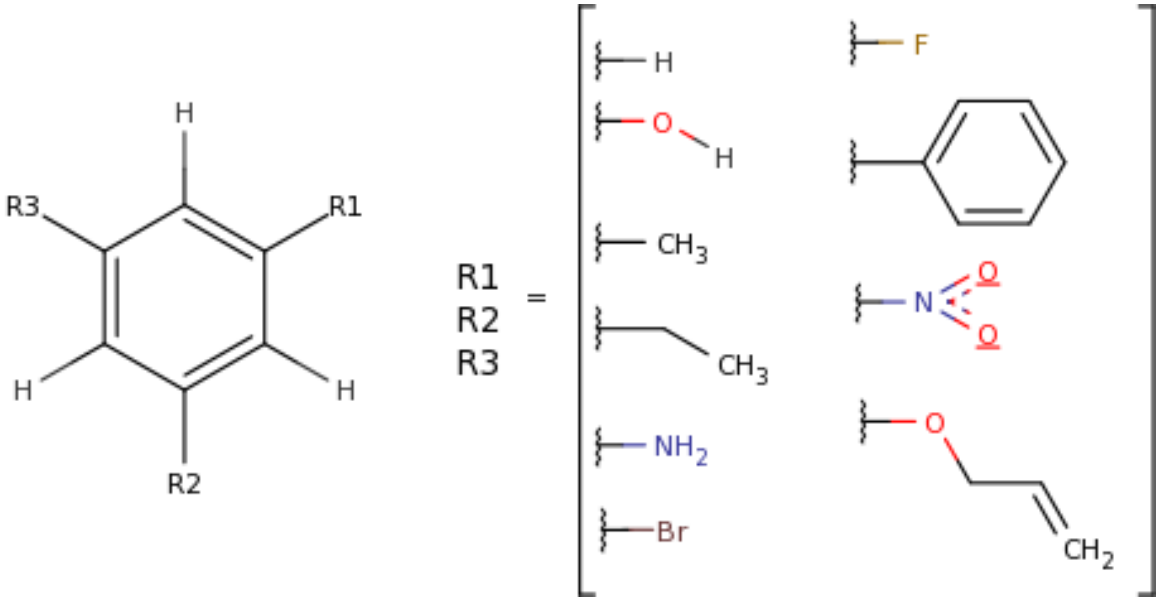


Figure D.1: A common benzene core with six independent R groups all sharing the same substituent set. With only 3 mutation sites, 10^3 molecules are possible with 220 unique ones due to symmetry. With all 6 sites mutatable, 10^6 total molecules are possible, with only 86,185 unique molecules.

We have in Fig. D.1 a benzene ring as common core with $N_C = 3$ carbon atoms to attach R-groups to, and $N_S = 10$ substituents per R-group giving $N_S^{N_C} = 10^3$ possible combinations, but only 220 of the molecules will be chemical unique from symmetry operations, computed by Burnside's Lemma. [213]. We can check the identical molecules as validation that we get the same free energy of solvation from both of them. These diverse groups provide a diverse set of charges and sizes, as well as steric hindrances to one another, forcing arrangement between R-groups.

Electrostatics on the common carbons are modified, but their Lennard-Jones terms will not. None of the R-groups break the aromaticity of the benzene ring, so the carbons can always be treated as aromatic carbons. However, as chemical identity of

the examol changes, the partial charges on the common carbon cores may change as well.

D.1.1 Optional: Simplifying the Examol by Reducing Number of Interactions

We can tabulate intramolecular nonbonded forces with rigid constraints. Fully rigid benzene, constraining bonds and torsions, allows tabulating all intramolecular nonbonded interactions. If only bonds are constrained, small deviations in the torsional forces may cause complications if we assume they are constant. Imposing these constrains will allow testing other energy evaluations and sampling methods without having to worry about intramolecular nonbonded interactions first. The deviations in free energy from unconstrained should be small given the rigid nature of the molecule. This option is not exercised in our work.

We chose only to do initial tests with 3 mutatable carbon sites. It is more helpful for small scale testing to only do the para- substituents (2,4,6), reducing the number of real molecules down to 10^3. Only having para- substituents also removes torsions between different R-groups, simplifying initial setup and data collection.

For simulating a real protein-ligand, we can look at the binders and non-bonded tested on the T4-Lysozyme molecule. We can look at both the polar and non-polar site and see if we can construct an Examol which has all the binders as individual R-groups off the common carbon atoms. This does mean that the non-alchemical ligand atoms may not be a benzene ring, but instead just two carbons linked together. So long as we have at least two carbons non-alchemically modified, we will not need to worry about intramolecular angle forces (explained below).

D.1.2 Building the Examol in a Simulation

A single super-structure molecule is simulated in OpenMM. Our molecule consists of the benzene-ring core, then all 10 R-groups coming from each carbon. We create multiple files, 1 "core file" and 30 "R-files." The core file is read into OpenMM and define all the coordinates bonded terms for the core and will not change with state. Each R-file has the coordinates for the different substituents, e.g. three R-files have the nitro groups, one at each mutatable site. The R-files also have the common core present so its easier to build of existing files. The common cores are aligned between the core file and the R-groups, then the substituent coordinates and bonded terms are combined into a single molecule with appropriate λ assignment.

D.2 Assigning λ Values to Examol

Substituents on a single R-group do not interact with each other. We define the Interaction Groups inside OpenMM to prevent these groups from calculating (see section D.4.1). This reduces the number of energy evaluations and remove overlapping atom energies.

Each R-group is assigned an i-th index, λ_i, each substituent is then assigned a j-th index, $\lambda_{i,j}$, where $i = \{0, 1, 2, 3\}$ for $N_i = 3$ and $j = \{0, 1, 2, \ldots, 10\}$ for $N_j = 10$. Each λ falls within the range of

$$0 \leq \lambda_{i,j} \leq 1 \tag{D.1}$$

$$\tag{D.2}$$

A total of $N_C N_S = N_\lambda = 30$ λ values are recorded to determine which thermodynamic state the molecule is in. Each state is defined as a tuple of 30 floats on the domain $[0, 1]$ since we will divide the λ range into different basis functions.

We use $N_B = 4$ basis functions per substituent (Electrostatics, LJ to Cap, Repulsive, and Attractive) with reaction field electrostatics to define most of the alchemical interactions. This results in $N_C N_S N_B + 2 = N_U = 120$ energies per sampled time step that must be evaluated and stored. The extra two comes from the energy of the non-alchemical system, and the total potential energy. This is significantly cheaper than storing 10^3 energies plus any intermediate states as would be required with soft core methods, and the combination of Interaction and Force Groups of OpenMM will make sure these calculations are quick since we can query each R-group separately. Disk space required is comparable to storing the coordinates of 40 extra atoms at each time step. There are additional basis functions to evaluate for alchemical atoms interacting with other alchemical atoms, however, these are computationally cheap as they are over much fewer atoms and discussed below.

Energy evaluations will be limited at a certain point. If we try to extend this method too far, there is a threshold when we are doing more energy evaluations than soft core. This is reached when

$$\begin{aligned} N_S^{N_C} &< N_C N_S N_B \\ N_S^{N_C - 1} &< N_C N_B \end{aligned} \tag{D.3}$$

which will only happen at low N_C or N_S and/or an unreasonably large N_B, the later of which will not happen. Because we want this method to be used for even larger systems eventually, this threshold is unlikely to be reached.

Within OpenMM, each lambda will be assigned as a global parameter to its force and indexed accordingly. There will likely need to be $N_B N_S$ forces held in memory to update the parameters, but this should be relatively low cost.

D.3 Choosing a starting molecular state

The initial state is set to the decoupled state so multiple simulations worth of data all have a common starting state, and we only have to store one equilibrated starting structure. Generally high entropy states (small R groups) should be started with to maximize initial transition rates.

For notation shorthand, "TS" will stand for "Thermodynamic State."

D.4 Classifying Types of Interactions

Total potential energy is simple enough to evaluate, however the basis functions are not. The potential energy can be decomposed in the following manner

$$\begin{aligned} U_{\text{total}} &= U_N(\mathbf{u}(r)) \\ &+ \sum_{i}^{N_i} \sum_{j}^{N_j} U_{AN}(\mathbf{u}(r), \lambda_{i,j}) \\ &+ \sum_{i}^{N_i} \sum_{j}^{N_j} \sum_{i_2=i+1}^{N_i} \sum_{j_2}^{N_j} U_{AA}(\mathbf{u}(r), \lambda_{i,j}\lambda_{i_2,j_2}) \end{aligned} \tag{D.4}$$

where each term represents the following interaction types:

- U_N are the non-alchemically modified interactions, this is dominated by solvent-solvent interactions. This also includes the common core Lennard-Jones interactions with itself, or intra-R-group interactions as we are only decoupling the R-groups, not annihilating them.

- U_{AN} are alchemical interactions controlled by a single λ variable. These are predominantly the solute-solvent interactions, with some additional bonded R-group/core interactions.

- U_{AA} are alchemical-alchemical interactions and mainly inter-R-group solute interactions. These are more complicated to evaluate since we have not previously defined how they should be computed and are controlled by both a $\lambda_{i,j}$ and a λ_{i_2,j_2}. We note that the third summation of the U_{AA} interactions only loops over a subset of N_i since R-groups on the same site do not interact with each other, and so as not to double count the interactions.

The following subsection summarizes the OpenMM interaction and force groups to make computational efficient energy evaluations. Details of each evaluation are in the sections following the summary.

D.4.1 Summary of OpenMM Force and Interaction Group Assignments

We expand the number of force groups available to OpenMM but remove the combinatorial selection of them. OpenMM natively has 32 force groups that can be selected in any combination by passing in a 32 bit integer that serves as a bitmask. We wish to reduce the number of forces and particles called in any energy call so we can compute $\partial u/\partial\lambda$ efficiently. We expanded the number of available force groups to 512, but made it so the integer is simply the number of the group, and not a bitmask. The ForceGroup assigned to any given force was based on why type of alchemical

interactions were involved according to the logic

$$\text{Group} = \begin{cases} 0 & \text{for non-alchemical/non-alchemical} \\ 1 + i \cdot N_i * \cdot j & \text{for alchemical/non-alchemical} \\ N_i \cdot N_j + 1 & \\ \quad + \dfrac{i \cdot (2 \cdot N_i - i - 1)}{2} N_j^2 & \\ \quad + j \cdot N_j \cdot (N_i - i - 1) & \text{for alchemical/alchemical} \\ \quad + N_j \cdot (i_2 - i - 1) & \\ \quad + j_2 & \end{cases} \tag{D.5}$$

By assigning different sets of atoms to different force groups, we can query very small numbers of atoms over which we compute the force energy, after updating the various λ. Because so few atoms are looped over in any one force group, the computational efficiency should be faster despite the number of calls to energy evaluations. Note that there is never any $\lambda_{i,*}$ interacting with the same $\lambda_{i,*}$ as these interactions should never be observed. However, there does exist an interaction between atoms on the same $\lambda_{i,j}$, which have been spun off into their own force group as they are a constant with the change of state. The number of force groups can be reduced, but then more λ must be updated to evaluate basis functions and more atoms must be looped over every time.

D.4.2 Non-Alchemical/Non-Alchemical Potential Energy Evaluation

These energies compose all of U_N are straightforward to evaluate. All of the solvent atoms are added to their own interaction and force groups to speed up the calculation of this energy. Electrostatics are handled by PME for accuracy, and because we do not have to worry about computing the long range electrostatic contribution alchemically as discussed in Chapter 3 and section 5.3. Because our solute core has an alchemically

changing partial charge, only the only solute interactions in this group are the common core Lennard-Jones interactions. These interactions do not need run through the Custom Nonbonded Force and should be very quick to evaluate.

D.4.3 Alchemical/Non-Alchemical Potential Energy Evaluation

The U_{AN} class of interaction computes all of the solvent-solute interactions and is the most expensive part of the energy evaluation. We create assign a force group for this type of interaction for each $\lambda_{i,j}$ and the alchemical atoms they control. This way, we are not excessively looping over and subsequently ignoring through a $\times 0$ multiplier all the solute-solute and intramolecular interactions. This also excludes atoms on the same common carbon atom from interacting as they would not in a realistic molecule anyways, again reducing the number of atoms we must loop over. Electrostatics are handled by reaction field initially to remove the long range electrostatic interactions that we observed from Chapter 3 and section 5.3.

This process could theoretically be accelerated if we compute the interactions of each force group in parallel, but that is beyond the scope of this work.

D.4.4 Alchemical/Alchemical Potential Energy Evaluation

Computing U_{AA} requires its own routine since the number of evaluations is large. Each intramolecular alchemical/alchemical interaction depends on the pair of $(\lambda_{i,j}, \lambda_{i_2,j_2})$ being evaluated. This means there is an enumerated N_λ^2 non-unique combinations that could be evaluated, however, the work load is significantly less than alchemical/non-alchemical interactions. We make a few assumptions about the interactions and then simplify the number of energy evaluations we carry out and store.

The following rules are applied to this class of interactions:

- Atoms controlled by the same $\lambda_{i,j}$ interact fully at all times. E.g. the two oxygens on a nitro- group always interact fully.

- Atoms on $\lambda_{i,j}$ which share the same common carbon atom with atoms of $\lambda_{i,k\neq j}$ do not interact.

- The potential energy of evaluating $\lambda_{i,j}$ atoms interacting with $\lambda_{h\neq i,*}$ atoms is identical to $\lambda_{h\neq i,*}$ atoms interacting with $\lambda_{i,j}$ atoms. This follows standard potential energy rules from most simulation packages.

- The total alchemical potential of atoms on $\lambda_{i,j}$ interacting with atoms on $\lambda_{h\neq i,*}$ is $U_{AA}(r, \lambda_{i,j}, \lambda_{h.k}) = \lambda_{i,j}\lambda_{h,k}u(r)$, i.e. the non-alchemical potential of the two sets of atoms scaled by the product of the their λ's. These pairs use linear scaling as they should not frequently, if ever, overlap each other as they do with the solvent. If need be, we can change this to using multiple basis functions if it improves statistical efficiency.

This scheme means there is a single basis function per unique pair of $\lambda_{i,j}$ and $\lambda_{h,k}$. Because there are no interactions between atoms on different $\lambda_{i,*}$ but the same common carbon, our total number of basis functions we need to evaluate is:

$$N_{B,\text{intra}} = N_S^2 \left[N_C^2 - \frac{N_C\left(N_C + 1\right)}{2} \right] \tag{D.6}$$

which goes to $N_\lambda^2/2$ in the limit of large N_C. For our system of $N_C = 3$ and $N_S = 10$, this is 300 energies which will require the equivalent disk space of 100 atom coordinates per iteration.

Evaluating these energies is quicker than first appears. Remember that each substituent group only has a few atoms on them, and even more complex groups will have significantly fewer atoms than the entire system. Tryptophan for example only has 27 atoms and it is unlikely larger alchemical groups will be appeared. This

means that if we assign all the atoms in a λ group to an interaction group to the other alchemical atoms, then we are looping over very few atoms for each force evaluation. If we control this with reaction field electrostatics, then we do not have to worry about PME complications in the energy evaluation. We further reduce the computational cost by assigning each pair of $(\lambda_{i,j}, \lambda_{i_2,j_2})$ into its own force group, who's number is computed by the third condition of Eq. (D.5).

Only one basis function is needed per unique λ pair interaction since solvent would need displaced first to have the atoms on top of each other. In all physical molecules, there will never be atoms from two R-groups on top of each other. Any phase space which has such atomic overlap does not need to be heavily sampled as it shares very little phase space overlap with the molecules of interest. Further, there is always be water present which would need to be displaced before such an interaction could occur. Because we need the water to leak into the atomic sites from different R-groups and not the R-groups leaking into each other, there is no need for energy decomposition beyond linear scaling.

D.5 Computing Types of Interactions

This section details the equations used to compute all alchemical interactions.

Harmonic Bonds: ALL

No harmonic bonds are modified to preserve molecular structure.

Harmonic Angle Force: Alchemical/Non-alchemical

$$U_{\text{angle}}(\theta, \lambda_{i,j}) = \lambda_{i,j} \frac{1}{2} k(\theta - \theta_0)^2 \tag{D.7}$$

where θ is the angle formed by the three particles, θ_0 is the equilibrium angle, and k is the force constant. At least one particle must be an alchemical particle and one particle must be common core.

Harmonic Angle Force: Alchemical/Alchemical

Because we are mutating benzene, there are no angles formed between alchemical particles on different R-groups. The more general case would use Eq. (D.7) but replace $\lambda_{i,j}$ with $\lambda_{i,j}\lambda_{i_2,j_2}$.

Periodic Torsion Force: Alchemical/Non-alchemical

$$U_{\text{tors}}(\theta, \lambda_{i,j}) = \lambda_{i,j} k \left(1 + \cos\left(n\theta - \theta_0\right)\right) \tag{D.8}$$

where θ is the dihedral angle formed by the four particles, θ_0 is the equilibrium angle, k is the force constant, and n is the periodicity. At least one particle must be an alchemical particle and one particle must be common core.

Periodic Torsion Force: Alchemical/Alchemical

Because we are mutating the 2,4,6 positions, there are no dihedral formed between alchemical particles on different R-groups. The more general case, or if all six sites of benzene were mutated, would use Eq. (D.8) but replace $\lambda_{i,j}$ with $\lambda_{i,j}\lambda_{i_2,j_2}$.

Nonbonded Force: Alchemical/non-alchemical

These are controlled through the AR/C/E-WCA schedule detailed at length in Chapter 3. Each $\lambda_{i,j}$ variable maps on to the value each $h_{E,C,A,R}(\lambda_{i,j})$ switch with one-to-one values.

Nonbonded Force: Alchemical/Alchemical

These are controlled through the AER-WCA schedule with linear scaling. Reasoning is listed in the previous section and section 5.2.2. Each alchemical switch follows $h_{E,A,R}(\lambda_{i,j}\lambda_{i_2,j_2}) = \lambda_{i,j}\lambda_{i_2,j_2}$ since the switch is linear. Because each switch has the same value, there is a single basis function for these interactions, all scaled by $\lambda_{i,j}\lambda_{i_2,j_2}$, for a given pair of $\lambda_{i,j}$ and λ_{i_2,j_2}.

D.6 λDX Sampling Algorithm

This section details the pseudo-code used in our CustomIntegrator built into OpenMM. Full code is available on GitHub. [214].

1. Draw Cartesian particle velocities from the Maxwell-Boltzmann distributions according to

$$v_c = \sqrt{k_B T / m_c} \cdot N(0,1) \tag{D.9}$$

   where v is velocity (in 3D), subscript c indicates variables in Cartesian space, k_B is the Boltzmann constant, T is target temperature, m is particle mass, and $N(0,1)$ is a draw from a normal Gaussian distribution.

2. Draw alchemical walker velocities from the Maxwell-Boltzmann distributions according to

$$v_\lambda = \sqrt{k_B T / m_\lambda} \cdot N(0,1) \tag{D.10}$$

   for each λ variable where the subscript λ indicates variables in λ space.

3. Compute total initial energy, E_O including total potential, kinetic energy in Cartesian space, and kinetic energy in λ space.

4. Disable alchemical/alchemical interactions to switch into approximate energy space for faster internal MC loops

5. Compute total approximate initial energy, E'_O

6. Carry out Hybrid MC time steps. We use the following symmetric integration steps with Δt the time step symbol, $p_{\{c,\lambda\}}(\Delta t)$ for momentum steps, and $q_{\{c,\lambda\}}(\Delta t)$ for position steps.

$$p_\lambda(1/2\Delta t)p_\lambda(1/2\Delta t)p_c(1/2\Delta t)q_c(\Delta t)q_\lambda(\Delta t)p_c(1/2\Delta t)p_\lambda(1/2\Delta t) \tag{D.11}$$

We note that the $q_\lambda(\Delta t)$ and $q_c(\Delta t)$ are commutable operators. We also combine the first and last steps of this MD propagation algorithm to reduce the number of force evaluations without breaking symmetry.

For $q_\lambda(\Delta t)$: Use hard, reflective walls at $\lambda = 0$ or $\lambda = 1$ which reverse momentum in the event that a λ escapes its $[0, 1]$ domain. Ensure the positions and velocities in λ space are adjusted before a new force call is made.

7. Compute total approximate final energy, $E_N^{'}$

8. Accept the MD moves for HMC with acceptance (see also Eq, (5.3))

$$\alpha_{O\to N}^{'} = \min\left(\exp\left[-\beta\Delta E^{'}\right], 1\right) \tag{D.12}$$

9. If rejected: reset all positions and velocities in both Cartesian and λ space.

   Since we only have one attempted inner HMC move per outer MC move, we do not have to re-randomize velocities at this point, only reset them.

10. Enable alchemical/alchemical interactions to switch back into full potential and out of approximate potential.

11. Compute total final energy, E_N

12. Accept the *entire* process with acceptance (see also Eq. (5.4))

$$\alpha_{O\to N} = \min\left(\exp\left[-\beta(\Delta E - \Delta E')\right], 1\right) \tag{D.13}$$

13. Repeat procedure.

Acknowledgments

I would like to thank the entire Shirts Research Group at the University of Virginia and at Colorado University Boulder for all their time, support, and input to my work over the years. I would also like to thank the following people for helpful discussion and assistance with code: John Chodera and Kyle Beauchamp at Memorial Sloan-Kettering Cancer Center, Peter Eastman and the rest of the OpenMM team at Stanford. I'd also like to thank David Mobley, Lingle Wang, and Mark Abraham for feedback. I thank Jacob Monroe, former group member at University of Virginia for simulation data on the host-guest systems. Finally, I would like to acknowledge funding from the NSF through grant CHE-1152786.

Bibliography

[1] Mcdonnell Douglas. Boeing's Seventh Wonder. *IEEE Spectr.*, 23(10):20–23, 1995.

[2] Walter S. Woltosz. If we designed airplanes like we design drugs. *J. Comput. Aided. Mol. Des.*, 26(1):159–163, 2012.

[3] Jonathan Kang. What is it like to design CPUs for a living?, 2012.

[4] Roland E. Meissner. Plant Layout. In *Kirk-Othmer Encycl. Chem. Technol.* John Wiley & Sons, Inc., New York, NY, 2000.

[5] Jaydeep Balakrishnan, Chun Hung Cheng, Daniel G. Conway, and Chun Ming Lau. A hybrid genetic algorithm for the dynamic plant layout problem. *Int. J. Prod. Econ.*, 86(2):107–120, 2003.

[6] A. Burdorf, B. Kampczyk, M. Lederhose, and H. Schmidt-Traub. CAPD - Computer-aided plant design. *Comput. Chem. Eng.*, 28(1-2):73–81, 2004.

[7] D. Hofmann, M. Heuchel, Yu Yampolskii, V. Khotimskii, and V. Shantarovich. Free volume distributions in ultrahigh and lower free volume polymers: Comparison between molecular modeling and positron lifetime studies. *Macromolecules*, 35(6):2129–2140, 2002.

[8] P. Bernardo, E. Drioli, and G. Golemme. Membrane gas separation: A review/state of the art. *Ind. Eng. Chem. Res.*, 48(10):4638–4663, 2009.

[9] Jie Xiao, Yinlun Huang, and Charles W. Manke. Computational Design of Polymer Nanocomposite Coatings: A Multiscale Hierarchical Approach for Barrier Property Prediction. *Ind. Eng. Chem. Res.*, 49(17):7718–7727, sep 2010.

[10] Jie Xiao, Yinlun Huang, and Charles Manke. Computational design of thermoset nanocomposite coatings: Methodological study on coating development and testing. *Chem. Eng. Sci.*, 65(2):753–771, 2010.

[11] I. M. Kapetanovic. Computer-aided drug discovery and development (CADDD): In silico-chemico-biological approach. *Chem. Biol. Interact.*, 171(2):165–176, 2008.

[12] Riccardo Baron, editor. *Computational Drug Discovery and Design*. Humana Press, New York, NY, 2012.

[13] Honglin Li, Mingyue Zheng, Xiaomin Luo, Weiliang Zhu, and Hualiang Jiang. Computational Approaches to Drug Discovery and Development. In *Chem. Biol. Approaches to Drug Discov. Dev. to Target. Dis.*, pages 23–40. John Wiley & Sons, Inc., Hoboken, NJ, 2012.

[14] Niels Hansen and Wilfred F. van Gunsteren. Practical Aspects of Free-Energy Calculations: A Review. *J. Chem. Theory Comput.*, pages 2632–2647, May 2014.

[15] Christopher M Dobson. Chemical space and biology. *Nature*, 432(7019):824–828, December 2004.

[16] Radislav a Potyrailo, Krishna Rajan, Klaus Stowe, Ichiro Takeuchi, Bret Chisholm, and Hubert Lam. Combinatorial and High-Throughput Screening of Materials Libraries : Review of State of the Art Combinatorial and High-Throughput Screening of Materials Libraries : Review of State of the Art. *ACS Comb. Sci.*, 13:579–633, 2011.

[17] Andrew R. Leach and Michael M. Hann. Molecular complexity and fragment-based drug discovery: Ten years on. *Curr. Opin. Chem. Biol.*, 15(4):489–496, 2011.

[18] Peter G. Schultz. Commentary on combinatorial chemistry. *Appl. Catal. A Gen.*, 254(1):3–4, 2003.

[19] Leslie G. Aronovitz, Geraldine Redican-Bigott, Shirin Hormozi, Julian Klazkin, David Lichtenfeld, and Stephen Ulrich. New Drug Development: Science, Business, Regulatory, and Intellectual Property Issues Cited as Hampering Drug Development Efforts. Technical Report November, United States Government Accountability Office, Washington, DC, 2006.

[20] Tingting Liu, Dong Lu, Hao Zhang, Mingyue Zheng, Huaiyu Yang, Yechun Xu, Cheng Luo, Weiliang Zhu, Kunqian Yu, and Hualiang Jiang. Applying high-performance computing in drug discovery and molecular simulation. *Natl. Sci. Rev.*, 3(1):49–63, 2016.

[21] David E Shaw, Martin M Deneroff, Ron O Dror, Jeffrey S Kuskin, Richard H Larson, John K Salmon, Cliff Young, Brannon Batson, Kevin J Bowers, Jack C Chao, Michael P Eastwood, Joseph Gagliardo, J P Grossman, C Richard Ho, Douglas J Ierardi, István Kolossváry, John L Klepeis, Timothy Layman, Christine Mcleavey, Mark A Moraes, Rolf Mueller, Edward C Priest, Yibing Shan, Jochen Spengler, Michael Theobald, Brian Towles, and Stanley C Wang. Anton, a Special-Purpose Machine for Molecular Dynamics Simulation. *Commun. Am. Chem. Soc.*, 51(7):91–97, 2008.

[22] John E. Stone, David J. Hardy, Ivan S. Ufimtsev, and Klaus Schulten. GPU-accelerated molecular modeling coming of age. *J. Mol. Graph. Model.*, 29(2):116–125, 2010.

[23] Weiguo Liu, Bertil Schmidt, Gerrit Voss, and Wolfgang Müller-Wittig. Accelerating molecular dynamics simulations using Graphics Processing Units with CUDA. *Comput. Phys. Commun.*, 179(9):634–641, 2008.

[24] J. A. van Meel, A Arnold, D Frenkel, S. F. Portegies Zwart, and R. G. Belleman. Harvesting graphics power for MD simulations. *Mol. Simul.*, 34(3):259–266, 2008.

[25] MS Friedrichs, Peter Eastman, V Vaidyanathan, M Houston, L LeGrand, A.L. Beberg, D.L. Ensign, C.M. Bruns, and Vijay S. Pande. Accelerating molecular dynamic simulation on graphics processing units. *J. Comput. Chem.*, 30(6):864–872, 2009.

[26] Michael R. Shirts and Vijay S. Pande. Screen Savers of the World Unite! *Science*, 290(5498):1903–1904, 2000.

[27] Gideon Juve, Ewa Deelman, Karan Vahi, Gaurang Mehta, Bruce Berriman, Benjamin P Berman, and Phil Maechling. Scientific Workflow Applications on Amazon EC2–includesz HPC and cloud. In *Proc. 5th IEEE Int. Conf. E-Science Work.*, pages 59–66, 2009.

[28] Kai J Kohlhoff, Diwakar Shukla, Morgan Lawrenz, Gregory R Bowman, David E Konerding, Dan Belov, Russ B Altman, and Vijay S Pande. Cloud-based simulations on Google Exacycle reveal ligand modulation of GPCR activation pathways. *Nat. Chem.*, 6(1):15–21, 2014.

[29] Pengyu Ren and Jay W. Ponder. Consistent treatment of inter- and intramolecular polarization in molecular mechanics calculations. *J. Comput. Chem.*, 23(16):1497–1506, 2002.

[30] Jay W Ponder, Chuanjie Wu, Vijay S Pande, John D Chodera, Michael J Schnieders, Imran Haque, David L Mobley, Daniel S Lambrecht, Robert a Distasio, Martin Head-gordon, Gary N I Clark, Margaret E Johnson, and Teresa Head-gordon. Current Status of the AMOEBA Polarizable Force Field. *J. Phys. Chem. B*, 114(8):2549–2564, 2010.

[31] Yue Shi, Zhen Xia, Jiajing Zhang, Robert Best, Chuanjie Wu, Jay W. Ponder, and Pengyu Ren. Polarizable atomic multipole-based AMOEBA force field for proteins. *J. Chem. Theory Comput.*, 9(9):4046–4063, 2013.

[32] Peter Eastman, Mark S. Friedrichs, John D. Chodera, Randall J. Radmer, Christopher M. Bruns, Joy P. Ku, Kyle A. Beauchamp, Thomas J. Lane, Lee-Ping Wang, Diwakar Shukla, Tony Tye, Mike Houston, Timo Stich, Christoph

Klein, Michael R. Shirts, and Vijay S. Pande. OpenMM 4: A Reusable, Extensible, Hardware Independent Library for High Performance Molecular Simulation. *J. Chem. Theory Comput.*, 9(1):461–469, January 2013.

[33] A. C T Van Duin, Siddharth Dasgupta, Francois Lorant, and William A. Goddard. ReaxFF: A reactive force field for hydrocarbons. *J. Phys. Chem. A*, 105(41):9396–9409, 2001.

[34] Kimberly Chenoweth, Adri C. T. van Duin, and William A. Goddard III. ReaxFF reactive force field for molecular dynamics simulations of hydrocarbon oxidation. *J. Phys. Chem. A*, 112(5):1040–1053, 2008.

[35] Sander Pronk, Szilárd Páll, Roland Schulz, Per Larsson, Pär Bjelkmar, Rossen Apostolov, Michael R. Shirts, Jeremy C Smith, Peter M Kasson, David van der Spoel, Berk Hess, and Erik Lindahl. GROMACS 4.5: a high-throughput and highly parallel open source molecular simulation toolkit. *Bioinformatics*, 29(7):845–54, April 2013.

[36] Szilárd Páll, Mark James Abraham, Carsten Kutzner, Berk Hess, and Erik Lindahl. *Tackling Exascale Software Challenges in Molecular Dynamics Simulations with GROMACS*, pages 3–27. Springer International Publishing, Cham, 2015.

[37] Mark James Abraham, Teemu Murtola, Roland Schulz, Szilárd Páll, Jeremy C. Smith, Berk Hess, and Erik Lindah. Gromacs: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX*, 1-2:19–25, 2015.

[38] GROMACS. `http://www.gromacs.org/` (accessed Dec 02, 2013).

[39] OpenMM. `http://simtk.org/home/openmm` (accessed Jul 8, 2012), available through SimTK.

[40] Joshua A. Anderson, Chris D. Lorenz, and A. Travesset. General purpose molecular dynamics simulations fully implemented on graphics processing units. *J. Comput. Phys.*, 227(10):5342–5359, 2008.

[41] Jens Glaser, Trung Dac Nguyen, Joshua A. Anderson, Pak Lui, Filippo Spiga, Jaime A. Millan, David C. Morse, and Sharon C. Glotzer. Strong scaling of general-purpose molecular dynamics simulations on GPUs. *Comput. Phys. Commun.*, 192:97–107, 2015.

[42] HOOMD-blue. `http://codeblue.umich.edu/hoomd-blue` (accessed May 14, 2016).

[43] TINKER Molecular Modeling `http://dasher.wustl.edu/tinker/` (accessed May 5, 2016).

[44] Lingle Wang, Yujie Wu, Yuqing Deng, Byungchan Kim, Levi Pierce, Goran Krilov, Dmitry Lupyan, Shaughnessy Robinson, Markus K Dahlgren, Jeremy Greenwood, Donna L Romero, Craig Masse, Jennifer L Knight, Thomas Steinbrecher, Thijs Beuming, Wolfgang Damm, Ed Harder, Woody Sherman, Mark Brewer, Ron Wester, Mark Murcko, Leah Frye, Ramy Farid, Teng Lin, David L Mobley, William L Jorgensen, Bruce J Berne, Richard A Friesner, and Robert Abel. Accurate and Reliable Prediction of Relative Ligand Binding Potency in Prospective Drug Discovery by Way of a Modern Free-Energy Calculation Protocol and Force Field. *J. Am. Chem. Soc.*, feb 2015.

[45] Hanna Geppert, Martin Vogt, and Jürgen Bajorath. Current trends in ligand-based virtual screening: molecular representations, data mining methods, new application areas, and performance evaluation. *J. Chem. Inf. Model.*, 50(2):205–16, February 2010.

[46] Lu Wencong. Data Mining and Discovery of Chemical Knowledge. In Mohamed Medhat Gaber, editor, *Sci. Data Min. Knowl. Discov. SE - 11*, pages 269–317. Springer Berlin Heidelberg, 2010.

[47] Hassan Safouhi and Ahmed Bouferguene. Computational Chemistry. In Mohamed Medhat Gaber, editor, *Sci. Data Min. Knowl. Discov. SE - 8*, pages 173–206. Springer Berlin Heidelberg, 2010.

[48] Anubhav Jain, Shyue Ping Ong, Geoffroy Hautier, Wei Chen, William Davidson Richards, Stephen Dacek, Shreyas Cholia, Dan Gunter, David Skinner, Gerbrand Ceder, and Kristin a. Persson. Commentary: The Materials Project: A materials genome approach to accelerating materials innovation. *APL Mater.*, 1(1):011002, 2013.

[49] Chris Oostenbrink and Wilfred F. van Gunsteren. Free energies of ligand binding for structurally diverse compounds. *Proc. Natl. Acad. Sci.*, 102(19):6750–6754, 2005.

[50] Chris Oostenbrink and Wilfred F. van Gunsteren. Efficient calculation of many stacking and pairing free energies in DNA from a few molecular dynamics simulations. *Chemistry*, 11(15):4340–8, July 2005.

[51] Jennifer L. Knight and Charles L. Brooks. Multisite λ Dynamics for Simulated StructureActivity Relationship Studies. *J. Chem. Theory Comput.*, 7:2728–2739, 2011.

[52] Kira a. Armacost, Garrett B. Goh, and Charles L. Brooks. Biasing Potential Replica Exchange Multisite λ-Dynamics for Efficient Free Energy Calculations. *J. Chem. Theory Comput.*, 11:1267–1277, February 2015.

[53] Andreas Kukol, editor. *Molecular Modeling of Proteins*. Springer, New York, 2015.

[54] L. N. Naden, T. T. Pham, and M. R. Shirts. Linear Basis Function Approach to Efficient Alchemical Free Energy Calculations. 1. Removal of Uncharged Atomic Sites. *J. Chem. Theory Comput.*, 10:1128–1149, 2014.

[55] Levi N. Naden and Michael R. Shirts. Linear basis function approach to efficient alchemical free energy calculations. 2. inserting and deleting particles with coulombic interactions. *J. Chem. Theory Comput.*, 11:2536–2549, 2015.

[56] Levi N Naden and Michael R Shirts. Rapid Computation of Thermodynamic Properties over Multidimensional Nonbonded Parameter Spaces Using Adaptive Multistate Reweighting. *J. Chem. Theory Comput.*, 12:1806–1823, 2016.

[57] Michael R. Shirts, Jed W. Pitera, William C. Swope, and Vijay S. Pande. Extremely precise free energy calculations of amino acid side chain analogs: Comparison of common molecular mechanics force fields for proteins. *J. Chem. Phys.*, 119(11):5740, 2003.

[58] Michael R. Shirts and Vijay S. Pande. Solvation free energies of amino acid side chain analogs for common molecular mechanics water models. *J. Chem. Phys.*, 122(13):134508, April 2005.

[59] Kai Kadau, John L Barber, Timothy C Germann, Brad L Holian, and Berni J Alder. Atomistic methods in fluid simulation. *Philos. Trans. A. Math. Phys. Eng. Sci.*, 368(1916):1547–60, April 2010.

[60] Himanshu Paliwal and Michael R. Shirts. A benchmark test set for alchemical free energy transformations and its use to quantify error in common free energy methods. *J. Chem. Theory Comput.*, pages 4115–4134, 2011.

[61] Caroline Desgranges and Jerome Delhommelle. Evaluation of the grand-canonical partition function using expanded Wang-Landau simulations. I. Thermodynamic properties in the bulk and at the liquid-vapor phase boundary. *J. Chem. Phys.*, 136(18):184107, 2012.

[62] Yong Zhang and Edward J Maginn. A comparison of methods for melting point calculation using molecular dynamics simulations. *J. Chem. Phys.*, 136(14):144116, April 2012.

[63] Hari S Muddana, Andrew T Fenley, David L Mobley, and Michael K Gilson. The SAMPL4 host-guest blind prediction challenge: an overview. *J. Comput. Aided. Mol. Des.*, 28(4):305–17, April 2014.

[64] Che-lun Hung and Chi-chun Chen. Computational Approaches for Drug Discovery. *Drug Dev. Res.*, 75:412–418, 2014.

[65] Denis S Abrams and John M Prausnitz. Statistical thermodynamics of liquid mixtures: A new expression for the excess Gibbs energy of partly or completely miscible systems. *AIChE J.*, 21(1):116–128, 1975.

[66] Aage Fredenslund, Russell L. Jones, and John M. Prausnitz. Group-contribution estimation of activity coefficients in nonideal liquid mixtures. *AIChE J.*, 21(6):1086–1099, 1975.

[67] Abolghasem Jouyban. *Handbook of Solubility Data for Pharmaceuticals*. CRC Press, Boca Raton, 2010.

[68] Philip J Hajduk and Jonathan Greer. A decade of fragment-based drug design: strategic advances and lessons learned. *Nat. Rev. Drug Discov.*, 6(3):211–9, March 2007.

[69] Paola Gramatica. Principles of QSAR models validation: internal and external. *QSAR Comb. Sci.*, 26(5):694–701, May 2007.

[70] Alexander Tropsha. Best Practices for QSAR Model Development, Validation, and Exploitation. *Mol. Inform.*, 29(6-7):476–488, July 2010.

[71] William L. Jorgensen, David S. Maxwell, and Julian Tirado-Rives. Development and testing of the OPLS all-atom force field on conformational energetics and properties of organic liquids. *J. Am. Chem. Soc.*, 7863(15):11225–11236, 1996.

[72] George A Kaminski, Richard A Friesner, Julian Tirado-rives, and William L Jorgensen. Evaluation and Reparametrization of the OPLS-AA Force Field for Proteins via Comparison with Accurate Quantum Chemical Calculations on Peptides. *J. Phys. Chem. B*, 2(105):6474–6487, 2001.

[73] Junmei Wang, Romain M. Wolf, James W. Caldwell, Peter A. Kollman, and David A. Case. Development and testing of a general amber force field. *J. Comput. Chem.*, 25(9):1157–74, July 2004.

[74] In Suk Joung and Thomas E. Cheatham. Determination of alkali and halide monovalent ion parameters for use in explicitly solvated biomolecular simulations. *J. Phys. Chem. B*, 112(30):9020–41, July 2008.

[75] Chris Oostenbrink, Alessandra Villa, Alan E Mark, and Wilfred F van Gunsteren. A biomolecular force field based on the free enthalpy of hydration and solvation: the GROMOS force-field parameter sets 53A5 and 53A6. *J. Comput. Chem.*, 25(13):1656–76, October 2004.

[76] K. Vanommeslaeghe, E. Hatcher, C. Acharya, S. Kundu, S. Zhong, J. Shim, E. Darian, O. Guvench, P. Lopes, I. Vorobyov, and A. D. MacKerell Jr. CHARMM General Force Field (CGenFF): A force field for drug-like molecules compatible with the CHARMM all-atom additive biological force fields. *J. Comput. Chem.*, 31(4):671–690, 2011.

[77] Andrew S. Paluch and Edward J. Maginn. Predicting the Solubility of Solid Phenanthrene: A Combined Molecular Simulation and Group Contribution Approach. *AIChE J.*, 59(7):2647–2661, July 2013.

[78] Andrew S. Paluch and Edward J. Maginn. Efficient Estimation of the Equilibrium Solution-Phase Fugacity of Soluble Nonelectrolyte Solids in Binary Solvents by Molecular Simulation. *Ind. Eng. Chem. Res.*, 52:13743–13760, 2013.

[79] Andrew J. Schultz and David A. Kofke. Quantifying Computational Effort Required for Stochastic Averages. *J. Chem. Theory Comput.*, 10:5229–5234, November 2014.

[80] Thomas J Lane, Diwakar Shukla, Kyle a Beauchamp, and Vijay S Pande. To milliseconds and beyond: challenges in the simulation of protein folding. *Curr. Opin. Struct. Biol.*, 23(1):58–65, February 2013.

[81] Daan Frenkel and Berend Smit. *Understanding Molecular Simulation*. Academic Press, San Diego, 2nd edition, 2002.

[82] TC Beutler, AE Mark, and RC van Schaik. Avoiding singularities and numerical instabilities in free energy calculations based on molecular simulations. *Chem. Phys. Lett.*, 222(June):529–539, 1994.

[83] M. Zacharias, T. P. Straatsma, and J. A. McCammon. Separation-shifted scaling, a new scaling method for Lennard-Jones interactions in thermodynamic integration. *J. Chem. Phys.*, 100(12):9025, 1994.

[84] Tri T. Pham and Michael R. Shirts. Identifying low variance pathways for free energy calculations of molecular transformations in solution phase. *J. Chem. Phys.*, 135(3):034114, 2011.

[85] Thomas Steinbrecher, David L. Mobley, and David A. Case. Nonlinear scaling schemes for Lennard-Jones interactions in free energy calculations. *J. Chem. Phys.*, 127(21):214108, December 2007.

[86] Jozef Hritz and Chris Oostenbrink. Hamiltonian replica exchange molecular dynamics using soft-core interactions. *J. Chem. Phys.*, 128(14):144121, April 2008.

[87] Sereina Riniker, Clara D Christ, Halvor S Hansen, Philippe H Hünenberger, Chris Oostenbrink, Denise Steiner, and Wilfred F. van Gunsteren. Calculation of relative free energies for ligand-protein binding, solvation, and conformational transitions using the GROMOS software. *J. Phys. Chem. B*, 115(46):13570–7, November 2011.

[88] Floris P Buelens and Helmut Grubmüller. Linear-scaling soft-core scheme for alchemical free energy calculations. *J. Comput. Chem.*, pages 25–33, September 2011.

[89] David L. Mobley, Christopher I. Bayly, Matthew D. Cooper, Michael R. Shirts, and Ken A. Dill. Small molecule hydration free energies in explicit

solvent: An extensive test of fixed-charge atomistic simulations. *J. Chem. Theory Comput.*, 5(2):350–358, February 2009.

[90] Jed W. Pitera and Wilfred F. van Gunsteren. A comparison of non-bonded scaling approaches for free energy calculations. *Mol. Simul.*, 28(1-2):45–65, 2002.

[91] Sereina Riniker, Clara D. Christ, Niels Hansen, Alan E. Mark, Pramod C. Nair, and Wilfred F. van Gunsteren. Comparison of enveloping distribution sampling and thermodynamic integration to calculate binding free energies of phenylethanolamine N-methyltransferase inhibitors. *J. Chem. Phys.*, 135(2):024105, July 2011.

[92] Thomas Steinbrecher, InSuk Joung, and David A. Case. Soft-core potentials in thermodynamic integration: comparing one- and two-step transformations. *J. Comput. Chem.*, 32(15):3253–63, November 2011.

[93] Andrew Gelman and Xiao-Li Meng. Simulating normalizing constants: from importance sampling to bridge sampling to path sampling. *Stat. Sci.*, 13(2):163–185, May 1998.

[94] Daniel Shenfeld, Huafeng Xu, Michael Eastwood, Ron Dror, and David Shaw. Minimizing thermodynamic length to select intermediate states for free-energy calculations and replica-exchange simulations. *Phys. Rev. E*, 80(4):1–4, October 2009.

[95] Arnaud Blondel. Ensemble variance in free energy calculations by thermodynamic integration: theory, optimal "Alchemical" path, and practical solutions. *J. Comput. Chem.*, 25(7):985–993, May 2004.

[96] Gavin E Crooks. Measuring Thermodynamic Length. *Phys. Rev. Lett.*, 99(10):10–13, September 2007.

[97] Tri T. Pham and Michael R. Shirts. Optimal pairwise and non-pairwise alchemical pathways for free energy calculations of molecular transformation in solution phase. *J. Chem. Phys.*, 136(12):124120, March 2012.

[98] Robert W Zwanzig. High-Temperature Equation of State by a Perturbation Method. I. Nonpolar Gases. *J. Chem. Phys.*, 22(8):1420, 1954.

[99] Shankar Kumar, John M Rosenberg, Djamal Bouzida, Robert H Swendsen, and Peter A Kollman. THE weighted histogram analysis method for free-energy calculations on biomolecules. I. The method. *J. Comput. Chem.*, 13(8):1011–1021, 1992.

[100] Shankar Kumar, John M Rosenberg, Djamal Bouzida, Robert H Swendsen, and Peter A Kollman. Multidimensional free-energy calculations using the weighted histogram analysis method. *J. Comput. Chem.*, 16(11):1339–1350, 1995.

[101] Michael R. Shirts and John D. Chodera. Statistically optimal analysis of samples from multiple equilibrium states. *J. Chem. Phys.*, 129(12):124105, September 2008.

[102] Anita de Ruiter and Chris Oostenbrink. Efficient and Accurate Free Energy Calculations on Trypsin Inhibitors. *J. Chem. Theory Comput.*, 8(10):3686–3695, October 2012.

[103] Jed W. Pitera and Wilfred F. van Gunsteren. One-Step Perturbation Methods for Solvation Free Energies of Polar Solutes. *J. Phys. Chem. B*, 105(45):11264–11274, November 2001.

[104] Clara D. Christ and Wilfred F. van Gunsteren. Enveloping distribution sampling: a method to calculate free energy differences from a single simulation. *J. Chem. Phys.*, 126(18):184110, May 2007.

[105] Charles H Bennett. Efficent Estimation of Free nergy Differences from Monte Carlo Data. *J. Comput. Phys.*, 268:245–268, 1976.

[106] Peter. Kollman. Free energy calculations: Applications to chemical and biochemical phenomena. *Chem. Rev.*, 93(7):2395–2417, November 1993.

[107] Andrew S. Paluch, Dan D. Cryan, and Edward J. Maginn. Predicting the Solubility of the Sparingly Soluble Solids 1,2,4,5-Tetramethylbenzene, Phenanthrene, and Fluorene in Various Organic Solvents by Molecular Simulation. *J. Chem. Eng. Data*, 56(4):1587–1595, April 2011.

[108] D Bemporad, C Luttmann, and J W Essex. Computer simulation of small molecule permeation across a lipid bilayer: dependence on bilayer properties and solute volume, size, and cross-sectional area. *Biophys. J.*, 87(1):1–13, July 2004.

[109] Yuqing Deng and Benot Roux. Hydration of amino acid side chains: Nonpolar and electrostatic contributions calculated from staged molecular dynamics free energy simulations with explicit water molecules. *J. Phys. Chem. B*, 17:16567–16576, 2004.

[110] Berk Hess, Carsten Kutzner, David van der Spoel, and Erik Lindahl. GROMACS 4 : Algorithms for Highly Efficient , Load-Balanced , and Scalable Molecular Simulation. *J. Chem. Theory Comput.*, 4(3):435–447, 2008.

[111] B R Brooks, C L Brooks Iii, A D Mackerell, L Nilsson, R J Petrella, B Roux, Y Won, G Archontis, C Bartels, S Boresch, A Caflisch, L Caves, Q Cui, A R Dinner, and M Feig. CHARMM : The Biomolecular Simulation Program. *J. Comput. Chem.*, 30:1545–1614, 2009.

[112] David A. Case, T. A. Darden, Thomas E. Cheatham, C. L. Simmerling, J. Wang, R. E. Duke, R. Luo, R. C. Walker, W. Zhang, K. M. Merz,

B. Roberts, S. Hayik, A. Roitberg, G. Seabra, J. Swails, A. W. Goetz, I. Kolossváry, K.F. Wong, F. Paesani, J. Vanicek, R. M. Wolf, J. Liu, X. Wu, S.R. Brozell, Thomas Steinbrecher, H. Gohlke, Q. Cai, X. Ye, M.-J. Hsieh, G. Cui, D. R. Roe, D. H. Mathews, M. G. Seetin, R. Salomon-Ferrer, C. Sagui, V. Babin, T. Luchko, S. Gusarov, A. Kovalenko, and Peter A. Kollman. *AMBER 12*. University of California, San Francisco, 2012.

[113] Jiyao Wang, Yuqing Deng, and Benoît Roux. Absolute binding free energy calculations using molecular dynamics simulations with restraining potentials. *Biophys. J*, 91(8):2798–814, October 2006.

[114] David L. Mobley, Elise Dumont, John D. Chodera, and Ken A. Dill. Comparison of charge models for fixed-charge force fields: small-molecule hydration free energies in explicit solvent. *J. Phys. Chem. B*, 111(9):2242–54, March 2007.

[115] John D. Weeks, David Chandler, and Hans C. Andersen. Role of Repulsive Forces in Determining the Equilibrium Structure of Simple Liquids. *J. Chem. Phys.*, 54(12):5237, 1971.

[116] Kai Wang, John D. Chodera, Yanzhi Yang, and Michael R. Shirts. Identifying ligand binding sites and poses using GPU-accelerated Hamiltonian replica exchange molecular dynamics. *J. Comput. Aid. Mol. Des.*, 27(12):989–1007, 2013.

[117] YANK. `http://simtk.org/home/yank/` (accessed Jul 8, 2012), available through SimTK.

[118] Peter Eastman and Vijay Pande. OpenMM: A Hardware-Independent Framework for Molecular Simulations. *Comput. Sci. Eng.*, 12(4):34–39, July 2010.

[119] Peter Eastman and Vijay S Pande. Efficient Nonbonded Interactions for Molecular Dynamics on a Graphics Processing Unit. *J. Comput. Chem.*, 31:1268–1272, 2009.

[120] Hiroaki Fukunishi, Osamu Watanabe, and Shoji Takada. On the Hamiltonian replica exchange method for efficient sampling of biomolecular systems: Application to protein structure prediction. *J. Chem. Phys.*, 116(20):9058, 2002.

[121] John D. Chodera and Michael R. Shirts. Replica exchange and expanded ensemble simulations as Gibbs sampling: simple improvements for enhanced mixing. *J. Chem. Phys.*, 135(19):194110, November 2011.

[122] Peter Eastman and VS Pande. Constant constraint matrix approximation: a robust, parallelizable constraint method for molecular simulations. *J. Chem. Theory Comput.*, (2):434–437, 2010.

[123] Bradley Efron and Robert Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall/CRC, Boca Raton, FL, 1993.

[124] Jean-Paul Ryckaert, Giovanni Ciccotti, and Herman J.C. Berendsen. Numerical integration of the Cartesian Equations of Motion of a System with Constraints: Molecular Dynamics of n-Alkanes. *J. Comput. Phys.*, 341, 1977.

[125] Shuichi Miyamoto and Peter A. Kollman. Settle: An analytical version of the SHAKE and RATTLE algorithm for rigid water models. *J. Comput. Chem.*, 13(8):952–962, 1992.

[126] Kim-Hung Chow and David M. Ferguson. Isothermal-isobaric molecular dynamics simulations with Monte Carlo volume sampling. *Comput. Phys. Commun.*, 91(1-3):283–289, September 1995.

[127] Johan Å qvist, Petra Wennerström, Martin Nervall, Sinisa Bjelic, and Bjørn O. Brandsdal. Molecular dynamics simulations of water and biomolecules with a Monte Carlo constant pressure algorithm. *Chem. Phys. Lett.*, 384(4-6):288–294, January 2004.

[128] Michael R. Shirts, David L. Mobley, John D. Chodera, and Vijay S. Pande. Accurate and efficient corrections for missing dispersion interactions in molecular simulations. *J. Phys. Chem. B*, 111(45):13052–63, November 2007.

[129] Powell M.J.D. *Advances in Optimization and Numerical Analysis*. Mathematics and Its Applications. Springer, 1994.

[130] SciPy: Open Source Scientific Tools for Python. `http://www.scipy.org/` (accessed May 3, 2014).

[131] FN Fritsch and R.E. Carlson. Monotone piecewise cubic interpolation. *SIAM J. Numer. Anal.*, 17(2):238–246, 1980.

[132] FN Fritsch and J. Butland. A method for constructing local monotone piecewise cubic interpolants. *SIAM J. Sci. Stat. Comput.*, 5(2):300–304, 1984.

[133] Michael R. Shirts and David L. Mobley. An introduction to best practices in free energy calculations. In Luca Monticelli and Emppu Salonen, editors, *Biomolecular Simulations*, volume 924 of *Methods in Molecular Biology*, pages 271–311. Humana Press, 2013.

[134] Michael R. Shirts, David L. Mobley, and John D. Chodera. Chapter 4 Alchemical Free Energy Calculations: Ready for Prime Time? *Annu. Rep. Comput. Chem.*, 3(07):41–59, 2007.

[135] John D. Chodera, David L. Mobley, Michael R. Shirts, Richard W. Dixon, Kim Branson, and Vijay S. Pande. Alchemical free energy methods for drug discovery: progress and challenges. *Curr. Opin. Struc. Bio.*, 21(2):150–60, April 2011.

[136] MichaelR. Shirts. Best practices in free energy calculations for drug design. In Riccardo Baron, editor, *Computational Drug Discovery and Design*, volume 819 of *Methods in Molecular Biology*, pages 425–467. Springer New York, 2012.

[137] Haiyan Liu, AE Mark, and Wilfred F. van Gunsteren. Estimating the relative free energy of different molecular states with respect to a single reference state. *J. Phys. Chem.*, 3654(96):9485–9494, 1996.

[138] Chris Oostenbrink. Efficient free energy calculations on small molecule host-guest systemsA combined linear interaction energy/one-step perturbation approach. *J. Comput. Chem.*, 30(2):212–221, 2009.

[139] Ulrich Essmann, Lalith Perera, Max L. Berkowitz, Tom Darden, Hsing Lee, and Lee G. Pedersen. A smooth particle mesh Ewald method. *J. Chem. Phys.*, 103(19):8577, 1995.

[140] M. Parrinello and A. Rahman. Polymorphic transitions in single crystals: A new molecular dynamics method. *J. Appl. Phys.*, 52(12):7182, 1981.

[141] Shuichi Nosé and M L Klein. Constant pressure molecular dynamics for molecular systems. *Mol. Phys.*, 50(5):1055–1076, 1983.

[142] Michael R. Shirts. Simple Quantitative Tests to Validate Sampling from Thermodynamic Ensembles. *J. Chem. Theory Comput.*, 9:909–926, 2013.

[143] John D. Chodera, William C. Swope, Jed W. Pitera, Chaok Seok, and Ken A. Dill. Use of the Weighted Histogram Analysis Method for the Analysis of Simulated and Parallel Tempering Simulations. *J. Chem. Theory Comput.*, 3(1):26–41, January 2007.

[144] Analysis code and example implementation of basis functions for OpenMM and YANK can be found in the basisanalyze repository on GitHub at `https://github.com/shirtsgroup/basisanalyze` with commit hash d2d18e440c.

[145] Nicolas Lartillot and Hervé Philippe. Computing Bayes factors using thermodynamic integration. *Syst. Biol.*, 55(2):195–207, April 2006.

[146] Joseph E. Basconi and Michael R. Shirts. Effects of Temperature Control Algorithms on Transport Properties and Kinetics in Molecular Dynamics Simulations. *J. Chem. Theory Comput.*, 9(7):2887–2899, July 2013.

[147] C. C. Wang, G. Pilania, S. A. Boggs, S. Kumar, C. Breneman, and R. Ramprasad. Computational strategies for polymer dielectrics design. *Polymer*, 55(4):979–988, 2014.

[148] M. Harini, Jhumpa Adhikari, and K. Yamuna Rani. A review on property estimation methods and computational schemes for rational solvent design: A focus on pharmaceuticals. *Ind. Eng. Chem. Res.*, 52:6869–6893, 2013.

[149] Qingyuan Yang, Dahuan Liu, Chongli Zhong, and Jian-rong Li. Development of Computational Methodologies for Metal Organic Frameworks and Their Application in Gas Separations. *Chem. Rev.*, 113(10):8261–8323, 2013.

[150] Luca Monticelli and D.Peter Tieleman. Force Fields for Classical Molecular Dynamics. In Luca Monticelli and Emppu Salonen, editors, *Biomol. Simulations SE - 8*, volume 924 of *Methods in Molecular Biology*, pages 197–213. Humana Press, New York, NY, 2013.

[151] D Yin and a D MacKerell Jr. Combined Ab Initio / Empirical Approach for Optimization of Lennard Jones Parameters. *J. Comp. Chem.*, 19(3):334–348, 1997.

[152] I Jen Chen, Daxu Yin, and Alexander D. MacKerell. Combinedab initio/empirical approach for optimization of Lennard-Jones parameters for polar-neutral compounds. *J. Comput. Chem.*, 23(2):199–213, 2002.

[153] Bernhard Eckl, Jadran Vrabec, and Hans Hasse. Set of Molecular Models Based on Quantum Mechanical Ab Initio Calculations and Thermodynamic Data. *J. Phys. Chem. B*, 112:12710–12721, 2008.

[154] A D Mackerell, D Bashford, M Bellott, R L Dunbrack, J D Evanseck, M J Field, S Fischer, J Gao, H Guo, S Ha, L Kuchnir, K Kuczera, F T K Lau, C Mattos, S Michnick, T Ngo, D T Nguyen, B Prodhom, W E Reiher, B Roux, M Schlenkrich, J C Smith, R Stote, J Straub, M Watanabe, J Wio, D Yin, and M Karplus. All-Atom Empirical Potential for Molecular Modeling and Dynamics Studies of Proteins. *J. Phys. Chem. B*, 5647(97):3586–3616, 1998.

[155] David A. Pearlman, David A. Case, James W. Caldwell, Wilson S. Ross, Thomas E. Cheatham, Steve DeBolt, David Ferguson, George Seibel, and Peter Kollman. AMBER, a package of computer programs for applying molecular mechanics, normal mode analysis, molecular dynamics and free energy calculations to simulate the structural and energetic properties of molecules. *Comput. Phys. Commun.*, 91(1-3):1–41, September 1995.

[156] Wilfred F. van Gunsteren and Alan E. Mark. Validation of molecular dynamics simulation. *J. Chem. Phys.*, 108(15):6109, 1998.

[157] Wilfred F van Gunsteren, Dirk Bakowies, Riccardo Baron, Indira Chandrasekhar, Markus Christen, Xavier Daura, Peter Gee, Daan P Geerke, Alice Glättli, Philippe H Hünenberger, Mika A Kastenholz, Chris Oostenbrink, Merijn Schenk, Daniel Trzesniak, Nico F A van der Vegt, and Haibo B Yu. Biomolecular modeling: Goals, problems, perspectives. *Angew. Chem. Int. Ed. Engl.*, 45(25):4064–92, June 2006.

[158] John G. Kirkwood. Statistical Mechanics of Fluid Mixtures. *J. Chem. Phys.*, 3(5):300, 1935.

[159] F. Weinhold. Metric geometry of equilibrium thermodynamics. *J. Chem. Phys.*, 63(6):2479, 1975.

[160] Stefan Boresch and Franz Tettinger. Absolute binding free energies: a quantitative approach for their calculation. *J. Phys. Chem. B*, 107:9535–9551, 2003.

[161] Di Wu and David A Kofke. Phase-space overlap measures. II. Design and implementation of staging methods for free-energy calculations. *J. Chem. Phys.*, 123(8):084109, August 2005.

[162] Himanshu Paliwal and Michael R. Shirts. Using multistate reweighting to rapidly and efficiently explore molecular simulation parameters space for nonbonded interactions. *J. Chem. Theory Comput.*, 9(11):4700–4717, 2013.

[163] Michael R. Shirts and Vijay S. Pande. Comparison of efficiency and bias of free energies computed by exponential averaging, the Bennett acceptance ratio, and thermodynamic integration. *J. Chem. Phys.*, 122(14):144107, April 2005.

[164] DM Huang, PL Geissler, and David Chandler. Scaling of hydrophobic solvation free energies. *J. Phys. Chem. B*, 105:6704–6709, 2001.

[165] F V Grigoriev, M V Basilevsky, S N Gabin, A N Romanov, and V B Sulimov. Cavitation free energy for organic molecules having various sizes and shapes. *J. Phys. Chem. B*, 111(49):13748–55, December 2007.

[166] Analysis code for the ion parameter project can be found in the ion-parameter repository on GitHub at `https://github.com/shirtsgroup/ion-parameters` with commit hash 7ff188b0bc or later. Accessed June 3, 2015.

[167] Robert A. Kuharski and Peter J. Rossky. Solvation of hydrophobic species in aqueous urea solution: a molecular dynamics study. *J. Am. Chem. Soc.*, 106(20):5794–5800, October 1984.

[168] For explicit numbers, please see the tabulated data in the supplementary information.

[169] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In Evangelos Simoudis, Jiawei Han, and Usama Fayyad, editors, *Proc. 2nd Int. Conf. Knowl. Discov. Data Min.*, pages 226–231, Menlo Park, California, 1996. AAAI Press.

[170] Joseph B Kruskal. On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem. *Proc. Am. Math. Soc.*, 7:48–50, 1956.

[171] R. Duda and P. Hart. *Pattern Classif. Scene Anal.* John Wiley and Sons, New York, NY, 1st edition, 1973.

[172] Andrew T Fenley, Hari S Muddana, and Michael K Gilson. Entropy-enthalpy transduction caused by conformational shifts can obscure the forces driving protein-ligand binding. *Proc. Natl. Acad. Sci. U. S. A.*, 109(49):20006–20011, December 2012.

[173] Andrew T Fenley, Benjamin J Killian, Vladimir Hnizdo, Adam Fedorowicz, Dan S Sharp, and Michael K Gilson. Correlation as a Determinant of Configurational Entropy in Supramolecular and Protein Systems. *J. Phys. Chem. B*, 118:6447–6455, 2015.

[174] Alan Grossfield, Pengyu Ren, and Jay W. Ponder. Ion Solvation Thermodynamics from Simulation with a Polarizable Force Field. *J. Am. Chem. Soc.*, 125(50):15671–15682, 2003.

[175] Kasper P. Jensen and William L. Jorgensen. Halide, ammonium, and alkali metal ion parameters for modeling aqueous solutions. *J. Chem. Theory Comput.*, 2(6):1499–1509, 2006.

[176] Johan Aqvist. Ion-Water Interaction Potentials Derived from Free Energy Perturbation Slmulations. *J. Phys. Chem.*, 94:8021–8024, 1990.

[177] Dmitrii Beglov, Benoit Roux, and Canada Hc. Finite representation of an infinite for computer simulations bulk system : Solvent boundary potential. *J. Med. Phys.*, 100(June):9050–9063, 1994.

[178] David E. Smith and Liem X. Dang. Interionic potentials of mean force for SrCl2 in polarizable water. *Chem. Phys. Lett.*, 230(1-2):209–214, 1994.

[179] Liem X. Dang. Development of nonadditive intermolecular potentials using molecular dynamics: Solvation of Li+ and F ions in polarizable water. *J. Chem. Phys.*, 96(9):6970, 1992.

[180] Liem X Dang. Free energies for association of Cs + to l & crown-6 in water . A molecular dynamics study including counter ions. *Chem. Phys. Lett.*, (September):2–5, 1994.

[181] Liem X Dang. Mechanism and Thermodynamics of Ion Selectivity in Aqueous Solutions of 18-Crown-6 Ether : A Molecular Dynamics Study. *J. Am. Chem. Soc.*, 117(11):6954–6960, 1995.

[182] Liem X. Dang and Bruce C. Garrett. Photoelectron spectra of the hydrated iodine anion from molecular dynamics simulations. *J. Chem. Phys.*, 99(4):2972, 1993.

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

[184] P Hünenberger and M Reif. *Single-Ion Solvation: Experimental and Theoretical Approaches to Elusive Thermodynamic Quantities*. RSC theoretical and computational chemistry series. Royal Society of Chemistry, 2011.

[185] Andrew Pohorille, Christopher Jarzynski, and Christophe Chipot. Good practices in free-energy calculations. *J. Phys. Chem. B*, 114(32):10235–53, August 2010.

[186] Zhiqiang Tan. On a Likelihood Approach for Monte Carlo Integration. *J. Am. Stat. Assoc.*, 99(468):1027–1036, December 2004.

[187] Pavel V. Klimovich, Michael R. Shirts, and David L. Mobley. Guidelines for the analysis of free energy calculations. *J. Comput.-Aided Mol. Des.*, 29(5):397–411, 2015.

[188] Cassiano G. Aimoli, Edward J. Maginn, and Charlles R.A. Abreu. Force field comparison and thermodynamic property calculation of supercritical CO2 and CH4 using molecular dynamics simulations. *Fluid Phase Equilib.*, 368:80–90, April 2014.

[189] Carlos Avendaño, Thomas Lafitte, Amparo Galindo, Claire S. Adjiman, George Jackson, and Erich A. Müller. SAFT-γ force field for the simulation of molecular fluids. 1. A single-site coarse grained model of carbon dioxide. *J. Phys. Chem. B*, 115:11154–11169, 2011.

[190] Frank H Allen, Olga Kennard, David G Watson, Lee Brammer, and A Guy Orpen. Tables of Bond Lengths determined by X-Ray and Neutron Diffraction. Part 1. Bond Lenghts in Organic Compounds. *J. Chem. Soc. Perkin Trans. II*, (2):1695–1914, 1987.

[191] Brian L Bray. Large-scale manufacture of peptide therapeutics by chemical synthesis. *Nat. Rev. Drug Discov.*, 2(7):587–593, 2003.

[192] Tuomo Kalliokoski. Price-Focused Analysis of Commercially Available Building Blocks for Combinatorial Library Synthesis. *ACS Comb. Sci.*, 17(10):600–607, 2015.

[193] Dieter W. Heermann, Peter Nielaba, and Mauro Rovere. Hybrid molecular dynamics. *Comput. Phys. Commun.*, 60(3):311–318, 1990.

[194] B. Mehlig, D. Heermann, and B. Forrest. Hybrid Monte Carlo method for condensed-matter systems. *Phys. Rev. B*, 45(2):679–685, 1992.

[195] Stefan Boresch and Martin Karplus. The Jacobian factor in free energy simulations. *J. Chem. Phys.*, 105(12):5145, 1996.

[196] Araz Jakalian, David B. Jack, and Christopher I. Bayly. Fast, efficient generation of high-quality atomic charges. AM1-BCC model: II. Parameterization and validation. *J. Comput. Chem.*, 23(16):1623–1641, 2002.

[197] G. M. Torrie and J. P. Valleau. Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling. *J. Comput. Phys.*, 23(2):187–199, 1977.

[198] E Marinari and G Parisi. Simulated Tempering: A New Monte Carlo Scheme. *Eur. Lett. Eur. Lett*, 19(196):451–458, 1992.

[199] A. P. Lyubartsev, A. A. Martsinovski, S. V. Shevkunov, and P. N. Vorontsov-Velyaminov. New approach to Monte Carlo calculation of the free energy: Method of expanded ensembles. *J. Chem. Phys.*, 96(3):1776, 1992.

[200] Caroline Desgranges and Jerome Delhommelle. Evaluation of the grand-canonical partition function using expanded Wang-Landau simulations. II. Adsorption of atomic and molecular fluids in a porous material. *J. Chem. Phys.*, 136(18):184108, 2012.

[201] Michael Habeck. Bayesian Estimation of Free Energies From Equilibrium Simulations. *Phys. Rev. Lett.*, 109(10), sep 2012.

[202] Xianjun Kong and L Brooks III Charles. λ-dynamics: A new approach to free energy calculations. *J. Chem. Phys.*, 105:2414–2423, 1996.

[203] David Dubbeldam, Ariana Torres-Knoop, and Krista S. Walton. On the inner workings of Monte Carlo codes. *Mol. Simul.*, 39(14-15):1253–1292, 2013.

[204] Alexandros Beskos, Gareth O Roberts, Jesus-Maria Sanz-Serna, and a M Stuart. Optimal tuning of the hybrid Monte-Carlo algorithm. *Arxiv Prepr. arXiv10014460*, 106(6):2077–2082, 2010.

[205] Lev D. Gelb. Monte Carlo simulations using sampling from an approximate potential. *J. Chem. Phys.*, 118(17):7747–7750, 2003.

[206] Jeff D. Kahn, Nathan Linial, Noam Nisan, and Michael E. Saks. On the cover time of random walks on graphs. *J. Theor. Probab.*, 2(1):121–128, 1989.

[207] Johan Jonasson and Oded Schramm. On the cover time of planar graphs. *Electron. Commun. Probab.*, 5:85–90, 2000.

[208] Ilario G. Tironi, Rene Sperb, Paul E. Smith, and Wilfred F. van Gunsteren. A generalized reaction field method for molecular dynamics simulations. *J. Chem. Phys.*, 102(13):5451, 1995.

[209] Jacob I. Monroe and Michael R. Shirts. Converging free energies of binding in cucurbit[7]uril and octa-acid host–guest systems from SAMPL4 using expanded ensemble simulations. *J. of Comp.-Aided Mol. Des.*, 28(4):401–415, 2014.

[210] David Chandler and Hans C. Andersen. Mode Expansion in Equilibrium Statistical Mechanics. II. A Rapidly Convergent Theory of Ionic Solutions. *J. Chem. Phys.*, 54(1):26, 1971.

[211] A. Ciach, W. Góźdź, and G. Stell. Field theory for size- and charge-asymmetric primitive model of ionic systems: Mean-field stability analysis and pretransitional effects. *Phys. Rev. E*, 75(5):051505, May 2007.

[212] O. Patsahan and A. Ciach. Field theory for size- and charge-asymmetric primitive model of ionic systems: Mean-field stability analysis and pretransitional effects. *Phys. Rev. E*, 75(5):051505, May 2007.

[213] William Burnside. *Theory of Groups of Finite Order*. Cambridge University Press, New York, 2nd edition, 2012.

[214] Our Examol implementation can be on GitHub at `https://github.com/Lnaden/examol/tree/nadenbuild/nadenexamol` with commit hash eb332da or later. Accessed July 3, 2016.