pH-DEPENDENT FREE ENERGY CALCULATIONS FOR EXPLICIT SOLVENT MOLECULAR DYNAMICS

by

Andrew Alva Petersen

BS, Morehouse College 1996

Submitted to the Graduate Faculty of

Arts and Sciences in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2006

UNIVERSITY OF PITTSBURGH

FACULTY OF ARTS AND SCIENCES

This dissertation was presented

by

Andrew Alva Petersen

It was defended on

June 5, 2006

and approved by

David Jasnow, Professor, Department of Physics, University of Pittsburgh

X. L. Wu, Professor, Department of Physics, University of Pittsburgh

Daniel Zuckerman, Assistant Professor, Department of Computational Biology, School of

Medicine, University of Pittsburgh

Ralph Roskies, Professor, Physics Department, University of Pittsburgh

Robert H. Swendsen, Professor, Physics Department, Carnegie Mellon University

Dissertation Advisor: John M. Rosenberg, Professor, Biological Sciences, University of

Pittsburgh

ii

pH-DEPENDENT FREE ENERGY CALCULATIONS FOR EXPLICIT
SOLVENT MOLECULAR DYNAMICS

Andrew Alva Petersen, PhD

University of Pittsburgh, 2006

Designing drugs for treating diseases is one of the main motivations for understanding how proteins are able to recognize their substrates. Recent growth in computational power has encouraged the use of numerical tools like atomic detailed molecular dynamics for investigating proteins.

Until recently, atomic detail molecular dynamics did not allow for the transfer of protons in the solute or solvent of the model during dynamics. Modeling this transfer in the protein is important because there are seven titratable amino acids. This means that they can exist in different protonation states or states of charge. The most important titratable sites are usually deeply buried. Several methods are available for doing proton dynamics for the titratable amino acids of the solute. Unfortunately deeply buried sites challenge available methods because the models need to capture the hydrophobic effect of buried regions, the hydrophilic effect of solvent penetration and the subtlety of charged networks. These effects sometimes assist, compete, or balance each other.

One solution for the above challenges is to exploit the accuracy that comes with a full atomic detailed explicitly solvated model. However such an approach runs into problems because protonation state changes at 300K require unreasonably long simulations due to solvent reorientation relaxation times. As a result, currently available methods compromise the atomic detail description in some way, either by using continuum protonation states, by using continuum solvent, or by stepping back from the atomic detail description.

Our method uses both discrete protonation states and atomic detail explicit solvent. The water orientation problem is overcome by using elevated temperatures, and the information from

a wide range of temperatures, including those at 300K, are woven together with our Weighted Histogram algorithm. This then gives us an accurate density of states, from which we can calculate a full range of thermodynamic results.

We used our methods to calculate the Bond Dissociation Energy (BDE) of the $H-S$ bond in the solvated single site Cysteine system. $BDE_{CYS}^{calc} = 90.3 \pm 1 kcals/mole$. We have found this number agrees to within 3% of the experimental BDE of a very similar bond in thiomethane, $H-SCH_3$. $BDE_{thio-methane}^{exp} = 88 \pm 1 kcals/mole$. This is very good agreement and is some validation of our methods.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

Our interest in proton dynamics and *pKa* calculations began as a result of investigating the specificity of the *EcoRI − DNA* system. Specifically, we were probing the nature of the protein-DNA interaction using Molecular Dynamics and Free Energy Perturbation calculations. We were trying to calculate various interaction energy components, such as the contributions of specific base-pairs towards the overall protein-DNA interaction, and compare those results with experimental numbers. This comparison with experimental results required a level of precision that demanded us to properly model the protonation state of our system. As we studied available methods for solving our problem, we also became aware of the challenges that faced protonation state or *pKa* calculations, and we saw how challenged the available methods were to solve these problems to the level of precision that we wanted. We then understood that we were in a unique position to make a substantial contribution in this area because of the special talents of the individuals of our group: Prof. J. Rosenberg and Prof. R. Swendsen brought together experience in Structural Biology, Molecular Dynamics, Monte Carlo Methods, Statistical Mechanics and Histogram Analysis. My contribution towards the vision was naiveté: I, blinded by the handsome vision of the project, failed to appreciate the length of time required to do all the work. I am still blinded. I am made aware only by my wife Hazel-Ann, who somehow more resistant to the project's charms, has a much fuller appreciation of the sacrifice of time and work that I have made. I am very happy with the end product and I realize that things, thank God, worked out for the best, naiveté and all.

1.0 INTRODUCTION

1.1   OVERVIEW

Proteins are made up of amino-acids and there are several amino-acid types that are titratable. That is, they can absorb or shed protons, thereby existing in different charged or protonation states. This work is inspired by the need to accurately describe the protonation states of systems such as protein-DNA complexes. The purpose of studying these protein-DNA simulations is to elucidate molecular mechanisms of sequence specific protein-DNA interactions as well as address broader issues of enzyme-substrate recognition and the molecular basis of specificity.

Protein-DNA complexes, and other such macromolecule complexes, have interfaces between macromolecules that often contain many titratable and/or charged groups. These charged groups, which may have been near the surface before complex formation, often become deeply buried after complex formation. The problem of assigning protonation states to titratable sites is very important for the proper electrostatics of Molecular Dynamics (MD) simulations. Assigning fixed protonation states is insufficient to model dynamic protonation state effects such as those that would be expected to correlate with configurational changes. In terms of model detail, there is a range of methods available for modeling protonation state effects. Consider the least detailed end of the spectrum. Several continuum solvent methods have been developed for protonation state determination where the solvent is treated as a macroscopic dielectric, and some even allow for dynamical protonation effects. These methods give reasonable results for titratable sites near the surface. However for deeply buried sites, or sites involved in electrostatic networks, there is no consistent value or reasonable way of choosing a dielectric constant to get the calculated *pKas* to agree with experiment for all titratable sites[1,2]. The advantage of such methods is their computational efficiency. These methods represent one end of the spectrum for modeling protonation state effects.

The other end of the spectrum would be a full Ab Initio quantum mechanical simulation that would allow the solvent water molecules to be titratable as well as the titratable sites of interest. Current computational resources would limit such treatment to the smallest systems for relatively short simulations. They therefore cannot be currently used to simultaneously calculate macro *pKas* for any solvated proteins of interest, or any charge network regions of interest that are not highly localized.
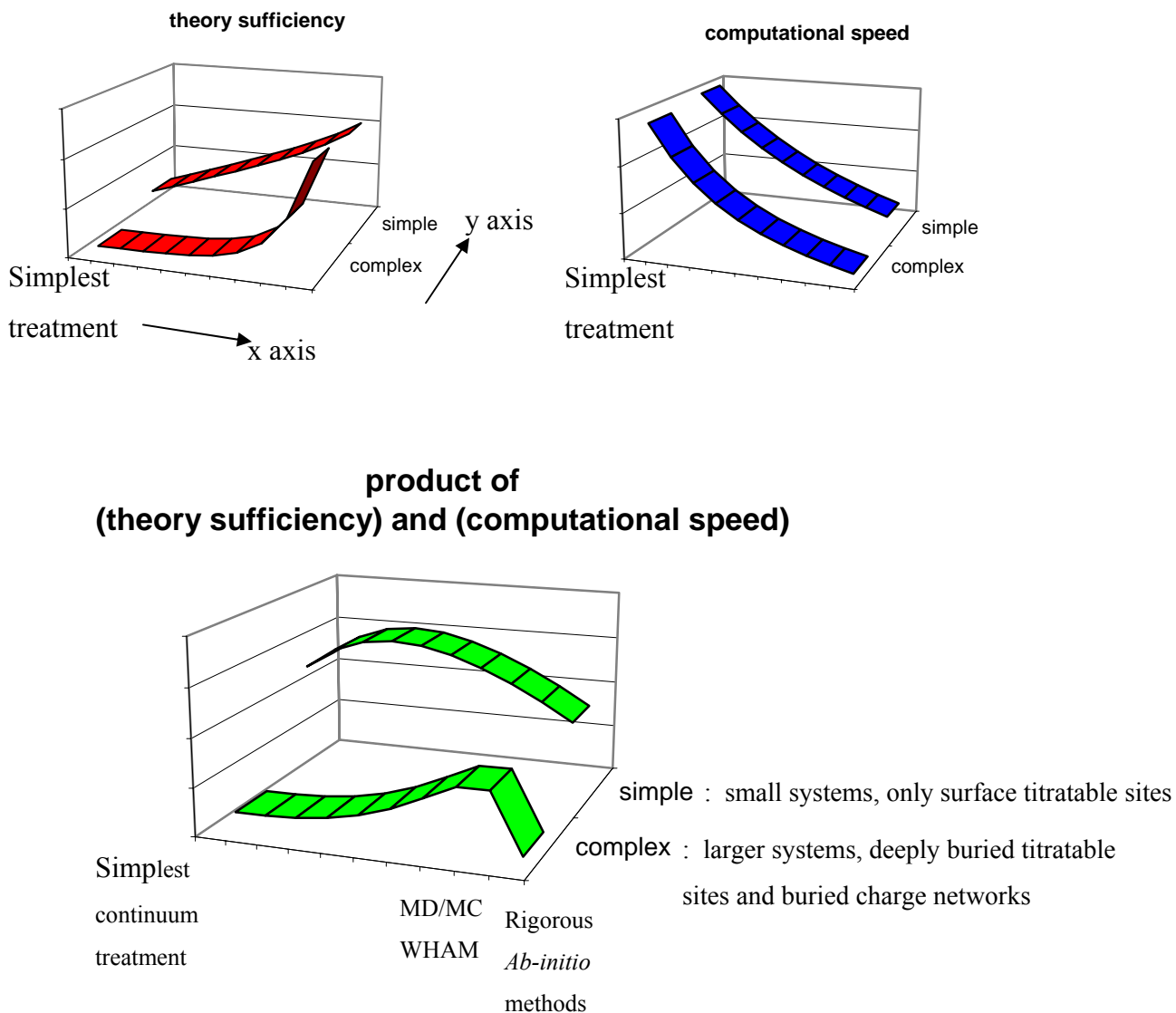
Figure 1: Computational Feasibility of Various Proton Dynamics Models

The plots above summarize our dilemma, and are typical of computational biophysics problems (Quantitative rigor is absent from the plots above. They serve only as an aid to qualitative

description). The x-axis of all three plots represents a range of protonation assignment approaches: from the simplest continuum, macroscopic, static protonation assignment methods, to the most rigorous *Ab Initio* dynamic protonation assignment methods. The y-axis represents a range of systems, in which the foreground represents complex systems. These will be larger systems or systems that contain buried titratable sites, buried networks that contain titratable sites or titration state fluctuations that are suspected of being correlated with configurational fluctuations. Many systems of interest fall into this category, including protein-DNA systems that we examine (protein-DNA systems are of interest to us because they are models of specificity). We place our MD/MC-WHAM approach closer to the right side extreme of the x-axis. The only feature that separates our method from the extreme right side is that we are using classical mechanical force field dynamics, as opposed to *Ab Initio* dynamics. Our approach uses full atomic detail to describe both solute and solvent, and the titratable sites are modeled as discrete states. A brief description of these two features now follows. The full atomic detailed description is considered the most accurate description next to the *Ab Initio* description (see Figure 19: Feasibility of various modeling methods). In summary, the full atomic detailed description is one where every atom in a solvated biomolecule is explicitly represented and assigned mass, charge, van der Waals parameters etc. Further discussion on atomic detail is given in section 1.11.3, "Molecular Dynamics". Discrete protonation state modeling means that the transitions of a titratable site from one state to the next are not continuous but are discrete, which is a more accurate representation of nature (there is no such thing as half a proton). We believe our method is one of the few options for achieving reasonable results for complex systems of interest, using currently available computational prowess, such as that available on in-house computing clusters or at supercomputing centers.

The problem with using atomic detailed explicit solvent classical mechanics force fields to model protonation is that at 300K the energy barriers that separate the protonation states are such that unreasonably long simulations would be required to properly sample all protonation states. This is because in our simulations, the waters that surround a titratable/charged group orient in response to the electrostatics of the titratable/charged group and form a solvation shell. One possible solution would be to use a titratable water model. However a typical solvated system for MD simulation has thousands of water molecules that are highly mobile. So every one of the waters in the system would have to be titratable not just the ones surrounding a titratable

site at one snapshot in time, thereby rendering such a system model computationally unfeasible. Besides, a titratable water model will still have solvation barriers that are still likely to necessitate unreasonably long simulation times. Our approach to crossing the barrier separating protonation states within quick simulation times at 300K is to use a "simulated annealing ensemble" and Weighted Histograms (WHAM[63,64]) to calculate the density-of-states using many trajectories generated under many conditions. Some of these trajectories are generated at high temperatures where the ionization transition rates are high and protonation state sampling is vigorous. The high temperature and low temperature trajectories are woven together with WHAM[63,64]. The high temperature information serves to get an approximation for the density-of-states, and the low temperature information serves to fine tune the weighting factors to yield a density-of-states description that is accurate for calculating many thermodynamic parameters, including *pKas*, at 300K.

1.2   MOLECULAR BIOPHYSICS

1.2.1   Why study Molecular Biophysics? Fun.

All of science is driven by the curiosity of understanding how things work. The Natural Sciences are specifically driven by the curiosity of the mechanics of how nature works. The Molecular Biophysicist's main target of investigation, small biological systems like proteins, offer manifold and rich opportunities to understand some of the most complex machinery in nature, and also understand some of the most important machines of nature. In the realm of understanding the mechanics of nature, what could be more important than the mechanics relating to that which supports life? Because biological systems like proteins are considered the most complex sub-cellular systems of nature, Biophysicists are therefore inspired to investigate these systems and learn: from either the designs of God, or the designs of the evolutionary forces of time, chance and selection, or the evolutionary forces of time, chance and selection set in place by God.

A widely accepted definition of Biophysics is the application of physics disciplines to Biological systems. The phrase Biophysics was only first used about 50 years ago. As a result, the Biophysics field is still the arena of vigorous culture clashes between the Biological Science and Physics cultures, and is attractive to those people that see these clashes as evidence of a field full of opportunity. Biological Science culture traditionally emphasizes, and amasses detail[3]. The reason being that the living world is inherently complex, diverse and changes so much over time and environment.

On the other hand, the physics culture tends to approach understanding natural phenomena by looking for Universal laws, finding what the phenomena have in common and to simplify. Many physicists have been lured into a world they see filled with unruly details, crying out for them to subdue and bring order to the chaos. The complexity of seemingly simple biological processes usually subdues any excessive confidence.

1.2.2   Why study Molecular Biophysics? Important.

Understanding how proteins work is the key to understanding why they don't work. This field is therefore applicable to understanding diseases and designing drugs. The secrets of specificity (how proteins are able to selectively bind to substrates) may reveal keys for drug design. One indicator of the relationship between Biophysics and treating disease is the fact that more and more research hospitals have Structural and Computational Biophysics research departments.

1.2.3   Why study Molecular Biophysics? Profitable

For several centuries, one can only make progress in science if it is treated as a career instead of a hobby. Therefore the practicality and profitability have to be considered in choosing a field of science. Not only is Molecular Biophysics fun and important, but career wise, it is practical as well. Molecular Biophysics attracts significant funding from both private and government sources, medical and general-scientific institutions. Over the past 20 years, Molecular Biophysics Programs have seen a multi-fold funding increase relative to the total NSF budget (see Figure 3 on the next page, page 6). Figure 2 shows that the traditional source of interest in Molecular Biophysics was from investigators in the field of medical research (NIGMS, the

National Institute of General Medical Science, is a sub-institute of the National Institute of Health).
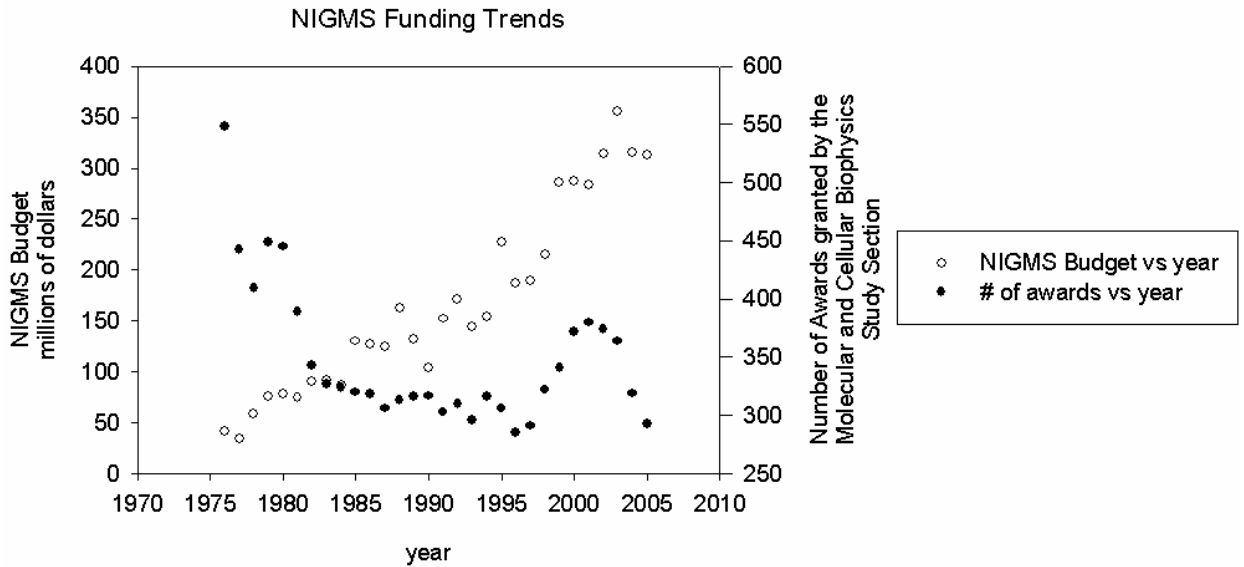
Figure 2: NIGMS Funding Trends[4,5]

Figure 3, showing an almost inverse trend compared to Figure 2, shows the maturity of the field over time so that it attracts the interest of investigators from the fundamental sciences.
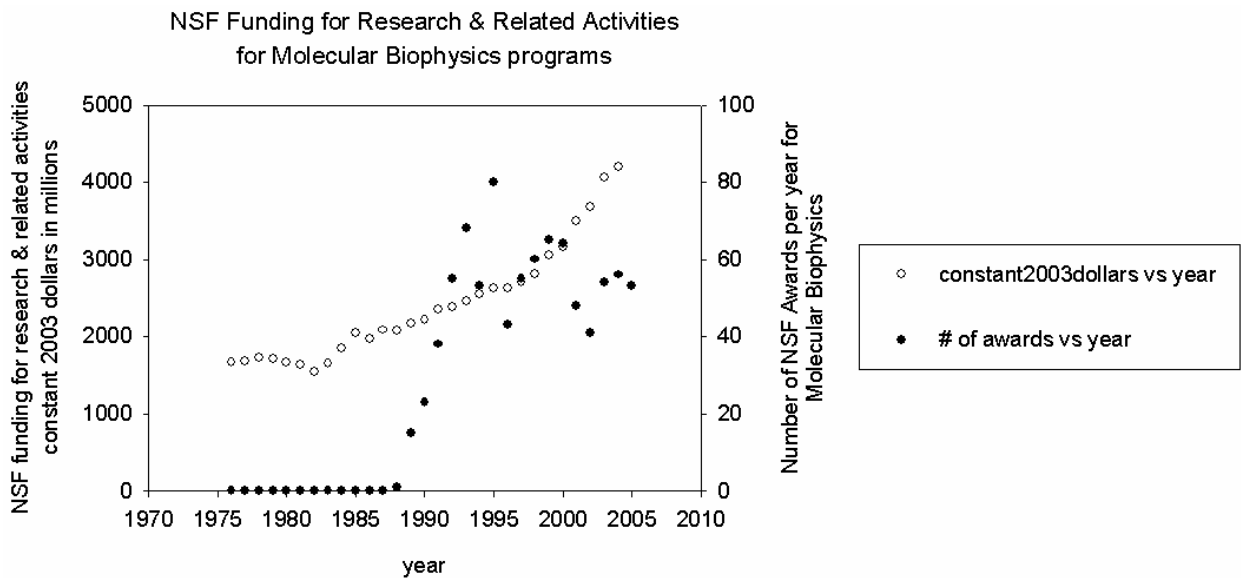
Figure 3: NSF Funding for Molecular Biophysics Programs[6,7,*]

This is the reason why more Molecular Biophysics research is overflowing from medical research institutions, and driving stakes into scientific research institutions. The University of Pittsburgh is typical of many academic research institutions that have built graduate Molecular Biophysics Programs within the past fifteen years.

Trends for Supercomputing resources dedicated to Molecular Biophysics are shown below in Figure 4. Data can only be tracked back to 1996. However, within the past 10 years, supercomputing prowess has improved significantly. The plot shows that, through more scalable models, and through more scalable and faster algorithms, computational Molecular Biophysics investigators have been able to do better than keep pace with supercomputing performance, and are fully exploiting supercomputing resources (the plot shows the number of LARGE allocations vs. year. Large allocations can only be efficiently used with highly parallelizable algorithms).
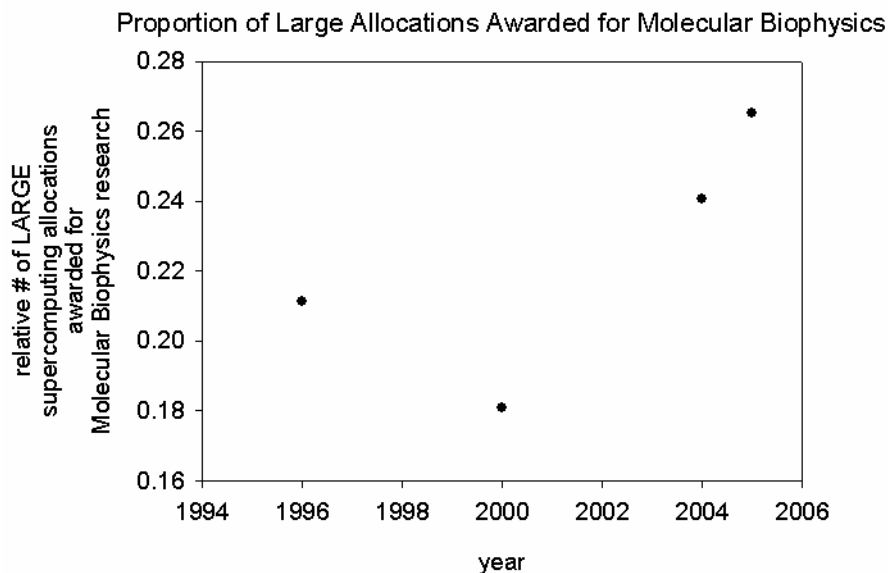
Figure 4: Supercomputing Awards for Molecular Biophysics[8]

1.3   SURVEY OF BIOLOGICAL PHENOMENA OF INTEREST

1.3.1   Protein Folding

Amino acids are the building blocks of which proteins are made. There are 20 unique amino acids. Every protein has a unique sequence of these amino acids. This unique sequence is known as the primary sequence and can be thought of as a chain of links, where each link is an amino acid. This chain is generated one amino acid link at a time by ribosome acting with instructions from tRNA. The protein may then spontaneously fold on it own, or may require help from other proteins called chaperones[9]. For many proteins, its primary sequence determines its structure and function. What's so amazing about folding is that the protein starts the process with a vast number of possible conformations. It then follows free-energy reducing folding pathways to end up in its 'native configuration'. The folded protein, in its native configuration, is a stable three-dimensional structure. It can be thought of as a balled-up chain, because the amino acids share important interactions in more than the one dimension of the link sequence.

Despite the fact that the folding process is not yet completely understood, there are plausible models for folding and some consensus on the thermodynamics of the process. One model for describing folding has the primary sequence first form local alpha-helix and beta-sheet structures due to hydrophobic interactions, then the local structures aggregate via longer range interactions. Another model (not necessarily independent of the fore-mentioned model) has the primary sequence collapse into a molten globule form then more slowly follow a number of possible pathways to the native conformation. There is some consensus that thermodynamics of the process resembles a funnel with bumpy surfaces. As folding proceeds down the funnel, the protein lowers its free-energy and reduces its entropy by narrowing the number of configuration possibilities for the next folding step[10,11,12,13,14,15,16].

Through the decades investigators have invested a lot into understanding protein folding: Mutation, chaperone, catalyst and environment monitoring experiments; folding theories and computational models, have all been thrown at the protein folding challenge. A lot of headway has been made into understanding how protein folding overcomes the apparently daunting barriers of entropy, but there are so many more secrets and the current folding theories and

models are still so inadequate, that protein folding is still considered as much of a holy grail as it was 20 years ago.

1.3.2   Molecular Recognition and Protein Specificity

Molecular recognition is the ability of one molecule to recognize and interact with another molecule. Many factors bear on how a molecule is able to recognize its substrate. Some of the more obvious factors are size, shape and charge of the substrate. However the recognition ability can be very refined, allowing the biomolecule to perform complex tasks. Therefore there are many other important factors that go into a biomolecules ability to recognize. For example, subtle changes in environment could be essential for biomolecular recognition and function. Hemoglobin is a good example of a biomolecule with admirably refined molecular recognition and which can significantly change operation details by "sensing" environmental pH changes. Near the lungs, hemoglobin has a high affinity for the oxygen molecule, which it selectively binds to. Via the blood stream, the hemoglobin carries the oxygen to the muscles. Here, the environment is acidic. This causes the release of the oxygen and the binding of a CO_2 molecule. The hemoglobin then transports the CO_2 back to the lungs for exhalation, and the cycle repeats. The function of hemoglobin has been the target of much study for a long time. It is a testimony of the complex nature of its function that it is still a target of much study, considering the improvements in structural experiments, computational modeling and computational resources. However CO is poisonous because, in the lungs, hemoglobin has a greater affinity for CO than it does the oxygen. This is an example of hemoglobin's recognition failing, with deadly effects. We will now visit a class of biomolecules that has extremely refined recognition ability, some of the best in the business. These are proteins that are involved in protein-nucleic acid interactions. Understanding these interactions is of utmost biological importance, since these interactions are the key to biological regulations and DNA repair. EcoRI is a restriction endonuclease that has a very high ability to recognize its cognate DNA substrate (a DNA fragment with a specific sequence, in this case GAATTC). EcoRI can cleave its cognate DNA substrate more that 10^5 times faster than it can cleave an alternatively sequenced DNA strand under standard conditions[17]. This recognition ability of a protein is called specificity.

What drives the study to understand protein recognition and specificity is not simply to satisfy a hunger for understanding some of nature's most complex mechanisms. What also drives this study is the need to improve protein engineering and design. Current pharmaceuticals rely on specific interactions with their intended targets. If the artificial molecule is too promiscuous (the opposite of having a high degree of specificity), its efficacy will be reduced and there will be increased side effects. Protein engineering and design that employs a better understanding of specificity, offers the potential to engineer therapeutics that possess better specificity for an intended target substrate, and better environmental sensitivity.

1.3.3 Ion Channels and Ion Pumps

Cells are bounded by a bilayer membrane. Ion channels and ion pumps are molecular devices that are embedded into bilayer membranes and are gatekeepers responsible for the transfer of specific ions in and out of the cell. They can sense external conditions and respond by adapting their permeability to specific ions. Ion channels simply allow the passage of specified ions under specified conditions, whereas ion pumps actively pull specified ions across the bilayer membrane. One example of such a device is the sodium-potassium pump. This device binds sodium ions and ATP on the inside of the cell's plasma membrane, expels the sodium ions from the cell, binds potassium ions on the extracellular side, pumps them into the cell, and then releases ADP. This device therefore plays an important role in the complex energy relay system of energy transfer in living organisms.

1.3.4 Water

Why should the behavior of water be considered a biological phenomenon of interest? Biomolecules function in an aqueous environment and water plays an important role in the whole range of fascinating biomolecular phenomena. In the words of Gerstein and Levitt (1998)[18]:

> When scientists publish models of biological molecules in journals, they usually draw their models in bright colors and place them against a plain, black background. We now know that the background in which these molecules exist -water- is just as important as they are.

In this section we will briefly survey the important roles played by water, and give the reader a glimpse of why water deserves to be considered a biomolecule of interest. The growing recognition of water's importance in the function of biomolecules is underscored by the following fact: In the history of evolution of Molecular Dynamics force fields, new models for water outnumber new models for amino-acids two-to-one.

1.3.4.1 Bulk water

Interestingly enough, if the mixed biophysics demographic were crudely separated into those of physics and those of biology ancestry, the drive to elevate water to the status of biomolecule is the result of the admiration of water by those of physics ancestry. Biologists seldom work with anything but an aqueous environment, so they often take for granted the peculiarities of water's behavior. However the condensed-matter physicist is well aware that water breaks all the rules of liquid-state theory.

Water is a unique liquid with profound characteristics even when it's all by itself, in bulk form, not interacting with any biomolecules. The key to most of water's anomalies is the hydrogen bond network. Hydrogen bonds are strong and directional, causing a tetrahedral motif to be repeated throughout all three dimensions of the water. It is not a regular structure because it is constantly being rearranged on a sub-picosecond time scale. The short-ranged order of the tetrahedral network of hydrogen bonds prevents the molecules from moving too close to each other. However water molecules often move closer to each other than the tetrahedral structure would allow because the tetrahedral structure is continually being broken. When water cools from 4 degrees to 0 degrees Celsius, the tetrahedral network gradually freezes into a regular structure. This explains waters expansion on freezing.

1.3.4.2 Solvation Shells and Hydrophobicity

The behavior of pure bulk water is strange enough, far more for water's interactions with biomolecules. So we should also be suspicious of any simple model of how water interacts with solute. Much of the discussion of water-biomolecule interaction is wrapped up in the discussion about the nature of hydrophobicity. Hydrophobic interactions are very important in biophysics. They are considered to be the driving force for protein folding and lipid self-assembly into membranes. There is a lot of literature about how water forms or does not form solvation shells

in hydrophobic or hydrophilic regions of a solute, and how these shells explain the thermodynamics of hydrophobicity during protein folding or lipid layer assembly. However it is experimentally challenging to "see" solvation shell structure, so the lack of evidence, combined with water's ability to form intriguing interactions, gives good reason to be skeptical about simple models of water-solute interaction. That hydrophobic interactions can be long ranged[19] (up to dozens of nanometers in length or several hundred molecular diameters[20]) further challenges experimental and computational tools for investigating hydrophobic interactions. Single-molecule probe experimental techniques, such as the atomic force microscopy, promise to shed light on the nature of solvation shell[21] structure and related thermodynamics.

1.3.4.3 Trapped water, Proton Wires

In x-ray protein crystal structure determination, it is common to consider bound water as part of the structure. These are water molecules that have been adopted from the solvent to form part of the structure of the macromolecule. In many cases the water is trapped inside very small cavities that are only nanometers wide. Given water's reputation for intriguing interactions, it is reasonable to expect water to behave at least as mysteriously under these conditions as it does in bulk solvent. In some cases, trapped water forms one-dimensional chains of hydrogen bonded water molecules known as proton wires. Proton wires can play an important role in the rapid translocation of protons in proton transport networks. Proton wires are centrally involved in essential metabolic processes within plant and animal cells, such as photosynthesis and pumping ions from one side of a membrane to the next.

1.3.5   Time and length scales of various phenomena of interest

Knowing the approximate time and length scales of the various phenomena of interest is useful. It facilitates a quantitative description of the phenomena. It helps in understanding the feasibility and the limitations of various experimental methods for exploring various phenomena. It also helps to understand the feasibility and limitations, of various computational models running on a given computational resource, for exploring various biological phenomena. The issue of computational feasibility will be discussed further in section 1.11.4.

The plot below (Figure 5) shows that interesting biological phenomena span orders of magnitude in length and time. The discussions of sections 1.3 and 1.4 also tell us that many of the large-scale phenomena of interest are driven by small-scale behavior, so understanding the first is facilitated by understanding and accurately modeling the latter. For example, protein folding and lipid membrane assembly (large-scale phenomena) is driven by forces that are the result of water-solute interaction behavior, in which the hydrogen-bonding networks play a central role. In other words, accurate simulations of the larger-scale phenomena cannot happen without computer models that capture the behavior of the small-scale phenomena to a sufficient degree of accuracy. Therein is the challenge of computational modeling of biomolecules: spanning several orders of magnitude of time and space to connect the most detailed steps of the model to simulating a phenomenon of interest.

In order to help get a perspective of the time and length scales in which various phenomena occur, the following measurements are helpful. The speed of sound in water is about 15Å/ps. The period of the smallest oscillations and the period of the largest oscillations of the nodes in typical systems, determined from NMR experiments, range from about 5 femtoseconds(fs) to 50 fs, and the corresponding magnitudes of oscillation range from 0.1 Å (typical covalent bond length is about 1.2 Å and the typical covalent bond distortion amplitude is about 0.1 Å) to 5 Å. These all give an idea of the speed at which different genres of information may travel across a solvated biological system.

Protein assembly occurs on time scales that range from microseconds to seconds. Circular Dichroism Absorption Spectroscopy, Fluorescence Spectroscopy and Raman Excitation experiments have shed light on protein folding rates.[22]

Most involved protein functions occur on the scale of nanoseconds. This estimate is obtained from direct observation of vibrational relaxation of the collective modes in proteins using Far Infrared and Medium Infrared pump probes[23].

Many simpler biological functions, like those that involve proton transport in a network of titratable sites, happen on a sub picosecond scale (<200 fs). These approximations are arrived at by femtosecond 2D IR spectroscopy of systems containing hydrogen bond networks[24].
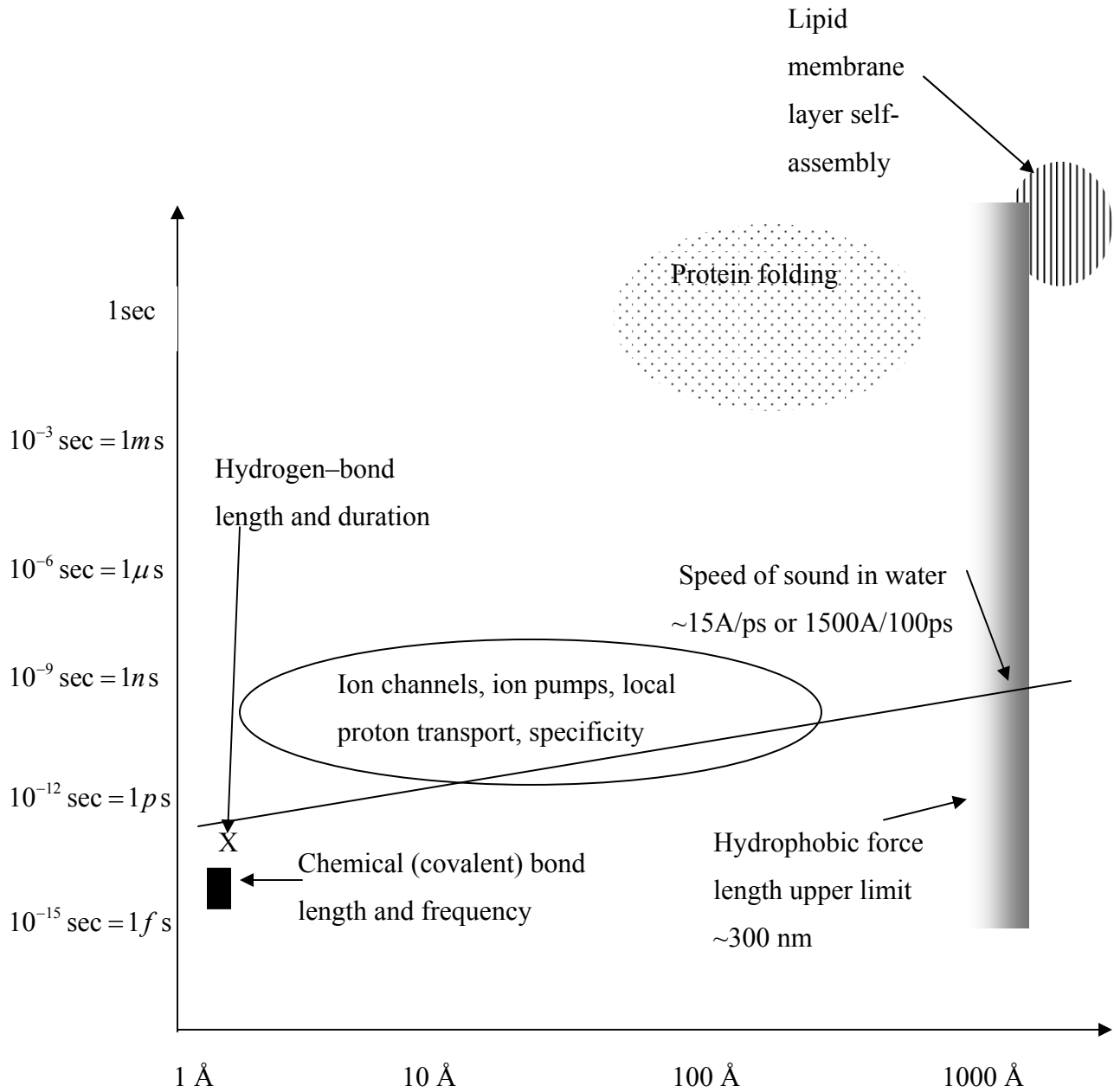
Figure 5: Time and Length scales of various biological phenomena of interest

1.4   SURVEY OF PROTON DYNAMICS IN BIOLOGICAL SYSTEMS

Proton dynamics plays important roles in the function of a wide range of molecular devices. In this chapter we emphasize this ubiquity by surveying the widespread involvement of proton dynamics in a range of biomolecular systems.

1.4.1   Water, Proton Dynamics, and Solvated Systems

It is appropriate for us to start our survey of proton dynamics in biological systems with water. As will be seen in the following sections, water is integral to the function of biological systems and, as mentioned in the previous sections, exhibits behavior that is complex and very different from that of a simple liquid.

By definition, *in vitrio* biological systems are in a solvated environment. This environment plays an integral part in the function of biological systems. The main reason for this is that the solvent is an important facilitator and channel for proton movement. The following sections will emphasize the importance of solvent and the solvent-solute interaction for proton dynamics and biological system function.

1.4.1.1   Proton and H_3^+O dynamics in water

Water is a fascinating liquid with regards to its role in supporting life and the physical chemistry of its nature. The structure of water has been the target of study of many works. This section will summarize the structure of water and emphasize the proton dynamics aspect of water.

Water is a polar molecule, and this property is one of the main reasons for its interesting characteristics. In this molecule, two hydrogen atoms donate their electrons to the orbital of the oxygen. The oxygen receives these donations from asymmetric positions, such that the hydrogen donors are on one side of the oxygen nucleus (subtending an angle of 108 degrees). The water molecule therefore consists of three atoms, joined to each other by two ***chemical or covalent*** O-H bonds. These **inter-atomic** chemical bonds that join the atoms of a water molecule, like any chemical bond, cannot be explained simply in terms of classical electrostatics. They can be explained in terms of quantum mechanics principles that relate to atom wave function overlap or electron orbital overlap[25]. (Pauling 1960 "The Nature of the Chemical Bond").

However, the **inter-molecular** water interactions can be described in terms of much simpler classical electrostatics. The most significant of these interactions is the ***hydrogen-bond.*** The term "hydrogen-bond" is very suggestive of a bond that is chemical or a covalent in nature, but it is not. A real chemical bond, as just mentioned, cannot be described in terms of classical electrostatics, but a hydrogen-bond can, and such an explanation now follows. Using a classical description of water's oxygen, the oxygen atom has 6 of its 8 electrons in the outer orbital (so these 6 are available for chemical bonding or other interactions). Two of the six outer orbital electrons are involved in the two chemical bonds with the hydrogen atoms (one electron in each chemical bond). This leaves two lone pairs of electrons on the side of the oxygen that faces away from the chemically bonded hydrogen atoms. Hydrogen bonding is simply the electrostatic attraction between those negatively charged electron pairs of the oxygen and the positively charged electron-stripped hydrogen nuclei of a neighboring water molecule. The two free oxygen electron pairs are able to form two hydrogen bonds. At any given instant, this interaction between neighboring water molecules takes place throughout the liquid, ordering the orientation of all the molecules and resulting in the unique structure of water. However, these hydrogen bonds have a short life time, and are continually being broken and remade, which allows the molecules to switch hydrogen bonding partners on a time scale of under two hundred femtoseconds[24]. It is this flexibility of each molecule within the structure that allows water to be a liquid.
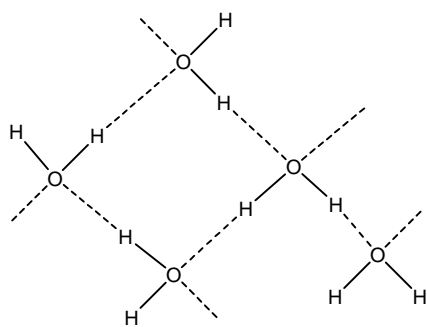
Figure 6: 2D representation of Hydrogen Bonds in Bulk Water[26]

Above is a 2D representation of five water molecules, and the hydrogen-bonds of the hydrogen-bonding network are represented as dotted lines, and the chemical bonds are solid lines.

The oxygen of the water can do more than ***covalently bond*** to two hydrogen atoms, and ***hydrogen bond*** two others. Sometimes one of the ***hydrogen bonds*** becomes promoted to a ***covalent bond,*** allowing the water oxygen to have three covalent bonds, $H_2O + H^+ \rightleftarrows H_3^+O$. This ability to covalently bind three hydrogen atoms, the temporary nature of this three covalent bond status, the hydrogen bonding network, and the temporary nature of the hydrogen bonds, all contribute to the ability of proton travel. The protons use the hydrogen bonding structure of water as channels of travel. The protons also use the temporary nature of the hydrogen bonds and the temporary nature of the three-covalent bond status for "virtual" travel. That is, subtle shifts in the hydrogen bond network allow the H_3^+O ion to shift from one molecule to its neighbor. This is very similar to how holes travel in semiconductors. It is interesting to note that the conductivity of water (5×10^{-4} to 2 seimens/meter), in between that of an insulator and a conductor, is similar to that of semiconductors used to fabricate electronic devices (≈ 0.15 seimens/meter)[27]. The diagram below describes this virtual H_3^+O travel.
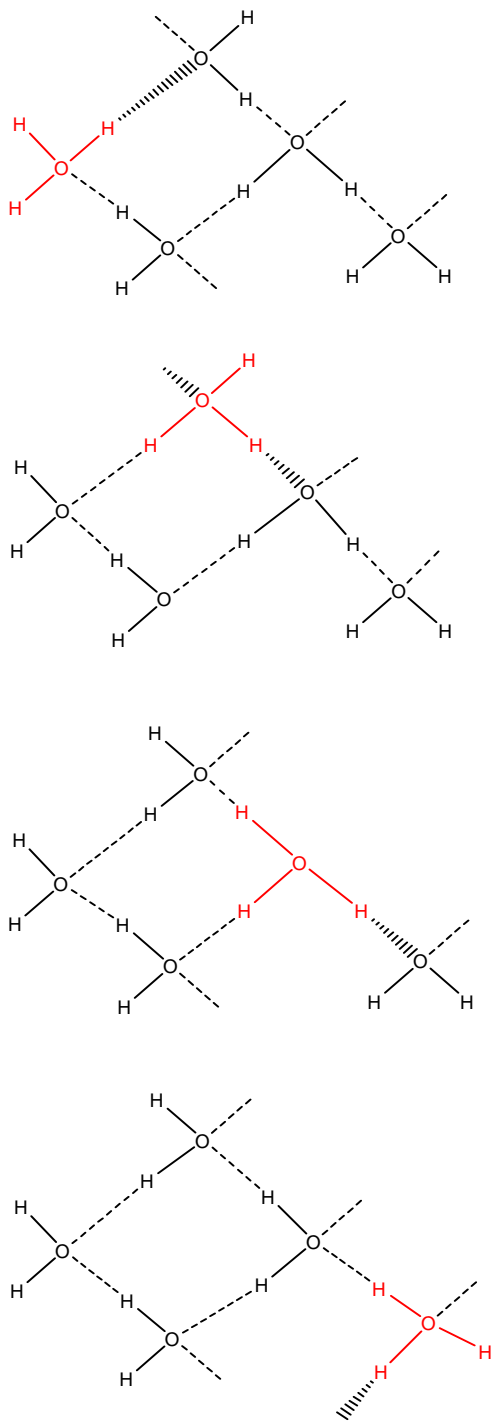
Figure 7: 2D representation of Proton travel in Bulk Water[26]

As before, the hydrogen bonds are represented as dotted lines, ------. The H_3^+O ion is represented in red, and imminent covalent bond formation is represented by the wedged dotted line, """""""". Notice that we expanded on H_3^+O ion travel, but not on proton (H^+) travel. Where solvent (water) is concerned H^+ (proton) travel and H_3^+O travel are synonymous. The effect is the same, which is that one electron charge is moved from one location of the solvent to the next. The difference between the H^+ ion and the H_3^+O ion has more significance where the water interacts with the titratable site of the solute. This solvent-solute proton interaction will be discussed in more detail in the next section. For all the reasons described above, we can summarize the nature of proton dynamics in water as follows. Subtle shifts in the hydrogen bonding structure within water allow for the disappearance of H_3^+O in one location and its simultaneous formation in a nearby location, which allows for very effective, efficient and rapid "virtual" travel of an electron-charge. This ability to rapidly whisk protons to or from any location effectively makes water a good proton reservoir. It's a proton reservoir because in a solvated biomolecule system, proton transfer to or from the solvent does not make a significant change to the concentration of protons in the solvent. It is a *good* proton reservoir because the transfer of protons to and from the solvent, due to the water, happens without any hysteresis.

1.4.1.2   Proton Dynamics in Titratable Regions of a Solute

We have just discussed how effective water is at being a proton reservoir. If a titratable region is solvent exposed, it can exchange protons with the solvent. Where proton dynamics is concerned, the main difference between solute and solvent is that the +1 electron charge travels through the solute in the form of a H^+ proton, and it travels through the solvent in the form of H_3^+O ions. The transfer of +1 electron charge at the solvent-solute interface is initiated by the making or breaking of one of the bonds in H_3^+O, where this water molecule (or ion) is the one the titratable site interacts with. The diagram below illustrates this transfer of proton from Cysteine to water.

protonated cysteine

deprotonated cysteine
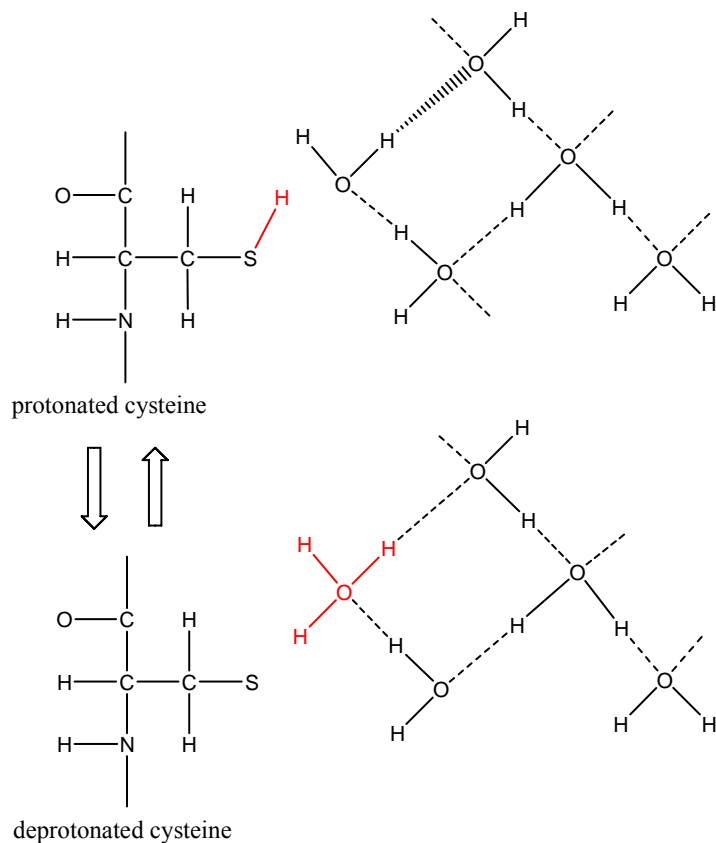
Figure 8: Proton Dynamics at the solute-solvent interface[26]

1.4.1.3 Hydrolysis of water, Peptide Bond hydrolysis, ATP hydrolysis

Water hydrolysis is simply the dissociation of water into ions, $H_2O \rightarrow H^+ + OH^-$.

Peptide bond hydrolysis is simply using the ions produced by water hydrolysis to help attack and break a peptide bond. Many important mechanisms, such as ATP (Adenosine triphosphate) synthesis and function, involve water hydrolysis at the core of their operations.

ATP molecules are the fuel cells for many molecular machines and devices within cells.

$$H_2O + ATP \rightleftarrows ADP + P_i + H^+ + Energy$$

ATP releases its energy to the molecular machines, according to the right-sided progression of the equation above. In the next section, we will see protons in action in a completely different way (together with membranes, channels and pumps), for the very purpose of synthesizing ATP.

1.4.2   Bilayer Membranes and their insulating properties

Cells are packaged with, bounded by, and interact with their surroundings through a 'skin' known as a 'plasma membrane' or a 'bilayer membrane'. These membranes are essential to the cell, serving not only the purpose of containing and protecting the cell's contents, but also allowing the cells to 'breathe' by virtue of molecular devices like Ion Channels and Ion Pumps that are embedded into the cellular membranes. These molecular devices are gatekeepers responsible for the transfer of specific ions, including protons, in and out of the cell.

These devices depend on the insulating, proton impervious and ion impervious properties of the membrane bilayer. After all, ion gatekeepers only make sense if the ions are unable to penetrate anywhere else. Now we will take a look at how these membranes play an important role in the function of cells.

1.4.2.1   Proton Gradients across Mitochondria Membranes

Eukaryotes (simple animal cells) contain organelles called mitochondria. Mitochondria are responsible for the final stages of food metabolization, the conversion of chemical energy into ATP (Adenosine triphosphate) molecules.

mitochondria

outer membrane

inner membrane

1um

10um

Eukaryote (simple animal cell)
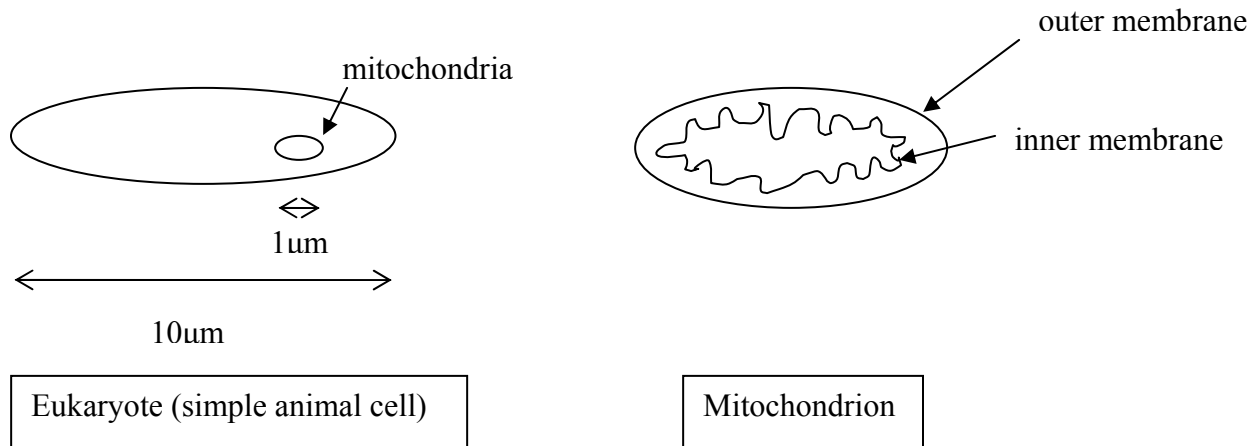
Mitochondrion

ATP molecules are the fuel cells for many molecular devices. The formation of ATP is therefore very important, because living organisms need a lot of it all of the time. Proton gradients and proton transfer across mitochondrion membranes play an important role in ATP generation in animals.

Protons

ATP Synthase

Inner mitochondrion membrane

$ADP + H^+ + phosphate + Energy \longrightarrow ATP$
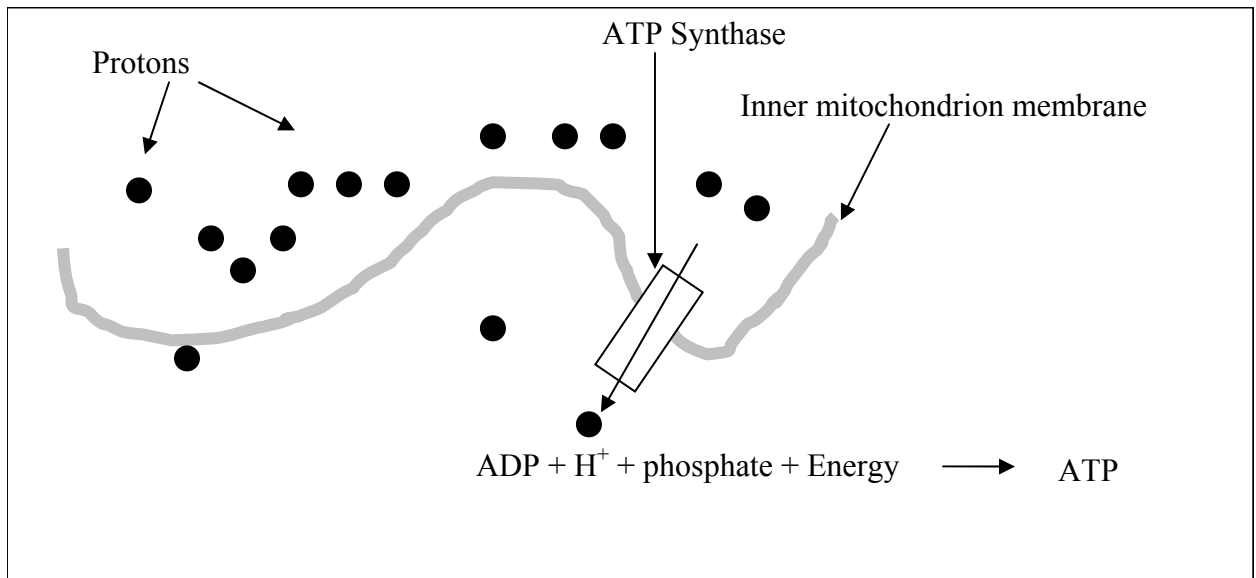
Figure 9: Proton Gradient across Inner Mitochondrion Membrane

Normally, the inner membrane is impervious to protons. Proton pumps maintain a high concentration of protons on the outside of the membrane, relative to the proton concentration

inside the membrane. This allows other membrane motors embedded in the membrane to harness the energy of the proton gradient to do useful work. One such membrane machine that does this is ATP synthase. ATP synthase harnesses energy from the proton movement down the proton gradient and reentering the cell. This energy is used to synthesize ATP (see Figure 9). ATP molecules are the fuel cells for many molecular machines. That is, ATP is the package that contains energy and by which energy is delivered to these devices, according to the reverse reaction shown in Figure 9, i.e. $ATP \rightarrow ADP + H^+ + phosphate + Energy$.

1.4.2.2 Proton Gradients across Chloroplast Membranes of Prokaryotes

Prokaryotes (simple plant cells) contain organelles called chloroplasts. They use the energy from sunlight to maintain a proton gradient across their membrane. ATP is not only the fuel cell for animal cells, but for plant cells as well. In plant cells the ATP synthesis happens in a way very similar to that in the mitochondria of animal cells (discussed in the section above).

1.4.2.3 Proton Gradients across Bacteria and Archaea Membranes

Bacteria also maintain a proton gradient across their membranes, and are able to generate ATP in fashions very similar to mitochondria and chloroplasts (see previous 2 sections).

Archaea are microorganisms that exist in extreme pH, salt concentration, and temperature environments. The ATP production in archaea also depends on a proton gradient across its membranes, and ATP is synthesized in a way familiar to what was discussed above.

1.4.2.4 Other devices that use Proton Gradients across membranes

There are several other well-known machines, embedded in membranes, which utilize the energy of the membrane proton gradient. We will briefly mention just two more such devices.

Lactose permease allows a proton to enter the cell. It uses the energy from this process to bring a sugar molecule into the cell. It does so by allowing the proton to enter such that the proton is attached to a sugar molecule and drags the sugar molecule along with it.

Flagellar motors allow bacteria to swim. They use the proton gradient to generate torque, which turns a helical spindle (flagella) that protrudes outside the bacteria, like a propeller.

1.4.3   Proton Dynamics in Enzyme Catalysis: Serine Protease

Catalysts boost the rate of a chemical reaction. Catalysts that are manufactured by cells are called **enzymes**. Enzymes work by binding to some intermediate state of the substrate, somehow reducing the activation energy barrier of a specific reaction. As with all catalysts, during the cycle of the reaction the enzymes remain chemically unchanged. Enzymes are very specialized with respect to the reaction they catalyze. Therefore there are many different enzymes and categories of enzymes. Proton transport or relay often plays an important role in the function of enzymes. I will use serine protease as an example to demonstrate the very important role of proton transport in a very important function.

Serine Protease catalyzes the breaking of protein peptide bonds in a process known as hydrolysis. Hydrolysis is the process of breaking a bond with assistance from the ions of dissociated of water. Therefore Serine Protease has the effect of breaking long hydrocarbon chains into smaller pieces. This enzyme is important for digestion, blood clotting and suppressing virus invasion. The following series of snapshots[28, 29, 30] demonstrate how serine protease works, and how proton dynamics and hydrolysis play an important part in the process.
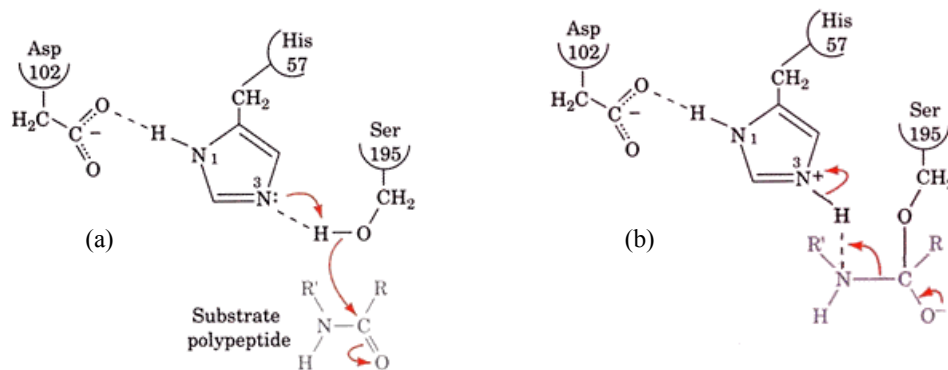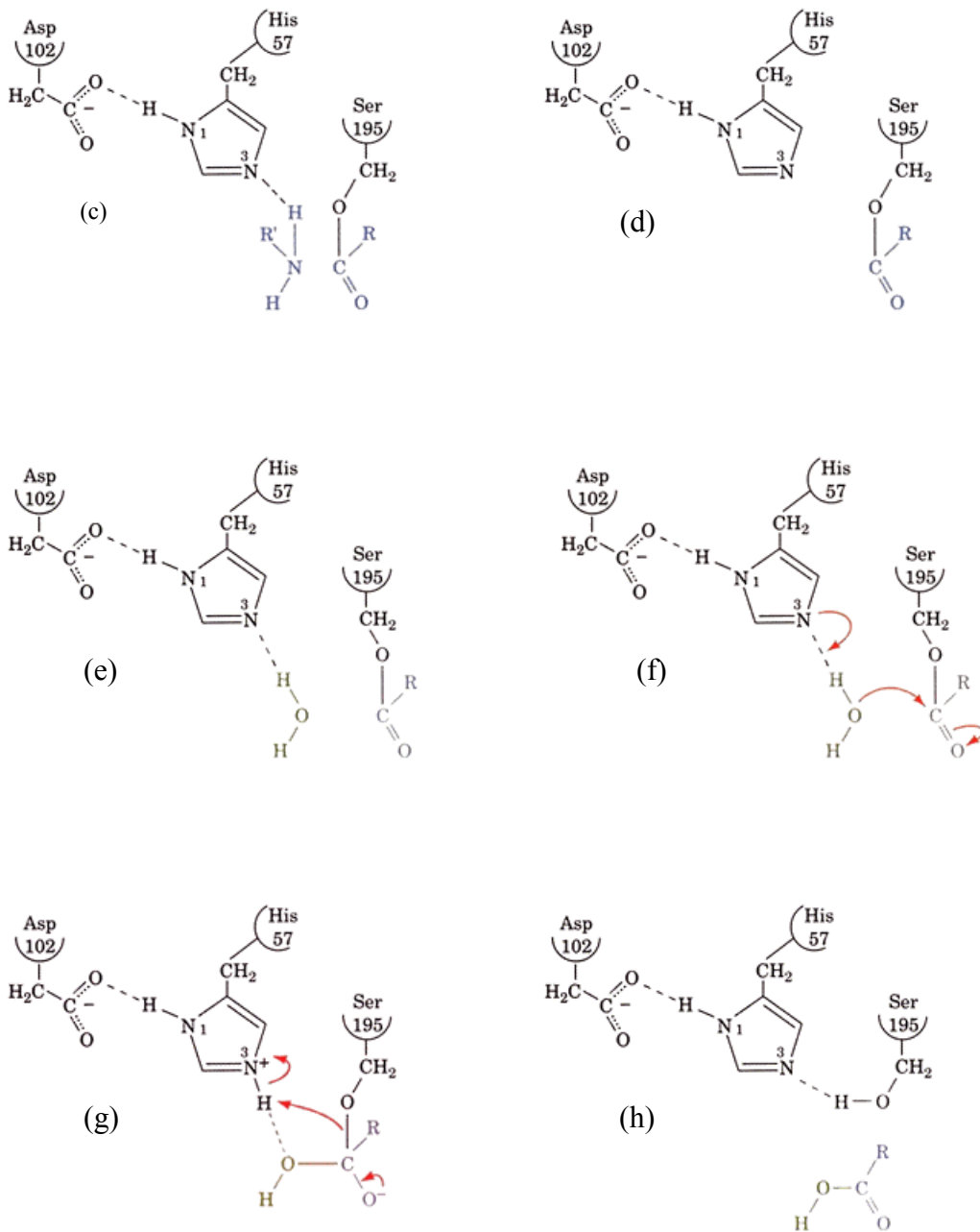
Figure 10: Serine Protease catalysis[28,29,30] (Voet & Voet, 1995) (T. Rose & E. Di Cera, Department of Biochemistry & Molecular Biophysics, Washington University School of Medicine)

Serine protease has an active site that contains three titratable amino acids in an electrostatic network. The amino acids are Asp102, His57, and Ser195, and they work together to cleave polypeptide bonds. The three amino acids are represented at the top of each diagram. The

polypeptide substrate (just one link of it) is represented near the bottom of each diagram. The red arrows show the impending relocation of an electron pair, and hence the impending creation or destruction of a chemical bond. The subsequent diagram shows the result of the rearrangement, and also (again with the red arrows) indicates impending rearrangements for the next step.

In Figure 10(c), the polypeptide bond cleavage has been completed, and in Figure 10(d), the C-terminal fragment (the polypeptide fragment that contained the old C-terminal and the Nitrogen from the Nitrogen end of the cleaved bond) is removed from the picture. In Figure 10(e), a water molecule comes into the picture. In Figure 10(f) to Figure 10(h), the water molecule is used to cap off the new C-terminal end and to return the active site to its original state.

1.4.4   Proton Dynamics in Hemoglobin: The Bohr Effect

The Bohr Effect is the pH, configurational and other environmental dependence of hemoglobin's affinity for oxygen. Protons play an important role in the Bohr Effect, and in the next two sections we will use hemoglobin as an example for the discussion of the role of proton dynamics in allostery and, usually related, the role of proton dynamics in conformational change.

For thirty years, investigators have been trying to properly model hemoglobin. Part of the difficulty is because there is a great deal of controversy about precisely which groups are the pKa-shifted groups responsible for the pH dependence of hemoglobin's oxygen affinity (the Bohr Effect). The discussions that follow do not contribute to that debate, but simply serve to give an introduction to how hemoglobin works, and to emphasize that proton dynamics is an important part of the process.

On a tangential note, we are hopeful that our proton dynamics modeling method, the heart of this dissertation, will in the near future, make a significant contribution toward modeling hemoglobin and the hemoglobin function debate.

1.4.4.1 Proton Dynamics and Feedback Control (Allostery)

Proteins, and macromolecules made up of globular proteins, are able to execute sophisticated and detailed functions as a result of complicated, long-range, interactions between residues. The

function of Hemoglobin is a good example. Hemoglobin is a tetramer (consists of four subunits), and each subunit is a globular protein with an active site that contains an iron molecule that binds an oxygen molecule. The binding of the active sites is coordinated, even though they are on opposite sides of the macromolecule and are separated by about 2.5 nm. The binding to oxygen at one active site predisposes the other active sites to also bind oxygen. This is a feedback control that consists of interactions between active sites that span large distances (allostery), and is key to the sophisticated function of hemoglobin. What follows is a summary of how hemoglobin works, with emphasis on the role that proton dynamics plays.

Muscle activity produces CO_2, which dissolves in the blood to form acid.

$$CO_2 + H_2O \longleftrightarrow H_2CO_3 \longleftrightarrow HCO_3^- + H^+$$
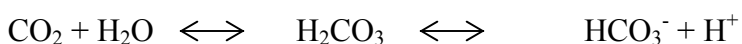
These acidic conditions trigger oxygen-laden hemoglobin to do a series of things. First, protons and CO_2 are bound (the binding site of the CO_2 is a site different from the oxygen binding site). The binding of the protons and the CO_2 triggers the other sites to release their oxygen molecules (the Bohr Effect). One of the ways in which the active sites of hemoglobin are able to communicate subtle information over such large distances, is by conformational changes. These conformational changes are the result of the cooperative effects of many weak interactions. In the next section we will see the role of proton dynamics in the conformational changes in hemoglobin.

1.4.4.2   Proton Dynamics and Conformational Change

Hemoglobin is a good example for examining how a cascade of many subtle effects, including proton movement, can result in conformational changes. In the case of hemoglobin, the effects of these conformational changes can be observed on the macroscopic level. Deoxygenated hemoglobin crystals have a needle like shape, and oxygenated hemoglobin crystals have a plate-like shape[31, 32]. Deoxygenated hemoglobin is blue in color, and oxygenated hemoglobin is red in color. First we will look at the conformational changes that occur to a single subunit upon binding an oxygen molecule. Then, we will look at the coordinated interactions between the subunits that are the keys to hemoglobin's allostery.

Each hemoglobin subunit consists of a Heme group that contains iron. This Heme group, shown in red in Figure 11[33] below, has a "domed shape" as a result of the pressure from the Histidine electron cloud (shown in blue).
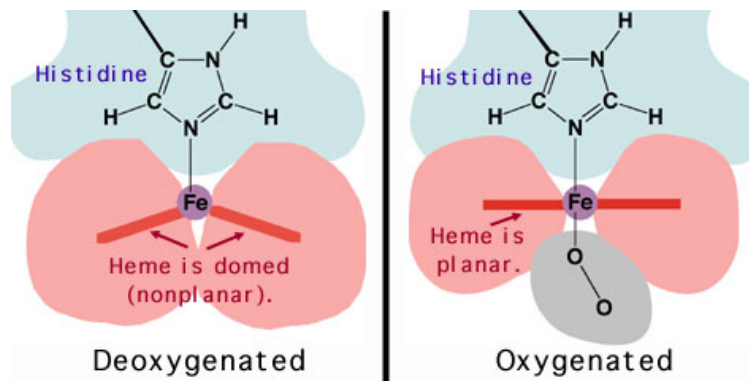
Figure 11: Hemoglobin Subunit[33]

(R. Frey URL: www.chemistry.wustl.edu/~edudev/LabTutorials/Hemoglobin/MetalComplexinBlood.html)

Oxygen molecule (shown as the gray molecule on the right) binding causes the iron to be pulled planar relative to the rest of the heme group. This action also pulls on the histidine, which results in changes at the interface with the other subunits. We will now take a look at what happens at that interface.
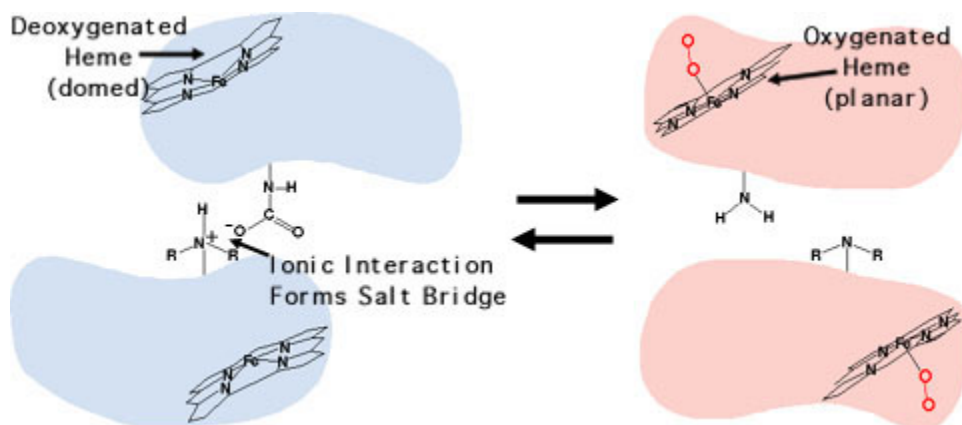
Figure 12: Heme Subunit Interaction[33]

(R. Frey URL: www.chemistry.wustl.edu/~edudev/LabTutorials/Hemoglobin/MetalComplexinBlood.html)

Figure 12 shows the interaction between two subunits. The diagram on the left shows that the deprotonated form of hemoglobin is stabilized by a salt bridge. That is, there are oppositely charged groups in close proximity that attract one another. These charged groups belong to histidine amino acids, and are different from the histidine residues that wedge the heme group as shown in Figure 11. Thus the heme groups are non-planar, and oxygen

binding is not favored. On the right, there is no such salt bridge, and the oxygenated heme groups have a planar shape.

Let's start with the oxygenated form on the right, and summarize how acidic conditions, protons and CO_2 cause the configurational shift to the left, and the consequent release of the oxygen molecule. The oxygenated form on hemoglobin, depicted on the right of Figure 12, exists in a pH 9 environment. In the presence of CO_2 and a pH of around 7 (which is the case in the environment of active muscle), histidines absorb protons and are ionized according to the following reaction.

Histidine Residue   Charged Histidine Residue
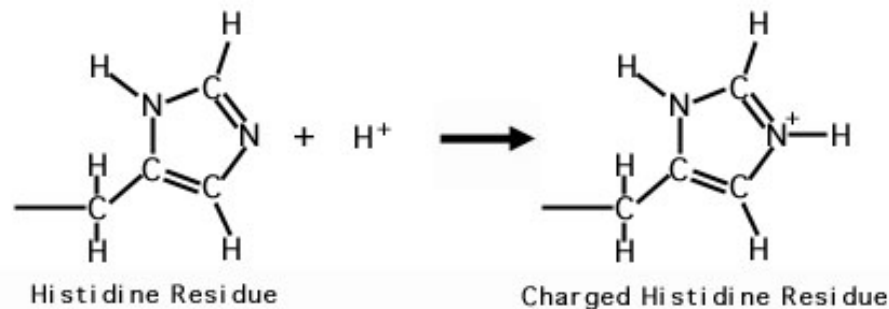
These protonated histidines form the positive side of the salt bridge depicted on the left of Figure 12. The carbon dioxide is absorbed by the amino group of some of the amino acids on the interface.
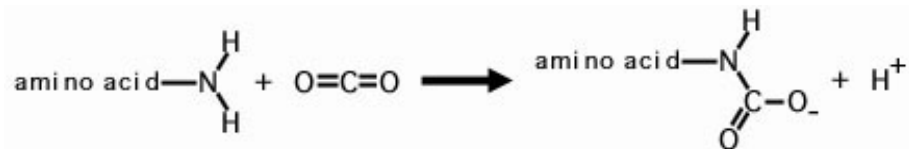
This forms the negative carboxyl group on the complementary side of the salt bridge. This summarizes how CO_2 and protons cause ionization and subsequent salt-bridge formation at the sub-unit interface.

1.4.5   Proton Wires

In section 1.4.1 we talked about bulk water having such interesting behavior that it deserves the designation "biomolecule". In that section we also noted that the behavior of bound water is no less intriguing and important.

Decades ago, X-ray crystallographers had suspected early on that certain bound water molecules play an important part in the structure and therefore function of biomolecules. This suspicion grew more and more convincing as the decades passed and X-ray hardware and software yielded better and better resolution. One side effect of the improving resolution was the ability to "see" bound water more clearly. Another indirect side effect of increasing resolution was the tacit acceptance of bound water as part of the structure of the biomolecule, placing the persistently consistent bound water molecules on the same footing as well resolved amino acids.

1.5   ELECTROSTATICS & PROTONATION STATE OF PROTEINS

1.5.1   Electrodynamics of Biological Systems

This section, section 1.5, focuses on electrostatics of biomolecules. One may ask "instead, why not concern ourselves with the electrodynamics of solvated biomolecules?" Certainly, the electric field of a real biomolecule and a simulated one is dynamic. However $d\vec{E}/dt$ is not big enough to generate a significant $\vec{B}$ field. So for biomolecule simulation models, like MD models, time is discretized and numerical solutions for the electrostatic field are performed at each time step, and these numerical calculations ignore any $\vec{B}$ field electrodynamic effects because the system in each snapshot is considered to be in a quasistatic state.

1.5.2   What is Electrostatics in Biomolecules?

The term electrostatics (as well as the broader term electrodynamics) refers to a classical Maxwellian treatment of electric and magnetic fields. In other words, the electric and magnetic fields are considered to obey Maxwell's equations. However when the term electrostatics is used in the context of a snapshot of a biological system, the term only roughly approximates "all electric field characteristics that fall into the category of classical Maxwellian treatment". In discussions of biomolecules, electrostatics has a slightly narrower definition. I will begin describing what is meant by electrostatics in biomolecules by describing what is not.

The force field that acts within a solvated biomolecule is considered to have three main components: Covalent bond forces, van der Waals forces and Electrostatic forces. The covalent bond forces are those that act on the biomolecule's atoms as a result of their chemical bonding with neighboring atoms. van der Waals forces are attractive at long distances but sharply repulsive at very short distances. The long-range attractive van der Waals forces are known as van der Waals dispersion forces. This force originates because the electron density surrounding an atom is dynamic. So at almost any instant in time, the electron cloud surrounding the atom is asymmetric, causing the atom to appear as a dipole. This will induce complementary dipoles in neighboring atoms because their electron density clouds are also dynamic. And those atoms then go on to induce temporary dipoles in their neighbors, etc. This is how van der Waals dispersion forces can affect an attractive force throughout neutral atoms in a system. This attractive force is responsible for the gas to liquid transition as gasses are cooled. van der Waals repulsion occurs between two atoms that get too close, causing their electron clouds to overlap. As one would imagine, the like charged electron clouds push off violently from each other. This type of repulsion is significant at lower temperatures or densely packed molecules, hence the reason for the incompressibility of liquids and solids.

Technically the van der Waals attractive dispersion force is a Maxwellian phenomenon, but as pertains to a snapshot of a biomolecule, it is not considered as electrostatics in the popular use of the word. Electrostatics in biomolecules is considered to only encompass the electric field affected by atoms or small groups of atoms that are permanently charged (permanent monopoles) or are permanent dipoles. Limiting the term electrostatics to refer to the effects of permanent monopoles and dipoles, and not temporary dipoles came about as a result of visualization from the reference point of the atom by those modeling biomolecules.

From the reference point of an observer, all atoms, with permanent or temporary monopoles or dipoles, affect an electric field that is electrodynamic in nature. But from the reference point of the atom, if the atom has permanent dipoles, its dipole parameters are "*static*" so its contribution to the electric field of the system is considered an "electro*static*" contribution. But if the dipoles of an atom are temporary, then from the reference of that atom, its dipole properties fluctuate over time, i.e. they are "dynamic". Hence those van der Waals dispersion effects don't qualify for "electrostatics", but the others do. A quick note is in order here concerning Molecular Dynamic simulations. Typical molecular dynamic models further enforce

this use of "electrostatics", because the temporary dipole effects are not modeled as being fixed dipoles discretized over time. That is, a temporary dipole is NOT modeled as having fixed dipole values in one snapshot of time and the dipole parameter changes from snapshot to snapshot. Instead, the van der Waals dispersion forces are accounted for by using a Lennard-Jones type potential for short ranges and a continuum treatment for longer ranges.

Here it's also important to note that at the time of writing some well known MD simulation force field packages do have the functionality of representing temporary dipoles as fixed dipoles discretized over time. However this option is in the process of going through the rigors of being tested and checked out by the scientific community. Some of the things that need to be checked are how to alter the van der Waals parameters, how to attenuate the attractive Lennard-Jones terms, or whether they should be done away with altogether, since the temporary dipoles are now being explicitly modeled. This option also uses significantly more computational resources per MD step. This is an exciting development in MD force fields, however at the time of writing it is not an option that is accepted as standard protocol (nothing about our simulation methods prevents the use of this option). What the most detailed MD force fields do is beside the point. The purpose of the above discussion is to explain the history behind the use of "electrostatics" in biological systems.

1.5.3   The importance of Electrostatics in Biomolecules

Electrostatics in biomolecules, defined as discussed above, plays a pivotal role in a wide range of biological processes, from protein folding to protein function. Many effects in biological systems are fundamentally electrostatic in nature. The effects of solvent exposure, pH or proton concentration, ion concentration, solvation shells, salt concentration, protonation state and proton dynamics, and to some extent hydrophobicity are all fundamentally electrostatic in nature (hydrophobicity is largely an entropic effect). As a result, it is hard to find bio-molecular phenomena that are not fundamentally electrostatic in nature. Electrostatics plays an important role in protein folding, specificity, enzyme catalysis and ion channels.

One of the reasons why the electrostatics of permanent monopoles or dipoles of atoms, groups of atoms or molecules plays such an important role in biological systems is because the electrostatic effects are long range effects. This long-range electrostatic effect exists for the

following reason: Recall that biomolecules exist in a solvated environment. The water molecule is dipolar in nature. The electric field strength drops off as $1/r^2$ or $1/r^3$, where r is the distance away from a monopole or dipole. However there is the over-compensating effect that the number of poles within the distance r increases cubically with r. Hence the long-range nature of electrostatics in solvated systems.

The function of the $EcoRI-DNA$ complex and hemoglobin are good examples of the importance of electrostatics. In the presence of a Mg^{2+} ion at a critical position relative to the $EcoRI$, the $EcoRI$ will bind its DNA substrate and dismantle the strands. But if there is no Mg^{2+} ion in that position, the DNA substrate will simply be bound, and not divided. It is believed that the Mg^{2+} ion in that special position causes the deprotonation of several surrounding sites, and the consequent formation of an electrostatic network that facilitates the EcoR1 to perform the task of DNA separation. For a discussion on the role of electrostatics in the function of hemoglobin, please see section 1.4.4.

Having discussed the importance of the electrostatics of permanent monopoles and dipoles, we will now turn our attention to the importance of the titration process, a process that significantly alters the monopole or dipole character of a titratable site!

1.5.4   The importance of the Protonation State

Titration of a site is basically the process of protonation or deprotonation of that site. This process will therefore transfer a proton to or from the site, thereby altering the permanent monopole or dipole characteristics of that site. This can lead to some dramatic changes in function or configuration of the biomolecule. Hemoglobin is a good example of protonation state changes triggering substantial function and configurational changes (see discussion on Hemoglobin, section 1.4.4).

1.6   FACTORS AFFECTING PROTONATION STATE OF PROTEINS

The term protonation state may refer to a single titratable site, or to a whole biomolecule that contains many titratable sites. Applied to one site, it is simply describes the protonation state of that site, and is analogous to a scalar, single valued, quantity. Applied to a biomolecule of many titratable sites, it is best described as a vector, where each element of the vector describes the protonation state of one particular site. Each vector element corresponds to one titratable site of the system, so the protonation state vector contains as many elements as there are titratable sites.

We will see that many factors may go into the protonation state of a given titratable site. Therefore in all but the simplest cases, simply viewing the structure of the protein and assessing the environment at a site is not good enough to determine the protonation state at that site. As a matter of fact even the most sophisticated models for protonation state calculations are challenged in many cases. There is also another layer of complexity that challenges protonation state calculation methods, and that is that for many sites in real biomolecules, the protonation state of a titratable site is not a static thing. So now I will discuss the many factors that go into determining a site's protonation state.

1.6.1   What is pH ?

Pure water dissociates according to $H_2O \rightleftharpoons H^+ + OH^-$ and does so such that its ion concentrations are $[H^+] = 10^{-7}$ and $[OH^-] = 10^{-7}$ moles/liter. The pH of an environment gives us a measure of the proton concentration in that environment and is the most obvious environmental parameter that affects the protonation state of a protein. pH is defined as follows: $pH = -\log_{10} \dfrac{[H^+](moles/litre)}{1(moles/litre)}$.   The argument of the logarithm is a dimensionless ratio.   The numerator is the concentration of active H^+ ions, the denominator is the concentration of active H^+ in some standard state, set to $1(mole/litre)$. So the lower the pH, the higher the proton concentration.   The pH of pure water is therefore 7 and a solution of $pH = 7$ is described as having neutral pH. A solution with pH less than 7 is called acidic. A

solution of pH above 7 is called basic. One way to increase the active proton concentration of an environment is by introducing acid. Acids increase proton concentration by dissociation of their molecules. For example, the carboxyl group $-COOH$ in acetic acid dissociates via $-COOH \rightleftharpoons -COO^- + H^+$. This is the reason why environments that have low pH's are called acidic. One way of decreasing proton concentration (or increasing pH) is to introduce salts that gobble up protons. For example, sodium hydroxide molecules, $NaOH$, readily eat up protons according to the equilibriums $NaOH \rightleftharpoons Na^+ + OH^-$ and $OH^- + H^+ \rightleftharpoons H_2O$

The pH range is typically from 0 to 14. The pH of drinking water is 6.5-8.0. Physiological pH, that is the pH of the environment of most biomolecules, is about 7.4. The pH of human blood is 7.35-7.45. The pH of human stomach contents is 1.0-3.0.

1.6.2 What is a *pKa* ?

$$pKa = pH - \frac{1}{\log_e 10} \bullet \frac{G_{protonated} - G_{deprotonated}}{kT},$$ where $G_{protonated} - G_{deprotonated}$ is the free energy difference between the protonated state and the deprotonated state. k and T are Boltzman's constant and temperature respectively. The protonated state can be stabilized ($G_{protonated} - G_{deprotonated}$ dropped) by decreasing the pH, and the deprotonated state can be stabilized ($G_{protonated} - G_{deprotonated}$ increased) by increasing the pH. The explanation of the meaning of pKa will start by considering the special condition where $G_{protonated} = G_{deprotonated}$, that is making $pKa = pH(G_{protonated} = G_{deprotonated})$.

The most simplistic explanation of titration is that it is the process of gradually changing the pH of an environment for the purpose of affecting a change in the protonation state of a titratable site. So for a system consisting of a single protonated site, if the pH starts from a low value and is gradually increased, at some point of the pH range it will suddenly become deprotonated. The pH at which this happens ($G_{protonated} = G_{deprotonated}$) is called the titration point, or the pKa. Similarly, if a system consisting of a single site is deprotonated, the pH can be gradually changed from high to low until the site changes its state to protonated. The pH at

which that happens will be the *pKa* of that site. *pKa* is therefore a measure of how easy or difficult it is for a site to absorb or lose protons. A site that has a low *pKa* will stay deprotonated in most cases and for most of the *pH* range. It will only become protonated in conditions where the *pH* less than its *pKa*. Similarly a site that has a very high *pKa* will stay protonated in most cases and for most of the *pH* range. It will only become deprotonated in conditions where the *pH* is greater than its *pKa*.

The above description of titration gives a good feel for what a *pKa* is, but is simplistic. Real systems being titrated do not consist of a single titratable site. A typical system will contain many titratable sites, which may be of the same type or may be of different types.   Their protonation state changes with time, even under constant *pH* conditions. So to describe the *pKa* of a real site, we have to introduce the concept of averaging its protonation behavior over a sufficiently long period of time at a fixed *pH*. For systems consisting of only one type of titratable site, at a given *pH*, the time average protonation behavior of one site is equal to the ensemble average protonation behavior of many sites in one snapshot of time. The important question for a site, at any given *pH,* is therefore "for what proportion of the time is this site protonated vs. deprotonated?" In other words "what is the protonation/deprotonation occupancy ratio?"

Therefore if a titratable site is in conditions where the *pH* is less than its *pKa*, its protonation/deprotonation occupancy ratio will be greater than one. If the conditions are such that the *pH* is greater than the *pKa* of the site, then the site's protonation/deprotonation occupancy ratio will be less than one. If the *pH* is equal to the *pKa* of the site, then the protonation/deprotonation occupancy ratio is one. This is how the titration point/ *pKa* of a site is determined. The *pKa* of the site is the *pH* of the conditions such that 50% of the time the site is protonated and 50% of the time the site is deprotonated. This is equivalent to saying that for an ensemble of identical molecules, the *pKa* of a particular site in the molecule is the *pH* of the conditions such that, in one snapshot of time, for 50% of the molecules the site is protonated, and for the other 50% of the molecules, the site is deprotonated. Yet another way of saying the same thing is that the *pKa* of a site is the *pH* at which the protonated state and the deprotonated state have the same free-energy, since a protonation/deprotonation occupancy ratio

of 50/50 simply means that the free energies of the protonated state and the deprotonated state are the same ($G_{protonated} = G_{deprotonated}$).

The above definition of pKa limits the measurement of a pKa to the very specific condition where the protonation/deprotonation occupancy ratio is 50/50. However $pKa's$ can also be described in terms of the relative free energies of the protonated state and deprotonated state, $pKa = pH - \log(e)\dfrac{G_{protonated} - G_{deprotonated}}{kT}$. This definition is useful for pKa calculation methods where $G_{protonated} - G_{deprotonated}$ calculations are possible. Experimental pKa measurements are based on the narrower definition of pKa where ($G_{protonated} = G_{deprotonated}$) so the second term on the right is zero, so $pKa = pH(G_{protonated} = G_{deprotonated})$.

1.6.3 Titratable amino acids

Proteins are assembled from a primary sequence of amino acids. There are 20 different amino acids, and seven of them are titratable. They are Arginine, Aspartic acid, Glutamic acid, Cysteine, Histidine, Lysine and Tyrosine. What follows in Table 1 below is a brief description of them. Note that the *pKas* shown below in Table 1 are for the sites on the side chains with Acetyl and N-Methyl groups capping the amino-acid backbone.

Titratable amino acid	*pKa*	Deprotonated charge	Protonated Charge
Arginine	12.5	Neutral	+1e
Lysine	10.2	Neutral	+1e
Histidine	9.2	Neutral	+1e
Aspartic acid	3.9	-1e	Neutral
Glutamic acid	4.1	-1e	Neutral
Tyrosine	10.1	-1e	Neutral
Cysteine	8.3	-1e	Neutral

Table 1: Titratable Amino Acids

1.6.4   Free energy components that contribute towards *pKa* values

We know what the *pKa* values are for all of the titratable amino acids; they have been determined experimentally, and their values are listed in Table 1 above. Note that these *pKa* values only hold for one condition: These are the *pKa* values of the residues when they are isolated in solvent. In other words, any one of the titratable residues would have the listed *pKas* only if it was the only residue of the biomolecule in solution. It is "isolated" meaning that it does not interact with any other titratable sites, does not interact with any other biomolecule or any other residue of the biomolecule.   It is therefore completely solvent exposed and is subject to no hydrophobic effects.   From here on I will therefore describe the experimentally determined *pKas* listed above as *pKas* for isolated residues or "isolated *pKas*".

However if any of the titratable residues was part of a folded protein or biomolecule, the *pKa* it exhibits may be different from the isolated *pKa*. Such a shift away from the isolated *pKa* is as a result of its environment. That is, the residue interacts with other titratable sites, other residues of the biomolecule, or other biomolecules.

1.6.4.1 *pKa* components invariant with environmental changes

Consider the isolated *pKas* for all of the titratable amino acids given in Table 1. All of these titratable amino acids have their isolated *pKas* experimentally measured under identical conditions. Yet the titratable amino acids have isolated *pKas* that almost occupy the full *pH* range. The reason for this is because even though the solvent exposure of the titratable groups is the same between the different amino acids, there are a few other factors responsible for why the isolated *pKas* range from 12.5 to 3.9. These factors have to do with the nature of the amino acid side chain that the titratable PROTON is connected to.   These factors I describe as "intrinsic" factors, and will be the subject of discussion in this section.

Recall that a *pKa* is related to the protonation-deprotonation free energy difference. It is a measure of how hard it is to add or remove a proton from a titratable site. Adding or removing a proton from a titratable site involves making or breaking the covalent bond that binds the proton. A strong bond to the proton will contribute towards a higher *pKa* , and a weak bond to

the proton will contribute towards a lower *pKa*. This effect can be seen if we compare the titratable amino acids that have isolated *pKas* at the approximate extremes of the *pKa* range, Aspartic acid (*pKa*=3.9) and Lysine (*pKa*=10.8). In Aspartic acid, the titratable proton is connected to an oxygen atom, but in Lysine, the titratable proton is connected to a nitrogen atom. Oxygen and nitrogen have atomic numbers 8 and 7 respectively, which means that there are 6 and 5 electrons in their outer orbitals respectively. Oxygen is much more electronegative than nitrogen, because it only needs 2 electrons to complete its outer orbital (8 electrons required to complete the outer *d* orbital) as opposed to nitrogen which is three electrons short of filling the *d* electron orbital. At neutral *pH* (*pH*=7), the strongly electronegative oxygen at the titration site of the protonated Aspartic acid (represented on the left of the diagram below) will easily strip the lone electron from the titratable Hydrogen in order to complete its complement of *d* shell electrons. At a *pH* of 7, the deprotonated state is more stable. This relative stability of the deprotonated state (or instability of the protonated state) is the reason for the low *pKa* of Aspartic acid.

Note the one-way arrow.
Dissociation at pH = 7
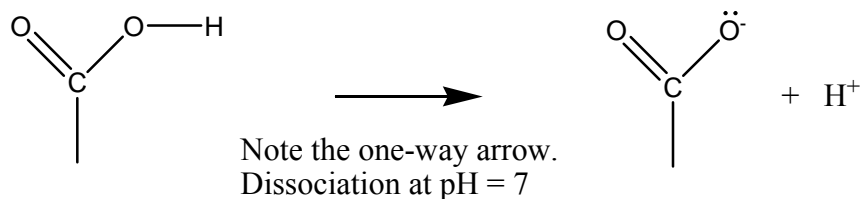
However the nitrogen in Lysine's titratable site is not as electronegative, so at a *pH* of 7, it cannot do as the oxygen in Aspartic acid's titratable site. Protonated Lysine, represented below on the left, cannot strip the electron away from the hydrogen leaving a proton. Notice the direction of the dissociation arrow.

Protonation at pH = 7
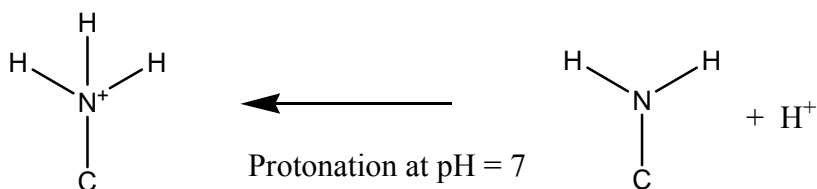
This means that for Lysine, at a pH of 7, the free energy of the protonated state is lower than that of the deprotonated state, hence it's high pKa.

We have just discussed that the nature of the chemical bond between the titratable proton and its titration site is the major factor contributing to the isolated pKa values. Notice that the titratable sites in Aspartic acid and Glutamic acid are identical. We would therefore expect that the isolated pKa for Aspartic acid and for Glutamic acid to be very close, and indeed they are ($pKas$ of 3.9 and 4.1 respectively).

There is another, much less influential factor that affects isolated pKa values. This factor is NOT invariant with environmental changes, but it does play a part in the isolated pKa value, so it will be briefly alluded to here, and spoken of in more detail in following sections. The charged protonation state of a residue affects an attractive polarization field with the surrounding water solvent. This effect will contribute towards making the charged state more stable, regardless of whether that charged state is a deprotonated state (as is the case with Asp, Glu, Cys, or Tyr) or whether that charged state is a protonated state (as is the case with His, Lys, Arg). Therefore this factor contributes slightly towards dropping the values of the isolated $pKas$ for Asp, Glu, Cys and Tyr. This factor correspondingly contributes slightly towards raising the isolated pKa values for His, Lys and Arg. If these residues were in hydrophobic cores instead of being isolated and completely solvent exposed, this effect will have a completely reversed effect on the $pKas$. This hydrophobic environment effect will be discussed more in sections 1.6.4.2 and 1.6.7.

1.6.4.1.1 Aspartic and Glutamic Acid comparisons

Aspartic acid and Glutamic acid have a very similar structure, differing only in that Glutamic acid's side chain is one CH_2 group longer.
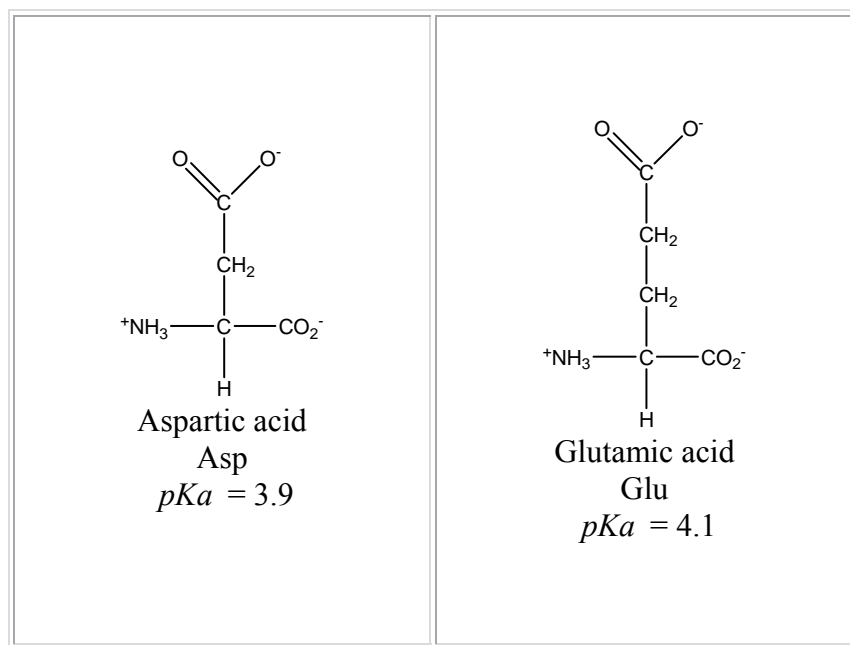
Figure 13: Asp and Glu Comparison[26]

Note that in Figure 13 above, the Acetyl and N-Methyl blocking groups at the ends of the amino-acid backbone fragment are not shown and that the indicated *pKas* are for the amino-acids with these blocking groups. With regards to the Aspartic and Glutamic acid comparison, note that although the titratable regions are identical ($-COOH \rightleftharpoons -COO^- + H^+$), their *pKa* values differ by a few tenths. This shows that the composition of the rest of the side chain does also contribute slightly to the isolated *pKa* value. The specific reasons for the Asp and Glu *pKa* differences are complicated and include orientation of the side chain with respect to the backbone and the hydration effect changes due to the additional $-CH_2-$ of Glu.

We can therefore summarize the contributions towards the isolated *pKa* values, or the relative stabilities of the protonation states in the isolated conditions. In order of influence, they are the chemical composition of the immediate titration region, the charge/neutrality solvation effects of the protonation state, and the chemical composition of the rest of the side chain. The first and last factors, the chemical composition of the immediate titration region and the chemical composition of the rest of the side chain, are the factors that are invariant with environmental change. They involve chemical electron cloud interactions. These factors are invariant because when the titratable amino acids are part of a biomolecule and interact with other parts of a biomolecule, these factors stay the same.

The second factor, the charge/neutrality solvation effects of the protonation state, is an environmental factor.   It tends to move the *pKa* in one direction in solvent exposed environment, but tends to shift the *pKa* in the opposite direction in a hydrophobic environment. We will discuss these and other environmental effects in the next section.

1.6.4.2 *pKa* variation with environmental changes

Titratable amino acids will only exhibit the *pKa* values listed above if they are isolated. However when titratable amino acids are in real in-vitrio biomolecules, the titratable sites interact with other parts of the biomolecule, other titratable sites or parts of other biomolecules. These effects are what I describe as "environmental" effects that cause the *pKa* values of a titratable site to shift away from its isolated *pKa*.   Note that these environmental effects can all be described as electrostatic in nature.   They do not involve any chemical electron cloud interactions as did the invariant factors.

1.6.4.2.1 Solvent Exposed environments

The charge or neutrality of a titratable site affects the *pKa* in ways that depend on the environment.   If a site is solvent exposed, the charged version will effect polarizing of surrounding solvent, which helps to make that charged state more stable, regardless of whether that charged state is a deprotonated state (as is the case with Asp, Glu, Cys, or Tyr) or whether that charged state is a protonated state (as is the case with His, Lys, Arg).   Therefore this factor contributes slightly towards dropping the values of the *pKas* for Asp, Glu, Cys and Tyr.   This factor correspondingly contributes slightly towards raising the *pKa* values for His, Lys and Arg. It is important to note here that the *pKa* shifts just described are not shifts from the isolated *pKas*, because the isolated *pKa* value includes this effect.   Recall that this effect of a charged titratable site polarizing surrounding water is also present in the isolated *pKa* conditions.

1.6.4.2.2 Hydrophobic environment

If these residues were in hydrophobic cores instead of being isolated and completely solvent exposed, there will be a completely reversed effect on the *pKas*, because a buried hydrophobic environment will help make the neutral state significantly more favorable, regardless of whether

that neutral state is a protonated state (as is the case with Asp, Glu, Cys, or Tyr) or whether that neutral state is a deprotonated state (as is the case with His, Lys, Arg). Therefore this factor contributes significantly towards raising the values of the $pKas$ for Asp, Glu, Cys and Tyr. This factor correspondingly contributes significantly towards dropping the pKa values for His, Lys and Arg. It is important to note here, in contrast to the solvent exposed case, that the pKa shifts just described are shifts from the isolated $pKas$ and are often quite significant. Recall that isolated pKa conditions are completely solvent exposed conditions.

1.6.4.2.3 Charged environment

Finally we discuss the most obvious factor that affects the pKa of a site. That is, the electrostatic effects of nearby charges as would occur with charged ions or charged amino acids in the immediate surroundings. In a negatively charged environment, the positive/neutral titratable sites will have their $pKas$ shifted up, while the negative/neutral titratable sites will have their $pKas$ shifted down. In a positively charged environment, the positive/neutral titratable sites will have their $pKas$ shifted down, while the negative/neutral titratable sites will have their $pKas$ shifted upwards.

1.6.5 Effects of pH on protonation state

The pH of an environment, a measure of the concentration of protons in the environment, is the environmental factor that has the most direct effect on the protonation state of a biomolecule. The titratable sites most affected by the environment's pH are those titratable sites that are on the surface of the biomolecule, because they are in direct contact with the solvent. This is because the pH of the environment specifically means the pH of the solvent that surrounds the biomolecule. A lower pH will tend to protonate sites, and a higher pH will tend to deprotonate sites.

Protonation changes due to pH changes will change the net charge of the biomolecule. If the pH drops and sites pick up protons, the net charge will change in the positive direction. If the pH increases and causes titratable sites to lose protons, the net charge will change for the

negative direction. These isoelectric changes can be far reaching, going way beyond affecting the protonation states of surface titratable groups and affecting fundamental changes in the structure and performance of the biomolecule. Hemoglobin is a very good and well-studied example of precisely this effect (see section 1.4.4). In summary, under the conditions at the lungs, it binds oxygen and releases carbon dioxide. The hemoglobin then travels to the muscles via the blood stream. Under the more acidic conditions of the muscles, it releases the oxygen and binds the carbon dioxide, which in then transports back to the lungs, and the cycle continues.

1.6.6 Solvent exposed titratable sites

Solvent exposed titratable sites generally exhibit a pKa close its isolated pKa. However the pKa of solvent exposed site will shift away from the isolated pKa in the presence of other nearby charged groups, such as solvent ions. One way in which ion presence can affect pKa shifts is if the ion concentration is different from the ion concentration of the isolated pKa measurement conditions. Another way that ion presence can affect pKa shifts is if an ion assumes a particular position with respect to a biomolecule, in such a way that it is an important part of the function and structure of the biomolecule. The Mg^{2+} ion of $EcoRI$ is a good example of this. In the presence of a Mg^{2+} ion at a critical position relative to the $EcoRI$, the $EcoRI$ will bind the DNA substrate and dismantle the strands. But if there is no Mg^{2+} ion in that position, the DNA substrate will simply be bound, and not divided. It is believed that the Mg^{2+} ion in that special position causes the deprotonation of several surrounding sites, and the consequent formation of an electrostatic network that facilitates the $EcoRI$ in performing the task of DNA separation[34].

1.6.7 Deeply buried titratable sites

Titratable sites that are deeply buried in a purely hydrophobic core will generally assume a neutral protonation state. So acidic residues like Aspartic acid, which are usually deprotonated and negatively charged when solvent exposed, will have their $pKa's$ shifted upwards and

become neutral and protonated. Similarly basic sites like Lysine, which are usually protonated and positively charged when solvent exposed, will have their $pKa's$ shifted downwards and become neutral and deprotonated.

However "deeply buried" is not a quantitative term. Neither is "purely hydrophobic core". The reality is that the ability of a hydrophobic environment to force neutrality on a usually charged titratable site depends on the distance of the titratable site from bulk solvent (i.e., a measure of how "deeply buried"). It also depends on the composition of the hydrophobic core. The following section will discuss occurrences of sites that are both charged and buried

1.6.8   Sites that are charged and buried

The following sections will discuss the energetics of various scenarios that stabilize titratable sites that are both charged and buried.

1.6.8.1 Salt bridge

One way in which a buried titratable site can maintain its charge is if it interacts with another buried titratable site of complementary charge that is close enough. This type of interaction is described as a salt-bridge, because it mimics the oppositely charged attraction of ions in a salt molecule. Salt-bridge formation is an important part of hemoglobin function, and this is discussed in section 1.4.4.

1.6.8.2 Electrostatic networks

The salt-bridge, described above, can be considered the most basic form of charge network. Charge networks may involve more than half a dozen buried titratable sites. Their stability is often the result of a delicate, subtle and complex electrostatic balance. Such networks are often found in the active sites of biomolecules. See Serine Protease discussion (section 1.4.3) for an example of a relatively simple electrostatic network.

1.6.8.3 Local configuration fluctuations

Electrostatic networks may be dynamic in character because they may be correlated with configurational dynamics. The protonation state of sites in electrostatic networks of mobile regions of a biomolecule is therefore expected to change with time and configuration. This protonation state-configuration dynamics correlation may play a critical role in the function of the biomolecule, as it does in hemoglobin, section 1.4.4.

1.7   EXPERIMENTAL TOOLS FOR INVESTIGATING PROTEINS

1.7.1   Structural Methods

The structures of many biomolecules have been determined by X-Ray crystallography. The sample preparation starts with the biomolecule of interest dissolved in a solution of buffers. If the conditions of pH and salt concentration are right, crystals of the biomolecule begin to grow over a period of weeks. These crystals are then flash-frozen, mounted on a rotating stage and an X-Ray beam is shot through the crystal. A detector catches the x-rays that were scattered by the crystal, and analysis of this scattered radiation is used to construct the structure of the biomolecule. The major bottleneck with the throughput of this technique is the process of growing crystals. There is no definitive way of knowing beforehand the right conditions for crystallization. X-Ray crystallography can capture the positions of the heavy atoms of the system (typically all atoms bigger than Hydrogen), including the oxygen atoms of bound water, provided they do not move too much. Just like a long exposure picture taken on film, the heavy atoms that are very mobile, such as those of the bulk solvent, are not resolved. But the heavy atoms of the biomolecule and the bound water are usually well resolved.

Nuclear Magnetic Resonance (NMR) imaging is another tool for structure determination, but its usefulness is not limited to biology. A sample of solution containing the biomolecule of interest is subjected to a strong modulating magnetic field. This induces spin of the nuclei of the atoms of the sample. This spin in turn induces a reaction field that interacts and modifies the

original magnetic field. Sensors can measure these modifications, and that information is then used to determine the structure of the biomolecule. NMR also yields limited dynamic information. The NMR technique is usually used to solve smaller structures. However development of NMR theory, methods and implementation are allowing NMR to be used to solve the structure of larger and larger biomolecules. Unlike X-Ray crystallography, the protein does not have to be crystallized, so the preparation process for the sample is not as involved. This is a significant advantage because crystallization is somewhat of an art and not yet a science, and finding the crystallization conditions for a never before crystallized protein is no guarantee.

Knowing the position of bound water is important because it allows insight into water penetration. As we have seen, this is an important aspect of the structure because of its relation to proton dynamics. Both X-Ray crystallography and NMR allow investigators to see bound water.

These structural methods discussed are invaluable because they provide the starting configuration for computational simulation. Because the protein-folding problem is not completely solved, it cannot be simulated. Therefore there is no computational way to start with a primary structure and derive a sufficiently accurate protein structure. Computational simulation is therefore indebted to structural methods like X-ray crystallography and NMR imaging to provide the structure of the biomolecule, from which the simulation may start.

1.7.2 Experimental, Thermodynamic and Other "Wet Lab" Methods

There are several laboratory methods such as titration, reaction rate control methods and calorimetric methods that yield valuable information about the *pKas* and thermodynamics of a biomolecule. It is these results that serve as a benchmark for our computational thermodynamic results.

1.8 COMPUTATIONAL BIOPHYSICS: THEORY OR EXPERIMENT?

Traditionally, all computational investigations are considered to be theoretical. What we do certainly qualifies as such. We are building a model, and testing the computational results against experimental results. We are therefore seeking to validate our "theory", which in our case is our computational model.

However there is also a lot of experimental flavor to this work. Figure 20 and the subsequent discussion, gives a feel for how central *numerical analysis via computer experiment of various models* is to computational molecular biophysics. Apart from simply validating our model, we can go further to use our computer model as a tool for investigation. For example, we hope to apply our model to understand specificity between protein and DNA: to break down the protein and DNA binding into components. This is something that can't be done in laboratory experiments, but can be done in computer experiments.

1.9 SURVEY OF COMPUTATIONAL RESOURCE EVOLUTION

The growing prowess of computational resources has been an indispensable catalyst for applying atomic detailed molecular dynamics to the computational investigation of a broader and broader range of biological phenomena. A tide of investigators are pressing hard for either increasing force field accuracy, increasing system size, or increasing simulation length. The gate restraining them is computational power, even though that gate has yielded a lot of ground. This is because the yielded territory has been so fruitful, and the promise of further territory so alluring, that the appetite has only been wetted instead of being satisfied.

It is therefore fitting that I devote a few sections to computational resource evolution, pointing out correlations with the feasibility of more accurate models, larger systems and longer simulations. As a true node-hour consumer, I have a duty to add to the din of demand for more and faster computational resources, so these sections also serve to fulfill that duty.

48

1.9.1　Hardware improvements

In terms of computer hardware improvements over time, the most dramatic is processor clock speed. The figure below shows the clock speed improvements for the Alpha processor over ten years. The Alpha processor, for many years, was considered the highest performing 64-bit high-end computing processor. Many supercomputers running today have Alpha processors at their core. This is the case with our workhorse, the PSC's Lemieux, which is also the workhorse of dozens of other account holders. Production of Alpha processors stopped in late 2004 because of insufficient volume of sale, due to the Alpha lineage not migrating to the high volume desktop market.
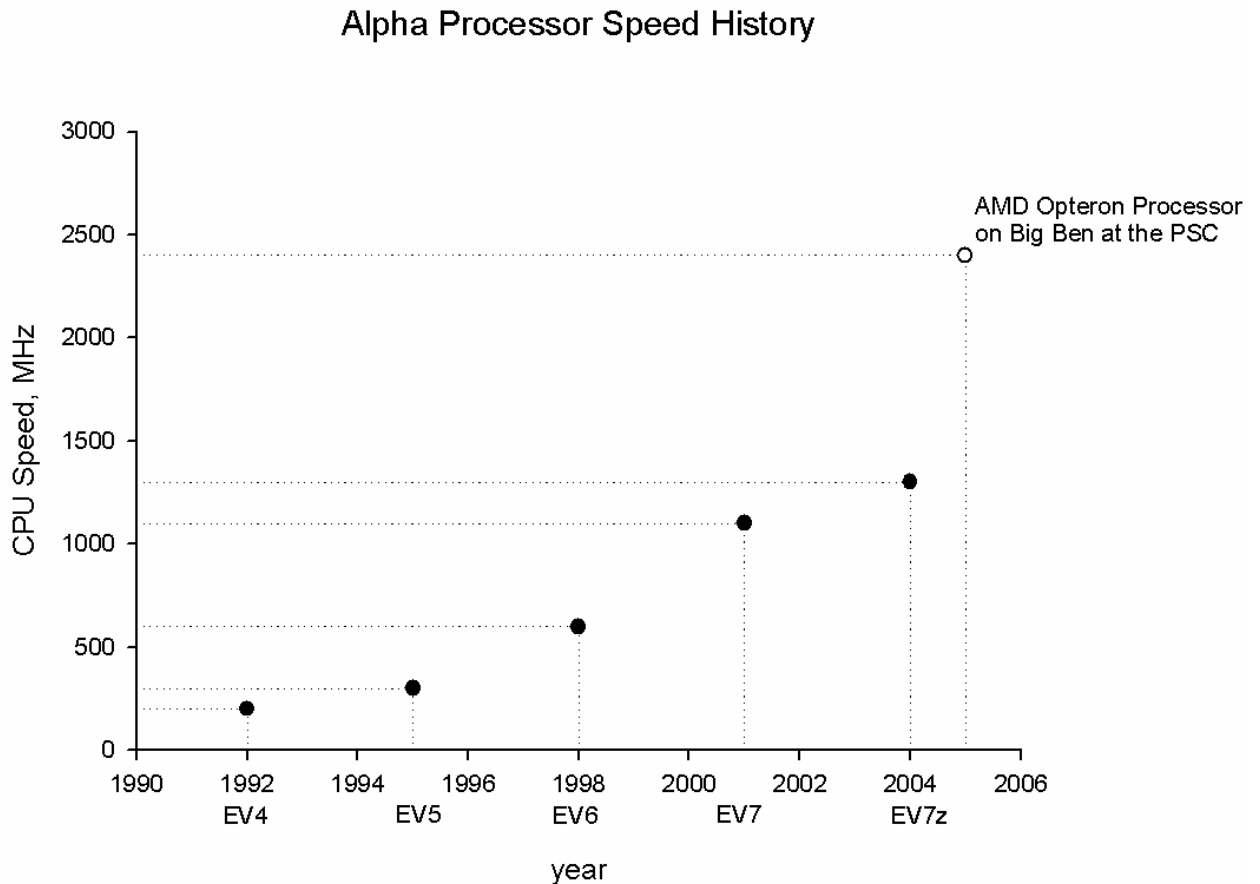
Figure 14: Improvements in processor performance

The solid dots show Alpha processor speed through the years[35,36]. The hollow dot on the far right refers to the AMD Opteron processors in Big Ben, PSC's newest supercomputer. The Opteron processor is a 64-bit processor that is available for desktop machines. The high volume desktop market served as a good foundation for the Opteron's entry into the supercomputing world.

Computational performance cannot be described in terms of processor speed alone. There are many other factors that affect microprocessor performance, like architecture, memory bandwidth and memory latency. However the above plot comes close to conveying the microprocessor technology contribution to the quickly expanding the barriers of computational performance. The processor speed increase is about 30% per year, and processor performance increase (considering architecture, memory etc, as well as processor speed) is about 40% per year[36].

Increased throughput of MD simulations is as much or more, a function of inter-node communication, as it is a function of single node performance. More effort is put into designing inter-node communication architecture than any other aspect in supercomputers. Even in in-house Beowulf type clusters, the inter-node communication hardware is usually more expensive than all the nodes combined. As with processor performance, there is more that one metric to describe inter-node communication performance. The two most important are latency and bandwidth. I have chosen inter-node latency as the most demonstrative metric of evolving inter-node communication performance.

Inter-Node Latency
linear plot

Ethernet	500 microseconds
Intel Gigabit Ethernet	170 microseconds
(1995) Myricom Myrinet	33 microseconds
(2000) Quadrics Interconnect, Lemieux	5 microseconds
(2005) Cray SeaStar Interconnect, BigBen	<3 microseconds
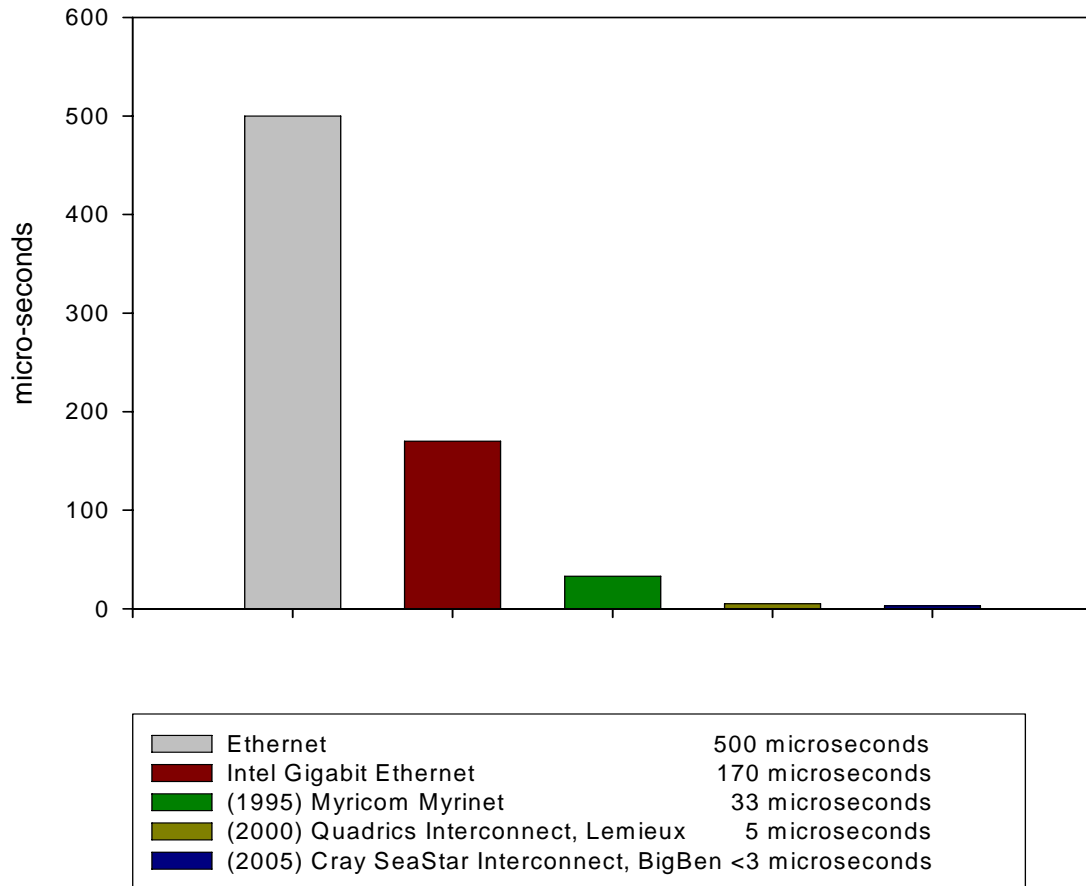
Figure 15. Latencies for an assortment of inter-node communication systems: Linear Plot
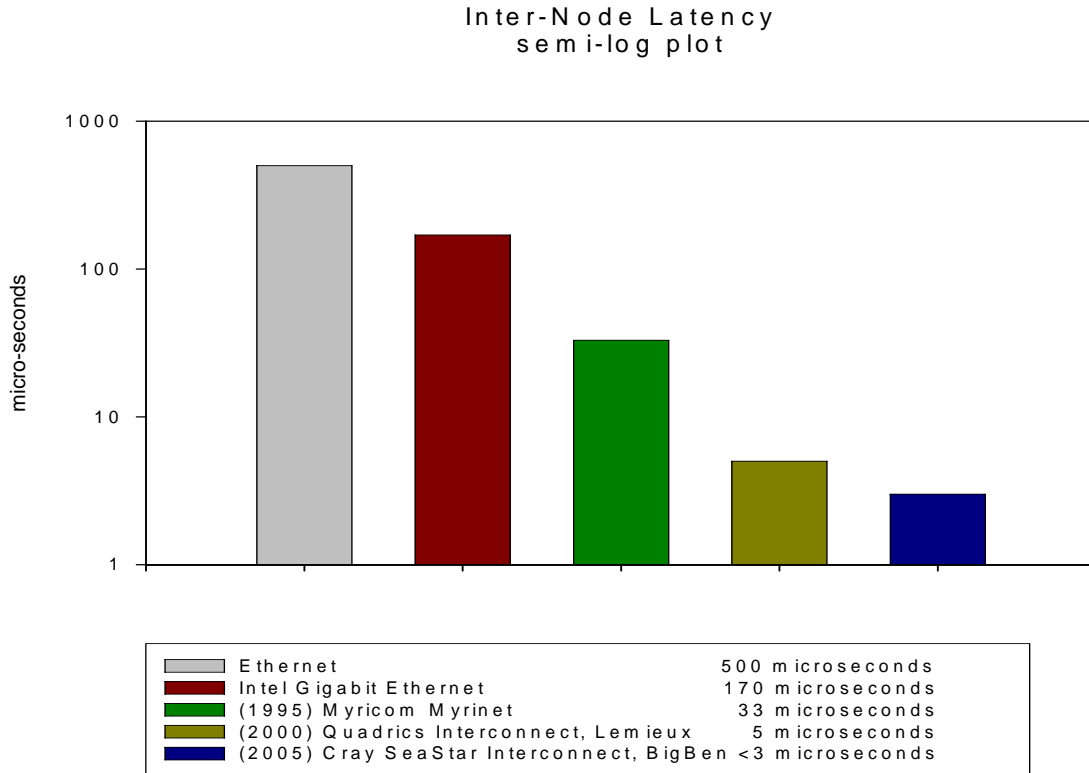
Figure 16. Latencies for an assortment of inter-node communication systems: Semi-log plot

Latency is the time required for a zero byte (or very small) message to travel from one node to the next. The computer network industry is built on the TCP communication protocol, and this is the protocol used by Ethernet switches, such as the Ethernet and the "Intel Gigabit Ethernet" switches shown in Figure 15 and Figure 16 above. These systems were not designed for parallel computing. So the overhead to pass messages (latency) is too much, although the bandwidth is acceptable. As a result, multi-node machines with these types of switches, (like Beowulf clusters with Ethernet switches), do not scale well past four nodes. Intel Gigabit Ethernet switches are worth special mention. Even though they use the TCP protocol, they are considerably faster (lower latency) than machines of that class. Beowulf clusters with these types of switches will yield decent scaling up to six nodes.

On the RHS of Figure 15 and Figure 16, three high performance inter-node communication systems for massively parallel processing are mentioned. They are Myrinet, Quadrics and SeaStar interconnects. There is one other recent high performing system that deserves mention, and that is the Infiniband systems sold by Mellinox Technologies[37,38]. These systems use customized communication protocols, which allow them to have low latencies and

high bandwidth. Unlike the TCP protocols, these protocols are executed by separate processors that are part of the interconnect system. That way, the CPU does minimal communication work, freeing up the CPU for job related processing.

The Myrinet systems (by Myricom) are very popular, highly portable, and can be purchased and built in a modular manner.   These are the systems of choice for investigators that want to build their own in-house highly scalable clusters. These systems can connect hundreds of nodes.   The marketing for Quadrics and Infiniband interconnects was initially aimed at massively parallel computing centers, however both systems now compete with Myrinet for the smaller cluster market as well. The cost of these systems (Infiniband, Myrinet and Quadrics) is about $1500.00 per node. The SeaStar interconnect is a Cray development for use in their super-computers.

1.9.2　　　　　Code improvements

Single processor performance
comparison for
11 years of Amber code
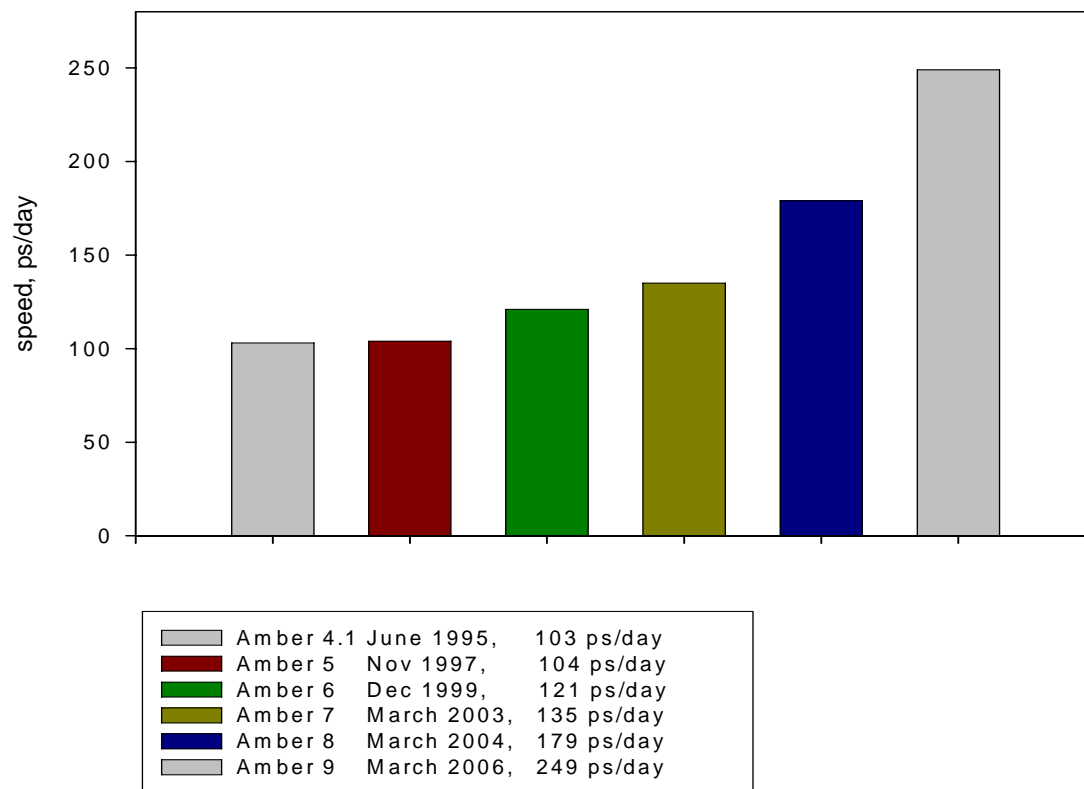
Figure 17: Code improvement as relates to single processor runs

1.9.3   Considering all improvements

Amber throughput improvement over 7 years

Legend:
- Amber6, 1999,   346ps/day,   32procs
- Amber7, 2003, 2160ps/day, 16procs
- Amber8, 2004, 6288ps/day, 256procs
- Amber9, 2006, 15428ps/day, 256procs
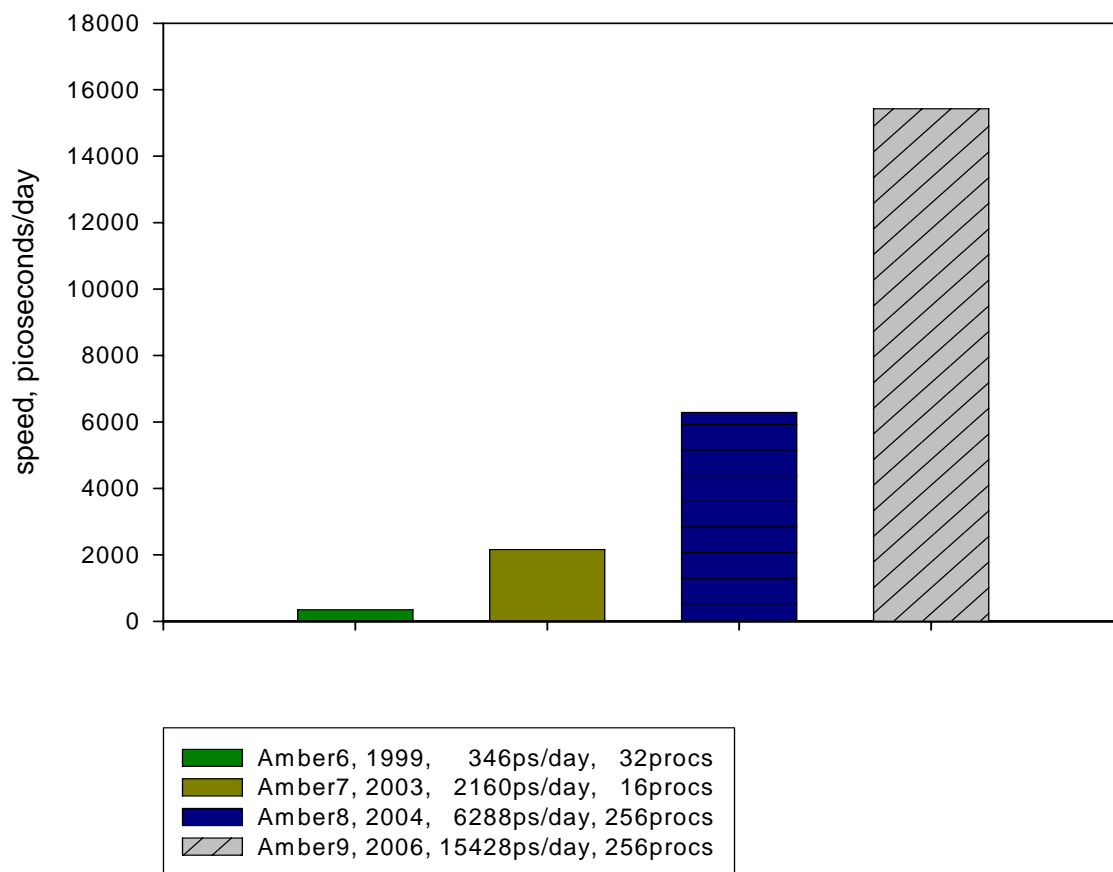
Figure 18: MD throughput improvements

What about interconnect hardware improvements and code parallelization improvements? The plot above takes everything into consideration. Figure 18 above compares the throughput performance of the Amber code through the years on various platforms. This plot considers all improvements: processor speed, memory bandwidth, inter-node communication, algorithm and compiler improvements. Even though the Figure 18 plot above compares code performance on different platforms, the comparison is appropriate because it captures the sum effect of all the hardware and software improvements over the years.

The throughput measurements were calculated based on timings for the "jac" (Joint Amber/Charrm DHFR) benchmark. This is the protein DHFR, solvated with TIP3 water, in a

periodic box. There are 23,558 total atoms, and PME used with a direct space cutoff of 9 Å. A system of 23k atoms is a relatively small system by today's standards.

The most obvious conclusion is that the throughput has improved by almost 2 orders of magnitude over the span of 7 years. Another clear conclusion is that the Amber code scales better by about one order of magnitude. Comparison of Figure 17 and Figure 18 show that parallel processing related hardware and software improvements account for the bulk of the throughput improvement. Single processor performance improves by only a factor of 2, but the other factor of about 25 (the total throughput improvement factor is about 50) comes from parallel processing related hardware and software improvements.

Amber6 benchmark was performed on an Origin 2000 R10000, 250MHz machine, 64procs.[39, 40].

Amber7 benchmark was performed on an SGI Altix, 1500MHz machine, 16procs.[41].

Amber8 benchmark was performed on an IBM Power4 P655+, 1500MHz, 256procs [42].

Amber9 benchmark was performed on an IBM P655+, 1700MHz, 256procs[43].

1.10 MODELING BIOMOLECULE ENERGETICS

1.10.1   Implicit solvent Poisson-Boltzmann type models

In implicit solvent models, the water is modeled as a macroscopic entity to which is assigned a large dielectric constant. The solute is typically modeled microscopically and is also is assigned a dielectric constant, much smaller that that of the solvent. The energetics of the system is calculated by a Poisson-Boltzmann (PB) type calculation, which is derived from Gauss's law. Gauss's law and the PB treatment relate the divergence of the electric field to the charge density distribution. Gauss's Law is $-\nabla \cdot \varepsilon \nabla \phi = \rho$, where ϕ, ε and ρ are the electrostatic potential, the electrostatic permittivity and charge density respectively. The full Poisson-Boltzmann equation is[44,45]

$$-\nabla \cdot \varepsilon \nabla \phi = \rho + \sum_{i} q_i n_i e^{q_i \phi / kT}$$

In the PB treatment, the charge density of the solvent salt is described by a Boltzmann distribution (the second term on the right). q_i, n_i, k and T are the charge of the ith ionic species, the concentration of the ith ionic species, Boltzmann's constant and absolute temperature respectively. The exponential is often approximated by only considering the first term (linear term) in the Taylor series expansion, yielding the Linear Poisson-Boltzmann equation (LPB)

$$-\nabla \cdot \varepsilon \nabla \phi + (2Ie^2 / kT)\phi = \rho$$

where e is the unit electric charge and the ionic strength is $I = \sum_i \frac{1}{2}(q_i^2 / e^2)n_i$.

The electric potential ϕ for all locations of the system is solved by discretizing space into cubic grids and solving the LPB equation numerically using a finite difference approach. This means that, in the computational implementation, ρ and ε are described as matrices representing the charge density and the electric permittivity at all locations of the system grid. For the parts of the grid in the solute, the electric permittivity (dielectric constant) is assigned a smaller value relative to the dielectric constant assigned to regions of the grid that represent the solvent. Therefore the choice of values for the solute and solvent dielectric constants matters for the calculation.

The advantage of these PB type calculations is speed. This method takes advantage of the macroscopic description of the water. The only water related term that enters the calculation is the dielectric constant of water. There are no other water parameters that enter the calculation, and there are no water configuration terms that enter the calculation, except for the size of the solvation box.

The user must choose the dielectric values based on an empirical process: comparing calculated results to experimental results and fitting the dielectric values accordingly. Recommended dielectric solute constants range from 2 to 20 (with a solvent dielectric constant fixed at 80 for all cases). The superficial reason for such a large range for the recommended dielectric constant is because there is no one uniform empirical fitting method for their derivation. The underlying reason for such a large range is that modeling water as a macroscopic entity and assigning it a large dielectric constant (relative to the dielectric constant of the solute) is insufficient to capture the behavior of water and the biomolecule. Recall how strange water is, especially in its interaction with solute (section 1.4.1).

1.10.2   Langevin Dipole models

Langevin dipole models have been extensively used by the Arieh Warshel group[46]. This is a microscopic or semi-microscopic approach, in which the system electrostatics is modeled as a combination of permanent dipoles, inducible dipoles and charges. The most detailed of these models require no assignment of dielectric constants, and the less detailed models do require the assignment of dielectric constants, but the values to be used are consistent, or there is a well-defined method for choosing which dielectric constant goes with which regions. The system is described as a lattice, in which the dipoles are free to orient according to the Langevin response function[47],

$$\vec{\mu}^L = \mu_o \hat{E}_o (\coth y - \frac{1}{y}) \quad \text{with} \quad y = \frac{\mu_o E_o}{kT} \ .$$

$\vec{\mu}^L, \mu_o, \hat{E}_o, k$ and T are the thermally averaged dipole, the dipoles permanent moment, the electric field unit vector, Boltzmann's constant and absolute temperature.

1.10.2.1 Atomic Detail description
The atomic detail description uses a classical mechanical force field and has the following character. Each atom is modeled as a mass with a point (monopole) charge and van der Waals parameters. The covalent bonds are springs to which equilibrium lengths, equilibrium angles, linear stiffness coefficients and angular stiffness coefficients are assigned. Systems are typically solvated with explicit water molecules, and each water molecule modeled in explicit H_2O atomic detail. Periodic Boundary conditions are almost always performed on such solvated systems to eliminate boundary condition complications. The basic form of the force field is as follows:

$$U(R) = \sum_{bonds} K_r (r - r_{eq})^2 \qquad \text{bond}$$

$$+ \sum_{angles} K_\theta (\theta - \theta_{eq})^2 \qquad \text{angle}$$

$$+ \sum_{dihedrals}^{n} \frac{V_n}{2} (1 + \cos[n\phi - \gamma]) \quad \text{dihedral}$$

$$+ \sum_{i<j}^{atoms} (\frac{A_{ij}}{R_{ij}^{12}} - \frac{B_{ij}}{R_{ij}^6}) \qquad \text{van der Waals}$$

$$+ \sum_{i<j}^{atoms} \frac{q_i q_j}{\varepsilon R_{ij}} \qquad \text{elecrostatic}$$

The following sections will go into more detail about the parameters for each component of $U(R)$.

1.10.2.2 Atom parameters

For most current force fields, there are three atom parameters assigned to each atom. These are the mass, the van der Waals parameters and the partial charge. The mass and the van der Waals parameters are assigned to an atom according to its atom type. The partial charge assigned is independent of the atom type. The term "atom type" in the molecular dynamics force field context refers to more than simply the atomic element. For example a C_α carbon and a C_β carbon may share the same element, carbon, but because their chemical bonding is different, they are different atom "types". Both carbon versions will have the same mass (12amu) but their partial charges (permanent monopole charges) are different and their van der Waals parameters are different. The following sections will discuss the partial charges and the van der Waals parameters derivations. Van der Waals parameters and the mass assigned to an atom depends on the atom "type". However the partial charge assigned to an atom is independent of the atom "type".

1.10.2.2.1 Partial Charges

The partial charge of an atom is independent of the atom type. It is assigned to each atom, and is the permanent monopole assigned to that atom. The term "partial charge" came about as follows. In the very early molecular dynamic models, proteins were first modeled such that the amino acids were the elemental units, which were either neutral or had a charge magnitude of

one electron charge. The trend toward atomic detail necessitated that the amino acids themselves be constituted of covalently connected atoms as the elemental units. This allowed for charge distribution schemes within the amino acid to more realistically model the amino acid. Whatever the charge distribution scheme, it was subject to the constraint that the sum of the charges within the amino acid had to be correct. The charge distribution was implemented by assigning charges to the atomic positions of the atoms within the amino acid. These charges could be positive or negative, and were usually fractions of an electron charge. The sum of charges on all the atoms types in an amino-acid had to add up to the correct charge of the amino-acid, which is a whole number of electron charges, either 0, -1 or +1 electron charges. Hence the term "partial charge" was used to reflect the fact that each atom within the amino acid bears part of the charge of the whole amino acid, which is a whole number of electron charges. We will now discuss how those partial charges are determined.

So far we have discussed two constraints on the charge distribution within the amino acid. The first is that the sum of the charges of the charge distribution must be correct, that is equal to the charge the amino-acid supposed to have, which is either 0, -1 or +1 electron charges. The second is that the charge distribution consists of monopoles that are centered at the atom positions. These partial charges, or monopoles centered on the atoms, are derived as follows. First Ab Initio measurements of the electron potential surrounding an amino acid are made. Then using an atomic detail model of the amino acid, partial charges are placed on the positions of the atom types in order to best fit the Ab Initio electron potential, with the constraint that the monopole sum is correct for the amino-acid.

Discussed was the general overview for deriving the partial charges. Actual partial charge derivation requires many more considerations, constraints and restraints. Here is a quick summary of all of the factors that go into a partial charge distribution scheme for an amino-acid: The amino-acid's atom configuration that was used in the Ab Initio calculation, the Ab Initio method used to generate the Ab Initio electric potential, the precision of the Ab Initio electric potential description (the number of grid points per unit volume for which electric potential measurements were made), the charge fitting algorithm used to fit the partial charges on the atom type centers and thereby reproduce the Ab Initio electric potential, additional constraints and restraints such as enforcing symmetry (e.g. the partial charges of the $-CH_2-$ group are usually fit so that both hydrogen atoms have the same charge) etc.

1.10.2.2.2   VDW parameters

van der Waals parameters are assigned to atoms, come in pairs, depends on the type of atom, and are derived by empirical fitting. Enthalpy and separation experimental measurements are made of small molecules. Computational enthalpy and separation measurements are then made of the same small molecule, and the van der Waals parameters are fit to reproduce the experimental numbers. The version of hybridization of the heavy atoms of the small molecule, which loosely translates into the atom "type" in a molecular dynamics model, is then assigned these van der Waals parameters.

1.10.3   Electrostatic long range effects

In early Molecular Dynamics models, or for small Molecular Dynamics models, the electrostatic potential at any point in the system, E_i, is calculated according to a straightforward sum of all of the contributions from all of the monopole pairs in the system.

$$E_i = \sum_{j \neq i} (q_j \frac{1}{4\pi\varepsilon R_{ij}}) \text{ where } R_{ij} = |\vec{r}_i - \vec{r}_j|$$

This sum is performed for every atom position ($\vec{r}_i$) in the system, in order to calculate the electrostatic force contribution on every atom for the purpose of calculating the new velocities and new positions of the atoms.

This sum is performed for every time step of the simulation, and is the most time consuming part of the calculation. The number of pair-wise sums goes as N^2, where N is the number of atoms in the system, so the computation time goes as N^2 with the system size N. This effect is such that this method cannot be used for modest sized systems. Early Molecular Dynamics models addressed this by using a "cut off" scheme. That is, only the pair-wise monopole contributions that lay within some cut off distance (e.g. 8 Angstroms) of an atom were considered for the electric potential calculation at the position of that atom. Such a cut off scheme solved the problem of calculation time growing exponentially with system size. However inaccuracies in the electrostatic calculations using such methods were not insignificant. This is because of the long-range electrostatic field effects in solvated systems.

The reasons for long-range electrostatic effect in solvated systems is similar to the reasons why 19th century calculations predicted that the night sky should be brilliant. Long before the understanding that the universe is expanding and assuming a constant density of stars in the universe, scientists considered the light reaching a point in the universe from all the stars within a solid angle subtended to that point. By integrating the contributions from infinitesimal shells over all distances from the point, scientists predicted that the night sky should be infinitely bright. This is because the number of stars in each shell increases with r^2, which compensates for the light intensity decay with distance ($1/r^2$). The idea of "dark matter", which cancelled the effect of the light from the stars, was proposed as a possible explanation. Current understanding of the expansion of the universe and the subsequent net red shift effect explains the night's darkness.

The problem with point charge contributions to a point from a periodic infinite array of solvated neutral cells is not quite so bad. In an infinite array of neutral cells, the total sum of the positive charges equals the total sum of the magnitude of the negative charges. If the charge distribution is overall neutral and the distances are large enough, the contributions from the positive charges will cancel the contributions from the negative charges (analogous to the "dark matter" counteracting the light of the stars). This means that for periodic solvated systems, a very large cut-off scheme will work. However in practice such a large cut-off (in the order of hundreds of angstroms) is not computationally feasible.

In the 1920's, crystallographer Paul P. Ewald[48,49] needed to calculate the Coulomb energy in salt crystals armed only with manual adding machines. He was able to calculate this by increasing the complexity of the sum so that the infinite sum could be converted into two finite sums. Taking advantage of the fact that the Fourier Transform of a Gaussian is a Gaussian, he added Gaussian charge distributions in such a way that there was a convergent direct space sum and convergent reciprocal space sum.

In more modern times, computational simulations of solvated systems used cut-off methods for electrostatic calculations. It was realized that the errors due to implementing feasibly short cut-offs were not insignificant, and that no matter how fast the computer, the cut-off could not be made large enough for the sum to be convergent. Then Ewald's method was rediscovered and adapted for solvated biological simulations[50].

The following flow of the Ewald explanation follows that given by David Kofke, Department of Chemical Engineering, SUNY Buffalo[51]. Consider a periodic system consisting of an original simulation volume L^3 with N point charges and an infinite number of image volumes. Each image volume is identified with the vector $\bar{R}_n$ where n is an integer >1, and the original cell is identified with $\bar{R}_0 = \bar{0}$ The electrostatic energy is calculated only for positions within the original volume. Consider some point charge q_i at some position $\bar{r}_i$ in the original cell.

$$v(\bar{r}_i) = \sum_j^N \sum_{n=0}^{\infty} \frac{q_j}{\left| \bar{r}_i - \bar{r}_j + \bar{R}_n \right|} \qquad \{n=0,\ j \neq i \qquad \text{is the potential at this}$$

position due to all the surrounding charges. The condition $\{n = 0,\ j \neq i$ excludes the self-energy terms. The total electrostatic energy of the system is therefore

$$\frac{1}{2} \sum_i^N \sum_j^N \sum_{n=0}^{\infty} \frac{q_i q_j}{\left| \bar{r}_i - \bar{r}_j + \bar{R}_n \right|} = \frac{1}{2} \sum_i^N q_i v(\bar{r}_i) \qquad \{n = 0,\ j \neq i$$

When $n = 0$ the $\sum_j^N$ sums the contributions from the q_j charges in the original simulation cell, and when $n > 0$, $\sum_{n>0}^{\infty}$ sums the contributions from the q_j charges in the array of infinite image cells.

This periodic system of infinite cells lends itself to Fourier Transform solutions. However the point charges $q_j \delta(\bar{r} - \bar{r}_j)$ do not. If the point charges were smoothed such that the charge distributions were spherical Gaussians,
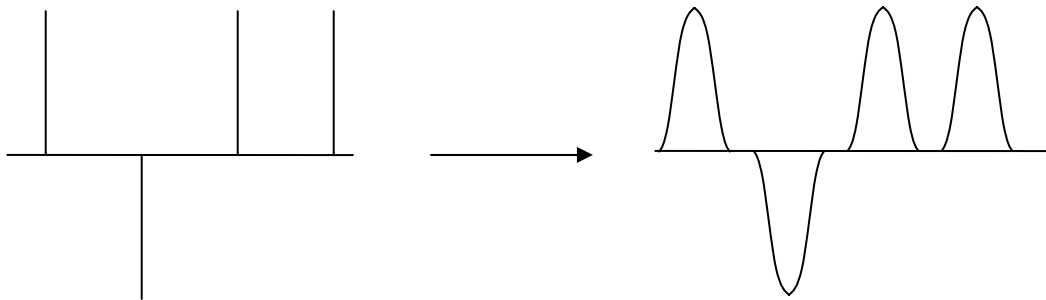
then the sum for the electrostatic energy can converge. That is ρ_j, the charge density near the *jth* charge is:

$$\rho_j(\overline{r}) = q_j \delta(\overline{r} - \overline{r}_j) \rightarrow \rho_j(\overline{r}) = q_j \left(\frac{\alpha}{\pi}\right)^{3/2} e^{-\alpha|\overline{r}-\overline{r}_j|^2}$$

Where $1/\sqrt{\alpha}$ is proportional to the width of the Gaussian distribution. The charge density for

the whole system becomes $\rho(\overline{r}) = \sum\limits_{n=0}^{\infty} \sum\limits_{j=1}^{N} q_j \left(\frac{\alpha}{\pi}\right)^{3/2} e^{-\alpha|\overline{r}-\overline{r}_j+\overline{R}_n|^2}$ so the total energy of the system

becomes

$$U_r = \frac{1}{2}\sum\limits_i^N q_i v(\overline{r}_i) = \frac{1}{2}\sum\limits_{n=0}^{\infty}\sum\limits_{i=1}^{N}\sum\limits_{j=1}^{N} q_i q_j \left(\frac{\alpha}{\pi}\right)^{3/2} e^{-\alpha|\overline{r}_i-\overline{r}_j+\overline{R}_n|^2}$$

This can be expressed in terms of the inverse of its Fourier Transform

$$U_r = \frac{1}{2}\sum\limits_{\overline{k}}^{\infty} e^{i\overline{k}\cdot\overline{r}} \sum\limits_i^N q_i \hat{v}(\overline{k})$$

$\hat{v}(\overline{k})$, the Fourier Transform of $v(\overline{r})$ is obtained from the Poisson relation $\nabla^2 v(\overline{r}) = -4\pi\rho(\overline{r})$.

Using the Fourier Transform property for derivatives, $FT[\nabla^2 v(\overline{r})] = -\left|\overline{k}\right|^2 \hat{v}(k) = -4\pi\hat{\rho}(\overline{k})$

So $\hat{v}(\overline{k}) = \dfrac{4\pi\hat{\rho}(\overline{k})}{\left|\overline{k}\right|^2}$ which makes $U_r = \dfrac{1}{2}\sum\limits_k^{\infty} e^{i\overline{k}\cdot\overline{r}} \sum\limits_i^N q_i \dfrac{(4\pi\hat{\rho}(\overline{k}))}{\left|\overline{k}\right|^2}$. The Fourier Transform of

Gaussian $\rho(\overline{r})$ is $\hat{\rho}(\overline{k}) = \frac{1}{V}\int\limits_V d\overline{r} e^{-i\overline{k}\cdot\overline{r}} \rho(\overline{r}) = \frac{1}{V}\sum\limits_j q_j e^{-i\overline{k}\cdot\overline{r}_j} e^{-k^2/4\alpha}$

So that makes $U_q = \frac{1}{2}\sum\limits_{\overline{k}} \dfrac{4\pi}{k^2 V} e^{-k^2/4\alpha} \sum\limits_{i,j} q_i q_j e^{i\overline{k}\cdot(\overline{r}_i-\overline{r}_j)} = \frac{1}{2}\sum\limits_{\overline{k}} \dfrac{4\pi}{k^2 V} e^{-k^2/4\alpha} \left|S(\overline{k})\right|^2$

where $S(\overline{k}) = \sum\limits_i q_i e^{i\overline{k}\cdot\overline{r}_i}$

There are two corrections needed. The first correction to be discussed is one we call the

self-interaction correction. Recall that $U_r = \dfrac{1}{2}\sum\limits_i^N q_i v(\overline{r}_i)$ where q_i is a point charge at $\overline{r}_i$ and

$v(\overline{r}_i)$ represents potential due to all the Gaussians of the system, including its own. So the

correction involves removing the interaction between the point charges and their own Gaussian

charge distributions $\rho_j(\overline{r}) = q_j(\alpha/\pi)^{3/2} e^{-\alpha|\overline{r}-\overline{r}_j|^2}$. The potential due to such a distribution

centered at $\bar{r}_j$ is $v_j^G(r) \equiv \dfrac{q_j}{|\bar{r}-\bar{r}_j|} erf\left(\sqrt{\alpha}\,|\bar{r}-\bar{r}_j|\right)$. This is verified by substituting $v_j^G(\bar{r})$ into the

Poisson equation $\nabla^2 v_j^G(r') = \frac{1}{r'^2}\frac{\partial}{\partial r'}\left(r'^2\frac{\partial}{\partial r'}\right)v_j^G(r') = -4\pi\rho(r')$ where $r' = |\bar{r}-\bar{r}_j|$.

Since the potential where the point charge is located is in the center of the Gaussian $(\bar{r}=\bar{r}_j$ and $|\bar{r}-\bar{r}_j|=0)$ the self-interaction energy for one point charge q_i is

$\dfrac{1}{2}q_i v_i^G(0) = \dfrac{1}{2}q_i\left[2q_i(\alpha/\pi)^{1/2}\right]$. So the total self-interaction energy is $U_{self} = \left(\dfrac{\alpha}{\pi}\right)^{\!1/2}\sum_i q_i^2$. Note

that this term depends only on the value of the charges not their positions, so it need only be calculated once at the beginning of the simulation.

  The next correction involves correction for the use of the Gaussian charge distribution instead of point charges. We can do this by adding the correct potential and subtracting the Gaussian one.
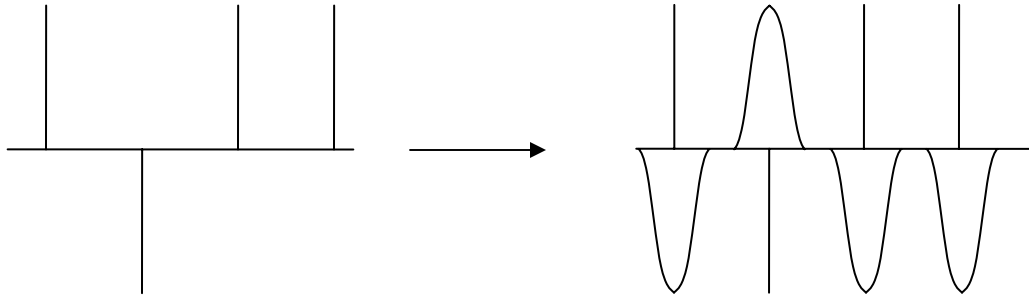
$$\Delta v_j(\bar{r}) = v_j^{\text{point\_charges}}(\bar{r}) - v_j^G(\bar{r})$$

$$= \frac{q_j}{|\bar{r}-\bar{r}_j|} - \frac{q_j}{|\bar{r}-\bar{r}_j|} erf\left(\sqrt{\alpha}\,|\bar{r}-\bar{r}_j|\right)$$

$$= \frac{q_j}{|\bar{r}-\bar{r}_j|} erfc\left(\sqrt{\alpha}\,|\bar{r}-\bar{r}_j|\right)$$

So the correction energy is $\Delta U_d = \frac{1}{2}\sum_{i\neq j} q_i \Delta v_j |r_{ij}| = \frac{1}{2}\sum_{i\neq j}\dfrac{q_i q_j}{|r_{ij}|} erfc\left(\sqrt{\alpha}\,|r_{ij}|\right)$

Notice that ΔU_d only considers interactions between charge distributions in the original cell. This is because ΔU_d is a short ranged function because from q_i's position, for large r_{ij}, the point charge q_j and its inversely charged Gaussian look the same. ΔU_d is therefore done in real

space. All contributions from $r_{ij} > r_{cut-off}$ can be therefore be ignored and in practice $r_{cut-off} \ll L$, the length of the cell.

In summary $U_{total} = U_r + \Delta U_d - U_{self-int}$. Consider the first term, the one done in reciprocal space. The number of *ks* required in the reciprocal sum (k_{max}) is proportional to inverse of the Gaussian thickness, i.e. $k_{max} \propto \sqrt{\alpha}$. So the sum over 3D $\bar{k}$ requires $O(\alpha^{3/2}L^3)$ terms. Also note that each term requires the evaluation of $S(\bar{k})$, which has N terms. So the U_r term requires $O(N(\sqrt{\alpha}L)^3)$ operations. The scaling of the $U_{self-int}$ term can be ignored since it need only be done once at the start of the simulation. Now consider the number of operations needed for the direct sum ΔU_d term. The energy contribution at all N positions of the charges is calculated, but each of these terms only considers interactions within a cut-off $r_{cut-off}$. This cut-off distance is proportional to the Gaussian width, i.e. $r_{cut-off} \propto 1/\sqrt{\alpha}$. The number of interactions within a cut-off volume $r_{cut-off}{}^3$ is $\rho r_{cut-off}{}^3 = \frac{N}{V} r_{cut-off}{}^3$. So the total number of operations required for the direct sum part is $O\left(N^2/\left(\sqrt{\alpha}L\right)^3\right)$. Minimizing the total number of operations $O\left(N^2/\left(\sqrt{\alpha}L\right)^3 + N\left(\sqrt{\alpha}L\right)^3\right)$ with respect to $\left(\sqrt{\alpha}L\right)^3$ gives $\left(\sqrt{\alpha}L\right)^3 = \sqrt{N}$. Therefore the Ewald method scales as $O(N^{3/2})$. The Particle Mesh Ewald method further improves performance by assigning the charge densities $\rho(r)$ to a grid, and then calculating $\hat{\rho}(\bar{k})$ by FFT. This method scales as $O(N \log N)$ [52]

1.10.4   van der Waals interactions

The van der Waals potential at the position of an atom is also calculated by dividing the system into two regions, one within some cut off distance and the other outside the cut off region. The cut off radius is usually about 8 Angstroms. The van der Waals contribution within the cut off region is calculated according to the following pair-wise sum, $\sum_{i<j}^{atoms} (\frac{A_{ij}}{R_{ij}^{12}} - \frac{B_{ij}}{R_{ij}^{6}})$. R_{ij} is the

distance between atoms i and j. A_{ij} and B_{ij} are functions of the van der Waals parameters of atoms i and j, i.e. A_i, B_i, A_j and B_j.

1.10.5   Bond parameters

Every covalent linear bond in an atomic detail model is represented as a spring with two parameters, an equilibrium length and a spring-stiffness. These parameters are derived from x-ray and NMR data of small molecules. Every unique combination of 2 atom *types* yields a unique bond *type*. So a $C_\alpha - C_\beta$ bonds and a $C_\alpha - N$ bond are distinguished as different bond *types* having unique parameters.

Angle bond and dihedral bond types are derived and defined in similar ways. Every unique combination of 3 atom *types* yields a unique bond angle *type*. Every unique combination of 4 atom *types* yields a unique dihedral bond *type*. One would expect that the number of unique combinations of atom types would make for very large databases, especially for the angle and dihedral bonds. These databases are large, but they are not so large because there is a lot of degeneracy among the different atom types, linear bond types, angle bond types and dihedral bond types. The potential energy of the system due to bond distortion, $U_{bonds}(R)$, is given by:

$$
\begin{aligned}
U_{bonds}(R) = &\sum_{bonds} K_r (r - r_{eq})^2 && \text{linear bonds} \\
&+ \sum_{angles} K_\theta (\theta - \theta_{eq})^2 && \text{angle bonds} \\
&+ \sum_{dihedrals}^{n} \frac{V_n}{2}(1 + \cos[n\phi - \gamma]) && \text{dihedral bonds}
\end{aligned}
$$

K_r, r_{eq} and $(r - r_{eq})$ are the linear bonds stiffness, linear bond equilibrium length and linear bond distortion respectively. K_θ, θ_{eq} and $(\theta - \theta_{eq})$ are the angle stiffness, equilibrium angle and bond angle distortion respectively. $\frac{V_n}{2}(1 + \cos[n\phi - \gamma])$ is the sinusoidal dihedral energy function, where V_n is the maximum of the *nth* term of the dihedral function which has a periodicity of n, γ is the phase and $n\phi - \gamma - 180^o$ is the distortion away from the potential minimum.

1.10.6 Quantum Chemistry models

Quantum Chemistry models use Quantum Mechanics principles. It is the most computationally demanding of all the models and its use is limited to treating small regions of the biomolecule. Because its theory goes down to the electronic arrangements of the molecules, it is versatile and has a wide range of applications, including modeling of breaking and making of chemical bonds.

Quantum Chemistry methods are based on a solution of Schrodinger's Equation. The time-independent Schrodinger Equation is $\hat{H}\Psi = E\Psi$. $\hat{H}, \Psi$ and E are the Hamiltonian, the wave function, and the energy of the system respectively. The exact solution exists only for the single hydrogen atom system. Solutions for larger systems require approximations to the Schrodinger's Equation. Several methods exist for doing Quantum Chemistry energy calculations, and I will talk about a few of the main methods and the related approximations used for small biological molecules.

The Born-Oppenheimer approximation[69] is almost a universal approximation for Quantum Chemical methods. In this approximation, the mass of the nucleus is considered to be large relative to that of the electron, so that the motion of the electron and that of the nucleus is considered to be uncoupled. This simplifies Schrodinger's Equation by allowing the several terms to be dropped, and the wave function of the molecule can be written as a product, $\Psi_{molecule} = \Psi_{electrons}\Psi_{nuclei}$. The Schrodinger Equation is solved for the electronic Hamiltonian only, and the other terms in the Hamiltonian are dealt with otherwise. Each wave function of the system, Ψ_i is described as a linear sum of basis functions, $\Psi_i = \sum_n c_{in}\phi_n$, where $\phi_n's$ are the predetermined basis functions and the $c_{in}'s$ are determined with an iterative scheme. Most methods add additional layers of approximations. The most popular methods are Hartree-Fock, Density Functional Theory and Molecular Orbital methods[53].

Because Quantum Chemistry is so computationally demanding, modeling of biomolecules is usually done by dividing the system into two regions. The core region targets the area of interest, which would have a few dozen atoms at most. The outer region is handled with Classical Mechanics and various schemes are used to couple the two regions. For such a model the Hamiltonian takes the form $\hat{H} = \hat{H}_{QM} + \hat{H}_{QM/MM} + \hat{H}_{MM}$. $\hat{H}$ is the Hamiltonian of the

whole system, $\hat{H}_{QM}$ is the Hamiltonian that represents the Quantum Chemistry region, $\hat{H}_{MM}$ represents the Classical Mechanical region and $\hat{H}_{QM/MM}$ represents the influence of the Classical Mechanical region on the Quantum Chemistry region.

1.11 COMPUTATIONAL TOOLS FOR DYNAMIC ANALYSIS

All of the reasons for dynamical analysis of biomolecules can be placed into two main categories. The first bin contains reasons relating to understanding the collective motion of the system. This is the most transparent reason because the most obvious question about how a biomolecule performs a task is how its configurational changes or dynamics allow it to perform that task. The second bin of reasons relate to more exhaustive analysis of the trajectories for thermodynamic calculations via density of state and ensemble approximations. The first set of reasons, those relating to investigating collective motion, were historically the first reasons that attracted investigation of the dynamics of biomolecules. We will start our survey of dynamical analysis tools with one of the earliest of such tools, Normal Mode Analysis.

1.11.1 Normal Mode Analysis

Normal mode analysis is one way of investigating collected or correlated dynamics within biomolecules. For Normal Mode Analysis, the system is modeled such that the underlying character of the model is a harmonic Classical Mechanical force field acting on the atoms, as a result of the covalent bonds connecting the atoms modeled as springs. Taking advantage of this harmonic description, Normal Mode Analysis is then performed, where the description of the system's propagation is changed from a coupled representation to a decoupled representation, thereby making the normal modes, the collective motion, and the correlated movement of the system transparent. If there is collective motion in the system, (which is usually slow and large) this analysis will allow that aspect of the dynamics to be easily revealed.

A harmonic system of masses can be described as a force-matrix acting on a position vector to yield a vector that describes the time evolution of the system's configuration. The core

of Normal Mode Analysis is to diagonalize that matrix to obtain the frequencies and forms of the normal modes.   Once the frequencies and forms of the modes are known, magnitudes, time scales and correlations of atomic fluctuations can be calculated.

If η is the matrix describing displacement from equilibrium of every mass in the system, then for our harmonic system:

$$H \cong \frac{1}{2}\eta^T \frac{\partial^2 H}{\partial q_i \partial q_j}\eta$$

The above eigenvalue problem is then solved to yield a set of harmonic oscillators.

The biomolecule is therefore modeled as a bunch of masses connected by springs, with the masses also subject to an-harmonic potentials such as electrostatic and van der Waals potentials. In the model, the masses do not move far from their equilibrium position, thereby allowing a harmonic approximation, even though an-harmonic potentials like electrostatic or van der Waals potentials are present.   For closely packed systems at low temperatures and only moderate collective motion, the above model is sufficient.

The harmonic approximation is insufficient to describe the dynamics of biomolecules when large collective motions take place. The Principal Component method is an analogy to the normal mode method at room temperature (where the anharmonicity plays a negligible role). It does not assume harmonicity, so it can be used to investigate biomolecules that engage in large-scale collective motion.

The advantage of normal mode and principal component type methods is that it is computationally quick to perform on relatively large systems in vacuo, and is good for analysis of large-scale collective motions. However this computational speed advantage is attenuated when the system includes explicit solvent, and its usefulness does not include analysis of localized phenomena on the atomic detail scale. Therefore phenomena like specificity, proton dynamics, hydrogen bond networks or ionizable site networks cannot be explored with these methods.   Based on these methods, only limited thermodynamic calculations can be made because the approximations made in handling the harmonicity or an-harmonicity of the target system often accrue to produce significant error.

1.11.2   What is Monte Carlo? A short overview of MC

The Monte Carlo simulation techniques were introduced by Metropolis *et al* in 1954 and were used extensively to investigate phase transitions in simple models. These techniques established their value on simple lattice-type systems, where the constituent particles have very few parameters, and very few degrees of freedom. Monte Carlo simulations were also found to be useful in situations where the investigators were interested in other properties besides phase transitions, and so started to be used to predict a range of behaviors in chemical and biological systems[54, 55]. MC can be used to sample the system phase space and scaling arguments can be used to infer the time dependence of the dynamics of the system. Earlier MC methods had a big problem tackling models that have many parameters and many degrees of freedom, such as models of biological systems, because the method was too computationally expensive. However new techniques, algorithm improvements, hybrid MC-dynamics methods and computational power improvements have made Monte Carlo methods useful in complex systems, where the simulated particles have many parameters and many degrees of freedom. Monte Carlo theory and algorithms have evolved to the point where they can do biomolecular simulations[56]. However, where biomolecular simulations are concerned, Monte Carlo is a distant second to Molecular Dynamics in popularity, despite MC's advantage of crossing phase space energy barriers (which MD notoriously does not). There are two reasons for this. The first is that MD is more established. The second reason, related to the first, is that the development of efficient hybrid MC methods continues, and the laborious task of code writing for these developments puts the cost having these advantages in perspective.

In Molecular Dynamics, a classical mechanical force field expresses a force on the particles of the system. The position of each particle is updated every Δt increment of time, according to a numerical solution of Newton's second law of motion. In Monte Carlo simulations, the system particles are subject to the same classical mechanical energy field, however its derivative is not needed since only the energies and not the forces are needed. The new positions are determined probabilistically as follows. Along any one degree of freedom, the particle has a few options for a new position. The classical mechanical force field is then used to determine the energy penalty of each option, and each option is assigned a Boltzmann factor

appropriate for its respective energy penalty. A random number generator then randomly selects one of the Boltzmann weighted options.

1.11.3   Molecular Dynamics (MD)

In MD Newton's Second Law, $\vec{a} = \vec{F}/m$, is solved numerically one incremental time step (Δt) at a time, for every particle in the system. The force field acts on the particles, causing the velocities and positions of the particles to be updated according to the approximated numerical solution. Molecular Dynamics simulations can record the evolution of system configuration over simulated time (a trajectory). The force on the particles is the gradient of the system energy, $\vec{F} = -\vec{\nabla}E$. There are many algorithms for performing the numerical integration for the updated velocities and positions, but one of the most popular is the Verlet leapfrog algorithm[57, 58, 59, 60]. The system energetics (E) can be derived from a Macroscopic, Atomic Detailed or Quantum Chemistry description. Because the system energetics has to be recalculated at every time step, the choice depends on the size of the system and the available computational power.

MD started out as a tool for biomolecule crystallographers. X-ray crystallography provided a leap into the understanding of the function of biomolecules because it allowed the structure to be determined. Analysis of the structure gave insight into how the biomolecule executed its function. Some dynamic analysis was possible, using the R factors[61] of the electron density map. R factors are one measure of the quality of x-ray protein models, and regions of the protein that are more mobile tend to have larger R factors. Molecular Dynamics (MD) was born out of the attempt to take the analysis of a biomolecular system further than was possible with structural analysis. In the early history of MD, it was simply used as an appendage of structural analysis, and was only used by those involved in structural analysis. Now, MD has evolved to the extent that persons perform MD related research and have very little experience with processing crystallographic electron density maps.

The first MD models were in vacuo models. That is, there was no explicit modeling of the solvent. The solvent, if represented, was represented as a continuum. The covalent bonds were modeled as springs, whose lengths and stiffnesses were determined using the best NMR and X-Ray data at the time. The atoms were modeled as masses equivalent to their amu weight,

had a monopole charge (called a "partial charge") and also had VDW parameters. However, only the heavy atoms were modeled, and hydrogen atoms were not. They were incorporated into the heavy atom they were connected to. So for example, the $-CH_3$ group will be modeled as a single "ball" or "united atom". This method of modeling such groups is called the "united atom" model. Covalent bond angles and dihedrals were also represented.

As computational resources allowed, water solvent then became explicitly modeled with "periodic boundary conditions" and Ewald long-range electrostatics, which was developed as a computationally feasible means of capturing the long-range electrostatic effects. Current Molecular Dynamics software can allow one to generate nanoseconds/day of trajectory for atomic detail explicit solvent biomolecule models. Typical models use a classical mechanical force field and have the following character. Each atom is modeled as a mass with a point (monopole) charge and van der Waals parameters. The covalent bonds are springs to which equilibrium lengths, equilibrium angles, linear stiffness coefficients and angular stiffness coefficients are assigned. Systems are typically solvated with explicit water molecules, and each water molecule modeled in explicit H_2O atomic detail. Periodic Boundary conditions are almost always performed on such solvated systems to eliminate boundary condition complications. Pressure and temperature control algorithms are added, allowing for trajectory evolution in the NTP ensemble. These trajectories can be used for thermodynamic calculations.

One disadvantage of atomic detail molecular dynamics is that it samples a small region of the energy landscape. As a result, one cannot perform extensive thermodynamic calculations on the trajectories. Our method of integrating MD, MC and WHAM, does allow for extensive thermodynamic calculations. This is because WHAM is used to weave together simulations generated under a wide range of conditions, yielding a good density-of-states description.

1.11.4   Feasibility of the various modeling methods
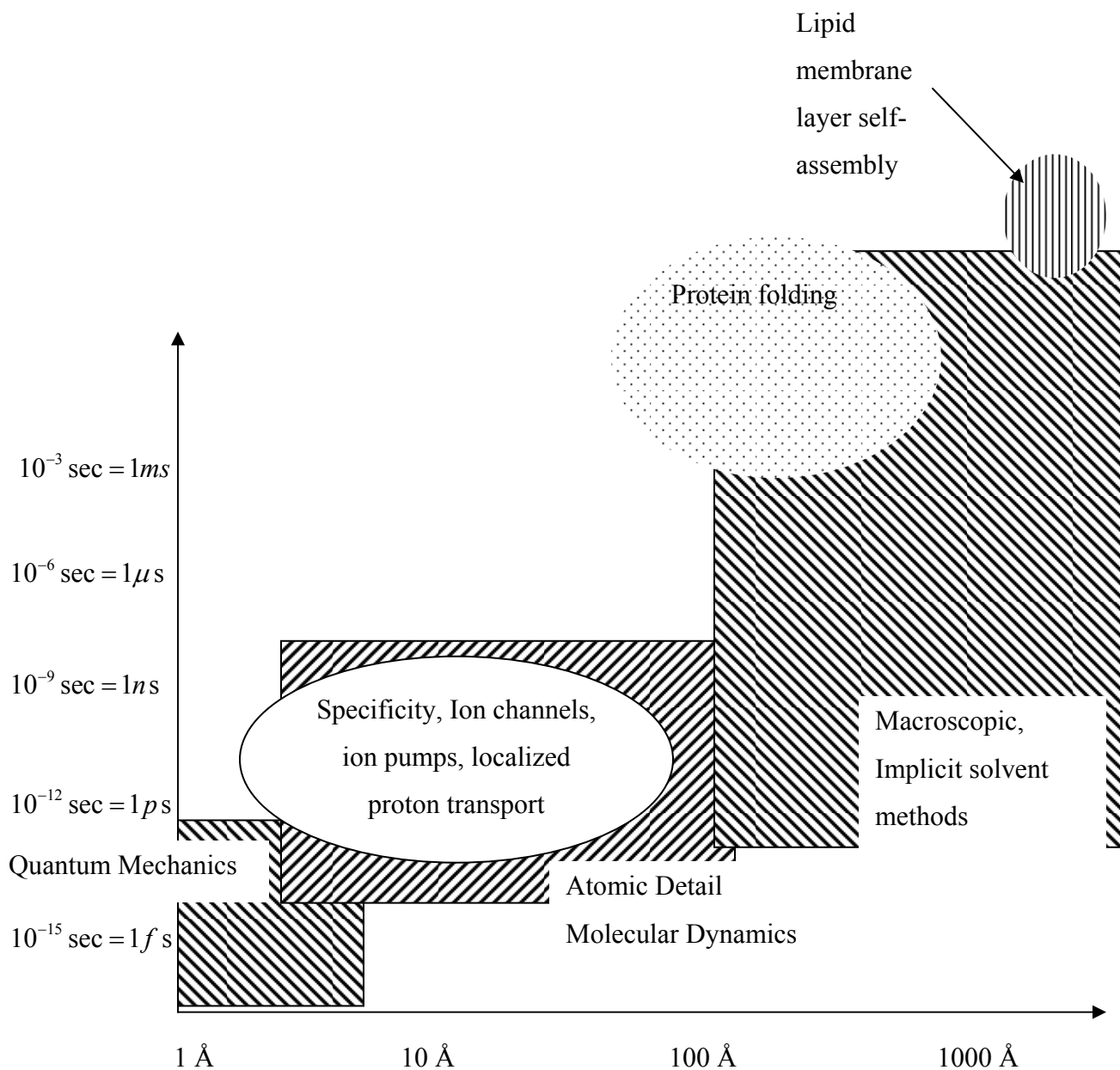
Figure 19: Feasibility of various modeling methods

The plot above attempts to give a feel for the feasibility of various computational methods used in investigating a wide range of biomolecular phenomena of interest. This is based on popular

usage using current computational resources such as in house computing clusters or medium sized allocations at supercomputing centers. The hatched boxes in the plot represent the basic categories of methods. The Quantum Mechanics methods, atomic detail molecular dynamics methods, and macroscopic implicit solvent methods are shown at the bottom left, center, and top right respectively. The overlap between these basic categories is somewhat underestimated in the plot above because there are several hybrid methods.

Pure Quantum Mechanical methods are typically carried out on systems of with only a few dozen atoms, and Quantum Mechanics simulations (simulations where the force field is derived from Ab Initio methods) are typically only a few picoseconds long at most. In order to apply QM methods to larger systems, it is quite common to use hybrid QM – Classical Mechanics methods. With such methods the system is divided into two zones. The core zone contains the part of the system of interest and is limited to a few dozen atoms at most, and the QM type calculations are performed for this part of the system. This outer zone is usually handled with atomic detail classical mechanical force fields. This type of hybrid method can be applied to systems larger than those indicated in the plot above. However the region of interest still has to be highly localized for its complete inclusion into the core zone. These methods are therefore challenged when the region of interest is extended. Such is the case with site interaction networks involving many titratable sites. Such networks may span over lengths of dozens of Angstroms. Another case of an extended region of interest is that of ligand-binding interfaces. Again, such regions may extend over dozens of Angstroms. Typical phenomena investigated by these methods are localized phenomena where bonds are being created or destroyed, bonds change hybridization, electron transfer or highly localized proton transport.

The central part of the plot represents Atomic Detail Molecular Dynamics. These methods can be used to investigate a broad spectrum of biological phenomena. They can be used on various sized systems that span several orders of magnitude of length, and they can investigate various phenomena that occur on time scales that span several orders of magnitude. As a result, these methods are the most useful, the most popular, and the most developed in terms of their computational performance evolution.

Protein folding and lipid membrane assembly are generally tackled with much more simplified models. The lengths and times of these phenomena are at the large end of the scale. Typically in these models there is a macroscopic description of both the solvent and the solute.

Hybrid atomic detail MD- macroscopic models also exist. But that is not the only reason that atomic detail MD is encroaching into regions of larger time and space. Because so much effort is put into improving MD algorithm performance, and also because computational resources are growing more powerful, atomic detail MD is tackling protein folding with increasing occurrence over the past ten years.

1.11.5   Density of states theory applied to the biochemical ensemble

Early statistical mechanics theories, including those relating to density-of-states, were developed in the context of analytical analysis of simple models in condensed matter physics, such as those discussed in section 1.11.6. They can be applied to biological simulations with no loss of rigor however the complexity of biological Hamiltonians requires numerical solutions. The density of states of a system, as the name suggests, is the property that describes how closely packed the energy levels are in that system. **It is very useful.** A good description of a system's density-of-states will allow for a full range of thermodynamic estimates via the calculation of ensemble averages. For our constant *pH* simulation methods, we have had to make some extensions to the most commonly recognizable forms of the density of states related equations. This is done at length in section 3.0.

1.11.6   A short overview of Weighted Histograms

Single histogram methods were first introduced in 1960. They were developed to assist Monte Carlo methods in finding phase transitions of simple two-dimensional models. Finding the phase transitions without histogram methods was difficult, because each Monte Carlo simulation would only sample a narrow region of the phase space of the system parameters. Histograms allowed one to get information on a broader region of the phase space, and consequently get information for a broader region of the energy landscape. This made locating phase transitions easier.

In 1989, Swendsen and Ferrenberg[62] introduced the multiple-histogram method for combining several Monte Carlo simulations. The method was initially tested on the two-dimensional Ising model. Although this method was originally applied to locate phase

transitions, the power of this method to combine the information of many simulations led to far reaching applications that had nothing to do with phase transitions. This method could take a very finite number of simulations, combine the information, and produce a continuum of thermodynamic results for a phase space range as large as that spanned by the simulations.

More details about Weighted Histogram theory is given in section 3.0 however I will quickly summarize how these methods can yield a continuum of thermodynamic results from a finite number of simulations. In Weighed Histogram Methods, the potential energies of simulation snapshots are binned. Bins with high counts correspond to high probabilities or low free energies, and bins with low count correspond to low probabilities or high free energies. These counts therefore allow one to estimate the density of states, which in turn allows for a continuum of thermodynamic estimates.

1.11.7 Biomolecules, MD and WHAM

The Hamiltonian of an Atomic Detailed Molecular Dynamical biosystem is well defined, so there was nothing to stop the multiple-histogram technique from being used in Biomolecular Dynamical systems. The multiple-histogram technique was reformulated to accommodate biomolecular Hamiltonians and this extension was called the Weighted Histogram Analysis Method, or WHAM. It was applied for the first time on a complex biomolecular Hamiltonian to generate a Potential of Mean Force profile of the pseudorotation phase angle of a sugar ring[63, 64].

Atomic Detail Molecular Dynamics and WHAM are a good combination. The traditional problem with atomic-detail MD is that it samples only a small region of the energy landscape. This means that such simulations could yield very limited results, because the sampling was so narrow. The sampling could be broadened by running simulations under different conditions. The problem then becomes "how to combine the information from all these simulations in a useful way?" WHAM solves that problem.

1.12 SURVEY OF METHODS FOR MODELING PROTON DYNAMICS

1.12.1 Overview: Looking at the Big Picture

Analytical   Numerical

Statistical   Statistical

Methods   Methods

Ideal Gas Model:

No interaction except collision and momentum exchange.

Neglect duration of collision, and all other interaction.

Good model for low-density inert gasses.

Real Gas Model:

Includes inter-particle interaction

such as Lennard-Jones interaction,

$U = U(r)$.

Good model for higher density gasses.
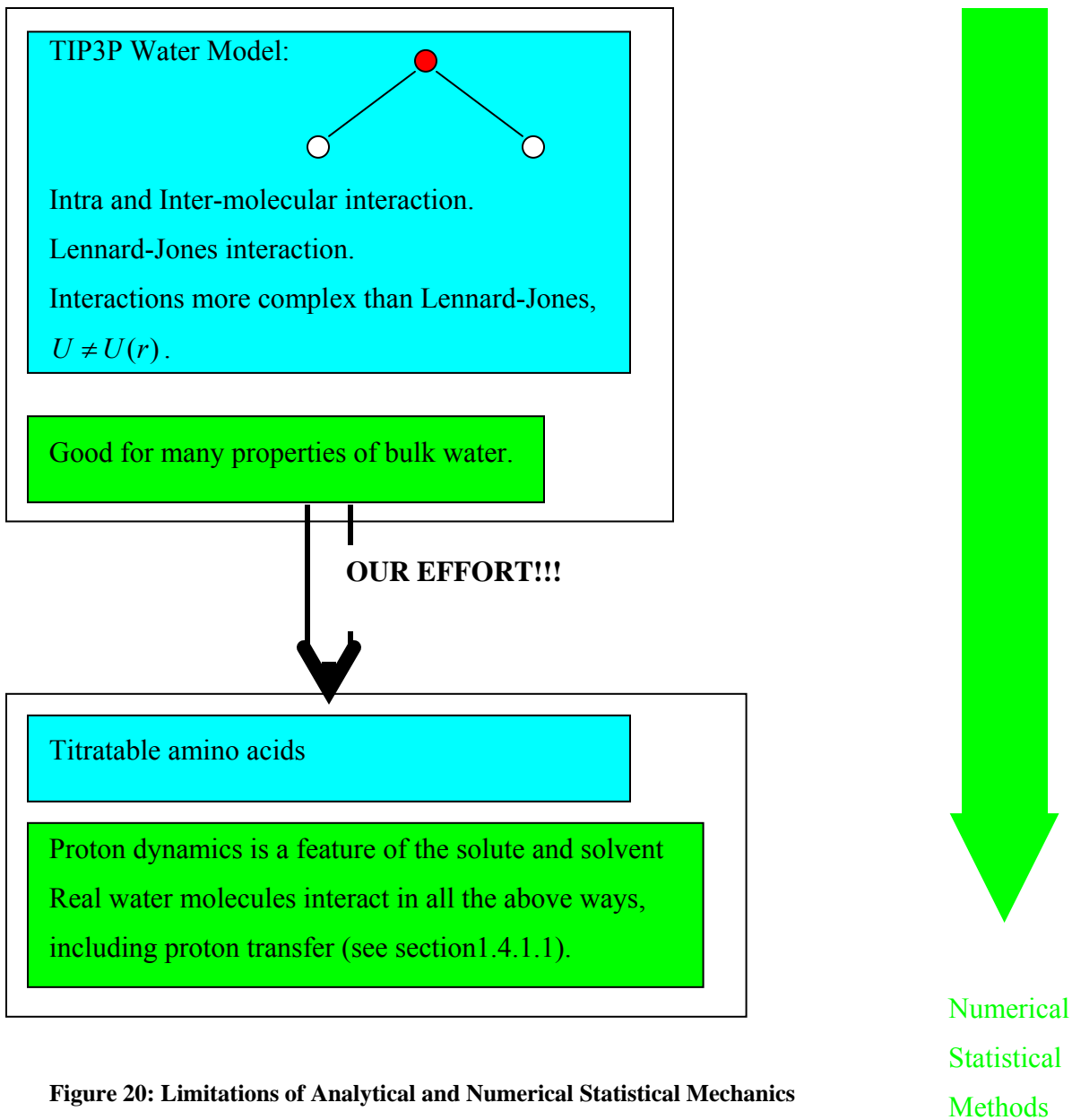
Figure 20: Limitations of Analytical and Numerical Statistical Mechanics

In Figure 20, we attempt to put in perspective how our work fits into big picture of the evolution of analytical and numerical statistical mechanics methods. Classical Statistical Mechanics theory developed from **analytical methods** applied to simple models, such as the "hard spheres" model.

The top model in our Figure 20 is the Ideal Gas model. In this model, the particles are elastic hard spheres that have no interaction with each other except during the infinitesimally small collision times where momentum is exchanged. This model ignores all other inter-particle interactions, and ignores the duration of the collision. Theoretical predictions based on this model are good for predicting several behaviors of low-density inert gasses. In low-density inert gasses the inter-molecular distances, on average, are relatively large and the dominant dynamics is well approximated by modeling the molecules as hard elastic spheres. Therefore these simple models combined with analytical statistical mechanics can take one quite far, as far as calculating many thermodynamic properties of low-density inert gasses. However numerical methods are needed to go as far as Critical Point predictions.

The next types of models, the Real Gas type models, allow for additional interactions between the system particles, such as Lennard-Jones type interactions. In these types of models, the inter-particle interaction is a function of the inter-particle distance. This potential may take many forms, but special note needs to be made concerning the Lennard-Jones (or 6-12) potential. The Lennard-Jones potential does a very good job of modeling many inter-particle interactions. This potential interaction has allowed for the verification of analytically and numerically derived statistical mechanic results for a wide range of systems. Many properties, not only of gasses, but also of liquids and solids, can be explained very well by this crude model[65,66]. Analytical analysis of these types of models is very limited. With analytical methods, if the $U(r)$ interaction potential is not too complicated, the first several virial coefficients can be calculated, and a few thermodynamic results can be calculated without severe approximations. Numerical methods yield many more results.

Now we come to the next class of models, such as the TIP3P water group. I use TIP3P as representative of this class because it is the most extensively used (basically because it does a good job of modeling water and because of its computational feasibility). I also use TIP3P in a much broader representation sense: I'm using TIP3P as representative of a class that includes both solute (proteins, for instance) and solvent "atomic detail" models. In terms of complication, the TIP3P is about in the middle of the class of atomic detail water models. The interaction between TIP3P molecules is **NOT** simply $U = U(r)$, for several reasons. The most obvious being that TIP3P has dipole character, so the inter-molecular interaction also depends on orientation. This dipole character results from a negative point charge on the oxygen, and

positive point charges on each of the two hydrogen atoms. TIP3P has intra-molecular as well as inter-molecular interaction (the $O-H$ bonds are not rigid, but are springs).

At this point (the TIP3P class of models), the usefulness of analytical statistical mechanics is next to none. One would not be able to find literature on analytically derived thermodynamic results for systems with TIP3P type models. There is a "pseudo-analytical" class of techniques for analysis of systems with TIP3P level of molecular and atomic modeling. I put Normal Mode Analysis, and Principal Component Analysis in this class. They are cousin to analytical analysis because their approach to finding solutions to the system is analytical in nature. However a computer is still necessary to solve the system of very large matrices that describe the dynamics of the system. These methods introduce errors due to approximations because of assumptions made in handling the anharmonic character of the system. These approximations can accrue to produce significant error. See section 1.11.1 for more detail on these methods. Therefore, almost all of the statistical mechanics analysis of TIP3P type systems is numerical, so computers are indispensable. Computer simulations of TIP3P show that this model does a reasonably good job of reproducing hydrogen bonding and many other properties of bulk water (see Figure 27).

There are several evolutionary directions for the next generation of models based on the simple atomic detail model. Features such as modeling polarizability, electron lone pairs and titratable amino acids are already implemented into recent versions of Molecular Dynamics packages. We think the most important evolutionary direction is that of modeling titratable amino acids and we believe our method is an efficient way of accurately doing that.

However in real systems, proton dynamics is as much a feature of the solvent as it is of the solute. One way to improve upon the accuracy of the last generation of water models is to have the water model interact in all the above ways, but also with one important addition. Two water molecules of this model need to interact with each other by chemically changing each other as a result of proton transfer (which is what actually happens in real water). In other words, the water model needs to be titratable. In real water, at room temperature and pressure, two water molecules hydrogen-bonded to each other will transfer a proton about once every 20 picoseconds. Modeling proton dynamics in the solvent is hindered because of a lack of a computationally feasible titratable water model.

There are several proton dynamics schemes already available. Some groups come at it from a Quantum Mechanics approach, most come at it from a Classical Mechanical approach. We believe our Classical Mechanical method has a clear edge because of its feasibility, accuracy and precision.

One of the points that Figure 20 tries to emphasize is the very heavy dependence of statistical mechanical biological analysis on computer simulations. As a result there is close correlation between the developments in biological system modeling the developments in computational resources. Please see section 1.9, "SURVEY OF COMPUTATIONAL Resource Evolution" for a little history of this correlation.

1.12.2 A reminder of the importance of proton dynamics

In section 1.5 we discussed that electrostatics in biological systems is important, especially since the electrostatics in solvated biological systems has long-range effects. Accurate modeling of electrostatics is therefore important. Closely related to the issue of accurate electrostatic modeling is the modeling of proton dynamics, because protons carry one electron charge. So the need for more accurate electrostatic modeling cannot be satisfied without addressing the issue of proton dynamics. Now we will survey the pros, cons and limitations of some popular models that allow for proton dynamics.

1.12.3 Basic principles for pKa calculation methods

All pKa calculation methods have in common the following basic principles.

$$pK_a = pH - \frac{1}{kT \log_e 10} \Delta G = pH - \frac{1}{2.303kT} \Delta G$$ where ΔG is the free energy change upon protonation. How the ΔG is calculated depends on the specific method. For many methods, a Monte Carlo process attempts to place a charge of one proton, $+1e$, on the titratable site. Successful attempts are accrued towards a protonation occupancy total, and failed attempts

are accrued towards a deprotonation occupancy total. Some level of Statistical Mechanics theory is then used to translate all of the Monte-Carlo outcomes into a ΔG.

In order to calibrate these methods, the concept of a model pKa, pKa_{model}, is introduced. The pKa_{model} is the pKa of a single solvated titratable amino acid. If the force field used to calculate ΔG modeled nature exactly, then of course the pKa calculated from ΔG for the single solvated titratable amino acid residue would equal the experimental pKa of that titratable residue. However, because the force field does not model nature exactly, the pKa_{model} acts as a force field correction number. The pKa_{model} is included into the calculation as follows. Instead of only measuring a ΔG, the absolute free energy change upon protonation, what is actually measured is a $\Delta\Delta G$, the free energy change for protonation of the titratable site in the protein relative to that of the single residue, or model. So, two calculations are performed. One is the ΔG_{model} calculation for protonating the single solvated titratable residue. The single solvated titratable amino acid is described as the model, and corresponds to the isolated system referred to in section 1.6.4. Th other is the $\Delta G_{protein}$ calculation for protonating the titratable site in the protein. Said another way, this gives a pKa shift, rather than an absolute pKa.

$$\Delta pKa = \frac{1}{kT\log_e 10}(\Delta\Delta G), \qquad \text{where} \qquad \Delta\Delta G = \Delta(\Delta G) = \Delta G_{\text{protein}} - \Delta G_{\text{model}}, \qquad \text{and}$$

$pKa_{\text{model}} = pH - \frac{1}{kT\log_e 10}\Delta G_{\text{model}}$ and $\Delta pKa = pKa_{protein} - pKa_{\text{model}}$. The pKa of the titratable site in the protein is then calculated as: $pKa = pKa_{\text{exp}} + \Delta pKa$.

Many of these methods allow for configuration changes. The Monte Carlo proton dynamics is periodically interrupted to allow for a few steps of molecular dynamics. The intent is to capture more accurate proton dynamics by allowing the system to explore a range of configurations.

1.12.4 Proton Dynamics using Poisson-Boltzmann type models

In the PB implicit solvent models, the water is modeled as a macroscopic with a large dielectric constant. The solute is typically modeled in atomic detail including the titratable sites and is

assigned a smaller dielectric constant. A Monte Carlo process attempts to add or remove a charge of one electron charge to or from all of the titration sites in turn. The energetics of each protonation state of the system is calculated by a Poisson-Boltzmann type calculation. It is the value of the solute dielectric constant relative to that of the solvent dielectric constant that matters for the calculation.

The advantage of these Poisson-Boltzmann type calculations is speed. It is possible, with this method, to perform hundreds of thousands of Monte-Carlo sweeps on each of the hundreds of titratable sites of a large protein, on a moderately powered workstation. The model gives reasonable results for simple cases of sites that are solvent exposed and are not committed to involved site network interactions.

The first problem that arises when conducting these calculations is the choice of a dielectric constant for the solvent and for the solute (the biomolecule). A quick survey of the literature will reveal recommended dielectric solute constants in a range from 2 to 20 (with a solvent dielectric constant fixed at 80 for all cases) with no definitive rules for which value to use in which circumstances. This represents an energy difference of a factor of 10. There is a database of well-established experimental pKa measurements that were conducted on several proteins. The wide range of dielectric values is a result of attempts to fit the calculated pKa to the experimental pKa. It is important to note here that the variation in the dielectric constant that comes from fitting the pKa values of the different titratable sites in ONE protein, is as much, often more, than the variation of dielectric constant assigned to each protein such that it gives the best fit for all its titratable sites in each protein.   In other words, the intra protein dielectric constant variation is as much, often more, than the inter protein dielectric constant. Granted that there is a large range of empirically fit dielectric constant values, what about the rules that guide the user about what dielectric constant to use in what circumstance? For starters, the fact that the intra dielectric constant variation is larger than the inter dielectric constant variation rules out any rules that recommend using one dielectric constant for one genre of proteins, and other dielectric values for a different class of proteins.

There have been several attempts to divide the protein into regions based on solvent exposure. Parts of the protein that are solvent exposed will be assigned a higher dielectric constant, and parts of the protein that are in the hydrophobic core are assigned another dielectric constant. However it is difficult to describe water penetration effects with simple solvent

exposure parameters, because the bound water demonstrates a wide range of behavior depending on the titratable site networks that it interacts with.

This leads to another issue of titratable site networks. The titratable sites interact with each other. Even if the complications of water penetration were removed from the picture, there would still be issues. That is if we considered a subset of titratable sites in our database such that these sites formed networks with minimal influence from water penetration, we would find that we would need different rules for different types of networks. In the simplest situation of a network consisting of only two sites, we would find that the appropriate dielectric for one member of the network pair might differ widely from the appropriate dielectric for the other member of the network pair. In other words, even the intra network value for the dielectric constant shows wide variation. Then of course there is the issue of quantifying water penetration or solvent exposure effects.

In summary the problem is not only is there a wide range of dielectric values to fit the data, but also that there are no definitive rules for which dielectric values to use in which circumstances. The literature contains a lot of analysis that justifies why a site would exhibit a certain dielectric constant, but there is no definitive compilation of these analysis into rules that would allow a user to make a-priori decisions of what dielectric value to use where. The underlying problem is that the macroscopic dielectric is an insufficient model for describing the solute and the solvent. The dielectric assignment model is based on modeling the solute and solvent as simple bulk dielectrics. Of course this ignores hydrogen bonding networks and the proton transport mechanisms that play an important role in the water-solute interaction (see section 1.4.1).

1.12.5   Proton Dynamics with Langevin Dipole models

The system is modeled as a lattice containing a combination of permanent and inducible dipoles. For proton dynamics this model as an advantage over the Poisson-Boltzmann type models because the most detailed of these models do not require assignment of dielectric constants, and the less detailed models do require the assignment of dielectric constants, but the values to be used are more consistent and span a narrower range[2].

These models may be microscopic, but lack atom detail. By backing slightly away from full atomic detail, there are some important system behaviors that may be lost. Full atomic detail of the microstates allows for different exit or entry points of protonation on the titratable site. These positions mean a great deal to the *pKa* of a site if that site is involved in a close electrostatic network. As mentioned, the different positions of proton exit or entry may make as much as 2 *pH* units of difference to the *pKas* of the involved sites. It is difficult to capture this behavior without a full microscopic multiple sub-ionization state description.

On a less important, convenience related note. All approaches that are not fully atom detailed suffer from the fact that there is not a seamless transition from widely used atom detailed structures to the microscopic or the semi-macroscopic descriptions Langevin Dipole description. There is some learning curve involved in converting to the new model and getting things set up, unless all configuration and protonation dynamics were performed in the Langevin Dipole language.

1.12.6 Challenges to explicit atomic detail solvent models

Full atomic detail models are the most widely used simulated models. By virtue of their atomic detail, their trajectories are considered to capture information that would be neglected by less detailed models. This neglected information could accrue over space (the dimensions of the model system) and time to produce significant error. However, using atomic detailed water (solvent) and discrete protonation states is a problem. For a solvent exposed titratable site, the water (atomic detailed water model has a dipolar character) orients to adapt to the field of the protonation state (minimize the electrostatic energy). The solvation shell that forms tends to lock in that protonation state and unreasonably long simulations would be required to get ionization transitions (see section 4.3.3 page 146). For conveniently short simulations, this makes the other ionization state inaccessible to a Monte Carlo selection process.

One solution is to make the protonation state a continuum instead of discrete. Then one can use a *free energy perturbation scheme* (FEP) to force the system from one titratable state to the next[67, 68]. The problem with this is that FEP simulations have to be in equilibrium. Crossing the solvation barrier is like crossing phases. Equilibrium is a big problem with these methods, even when the protonation parameter increments are ever so tiny. One method that could help is

to use FEP to cross the barrier both ways, thousands of times. However such an approach has questionable computational feasibility.

1.12.7   Summary of proton dynamics challenge

The elusive ideal model for proton dynamics would have both explicit atomic detailed solvent, and discrete protonation states. However the solvation shell for such models prevents protonation state transitions during simulations of reasonable length. There are several workarounds. One is to make the solvent a continuum. Another is to make the protonation state a continuum. The last section discussed the problems associated with those two approaches. Another solution is to slightly back off from the atomic detail, and use a Langevin dipole approach.

Our method uses discrete protonation states and explicit solvent. High temperature simulations are used to get good ionization state transition rates. WHAM is used to combine the simulations generated over the wide range of temperatures.

1.13   OUR SOLUTION: THE TRINITY OF MD, MC AND WHAM

Our explicit solvent method uses discrete protonation states model and overcomes the barrier problems by using high temperature simulations and using Weighted Histograms (WHAM) to bring together information from a wide rang of simulations.

1.13.1   Summary of our theory

The model we use to describe our biomolecular systems is the same as the Amber8 explicit solvent atomic detail model. This is a classical mechanical force field model. The configuration of the system evolves according Newton's Laws of motion. The proton dynamics evolves according to a probabilistic Monte Carlo selection of discrete microstates. Each titration site is

allowed to occupy many discrete microstates, and the microstates themselves are described with atomic detail.

WHAM allows us to weave together simulations generated under a wide range of conditions and hence gives us an accurate description of the density of states, which can then be used to give us a complete range of thermodynamic results.

Atomic detail Molecular Dynamics, Monte Carlo selection of discrete microstates, and WHAM work together like a trinity with symbiotic unity. The members of the trinity are symbiotic. One member of the trinity depends on the other members for its functional completeness, and cannot be productive without the other two. The product of this unity is a good description of the density-of-states. Consider the following examples of this symbiosis.

The regular atomic detail MD force field description is not truly atomically detailed. It can only be truly atomically detailed if it incorporates a discrete protonation microstate model for its titratable sites. But classical MD is not able to sample these atomic detail discrete protonation states, so Monte Carlo is needed for the selection of these discrete atomic detail protonation states. Atomic detail MD typically samples only a small region of the energy landscape, which means that its trajectories can't yield very good density-of-state descriptions, which means that the ensuing thermodynamic calculations are limited in accuracy and scope. The problem of better sampling is alleviated somewhat by incorporating MC selection of the discrete microstates, but the real breakthrough in sampling with our method comes by way of a "simulated annealing ensemble". What we specifically mean by this is that we generate equilibrated trajectories under a wide range of conditions. We are able to weave together the information from the many trajectories with WHAM.

The MC selection process is central to proper sampling of the protonation states. However MC by itself is not enough to allow the system to access all of the protonation states. It is the high temperature trajectories of the "simulated annealing" data set that works together with the MC selection process to properly allow the system to access all of the microstates.

WHAM is used to weave together the information from all of the simulations that are generated under different conditions. Doing this, it gives us a good description of the density-of-states, hence allows for a wide range of thermodynamic results of good accuracy.

1.13.2   Advantages of the MD/MC algorithm

1.13.2.1 Staying in equilibrium

In the section "survey of modeling proton dynamics", 1.12, we saw that some of the explicit solvent methods use a "continuum" of microstates. Using this continuum pathway, the system is forced, by free energy perturbation methods, to go from one protonation state to the next. The free-energy difference between the protonation states is calculated in this way, and a resulting pKa can be calculated. There are several problems with this method. There are large energy barriers to cross in going from one protonation state to the next. This is because of the solvation shell formed by the atomic detail water model, which forms in response to the electrostatics of the protonation state. Crossing this barrier in going from one protonation state to the next means a rearrangement of the surrounding waters. In other words, the system crosses phases. This means that such a free energy perturbation pathway runs a high risk of not being in equilibrium at all points of the pathway. Free energy perturbation is only accurate if the system is in equilibrium for the whole pathway. Enforcing equilibrium for such a system means using both a very slow pathway, and also going back and forth between the protonation states many times (for good enough statistics). Having to do this challenges this method in terms of computational feasibility.

In our method, our simulations are in equilibrium all of the time. We use discrete microstates. We achieve rigorous protonation state sampling with the combination of MC selection and a "simulated annealing ensemble" to high temperatures. Our relative free energies are derived NOT from driving the system from one protonation state to the next. Rather they are derived by weaving together all of the information from trajectories generated under a wide range of conditions, to get a good density-of-states description.

1.13.2.2 Improved electrostatic modeling

Much has been said in previous sections about the importance of good electrostatic modeling. Proton dynamics involves moving around protons of 1 electron charge. So attempts to take full advantage of atomic detail for a detailed electrostatic description will be handicapped without incorporating proton dynamics.

A discrete microstates model is better for electrostatics. In the discrete model, the sites can have only two ionization states. The proton is either there or not there. Using a continuum of ionization states cannot capture correlations between proton dynamics and the configurational dynamics. In other words, ½ a proton on a site is not a substitute for a site being protonated 50% of the time, and deprotonated 50% of the time. Because there is no such thing as ½ a proton, the discrete ionization state model is also much more intellectually satisfying.

Not only does our model have discrete ionization states, but also the ionization state may have several discrete microstates! These microstates distinguish them selves by the orientation of the proton relative to the titration site, and the exit or entry point of the proton to or from the titration site. By exit and entry points, I mean the entry point of the proton upon protonation, or the exit point of the proton upon deprotonation. This yields better electrostatics for the following reasons. In titratable sites networks, even those as simple as two site networks, the titration sites in the hydrogen-bond network are usually closer than 6 Angstroms from each other. So the location of the exit or entry point of protonation on a site could make a big difference to the energetics of the hydrogen bonds. The exit or entry point location on one side of the titration site compared to the opposite side of the site could mean a difference of 2 Angstroms in the length of the consequently formed hydrogen-bonds (see Figure 21: Cysteine microstates). This clearly would make a big difference in the energetics of the system. Our implementation of discrete ionization states and discrete microstates therefore provides an electrostatic description of dynamic biological systems that takes full advantage of the atomic detail force field model.

Most of the macroscopic methods use discrete ionization states. However there are advantages to using atomic detail and explicit solvent as opposed to macroscopic descriptions of the system. Macroscopic descriptions require the assignment of dielectric constants to the solute and the solvent. These assignments are a challenge, because there is no well-defined way of assigning appropriate dielectric constants to the protein or to regions of the protein.

1.13.2.3 More accurate trajectories

In the previous section we discussed the improved electrostatic description that goes with the proper inclusion of proton dynamics. This will in turn yield more accurate trajectories. In the world of computer simulation, more accurate trajectories are usually only meaningful because they imply more accurate density of state descriptions and consequently more accurate

thermodynamic calculations. However more accurate trajectories are useful in their own right. Recall that concerted motion is an important part of the function of many biomolecular systems. The inclusion of atomic detail proton dynamics, and any trajectory improvements that result, may well yield concerted motions in the simulations that shed light on the correlation between concerted motion and function.

1.13.2.4 More accurate configurational sampling

Atomic detail MD typically samples only a small region of the system's energy landscape. Unlike standard MD protocol, our MD/MC algorithm will allow for dynamic protonation assignment during the course of the simulation. This in turn will affect configuration-protonation state correlation, hence broadening the sampling for more accurate thermodynamic calculations.

1.13.3   Advantages of the Simulated Annealing Ensemble

Our Simulated Annealing Ensemble consists of equilibrated system trajectories generated under a wide range of conditions. Simulated Annealing in computational simulations is generally thought of as the process of heating up and cooling down ONE system for equilibration or other reasons. This implies that systems undergoing such a process are not in equilibrium, but rather are having gradual state variable changes imposed on them. For this reason we use the term "simulated annealing ensemble" as opposed to simply "simulated annealing". In our method we use an ensemble of systems that differ from one another only in that they are generated and EQUILIBRATED under a wide range of conditions. Each simulation in the ensemble is completely EQUILIBRATED, and its state variables stay fixed during the course of its generation.

The "simulated annealing ensemble" is necessary for the following reasons. Because we use an explicit solvent, atomic detail, discrete microstate model, efficient sampling of the ionization states requires that the system be simulated at elevated temperatures (>1000K). At 300K, the system in one ionization state will take an unreasonably long time to access the other ionization state. It is more efficient to elevate the temperature to get frequent transitions for good statistics.

However a single high temperature trajectory is not good enough. We hope to use the simulations to calculate thermodynamic quantities at room temperature and pressure. The high temperature trajectories alone, fed into WHAM, will give a density of states description that may be sufficient for calculating thermodynamic quantities at >700K, but not for 300K, 1atm. Lower temperature trajectories, such as those at 300K and 1atm, need to be combined in WHAM with the higher temperature trajectories to yield a density of states description appropriate for "room condition" thermodynamic calculations.

However a single high temperature trajectory, and a single low temperature trajectory are not good enough. In order for WHAM to weave together the high and low temperature information in an efficient and meaningful way, there needs to be overlap of the "effective energy" histograms of the simulations. So we need many trajectories generated under conditions that span the whole range, from the highest temperature trajectory all the way down to 300K 1atm. For good histogram overlap, this usually numbers about 200 simulations. Hence our term, "simulated annealing *ENSEMBLE*"

Although the deployment of the "simulated annealing ensemble" was one of necessity, there are many advantages to doing this. Because the system is represented as an ensemble of trajectories generated under widely different conditions, the configurational sampling is excellent, and WHAM weaves together a very powerful description of density of states. This power for calculating thermodynamic parameters is demonstrated in plots such as the one shown in Figure 32.

All of our trajectories are equilibrated. This was mentioned before, but it is worth repeating in the context of comparison with some explicit solvent methods. Some other explicit solvent methods attempt to cross the water shell barrier by driving the protonation state of the system with Free Energy Perturbation. The problem with such an approach is that the system is not equilibrated along the entire pathway because the system has to cross a phase due to water reorientation.

1.13.4 Advantages of using WHAM with MD/MC trajectories

Our WHAM theory is rigorous and powerful. It allows for bringing a wide range of information to bear in the pursuit of accurate thermodynamic results. In the previous section we have already

discussed how WHAM brings information, from trajectories generated under wide range of conditions, to bear on achieving a good density of states description. Now we will see how WHAM brings information of different *types* to bear on achieving good thermodynamic results. For example, consider a typical pKa calculation, conducted using typical methods.

$E^{pot}_{protonated}, E^{pot}_{deprotonated}$ $\longrightarrow$

| MC protonation state selection | $\longrightarrow$ | Occupancy statistics |

$\downarrow$

| pKa calculation |
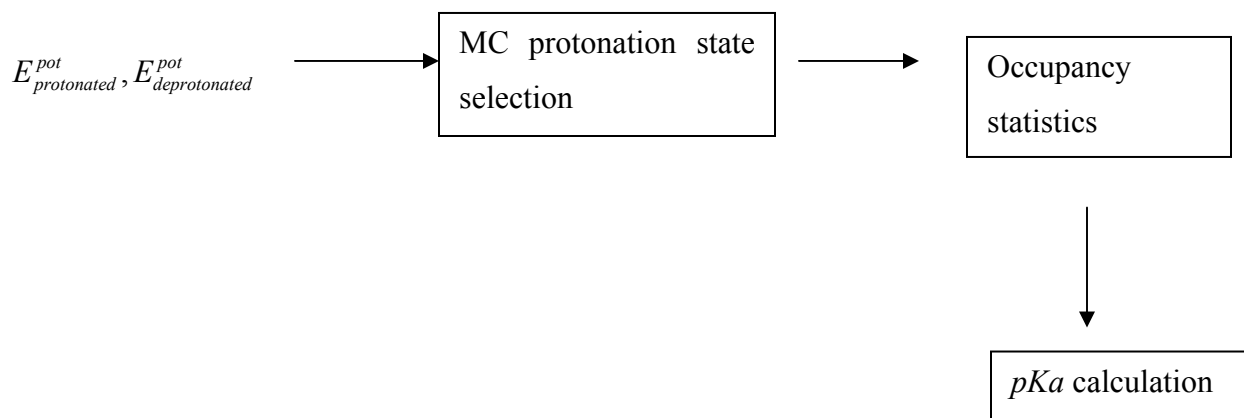
The potential energies of the various protonation states are measured. The MC algorithm then chooses a protonation state based on those potential energies. This process repeated many times yields occupancy statistics, from which *pKas* are derived. So the *pKas* are a direct result of the occupancy statistics, and are an indirect result of the system energies. However, using WHAM gives us the following advantage.

$E^{pot}_{protonated}, E^{pot}_{deprotonated}$ $\longrightarrow$

| MC protonation state selection | $\longrightarrow$ | Occupancy statistics |

$\downarrow$
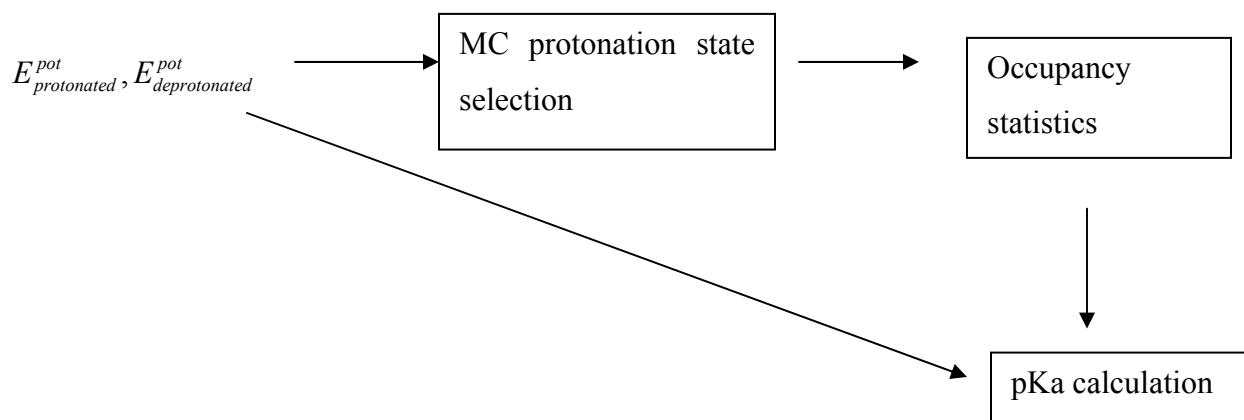
| pKa calculation |

The calculated pKa numbers are derived directly from both *types* of information. Both the system energies and the occupancy statistics go directly into the pKa calculation (See equations in section 3.8).

And of course, we can blend information from different simulations:

$$E^{T_1}_{protonated}, E^{T_1}_{deprotonated} \rightarrow \boxed{\text{MC protonation state selection}} \rightarrow \boxed{\text{Occupancy statistics}}$$

$$\boxed{\text{pKa calculation}}$$

$$E^{T_2}_{protonated}, E^{T_2}_{deprotonated} \rightarrow \boxed{\text{MC protonation state selection}} \rightarrow \boxed{\text{Occupancy statistics}}$$
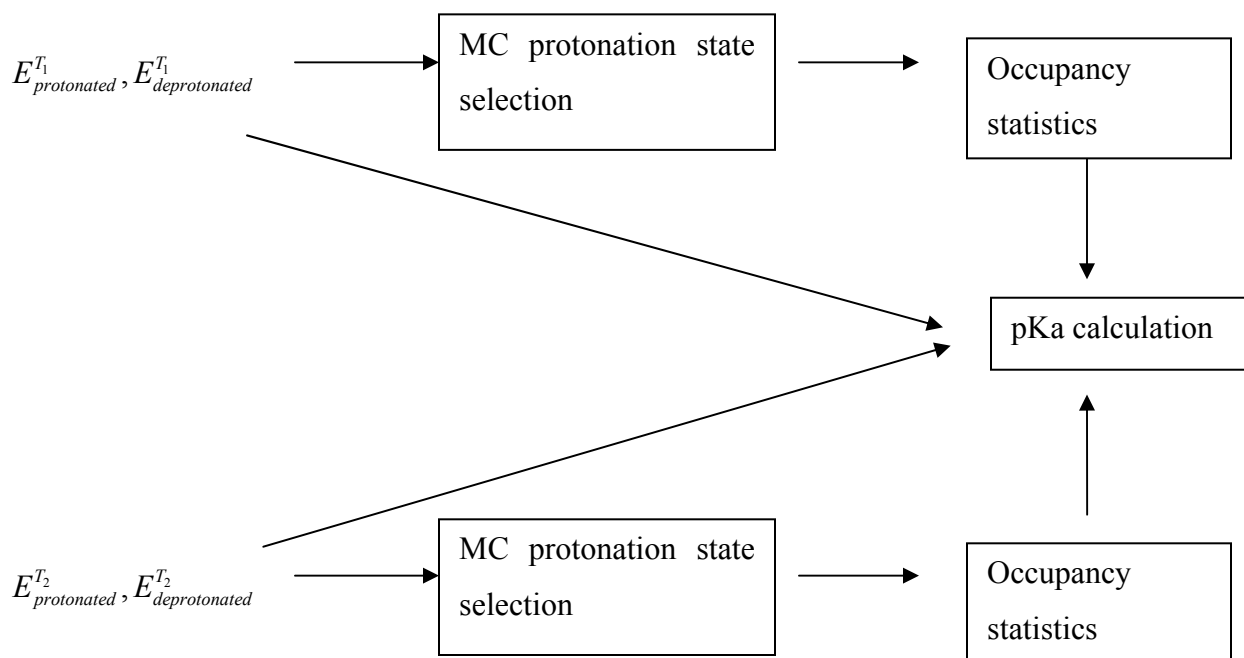
This power of WHAM to use information from a wide range of trajectories and also different types of information is applied not only to pKa calculations, but THE FULL RANGE OF THERMODYNAMIC CALCULATIONS.

1.13.5 Advantages of user friendliness

Much work has gone into making the generation of the MD/ MC trajectories, the generation of these trajectories for a wide range of conditions required for the simulated annealing ensemble, and the WHAM analysis of these trajectories as seamless as possible. Anyone familiar with using Unix type operating systems, Perl scripts and popular MD packages (such as AMBER's sander, CHARM, or NAMD) will find our package easy to use. Our algorithms have been tested on both 32-bit and 64-bit platforms and runs on Beowulf clusters and several different types of supercomputer architectures.

The first step is the creation of special parameter files for the MD/MC algorithm. These files have enough information in them to allow for the generation of the right force field

parameters for any protonation state of the system. The process for the creation of these parameter files is easy and well defined. The easiest way to do it is to simply use the AMBER8 Xleap parameter database that we provide and follow a few easy well defined steps.

The MC algorithm is fully integrated into the AMBER7 sander MD algorithm, so the transition from the MD sub cycle to the MC sub cycle is completely seamless to the user. Perl scripts generate the hundreds of input files required for the simulated annealing ensemble. Perl scripts are also able to automate the equilibration process of the trajectories within the simulated annealing ensemble. There are also Perl scripts that generate the job files for submission to the queue of the computing resource used.

There is an easy and well-defined protocol for how to input all of the trajectory information for WHAM to process. The MD/MC and WHAM algorithms were designed to work together so that there are NO formatting issues. WHAM completely understands the format of the MD/MC output.

The convergence time required for WHAM may require the submission of a series of jobs, where each job is short enough to reduce queue waits. There are Perl scripts that automate this process by chaining jobs together and arranging/creating relevant output and input information for exiting jobs and successive job respectively.

One of the most exciting and important user friendly features of our method is the ease with which the user can experiment with different water models, different force field parameters, and new exciting features like polarization. This is possible because a lot of the flexibility and features of the AMBER sander module and the AMBER Xleap module are retained. There is an extensive number of water models (including cutting edge untested water models), force field parameter databases (including some cutting edge untested polarization parameters), and features that come with the AMBER package, and they are almost all available for use in our method.

2.0 INTEGRATING MD, MC AND WHAM

The Hamiltonian of our system is described by $H = \sum \frac{p^2}{2m} + U(\bar{x}, \underline{\eta})$. The Potential energy is a function of both "configuration" $\bar{x}$ and protonation state $\underline{\eta}$. $\bar{x}$ represents the configuration of the whole system *except the titratable sites.* $\underline{\eta}$ primarily describes the protonation state of the system, and also the configuration of the titratable sites. In our description, protonation states are characterized by both force field parameters (like charge) and configuration of the titratable site.

One simulation cycle of our MD/MC code consists of an MD sub-cycle and an MC sub-cycle. The MD sub-cycle uses a fixed protonation state $\underline{\eta}$ and allows the "configuration $\bar{x}$" to evolve, enforcing constant N, T, and P. In the MC sub-cycle, the Monte Carlo sweeps act on the system with a fixed "configuration $\bar{x}$". Protons are allowed to jump on or off the titratable sites (allowing $\underline{\eta}$ to be updated), thus enforcing constant pH. Together one cycle simulates a true "Biochemical Ensemble", constant N, P, T and pH.

Equilibrated trajectories are generated at a wide range of temperatures and pH.
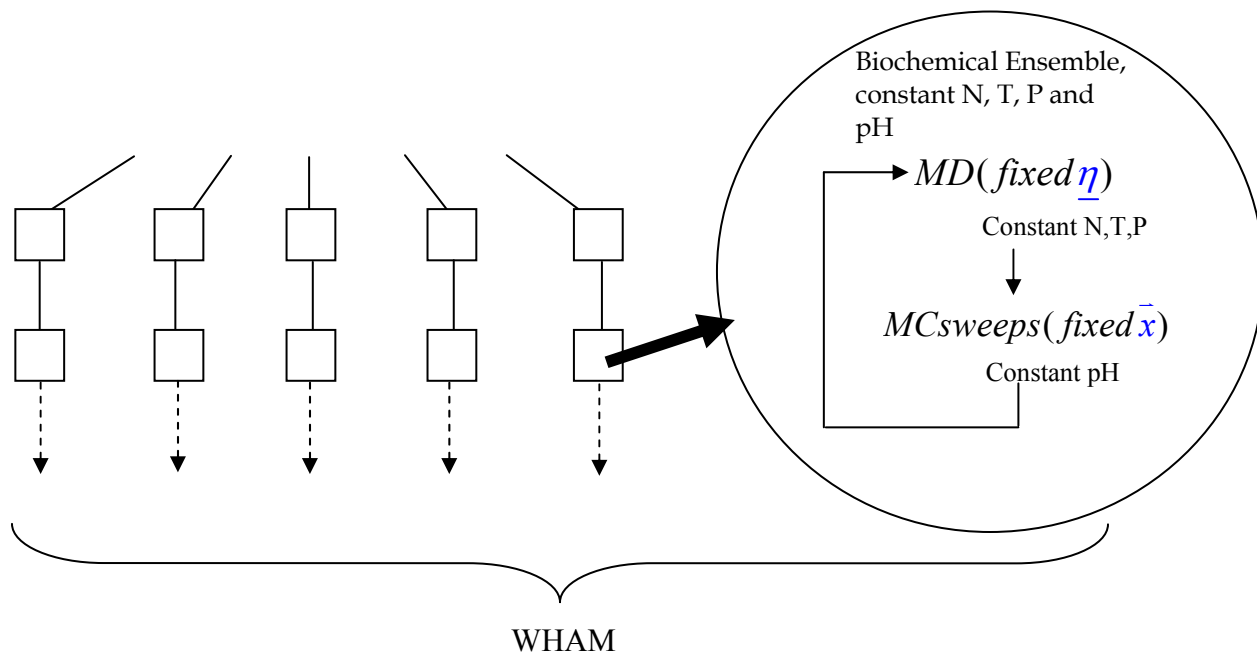
The information from all snapshots of all trajectories is fed to our WHAM algorithm, which allows us to weave all of this information together to give us a good description of the densities of states, and hence allows us to calculate a wide range of thermodynamic parameters such as free-energies and *pKas*.

Throughout this discussion of our MD/MC theory, we will use the example of a solvated system consisting of a single titratable Cysteine amino acid. This helps make the discussion a little less abstract.

2.1 RESERVOIRS THAT INFLUENCE OUR SYSTEM

There are three "baths" or reservoirs for our system, a temperature bath, a pressure bath and a *pH* bath corresponding to three intensive parameters, T, P and $\mu_{\mathrm{H}}(= -\frac{1}{\beta}\log_e 10\, pH)$. *T, P* and μ_H are the system temperature, pressure and proton chemical potential respectively. The temperature reservoir interacts with the system by influencing the system's particle velocities. If

the velocities or temperature of the system particles is lower than the target temperature (which is the temperature of the temperature reservoir), the system velocities will be scaled up in an attempt to get the system to the target temperature. Similarly if the temperature of the system is higher than the target temperature (the temperature of the reservoir), the velocities of the system will be scaled down. The second bath is the pressure bath. This reservoir interacts with the system by influencing the system volume. If the instantaneous pressure of the system is higher or lower than that of the target pressure (which is the pressure of the pressure reservoir) then the volume of the system is increased or decreased as it attempts to get the system pressure to match the target pressure. The *pH* bath or reservoir interacts with the system by exchanging protons with the system.

system   reservoir

If the energy of a protonated titration site is higher than the energy of its deprotonated state, then, if averaged over a sufficient period of time, the proton is removed from the titration site to the reservoir. If this situation is the same for many of the titratable sites of the system, there will be a net flow of protons from the system to the reservoir.

Similarly, the opposite proton flow occurs if the deprotonated sites cause the system to have higher energies relative to the protonated versions.

The presence of the temperature, pressure and the *pH* reservoirs is affected though the temperature, pressure and *pH* state variables, which are selected at the beginning of a simulation and fixed during the course of the simulation. A system that evolves under conditions of

constant temperature and pressure due to interaction with a temperature and a pressure bath is said to be in the *NPT* ensemble. But our system is in a different ensemble, the N, P, T, μ_H ensemble, which we describe as the "biochemical ensemble". At first glance the N, P, T, μ_H ensemble seems to have one variable too many, but the conjugate partner of the proton potential μ_H is L, the proton count of the proton reservoir. N is the atom count of all atoms of the system except the titratable protons, i.e. $N_{titratable\ protons} \notin N$. Rigorously, the total number of atoms in the system is $(N + N_{titratable\ protons} - L)$, which is not constant, even though N is constant. Our ensemble is therefore a mixed ensemble that consists of an *NPT* ensemble and a $PT\mu_H$ ensemble. Hence we describe our biochemical ensemble system as an $NPT\mu_H$ ensemble, which is consistent with the system atom count of $(N + N_{titratable\ protons} - L)$. Our rules for calculating thermodynamic quantities in this $NPT\mu_H$ ensemble are slightly different than the ones for the *NPT* ensemble, so we go through those rules in the next chapter (chapter 3.0).

2.2   POTENTIAL ENERGY FUNCTION & SYSTEM MICRO-STATES

The titratable hydrogen atom is connected to Cysteine by a three-fold dihedral bond. This means that if this hydrogen were forced to rotate about the sulphur-carbon bond, it would pass through three minima that are separated by 120 degrees. The three minima are where the H-S-C-H dihedral bond is 180 degrees (the state one minima), 300 degrees (the state two minima) and 60 degrees (the minima of state three). The dihedral potential function is continuous. This means that the dihedral can take on an infinite number of values, which means the titratable hydrogen can have an infinite number of positions. What defines a protonated state, and what distinguishes one microstate from the next is not some specific position of the hydrogen or some single value of the dihedral, but rather a range of values of the dihedral. State one is defined as such if the titratable hydrogen is positioned such that the H-S-C-H dihedral has a value in a range that is greater than 120 degrees and less than 240 degrees. State two is defined as such if the dihedral has a value in a range that is greater than 240 degrees and less than 360 degrees. State

three is defined as such if the dihedral has a value in a range that is greater than 0 degrees and less than 120 degrees.

Consider for example the four microstates of Cysteine.

state 0(charge=-1ec)   state1(charge=0ec)   state2(charge=0ec)   state3(charge=0ec)
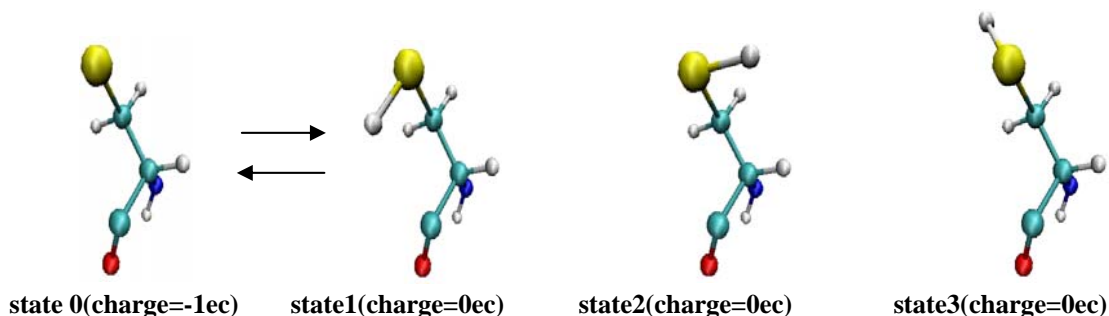
Figure 21: Cysteine microstates

State zero and state one are different ionization states. They share the same configuration, but the atom parameters, partial charges and bond parameters are different. The most significant difference is that the titration hydrogen of state zero is a ghost atom, having no partial charge and no van der Waals parameters. If states zero and one were the only protonation states of Cysteine, it would suffice if η carried no configuration information. However η must carry configurational information to distinguish between states one, two and three. These three protonation states are all protonated, these three states all share the same partial charge, atom and bond parameters. The only difference between them is that the titratable hydrogen is oriented differently relative to the rest of the Cysteine.

The potential energy of our MC/MD system, $U(\vec{x},\eta)$, is a function of system configuration and protonation state of the system. This is different from the usual molecular dynamics potential energy function, $U(\vec{x})$, which is a function of configuration alone. In our potential energy function, $U(\vec{x},\eta)$, x almost means the configuration of the system, but not quite. It means the configuration of all of the system except for the titration sites. η describes the protonation state of the system and the configuration of the titration state. Together, x and η describe the configuration of the whole system and the protonation state of the whole system. The reason why η describes both protonation state and also a little configuration information is because some of the microstates differ from each other only by configuration differences of the titratable group.

One of our MC/MD cycles consists of two sub-cycles. One sub-cycle is the Molecular Dynamics sub-cycle the other is the Monte-Carlo sub-cycle. During the molecular dynamics sub-cycle, the force-field parameters stay fixed and the atom positions are updated according to numerical integration of Newton's laws of motion. During this sub-cycle, the system is in contact with the temperature and pressure reservoirs for enforcing constant temperature and pressure conditions on the system. Because the force field parameters are fixed during this sub-cycle, the Cysteine cannot change ionization states during this sub-cycle. So if the system is in the deprotonated state (state zero) at the start of the molecular dynamics sub-cycle, it will stay deprotonated for the duration of the molecular dynamics sub-cycle. Similarly if the system is in one of the protonated states, it will stay protonated during the course of the molecular dynamics sub-cycle. However, it is possible for the system to change from one protonated microstate to the next. This is a rare occurrence for the following reasons. The first reason is that during the molecular dynamics part, the titratable hydrogen usually rattles around near the bottom of the potential energy well of the dihedral. Energy fluctuations of the system would have that hydrogen cross those barriers and go from one protonation state to the next. For room temperature simulations, that would happen about once every three thousand MD steps. However, recall that the molecular-dynamics sub-cycle is only twenty steps long. Seldom will there be a transition from one protonation state to the next within only twenty molecular dynamics steps, even at higher temperatures. This is our justification for saying that during the molecular dynamics sub-cycle, the x in $U(x,\eta)$ is updated but η is fixed.

During the molecular dynamics part of the sub-cycle, the pH component of the effective energy, $-\log 10\,pH \bullet L$, is ignored because this component does not change from one step to the next. This is because no ionization state changes take place during the molecular dynamics sub-cycle, so including this term in the energy calculations is useless.

During the Monte Carlo sub-cycle, the system is in contact with the pH reservoir, or a proton bath. The effective energy, $E_{pot} + PV - L \bullet \log 10\,pH$, is calculated for each state. Let's call them $E0, E1, E2, E3$. Boltzmann factors are then assigned to each state, $eE0, eE1, eE2, eE3$, where $eE0 = e^{-E0}$, and they are normalized so that the sum equals one. Each state is then assigned a range between 0 and 1, and the value of the range equals the value of the normalized Boltzmann factor.

0   1

state 0   state 1   state 2   state 3
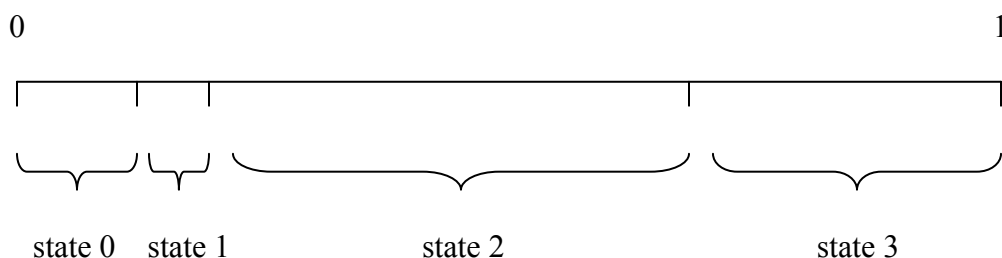
A random number generator then generates a number between zero and one, thereby selecting a state.

The Monte Carlo algorithm has to generate the states before their effective energy can be measured. The Cysteine system will enter the Monte Carlo sub-cycle in one of the four states. The other states are generated using a combination of parameter swapping and rotation of the titratable hydrogen dihedral bond (the $-C-S-$ bond). How this is done is important in addressing the issues of equilibrium and detailed balance, so I will now give more detail about this process. If the system enters the sub-cycle in state zero, the potential energy and other virial related routines are called to calculate the potential energy and the volume of the system. These are the same routines that are called during the molecular dynamics sub-cycle. The effective energy is then calculated as $E0 = E_{pot} + PV - 1.0 \cdot \log 10 \cdot pH$. The 1.0 factor is in the $-1.0 \cdot \log 10 \cdot pH$ term because the state is deprotonated, which means that one (1.0) proton is present in the proton bath, and absent from the Cysteine. State one is then generated by changing the partial charges, atom parameters and bond parameters to those of protonated Cysteine. However, no configurational changes are made. Recall that the configuration, which is the atom positions, of state zero is the same as that of state 1 (see Figure 21: Cysteine microstates). The potential energy and other virial routines are then called to calculate the potential energy and the volume of the system, and E1 is calculated. Note that E1 is protonated, so the proton is present on the Cysteine and absent from the proton bath, so there is no pH term, so $E1 = E_{pot} + PV$.

State two is then generated by rotating the titratable hydrogen's dihedral 120 degrees. Recall that in state one, there are an infinite number of positions that the hydrogen atom can occupy, partly because the dihedral can have an infinite number of values between 120 degrees and 240 degrees (the hydrogen can also occupy an infinite number of positions by virtue of

degrees of freedom orthogonal to the dihedral coordinate, such as the H-S bond coordinate or the H-S-C angle coordinate). During the molecular dynamics sub-cycle, most of the time, the hydrogen atom will be spent rattling around near the bottom of the potential well, but it is almost impossible that it will be exactly at the bottom of the well. So by rotating 120 degrees to generate state two from state one, we ensure that the position of the titratable hydrogen relative to the dihedral potential minima is the same in state two as it was in state one. Doing this preserves detailed balance and equilibrium. The usual virial routines are again called to calculate the potential energy and the volume, and E2 is calculated. In like fashion, state three is generated by another 120 degree rotation in the same direction, and E3 is calculated.

2.3   OUR EFFECTIVE ENERGY COMPONENTS

The dimensionless effective energy of our MC/MD models is $\frac{1}{kT}(E_{pot} + PV) + L\mu_H$. k and T are the Boltzmann's constant and temperature respectively. E_{pot} is the potential energy of the system. P and V are the pressure and volume of the system respectively. L is the number of protons in the proton bath. μ_H is the proton chemical potential of the proton bath.

The last term, $L\mu_H$ is the term that the constant pH functionality contributes to the effective energy. L, the number of protons in the proton bath, increases as the number of deprotonated states increases. μ_H, the chemical potential of the proton bath, is a measure of how energetically hard or easy it is to place or remove a proton to or from the proton bath. μ_H and L are conjugate variables. The chemical potential μ_H is related to the pH as follows. $\beta\mu_H = -\log 10 pH$.   Log10 is simply the constant 2.303…. and pH is the pH of system. The pH (and μ_H) of the system is a state variable, on the same footing as temperature and pressure. The pH, like temperature and pressure, is selected and set at the beginning of each simulation and stays fixed throughout the simulation so that equilibration can be attained. L is a configurational variable, like volume and potential energy. It may change (like E_{pot} and volume) during the course of the simulation regardless of if the system is in equilibration or not.

It will definitely change if the selected pH for the simulation is close to the pKa of any of the system's titratable sites

2.4   GHOST ATOMS

2.4.1   Introduction to Ghost Atoms

In the protonated states of Cysteine (states one, two and three), the force field parameters used are the same as those of the protonated form of Cysteine in the Amber8 force field. However, for the deprotonated state, state 0, the force field parameters used differ slightly from those of the deprotonated form of Cysteine in the Amber8 force field. The differences are related to the fact that our deprotonated system has a ghost-hydrogen, and the original Amber8 deprotonated Cysteine has no such thing. Our ghost hydrogen has mass and does interact with the rest of the solute via the linear, angle and dihedral bonds. However it has zero partial charge and its van der Waals parameters are zero. We will now discuss the effects of the differences between our deprotonated model, which contains ghost atoms, and the deprotonated Amber8 model, which has no ghost atoms. We will also discuss our justification for using ghost atoms. First we will state the differences between the deprotonated Cysteine models.

2.4.2   Summary Comparisons with our Ghost Atom Model

Our deprotonated Cysteine parameters are the same as those of the Amber8 deprotonated Cysteine, with an additional ghost atom and additional parameters relating to that ghost atom. The ghost hydrogen has the same mass as a regular hydrogen atom, 1.008 atomic mass units. There is no partial charge on the ghost atom, so there is no electrostatic interaction. The van der Waals parameters of the ghost hydrogen are zero, so there is no van der Waals interaction of the ghost hydrogen. All of the bond parameters, that is, the linear bond parameters, angle bond parameters and dihedral bond parameters connecting the ghost hydrogen in the deprotonated state, are all the same as the corresponding bond parameters of the hydrogen of the protonated

version. In the original Amber8 deprotonated Cysteine, these parameters ***do not*** exist because there is no ghost hydrogen attached to the sulphur. What follows is a discussion of our reasons and justifications for setting up the deprotonated model in this way.

2.4.3 Justifications For Using Ghost-Hydrogens

In a real system, there is no ghost hydrogen and the departing proton will go to another position in the solute or in the solvent (possibly forming H_3O^+). Since there may be over ten-thousand water molecules in a typical model, we cannot model this kind of action for it would mean making every water molecule of the solvent titratable, resulting in a much less computationally feasible model.

Implementing a scheme for creating and eliminating protons is an intuitive way of modeling ionization state transitions. However recall that our MD/MC algorithm is a modification of the Amber (sander) code that is not designed to model breaking/making of bonds required to add or remove atoms to or from a system. It is relatively easy to make modifications to the algorithm to make the proton go away, but hard to put it back. So instead of making it go away, we change it into a ghost atom which is easy to put back.

In our deprotonated model, the ghost proton has the proper mass and the usual bonding parameters, but there are no electrostatic or van der Waals interactions because the partial charge is zero and the van der Waals parameters are zero. Considering that our ghost-hydrogen has mass, and the bond parameters related to the ghost hydrogen retain their values, what about the bond vibrational energy components for the ghost atom in our deprotonated model, which would not exist in the real system? According to Equipartition theory, the bond vibrational energy contribution of the titratable hydrogen atom is on average $3 \times \frac{1}{2}kT$ (3 degrees of freedom). In a real system, the proton leaves the titratable site to go somewhere else, let's say to form H_3O^+ in the solvent. So the 3/2 kT bond vibrational energy term simply leaves the titratable site of the Cysteine solute and goes somewhere else, in this case, the solvent. But it does stay in the equilibrated system! This is consistent with our method, where the 3/2 kT stays in the system. Where our model differs from the real system is that in our model the 3/2 kT stays in position at

the deprotonated titration site on the solute, instead of going to another position or going to the solvent. This imprecision of our model for positioning the 3/2 kT will therefore have no effect on our final thermodynamic results.

2.5 EQUILIBRATION AND IONIZATION STATE TRANSITIONS

What about equilibration disturbances during ionization state changes? It may seem intuitive that when the system changes ionization states, equilibration would be hard to determine. For example, if during an MD sub-cycle, the system is equilibrated and is in the deprotonated state (state zero). Then the MC chooses state one, a protonated state. Then the next MD sub-cycle would see reorientation of the system, including the solvation shell around the titration site. It seems like this process would render a determination of the equilibrium of the system very difficult. However equilibrium can be determined for our systems for the following reasons.

Equilibrium determination can only be defined in terms of the time scale of observation. A cup of room-temperature water on a table in a humid room is in equilibrium, and can be determined to be in equilibrium, assuming the time scale of observation is seconds or minutes. But if the time scale of observation is too short (less than 10^{-15} seconds), it would not appear equilibrated. This is because the short observations would only capture a fluctuation, or a few system fluctuations, such as water molecules bursting the surface to dissipate into the air. But observation times on the order of seconds would reveal equilibrium between water molecules entering the air and molecules entering the liquid. The time scale of observation has to be long enough to capture many fluctuation events, in order to accurately average. In other words, the time scale of observation has to be several orders of magnitude longer than the time scale of the system fluctuation phenomena. In our MD/MC systems, we examine the equilibrium of the system over a period of nanoseconds (10^6 MD steps), 50,000 MC sweeps and thousands of ionization state transitions. Typical ways of determining equilibrium for an MD system will be to look at Potential energy, density, temperature and pressure over many picoseconds (> 100,000 MD steps) and see if there are drifts in these values, which would indicate that the system is not yet equilibrated. We use the same criteria for our MD/MC systems. We accept a system as equilibrated if it is stable during observation lengths of the system for over 10^6 MD steps, 10^5

MC sweeps and several thousands of ionization transitions. The latter (several thousands of transitions) dwarfs the others in importance. This is because 10^5 MC sweeps may seem like a lot but not all MC sweeps cause ionization state changes. Many MC sweeps leave the system unchanged. It is the number of transitions that say how many ionization "fluctuations" have occurred. As mentioned before, our observations must span many of these ionization "fluctuations" and they do, spanning several thousand of them.

3.0 WHAM THEORY, DEVELOPED AND EXTENDED

3.1   INTRODUCTION

WHAM is the adaptation of the Weighted Histogram formalism for biomolecular applications (S. Kumar, 1992, 1995)[63,64]. We needed to develop the method further to take into account some new ideas, and so we also undertook a revision of the notation to make the method easier to understand. Many of these new ideas were inspired by our work of adapting WHAM for use with constant pH simulations.

A related motivating factor for revision is the recognition that many of the difficulties encountered in learning "histogram" methods turned out to be difficulties with the underlying statistical mechanics. It is often stated that biomolecular problems can, in principle, be addressed *via* the rigorous application of the principles of statistical mechanics, but the "how to" is usually unstated because of the perceived numerical difficulties. Here, we emphasize that histogram methods facilitate a very general translation between statistical mechanics and numerically computed results; a clear understanding of these connections also facilitates a rational evaluation of the numerical difficulties.

We therefore begin with a review of the relevant statistical mechanics in the *NPT* ensemble because it is the ensemble most suitable for biological structure-function correlations. Then towards the end, we will extend it to our *NPTμ$_H$* biochemical ensemble (see section 2.1). The focus here is on the concept of the density of states and its related formalism because this is the "translation" between rigorous statistical mechanics and numerical results obtained *via* histogram methods. We show that the problem of calculating all the relevant thermodynamic parameters can be developed in the density of states formalism. However a rigorous, direct calculation of the density of states is not possible for most biomolecular systems. We then reach our central point, however; histogram methods allow one to estimate the density of states from

molecular dynamics or Monte Carlo simulations. This approach facilitates understanding the assumptions that underlie the connections between a set of trajectories (histograms) and the thermodynamic results based on them; which, in turn, enables accurate estimates of the *statistical* errors inherent in the calculation and provide guidance on which further calculations are needed to reduce those errors.

As mentioned above, one of the justifications for overhauling our notation and theoretical description was the adaptation of WHAM to constant pH simulations. The first sections describe our theory as relates to the usual MD simulation ensemble, the *NPT* ensemble. Then, in the last sections, our theory is reviewed and adapted for constant *pH* simulations in our mixed $NPT\mu_H$ biochemical ensemble (please see section 2.1). We do it this way so that the constant *pH* discussion will all be in one place, and not get lost amongst the general discussion.

3.2   THE DENSITY OF STATES

We begin with the molecular system in the *NPT* ensemble. The Hamiltonian is

$$\mathbf{H} = \sum_{i=1}^{3N} \frac{p_i^2}{2m_i} + U'(\vec{x}) \tag{3.1}$$

where N is the total number of atoms in the system and p_i and m_i are the momentum and mass of the *ith* particle respectively. We assume that the potential energy, $U'(\vec{x})$, is a function of the atomic coordinates, $\vec{x}$, only *i.e.* the Born-Oppenheimer approximation[69]. The partition function for the *NVT* ensemble can be written as:

$$Q_{NVT} = \int e^{-\beta \left[U'(\vec{x}) + p^2/2m \right]} d^{3N} p \, d^{3N} x \tag{3.2}$$

where $\beta = 1/k_B T$, where k_B is Boltzmann's constant, $\beta = 1/k_B T$, and the *x*-integrals are over the volume V. To find the *NPT* ensemble (later we will deal with the $NPT\mu_H$ ensemble) we integrate the Q_{NVT} over V to get

$$Q_{NPT} = \int \left(\int e^{-\beta \left[U'(\bar{x}) + p^2/2m \right]} d^{3N} p \, d^{3N} x \right) e^{-\beta(PV)} dV \tag{3.3}$$

Now we introduce an energy variable U and insert the expression $1 = \int \delta(U - U'(\bar{x})) dU$ into the integral over the coordinates.

$$Q_{NPT} = \int e^{-\beta \left[U'(\bar{x}) + PV + p^2/2m \right]} \delta(U - U'(x)) d^{3N} p \, d^{3N} x \, dV \, dU \tag{3.4}$$

Rearranging the variables of integration and introducing the density of states

$$\Omega(U,V) = \int e^{-\beta U'(\bar{x})} \delta(U - U'(x)) d^{3N} x$$

we can write

$$\begin{aligned} Q_{NPT} &= \int e^{-\beta \left[U'(\bar{x}) + PV + p^2/2m \right]} \delta(U - U'(x)) d^{3N} p \, d^{3N} x \, dV \, dU \\ &= \int \Omega(U,V) e^{-\beta \left[U' + PV + p^2/2m \right]} d^{3N} p \, dU \, dV \end{aligned} \tag{3.5}$$

We now introduce the spatial partition function, Z

$$Z_{NPT} = \int \Omega(U,V) e^{-\beta(U + PV)} dU \, dV \tag{3.6}$$

which gives us the relationship

$$Q_{NPT} = \int \Omega(U,V) e^{-\beta \left(U + PV + p^2/2m \right)} d^{3N} p \, dU \, dV = Z \int e^{\beta p^2/2m} d^{3N} p \tag{3.7}$$

Expression (3.3) for Q is the conventional definition of the partition function for the *NPT* ensemble in terms of the Hamiltonian, while the next expression (3.5) introduces the density of states, Ω. The Boltzmann term, $e^{-\beta \left(U + PV + p^2/2m \right)}$, remains, but it must now be weighted by the density of states, Ω, to account for the multiplicity of states with the same U and V.

The Boltzmann term, $e^{-\beta \left(U + PV + p^2/2m \right)}$, remains, but it must now be weighted by the density of states, Ω, to account for the multiplicity of states with the same U and V. The Boltzmann term depends only on the values of the potential energy and volume, as well as the pressure and temperature; this is independent of the structural and molecular details of the specific system. Thus, all the system-specific thermodynamics is contained in the density of states, Ω.

The unnormalized probability density ρ, that the system will be found in the neighborhood of the potential energy *u,* and the volume *v,* is given by:

$$\rho(u,v) = \Omega(u,v)e^{-\beta(u+Pv)} \tag{3.8}$$

Thus the unnormalized probability density of a microstates characterized by potential energy, u, and volume, v, is proportional to a simple Boltzmann factor that relates the total potential energy $(u + Pv)$ to k_BT. The constant of proportionality, Ω, measures the "number" of states, really the density of states characterized by u and v. Note that Ω encapsulates all the system-specific information, and it is independent of P and T.

Note that the unnormalized probability density in equation (3.8) is the integrand in the equation for Z, equation (3.6), and that (for any physically reasonable system) this probability becomes vanishingly small for very large potential energies relative to β. This will become very important in the application of simulations below because it facilitates importance sampling, *i.e.* we do not need to know the entire density of states, only those regions that make a statistically significant contribution under the relevant temperature-pressure conditions. This is what makes the problem computationally tractable.

The unnormalized probability distribution is used for most of the discussion here because it is more convenient in the derivations that follow. The "real" (normalized) probability density is given by:

$$p(u,v) = \frac{1}{Z}\rho(u,v) = \frac{1}{Z}\Omega(u,v)e^{-\beta(u+Pv)} \tag{3.9}$$

3.3   PRINCIPLES OF STRUCTURE-FUNCTION CORRELATION

In biological (and chemical) applications one often wants to know not only the overall thermodynamics, one also wants to know how those thermodynamics depend on critical aspects of the molecular structure. Here, we show that the density of states can be readily generalized to address these questions in two related, but fundamentally different ways.

First, there is a broad set of questions relating structural parameters to functional (thermodynamic) quantities. Examples include: In DNA-protein interactions, considerable attention has been given to the role of deformability[70]. Is it easier to bend certain sequences of DNA than others? Are certain sequences naturally bent? Many proteins that interact sequence-

specifically with DNA introduce unique distortions[71]. How much energy do they cost? How does this depend on base sequence? In protein folding studies, particular attention has been given to buried surface area[72]. What is the change in heat capacity associated with the burial of a given area of polar or non-polar surface? In general, how do relevant thermodynamic parameters depend on a critical hydrogen bond length or the value of a central torsion angle?

Addressing these questions requires the introduction of generalized coordinates to quantify these structural parameters. We therefore adopt the following notation for a set of generalized coordinates:

$$\xi_i = \Xi_i(\vec{x}) \tag{3.10}$$

Where ξ_i is the ith generalized coordinate. It is a function of the atomic coordinates, given by $\Xi_i(\vec{x})$; where the principal restrictions on the Ξ's are that they are single-valued, integrable and, of course that they can be calculated from the atomic coordinates, $\vec{x}$. All of the examples cited above satisfy these criteria; in addition, they are continuous and differentiable, as is usually the case. Of course, the generalized coordinates are multidimensional and we refer to the set of generalized coordinates by the generalized vector, $\underline{\xi}$.

In this discussion, we adopt the convention that the atomic coordinate vector will be written as $\vec{x}$; it is a 3N-dimensional vector where N is the number of atoms in the system. Generalized vectors will be written with underscores, as exemplified by $\underline{\xi}$. It is also obvious that we are using color to distinguish between different types of variable. Here we use blue to denote configurational (atomic) variables, such as those that would apply to a single "snap-shot" of a molecular dynamics trajectory. The density of states must now depend on the generalized coordinate vector, $\underline{\xi}$, in addition to the potential energy and volume. This is a central point.

Other biophysical questions cannot be addressed simply by generalized coordinates; rather, they require the partitioning of the potential energy as follows:

$$U = \sum \lambda_i U_i(\vec{x}) = \sum \lambda_i u_i = \underline{\Lambda} \cdot \underline{U} \tag{3.11}$$

Here, we represent the potential energy as the sum of individual components, $U_i(\vec{x})$, each multiplied by a coupling constant, λ. We refer to the set of coupling constants and potential energy values *via* the vectors, $\underline{\Lambda}$ and $\underline{U}$, respectively, and their sum by the dot product

as in the third expression of equation (3.11). Note that there need be no connection between $\underline{\Lambda}$ and $\underline{\xi}$, discussed above; generally they will be of different dimensions.

Changing the Hamiltonian necessarily changes the state of the system; by introducing this generalization, the applicable statistical mechanical ensemble becomes the $N,P,T,\underline{\Lambda}$ ensemble. Note that $\underline{\Lambda}$ is an independent variable of state (actually a set of state variables), formally no different from pressure and temperature. It is also now obvious that we are extending our color convention and the independent state variables are shown in red.

Perturbation studies are one important class of problems that can be addressed with this form of a potential energy function; here, the chemical identity of a critical moiety is changed. An example would be a functional group in a ligand where one value of $\underline{\Lambda}$ would correspond to one ligand and another value of $\underline{\Lambda}$ would correspond to a chemically substituted variant. Site directed mutational alterations of a protein or base-analog studies of DNA would be addressed similarly. In all cases the goal of the effort would be to calculate changes in thermodynamic values such as the Gibbs free energy or the enthalpy associated with the chemical changes modeled by changes in the values of $\underline{\Lambda}$. Specific examples will be discussed below.

Another class of questions can be answered by partitioning the Hamiltonian and investigating the contribution of individual terms. One example would address the role of electrostatic forces by partitioning the Hamiltonian into electrostatic and non-electrostatic terms, each with its own coupling coefficient. The contribution of electrostatic interactions to thermodynamic parameters could then be calculated. Another example would address solvation by partitioning the Hamiltonian into solute-solute, solute-solvent and solvent-solvent terms. Similarly, inter-macromolecular interactions could be investigated by further partitioning of the Hamiltonian into A-A, A-B and B-B terms, where A and B are two macromolecules.

Finally, generalizing the potential energy, as indicated above, facilitates the sampling of high-energy regions, such as energy barriers. If the barrier is an accurately modeled transition state, then the methods described here can also be used to investigate kinetic phenomena. High-energy regions can be effectively sampled by introducing a U_j with a minimum in the region of interest; while the physically relevant states are those with the corresponding $\lambda_j=0,$ accurate statistics must be gathered from simulations with non-zero values of $\lambda_j.$ This is discussed more fully in later sections.

With this treatment of the Hamiltonian, *i.e.* equations (3.10) and (3.11), the density of states can be written as:

$$\Omega\left(\underline{\xi},\underline{u},v\right) = \int \Pi\delta\left[\Xi_i(\vec{x})-\xi_i\right]\Pi\delta\left[U_j(\vec{x})-u_j\right]\delta[V-v]d^{3N}x\,dV \qquad (3.12)$$

3.4 THERMODYNAMIC VARIABLES

The partition function is therefore given by:

$$Z\left(\beta,P,\underline{\Lambda}\right) = \sum_{\underline{\xi},\underline{U},V}\Omega\left(\underline{\xi},\underline{U},V\right)e^{-\beta(\underline{\Lambda}\cdot\underline{U}+PV)} \qquad (3.13)$$

Here, we have written the integration of (3.6) as a summation to emphasize the connection with its numerical application; note that most of the variables, in principal, are continuous.

The corresponding expression for the unnormalized probability density is:

$$\rho\left(\underline{\xi},\underline{U},V\mid\beta,P,\underline{\Lambda}\right) = \Omega\left(\underline{\xi},\underline{U},V\right)e^{-\beta(\underline{\Lambda}\cdot\underline{U}+PV)} \qquad (3.14)$$

We are using the following notation: Equation (3.14) describes the unnormalized probability density in the neighborhood of $\underline{\xi},\underline{U},V$, given the macroscopic state specified by $\beta,P,\underline{\Lambda}$.

The central point remains: The unnormalized probability density is the product of a system-independent Boltzmann factor and the system-dependent density of states characterized by $\underline{\xi},\underline{U},V$; the density of states is independent of $\beta,P,\underline{\Lambda}$.

A direct consequence of equations (3.10) and (3.11) is that there is a microscopic free energy associated with the unnormalized probability density:

$$g\left(\underline{\xi},\underline{U},V\mid\beta,P,\underline{\Lambda}\right) = -\beta^{-1}\ln\rho\left(\underline{\xi},\underline{U},V\mid\beta,P,\underline{\Lambda}\right) \qquad (3.15)$$

Here, g is the Gibbs free energy of the microstate characterized by $\underline{\xi},\underline{U},V\mid\beta,P,\underline{\Lambda}$. This can also be written as:

$$g\left(\underline{\xi},\underline{U},V\mid\beta,P,\underline{\Lambda}\right) = \underline{\Lambda}\cdot\underline{U}+PV-\beta^{-1}\ln\Omega\left(\underline{\xi},\underline{U},V\right) \qquad (3.16)$$

Thus, the free energy of the microstate is the sum of the total potential energy (including the pressure-volume term) and the term involving the density of states. Note that all the system-specific information is contained in the density of states; *i.e.* it expresses the dependence on the structural/molecular details of the system.

In the density of states formalism, bulk thermodynamic parameters are obtained by simply integrating over the configurational (blue) variables. Hence, the macroscopic Gibbs free energy is given by:

$$G(\beta, P, \underline{\Lambda}) = -\beta^{-1} \ln \left[\sum_{\xi, \underline{U}, V} \rho(\xi, \underline{U}, V \mid \beta, P, \underline{\Lambda}) \right]$$

(3.17)

This formulation of the partition function and Gibbs free energy readily lends themselves to differentiation with respect to the independent (red) state variables, including the temperature; this facilitates the application of well-known principles. Thus, an expression for the enthalpy can be easily obtained:

$$
\begin{aligned}
H(\beta, P, \underline{\Lambda}) &= \left(\frac{\partial (G/T)}{\partial (1/T)} \right)_{\underline{\Lambda}, P} = \left(\frac{\partial (\beta G)}{\partial \beta} \right)_{\underline{\Lambda}, P} \\
&= \frac{\sum_{\xi, \underline{U}, V} (\underline{\Lambda} \cdot \underline{U} + PV) \rho(\xi, \underline{U}, V \mid \beta, P, \underline{\Lambda})}{\sum_{\xi, \underline{U}, V} \rho(\xi, \underline{U}, V \mid \beta, P, \underline{\Lambda})} \\
&= \left\langle \underline{\Lambda} \cdot \underline{U} + PV \right\rangle_{\beta, P, \underline{\Lambda}}
\end{aligned}
$$

(3.18)

The angle brackets in the last equation denote the ensemble average and reflect the well-known result that the enthalpy is the ensemble average of the total potential energy. Ensemble averages are very straightforward in the density of states formulation; here we show the ensemble average of an arbitrary quantity, ϑ, and the corresponding thermodynamic value, Θ:

$$\Theta(\beta, P, \underline{\Lambda}) = \left\langle \vartheta \right\rangle_{\beta, P, \underline{\Lambda}} = \frac{\sum_{\xi, \underline{U}, V} \vartheta(\xi, \underline{U}, V) \rho(\xi, \underline{U}, V \mid \beta, P, \underline{\Lambda})}{\sum_{\xi, \underline{U}, V} \rho(\xi, \underline{U}, V \mid \beta, P, \underline{\Lambda})}$$

(3.19)

The entropy is given by:

$$S(\beta, P, \underline{\Delta}) = \left(\frac{1}{T}\right)(H - G) = -\left(\frac{\partial G}{\partial T}\right)_P$$

$$= -\left(\frac{1}{T}\right)G + \left(\frac{1}{T}\right)\frac{\sum_{\underline{\xi}, \underline{U}, V}(\underline{\Delta} \cdot \underline{U} + PV)\rho(\underline{\xi}, \underline{U}, V \mid \beta, P, \underline{\Delta})}{\sum_{\underline{\xi}, \underline{U}, V}\rho(\underline{\xi}, \underline{U}, V \mid \beta, P, \underline{\Delta})}$$

(3.20)

Likewise, the heat capacity is given by:

$$C_P = \left(\frac{\partial H}{\partial T}\right)_{\underline{\Delta}, P} = \frac{\partial}{\partial T}\left\langle \underline{\Delta} \cdot \underline{U} + PV \right\rangle_{\beta, P, \underline{\Delta}}$$

$$= \left(\frac{1}{k_B T^2}\right)\left\langle \left(\underline{\Delta} \cdot \underline{U} + PV\right)^2 \right\rangle_{\beta, P, \underline{\Delta}} - \left(\frac{1}{k_B T^2}\right)\left\langle \underline{\Delta} \cdot \underline{U} + PV \right\rangle^2_{\beta, P, \underline{\Delta}}$$

$$= \left(\frac{1}{k_B T^2}\right)\left\langle \left(\underline{\Delta} \cdot \underline{U} + PV - \left\langle \underline{\Delta} \cdot \underline{U} + PV \right\rangle\right)^2 \right\rangle_{\beta, P, \underline{\Delta}}$$

(3.21)

This is the expression, in the density of states formalism, of the well-known result that the heat capacity is the ensemble-average of the fluctuations of the enthalpy.

3.5   POTENTIALS AND OTHER VARIABLES OF MEAN FORCE

One of the main reasons for introducing the generalized coordinate vector, $\underline{\xi}$, is that it is possible to calculate the Gibbs free energy as a function of $\underline{\xi}$. This is a well-known quantity called the potential of mean force; it can be readily expressed in the density of states formalism by:

$$g_{meanforce}\left(\underline{\xi} \mid \beta, P, \underline{\Delta}\right) = -\beta^{-1}\ln\left[\sum_{\underline{U}, V}\rho\left(\underline{\xi}, \underline{U}, V \mid \beta, P, \underline{\Delta}\right)\right]$$

(3.22)

All the relevant thermodynamic variables can be explored as a function of $\underline{\xi}$ by similar means:

$$h_{meanforce}\left(\underline{\xi} \mid \beta, P, \underline{\Delta}\right) = \frac{\sum_{\underline{U}, V}(\underline{\Delta} \cdot \underline{U} + PV)\rho\left(\underline{\xi}, \underline{U}, V \mid \beta, P, \underline{\Delta}\right)}{\sum_{\underline{U}, V}\rho\left(\underline{\xi}, \underline{U}, V \mid \beta, P, \underline{\Delta}\right)}$$

(3.23)

and

$$s_{meanforce}\left(\underline{\xi} \mid \beta, P, \underline{\Lambda}\right) = \left(\frac{1}{T}\right)\left[h_{meanforce}\left(\underline{\xi} \mid \beta, P, \underline{\Lambda}\right) - g_{meanforce}\left(\underline{\xi} \mid \beta, P, \underline{\Lambda}\right)\right] \tag{3.24}$$

and

$$c_{P,meanforce}\left(\underline{\xi} \mid \beta, P, \underline{\Lambda}\right) = \left(\frac{1}{k_B T^2}\right)\frac{\sum\limits_{\underline{U},V}\left(\underline{\Lambda}\bullet\underline{U} + PV - h_{meanforce}\right)^2 \rho\left(\underline{\xi},\underline{U},V \mid \beta, P, \underline{\Lambda}\right)}{\sum\limits_{\underline{U},V}\rho\left(\underline{\xi},\underline{U},V \mid \beta, P, \underline{\Lambda}\right)} \tag{3.25}$$

The additional variables of mean force are expected to be of interest in many structure-function correlations, including those alluded to here. For example in the analysis of energy barriers, it is expected that different barriers will present fundamentally different thermodynamics, even for those where the heights have similar potentials of mean force. Some may be dominated by the enthalpy of mean force while others could be entropically limited.

3.6   REPRESENTATIVE PROBLEMS

3.6.1   General Usefulness for Sampling Improvement

Atomic Detail Molecular Dynamics has a drawback of typically sampling a narrow region of the energy landscape, because it tends to stay at the bottom of whatever energy well it started in. So the simulations tend to have a lot of sampling in the area the simulation started and very little sampling everywhere else it is needed. This means that the trajectory only has information for a limited yield of thermodynamic calculations. Several techniques have been developed to increase the energy landscape sampling of atomic detail MD, such as hybrid MD-MC, replica swapping and adaptive integration. Replica swapping works by performing an ensemble of simulations with different initial or simulation conditions, and pairs of simulations periodically swap momenta or simulation conditions[73,74,75]. Adaptive integration alters the potential such that it becomes flat over the reaction coordinate so that barriers are overcome[76].
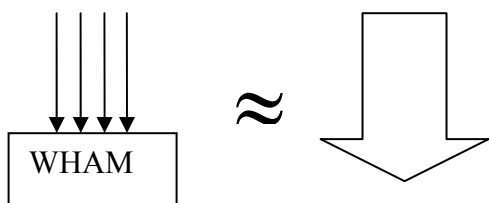
However another way to broaden sampling is to use high temperatures and WHAM. This has a big advantage over the other sample broadening methods mentioned because no additional programming is required. All that is required is to raise the thermostat (all simulation algorithms have this ability) and the use available WHAM algorithms. However using replica swapping or adaptive sampling tends to be model specific, so there is usually a lot of code writing required to get these methods to work with a particular system. With WHAM, the information from an ensemble of simulations, each with different initial conditions and simulation conditions (like temperature), can be brought together to yield a continuum of thermodynamic results for a range as broad as that represented by the ensemble. It can also be used in addition to replica swapping or adaptive sampling.

3.6.2   Accelerating Transition Rates with Thermodynamic Cycles

In our work with proton dynamics we have found that at 300K and for simulations that are short enough to be tractable, solvation shells inhibit ionization state transitions at rates that are statistically sufficient. This serves as an illustrative problem for a genre of problems involving barriers that can be crossed more easily at higher temperatures. Consider the thermodynamic cycle in Figure 22 below showing a protonated amino-acid as SH, and the deprotonated amino-acid as S^-. Suppose that we are really interested in the protonation free energy changes (ΔG) at 300K,

$$ SH \overset{\Delta G}{\rightarrow} S^- + H^+ \quad @300K. $$
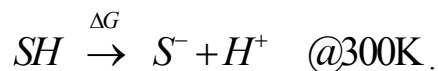
This calculation cannot be done directly because 300K simulations would have to be unfeasibly long to generate a statistically sufficient number of transitions. However it can be efficiently done by taking the system to higher temperatures where transition rates are significantly higher, building a thermodynamic cycle as shown below and using WHAM to combine the information from all simulations to yield relative free energies. This allows us to calculate ΔG by using

118

$\Delta G = \Delta G_1 + \Delta G_2 + ... \Delta G_7$.   This is a standard thermodynamic technique, transporting information from one part of the cycle to another by setting up the appropriate thermodynamic cycle. The example below shows how we used WHAM for thermodynamic calculations by using high temperature and a thermodynamic cycle to accelerate the crossing of an ionization barrier. As already suggested, the approach can be generalized for any situation where there is any type of barrier that can be crossed with high temperature.

$$\Delta G_4$$
$$\rightarrow$$

High Temperature

$$SH \rightleftharpoons \ S^- + H^+ \quad \text{high temperature}$$

$$\Delta G_3 \uparrow \quad \updownarrow \qquad \updownarrow \qquad \downarrow \Delta G_5$$

$$SH \qquad S^- + H^+$$

$$\Delta G_2 \uparrow \quad \updownarrow \qquad \updownarrow \qquad \downarrow \Delta G_6$$

$$SH \qquad S^- + H^+$$

$$\Delta G_1 \uparrow \quad \updownarrow \qquad \updownarrow \qquad \downarrow \Delta G_7$$

$$SH \qquad S^- + H^+ \quad 300K$$

$$\rightarrow$$

$$\Delta G$$
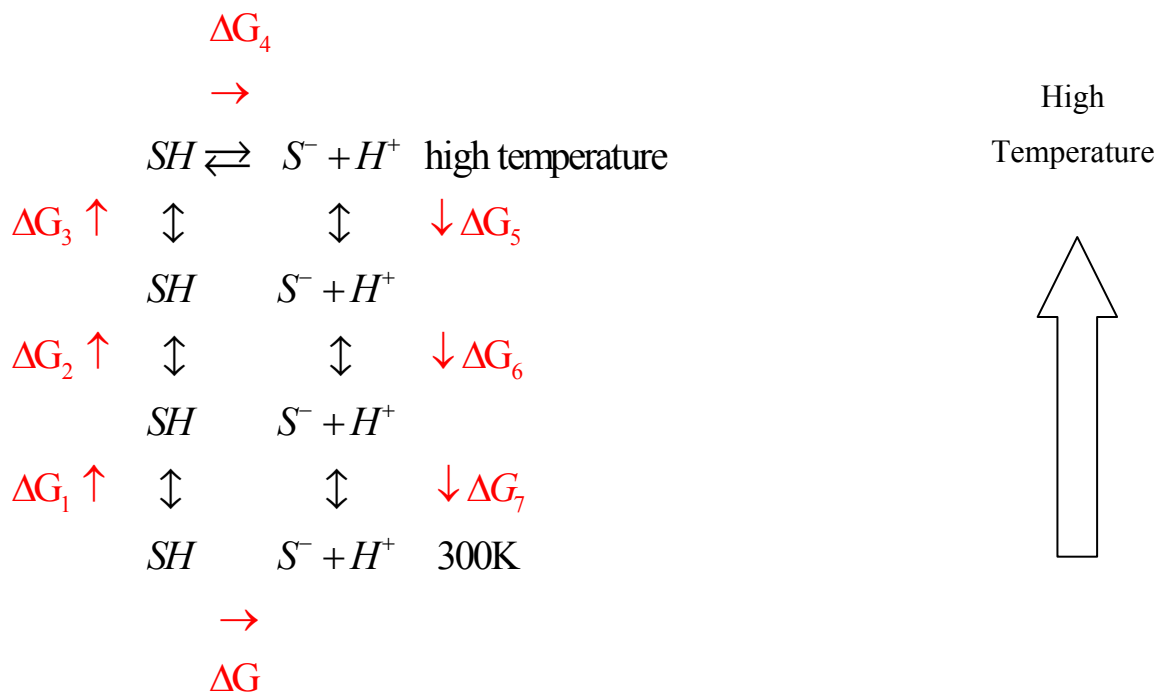
Figure 22: Thermodynamic Cycle and WHAM

3.6.3   Umbrella Sampling Type Calculations

In section 3.5 we looked at calculating Potentials of Mean Force (PMF), which is the Gibbs Free Energy as a function of some generalized coordinate vector $\underline{\xi}$. However the system may not naturally sample the region of $\underline{\xi}$ that we are interested in. In this case we can include a $\lambda_j u_{j'}$ term to the $\sum_i \lambda_i u_i$ sum for the purpose of biasing the system so that it does sample the $\underline{\xi}$ region of interest. These types of calculation are known as Umbrella Sampling calculations. As a

matter of fact, the first published application of WHAM for a biomolecule involved the generation of the PMF profile of the pseudorotation angle of the sugar ring in deoxyadenosine[63,64].

3.6.4 Ligand Binding Thermodynamic Cycles

One of the main advantages of the density of states formalism is that it allows one to set up biophysical problems in terms of rigorous statistical mechanics. This facilitates understanding the role of the specific approximations and numerical approaches used to calculate the numerical results. Here we discuss a few specific examples to illustrate this.

The application of the density of states formalism to the calculation of free energy and other variables is illustrated by the problem of base-analog substitutions in DNA binding *e.g.* with *Eco* RI endonuclease. Binding calculations for alternative ligands in drug discovery are formally identical as are analyses of site-directed mutations in proteins. One sets up the following thermodynamic cycle that transforms the DNA from the native to the analog-containing form. This must be done both in the complex and free DNA. Formally, this is done by setting up the Hamiltonian with a coupling constant, λ_1, such that one value ($\lambda_1=0$) corresponds to the native state while another ($\lambda_1=1$) generates the Hamiltonian for the analog-containing forms.

$$
\begin{array}{ccc}
 & \Delta G_1 & \\
P + D & \longrightarrow & PD \qquad \lambda_1 = 0 \\
\Delta G_A \downarrow & & \downarrow \Delta G_B \\
 & \Delta G_2 & \\
P + D^* & \longrightarrow & PD^* \qquad \lambda_1 = 1
\end{array}
$$

$$\Delta\Delta G \ = \ \Delta G_2 - \Delta G_1 \ = \ \Delta G_B - \Delta G_A$$
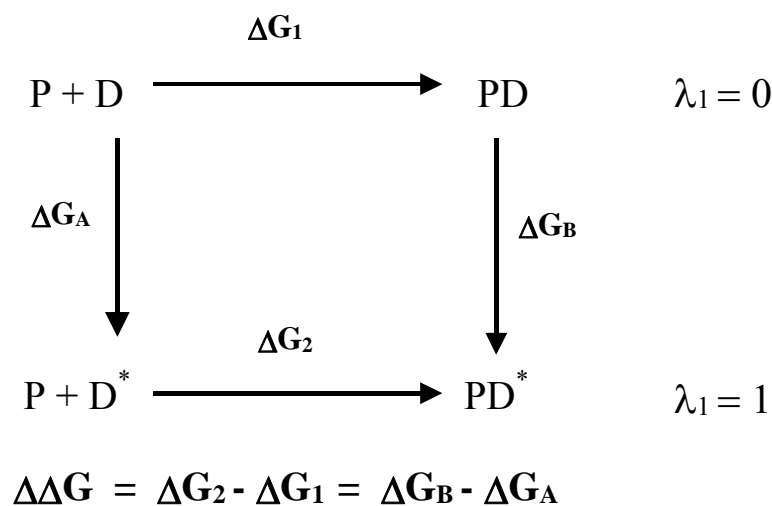
Figure 23: A representative thermodynamic cycle.

Here, ΔG_1 and ΔG_2 are the binding free energies for the native and analog-containing forms, respectively. The functionally interesting change is the differential free energy, $\Delta\Delta G$. In principle, this could be calculated by subtracting ΔG_1 from ΔG_2. For practical reasons, however, the calculation of these quantities is numerically intractable while the calculation of ΔG_A and ΔG_B is not. The thermodynamic cycle guarantees that their difference yields the desired value. The calculation of these values can be obtained from:

$$\Delta G = G\left(\beta, P, \underline{\Lambda}_1\right) - G\left(\beta, P, \underline{\Lambda}_0\right)$$
(3.26)

The preceding expression is used for either ΔG_A or ΔG_B, and is based on the Hamiltonians, as described. Clearly, this analysis can be extended to any other thermodynamic parameter, for example:

$$\Delta H = H\left(\beta, P, \underline{\Lambda}_1\right) - H\left(\beta, P, \underline{\Lambda}_0\right)$$
(3.27)

3.7 SINGLE HISTOGRAM METHODS

In this section we will describe how to obtain probability densities from a single simulation using single histogram equations. The next section will detail obtaining probability densities from multiple simulations. Because our main interest is multiple histograms, this section is brief since it only serves as a bridge of understanding the next section.

To this point in the discussion, the only assumption we have made is that it is possible to write probability as a product of the density of states and the Boltzmann factor.

$$\rho\left(\underline{\xi}, \underline{U}, V \mid \beta, P, \underline{\Lambda}\right) = \Omega\left(\underline{\xi}, \underline{U}, V\right) e^{-\beta\left(\underline{\Lambda} \cdot \underline{U} + PV\right)}$$
(3.28)

Consider a single simulation generated under $\beta, P, \underline{\Lambda}$ conditions. Let n be the total number of snapshots in the simulation. Let $N\left(\underline{\xi}, \underline{U}, V\right)$ be the number of snapshots, of this single simulation, that fall into the $\underline{\xi}, \underline{U}, V$ bin. Then the normalized probability $p\left(\underline{\xi}, \underline{U}, V \mid \beta, P, \underline{\Lambda}\right)$ can be estimated according to the following straightforward calculation.

$$p\left(\underline{\xi}, \underline{U}, V \mid \beta, P, \underline{\Lambda}\right) = \frac{N(\underline{\xi}, \underline{U}, V)}{n}$$
(3.29)

121

Note that we have introduced a new color, green. This color represents a new and significant phase of WHAM. Previously we stayed in the realm of theory. Now we enter the realm of numerical computation. All probability expressions in this chapter up to the previous section were completely rigorous. Now we make assumptions and estimates, and introduce statistical errors. Using this probability estimate and equation (3.9), we can write

$$p\left(\underline{\xi},\underline{U},V \mid \beta,P,\underline{\Delta}\right) = \frac{1}{Z} \bullet \Omega\left(\underline{\xi},\underline{U},V\right) e^{-\beta(\Delta\bullet\underline{U}+PV)} \longrightarrow \frac{N(\underline{\xi},\underline{U},V)}{n} = \frac{1}{Z} \bullet \Omega\left(\underline{\xi},\underline{U},V\right) e^{-\beta(\Delta\bullet\underline{U}+PV)} \quad (3.30)$$

So

$$\Omega\left(\underline{\xi},\underline{U},V\right)_{est} = Z \bullet \frac{N\left(\underline{\xi},\underline{U},V\right)}{n\,e^{-\beta(\Delta\bullet\underline{U}+PV)}} \quad (3.31)$$

Since the relationship between Z and the free energy of the simulation that generated the histogram (g) is $Z = e^{-g}$, we get

$$\Omega\left(\underline{\xi},\underline{U},V\right)_{est} = e^{-g} \bullet \frac{N\left(\underline{\xi},\underline{U},V\right)}{n\,e^{-\beta(\Delta\bullet\underline{U}+PV)}} \quad (3.32)$$

3.8   MULTIPLE HISTOGRAM METHODS

We will now pick up where we left off, with the density of states expression for the single histogram (3.32). Now consider an ensemble of R simulations, which generates R histograms. Now consider the nth simulation. Using equation (3.32), the density of states, determined from the information from the nth simulation only is

$$\Omega_n\left(\underline{\xi},\underline{U},V\right)_{est} = \frac{N_n\left(\underline{\xi},\underline{U},V\right)e^{-g_n}}{n_n\,e^{-\beta_n(\Delta_n\bullet\underline{U}+P_nV)}} \quad (3.33)$$

We can improve our estimate of the density of states if we consider information from all R simulations according to

$$\Omega\left(\underline{\xi},\underline{U},V\right)_{est} = \sum_{n=1}^{R} p_n \Omega_n\left(\underline{\xi},\underline{U},V\right). \quad (3.34)$$

That is, we sum over the Ω_n's and weight each term by some p_n weighting factor. The $\{p_n\}$ are chosen so that $\sum_{n=1}^{R} p_n = 1$ and that the error in the density of states $(\delta\Omega)^2$ is minimized with respect to p_n. These two conditions yield[77]

$$p_n(\underline{\xi},\underline{U},V) = \frac{n_n e^{g_n - \beta_n(\Delta_n \cdot \underline{U} + P_n V)}}{\sum_{m=1}^{R} n_m e^{g_m - \beta_m(\Delta_n \cdot \underline{U} + P_m V)}} \qquad (3.35)$$

Substituting (3.35) and (3.33) into (3.34) gives

$$\Omega\left(\underline{\xi},\underline{U},V\right)_{est} = \frac{\sum_{k=1}^{R} N_k\left(\underline{\xi},\underline{U},V\right)}{\sum_{m=1}^{R} n_m e^{g_m - \beta_m(\Delta_m \cdot \underline{U} + P_m V)}} \qquad (3.36)$$

So

$$\rho_{est}\left(\underline{\xi},\underline{U},V \mid \beta,P,\underline{\Delta}\right) = \Omega_{est} e^{-\beta(\Delta \cdot \underline{U} + PV)} = \frac{\sum_{k=1}^{R} N_k\left(\underline{\xi},\underline{U},V\right) e^{-\beta(\Delta \cdot \underline{U} + PV)}}{\sum_{m=1}^{R} n_m e^{g_m - \beta_m(\Delta_m \cdot \underline{U} + P_m V)}} \qquad (3.37)$$

As before, bulk parameters are obtained by simply integrating over the configurational (blue) variables, giving us a macroscopic Gibbs free energy.

$$e^{(-g_m)} = \sum_{\underline{\xi},\underline{U},V} \rho_{est}\left(\underline{\xi},\underline{U},V \mid \beta_m, P_m, \underline{\Delta}_m\right) \qquad (3.38)$$

$$G_{est}\left(\beta,P,\underline{\Delta}\right) = -\beta^{-1} \ln\left[\sum_{\underline{\xi},\underline{U},V} \rho_{est}\left(\underline{\xi},\underline{U},V \mid \beta,P,\underline{\Delta}\right)\right] \qquad (3.39)$$

For an ensemble average of some arbitrary quantity Θ,

$$\langle\Theta\rangle_{est\mid\beta,P,\underline{\Delta}} = \frac{\sum_{\underline{\xi},\underline{U},V} \Theta\left(\underline{\xi},\underline{U},V\right) \rho_{est}\left(\underline{\xi},\underline{U},V \mid \beta,P,\underline{\Delta}\right)}{\sum_{\underline{\xi},\underline{U},V} \rho_{est}\left(\underline{\xi},\underline{U},V \mid \beta,P,\underline{\Delta}\right)} \qquad (3.40)$$

Direct Summation:

The final stage of the computational formalization involves direct summation. $\underline{\xi},\underline{U},$ and V are continuous quantities and $(\underline{\xi},\underline{U},V)$ represents a bin of size $\underline{\xi}+\partial\underline{\xi},\underline{U}+\partial\underline{U},$ and $V+\partial V$. It is far more computationally efficient to do "direct summation", where each snapshot is itself its own bin.

$$\underline{\xi},\underline{U},V \rightarrow \underline{\xi}_{k,t},\underline{U}_{k,t},V_{k,t} \tag{3.41}$$

With direct summation, since each snapshot is its own bin, then $\left[\sum_{k=1}^{R} N_k\left(\underline{\xi},\underline{U},V\right)\right] \rightarrow 1$ in equation (3.37). It also means that equation (3.41) causes the $\sum_{\underline{\xi},\underline{U},V}$ sum in equation (3.38) to become sums over k and t, that is $\sum_{\underline{\xi},\underline{U},V} \rightarrow \sum_{k}\sum_{t}$. Substituting $\left[\sum_{k=1}^{R} N_k\left(\underline{\xi},\underline{U},V\right)\right]=1$ and equation (3.37) into equation (3.38), direct summation gives us the following expression for g_m, the relative free energy of the mth simulation.

$$e^{-g_m} = \sum_{k=1}^{R}\sum_{t=1}^{n_k} \frac{e^{-\beta_m\left(\Lambda_m \cdot \underline{U}_{k,t}+P_m V_{k,t}\right)}}{\sum_{r=1}^{R} n_r e^{\left[g_r-\beta_r\left(\Lambda_r \cdot \underline{U}_{k,t}+P_r V_{k,t}\right)\right]}} = \sum_{k=1}^{R}\sum_{t=1}^{n_k} \frac{e^{-\beta_m\left(\Lambda_m \cdot \underline{U}_{k,t}+P_m V_{k,t}\right)}}{z_{k,t}} \tag{3.42}$$

$$z_{k,t} = \sum_{r=1}^{R} n_r e^{\left[g_r-\beta_r\left(\Lambda_r \cdot \underline{U}_{k,t}+P_r V_{k,t}\right)\right]} \tag{3.43}$$

Notice that in the expression for e^{-g_m} above, we introduce a factor $z_{k,t}$ in the denominator terms. Two lines above, we already described g_m as the relative free energy of the mth simulation. $z_{k,t}$ can be described as the relative weighting factor of the tth snapshot of the kth simulation. In practice, for a system for which we have an ensemble of simulations, the two equations (for e^{-g_m} and $z_{k,t}$) are iterated to convergence. The starting point could be arbitrary or anything convenient such as setting all g_m's =1 (or $\{g_m\}=0^{63}$, it does not matter, the converged results do not depend on the starting points for $\{g_m\}$).

For a given set of simulations the relative free energy of the mth simulation g_m is completely determined by the set $\{z_{k,t}\}$. However g_m is a function of β_m, P_m and Λ_m, i.e.

$g_m = g_m(\beta_m, P_m, \Lambda_m)$. This means that the free energy value for the mth simulation relative to the other simulations is a function of the depends only on the state variables and not the set of $\{z_{k,t}\}$ chosen. In other words g_m is invariant with the simulations chosen to comprise the simulation set, provided the simulations are equilibrated, long enough and that there is sufficient counting-statistics.

When converged, the set of g_m's and $z_{k,t}$'s is a very convenient expression of the density of states. The following derivations for useful thermodynamic parameters will demonstrate how convenient this description is.

$$G_{est}(\beta, P, \underline{\Lambda}) = -\beta^{-1} \ln\left\{\sum_{k=1}^{R}\sum_{t=1}^{n_k} \frac{e^{-\beta(\underline{\Lambda} \cdot \underline{U}_{k,t} + PV_{k,t})}}{z_{k,t}}\right\}$$

(3.44)

is the calculated relative free energy of a trajectory (histogram) generated under $\beta, P, \underline{\Lambda}$ conditions. The ensemble average of an arbitrary quantity, ϑ, and the corresponding thermodynamic value, Θ now becomes:

$$\Theta_{est}(\beta, P, \underline{\Lambda}) = \langle\vartheta\rangle_{est|\beta,P,\underline{\Lambda}} = e^{\beta G_{est}(\beta,P,\underline{\Lambda})} \sum_{k=1}^{R}\sum_{t=1}^{n_k} \frac{\vartheta_{k,t} e^{-\beta(\underline{\Lambda} \cdot \underline{U}_{k,t} + PV_{k,t})}}{z_{k,t}}$$

(3.45)

Enthalpy, Entropy and Heat Capacity follow easily.

$$H_{est}(\beta, P, \underline{\Lambda}) = \langle\underline{\Lambda} \cdot \underline{U} + PV\rangle_{est|\beta,P,\underline{\Lambda}} = \underline{\Lambda} \cdot \langle\underline{U}\rangle_{est|\beta,P,\underline{\Lambda}} + P\langle V\rangle_{est|\beta,P,\underline{\Lambda}}$$

(3.46)

$$S_{est} = \left(\frac{1}{T}\right)(H_{est} - G_{est})$$

(3.47)

$$C_{P,est}(\beta, P, \underline{\Lambda}) = \left(\frac{1}{k_B T^2}\right)\langle(\underline{\Lambda} \cdot \underline{U} + PV)^2\rangle_{est|\beta,P,\underline{\Lambda}} - \left(\frac{1}{k_B T^2}\right)\langle\underline{\Lambda} \cdot \underline{U} + PV\rangle^2_{est|\beta,P,\underline{\Lambda}}$$

(3.48)

Potentials, Enthalpies, Entropies etc. of mean force:

We pointed out earlier the usefulness of potentials and other variables of mean force in biological systems. What follows are these variables recast with using our direct summation and density of states description.

$$\delta^{\underline{\xi}'}_{\underline{\xi}_{k,t}} = \begin{cases} 1, & \underline{\xi}_{k,t} \in N\left(\underline{\xi}'\right) \\ 0, & \underline{\xi}_{k,t} \notin N\left(\underline{\xi}'\right) \end{cases} \tag{3.49}$$

$\delta^{\underline{\xi}'}_{\underline{\xi}_{k,t}} = 1$ if the reaction co-ordinate for the k,t snapshot ($\underline{\xi}_{k,t}$) meets the given criteria ($\in N(\underline{\xi}')$).

$$g_{est}\left(\underline{\xi}' \mid \beta, P, \underline{\Lambda}\right) = -\beta^{-1} \ln\left\{ \sum_{k=1}^{R} \sum_{t=1}^{n_k} \frac{\delta^{\underline{\xi}'}_{\underline{\xi}_{k,t}} e^{-\beta\left(\underline{\Lambda}\bullet\underline{U}_{k,t} + PV_{k,t}\right)}}{z_{k,t}} \right\} \tag{3.50}$$

$$\vartheta_{est}\left(\underline{\xi}' \mid \beta, P, \underline{\Lambda}\right) = e^{\beta g_{est}\left(\underline{\xi}' \mid \beta, P, \Delta\right)} \sum_{k=1}^{R} \sum_{t=1}^{n_k} \frac{\Theta_{k,t} \delta^{\underline{\xi}'}_{\underline{\xi}_{k,t}} e^{-\beta\left(\underline{\Lambda}\bullet\underline{U}_{k,t} + PV_{k,t}\right)}}{z_{k,t}} \tag{3.51}$$

$$h_{est}\left(\underline{\xi}' \mid \beta, P, \underline{\Lambda}\right) = e^{\beta g_{est}\left(\underline{\xi}' \mid \beta, P, \Delta\right)} \sum_{k=1}^{R} \sum_{t=1}^{n_k} \frac{\left(\underline{\Lambda}\bullet\underline{U}_{k,t} + PV_{k,t}\right) \delta^{\underline{\xi}'}_{\underline{\xi}_{k,t}} e^{-\beta\left(\underline{\Lambda}\bullet\underline{U}_{k,t} + PV_{k,t}\right)}}{z_{k,t}} \tag{3.52}$$

$$s_{est}\left(\underline{\xi}' \mid \beta, P, \underline{\Lambda}\right) = \left(\frac{1}{T}\right)\left(h_{est}\left(\underline{\xi}' \mid \beta, P, \underline{\Lambda}\right) - g_{est}\left(\underline{\xi}' \mid \beta, P, \underline{\Lambda}\right)\right) \tag{3.53}$$

3.9   MODIFICATIONS FOR CONSTANT pH CALCULATIONS

Recall that in section 2.1 we affect constant pH simulations by coupling our system to a proton bath with proton chemical potential μ_H. The number of protons in the proton bath is L, so μ and L are conjugate variables. In this section, we will briefly repeat the main topics covered in the previous sections, paying attention to the modifications resulting from a $NPT\mu_H$ (constant pH) ensemble. We start with a review of the previous theory, but with appropriate modifications for constant pH simulations. However, we will put more emphasis on the beginning (the fundamentals), and the ending (the direct summation) descriptions.

Our Hamiltonian becomes

$$\mathbf{H} = \sum_{i=1}^{3N}\left(\frac{p_i^2}{2m_i}\right) + U\left(\vec{x}, \underline{\eta}\right) \tag{3.54}$$

due to the fact that our potential energy is now a function of atom positions ($\vec{x}$) and protonation state $\underline{\eta}$ which describes the protonation state of all titratable sites of the system (see section 2.2). The partition function now becomes

$$Q_{NPT\mu_H} = \int e^{-\beta\left[U(\vec{x},\underline{\eta})+p^2/2m\right]}d^{3N}pd^{3N}xd\underline{\eta}$$
$$= \int \Omega(U,V,L)e^{-\beta\left(U+PV+p^2/2m+L\mu_H\right)}d^{3N}pdUdVdL \qquad (3.55)$$
$$= Z\int e^{\beta p^2/2m}d^{3N}p$$

In the first expression for Q above, the integral is over all momenta, positions, and also all protonation states ($\delta\underline{\eta}$). The second expression introduces the density of states $\Omega(U,V,L)$, and in doing so the integral is converted to one over all momenta, energies, volumes and proton counts ($d^{3N}pdUdVdL$). The third expression isolates the integral over all momenta by introducing the spatial partition function

$$Z_{NTP\mu_H} = \int \Omega(U,V,L)e^{-\beta(U+PV+L\mu)}dUdVdL \qquad (3.56)$$

where Ω is the density of states, which contains all of the system-specific information. In principal the density of states for the system with energy u, volume v, and number of protons in the proton bath l, can be calculated from:

$$\Omega(u,v,l) = \int \delta\left(U(\vec{x},\underline{\mu})-u\right)\delta(V-v)\delta(L-l)d^{3N}xdVdL$$

The relation between the un-normalized probability density $\rho(u,v,l)$, and the density of states $\Omega(u,v,l)$ is:

$$\rho(u,v,l) = \Omega(u,v,l)e^{-\beta(u+Pv+l\mu)}$$

Treating the Hamiltonian according to equations $\xi_i = \Xi_i(\vec{x})$ and

$U = \sum \lambda_i U_i(\vec{x}) = \sum \lambda_i u_i = \underline{\Lambda}\bullet\underline{U}$ (3.11), we can write the density of states in a

manner analogous to equation

$$\Omega(\underline{\xi},\underline{u},v) = \int \Pi\delta\left[\Xi_i(\vec{x})-\xi_i\right]\Pi\delta\left[U_j(\vec{x})-u_j\right]\delta[V-v]d^{3N}xdV$$

:

127

$$\Omega\left(\underline{\xi},\underline{u},v\right)=\int\Pi\,\delta\left[\Xi_i(\vec{x})-\xi_i\right]\Pi\,\delta\left[U_j(\vec{x})-u_j\right]\delta[V-v]\delta[L-l]d^{3N}xdVdl$$

Writing the partition function as a summation instead of the integral form of equation

$$Z_{NTP\mu_H}=\int\Omega(U,V,L)e^{-\beta(U+PV+L\mu)}dUdVdL\qquad\text{gives us}$$

$$Z\left(\beta,P,\underline{\Lambda},\mu\right)=\sum_{\underline{\xi},\underline{U},V,L}\Omega\left(\underline{\xi},\underline{U},V,L\right)e^{-\beta(\underline{\Lambda}\cdot\underline{U}+PV+L\mu)}$$

and the corresponding expression for the un-normalized probability density is:

$$\rho\left(\underline{\xi},\underline{U},V,L\mid\beta,P,\underline{\Lambda},\mu\right)=\Omega\left(\underline{\xi},\underline{U},V,L\right)e^{-\beta(\underline{\Lambda}\cdot\underline{U}+PV+L\mu)}$$

By integrating probabilities over the configurational (blue) variables, we can get an expression fro the macroscopic Gibbs free energy:

$$G\left(\beta,P,\underline{\Lambda},\mu\right)=-\beta^{-1}\ln\left[\sum_{\underline{\xi},\underline{U},V,L}\rho\left(\underline{\xi},\underline{U},V,L\mid\beta,P,\underline{\Lambda},\mu\right)\right]$$

Calculated estimates for the density of states, analogous to equation (3.36)
now looks like:

$$\Omega\left(\underline{\xi},\underline{U},V,L\right)_{est}=\frac{\displaystyle\sum_{k=1}^{R}N_k\left(\underline{\xi},\underline{U},V,L\right)}{\displaystyle\sum_{m=1}^{R}n_m\,e^{g_m-\beta_m\left(\underline{\Lambda}_m\cdot\underline{U}+P_mV+L\mu_m\right)}}$$

So now the estimates for the probability density and the Gibbs free energy looks like:

$$\rho_{est}\left(\underline{\xi},\underline{U},V,L\mid\beta,P,\underline{\Lambda},\mu\right)=\Omega_{est}e^{-\beta(\underline{\Lambda}\cdot\underline{U}+PV+L\mu)}=\frac{\displaystyle\sum_{k=1}^{R}N_k\left(\underline{\xi},\underline{U},V,L\right)e^{-\beta(\underline{\Lambda}\cdot\underline{U}+PV+L\mu)}}{\displaystyle\sum_{m=1}^{R}n_m\,e^{g_m-\beta_m\left(\underline{\Lambda}_m\cdot\underline{U}+P_mV+L\mu_m\right)}}$$

$$G_{est}\left(\beta,P,\underline{\Lambda},L\right)=-\beta^{-1}\ln\left[\sum_{\underline{\xi},\underline{U},V,L}\rho_{est}\left(\underline{\xi},\underline{U},V,L\mid\beta,P,\underline{\Lambda},\mu\right)\right]$$

Now we apply direct summation $\xi, \underline{U}, V \rightarrow \underline{\xi}_{k,t}, \underline{U}_{k,t}, V_{k,t}$ as described before in equation $\xi, \underline{U}, V \rightarrow \underline{\xi}_{k,t}, \underline{U}_{k,t}, V_{k,t}$ and we finally arrive at our most powerful expressions for the density of states:

$$e^{-g_m} = \sum_{k=1}^{R} \sum_{t=1}^{n_k} \frac{e^{-\beta_m \left(\Lambda_m \bullet \underline{U}_{k,t} + P_m V_{k,t} + L_{k,t} \mu_m \right)}}{\sum_{r=1}^{R} n_r e^{\left[g_r - \beta_r \left(\Lambda_r \bullet \underline{U}_{k,t} + P_r V_{k,t} + L_{k,t} \mu_m \right) \right]}} = \sum_{k=1}^{R} \sum_{t=1}^{n_k} \frac{e^{-\beta_m \left(\Lambda_m \bullet \underline{U}_{k,t} + P_m V_{k,t} + L_{k,t} \mu_m \right)}}{z_{k,t}}$$
(3.57)

$$z_{k,t} = \sum_{r=1}^{R} n_r e^{\left[g_r - \beta_r \left(\Lambda_r \bullet \underline{U}_{k,t} + P_r V_{k,t} + L_{k,t} \mu_m \right) \right]}$$
(3.58)

Just as before, these above two equations containing g_m and $z_{k,t}$ are iterated to convergence, and the iteration can be started with all $g_m = 1$. Just as before we can calculate useful averages, such as those relating to the heat capacity calculation:

$$C_{P,est} \left(\beta, P, \underline{\Delta}, \mu \right) = \left(\frac{1}{k_B T^2} \right) \left\langle \left(\underline{\Delta} \bullet \underline{U} + PV + L\mu \right)^2 \right\rangle_{est|\beta,P,\underline{\Delta},\mu} - \left(\frac{1}{k_B T^2} \right) \left\langle \underline{\Delta} \bullet \underline{U} + PV + L\mu \right\rangle^2_{est|\beta,P,\underline{\Delta},\mu}$$

Applied to a free energy and other various calculations, we use the Dirac delta function

$$\delta_{\underline{\xi}_{k,t}}^{\xi'} = \begin{cases} 1, & \underline{\xi}_{k,t} \in N\left(\underline{\xi}' \right) \\ 0, & \underline{\xi}_{k,t} \notin N\left(\underline{\xi}' \right) \end{cases}$$

to discriminately do summations and get

$$g_{est} \left(\underline{\xi}' \mid \beta, P, \underline{\Delta}, \mu \right) = -\beta^{-1} \ln \left\{ \sum_{k=1}^{R} \sum_{t=1}^{n_k} \frac{\delta_{\underline{\xi}_{k,t}}^{\xi'} e^{-\beta \left(\underline{\Delta} \bullet \underline{U}_{k,t} + P V_{k,t} + L_{k,t} \mu \right)}}{z_{k,t}} \right\}$$

So far we have shown the power of this formulation to potentially calculate the full range of thermodynamic variables. We will see in the next section, that the form of this last equation is very useful for microstate free energies and consequently, *pKa* calculations.

3.10 OVERVIEW OF pKa RELATED CALCULATIONS

We will now demonstrate a *pKa* calculation. For simplicity sake, consider a system with one titratable site i, and this titratable site is Cysteine. Recall that for Cysteine, there is one

deprotonated microstate, $\eta_i = 0$, and there are three protonated microstates, $\eta_i = 1, 2, 3$ (see section 2.2).

All microstates belong to one of two mutually exclusive charge/ionization states, protonated and deprotonated, a_i and b_i, respectively.

Therefore
$$\eta_i \in a_i \text{ or } \eta_i \in b_i$$

The free energy of each micro-state is:

$$g_{est}\left(\eta_i = 0 \mid \beta, P, \underline{\Delta}, \mu\right) = -\beta^{-1} \ln\left\{ \sum_{k=1}^{R} \sum_{t=1}^{n_k} \frac{\delta_{\eta_{ik,t}}^{\eta_i = 0} e^{-\beta\left(\underline{\Delta} \cdot \underline{U}_{k,t} + PV_{k,t} + L_{k,t}\mu\right)}}{z_{k,t}} \right\} \quad (3.59)$$

$$g_{est}\left(\eta_i = 1 \mid \beta, P, \underline{\Delta}, \mu\right) = -\beta^{-1} \ln\left\{ \sum_{k=1}^{R} \sum_{t=1}^{n_k} \frac{\delta_{\eta_{ik,t}}^{\eta_i = 1} e^{-\beta\left(\underline{\Delta} \cdot \underline{U}_{k,t} + PV_{k,t} + L_{k,t}\mu\right)}}{z_{k,t}} \right\}$$

$$g_{est}\left(\eta_i = 2 \mid \beta, P, \underline{\Delta}, \mu\right) = -\beta^{-1} \ln\left\{ \sum_{k=1}^{R} \sum_{t=1}^{n_k} \frac{\delta_{\eta_{ik,t}}^{\eta_i = 2} e^{-\beta\left(\underline{\Delta} \cdot \underline{U}_{k,t} + PV_{k,t} + L_{k,t}\mu\right)}}{z_{k,t}} \right\}$$

$$g_{est}\left(\eta_i = 3 \mid \beta, P, \underline{\Delta}, \mu\right) = -\beta^{-1} \ln\left\{ \sum_{k=1}^{R} \sum_{t=1}^{n_k} \frac{\delta_{\eta_{ik,t}}^{\eta_i = 3} e^{-\beta\left(\underline{\Delta} \cdot \underline{U}_{k,t} + PV_{k,t} + L_{k,t}\mu\right)}}{z_{k,t}} \right\}$$

The free energy of deprotonation is

$$G(b_i \mid \beta, P, \underline{\Delta}, \mu) = g(\eta_i = 0 \mid \beta, P, \underline{\Delta}, \mu)$$

which is simply the free energy of the 0$^{\text{th}}$ microstate, $g(\eta_i = 0)$. The free energy of protonation is the sum of the protonated free energy microstates.

$$G(a_i \mid \beta, P, \underline{\Delta}, \mu) = \sum_{\eta_i \in a_i} g(\eta_i \mid \beta, P, \underline{\Delta}, \mu) \quad (3.60)$$

So the pKa is

$$pKa = \frac{-\log_{10} e}{kT}\left\{ G(a_i \mid \beta, P, \underline{\Delta}, \mu) - G(b_i \mid \beta, P, \underline{\Delta}, \mu) \right\} \quad (3.61)$$

4.0 COMPUTATIONAL METHODS FOR MD/MC AND WHAM

4.1   OVERVIEW

At this point, we have already built the theoretical foundation and the notation that allows us to give a concise overview of our MD/MD-WHAM methods for free energy calculations. This we will now do. The MD/MC algorithm is an extensive modification of the sander[78] program of the AMBER[78] suite. To avoid unnecessary abstraction, we will apply this summary of our methods to the solvated single site Cysteine system.

The Hamiltonian of our system of N atoms is described by $H = \sum_{i=1}^{3N} \frac{p_i}{2m_i} + U(\vec{x}, \underline{\eta})$. The potential energy is a function of both "configuration" $\vec{x}$ and protonation state $\underline{\eta}$. $\vec{x}$ represents the configuration of the whole system *except the titratable sites*. $\underline{\eta}$ primarily describes the protonation state of the system, but it also has a little configuration information, specifically the orientation of the titratable hydrogen. In our description, protonation states are distinguished by both charge and orientation of the proton.

Each titratable site is realistically modeled having several possible discrete microstates. So Cysteine model has a total of 4 protonation states: one deprotonated and 3 protonated. The protonation states distinguish themselves by the orientation of the 3-fold dihedral that connects the proton.

state 0(charge=-1ec)   state1(charge=0ec)   state2(charge=0ec)   state3(charge=0ec)
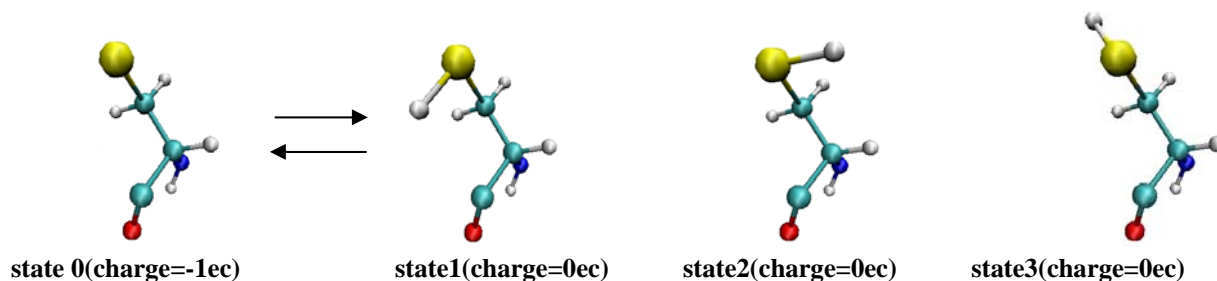
The absent proton in state 0 is modeled by a "ghost-proton" (no partial charge, no van der Waals parameters).

$\underline{\eta}$ describes the full set of η_i's (if the system has multiple titration sites), where $\eta_i = 0, 1, 2$ or 3 describes the protonation state of the ith titratable site

$l(\eta_i) = \{0, 1\}$ represents the number of protons in the proton bath for a particular value of η_i for site i. E.g. if the ith titratable site is Cysteine, and $\eta_i = 0$ (deprotonated), then $l(\eta_i) = 1$. If $\eta_i = 3$ (protonated), then $l(\eta_i) = 0$

$L(\underline{\eta}) = \sum_i l(\eta_i)$ is the total number of protons in the bath

One simulation cycle of our MD/MC code consists of an MD sub-cycle and an MC sub-cycle. The MD sub-cycle uses a fixed protonation state $\underline{\eta}$ and allows the configuration $\vec{x}$ to evolve, enforcing constant N, T, and P. In the MC sub-cycle, Monte Carlo sweeps act on the system to be updated), thus enforcing constant pH. Together one cycle simulates a "Biochemical Ensemble", constant N,P,T and pH.

Many of these trajectories are generated at different temperatures and pH.
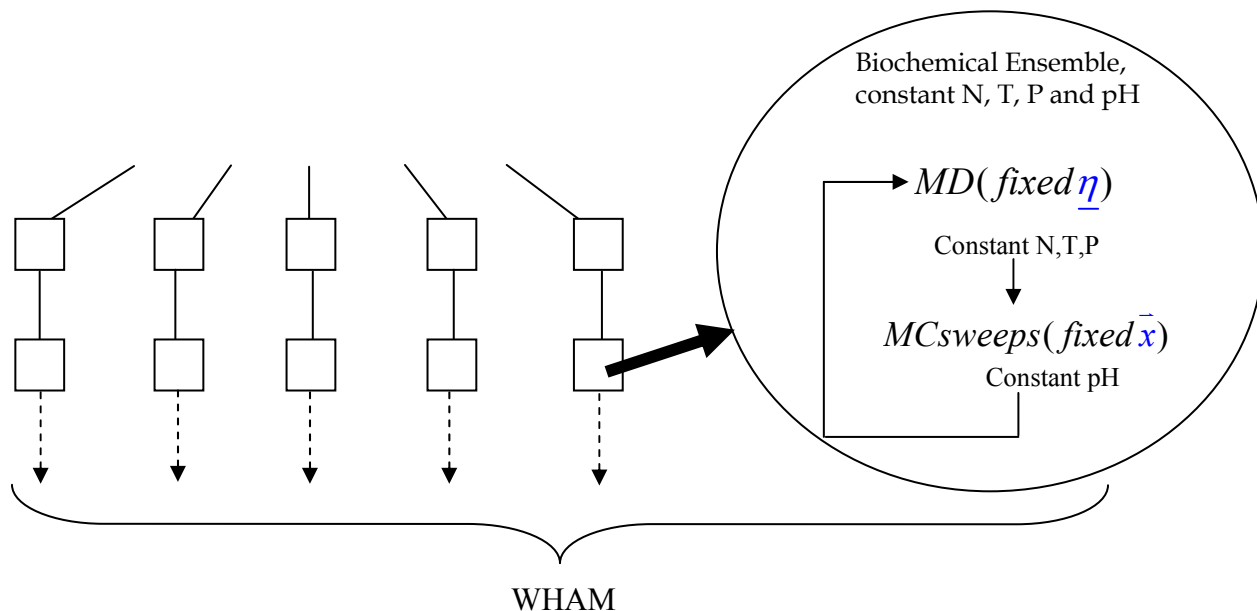
WHAM

All snapshots of all trajectories are then fed into our WHAM method for density-of-states, free energy, pKa and other thermodynamic calculations.

$e^{-\beta(\underline{\Lambda} \cdot \underline{U} + PV + L\mu_H)}$ is the Boltzmann factor associated with a snapshot of $\underline{U}, V, L$ potential energy, volume and bath protons respectively. β, Λ, P and μ_H represent the simulation conditions of constant temperature, optional restraint or perturbation parameter, constant pressure and constant pH $(\beta\mu_H = -\ln 10\, pH)$ respectively. If we know the density-of-states Ω, we can calculate any thermodynamic parameter. At the core of our WHAM algorithm is code that estimates the density-of-states in the following form.

$$e^{-g_m} = \sum_{k=1}^{R} \sum_{t=1}^{n_k} \frac{e^{-\beta_m(\Lambda_m \cdot U_{k,t} + P_m V_{k,t} + L_{k,t}\mu_m)}}{z_{k,t}}$$

$$z_{k,t} = \sum_{r=1}^{R} n_r e^{[g_r - \beta_r(\Lambda_r \cdot U_{k,t} + P_r V_{k,t} + L_{k,t}\mu_m)]}$$

g_m is the relative free energy of the mth simulation

$z_{k,t}$ is the relative weight of the tth snapshot of the kth simulation.

The g_m's and $z_{k,t}$'s are determined by iterating the 2 equations above until the g_m's converge. The set of g_m's and $z_{k,t}$'s for our trajectories is a convenient form of the density-of-states. From this we can determine enthalpy, entropy, heat capacity etc. at "ANY" $\beta, P,$ and pH.

Relative free energies of the a_i and b_i ionization states can be calculated as in equations of section 3.10 and a pKa can be calculated,

$$pKa = \frac{-\log_{10} e}{kT} \{ G(a_i \mid \beta, P, \mu_H, \Lambda) - G(b_i \mid \beta, P, \mu_H, \Lambda) \}$$

4.1.1 1st step, Calculating BDE^{calc}'s for every *type* of titratable site

Making and breaking the covalent bond of the titratable proton is a quantum effect, and cannot be simulated with the classical MD force field. Also, AMBER[78] was not designed to make or break chemical bonds, so there are no such parameters in the AMBER force field. Therefore the

first step is to calibrate our method by calculating Bond Dissociation Energies (BDE^{calc}) that are consistent with the AMBER force field for each *type* of titratable amino acid. This allows us to do constant pH simulations with the classical MD force field by factoring in the net "before and after" protonation effects. This is standard procedure for most constant pH simulation software. So for Cysteine for example,

$$BDE_{cys}^{calc} = -(pKa_{cys}^{*} - pKa_{cys}^{\exp})$$

$pKa_{cys}^{\exp}$ is the known *pKa* for isolated Cysteine

pKa_{cys}^{*} is the simulated *pKa* for which our MD/MC-WHAM algorithm calculates titration for the solvated model of a single Cysteine amino-acid, with the backbone capped with the Acetyl and N-Methyl groups.

Similarly for every type of titratable amino acid, we build a solvated model of the isolated amino acid (with the Acetyl and N-Methyl blocking groups) and calculate pKa_{TA}^{*}, (where (TA) is the titratable amino acid) and therefore obtain BDE_{TA}^{calc}. This only has to be done once for every force field. If the BDE^{calc} numbers for a force field are already published, then those numbers can be used and this step skipped.

4.1.2   2$^{\text{nd}}$ step: using the BDE^{calc}'s for *pKa* calculations

As we will see in the next chapter, calculating the BDE^{calc}'s is not an insignificant amount of work, but fortunately, it only needs to be done once for a force field, or published values can be used. The second step is to use these BDE^{calc} correction numbers to calculate the pKas of titratable amino acids in proteins. All that is required is the insertion of the BDE^{calc} values into the appropriate place in the input file for our MD/MC algorithm. Then the MD/MC algorithm can be run on a model of the protein and the trajectories analyzed via WHAM to yield pKa or other thermodynamic results. For example if we are simulating a protein with a Cysteine titratable site of interest and this site was the only site in the protein allowed to be titratable, the inclusion of the BDE_{cys}^{calc} term in the input for the MD/MC algorithm simply automates the modification of the effective energy of the system according to

$$\beta(U + PV) - (\log 10)(pH - BDE)l$$

where β, P and pH are state variables, set as input at the start of the MD/MC simulations. They represent temperature (1/kT), pressure and pH respectively.

U, V and l are the configurational variables of potential energy, volume and proton bath count respectively. When the single site protein system is deprotonated, $l = 1$ and when it is protonated, $l = 0$.

4.2 WHY START WITH THE BDE^{calc} FOR CYSTEINE?

As mentioned previously, BDE^{calc}'s for all of the titratable sites must first be worked out. This dissertation reports on the BDE^{calc} for Cysteine only.

4.2.1 Cysteine's simplicity & well defined Force Field parameters

Cysteine has one of the smallest and simplest side chains, so it is the smallest and simplest of the titratable amino acids.
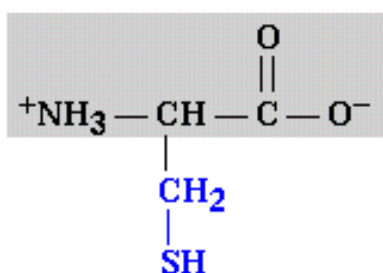
Figure 24: Cysteine
(www.mcmp.purdue.edu:4443/~mcmp304/AATutorial/arg-popup.shtml)

Amino Acid	mass, amu
Cysteine	103.1
Aspartate	114.1
Glutamate	128.1
Lysine	129.1
Histidine	137.1
Arginine	157.2
Tyrosine	163.1

Table 2: Mass of titratable amino acids

The AMBER8 ff03 force field parameters for the different ionization states of cysteine are well defined[79]. Having "good" ionization state parameters minimizes errors due to the force field, allowing us to focus on any systematic errors our methods introduce.

4.2.2 $BDE_{thio-methane}^{exp}$ allows for BDE_{cys}^{calc} comparison

The bond dissociation energy for the hydrogen-sulphur (H-S) bond (the site from which the proton dissociates from the molecule) in the thio-methane $H-SCH_3$ molecule ($BDE_{H-SCH_3}^{exp}$) is well defined[80]. This allows us to compare the calculated dissociation of the H-S bond in cysteine (BDE_{cys}^{calc}) with the experimental dissociation energy of the H-S bond in $H-SCH_3$. This is a very suitable comparison because the $H-S$ bond in cysteine ($H-SCH_2-\cdots$) and in thio-methane ($H-SCH_3$) have very similar chemical properties. This is an important comparison because it is a strenuous test for validating our methods and the quality of the force field model used for Cysteine.

4.2.3 Importance of $pKa(cys)$ calculations

pKa methods are most challenged when attempting to reproduce large pKa shifts for ionizable groups in protein interiors. These buried ionizable groups therefore form good benchmarks for testing pKa methods. The Asp-Cys buried network in Thioredoxin is such a network. The pKa shift of the Asp26 is among the highest observed, 5.3 pH units[81]! Therefore the $BDE's$ for Cysteine and Aspartic acids have the highest priority so that we can test our methods on the buried Asp-Cys network in Thioredoxin.

The thiol group in Cysteine's side chain makes it one of the most chemically reactive. For related reasons, cysteine has general biological importance. Many folded proteins owe their shape to disulfide bonds between cysteine molecules. Cysteine in the active site of cysteine protease assists binding the substrate and assists in the catalytic activity of the protease (see section 1.4.3 for discussion of a similar protease).

4.3   MD/MC ALGORITHM

4.3.1   MD/MC algorithm based on sander7 code

This algorithm is an extensive modification of the AMBER7 sander code (however the force field parameters used in our systems are the AMBER8 ff03 set). The modifications can be put into two groups.

One group of modifications concerns those relating to the sander input of two (instead of one) parameter files. So for the single cysteine system, these two parameter sets have all the information necessary for the four cysteine microstates. For a system of several titratable sites, these two parameter files will have all the information necessary for the MD/MC algorithm to swap various atom and bond parameters to create any of 2^N possible system ionization states ($> 4^N$ protonation microstates, considering each site has 4 or more microstates). One exception to this 2-parameter file method is Histidine, for which we have hard-coded some of the parameters into the MD/MC algorithm (because 2 parameter files is not enough to contain all three Histidines microstate information because no one microstate can be obtained simply by rotation of a dihedral). All microstates of the titratable sites are modeled after the corresponding AMBER8 ff03 amino acid versions.

The other group of modifications more directly concerns the Monte Carlo microstate selection code, which is responsible for making the probabilistic decision of which microstate to choose for each titratable amino acid.

One MD/MC cycle consists of an MD sub-cycle and a MC sub-cycle. The MD sub-cycle code is essentially the same as the regular sander code. During this part of the cycle, most of the usual MD features are available. For our pKa calculations, the MD simulation conditions are constant N, T and P.

The MC sub-cycle code allows addition or removal of protons from or to a proton bath with a pH (and $BDE^{calc}\text{'}s$) that are specified in the MC input file. This input file containing MC simulation parameters also specifies how many MD steps and how many MC sweeps are in the cycle. During this part of the cycle, parameter swapping occurs in order to generate both ionization states. However, to generate all of the microstates, protons on the titratable sites are allowed to rotate about their connecting dihedrals (see Figure 21: Cysteine microstates). For this

137

reason, the configuration is not strictly fixed. As per discussion in section 2.2, we therefore use the term "fixed configuration" to represent the whole system except for the protons on the titratable sites. In order to preserve detailed balance, the other microstates are generated from any one microstate by rotating the proton's dihedral by a fixed quantity (120° for cysteine), as opposed to rotating the proton's dihedral to its minima position.

4.3.2 Microstate modeling

4.3.2.1 Amber8* ff microstate models

*In our models, states with Ghost Atoms differ slightly from corresponding Amber8 models (see section 2.4.1).

The table below is a summary description of the microstates of each titratable amino acid. 1 e = one electron charge.

Titratable Amino Acid	Deprotonated charge	Protonated Charge	number of deprotonated states	number of protonated states	Total number of microstates
CYS (Cysteine)	-1 e	0 e	1	3	4
ASP (Aspartic Acid)	-1 e	0 e	1	4	5
GLU (Glutamic Acid)	-1 e	0 e	1	4	5
LYS (Lysine)	0 ec	+1 e	3	1	4
ARG (Arginine)	0 ec	+1 e	5	1	6
TYR (Tyrosine)	0 ec	+1 e	1	3	4
HIS (Histidine)	0 ec	+1e	2	1	3

Table 3: Titratable Amino Acid Microstates

138

More microstate detail for each titratable residue is given in the following sections.

4.3.2.1.1 Cysteine

4.3.2.1.1.1 Cysteine Atom Names

```
      H      HA   O
      |      |    ‖
 —— N —— CB —— C ——

   HB2 —— CT —— HB3
            |
          SG —— HG
```
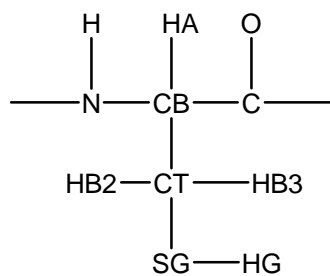
4.3.2.1.1.2 Cysteine Atom Types

```
        state 0                                    state 1

      H      H1   O                             H      H1   O
      |      |    ‖                             |      |    ‖
 —— N —— CT —— C ——                        —— N —— CT —— C ——

   H1 —— CT —— H1                           H1 —— CT —— H1
            |                                        |
          SH —— HV                                 SH —— HS
```
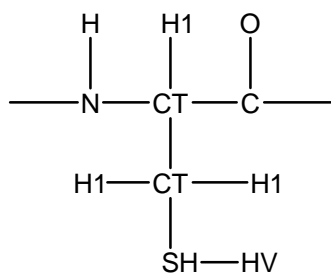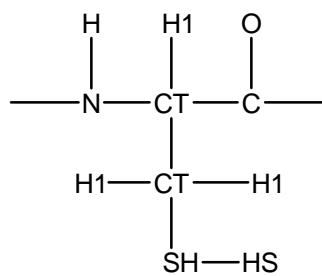
4.3.2.1.1.3 Cysteine Microstates

state 0   state 1   state 2   state 3

dihedral
$-120^0 <$HS-SH-CT-CT$<=120^0$ $0^0<$ dihedral $<=240^0$ $240^0<$dihedral$<=360^0$

140
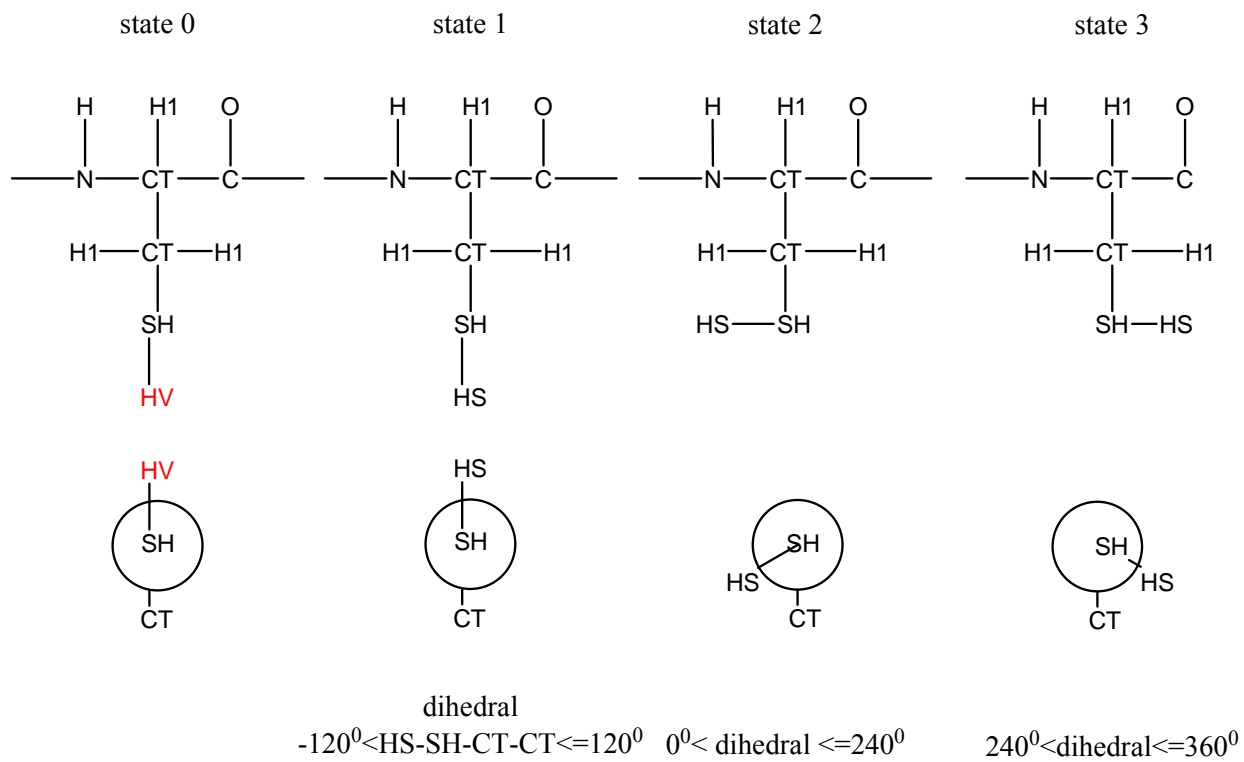
4.3.2.1.1.4 Cysteine charges

atom names	atom type (deprotonated)	atom type (protonated)	charge (deprotonated)	charge (protonated)
N	N	N	-0.416	-0.396
H	H	H	0.272	0.295
CA	CT	CT	-0.035	-0.035
HA	H1	H1	0.051	0.141
CB	CT	CT	-0.241	-0.221
HB2	H1	H1	0.112	0.147
HB3	H1	H1	0.112	0.147
SG	SH	SH	-0.884	-0.285
HG	HV	HS	0.0	0.189
C	C	C	0.597	0.643
O	O	O	-0.568	-0.585

4.3.2.1.2 Lysine

4.3.2.1.2.1 Lysine Atom Names

```
  H     HA    O
  |     |     ||
——N ——CA —— C——
        |
 HB2—CB—HB3
        |
 HG2—CG—HG3
        |
 HD2—CD—HD3
        |
 HE2—CE——HE3
        |
 HZ1—NZ—HZ2
        |
       HZ3
```
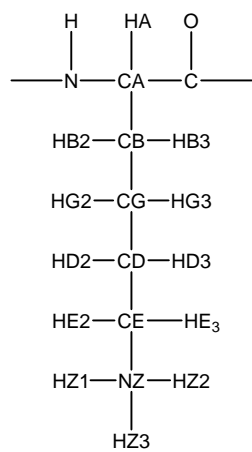
4.3.2.1.2.2 Lysine Atom Types

```
  H      H1     O              H      H1     O
  |      |      ||             |      |      ||
——N —— CT —— C ——           ——N —— CT —— C ——
         |                           |
  HC —— CT —— HC               HC —— CT —— HC
         |                           |
  HC —— CT —— HC               HC —— CT —— HC
         |                           |
  HC —— CT —— HC               HC —— CT —— HC
         |                           |
  HP —— CT —— HP               HP —— CT —— HP
         |                           |
  H —— N3 —— H                 H —— N3 —— H
         |                           |
         H                           HV

      state 0                     state 1
```
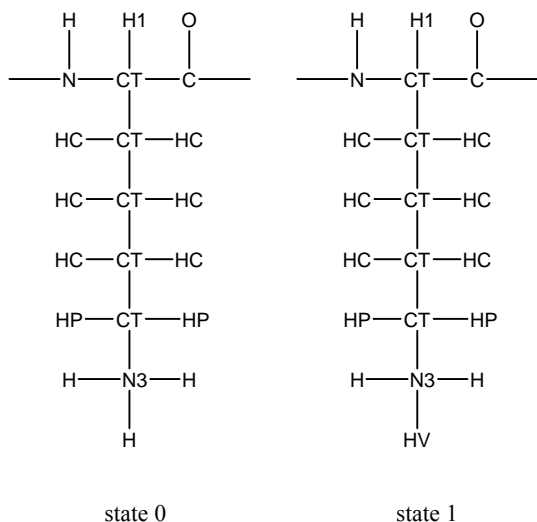
4.3.2.1.2.3 Lysine Microstates

```
  H    H1    O         H    H1    O         H    H1    O         H    H1    O
  |    |     ||        |    |     ||        |    |     ||        |    |     ||
——N — CT — C ——      ——N — CT — C ——      ——N — CT — C ——      ——N — CT — C ——
       |                   |                   |                   |
 HC — CT — HC        HC — CT — HC        HC — CT — HC        HC — CT — HC
       |                   |                   |                   |
 HC — CT — HC        HC — CT — HC        HC — CT — HC        HC — CT — HC
       |                   |                   |                   |
 HC — CT — HC        HC — CT — HC        HC — CT — HC        HC — CT — HC
       |                   |                   |                   |
 HP — CT — HP        HP — CT — HP        HP — CT — HP        HP — CT — HP
       |                   |                   |                   |
 H — N3 — H          HV — N3 — H          H — N3 — HV          H — N3 — H
       |                   |                   |                   |
       H                   H                   H                   HV

    state 0             state 1             state 2             state 3
```
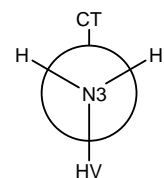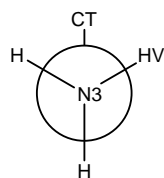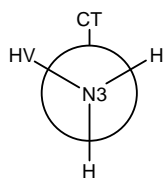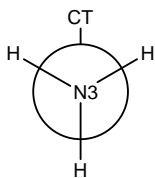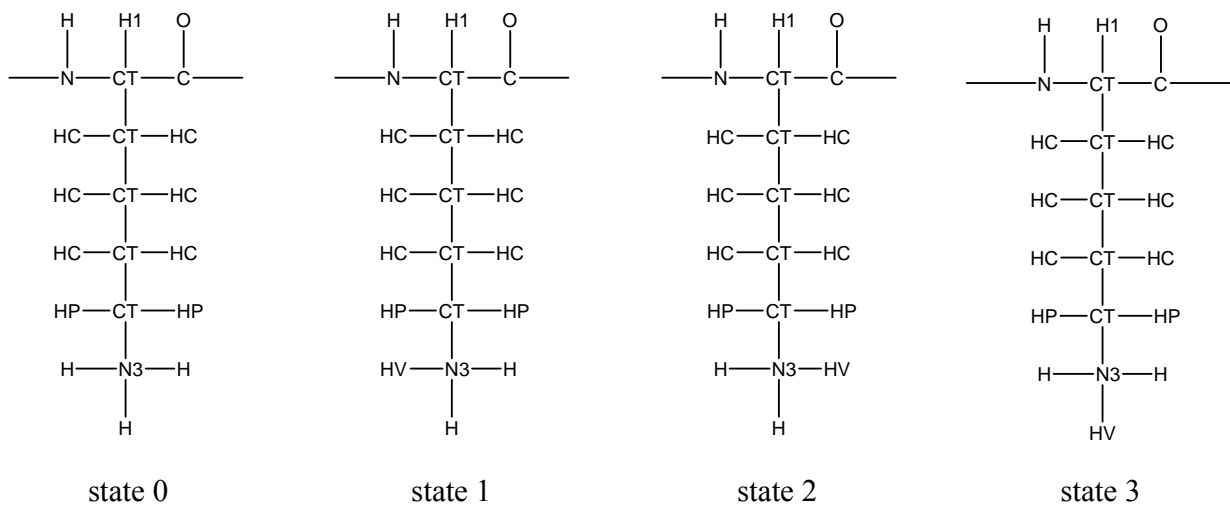
```
       CT                  CT                  CT                  CT
       |                   |                   |                   |
  H    O    H        HV    O    H        H     O    HV       H     O    H
       |                   |                   |                   |
       H                   H                   H                   HV

                    0<HV-H3-CT-CT<=120   120<dihedral<=240    240<dihedral<=360
```

4.3.2.1.2.4 Lysine Atom Charges

atom names	atom type (protonated)	atom type (deprotonated)	charge (protonated)	charge (deprotonated)
N	N	N	-0.4359	-0.4157
H	H	H	0.2513	0.2719
CA	CT	CT	-0.0388	-0.0721
HA	H1	H1	0.1295	0.0994
CB	CT	CT	-0.1083	-0.0485
HB2	H1	H1	0.0452	0.0340
HB3	H1	H1	0.0452	0.0340
CG	CT	CT	0.0333	0.0661
HG2	H1	H1	0.112	0.0104
HG3	H1	H1	0.112	0.0104
CD	CT	CT	-0.0478	-0.0377
HD2	HC	HC	0.0707	0.0115
HD3	HC	HC	0.0707	0.0115
CE	CT	CT	-0.0700	0.3260
HE2	HP	HP	0.1195	-0.0336
HE3	HP	HP	0.1195	-0.0336
NZ	N3	N3	-0.2504	-1.0358
HZ1	HV	H	0.2946	0.0
HZ2	H	H	0.2946	0.3860
HZ3	H	H	0.2946	0.3860
C	C	C	0.7351	0.5973
O	O	O	-0.5632	-0.5679

4.3.2.1.3 Arginine

4.3.2.1.3.1   Arginine Atom Names

```
      H     HA    O
      |     |     ||
  ——— N —— CA —— C ———

    HB2 —— CB —— HB3

    HG2 —— CG —— HG3

    HD2 —— CD —— HD3

           NE —— HE

           CZ
  HH11    /  \    HH21
    \    /    \    /
    NH1        NH2

     |          |
    HH12       HH22
```
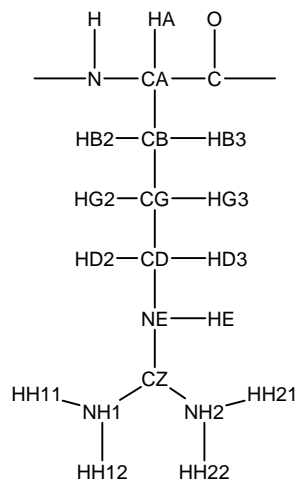
4.3.2.1.3.2   Arginine Atom Types

```
      H     H1    O              H     H1    O
      |     |     ||             |     |     ||
  ——— N —— CT —— C ———       ——— N —— CT —— C ———

    HC —— CT —— HC             HC —— CT —— HC

    HC —— CT —— HC             HC —— CT —— HC

    HC —— CT —— HC             HC —— CT —— HC

           N2 —— H                    N2 —— H

           CA                         CA
  H—— N2       N2 —— H      HV—— N2       N2 —— H
      |          |              |          |
      H          H              H          H

        state 0                    state 1
```
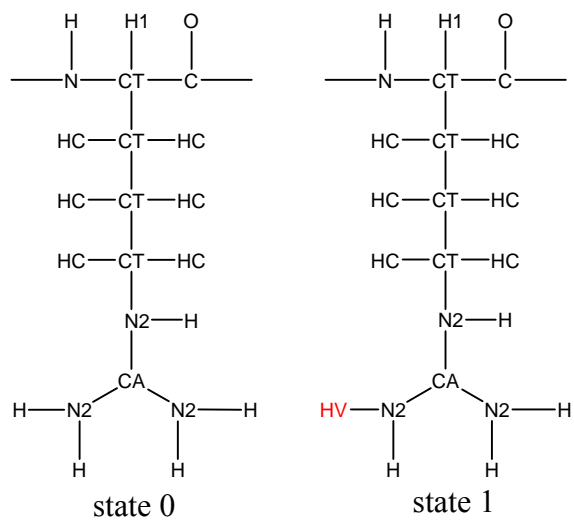
4.3.2.1.3.3 Arginine Microstates

state 0

state 1   state 2   state 3   state 4   state 5
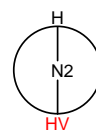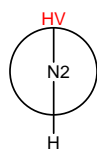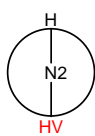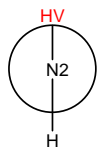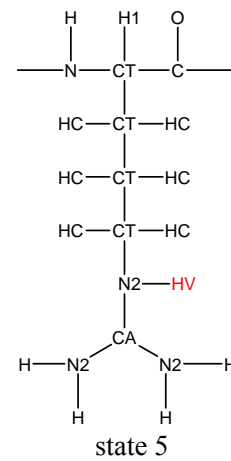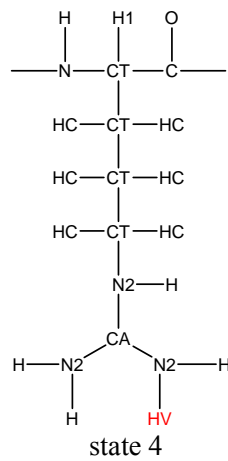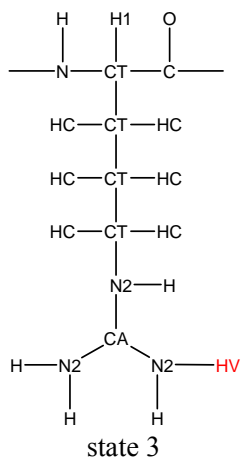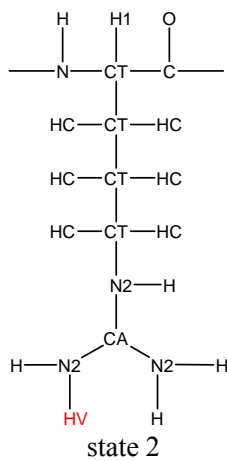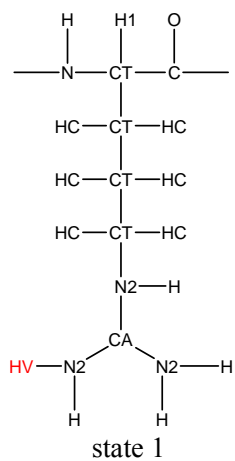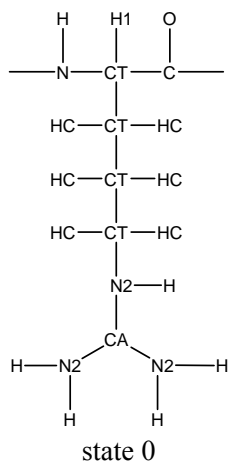
-90<HV-N2-CA-N2<=90   90<DIHEDRAL<=-270   -90<dihedral<=90   90<dihedral<=-270

4.3.2.1.3.4 Arginine Charges

atom names	atom type (protonated)	atom type (deprotonated)	charge (protonated)	charge (deprotonated)
N	N	N	-0.3009	-0.3009
H	H	H	0.2337	0.2337
CA	CT	CT	-0.1314	-0.1314
HA	H1	H1	0.0533	0.0533
CB	CT	CT	0.0367	0.0367
HB2	HC	HC	0.0280	0.0280
HB3	HC	HC	0.0280	0.0280
CG	CT	CT	0.0125	0.0125
HG2	HC	HC	0.0030	0.0030
HG3	HC	HC	0.0030	0.0030
CD	CT	CT	0.1263	0.1263
HD2	H1	H1	0.0681	0.0681
HD3	H1	H1	0.0681	0.0681
NE	N2	N2	-0.4649	-0.4649
HE	H	H	0.3263	0.3263
CZ	CA	CA	0.5655	0.4655
NH1	N2	N2	-0.6858	-0.7858
HH11	H	HV	0.3911	0.0
HH12	H	H	0.3911	0.2811
NH2	N2	N2	-0.6858	-0.7858
HH21	H	H	0.3911	0.2911
HH22	H	H	0.3911	0.2911
C	C	C	0.7303	0.7303
O	O	O	-0.5783	-0.5783

4.3.3   Problem of transitions

Vigorous transitions from one microstate, directly or indirectly, to all other microstates are important for proper configurational and ionization state sampling. Numerically, it is also

important for good statistics. The problem is that under the conditions of short simulation times (<100 picoseconds), room temperature and pressure, the single titratable site cysteine system does not sufficiently sample all microstates. At 300K, if it is in one of the protonated microstates, it will sufficiently sample the other two protonated microstates for simulation lengths on the order of tens of picoseconds. However, for the same simulation length, it will seldom go from protonated to deprotonated, or from deprotonated to protonated. This is because there are ionization state dependent solvation effects that constrain ionization state transitions. However we can achieve sufficient ionization state transitions by performing simulations at high temperatures, then use WHAM to combine all the different-temperature simulations for the purpose of 300K thermodynamic calculations.

As just mentioned, for conveniently short 300K simulations, the system cannot transition both ways to the other ionization state *at any one pH.* However, if we shift the *pH* during the simulation, we can force transitions in both directions. Even though our method does NOT use *pH* shifts to drive the system to and from ionization states, driving the system with *pH* shifts gives us a quick approximation of the amplitude of the driving factor that what would be needed. In the diagram and discussion below, we describe this driving factor in terms of a *pH* hysteresis amplitude.
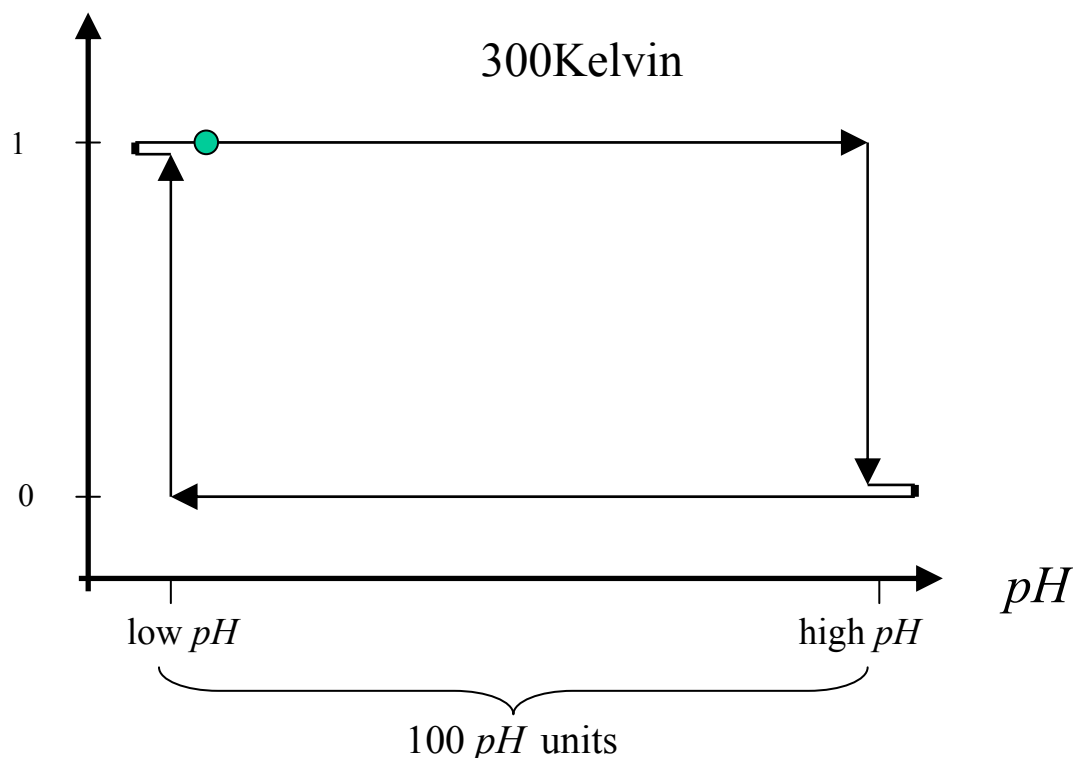
protonation

Figure 25: pH hysteresis at 300K

Figure 25 above is a symbolic demonstration of the *pH* hysteresis effect. In reality, our 300K simulations behave very much like this. Suppose we start where the green dot is, with the system in the protonated state. The arrow pointing to the right represents a series of short simulations (each about 100 MC sweeps, 2000 MD steps) of increasing *pH*. The system stays deprotonated until the pH has increased about 100 *pH* units, at which point it turns deprotonated (the downwards arrow on the right). The *pH* is then decreased, but the system stays deprotonated until the *pH* drops about 100 *pH* units. The cycle is repeated.

If we ran simulations infinitely long, there will be no hysteresis. The hysteresis comes about because we have to run short simulations. Even though the size of the hysteresis will decrease if we used longer simulations, the plot gives us a ballpark idea of the size of the factor needed to drive the system across the solvation shell for short simulations. It is about $100 pH$ units $@300\text{K}$. The fact that it is so large simply means that we are a long way from getting sufficient transitions. We do not see this as being reflective of the natural height of the barrier because our simulations are so short. This makes it clear why the MC selection routine

will very rarely to never cross the barrier for short simulations at 300K. As the temperature of the short simulations is increased, the amplitude of the hysteresis drops. At 800K and ***at one pH,*** it is possible to see transition cycles in the system on a simulated time scale of tens of picoseconds (i.e., no *pH* hysteresis amplitude for simulations of this length at this temperature). At 1320K, transitions occur about every 1000 femtoseconds (1000 MD steps), and at 2200K, transitions occur about every 20 femtoseconds.

Taken at face value, the hysteresis plot seems to suggest that it would take 100 *pH* units at 300K to cross the solvation shell barrier. This would then lead to the conclusion that our simulations require proton concentration changes of factors of 10^{100} which is clearly unphysical and raises suspicions about if our model even comes close to representing reality. Recall that these simulations are extremely short and we are only observing one or two transitions. Therefore the width of the hysteresis curve would be expected to over estimate the barrier height, probably by a significant amount. This overestimation of the barrier height is emphasized if we use the Eyring equation

$$k = \frac{k_B T}{h} e^{-\Delta G^+ / RT} \approx \frac{1}{\tau} \qquad (4.1)$$

to estimate the transition rate using a 100*pH*@300K barrier height. k, k_B, T, h, ΔG^+, R and τ are transition rate, Boltzmann's constant, temperature in Kelvin, Plank's constant, free energy of activation, molar gas constant and transition period respectively. Using a barrier height of 100*pH*@300K, the transition period works out to be 6×10^{95} seconds, which longer than the age of the universe. A better estimation of the barrier height at 300K can be determined by taking the barrier at 2200K (1 to 2 $k_B T$) and linearly extrapolating downwards to *T=300K* according to equation (4.1), which will increase the exponent by a factor of seven (2200K to 300K). This gives a barrier height of about 15 $k_B T$s, which then corresponds to a transition period of approximately 6×10^{-7} seconds (< half a microsecond). This is a much more realistic estimate of the barrier height. However, note that even with this more realistic barrier height estimate, the 600 nanosecond transition period is still not tractable.

We therefore concluded that we could not expect to observe sufficient transitions at 300K to get adequate sampling. Indeed, we found that 2200K was required to observe a large number of transitions within 100 picoseconds.

4.3.4 Low 300K Inter-Ionization transition rates

What is the nature of the solvation barrier that inhibits inter-ionization transitions for short simulations at 300K? One may think that the VDW energy differences contribute, especially when going from deprotonated to protonated. One may think this because the ghost proton in the deprotonated form has no VDW parameters, allowing the solvent will move in, then when the ghost proton tries to become protonated (VDW parameters turned on), there are steric clashes with the encroaching solvent which inhibit a transition to the protonated state. However a close look at the energetic components shows that the low 300K transition rates are due to electrostatic solvent effects. The VDW energy component is not a significant contributor.

Deprotonated cysteine has an overall charge of –1 electron charges (*e*), and the surrounding TIP3P waters (which have a dipole nature) orient themselves to minimize the potential of the electric field (the oxygen of the waters tend to point away from the negatively charged titration site, specifically the negative sulphur atom). Protonated cysteine is neutral, and the surrounding waters tend to orient themselves so that the oxygen points towards the positive proton of the *S-H* group. These solvation shell effects are so significant that the MC probability of selecting the other ionization state is too small for sufficient transitions for even the longest simulations. As for microstate transitions within an ionization state, at 300K we do observe intra protonation state transitions on the scale of picoseconds (thousands of MC steps, hundreds of MC sweeps).

4.3.5 Differing philosophies for accelerating ionization transitions

Here we will discuss different attempts at increasing the ionization-state transition rates at 300K so that there we can achieve statistically sufficient transitions for short simulations.

4.3.5.1 pH swapping, replica exchange scheme
In Figure 25, we do succeed in accessing ionization states within short simulation lengths by driving the *pH* in one direction and then another. What if we did use *pH* to drive the system instead of temperature, conducting many such Figure 25 simulations and used WHAM to connect them? There are several reasons against this idea. The main reason is the equilibrium

problem. When extreme *pH* causes the system to change ionization state, and this cycle happens a few times, the system cannot be confirmed to be in equilibrium. To verify equilibrium there would have to be many thousands of the kinds of cycles shown in Figure 25. Confirming equilibrium under those kinds of conditions would be difficult.

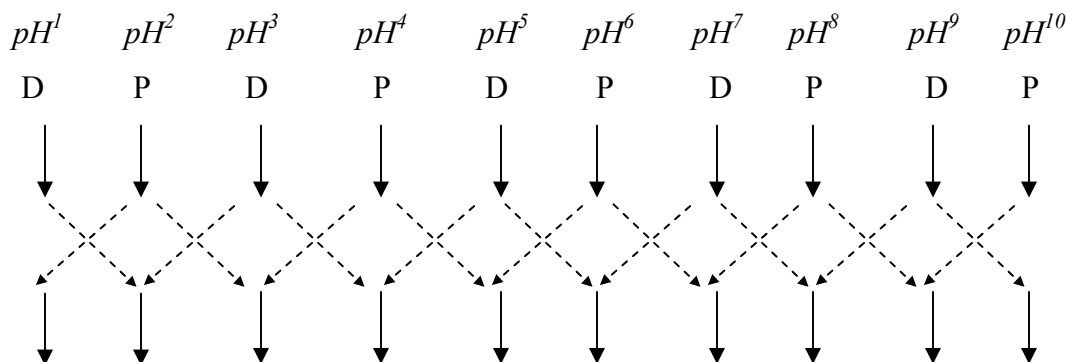We also experimented with a pH swapping, replica exchange scheme, which we will briefly discuss.

$$pH^1 \quad pH^2 \quad pH^3 \quad pH^4 \quad pH^5 \quad pH^6 \quad pH^7 \quad pH^8 \quad pH^9 \quad pH^{10}$$

$$D \qquad P \qquad D \qquad P \qquad D \qquad P \qquad D \qquad P \qquad D \qquad P$$

Figure 26: pH Replica Swapping

Each vertical track represents one of 10 different *pH*s. The starting array of simulations is a random selection of deprotonated (D) and protonated (P) snapshots. After some specified number of MD steps, there is a MC exchange attempt with its neighbor. After many such cycles, the deprotonated trajectories would tend to the higher *pHs*, and the protonated trajectories would tend towards the lower *pHs*. This scheme had some unacceptable artifacts. If one counted occupancy statistics going down a particular pH^i track, the occupancy ratio would be invariant with the range chosen for $pH^1 - pH^{10}$. One solution is to include a MC choice that kills off trajectories, so that the deprotonated-protonated ratio of the array does not stay constant. The problem with such a scheme is that one low energy trajectory quickly dominates the whole array.

4.3.5.2 Trying different FF parameters to improve transition rates

The ionization state dependent solvation effects discussed in section 4.3.3 Figure 25 are for Amber8 Cysteine and TIP3P water at 300K. Are there amino acid or water force field parameters that reduce the size of the ionization barrier? We did some manual and significant modification of the Cysteine atom charge distribution to see how that would affect the ionization barrier. It did not make a significant enough impact on the size of the barrier. Since the amino

acid models only vary subtly from one another, we concluded that different force field models of Cysteine would not help.

What about turning on polarization in the solute and the solvent? Polarization should definitely help. What about using different water models? There is a long list of things to try. However we realized that heading down this road meant that we were pursuing a philosophy that we did not like. We were forcing the model parameters to fit our method, instead of designing our method to work with the most commonly accepted model parameters.

4.3.5.3 Titratable water

In nature the water molecules are titratable, which facilitates the transfer of protons to or from a titratable site (the titratable behavior of water plays an important role in proton dynamics of biomolecules and this is discussed in section 1.4.1). Because we are using an un-titratable water model (TIP3P), it is possible that the implementation of a titratable water model will reduce the solvation shell effects. However modeling titratable water has the following disadvantages. Because water is highly mobile (in nature and in our simulations) and also because a titrated water molecule affects the hydrogen bonding interaction with its neighbors, it would not suffice to limit the titratable treatment to those water molecules that interact with the titration site. Therefore all of the water molecules of the model will have to be made titratable, which for a typical system may number over ten thousand. This means that modeling titratable water will face computational feasibility challenges. Besides even with titratable water there may still be solvation barriers that prevent short simulations from yielding ionization transition occurrences that are high enough for good sampling.

4.3.5.4 Use simulated annealing ensemble to accelerate transitions

We decided to design our method so that it works with the most widely accepted amino acid parameters (Amber ff03) and the most recommended water model for the ff03 parameters, the TIP3P water model. We use high temperature simulations to accelerate the ionization-state transitions and, combined with lower temperature simulations, we use WHAM to join these simulations that vary over a wide range of temperatures for thermodynamic calculations at 300K.

4.4   SIMULATED ANNEALING ENSEMBLE

4.4.1   Simulated Annealing Ensemble P-T path

Circumventing the ionization barrier for the single site models (like our solvated Cysteine model) requires that we first elevate the temperature, and then the high temperature trajectories are joined with lower temperature trajectories via histogram overlaps. Every simulation in the entire range is unquestionably in equilibrium. However, we also want to make sure that the *Pressure-Temperature ENSEMBLE PATH* is also in equilibrium, and does not cross phases. The *Pressure-Temperature ENSEMBLE PATH* describes the relation between the trajectories in the ensemble. It is **NOT** the P-T path of some single simulation, as in the common sense use of "simulated annealing", because each and every simulation of the ensemble is always equilibrated, with a pressure, temperature and pH that do **NOT** change.
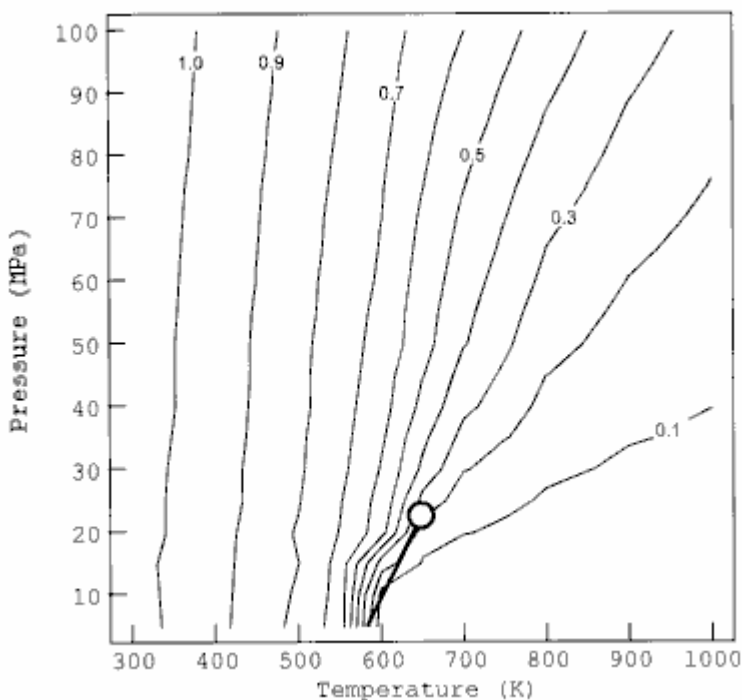
4.4.2   Critical point of TIP3P water

Figure 27: TIP3P Phase Diagram for high T-P[82, 83] (Kazuyoshi UEDA *et al*, 2004)

153

Figure 27 above shows the Critical Point region for TIP3P water. The plot shows constant density lines, with values in units of $g.cm^{-3}$. The experimental critical point of water is shown as the dark circle (647K, 22MPa, density 0.32 $g.cm^{-3}$). The density lines converge around the experimental critical point. This means that in this region, small changes in temperature or pressure cause big changes in density. Since this region is near the experimental critical point, one can conclude that TIP3P does a good job at reproducing the critical point of water. So the critical point of TIP3P water is near 647K, 22MPa, density 0.32 $g.cm^{-3}$.

The calculations pertaining to Figure 27 are based on simulations carried out with the CHARMM25 program[84]. It is reasonable to assume that using our AMBER ff03 TIP3P parameters and our simulation protocols, we would calculate a region for the critical point similar to that obtained in Figure 27. This is because TIP3P parameters and the simulation protocols used in the Figure 27 simulations are not so different from our TIP3P parameters and simulation protocols that we would expect significant changes in the position of the region of the critical point.

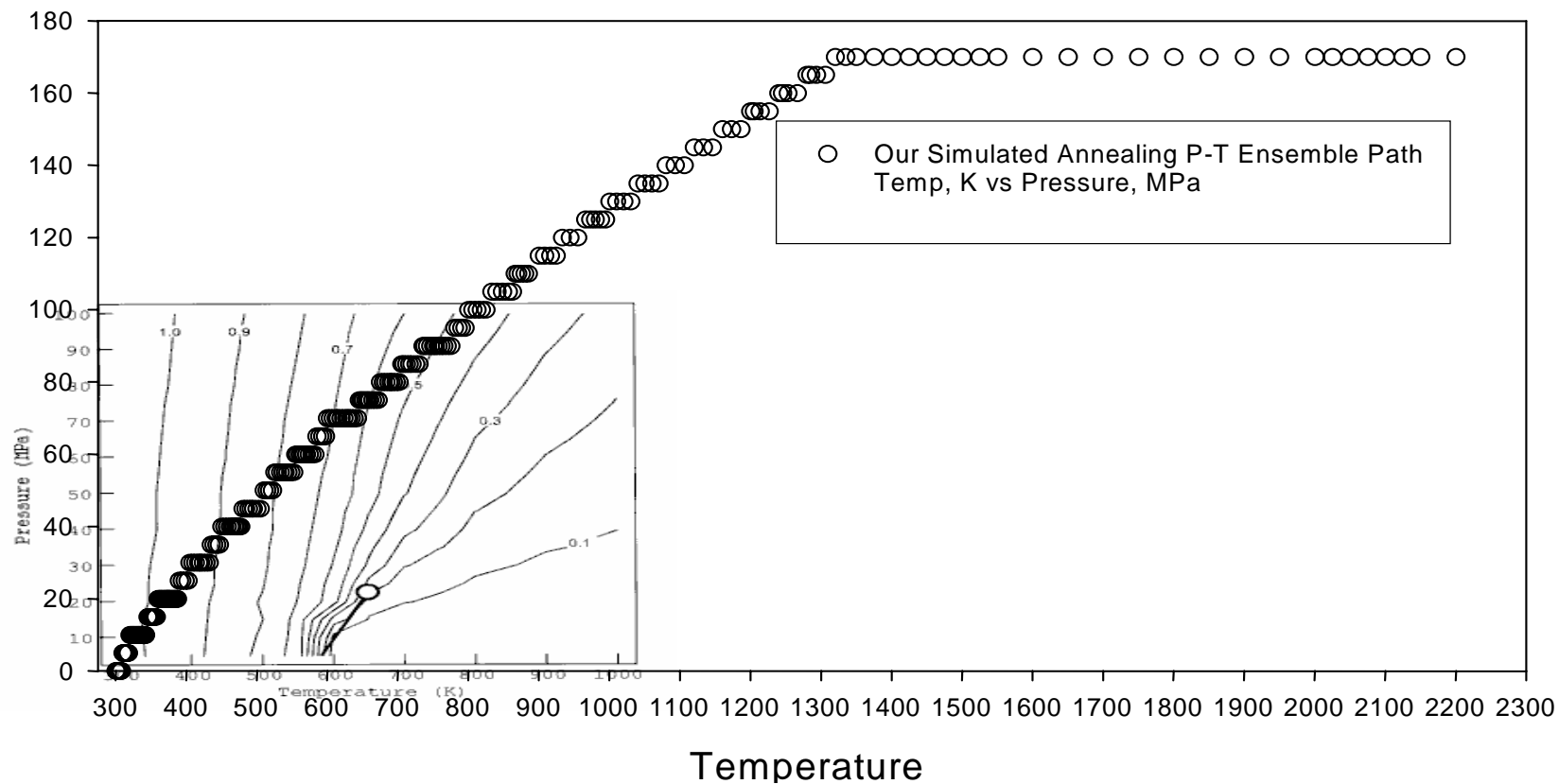### 4.4.3 Our P-T Path: Avoiding Phase Transition

Figure 28: Our P-T Ensemble Path

In above, we have superimposed the P-T plot of our Simulated Annealing Ensemble against the TIP3P phase diagram. This shows that our Simulated Annealing Ensemble Path is far away from the critical point, therefore avoiding any anomalies that result from crossing phases.

4.4.4 Our P-T path step size

The step sizes for our P-T simulated annealing ensemble path are shown in Figure 28 above. This begs an obvious question, "How do we choose the step size? Why not only a handful of simulations in the P-T phase space range shown in Figure 28?" Our step size must be such that we have sufficient histogram overlap between the effective energies of the simulations. Ideally, in order to maximize computational efficiency, our step size should be as large as would allow sufficient histogram overlap for some target level of precision of our calculations. Precision targets are spoken about at length in section 5.2.2.

4.5 WHAM ALGORITHM

In our MD/MC-WHAM method, the WHAM algorithm is put to use after the MD/MC trajectories are generated. The WHAM algorithm is able to combine all of the information from all to the trajectories for yielding a good density-of-states description.

4.5.1 Histogram overlaps

4.5.1.1 Importance of histogram overlaps

For WHAM to be able to combine the information from two (or more) simulations, the two simulations must be close enough to each other in the energy landscape such that there is "good" overlap of the histograms of the effective energies of the two simulations[85]. The Boltzmann Factor for our systems has the form $e^{-E_{effective}}$ where the effective energy of our systems

is $E_{effective} = \dfrac{1}{kT}(E_{pot} + PV) - \ln 10 \bullet pH \bullet L$. We aimed at having the histograms of the Effective

Energy of the simulations of the P-T ensemble path to overlap such that the distance between the histogram peaks was no more than one standard deviation. The result is the P-T phase space step size as shown above in Figure 28. The following two sections will detail exactly how we came to the step size results that we did.

4.5.1.2   Heat capacity calculation for approximate histogram spacing

Figure 28 shows that the highest temperature system of our ensemble is a system at 2200K and 1700atm. We used the WHAM feature that evaluates the heat capacity equation shown in section 3.9 to calculate the heat capacity of this system. This gives us an approximation of the width of the effective energy histogram, and also how far the temperature can be dropped to incur a histogram shift of half the histogram width. Hence we can calculate what the next lower temperature should be for the ensemble (of course, manual inspection of the histograms was used to verify the heat capacity predictions). We can repeat this calculation to get the next lower temperature, and so on. However, we can only do this as far down as 1320K, 1700atm. From 2200K down to 1320K, the pressure stays constant, so the "heat capacity at constant pressure" is useful for determining histogram overlaps. However, from 1320K downwards, the pressure also drops, so we can no longer use the constant-pressure heat capacity as a guide. In this range we simply had to measure the mean of the histograms, the r.m.s.d. of the histograms, and most importantly, count the number of snapshots in the overlap regions to make sure there was sufficient overlap.

4.5.1.3   Histogram Standard Deviation calculation overlap count

One useful feature of the WHAM code is to simply calculate the effective energy mean and r.m.s.d. for all of the trajectories of a dataset, and to write out the information in a convenient format. This allows the user to get quick approximations for where further simulations are needed for sufficient overlap.

Another useful feature of our code is the option to write out the calculated effective energy of every snapshot in the dataset, in a format convenient for plotting software like

SigmaPlot. The histogram function of the software can then be used to create histograms and to calculate the overlap count of the histograms.

4.5.1.4   Calculation convergence and histogram overlap correlation

It is intuitive to think that better histogram overlap will allow WHAM to give a better density-of-states description, which in turn will yield better result precision. However there is another practical advantage that is felt long before the stage of result precision analysis. Better histogram overlap causes more rapid convergence of the WHAM iterative scheme. The additional computer time used to generate more trajectories for the improvement of histogram overlap is more than made up for because less computer time is spent in WHAM to converge the free energies ($g_m's$). Up to a point. Our focus in this work is to hit calculated *pKa* precision targets, so we tended to err on the side of excessive overlap.

4.5.2   pKa calculation using high temperature bridge

There are two basic requirements for WHAM to work for pKa calculations:

    a)   WHAM needs trajectory snapshots that sample the appropriate regions of the effective energy landscape for which we are conducting pKa calculations. For example, if all we had were the 1280K high temperature data shown in Figure 31, we would be able to calculate, with reasonable accuracy, the pKa of the system for temperatures and pressures in the ranges of 1280-+20K, and 1650+-25atm. With only the 1280K data, we would NOT be able to calculate pKas at 300K&1atm with any accuracy.

    b)   WHAM needs histogram overlaps in order to effectively incorporate data from a range of temperatures and pressures. For example, suppose we wanted to calculate the pKa of the system at 1160K&1500atm, and we had the data for 1320K and 1160K shown in Figure 31. Including the 1320K data is useful because it helps improve the density of states description, hence they can contribute towards improving the accuracy of the result. However, if WHAM only had the 1320K and 1160K data, it will not effectively (or properly) incorporate the 1320K data, because there is no histogram overlap between the 1320K data and the 1160K data (see Figure 31). WHAM needs the snapshots for the

trajectories at 1320K, 1280K, 1240K, 1200K and 1160K. In other words, there must be "links" or histogram overlaps for WHAM to properly incorporate data from a wide range of conditions.

For a *straightforward* application of WHAM, there is one more important, and intuitive requirement for WHAM. The trajectories must have a statistically sufficient number of transitions. Consider the high temperature simulations of Figure 31 page 165. With this data, a *straight forward* application of WHAM is good for calculating *pKas* of the system for any temperature and pressure along our TP path (of Figure 28 page 155) in the approximate ranges of 1320K - 1160K and 1700atm-1450atm. However, if we need the pKa calculation for 300K&1atm, notice that the lower temperature data DO NOT have any transitions, so a *straightforward* application of WHAM will not work. Next I will detail how we use our WHAM algorithm to accurately perform pKa calculations such as those at 300K&1atm.

Consider the 1160K set of data, and the 1120K set of data shown in Figure 31 page 166. The 1160K data is part of the "high temperature bridge". It is a long simulation with lots of transitions and good sampling of all protonation microstates, as are all the five high temperature "bridge" simulations shown in Figure 31. Suppose we wanted to calculate the pKa for the system at 1120K&1450atm. We could simply use the 1160K&1500atm data for the 1120K&1450atm pKa calculation. However, suppose we wanted to incorporate 1120K trajectory data to improve the result, because the 1120K data better samples in the region. The problem is that the 1120K system will have less of a transition rate than the higher temperature 1160K trajectories. We will therefore need longer 1120K simulations to give us statistically sufficient sampling. So we gain much by including the 1120K simulations because it better samples the region for which we want to calculate pKas, but we lose some because there are fewer transitions. There is an alternative. With careful treatment, we can add a pair of short 1120K simulations to the 1160K dataset and get improved results for our 1120K, 1450atm pKa calculation as follows. The pair of short 1120K simulations consists of one short simulation in which the system is 100% deprotonated (1120Kd), and one short simulation in which the system is 100% protonated (1120Kp). Each member of this short pair of 1120K simulations has the same number of snapshots. Each member of this short simulation pair has a histogram that overlaps with the 1160K data histogram. Both short simulations have the same pH.

If we add such short 1120K simulation data to the 1160K data for WHAM, we will get results that are heavily dependent on the pH of the short simulation pair. The reason for this is obvious. WHAM sees an equal number of 1120K protonated and deprotonated snapshots that were generated under the same conditions, and assumes there are a healthy number of transitions (not so). It therefore gives a pKa result close to the pH of the 1120K pair, because by definition, the pKa is the pH of the system for which there is a 50/50 ratio of protonation and deprotonation occupancy. Seeing that we "get out a pKa that is close to the pH that we put in", is there a way to appropriately choose the pH of the short 1120K simulations? Yes there is. It can be done as follows:

(1)   Even though our pKa result is dependent on the pH of the short 1120Kd and 1120Kp simulations, and that we approximately "get out a pKa that is close to the pH that we put in", the result is not *completely* dependent on the pH of the short 1120K simulation pair because of the histogram overlap. The high temperature 1160K data acts as an anchor, pulling the result in the right direction, allowing for an iterative solution.

(2)   For a given protonation state, the Potential Energy + PV term of a snapshot is independent of the pH conditions of the simulation. So if we generated a simulation with no transitions, such as the 1120Kd or 1120Kp simulations, and we repeated the no-ionization state transition simulations with the same starting point and conditions except that the pH was different, we would get the same trajectory and the same Ep+PV terms for the snapshots. This means that for an iterative solution, we don't have to regenerate 1120Kd and 1120Kp simulations with different pHs every time we want to see how a different short pair pH effects our results. All we need is one 1120K-simulation pair of any pH.

For our 1120K, 1450atm pKa calculation, for the reasons outlined above, we are able to implement an iterative solution as follows:

(a)   Feed our WHAM algorithm the 1160K data, and the data for the pair of 1120K simulations. The 1120K simulation pair can have any pH.

(b)   WHAM will give a *pKa* result that is heavily influenced by the pH of the 1120K pair data, but will be pulled in the right direction.
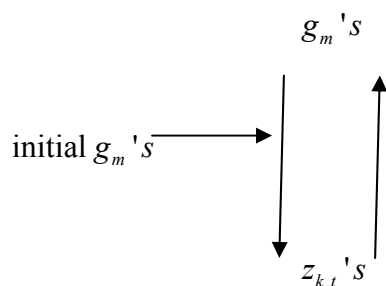
(c)   Our WHAM algorithm will take this calculated pKa and use it as the "new and improved" pH of the 1120K simulation pair, and redo the calculation.

(d)   The process is repeated until the calculated pKa converges, yielding an accurate value for the pKa@1120K, 1450atm.

We generalize this approach to the whole data set of Figure 31 as follows:

(a)   Feed our WHAM algorithm all of the "high temperature bridging" data, and all of the short lower temperature simulation pair data. Each member of a short simulation pair must have the same number of snapshots and the same pH as its differently ionized partner. Their histograms should overlap with those of neighboring simulations. The more reasonable the initial pH's chosen, the fewer iterations are needed for pKa convergence. In practice, the algorithm is smart enough to assign the same pH to each member of a low temperature pair, overriding the pHs in the file headers. In practice, the initial pH's assigned **CAN** be far from reasonable yet convergence will still occur only slightly less rapidly than if the initial pHs were closer to the converged values.

(b)   In the first iteration, our WHAM algorithm will calculate initial pKas for all of the simulation conditions represented by the short simulations.

(c)   Our WHAM algorithm will use this set of pKas to reassign the "improved" pHs for all of the short simulations, and repeat the calculation.

(d)   After several hundred of these "pKa iterations", the pKas will converge, yielding an accurate set of pKa values that correspond to all the simulation conditions of the short simulations, including the pKa@300K, 1atm for our Cysteine system. Such results are shown in Figure 32 page 169.

The iterative scheme of our WHAM algorithm can be summarized in the following diagrams.

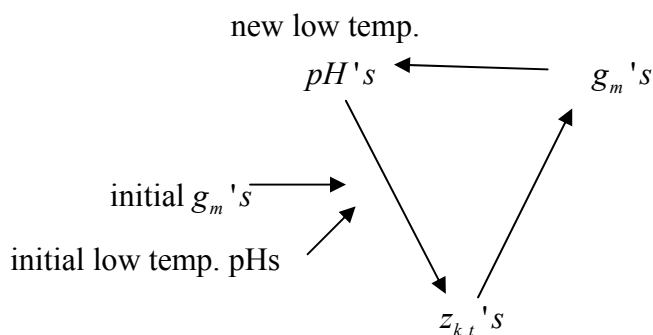Original WHAM iteration scheme   Our WHAM iteration scheme

Figure 29: WHAM pH Iterative Scheme

Details of the calculation of each element of the iteration are given by the equations in section 3.9.

The theory of our method can be summarized as follows: pH and occupancy ratio can be considered to be conjugate variables. In typical usage, the state variable pH is fixed and the configuration variable of occupancy ratio is observed. In our method we reverse engineer things. For the low temperature simulations, we set occupancy ratio = 1, and we ask WHAM let the *pH* float until it converges to the correct value that would make the (occupancy ratio = 1) condition true, which then means that the *pKa = converged pH*.

In short, for the low temperature simulations, we modify the conjugate pair

$$(pH, occupancy\ ratio) \tag{4.2}$$

so that it becomes

$$(pKa=pH, occupancy\ ratio=1) \tag{4.3}$$

162

5.0 BIOPHYSICAL RESULTS FOR CYSTEINE

5.1   HIGH TEMP. TITRATION CURVE

Our computational high temperature titration curve for Cysteine matches analytical expectations. Consider the following analytic analysis for the titration of a titratable site.

$$\frac{occ^1}{occ^0} = e^{\beta \Delta F_0 - (\ln 10) pH} \quad \text{where} \quad \frac{occ^1}{occ^0} \quad \text{is the protonated-deprotonated occupancy}$$

ratio, $\beta = 1/kT$ where T is the temperature of the simulation, pH is the pH of the simulation and ΔF_0 is the $\langle E_p + PV \rangle$ difference between the protonated and deprotonated states (where E_P, P and V are the potential energy, pressure and volume respectively).   For all Cysteine simulations of constant temperature and pressure, $e^{\beta \Delta F_0}$ is constant, so $\ln(occ^1 / occ^0)$, the log of the occupancy ratio, should be proportional to the pH of the simulation, with a gradient = $-\ln 10$ = -2.303.

Using the occupancy data generated by our code at five simulations run at different pH, we plot the log of the occupancy ratio vs. the pH.
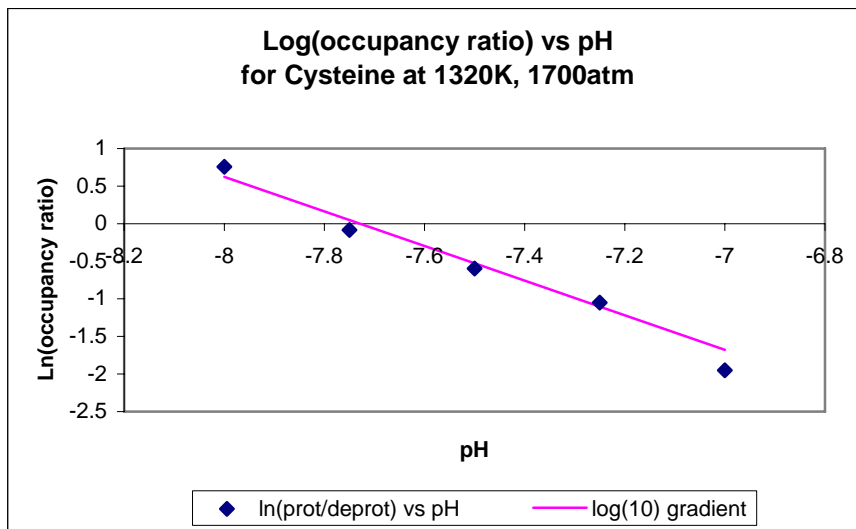
Figure 30: Titration curve for Cysteine at 1320K

This result, to a large extent, is an important validation check of our code and our methods. The reader may note that calculated pKa of Cysteine is −7.75, (intersection of the x-axis, were protonated occupancy = deprotonated occupancy) which is significantly different from Cysteine's known pKa of +8.3. This is not a bad sign. Instead this emphasizes the importance of the calculating Bond Dissociation Energy numbers. The necessity for calculating these Bond Dissociation Energies is discussed in section 4.1.1.

The reader may also note that the *pKa* calculation was done at a simulated temperature of 1320K, far higher than the temperatures we are interested in. This emphasizes the importance of our WHAM[63,64] methods for combining the information from different temperatures for our single titration Cysteine system. See Figure 31.

5.2 CALCULATED *BDE* FOR CYS

The high temperature simulations (represented in the top portion of Figure 31) are important for allowing our system to rapidly sample many configuration and protonation states, which improves the "density of states" description. We simulated about 3×10^7 MC sweeps for the higher temperatures. We simulated approximately 1000 MC sweeps for each of the 44 lower

164

temperature states of the Cysteine system. Proper equilibration of the 44 lower temperature systems at their respective temperatures pressures and ionization states requires about 30,000 service units. Then the high temperature 3×10^7 MC sweeps of the MD/MC simulation requires about 40,000 Lemieux service units. One MC sweep is made every 20 MD steps (20 fs).
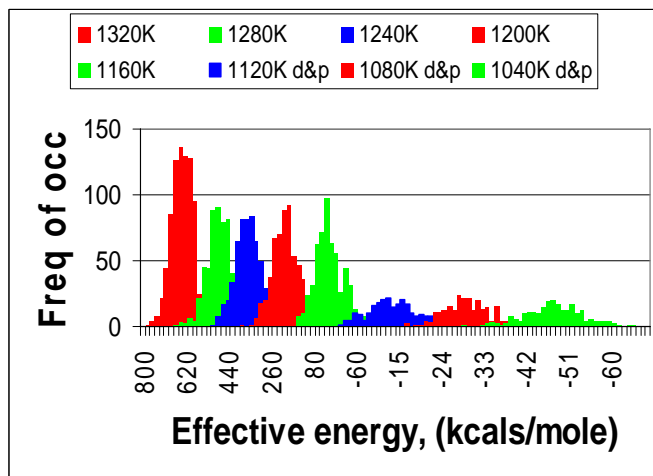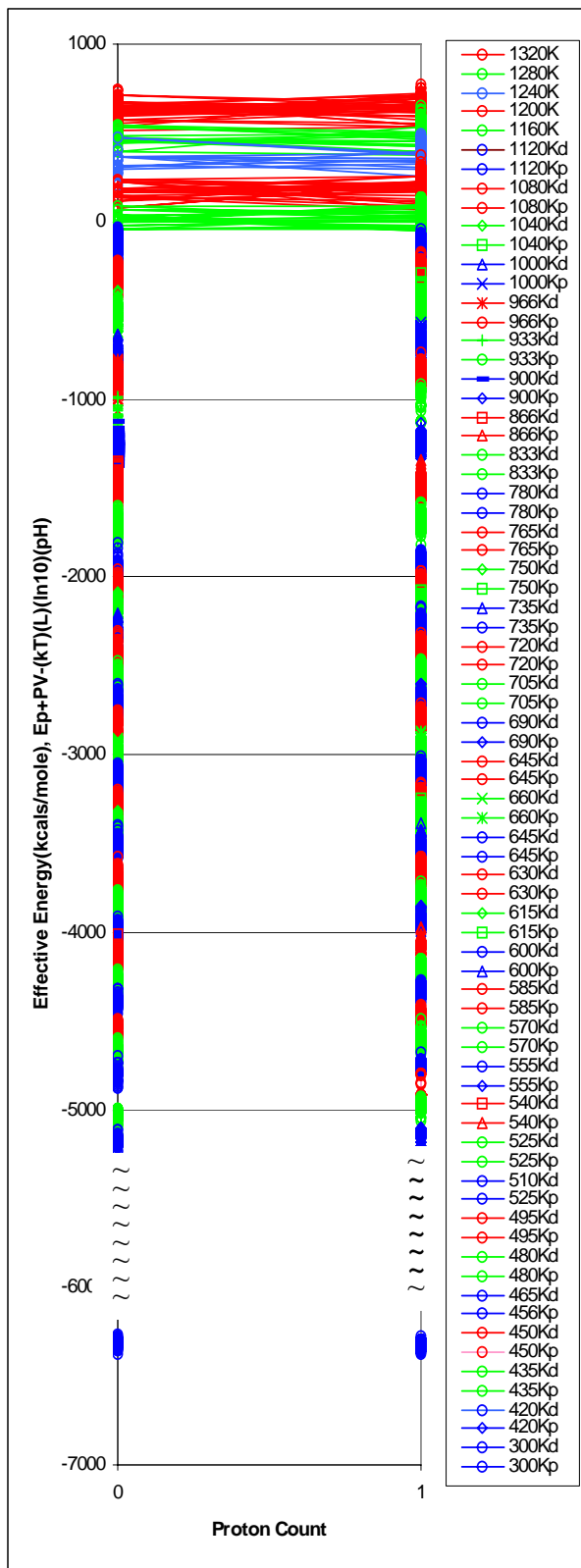
Figure 31: Simulated annealing ensemble, 1320K-300K

Table 4: P-T path, 1320K-300K

System#	Temp,K	Press,atm
1	1320	1700
2	1280	1650
3	1240	1600
4	1200	1550
5	1160	1500
6	1120	1450
7	1080	1400
8	1040	1350
9	1000	1300
10	966	1250
11	933	1200
12	900	1150
13	866	1100
14	833	1050
15	800	1000
16	780	950
17	765	900
18	750	900
19	735	900
20	720	850
21	705	850
22	690	800
23	675	800
24	660	750
25	645	750
26	630	700
27	615	700
28	600	700
29	585	650
30	570	600
31	555	600
32	540	550
33	525	550
34	510	500
35	495	450
36	480	450
37	465	400
38	450	400
39	435	350
40	420	300
41	405	300
40	390	250
43	375	200
44	360	200
45	345	150
46	330	100
47	320	100
48	310	100
49	300	1

The table above (Table 4) shows the temperature and pressure   specifics of the 49 Cysteine systems. Trajectories are generated for these 49 systems using our MD/MC algorithm, and the energies of their snapshots are plotted in Figure 31.   A proton count of 0 represents the deprotonated state, and a proton count of 1 represents the three protonated states. The left hand plot of Figure 31 shows that at higher temperatures the system readily samples the ionization states, represented by the colored high temperature lines near the top of the plot that go from one ionization state and back. In the lower temperature region, there are no such transitions. One TP link is therefore represented as a pair of equal length simulations, one completely protonated and one completely deprotonated (see section 4.5.2 for more detail on why we do this). For each simulation, the pressure corresponding to the temperature is shown in Table 4 page 167, but is not shown in Figure 31. As temperatures drop, the simulations must run longer for ionization state transitions to occur. We would therefore expect to see a gradual reduction in the number of transitions as the temperature drops. The reason why there appears to show a sudden cut off in transitions below 1120K is because the lower temperature simulations are not allowed to transition. Recall that in section 4.5.2 we have found that it is more efficient to generate high temperature "bridging" trajectories, and use WHAM[63,64] to combine them with short pairs of low temperature simulations that have no transitions. This way is much more computationally efficient than running low temperature simulations for many hundreds of nanoseconds to get statistically sufficient transitions (at temperatures near 300K, in section 4.3.3 we used the Eyring equation to estimate the transition period to be six hundred nanoseconds). The high temperature simulations altogether contain about 2500 transitions and represent a total of about 20 nanoseconds of MD/MC simulation. The lower temperature simulations are relatively short 20 picosecond simulations.

The right hand plot in Figure 31 shows the histograms for the five high temperature simulations plus three low temperature simulations (the x-axis) in terms of histogram frequency (y-axis). Here the histogram overlaps can be seen more clearly. The five HT simulations have larger histograms because these simulations are longer.

At the core of our WHAM algorithm is code that determines the relative free energy of the m th simulation, g_m, and the relative weight of the t th snapshot of the k th simulation, z_{kt}. This is done by iterating the following equations:

$$e^{-g_m} = \sum_{k=1}^{R} \sum_{t=1}^{n_k} \frac{e^{-\beta(\Lambda_m.U_{k,t} + P_m V_{k,t} + \mu_m L_{k,t})}}{z_{k,t}} \qquad z_{k,t} = \sum_{r=1}^{R} n_r e^{[g_r - \beta_r(\Lambda_r.U_{k,t} + P_r V_{k,t} + \mu_r L_{k,t})]}$$

The set of g_m's and $z_{k,t}$'s for our trajectories is a convenient form of the density-of-states. From this we can determine enthalpy, entropy, heat capacity etc at *ANY* temperature, pressure and pH i.e. $H_{est}(\beta, P, \Lambda, pH)$, $G_{est}(\beta, P, \Lambda, pH)$, etc.

There are statistical limitations that restrain our ability to accurately calculate these thermodynamic quantities at *ANY* set of state variables. For accurate results, we must have WHAM histogram energy overlap. In other words, we cannot, with any accuracy, calculate the pKa for Cysteine at 300K using only data generated at 1320K. There must be energy overlap. This is why the Figure 31 plot shows the energies of runs at different temperatures overlapping each other. Good statistics (lots of data in the "bridging" region) and histogram energy overlaps is the key to using WHAM capabilities effectively.

A glance at Figure 31 shows that we are making a lot of "relays" to go from 1320K to 300K. One of the strengths of WHAM is the ability to determine if we have sufficient statistics for our results, which is especially important for our calculations, which involve so many overlapping relays.
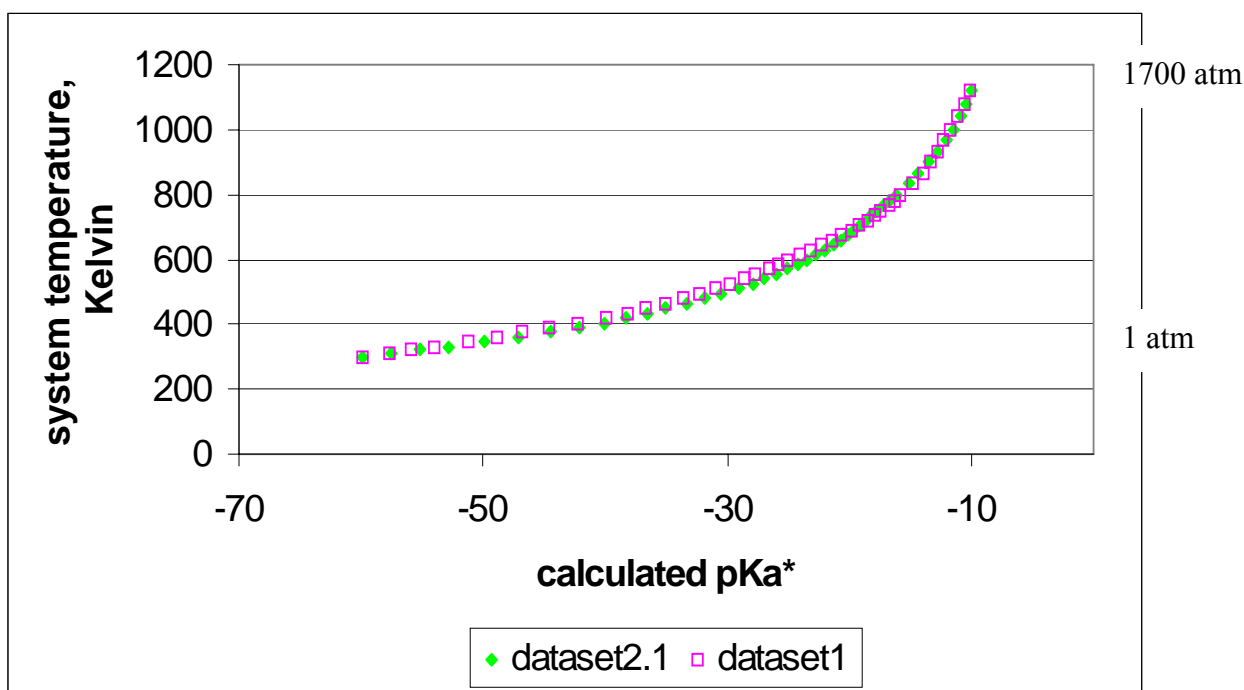

Figure 32: Calculated pKa* for Cysteine for a range of temperatures

In the plot above, WHAM is used to calculate pKa^*_{CYS} using the information from the simulations shown in Figure 31. Note that the pressure is reduced as the temperature is reduced. The pKa range is so low because we have not yet taken the calculated Bond Dissociation Energy number (BDE^{calc}_{cys}) into account (this part of the exercise is used to determine that number). Figure 32 above shows that the BDE^{calc} for Cysteine ($-(pKa^*_{cys} - pKa^{exp}_{cys})$) is about $-(-60 - 8.3) = 68.3$ pH units @ 300K. This is very encouraging, considering the BDE^{calc}_{cys} and the $BDE^{exp}_{thio-methane}$ comparison (discussed in section 5.2.1). Figure 30 shows two plots, the purpose of which is to simply to show the reproducibility of the result. These two plots also reveal relatively large precision errors, which brings us to next section in which we pursue much higher precision. With good histogram overlap and good statistics we aim to calculate BDE^{calc}'s to within 0.05 pH units. Our justification for pursuing this precision and our pursuit of this precision will be discussed in the next section, 5.2.1. The reader is reminded that the above work, i.e. calculating BDE^{calc} numbers for every *type* of titratable amino acid, only needs to be done *once* for a force field.

5.2.1   Accuracy and Precision

Our method, integration of Molecular Dynamics, Monte Carlo protonation state selection and Weighted Histograms, promises full atomic detail down to the solute proton dynamics level with computational feasibility. However, even detailed models and rigorous methods have errors. So for all such calculations, the errors need to be identified and quantified. These errors come from two sources.

> 1)   Systematic errors due to errors or approximations in the force field.
> 2)   Statistical errors due to counting statistics.

5.2.1.1 Systematic Errors due to Force Field or Methods: Accuracy
Accuracy errors come about as a result of systematic errors that are introduced by the force field or our methods. The results in this dissertation were based on the AMBER ff03 force field. Although the details of the code are tied to the Amber ff, the basic algorithm is not. It can be

used with any force field. We used the Amber ff03 models because our lab has a lot of experience with Amber. This force field belongs to a genre of force fields that are state of the art. Force fields like these are used to model a wide range of biomolecular systems, which is a testament to the accuracy and the capability of a state of the art classical mechanical model. In terms of minimizing accuracy errors by choosing one from the best genre of force fields, we simply cannot do better than this class of modeling. Developing our own force field is an intractable amount of work, and using a quantum chemistry model on proteins is an intractable amount of computation. One of the ways we can quantify accuracy errors is to compare calculated and experimental Bond Dissociation Energies, and this is shown and discussed in the following sections.

5.2.1.2 Statistical Errors due to Counting Statistics: Precision

Precision errors come in as a result of statistical errors and the central assumption of statistical errors is counting statistics. In section 3.7 of the WHAM theory, we see that WHAM makes transparent the connection between histogram count errors and thermodynamic result errors for a single histogram. However we are using many histograms in our *pKa* calculations and it is very hard to analytically calculate the error propagation when there is convolution of the counts of many histograms. However it is not hard to do it numerically, which is what follows in the next sections. What needs emphasis here is that WHAM and the related numerical analysis, tells us (a) where to most efficiently add simulation to yield the greatest increase in precision and (b) the computational cost to achieve a predetermined level of precision. This allows us to put a price tag on a given level of precision

### 5.2.2 Precision Pursuit: $0.05pH$ unit BDE^{calc} target precision

The applicability of these precision discussions is in no way limited to the single site Cysteine system, nor to *pKa* calculations. However the discussion about precision is made much less abstract if we use an example, such as the Cysteine single site *pKa* calculation. The following sections will explain why precise numbers are so important, and why our targeted precision is 0.05 pH units.

Routine experimental measurements of *pKas* in biological laboratories are performed with a precision of about 0.05 pH units. However, if we calculate *pKas* to a precision of 0.2 *pH* units that is more than good enough to see correlations between experimental and calculated *pKas*, and therefore this precision will be more than able to validate our method against experimental *pKas*. Recall from section 4.1.1 and 4.1.2 that the BDE^{calc} numbers will go into the *pKa* calculation, especially if there is interaction with other titratable sites, which is often the case. As a result, the errors in the BDE^{calc} numbers may have an additive effect on the *pKa* error. So whatever our target *pKa* precision is, the BDE^{calc} precision should be approximately one order of magnitude better. Hence our precision of 0.05 *pH* units for the BDE^{calc} numbers.

5.2.3   Precision Pursuit: Quantity of data & precision correlation

Our target precision for the BDE_{cys}^{calc} is 0.05 *pH* units. But the BDE_{cys}^{calc} is determined by calculating pKa_{cys}^{*} for the single site Cysteine system since $BDE_{cys}^{calc} = -(pKa_{cys}^{*} - pKa_{cys}^{exp})$. So our targeting of 0.05 *pH* unit BDE_{cys}^{calc} precision implies we are targeting 0.05 *pH* unit precision for pKa_{cys}^{*}. Having noted that BDE_{cys}^{calc} and pKa_{cys}^{*} precision are synonymous, using pKa_{cys}^{*} precision language in the following sections should not cause any confusion.

Here we will look at some specific calculations and show the correlation between the quantity of data and the calculated pKa precision. Recall that the calculated *pKa* for single site Cysteine system, pKa_{cys}^{*}, involves a data set that consists of a high temperature part, and a low temperature part. The high temperature part consists of long trajectories with many ionization state transitions, and the low temperature part consists of short trajectories that are locked into their ionization states (see Figure 31: Simulated annealing ensemble, 1320K-300K). We can represent the whole dataset with the following box diagram.
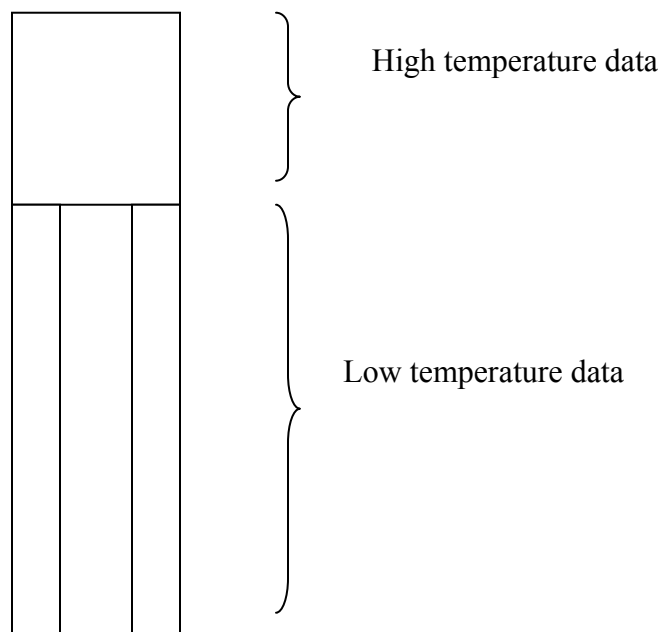
In the following sections, we will discuss how the *pKa* precision is affected by both the high temperature and low temperature data volume.

For this analysis, we use a new type of dataset, one that extends to much higher temperatures (2200K as opposed the 1320K as in Figure 31) and is much "lighter" (has simulations an order of magnitude shorter). The reason we changed our simulation ensemble design was because the old design (Figure 31) gave us a poor precision return for our computational investment. Figure 32 shows that the two datasets yield *pKa@*300K results that differ by over one *pH* unit, and the total simulation length of the dataset is about 22 nanoseconds. The increased temperature gives us more transitions, significantly reducing statistical errors due to too few ionization state transitions. Instead of several high-temperature (HT) simulations at different temperatures, all of the HT simulations have only one temperature, 2200K. For the following error sensitivity analysis, we consider only 4 low temperature (LT) links.
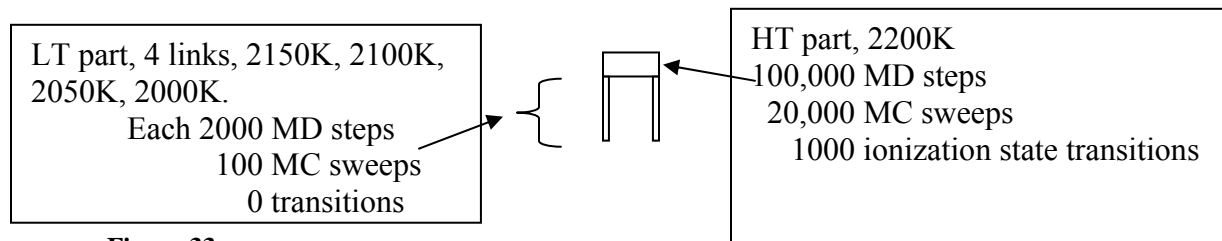
Figure 33

We generated 64 of the above dataset units. Then we proceeded to find out the most efficient way to add simulation volume for the purpose of increasing precision. Do we need longer HT simulations or LT simulations to most efficiently improve precision?
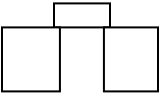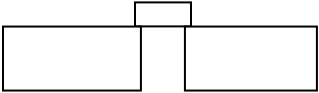
5.2.3.1 High Temperature Snapshot Volume & precision correlation

In this section we discuss how we keep the low temperature data volume fixed (though **different** LT datasets are used) and see how the *pKa* precision is affected as the high temperature data volume increases.

174
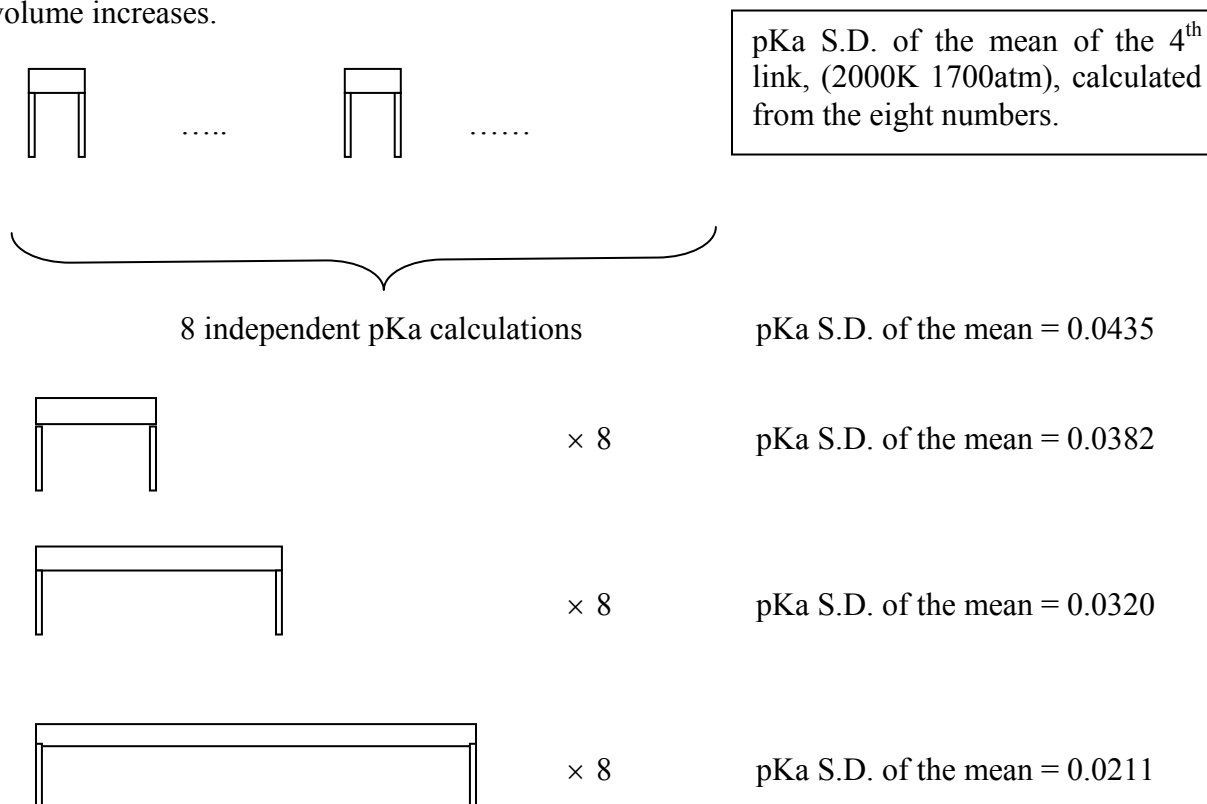
In the scheme above, we increase the HT data volume by factors of 2, 4, and 8. We do see a corresponding precision improvement for the $pKa@2000K$. Then we do a similar set of calculations, except we increase the LT data volume.

5.2.3.2   Low temperature snapshot volume & precision correlation

	× 8	pKa S.D. of the mean = 0.0435
	× 8	pKa S.D. of the mean = 0.0504
	× 8	pKa S.D. of the mean = 0.0457
	× 8	pKa S.D. of the mean = 0.0476

The S.D. of the mean does not improve with the LT volume increase! We may therefore conclude that for the 4th link pKa calculation (2000K), the number of transitions is dominating the counting statistics, so precision is most efficiently improved by adding HT simulations. We need to continue this type of analysis all the way down to 300K (about 240 links). We see that for only 4 links (2000K) the HT volume matters the most. But as we move further away from the HT bridge, we may see that the LT volume also matters.

5.2.4   *pKa* Error propagation down through the Histogram links

We would expect that the further away from the HT bridge, the larger the error in the *pKa* as we move down the links. We measured the S.D. of the mean for the results three "complete" datasets, that is, we went all the way down to 300K for three separate data sets.

× 3
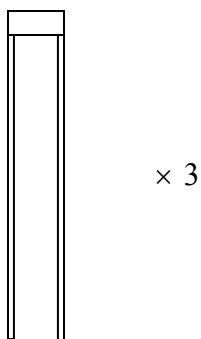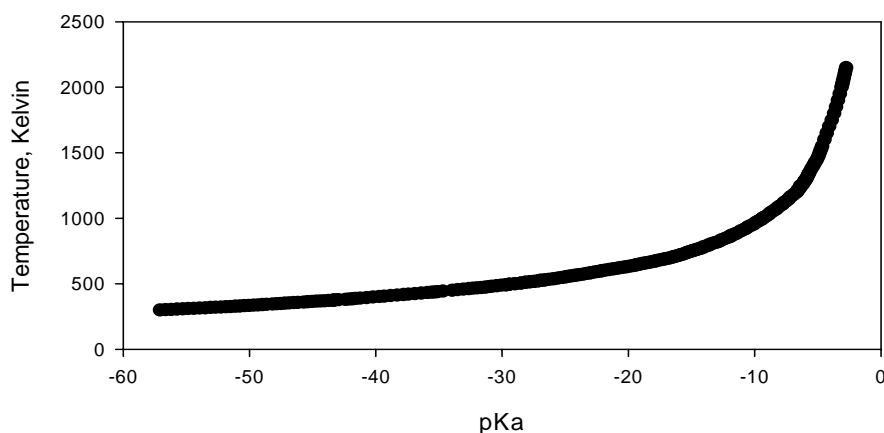
Temperature vs pKa

Figure 34: Temperature vs. pKa for 2200K-300K dataset

The temperature vs. *pKa* plot above plots the average calculated *pKas*. That is, the *pKa* of each link is averaged from the three numbers, where each number for each link is calculated from each of the three datasets. The *pKa* @ 300K, 1atm, averaged over three numbers, is –57.05 *pH* units with a S.D. of the mean of 0.6 *pH* units. This result is consistent with our preliminary result shown in Figure 32 page 169. The precision of this result is much better than that of the preliminary result (0.6 compared to ~2 pH units), despite the simulations of the new protocol being an **order of magnitude _less_** than those of the preliminary calculation. This S.D. of the mean improvement may be because of the larger number of High Temperature ionization state transitions (recall the new protocol has a highest temperature of 2200K, as opposed to 1320K), or better histogram link overlap of the new protocol. A note of caution about the 57.05 *pKa*@300K mean and the 0.6 *pKa* S.D. of the mean calculations: we have only calculated this

based on three numbers. The full set of 8 calculations needs to be completed for us to have more confidence in these mean and S.D. of the mean calculations. A note of optimism: the 57.05 $pKa^*_{CYS}@300K$ mean, when figured into the BDE^{calc}_{cys}, puts us within 2% of the $H-SCH_3$ BDE^{exp} $(88.6 \pm 1 kcal/mole)$.

We expect the precision to deteriorate as we go down the links from the High Temperature bridge. We have plotted both the "S.D. of the mean vs. the Link number" and the "S.D. of the mean vs. the Temperature" in the plots below. This gives us an idea of the precision deterioration trend.

S.D of the Mean vs Links
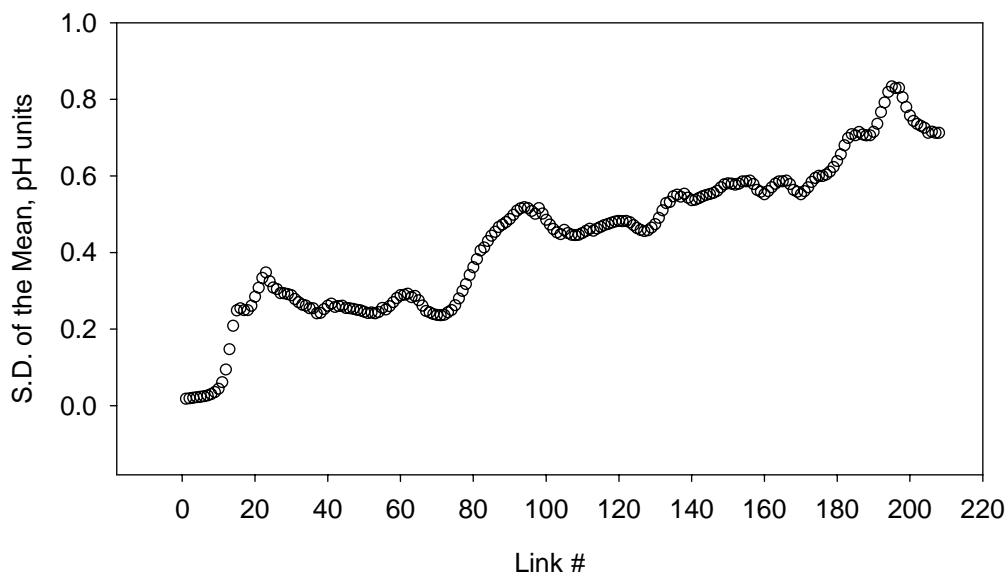
Figure 35: pKa S.D. of the mean vs. number of links

S.D. of the Mean vs Temp
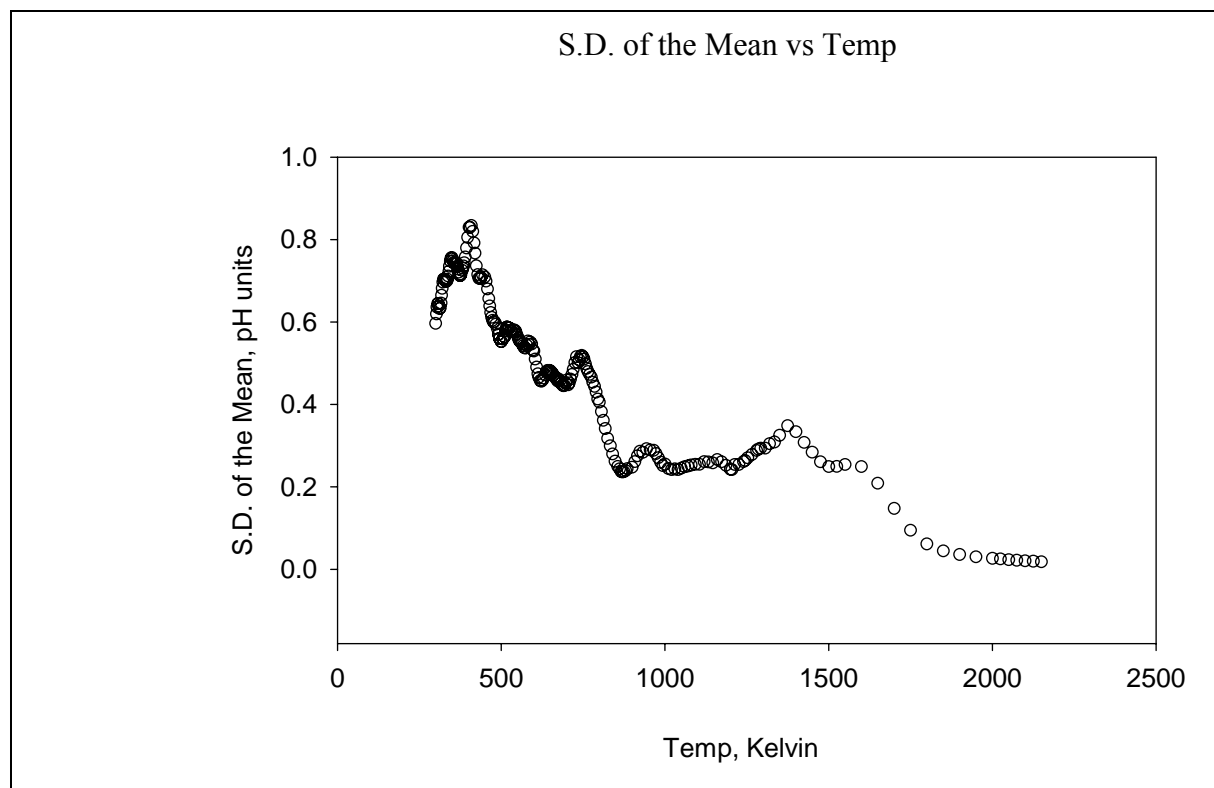
Figure 36: pKa S.D. of the mean vs. temperature

We believe that the plots are not smooth because we are only looking at three datasets, so the S.D. of the mean at each link is only based on three numbers. Therefore one of the future steps is to continue this analysis for the full eight datasets, which we reasonably expect will result in smoother plots. We also need to continue the sensitivity analysis as laid out on in the previous section in order to assess the most efficient way of hitting our precision target at 300K.

5.2.5 pKa Precision @ 300K Summary, Conclusion, and Future work

The purpose of the previous sections were to pin down where additional simulations were necessary to improve the precision of the $pKa@300K$. In the last section, we saw based on the precision plots of Figure 35 and Figure 36, the S.D. of the mean for the pKa is in the range 0.6-0.8. We need to complete this calculation for the full 8 datasets. Then we need to increase the high temperature volume and the low temperature volume in turn to

178

×8

see how the $pKa@300K$ precision is influenced. Hopefully, this will give us a data volume distribution protocol that, for eight of such datasets, will give us our target precision 0.05 pH units.

Figure 37: Smallest dataset

Now, let's consider the worst-case scenario. To hit our target precision (0.05 pH units) we need precision improvement by an order of magnitude. Suppose that, based on our 64 HT and LT datasets, we cannot find a data volume distribution protocol that reduces the S.D. of the mean by the required order of magnitude for 8 datasets. The worst-case scenario is that we can generate 256 times the number of datasets we used in Figure 34 (instead of only 64). Such a number of independent datasets will cause the pKa S.D. of the mean to drop by a factor of $\sqrt{256} = 16$, which will take the S.D. of the mean from its present 0.8 to the target 0.05.

In Figure 34 we use 3 of these "skinny" datasets, so 256 times as many is $256 \times 3 = 768$ of these datasets. Doing 768 such dataset generations and WHAM calculations is not so difficult. Recall each dataset of our new protocol is very lightweight, and can be generated in parallel. Each HT unit takes 1 processor-hour to generate, so generation of 768 of them would require 768 processor-hours. The thin LT unit pairs take about 30 **seconds** to generate, or 1/15 processor-hours. Doing all 244 links (from 2150K-300K) for 768 datasets will require 12493 processor-hours. The WHAM convergence for each of the datasets takes about 8 hours on 8 processors (64sus) and can be done simultaneously so 768 such convergences will take about 49,152 processor-hours. The initial equilibration of the 244LT +1HT simulations requires about 9764 processor-hours. This gives us a grand total of 96753 processor-hours. As mentioned before, this is the worse case scenario and we hope to find a data volume distribution scheme that allows us to hit our precision target with only a few dozen or so datasets.

One of the things that will be done in the near future is to quantize how sensitive the results are to choices of input parameters. We have already done some preliminary investigation into this and some of it is alluded to throughout the dissertation, but here I will summarize them.

179

The types of input parameter sensitivity tests first conducted were alluded to in section 4.3.5.2 page 151. We were rapidly driving the system from one ionization-state to the other by changing the *pH*. What we found was a large *pH* hysteresis in going from one state to the next (see Figure 25 on page 148). In an attempt to reduce the amplitude of the hysteresis, we experimented with changing some to the partial charges on the atoms of Cysteine. What we mentioned in section 4.3.5.2 was that the hysteresis amplitude did not change significantly enough for our purposes, but we did not mention that the hysteresis amplitude window did shift, in some cases by several *pH* units (several *kcals/mole*). At higher temperatures these shifts were seen more clearly and we were able to better quantize them. Consider the high temperature titration curve shown in Figure 30 page 164. This shows Cysteine titrating at a *pH* of −7.75 at a temperature of 1320K. We then dumped an additional charge of +0.5e (see "Cysteine charges" on page 141) on the nitrogen of the Cysteine backbone (as far away from the titration region) and the titration curve shifted 0.5 *pH* units in the negative direction, which is the expected direction. At 1320K, that represents a shift of about 3 *kcals/mole*. Therefore one of the near future calculations will be to continue the calculation all the way down to 300K and see the magnitude of the shift at 300K.

Another type of input parameter sensitivity test was alluded to in section 6.3.1.2 page 184. There we looked at how the titration curve was affected by different ratios of MD/MC steps. What we found was that the quality (uncertainty) improved with lower MD/MC ratios, but the titration curve itself did not shift in either direction.

5.3 BDE_{cys}^{calc} SUMMARY OF RESULTS

Some of the following BDE_{cys}^{calc} summarized results are extrapolated according the discussions of the previous section 5.2.5.

$$BDE_{thio-methane}^{exp} = 88.6 \pm 1 \, kcal/mole \,^{80}$$

$$BDE_{cys}^{calc} = -(pKa_{cys}^{*} - pKa_{cys}^{exp}) = -(-57.05 - 8.3)pH\text{units} @ 300K$$

$$= 90.3 \, kcals/mole$$

So $BDE_{thio-methane}^{exp}$ and BDE_{cys}^{calc} agree to within 3% of each other. Since Thio-methane and Cysteine are not identical, the *experimental* Bond Dissociation Energies for removing the proton from the sulphur may differ by about 5%, as determined from doing a survey that compares the Bond Dissociation Energies within pairs of very similar molecules. So the experimental error plus the uncertainty due to $Thio-methane \neq Cysteine$ is about 5.5 *kcals/mole* (6%), and our BDE_{cys}^{calc} is within that range. This very nice result may be fortuitous. The only way to know is to calculate *BDEs* for other titratable amino-acids and compare with experimental numbers. We did not set out to measure BDE_{cys}^{calc}. We set out to measure *pKa* shifts, so this is a very encouraging result.

The following table details the computational cost for various BDE_{cys}^{calc} precisions.

Desired BDE_{cys}^{calc} precision pH units@300K	# of small (Figure 37) datasets required	Processor-hours to Equilibrate of 245 systems	Processor-hours to generate HT datasets	Processor-hours to generate LT datasets	Processor-hours for WHAM convergence	Total processor-hours
0.05	768	9764	768	12493	49152	96753
0.10	192	9764	192	3124	12288	23368
0.20	48	9764	48	781	3072	13665
0.50	8	9764	8	125	492	10389
1.00	2	9764	2	32	123	2598

Table 5: Precision Cost Table

6.0 MC/MD ALGORITHM PERFORMANCE RESULTS

6.1   SINGLE NODE PERFORMANCE OF MC/MD ALGORITHM

Our MD/MC algorithm is an extensive modification of the Amber7-sander algorithm, and calls the same energy routines. The single node performance of our MD/MC code is as follows:

1)   One MC micro step is worth four MD steps.

2)   Execution time goes as $20+4N$, where N is the number of titratable sites.

So, for single titratable site simulations such as those mentioned in this paper, if one MC sweep occurs every 20 MD steps, our MD/MC algorithm is 20% slower than sander7. For a protein with 20 sites selected for titration, our MD/MC algorithm will run 500% slower than sander7.

6.2   POTENTIAL SINGLE NODE IMPROVEMENTS

Using an execution protocol of 1 MC sweep per 20 MD steps, a single site system is 20% slower and a 20-site system is 500% slower than the original sander7 code. The reason for this is that one MC sweep costs the same amount of time as 4 MD steps. The high cost of the MC sweep is because Cysteine has four microstates, and the same sander force/energy routines are called four times to calculate the energy of the whole system with the four different microstates. Theoretically, there is a much more efficient way to do it. If the energy of the system was broken up into components, such that some components of the system energy were invariant with microstate changes of the titratable site, and the other components of the energy were affected by microstate changes of the titratable site, then calculating the system energy for the four different microstates would only require four-fold recalculation of the components that change their energy. This would make a MC sweep cost only about 110% of an MD step.

However, it would require a big commitment to overhaul the core energy routines of sander. These routines are very dependable, very trusted, and have evolved across the span of time, Fortran versions and authors. It may make more sense to build our own energy routines from the ground up. Either way would require a serious commitment of time.

We have already committed a lot of time to writing tens of thousands of lines of code to get the MD/MC algorithm working. The best use of our resources at this time is to focus on "proof of concept". Our MD/MC code as it stands, 500% slower for a large 20-titratable site system, and scaleable to 64 processors, is more than good enough for "proof of concept". If or when our methods prove to be very useful to us or the community, and its use starts to become limited by compute power, then we or someone else could address the MC cost problem.

6.3   PARALLEL PERFORMANCE OF MC/MD ALGORITHM

Because our MDMC algorithm calls the same energy routines as the Amber7-sander algorithm on which it is based, it scales just like the Amber7-sander algorithm, which is 64 processors for large systems (>90,000atoms), and 16 processors for smaller systems (< 20,000atoms).

6.3.1   MD/MC trajectory generation improvements

Consider the data set represented in Figure 34 (data for the BDE^{calc} for Cysteine). There are 244 pairs of low temperature simulations, and one high temperature (2200K) simulation. The MD/MC algorithm has to generate all of the trajectories for several dozen or hundred such datasets, depending on the precision desired. Efficient generation of these trajectories is important for the feasibility of our approach, so what follows is a summary of the most important features of our code and execution methods that allow dozens of these datasets to be generated on a scale of hours.

One of the features of MD/MC algorithm is that one can control the number of sweeps in the MC sub-cycle, and the number of steps in the MD sub-cycle. One MC sweep costs about as

much as four MD steps. Reducing the number of useless MC sweeps or reducing the number of unnecessary force routine calls during the Monte Carlo sub-cycle is therefore one way of increasing execution efficiency.


6.3.1.1 One Monte Carlo sweep per sub-cycle

The choice of MD/MC input parameters for control of the number of MC sweeps per MC sub-cycle and MD steps per MD sub-cycle have to be chosen such that the trajectory is generated as efficiently as possible without significantly compromising the accuracy of the trajectory. We have found that executing more than one MC sweep per MC sub-cycle is a waste. The reason for this is that the effective energy differences between the protonization states are relatively large in the vast majority of instances. This means that the probability of the Monte Carlo algorithm choosing the lowest energy state is almost always very close to one, and the probability of the Monte Carlo algorithm choosing any other microstate is almost always very close to zero. This means that for the first step of the Monte Carlo sub-cycle, the microstate with the lowest energy is almost always chosen. For the second step of the Monte Carlo sub-cycle, the microstate energies stay the same because the system does not change configuration from the first MC step to the second. Therefore, in the second step of the Monte Carlo sub-cycle, the same microstate that was the lowest energy microstate in step one will again be the lowest energy microstate in the second step. So the same microstate will very likely be chosen again, and this will continue for all of the steps of the Monte Carlo sub-cycle. These additional steps do not add any information value to trajectory. For this reason we only use one Monte Carlo sweep per Monte Carlo sub-cycle.

6.3.1.2 Monte Carlo sweep: Molecular Dynamics step ratio, 1:20

During the Molecular Dynamics sub-cycle, the configuration changes. We just discussed how far apart the microstate energies usually are. However the effective energy relationship between the microstates changes dramatically with configuration changes. In other words a given configuration would have a certain effective microstate energy array, in which the microstates would have a certain order in terms of their effective energy values. However, within just a few femtosecond Molecular Dynamics steps the order of the microstates in terms of their effective energy values, can change completely. If computational efficiency was no issue, one Monte

184

Carlo sweep for every Molecular Dynamics step would give the maximum trajectory accuracy. However recall that we want to be able to execute trajectory generation as efficiently as possible without significantly impacting the trajectory accuracy. Only one molecular dynamics step between Monte Carlo sweeps is not efficient execution because we would have a situation very similar to what we previously discussed. Only one femtosecond of molecular dynamics evolution does not cause significant configuration change, therefore the relative effective energy microstate array (or the order of the micro-states with respect to their relative effective energies) does not change very much. This means that one Monte Carlo sweep after only one molecular dynamics step is too frequent. The question then becomes how to decide on how many molecular dynamics steps between Monte Carlo sweeps in order to effect significant enough configuration change. To decide this, we looked at the quality of the titration curves for different MC:MD ratio protocols such as the one shown below.
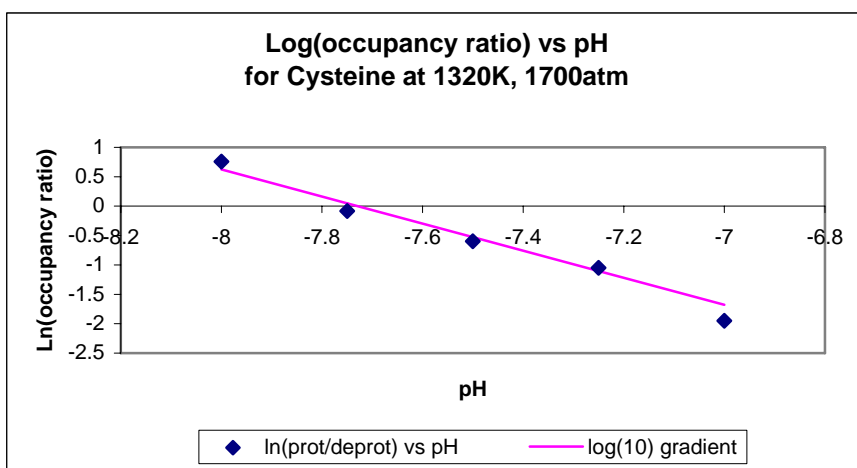
The above is a titration curve in the form of the natural log of the occupancy ratio versus pH. The gradient of this curve should be equal to –2.303. This theoretical gradient is represented by the pink line. (see section 6.2.6.2 for an explanation of the theoretical –2.303 gradient). There are several reasons for the data to deviate from the ideal. If the simulations were too short, there would be insufficient sampling of the system, which would be reflected as titration curve errors. Another reason for titration curve errors is if there is insufficient statistical sampling. This would happen if the simulations are long enough, but not enough Monte Carlo sweeps were performed during the course of the simulation. That is, if the number of molecular dynamics steps between Monte Carlo sweeps is too many. We have generated three sets of simulations for three such

185

plots, each plot representing a different MD/MC ratio. All the simulations are the same length. They are all 2 nanoseconds long, or 2 million molecular dynamics steps long, each step being one femtosecond long. Their being of equal length eliminates the differences in the titration curve errors being due to unequal molecular dynamics sampling of the energy landscape. These plots therefore allow comparison of how well each MD/MC ratio protocol does.

For 1:5 MC/MD, the Monte Carlo routine is called relatively often, only every five molecular dynamics steps. The simulation set generated by this protocol therefore consists of 2 million MD steps and 400,000 Monte Carlo sweeps. When plotted, the RMS pH deviation of the data for this protocol is 0.2 pH units.

For 1:20 MC/MD, the Monte Carlo routine is called every 20 molecular dynamics steps. This simulations generated by this protocol therefore consists of 100,000 Monte Carlo sweeps. When plotted, the RMS pH deviation for this data is only slightly worse, 0.23 pH units. This slight loss is statistical accuracy is well worth the 50% increase in execution speed!

For a 1:40 MC/MD protocol, the RMS pH deviation is 0.3 pH units. The execution speed improvement is only about 10% relative to the 1:20 protocol.

For the reasons just discussed we use the 1:20 MC/MD ratio as our standard execution protocol. Notice that all of the titration curve data discussed is for high temperature simulations. At lower temperatures, velocities would be lower, so that there would be less configurational changes per molecular dynamics steps. It is therefore expected that we would be able to get away with even more molecular dynamics steps between Monte Carlo sweeps at lower temperatures. The 1:20 ratio therefore serves as an upper limit for the MC/MD protocol ratio for simulations that range in temperature from 1320K and below. For simplicity sake, we use the same 1:20 MC/MD simulation protocol across all simulations (with one exception), even though the simulations may vary in temperature in order to simplify simulation protocols. The one exception is the highest 2200K bridging simulation (section Figure 33) for which we use a 1:5 MD/MC protocol because of the very high temperature.

6.3.1.3 Local Disk write

The trajectories for our datasets can be generated in an "embarrassingly parallel" fashion on hundreds of processors. The queues on most supercomputers favor jobs that are large (many processors) and short (less than three hours long). Through-put can therefore be greatly

improved if the submitted jobs request hundreds of processors and are shorter than three hours long. The trajectories for one data set are relatively short (1 processor-hour. for each 100 picosecond 2200K HT simulation). However dozens or hundreds of such datasets will be needed, depending on the target precision. Optimizing the execution speed is therefore important, and two ways of significantly increasing execution speed is to reduce the volume of output and to have all of the job output written to the processors local disks, since writing to the local disk is fastest.

There are two important output types of our jobs. One is the configuration information of the system, (that is the position of every atom in the system in Cartesian coordinates) which is updated with every molecular dynamics step and every Monte Carlo sweep. The second is the trajectory information, specifically the state variables (temperature, pressure and pH) and the configurational variables (protonation state, volume and potential energy). The output of the configuration information can be reduced to the point where it is an insignificant cost of computing time. This is because the only purpose of saving the configuration of the trajectory is for the purpose of restarting the trajectory, either in case of a system failure, or in case a longer trajectory is needed. As a result, the configuration information is made to be written about every half hour of computing time. This means that output of the configuration information represents no significant cost with respect to compute time. The trajectory information (the state variable and configuration variable information) on the other hand does represent somewhat of a significant cost with regards compute time. This is the information that is feed into our WHAM algorithm. There is therefore no circumventing the frequent output of this information at regular intervals. The trajectory information is made to be written for every Monte Carlo sweep (or every 20 molecular dynamics steps). This translates to approximately 2 kilobytes per second. We have found significant improvement in execution speed is achieved by taking advantage of fast output capabilities of the compute architecture.

Most supercomputing architectures consist of several disk storage areas for storing data. In approximate order from fastest to slowest, and least permanent to most permanent, are the local disks of the processors, the scratch or working disk, the home disk and the archive disk. The local disks of the processors provide the fastest input/output capability. However, they are extremely temporary, and information on these disks last only as long as the submitted job. It is therefore necessary to copy the data in the local disk off to a more secure disk before the job

ends. The job script code must be written to do this, but this is a small price to pay for the improvement in input/output speed, and overall speed improvement. Doing things this way, writing to the local disk, gives us almost a 50% speedup relative to writing to the scratch (working) directory.

7.0 WHAM ALGORITHM PERFORMANCE RESULTS

Our WHAM algorithm is also parallelized and scales with the size of the data set.

7.1   WHAM ALGORITHM PERFORMANCE EVOLUTION

To use our method to investigate any biomolecule, many trajectories must be generated at a wide range of temperatures and pH's, and each trajectory must be long (in the order of hundreds of picoseconds or nanoseconds). This is necessary in order to get proper sampling of the energy landscape, good histogram overlap and good statistics. This in turn yields good precision for our pKa or BDE^{calc} results and faster convergence of all calculated values.

After the MD/MC algorithm has generated these trajectories, our WHAM algorithm then has the task of analyzing all of the data to produce thermodynamic results. A typical dataset consists of trajectories totaling 100 nanoseconds. At a MD time step of 1 femtosecond and one Monte Carlo sweep every 20 MD steps, a 60 nanosecond dataset consists of 3 million snapshots; each with temperature, pressure, pH, potential energy, volume and protonation state information. To handle and iterate this volume of data to convergence, we have found it important to devote considerable resources towards structuring, parallelizing, refining and optimizing our WHAM algorithm to give us results in reasonable times. From our first WHAM code to the present version, there have been 24 major revisions of the code. Our final product is about 40 times faster than our early versions and can take the above data set mentioned, perform approximately 100 iterations and produce converged pKa numbers in approximately 10 minutes on twelve processors (1/6 hours x 12 processors = 2 service units). Below I will outline a few of the most important code improvements responsible for its computational efficiency.

7.1.1 Parallelization structure related improvements

Below are the three main sums that must be performed for each pKa iteration calculation (see section 3.9 and 3.10).

$$e^{-g_m} = \sum_{k=1}^{R} \sum_{t=1}^{n_k} \frac{e^{-\beta_m(\Lambda_m \bullet U_{k,t} + P_m V_{k,t} + L_{k,t}\mu_m)}}{z_{k,t}}$$

$$z_{k,t} = \sum_{r=1}^{R} n_r e^{[g_r - \beta_r(\Lambda_r \bullet U_{k,t} + P_r V_{k,t} + L_{k,t}\mu_m)]}$$

$$g_{est}\left(\underline{\xi}' \mid \beta, P, \underline{\Lambda}\right) = -\beta^{-1} \ln \left\{ \sum_{k=1}^{R} \sum_{t=1}^{n_k} \frac{\delta_{\underline{\xi}_{k,t}}^{\underline{\xi}'} e^{-\beta\left(\underline{\Lambda} \bullet \underline{U}_{k,t} + P V_{k,t}\right)}}{z_{k,t}} \right\}$$

7.1.1.1 Earlier versions

Previous versions of our WHAM code performed parallelization of the $\sum_{k=1}^{R} \sum_{t=1}^{n_k}$ sum by splitting up the $\sum_{k=1}^{R}$ sum among the processors. That is, for a data set of R simulations, each processor was responsible for a different subset of the R sums. The number of simulations assigned to each processor was approximately $\dfrac{R}{number\_of\_processors}$, where $number\_of\_processors$ is the number or processors assigned to the task. If N is the $number\_of\_processors - 1$, then the assignment per processor looks like below.

$$k = 1$$
$$k = 2$$
$$-$$

pe_0 or processor 0

$$-$$
$$-$$
$$-$$
$$-$$

pe_1

$$-$$
$$-$$
$$-$$
$$-$$
$$-$$
$$-$$

pe_N

$$k = R$$

The sum $\displaystyle\sum_{k=1}^{R}\sum_{t=1}^{n_k}$ is therefore broken up and executed as follows.

$$\sum_{k=1}^{R}\sum_{t=1}^{n_k} = \underbrace{\sum_{k=1}^{kend(pe_0)}\sum_{t=1}^{n_k}}_{pe_0} + \underbrace{\sum_{k=kstart(pe_1)}^{kend(pe_1)}\sum_{t=1}^{n_k}}_{pe_1} + \ldots\ldots \underbrace{\sum_{k=kstart(pe_N)}^{R}\sum_{t=1}^{n_k}}_{pe_N}$$

There were several limitations to this structure of parallelization.

1) R was the limit of the number of processors assigned to the job. If the data set consisted of 20 simulations, then 20 would be the maximum number of processors that could be effectively engaged in the calculation.

2) Even more restrictive were the pKa type calculations where only a subset R' of the R simulations $(R' < R)$ required pKa calculations performed. In typical calculations, we would need to iterate pKa values for 10 simulations, i.e. $R' = 10$ and $\displaystyle\sum_{k'=1}^{R'=10}$. In this case we will only be able to use a maximum of 10 processors.

3) This structure was also prone to load imbalance. If k_m was a long simulation with $n_{k_m}^{long}$ snapshots, and k_n was a short simulation with $n_{k_n}^{short}$ snapshots, with k_m and k_n assigned to different processors then there would be an

imbalance as follows.

$$\underbrace{\sum_{t=1}^{n_{k_m}^{long}}}_{pe_a} \qquad \underbrace{\sum_{t=1}^{n_{k_n}^{short}}}_{pe_b}$$

Computational time

of $pe_a \propto n_{k_m}^{long}$

Computational time

of $pe_b \propto n_{k_n}^{short}$
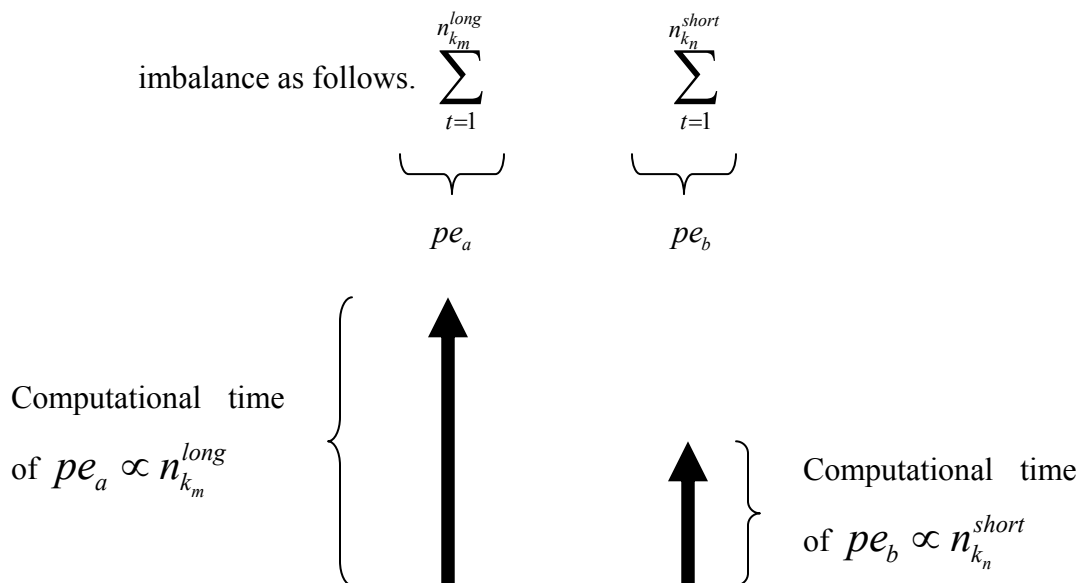
Processor b would hang, waiting for processor a.

7.1.1.2 Current version

The current version performs parallelization of the $\sum_{k=1}^{R}\sum_{t=1}^{n_k}$ sum by splitting up the

$\sum_{t=1}^{n_k}$ snapshot sum among the number of processors. So for a data set of R simulations, each

processor is responsible for summing a subset of the snapshots for every simulation. The number of snapshots of the kth simulation assigned to each processor is approximately

$\dfrac{n_k}{number\_of\_processors}$. If $N = number\_of\_processors - 1$, then

$k^{th} sim.$ ____ .. __ .. __ .. __ __ .. __ .. __ __ .. __ .. ___

$t=1 \quad t=2$ $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad t=n_k$
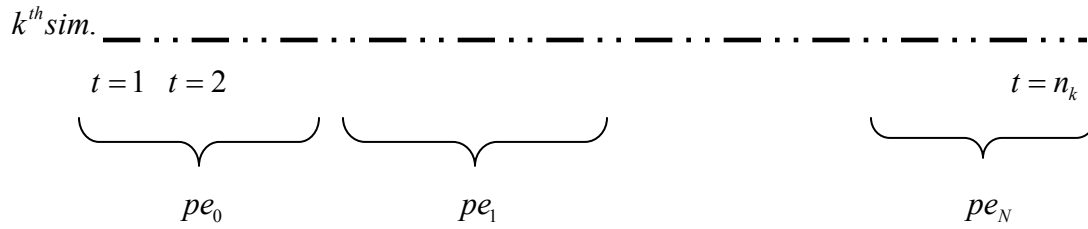
$pe_0 \qquad\qquad\qquad pe_1 \qquad\qquad\qquad\qquad\qquad\qquad pe_N$

This means that the $\displaystyle\sum_{k=1}^{R}\sum_{t=1}^{n_k}$ sum is broken up and executed as follows.

$$\sum_{k=1}^{R}\sum_{t=1}^{n_k}=\sum_{k=1}^{R}\left(\sum_{t=1}^{tend(pe_0)}+\sum_{t=tstart(pe_1)}^{tend(pe_1)}+......\sum_{t=tstart(pe_N)}^{n_k}\right)$$

$pe_0 \qquad\quad pe_1 \qquad\qquad pe_N$

The advantages of this structure are as follows

  3) There is no realistic limit on the number of processors that can be engaged in the sums and iterations, since n_k (number of snapshots in the kth simulation) is typically in the range of 10,000 to 500,000.

  4) For pKa type calculations, no restrictions on parallelization apply as did for the previous versions

  5) The load balancing is perfect. For two simulations m and n that vary widely in lengths, then

$$\sum_{k=1}^{R}\sum_{t=1}^{n_k}=\sum_{k=1}^{R}\left(\sum_{t=1}^{tend(pe_0)}+...\sum_{t=tstart(pe_a)}^{tend(pe_a)}+\sum_{t=tstart(pe_b)}^{tend(pe_b)}......\sum_{t=tstart(pe_N)}^{n_k}\right)$$

$pe_a \qquad\qquad pe_b$

Computational time of $pe_a \propto$ $\qquad\qquad\qquad$ Computational time of $pe_b \propto$

$$\dfrac{n_{k_n}^{short}+n_{k_m}^{long}}{num\_of\_procs} \qquad\Bigg\{\quad\Uparrow\quad\Uparrow\quad\Bigg\}\qquad \dfrac{n_{k_n}^{short}+n_{k_m}^{long}}{num\_of\_procs}$$

7.1.2 Communication reduction improvements

The less information that has to be broadcast between processors, the faster the speed of the algorithm. The largest array handled by the WHAM code is the $z_{k,t}$ array. A typical data set would have $R = 80$ simulations ($k = 1,80$) and each simulation may have in the order of $n_k = 50,000$ snapshots ($t = 1,50000$). The $z_{k,t}$ information is spread out among the processors as follows. Consider R simulations of various lengths and the simulations are ordered from longest to shortest (how the simulations are ordered is not important here).

$$
\begin{pmatrix}
z_{1,1} & z_{1,2} & \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & z_{1,n_1} \\
z_{2,1} & z_{2,2} & \cdots\cdots\cdots\cdots\cdots\cdots & z_{2,n_2} \\
\vdots & \vdots & \ddots & \\
z_{R,1} & z_{R,2} & \cdots\cdots\cdots & z_{R,n_R} \\
\end{pmatrix}
$$

$$
\underbrace{\qquad}_{pe_0} \quad \underbrace{\qquad}_{pe_1}
$$
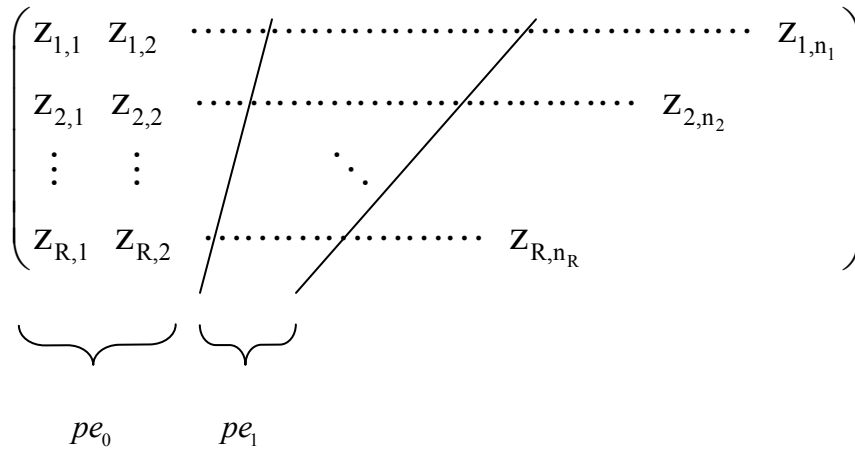
The $z_{k,t}$ information is spread out among the processors as such that each processor knows only a subset of the $z_{k,t}$ elements as shown above. Note that because the simulation lengths vary widely, the processor boundaries of the z matrix are neither straight nor smooth. Limiting the knowledge of each processor in this way helps a great deal with memory management.

When $\displaystyle\sum_{k=1}^{R}\sum_{t=1}^{n_k}\frac{numerator}{z_{k,t}}$ type sums are performed, communication load was improved as follows.

7.1.2.1 Early versions

Earlier versions conducted the second $\displaystyle\sum_{t=1}^{n_k}\frac{num}{z_{k,t}}$ sum by summing over all $t's$ of the kth

simulation. This required each processor to know all elements of the z matrix, which required a broadcast of millions of elements to all processors.

7.1.2.2 Current version

The current version conducts the sum where $\displaystyle\sum_{k=1}^{R}\sum_{t=1}^{n_k}\frac{num}{z_{k,t}} =$

$$\sum_{k=1}^{R}\sum_{t=1}^{tend(pe_0)}\frac{num}{z_{k,t}}+\cdots\sum_{k=1}^{R}\sum_{t=tstart(pe_a)}^{tend(pe_a)}\frac{num}{z_{k,t}} \quad + \quad \sum_{k=1}^{R}\sum_{t=tstart(pe_b)}^{tend(pe_b)}\frac{num}{k_{k,t}}\cdots\cdots\sum_{k=1}^{R}\sum_{t=tstart(pe_N)}^{n_k}\frac{num}{z_{k,t}}$$

$$pe_a \longleftrightarrow pe_b$$

Communication of
$N+1$ numbers between
all processors

Each processor sums only the terms containing the $z_{k,t}$ elements it is aware of. After these initial sums are performed, then the processors need to communicate only the $N+1$ subtotals between each other ($N+1$ is the total number of processors).

7.1.3 Execution methodology improvements

Figure 7 represents a typical set of data that WHAM may process for a BDE^{calc} calculation. I will present the main features of this data set using a simplified representation as shown.

Effective

Energy
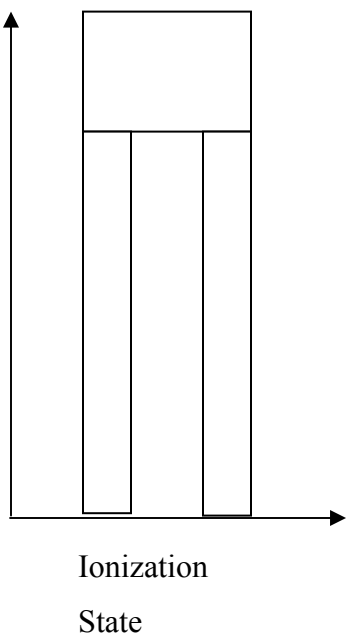
The top box represents the high temperature simulations shown at the top of Figure 7. Recall that for these high temperature simulations the trajectory will bounce between the protonated and deprotonated ionization states. The low temperature simulations occur in pairs and are represented as the legs. For each temperature and pressure there is a protonated and deprotonated simulation. These simulations stay in their ionization state so that there are no transitions. The pHs of these low temperature simulations are of no consequence because they are going to be "recalculated" (see section 5.3.5). Our WHAM algorithm iterates the z's, the g_m's and the low temperature pH's until there is convergence.

Ionization

State

$pH / pKa's$

$initial\ g_m's$

$g_m's$

$z_{k,t}'s$
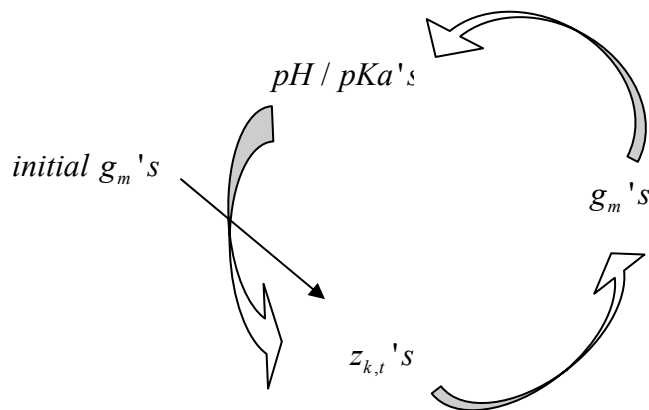
The high temperature part of this data set consists of about 60 trajectories, totaling about 60 nanoseconds of molecular dynamics steps and 3 million Monte Carlo sweeps (one Monte Carlo sweep for every 20 molecular dynamics steps). The trajectories of the high temperature data set

196

differ in temperature, pressure, and pH. The high-temperature temperature range is from 1320K to 1160K. The low temperature data set consists of 193 links, or 193 pairs of "locked" (protonated/deprotonated) simulations that range from 1160K to 300K,1 atm.

7.1.3.1 Earliest methods

In the earliest WHAM iteration methods, the calculation was carried out for one link at a time.

only one *pH*

initial g_m's

g_m's

$z_{k,t}$'s
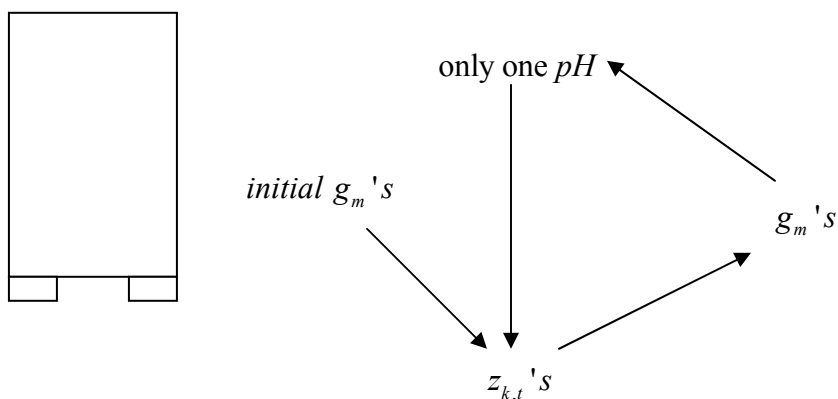
That is, the cycle was repeated until the pH for the one link pair converged.

Then a second calculation was performed for the second link pair. This time, the first link pair has a fixed pH that was previously calculated, and is treated in the same way as the other original high temperature simulations.

only one *pH* floats

initial g_m's

all g_m's float

all $z_{k,t}$'s float
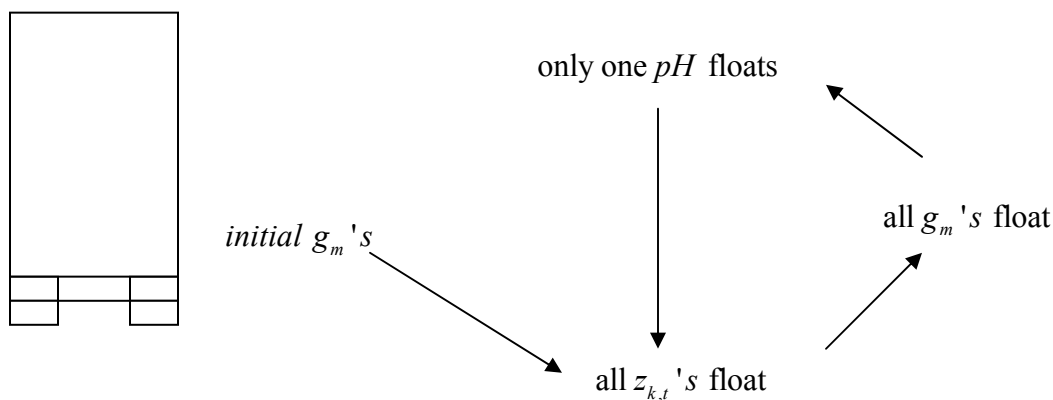
This process continues until one does all link pairs all the way down to 300K, 1 atm. The disadvantage of this method is as follows. Notice that in Figure 7, the effective energy histograms at each temperature are separated by approximately one standard deviation. This means that for the pKa/pH calculation at a given temperature, the information that influences that calculation comes from the trajectories at that temperature, but also the trajectories at the next
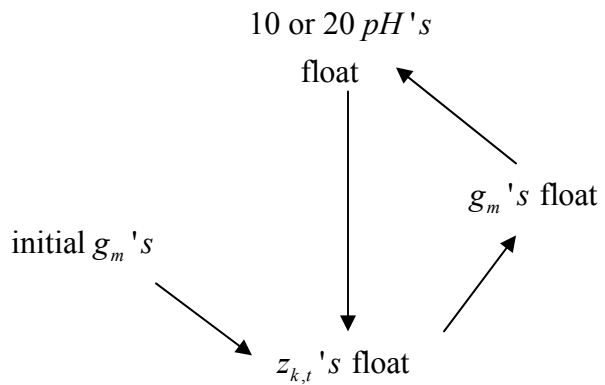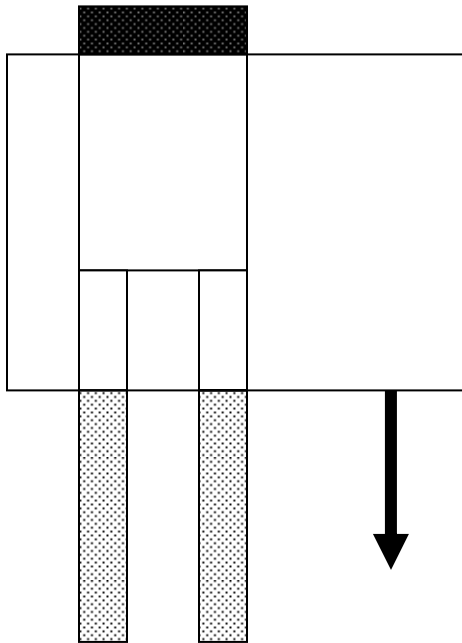
two highest temperatures and trajectories at the next two lowest temperatures. However, notice that in the way we have done things above, the lower temperature trajectories are excluded from the pKa calculation for any link. This leads us to a later version of execution methodology.

7.1.3.2 Later execution method

The next evolution in execution methodology was to allow all the low temperature region pH's to float. The advantage with this method was that all of the data went into every pH calculation. The drawback with this method was that the convergence was very slow, not because of the volume of data for WHAM to process (parallelization took care of that problem), but mostly because a large number of iterations were required for convergence. The number of iterations required for convergence goes up exponentially with the number of links or attempted pH calculations.

all low temp. $pH\text{'}s$ float

all $g_m\text{'}s$ float

initial $g_m\text{'}s$

all $z_{k,t}\text{'}s$ float
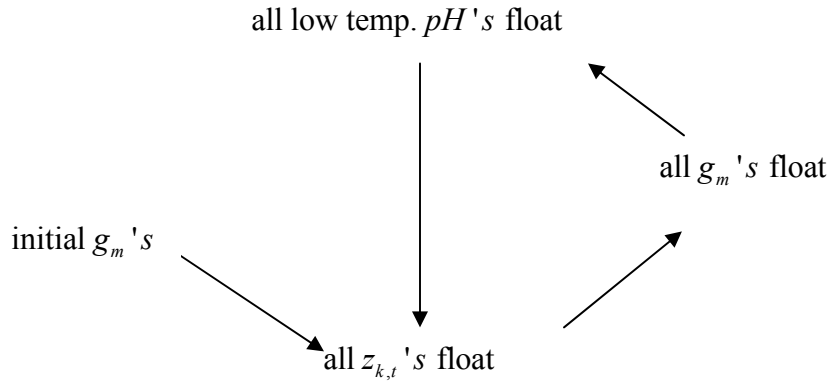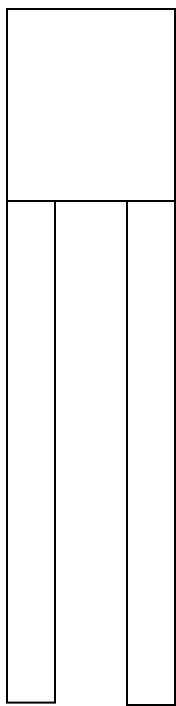
7.1.3.3 The Moving Window prototype method

10 or 20 $pH\text{'}s$
float

$g_m\text{'}s$ float

initial $g_m\text{'}s$

$z_{k,t}\text{'}s$ float

In this method, only the pHs for 10 or 20 links are calculated at a time, those that lie within the "window". Only the data that lies within the window goes into the calculations, and only the

pH's for the low temperature links that lie in the window are calculated. The calculation relating to one window is relatively quick, about 1/2 hour. For the data set we are considering, about 30 windows would be necessary to go all the way down to 300K, 1 atm. At time of writing this method was still in the prototype phase, so it is not clear if this method accelerates the convergence times.

7.2   COMPUTER RESOURCES AND PROVEN PLATFORMS

Our WHAM code (and our MD/MC code) runs on a range of platforms. Lemieux at the PSC and the IA-64 Linux cluster at NCSA were the main production resources, in that order. Our local resource was an AMD-mpich Beowulf cluster, which was used for a lot of the code development and debugging.   Proving our algorithm on machines with such different architectures and compilers helps the debugging process and gives us a high degree of confidence in its portability.

7.2.1   Lemieux at the PSC: Basic architecture

Processors:
64 bit processors, Compaq Alpha E45 processors, running at ~ 1GHz. Total of 3000 processors
Nodes:
Quad processors per node. 4 GB of memory per node. Total of 750 nodes.
Inter-node communication:
Quadrics interconnect, ~ $1.5\,\mu s$ latency.

Operating system:
Tru64 Unix, 64 bit enabled operating system
Compiler:
HP f90 compiler. Compiler options, level 5 optimization. Link mpi libraries (-lmpi). All other options remain at default settings (f90 -lmpi –O5 …)

7.2.2   NCSA Itanium/mpich cluster

Processors:

64-bit processors. Intel Itanium 2 processors, running at 1.3GHz – 1.5GHz.   Total of 1774 processors.

Nodes:

Dual processors per node. 4GB – 12GB of memory per node. Total of 887 nodes

Inter-node communication:

Myrinet interconnect, latency $\sim$ 2-3 μs

Operating system:

Linux RedHat

Compiler:

Intel f90 compiler.   Compiler options, link mpi (use mpif90) libraries and use level 5 optimization (mpif90 -O5)

7.2.3   Beowulf cluster

Processors:

32-bit processors. AMD Athlon processors, running at 1.3GHz – 1.5GHz. Total of 8 processors.

Nodes:

Single processors per node. 0.5GB – 1GB of memory per node. Total of 8 nodes

Inter-node communication:

Myrinet interconnect, latency $\sim$ 2-3 μs

Operating system:

Linux RedHat

Compiler:

Intel f90 compiler.   Compiler options, link mpi (use mpif90) libraries and use level 5 optimization (mpif90 -O5)

7.3   CONVERGENCE CRITERIA FOR THE FREE ENERGIES

Consider a full dataset, such as one of those discussed in the previous chapter. That is, a dataset containing a 2200K high temperature bridge and 244 low temperature links going all the way down to 300K. Convergence of this dataset requires tens of thousands of WHAM iterations running on eight processors for about twelve hours. The following sub-sections explore different convergence acceleration schemes.

7.3.1   Ferrenberg's accelerated convergence

In his PhD thesis, Ferrenberg[85] outlined a method for accelerating the convergence of the free energies. In what follows is our implementation of his scheme, closely following the same outline.

Recall that we must determine the set of free energy parameters $\{g_m\}$ self-consistently. This is accomplished by iterating the density-of-states expressions of section 3.9, which gives the result $e^{-g_m^{i+1}} = \sum_{k=1}^{R} \sum_{t=1}^{n_k} \dfrac{e^{-\beta_m(\Lambda_m U_{k,t}+P_m V_{k,t}+L_{k,t}\mu_r)}}{\sum_{r=1}^{R} n_r e^{[g_r^i-\beta_r(\Lambda_r U_{k,t}+P_r V_{k,t}+L_{k,t}\mu_r)]}}$ where i is the iteration index. A simple iteration of these equations converges slowly. The convergence can be accelerated by making use of the derivatives of the above equation. The derivative of g_m^{i+1} with respect to one of the free energies in the i-th iteration g_n^i can be calculated as follows.

$$\frac{\partial e^{-g_m^{i+1}}}{\partial g_n^i} = \sum_{k=1}^{R}\sum_{t=1}^{n_k} \frac{e^{-\beta_m(\Lambda_m U_{k,t}+P_m V_{k,t}+\mu_m L_{k,t})} \bullet n_n e^{[g_r^i-\beta_n(\Lambda_n U_{k,t}+P_n V_{k,t}+\mu_n L_{k,t})]}}{-(\sum_{r=1}^{R} n_r e^{[g_r^i-\beta_r(\Lambda_r U_{k,t}+P_r V_{k,t}+\mu_r L_{k,t})]^2})}$$

$$= n_n e^{g_n} \sum_{k=1}^{R}\sum_{t=1}^{n_k} \frac{e^{-\beta_n(\Lambda_n U_{k,t}+P_n V_{k,t}+\mu_n L_{k,t})} \bullet e^{-\beta_m(\Lambda_m U_{k,t}+P_m V_{k,t}+\mu_m L_{k,t})}}{-(\sum_{r=1}^{R} n_r e^{[g_r^i-\beta_r(\Lambda_r U_{k,t}+P_r V_{k,t}+\mu_r L_{k,t})]})^2}$$

$$= -e^{-g_m^{i+1}} \frac{\partial g_m^{i+1}}{\partial g_n^i}$$

$$\frac{\partial g_m^{i+1}}{\partial g_n^i} = n_n e^{g_m^{i+1}+g_n} \sum_{k=1}^{R}\sum_{t=1}^{n_k} \frac{e^{(-\beta_n\Lambda_n-\beta_m\Lambda_m)U_{k,t}+(-\beta_nP_n-\beta_mP_m)V_{k,t}+(-\beta_n\mu_n-\beta_m\mu_m)L_{k,t}}}{(\sum_{r=1}^{R} n_r e^{[g_r-\beta_r(\Lambda_rU_{k,t}+P_rV_{k,t}+\mu_rL_{k,t})]})^2}$$

Then, if $\{g_m^*\}$ are the desired fixed points of the iteration

$$g_m^{i+1}-g_m^* = \sum_n \frac{\partial g_m^{i+1}}{\partial g_n^i}(g_n^i-g_n^*) \quad\text{or}\quad g_m^{i+1}-\sum_n \frac{\partial g_m^{i+1}}{\partial g_n^i}g_n^i = \sum(\partial_{m,n}-\frac{\partial g_m^{i+1}}{\partial g_n^i})g_n^*$$

This is a linear set of equations that can be solved for $\{g_m^*\}$ which are then used as an improved solution, and the whole procedure can be repeated until convergence is achieved.

7.3.2   Why Ferrenberg's accelerated convergence is not feasible

The matrix form of the above linear set of equations is

$$B = AG \text{ or } \begin{pmatrix} g_1^{i+1}-\sum_n \frac{\partial g_1^{i+1}}{\partial g_n^i}g_n^i \\ \vdots \\ g_R^{i+1}-\sum_n \frac{\partial g_R^{i+1}}{\partial g_n^i}g_n^i \end{pmatrix} = \begin{pmatrix} \partial_{1,1}-\frac{\partial g_1^{i+1}}{\partial g_1^i} & \cdots & \partial_{1,R}-\frac{\partial g_1^{i+1}}{\partial g_R^i} \\ \vdots & \ddots & \vdots \\ \partial_{R,1}-\frac{\partial g_R^{i+1}}{\partial g_1^i} & \cdots & \partial_{R,R}-\frac{\partial g_R^{i+1}}{\partial g_R^i} \end{pmatrix}\begin{pmatrix} g_1^* \\ \vdots \\ \vdots \\ \vdots \\ g_R^* \end{pmatrix}$$

Ferrenberg compared convergence times for the free energies of 4 simulations, ($R=4$). For 4 simulations, matrix A takes the form of a 4×4 matrix and Ferrenberg found that when using straight iteration, the amount of computer time needed for convergence went up linearly with the number of decimal places desired for the free energies. But the accelerated algorithm converged much more rapidly. For free energies within 10^{-5} of their exact values, accelerated convergence got there one order of magnitude faster than straight convergence.

However our datasets contain in the order of 245 simulations ($R=245$). We implemented the above accelerated convergence scheme, solving of the above matrix system with a parallel BLAS lower-upper (LU) decomposition routine. The problem is that the number of elements of the A matrix goes as R^2. Even implementing optimizations, such as a parallel LU routine and carrying over relevant $\sum_{k=1}^{R}\sum_{t=1}^{n_k}$ sums that were already calculated from the e^{-g_m} and $z_{k,t}$ routines, for $R>15$ the gain in the reduced number of iterations was offset by the

significantly longer times for each iteration. We concluded that this accelerated convergence scheme was not suitable for us, unless we found a parallel LU decomposition routine that scaled much better and performed an order of magnitude faster than the one we tried. It is fully developed and embedded in the code for experimentation by others. But we commented them out, and implemented another accelerated convergence scheme.

7.3.3 Projected pKa accelerated convergence

Recall that the original WHAM iterative cycle is
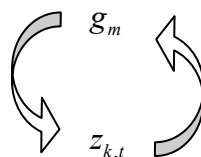
But in our pH iteration scheme, the iterative cycle becomes
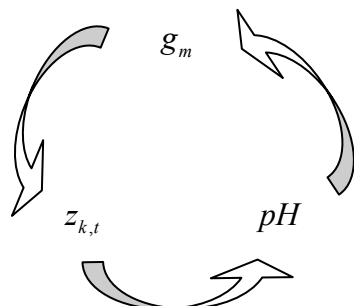
so that the set of $\{g_m\}$ and $\{pH\}$ are on the same footing in the sense that the convergence of the g_m's is synonymous with the convergence of the pH's. Since the calculated $\{pH\}$ and the $\{pH\}$ precision is what we are really after (not the $\{g_m\}$) it makes sense to place the check for convergence on the $\{pH\}$ instead. Another advantage to checking the convergence of the $\{pH\}$ is that the $pH_m^{i+1} - pH_m^i$ differences are much larger than the $g_m^{i+1} - g_m^i$ differences, so numerical evaluation of the convergence criteria is much easier. A pH precision of four decimal places requires g_m precision of six to eight decimal places.

The accelerated pH convergence scheme we use is as follows. For some pH of the $i-th$ iteration, pH_m^i, the pH of the $(i+1)-th$ iteration is determined by looking backwards at the previous 10 steps,
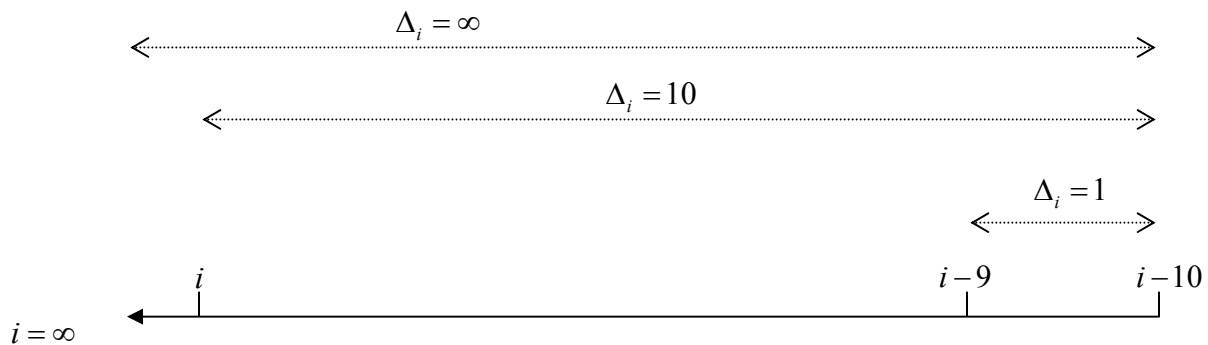
and then projecting forward to infinity for pH_m^{i+1}.
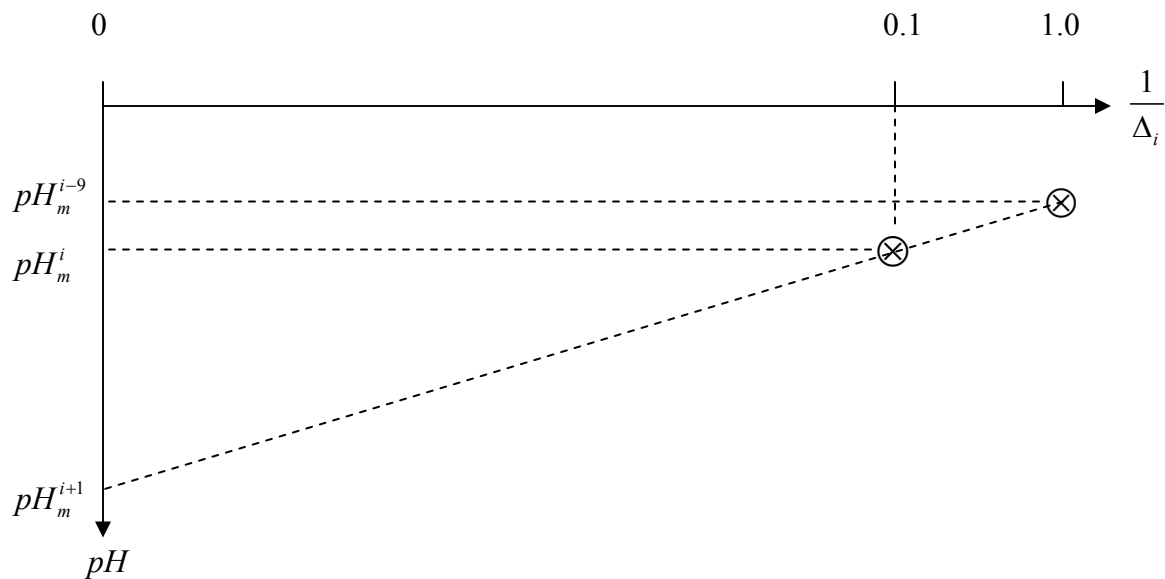

Figure 38: pH accelerated convergence

The whole process is repeated for all $\{ pH_m \}$ until convergence. Our pH convergence criterion is four decimal places. This scheme accelerates convergence by reducing the number of required iterations by a factor of about five.

7.4   POTENTIAL IMPROVEMENTS

For a full dataset of links from 2200K to 300K, (245 links), convergence to four decimal places of the *pH* values occurs slowly, even implementing the schemes outlined above. About 40,000 iterations are required. If a suitable number of processors are allocated such that one iteration is performed every second, then there will be convergence in about 11 hours. There is clearly a need for faster convergence.

More sophisticated convergence schemes should help. One idea would be to make the survey window variable instead of fixed. Instead of looking back 10 steps, then projecting forward to infinity, one could vary to number of steps looked backwards to. A window of 10 steps is good at the start of the iterations, but as convergence approached, the window could be reduced to 5 or 2 for more aggressive predictions.

8.0 FUTURE WORK, PROSPECTS AND FURTHER DISCUSSION

8.1   FINE TUNING OR VALIDATING FORCE FIELD PARAMETERS

This dissertation has only addressed the BDE^{calc} for Cysteine. One of the most immediate calculations that need to be performed is the BDE^{calc} numbers for all of the titratable amino-acids.

Recall that the BDE^{calc} for Cysteine is within 6% of the thio-methane *H-S* bond dissociation energy, and that we can calculate this BDE^{calc}_{cys} to a precision of 0.05 *pH* units. If the BDE^{calc}'s of the other titratable amino acids agree this well with their respective dissociation energies, this method will assume major significance in the field of force field development: our method will be an eligible tool for fine tuning force field parameters for the following reason. The dissociation energy is already built into the Amber force field in a very indirect way. The partial charges, van der Waals parameters, bond parameters and all of the ff parameters are all calibrated to fit an array of empirical data, so these dissociation energies are very indirectly in the Amber force field. There are many parameters that are empirically tweaked so that they collectively fit a wide database array. There is therefore some concern about compensating errors. However our methods, with precision at least as good as the precision of the experimental numbers, offer a direct comparison between experimental Bond Dissociation Energies and the calculated Bond Dissociation Energies. Our method may therefore serve the useful role of fine-tuning, validating or developing force field parameters.

8.2   MULTI-SITE FUNCTIONALITY

We will now discuss the application of our methods to multiple titratable site systems. In summary our methods allow for pKa calculation of multiple site systems that scales like $N \times 2$ instead of 2^N. It becomes easier to understand our approach to multiple site systems if one were to first spend a little time becoming familiar with some new notation and concepts. Previously, we talked about the proton chemical potential, μ $(= -kT \log 10 \times pH)$ as being a state variable, and being a single valued scalar. It is **easier** to understand the application of our MD/MC-WHAM methods to multiple sites if we consider two things. The first is that we remind ourselves of the discussion in section 4.5.2 pertaining to (4.2) (conjugate variables *pH* and *occupancy ratio*) and (4.3) (pKa=pH, occupancy ratio=1). That is we don't treat the conjugate pair {μ, *occupancy ratio*} in the usual way where the state variable μ is fixed and the configurational variable "occupancy ratio" is a function of μ, i.e. occupancy ratio(μ). Instead, we fix   occupancy ratio=1   and   we   allow   WHAM   to   find   the   correct μ such   that $pKa = \mu(\text{occupancy ratio} = 1)$. The second thing to consider is that $\underline{\mu}$ as an array of several values, $\underline{\mu} = (\mu_1, \mu_2, ...\mu_N)$, where N is the number of titratable sites in the system. We will discuss this concept and any related notation in the next sections.

8.3   PROTON CHEMICAL POTENTIAL AS A VECTOR

Reader please be aware that in our discussions, we will abbreviate "proton chemical potential" to simply "chemical potential".

 We are accustomed to describing the pH of a system with a scalar, single valued chemical potential, $\mu(= \frac{1}{\beta} \log 10 \times pH)$. Effective energy of a system is $1/kT(U + PV + \mu L)$, where the state variables are temperature T or $\beta(= 1/kT)$, pressure P and chemical potential μ. The configurational variables are the potential energy U, the system volume V and the number of

protons in the proton bath $L = l(\eta_1) + l(\eta_2) + ... l(\eta_N)$. The effective energy can therefore be rewritten as $1/kT(U + PV + \mu l(\eta_1) + \mu l(\eta_2) + ... \mu l(\mu_N))$.

Now we introduce the concept of micro-chemical potentials, which involves assigning a micro chemical potential to each titratable site.

Effective energy $= 1/kT(U + PV + \mu_1 l(\eta_1) + \mu_2 l(\eta_2) + ... \mu_N l(\eta_N))$

The chemical potential of site i, μ_i can be written as $\mu + \delta\mu_i$, so that

Effective energy $= 1/kT(U + PV + (\mu + \delta\mu_1)l(\eta_1) + (\mu + \delta\mu_2)l(\eta_2) + ... (\mu + \delta\mu_N)l(\eta_N))$

which is equal to $1/kT(U + PV + \mu L + (\delta\mu_1)l(\eta_1) + (\delta\mu_2)l(\eta_2) + ... (\delta\mu_N)l(\eta_N))$

The state variables of a system then become $\beta(=1/kT), P$ and $\underline{\mu}$ where $\underline{\mu} = (\mu, \delta\mu_1, \delta\mu_2, ..., \delta\mu_N)$. At this point, the concept of micro-chemical potentials may seem completely abstract. The purpose of this section is simply to explain the notation, and its usefulness will be revealed in the next section.
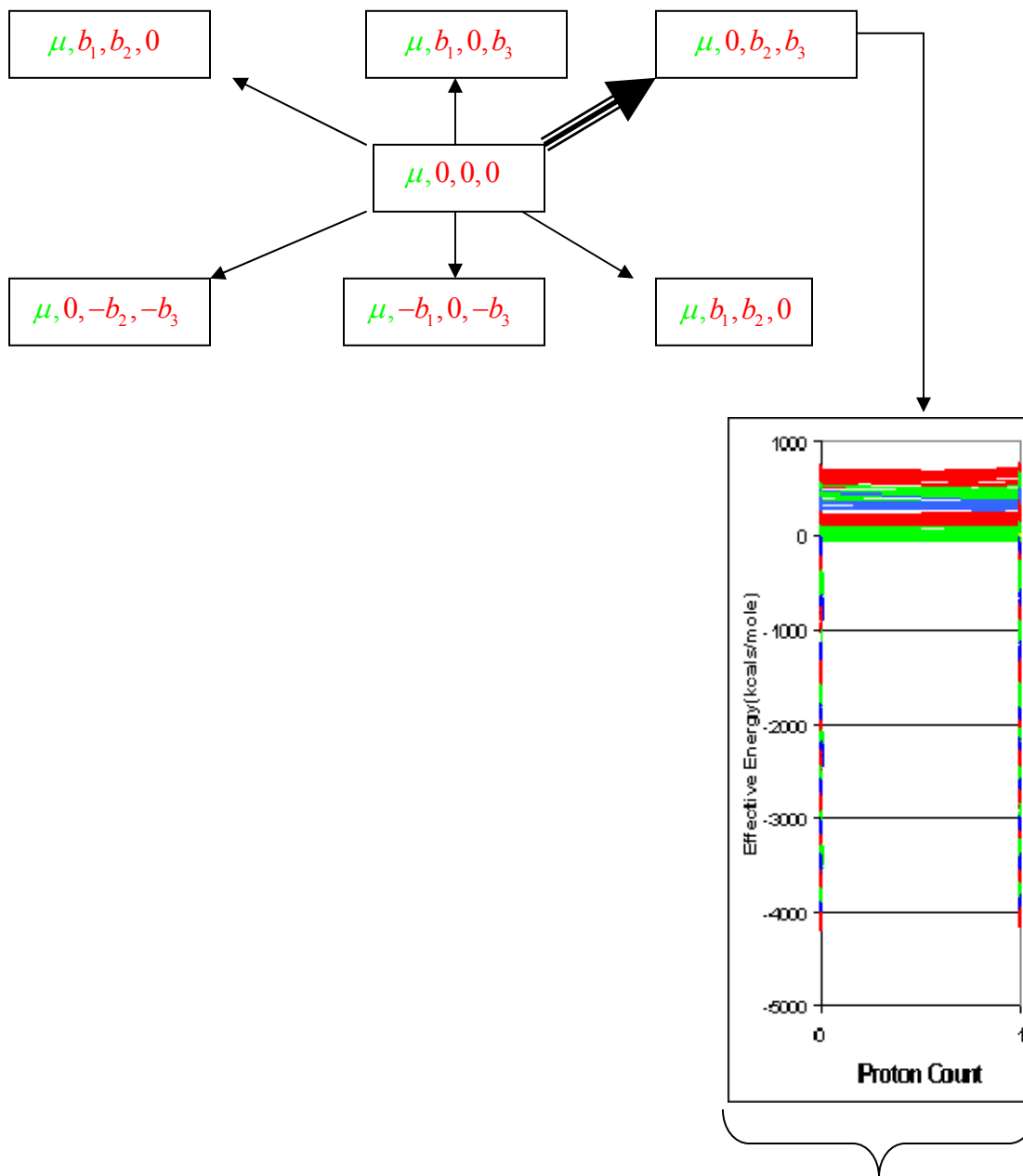

8.4   IMPLEMENTATION OF MULTIPLE SITE FUNCTIONALITY

Let us review our single site method (performed in section 5.2) in light of our new understanding and new notation for chemical potential (discussed in sections 8.3 above).

Let us briefly consider a system of N titratable sites. The chemical potential for this system can be described as $\underline{\mu}$, where $\underline{\mu} = \mu, \delta\mu_1, \delta\mu_2, ..., \delta\mu_N$. Returning to our single titratable site system, we have $\underline{\mu} = \mu, \delta\mu_1$. This simply means that the chemical potential of site one (the only site) is $\mu + \delta\mu_1$. For pKa calculations of this single site system, $\delta\mu_1 = 0$. In other words, $\underline{\mu} = \mu, 0$ or $\mu_1 = \mu + 0$. The green and the red components of the chemical potential signify different components of the chemical potential, one that is calculated and one that is a traditional fixed variable of state. Since the fixed red component of $\underline{\mu} = \mu, 0$ is zero, $\mu_1 = \mu$ is totally free to float in our pH iteration scheme, and it is a calculated value. The calculation of $\mu_1 = \mu$ @300K and 1atm is exactly what we did in section 5.2. I represent that calculation with the following diagram.
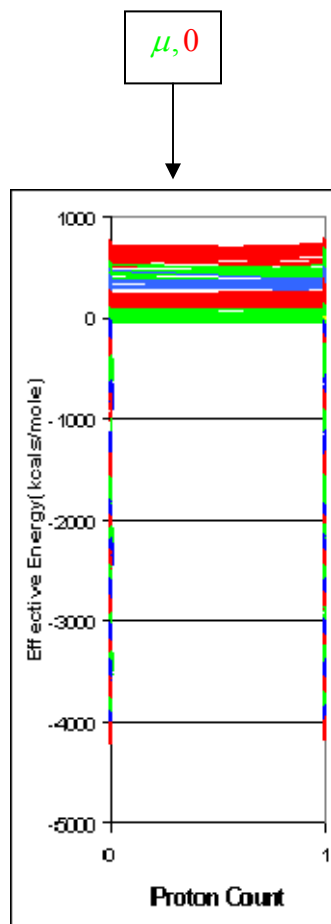
Now consider a three titratable site system, $\underline{\mu} = \mu, \delta\mu_1, \delta\mu_2, \delta\mu_3$ (which means $\mu_1 = \mu + \delta\mu_1$, $\mu_2 = \mu + \delta\mu_2$, $\mu_3 = \mu + \delta\mu_3$) and suppose we want to calculate the $pKa's$ for site number one. Note that we use $pKas$, plural, for site one. This is because in a multi-site system, each site may have several pKas, such as an acidic pKa and a basic pKa, if that site is involved in a network with other sites. For calculation of the pKas of site number one, we set up the calculation where $\underline{\mu} = \mu, 0, \delta\mu_2, \delta\mu_3$. Similarly, to calculate the pKas for sites two and three, we would set up the calculation as $\underline{\mu} = \mu, \delta\mu_1, 0, \delta\mu_3$ and $\underline{\mu} = \mu, \delta\mu_1, \delta\mu_2, 0$ respectively. The diagram below represents the manner in which such calculations would be performed, with special emphasis on the acidic micro pKa calculation for site one.

Figure 39: Acidic micro-pKa of site number 1

Consider the data that goes into the calculation above (the Acidic micro-pKa for site number 1). This data set consists of trajectories generated over a range of temperatures, a range of pressures, and a range of micro-pHs, where the range of $\delta\mu_2$ is $\delta\mu_2[0, b_2]$ and the range of $\delta\mu_3$ is $\delta\mu_3[0, b_3]$. The values of b_2 and b_3 do not matter very much. The important thing is that they

are large enough to force states two and three into protonation. Micro-chemical potentials $\delta\mu_1 = b_1, \delta\mu_2 = b_2$ and $\delta\mu_3 = b_3$ can be thought of as devices for forcing a protonation state on a site without affecting the pH or protonation state of another site. In reality, only about three values of $\delta\mu_i$ in the range $[0, b_i]$ should be enough for sufficient sampling and to get the job done.

In similar fashion, the basic micro-pKa for site one would be set up as described below.
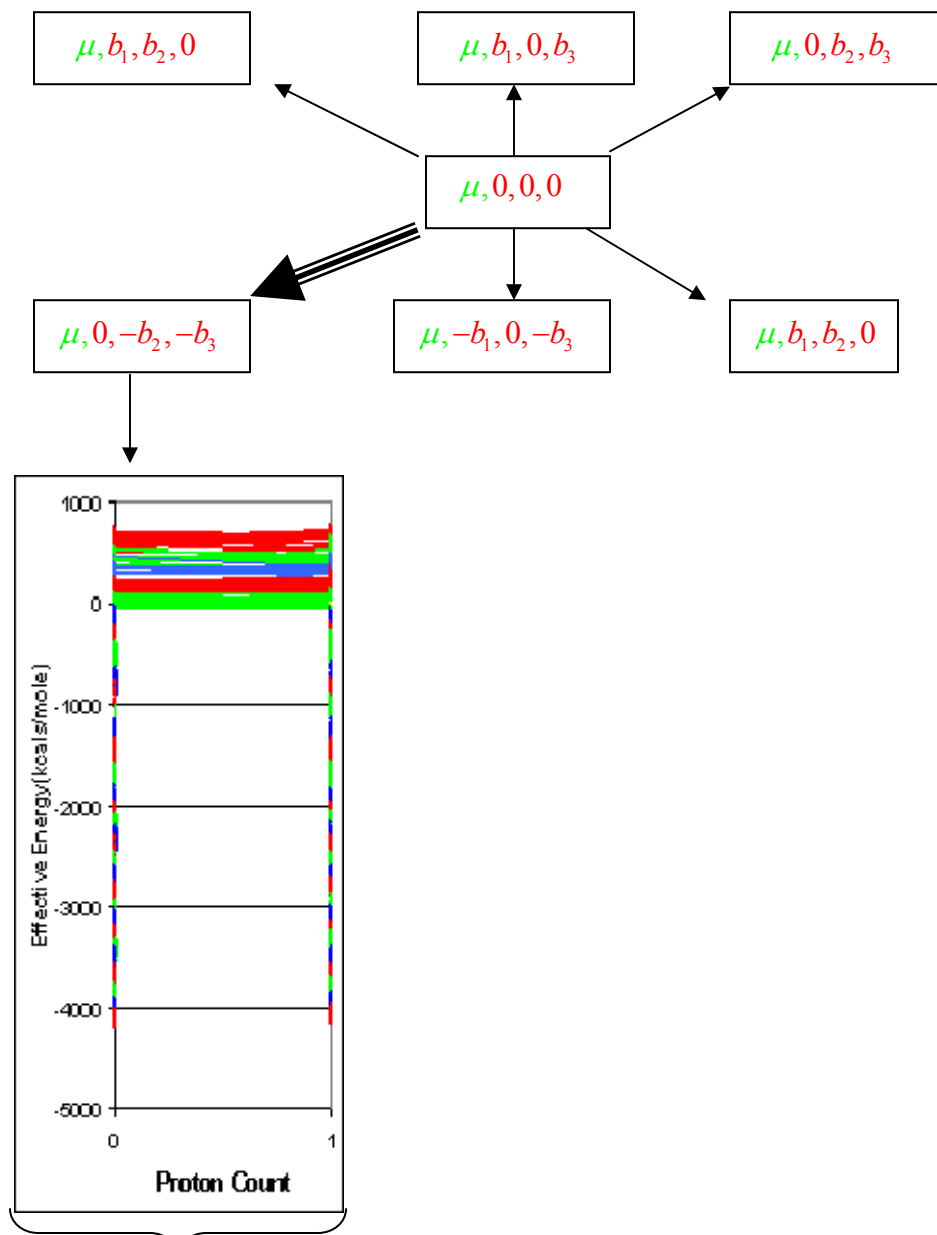
Basic micro-pKa of site # 1

Figure 40: Basic micro-pKa of site# 1

The simulations that go into this calculation would have the usual range of temperatures and pressures, and the site two and three micro pHs would range such as $\delta\mu_2[0,-b_2]$ and $\delta\mu_3[0,-b_3]$. Again, the important thing is that b_2 and b_3 are large enough to force sites two and three into deprotonation. We follow these examples to calculate the micro-pKas for the other titratable sites.

Consider again the acidic and basic micro *pKa* calculation for site one. One simplification that would have a minimal effect on results, is if $b_2 = b_3$. Then the chemical potential $\underline{\mu}$ could be simplified to μ, $\delta\mu_1$, $\delta\mu_a$ where $\delta\mu_a$ is the micro pH of all other sites. Assuming we use as many as four values in the range $\delta\mu_a[0,b]$, **the computational time taken by our method stays constant, regardless of the number of titratable sites, N.** Hence we have bypassed the need to explore every single interaction possibility, which causes other multiple site pKa methods to scale as 2^N or 4^N. In situations requiring calculations for all of the micro pKas of a system, our method will scale linearly with N.

How do we escape the need to explore all 4^N possible protonation states? In the discussion that follows we will see that the key is the high temperature. Consider the dataset shown in Figure 40, where energy is plotted against ionization state. The plots near the top represent the high temperature simulations, where there are ionization state transitions. The plot at the bottom represents a pair of 300K simulations. If our calculations were based on the 300K low temperature simulations alone, we would have to explore all 4^N possible protonation state possibilities. In not doing so, we run a risk of leaving out important information (low energy states) for our density of states description.

But in our method, the 300K information is only a small part of the density of states information. The bulk of the information comes from the high temperature simulations, which gives an approximate description of the density of states, and the low temperature simulations fine-tune it. Because high temperature allows the system to easily cross barriers, the high temperature simulations do an excellent job at vigorous sampling, and hence do an excellent job at discriminating between the important and the unimportant protonation states, giving a good approximation of the density of states. The low temperature simulations then serve the limited purpose of fine-tuning the weighting factors.

Therefore the risk of leaving out important information is minimized, and any errors introduced by the 300K simulations because it misses important information is minimized. So the high temperature bridge trick, which was a necessary nuisance in the single site system to overcome the solvation shell barrier, now saves us from the 4^N problem in multiple site systems!

9.0 SUMMARY AND CONCLUSION

We have presented here a method of doing proton dynamics for explicitly solvated proteins and conducting a full range of thermodynamic calculations for them. The main advantage of our method is that it models the system with atomic detail solvent and solute, and uses discrete protonation states. Preliminary results show that our method promises thermodynamic results of very good precision. Another promising advantage of our method is its feasibility for multi-site systems, with computer time growing as the number of titratable sites N, as opposed to growing with the total number of possible protonation microstates (2^N or 4^N).

The method uses Molecular Dynamics, Monte Carlo for discrete protonation state selection, and Weighted Histogram Analysis for blending a wide range of trajectories. The method still has to be filled out. Specifically, we need to run more calculations so that we can verify the predictions in the cpu cost-precision table for the BDE^{calc} for Cysteine (Table 5 page 181). Then, choosing some computationally feasible precision, we will simply use the protocol that was used to get that precision for Cysteine's BDE^{calc} to calculate the other BDE^{calc}'s. We also need to demonstrate how our method scales with the number of titration sites. Despite these things that still need to be done, the vast majority of the code writing is done, and its performance so far gives us elevated confidence that the method will work and do what it promises to do. Our BDE_{cys}^{calc} for the $H-S$ bond in Cysteine is within 3% of the experimental $H-SCH_3$ bond dissociation energy ($BDE_{thio-methane}^{exp}$). Where BDE_{cys}^{calc} precision is concerned, we can achieve a precision of 0.05 pH units with a less than 97,000 processor-hours.

As a result, one of the exciting promises the method makes is that it can do direct, accurate and precise measurements that can be compared to experimental dissociation energies (BDE^{exp}'s), and consequently be used as a method to fine-tune force field parameters. However

more of these BDE^{calc} 's need to be calculated, compared with appropriate dissociation energies and analyzed to see exactly the value of our method for validating force field parameters.

Of course, the main promise it makes is the ability to yield a full range of accurate and precise thermodynamic calculations.

BIBLIOGRAPHY

[1] Donald Bashford, Martin Karplus. pKa's of Ionizable Groups in Proteins: Atomic Detail from a Continuum Electostatic Model. *Biochemistry* 29:10219-10225 (1990)

[2] Clauda Schutz, Arieh Warchel. What Are the Dielectric "Constants" of Proteins and How To Validate Electrostatic Models? *Proteins* 44:400-417 (2001)

[3] Phillip Nelson. 2004. *Biological Physics*. New York: W.H Freeman. p16-17

[4] NIH Computer Retrieval of Information of Scientific Research (CRISP)

[5] NIH Office of Extramural Research, Award Data. 2006. *http://grants2.nih.gov/grants/award/success/Success_ByIC.cfm*

[6] NSF Budget Internet Information System. 2006. *http://dellweb.bfa.nsf.gov/*

[7] NSF Awards Abstract Database. 2006. *http://www.nsf.gov/awardsearch/*

[8] NCSA Past Awards Database.2006. *www.ncsa.uiuc.edu/UserInfo/Allocations/awards.html*

[9] Phillip Nelson. 2004. *Biological Physics*. New York: W.H Freeman. p60-61

[10] Privalov. P. L. *Adv. Prot. Chem.* **33,** p167-241 (1979)

[11] Creighton, T. E. *Protein Folding* (Freeman, New York) (1992)

[12] Ptitsyn, O. B. *Advances in Protein Chemistry* **47,** p83-229 (1995)

[13] Ptitsyn, O. B. & Uversky, V. N. *FEBS Lett.* **341,** p15-18 (1994)

[14] Kuwajima, K. *Prot. Struct. Funct. Genet.* **6,** p87-103 (1989)

[15] K. A. Lau & K. A. Dill, *Macromolecules,* **22**, p3986 (1989)

[16] Vijay S. Pande & Daniel S. Rkhasar, *PNAS,* **95,** p1490-1494, (Feb 1998)

[17] Lesser, D. R., Kurpiewski, M. R. and Jen-Jacobson, L. *Science* **241** p776-785 (1990)

[18] Mark Gerstein and Michael Levitt, *Simulating Water and the Molecules of Life,* Sci. Am. **279** p105 (Nov 1998)

[19] Craig , Ninham and Pashley, *J. Phys. Chem.* **97**, 10192 (1993)

[20] Ederth T, Tamada K, Claesson PM, Valiokas R, Colorado R Jr, Graupe M, Shmakova OE, Lee TR.J, Colloid Interface Sci. 2001 Mar 15;235(2):391-397

[21] T Uchihashi *et al,* Quantitative *measurement of solvation shells using frequency modulated atomic force microscopy,* Nanotechnology **16** S49-S53 (2005)

[22] Ballew, R.M, Sabelka, J. & Gruebele, M. (1990) *Proc. Natl. Acad. Sci. USA* **93**, 5759-5764

[23] Aihua Xie, Alexander F.G. van der Meer, Robert H. Austin (2002) *Excited-State Lifetimes of Far-Infrared Collective Modes in Proteins* Physical Review Letters, **vol 88, number 1**

[24] J.D. Eaves, J.J.Loparo, C.J.Fecko, S.T.Roberts, A.*Tokmakoff*, P.L.Geissler *Hydrogen bonds in liquid water are broken only fleetingly* PNAS **vol. 102, no. 37,** p13019-13022 (September 2005)

[25] Linus Pauling, *The Nature of the Chemical Bond*. Cornell Univ. Press, Ithaca, New York, 1960.

[26] Drawn using ChemDraw software. The ChemDraw is a registered trademark of CambridgeSoft Corporation (Cambridge Scientific Computing, Inc), www.combridgesoft.com, 100 CambridgePark, Cambridge MA 02140 USA

[27] Yang Ju *et al, Contactless measurement of electrical conductivity of semiconductor wafers using the reflection of millimeter waves*, Applied Physics Letters **Volume 81**, Issue 19, p. 3585-3587 (November 4, 2002)

[28] Adapted from Fig. 14-23 of Voet & Voet *Biochemistry,* 2nd ed., Wiley and Sons (1995)

[29] http://biochem.wustl.edu/~protease/ser_pro_overview.html

[30] Thierry Rose, PhD and Enrico Di Cera, MD, Department of Biochemistry and Molecular Biophysics, Washington University School of Medicine, Saint Louis, MO, U.S.A.

[31] Kavanaugh, J.S. *et al.* "High-resolution x-ray study of deoxyhemoglobin Rothschild 37beta trp->arg: a mutation that creates an intersubunit chloride-binding site," (1992) *Biochemistry,* **31**, 4111. Deoxyhemoglobin PDB coordinates, *Brookhaven Protein Data Bank.*

[32] Royer Jr., W.E. "High-resolution crystallographic analysis of co-operative dimeric hemoglobin," *J. Mol. Biol.,* **235**, 657. Oxyhemoglobin PDB coordinates, *Brookhaven Protein Data Bank.*

[33] Rachel Casiday and Regina Frey, Department of Chemistry, Washington University, St. Louis, MO 63130
http://www.chemistry.wustl.edu/~edudev/LabTutorials/Hemoglobin/MetalComplexinBlood.html

[34] P. Modrich, *Structures and Mechanisms of DNA Restriction and Modification Enzymes*, Quart. Rev. Biophys., 12:315-369, 1979

[35] Compaq Technology Brief, *Exploring Alpha Power for Technical Computing* April 2000

[36] Magnus Ekman, Fredrik Warg, Jim Nilsson *An In-Depth Look at Computer Performance Growth* Technical Report 2004-9, Chalmers University of Technology, Department of Computer Engineering, Göteborg 2004

[37] InfiniBand Trade Association. InfiniBand Architecture Specification, Specification, Release 1.0, October 24 2000.

[38] Mellanox Technologies. Mellanox InfiniBand InfiniHost Adapters, July 2002.

[39] Joachim Hein, Fiona Reid, Lorna Smith, Ian Bush, Martyn Guest, and Paul Sherwood. On the performance of molecular dynamics applications on current high-end systems. *Philosophical Transactions: Mathematical, Physical and Engineering Sciences* Volume 363, Number 1833, Pages 1987-1998, August 15, 2005.

[40] www.ks.uiuc.edu/Research/namd/performance.html

[41] http://amber.scripps.edu/amber7.bench4.html

[42] http://amber.scripps.edu/amber8.bench1.html

[43] http://amber.scripps.edu/amber9.bench1.html

[44] Davis, M. E., Madura, J. D., Luty, B. A., McCammon, J. A. *Electrostatics and Diffusion of Molecules in Solution: Simulations with the University of Houston Brownian Dynamics Program",* Comp. Phys. Comm, **62**, p187-197 (1991)

[45] Madura, J. D., Davis, M. E., Gilson, M. K., Luty, B. A., Wade, R. C., McCammon, J. A. *Biological Applications of Electrostatic Calculations and Brownian Dynamics Simulations,* Rev. Comp. Chem., **5,** p229-267 (1994)

[46] F. S. Lee, Z. T. Chu, A. Warshel, J. Comp. Chem., **14**, p161 (1993)

[47] McQuarrie DA. *Statistical Thermodynamics,* University Science Books (1973)

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

[49] Ziman, J.M., *Principles of the Theory of Solids,* Cambridge University Press, Cambridge, 2nd edition, 1972

[50] Darden T, Perera L, Li L and Pedersen L. (1999) "New tricks for modelers from the crystallography toolkit: the particle mesh Ewald algorithm and its use in nucleic acid simulations", *Structure* **7**, R55-R60.

[51] David Kofke, Department of Chemical Engineering, SUNY Buffalo NY

[52] R. W. Hockney, J. W. Eastwood, *Computer Simulation using Particles,* Adam Hilger (1998)

[53] Levine, Ira N. *Quantum Chemistry,* Prentice Hall (2000)

[54] J.A. McCammon and S.C. Harvey, *Dynamics of proteins and nucleic acids,* Cambridge University Press, New York, 1987

[55] C.H. Brooks III, M. Karplus and B.M. Pettitt, *Proteins: A theoretical perspective of dynamic, structure and thermodynamics,* in *Advances in Chemical Physics,* Vol. LXXI, edited by I. Prigogine and S. Rice, John Wiley and Sons, New York, 1988.

[56] Hu J, Ma A, Dinner AR., *Monte Carlo Simulations of Biomolecules: The MC module in CHARMM,* J Comput. Chem. **27** p203-216 (Jan 30 2006)

[57] Verlet, 1967

[58] Allen & Tildesley, 1989

[59] Brooks, Karplus & Pettitt, 1989

[60] McCammon & Harvey, 1987

[61] Christopher Hammond, *The Basics of Crystallography,* Oxford University Press (1997)

[62] Alan M. Ferrenberg and Robert H. Swendsen, *New Monte Carlo Technique for Studying Phase Transitions*, Phy. Rev. Lett. **61**, 2635 (1988)

[63] S. Kumar, D. Bouzida, R. H. Swendsen, P. A. Kollman, J. M. Rosenberg, J. Comp. Chem., **13**, 1011 (1992).

[64] S. Kumar, J. M. Rosenberg, D. Bouzida, R. H. Swendsen, P. A. Kollman, J. Comp. Chem., **16**, 1339 (1995).

[65] Ma, Shang-Keng, *Statistical Mechanics* , World Scientific Publishing Co, 1985

[66] I.R. McDonald and K. Singer, Disc. Far. Soc. **43,** 40 (1967)

[67] U. Börjesson and Hünenberger, *J. Chem. Phy.* **114,** 9706 (2001)

[68] T. Simonson, J.Carlsson, D.A. Case, *J Am. Chem. Soc.* **126,** 4167-4180 (2004)

[69] M.Born and R.Oppenheimer, *Ann. d. Phys.* **84**, p457 (1927))

[70] Keyes, R.S., Y.Y. Cao, E.V. Bobst, J.M. Rosenberg, A.M. Bobst (1996) *Spin-labeled nucleotide mobility in the boundary of the EcoRI endonuclease binding site*. J. Biomol. Struct. Dyn. **14**, 163-172

[71] Kumar, S., Y. Duan, P.A. Kollman, and J.M. Rosenberg (1994) *Molecular dynamics simulations suggest that the EcoRI kink is an example of molecular strain.* J. Biomol. Struct. Dynam. **12** p487-525

[72] D. Stikoff, K.A. Sharp & B. Honigg. (1994) *Accurate Calculation of hydration free energies using macroscopic solvent models.* J. Phys. Chem. **98** p1978-1988

[73] A. Mitsutake, Y. Sugita and Y. Okamoto. Biophys. J. **85,** p5-15 (2003)

[74] H. Nymeyer, S. Gnanakaran and A. E Garcia. Methods in Enzymology **383,** p119-149 (2004)

[75] X. Cheng, G. Cui, V. Hornak and C. Simmerling. J. Phys. Chem. B, **109,** p8220-8230 (2005)

[76] George Casella, Christian Robert *Monte Carlo Statistical Methods,* p621 Springer (2005)

[77] A.M. Ferrenberg, PhD Thesis, p36 Carnegie Mellon University, Pittsburgh, PA, 1989

[78] D. A. Case, T. A. Darden, T E. Cheatham, III, C. L. Simmerling, J. Wang, R. E. Duke, R. Luo, K. M. Merz, B. Wang, D. A. Pearlman, M. Crowley, S. Brozell, V. Tsui, H. Gohlke, J. Mongan, V. Hornak, G. Cui, P. Beroza, C. Schafmeister, J. W. Caldwell, W. S. Ross, and P. A. Kollman, AMBER 8, University of California, San Francisco (2004)

[79] Y. Duan, C. Wu, S. Chowdhury, M.C. Lee, G. Xiong, W. Zhang, R. Yang, P. Cieplak, R.Luo & T. Lee. *A Point-Charge Force Field for Molecular Mechanics Simulations of Proteins.*J. Comput. Chem. **24,** 1999-2012 (2003).

[80] David R. Lide, *CRC Handbook of Chemistry and Physics* 9-115 (1991-1992)

[81] Chivers PT, Prehoda KE, Volkman FB, Kim BM, Markley JL, Raines TR. Microscopic pKa value of Enscherichia coli thioredoxin. Biochemistry 1997; 36:14985-14991.

[82] Kazuyoshi UEDA, Taro KOMAI, Isseki YU and Haruo NAKAYAMA (2002) *Molecular Dynamics Study on the Density Fluctuation of Supercritical Water.* The Journal of Computer Chemistry, Japan, Vol. 1, No. 3 (2002)

[83] http://www.sccj.net/publications/JCCJ/v1n3/a12/text.html

[84] Bernard R. Brooks, Robert E. Bruccoleri, Barry D. Olafson, David J. States, S. Swaminathan, Martin Karplus, *J. Comput. Chem.*, **4**, 187 (1983).

[85] A.M. Ferrenberg, PhD Thesis, Carnegie Mellon University, Pittsburgh, PA, 1989