DEFIINING THE IMAGINARY EDGE OF HYDRATION ON PROTEINS OF KNOWN STRUCTURE

by

DANIEL ROBERT GRISHAM

A Dissertation submitted to the

School of Graduate Studies

Rutgers, The State University of New Jersey

In partial fulfillment of the requirements

For the degree of

Doctor of Philosophy

Graduate Program in Biomedical Engineering

Written under the direction of

Vikas Nanda

And approved by

_____

_____

_____

_____

New Brunswick, New Jersey

October 2018

ABSTRACT OF THE DISSERTATION

DEFINING THE IMAGINARY EDGE OF HYDRATION ON PROTEINS OF KNOWN STRUCTURE

By DANIEL ROBERT GRISHAM

Dissertation Director:

Vikas Nanda, Ph.D.

This work developed an experimentally verified computational protocol (called ZPRED) for modeling the zeta potential of molecules (e.g. proteins or drugs) from their structure with the long-term goal of being used as a design tool for predicting the onset of molecular self-assembly. The zeta potential (ζ) is the effective charge energy of a solvated molecule and is commonly used to assess how well separated molecules remain in solution (e.g. in pharmaceuticals, medical diagnostics, cosmetics, etc.). The ζ exists at a position away from the molecular surface, where ions and water no longer adhere to the protein, called the slip plane position (X_{SP}). However, the information gap is the X_{SP} is not generally defined and can vary based on solution conditions (ionic strength, pH, and temperature) as well as flow at the protein-solvent interface. Thus, the objective of this work aims to relate the X_{SP} of select, compact globular proteins to their solution conditions and attempts to extend the relation to general fibrillar proteins using a

collagen-like triple helix as a model system. Completing this objective tested the central hypothesis: the X_{SP} coincides with the solvation edge defined by the Stokes-Einstein hydrodynamic radius (Rh), and thus the two should hold the same dependence on solution conditions. The rationale is since diffusing globular proteins hold similar translational motion during electrophoresis; hydration should be equivalent when solution conditions are held constant with any deviation representing the difference in flow perturbations at the protein-solvent interface. This work was accomplished through variation in each solution condition ensuring ZPRED to be accurate for any general aqueous electrolyte solution. Experimental light scattering methods indicated coincidence of the X_{SP} and Stokes-Einstein hydrodynamic radius for a number of proteins in a wide range of solution conditions.

ACKNOWLEDGEMENTS

I am in debt to Dr. Vikas Nanda, the Rutgers Counseling, ADAP and Psychiatric Services and Jennifer Gomez-Vivas for providing a supportive and nurturing environment in the dark years of my graduate career. I would not be alive today without their influence on my life.

DEDICATION

To the best boss I have ever had,

Dr. Vikas Nanda

To my love,

Jennifer Gomez-Vivas

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

BLG	Bovine β-Lactoglobulin AB
BSA	Bovine Serum Albumin
DLS	Dynamic Light Scattering
ELS	Electrophoretic Light Scattering
EDL	Electric Double Layer
LYZ	Hen Egg White Lysozyme
NSE	Navier-Stokes Equation
PALS	Phase Analysis Light Scattering
PBE	Poisson-Boltzmann Equation
ZPRED	Zeta Potential Prediction in a General Aqueous Electrolyte Solution

INTRODUCTION

What is the zeta potential?

The zeta, or electrokinetic, potential (ζ) is the effective charge energy of a molecule in a specific solution. It is used to assess ionic adsorption processes and how well separated a molecule remains from itself (i.e. electrostatically stabilized) in a specific solution [1, 2]. Experimentally determined ζ values have been applied to optimize therapeutic antibodies and other proteins for formulation conditions that promote long-term dispersion stability [3, 4, 5, 6, 7, 8], and to study interactions between proteins with particles, materials and surfaces [9, 10]. Its value characterizes charge interactions at the interface between a solvated molecular surface and solvent in motion. In other words, the ζ can be thought as the electric potential acting on solution that is the result of adhered solvent dampening the surface potential. For lack of a better word, this effect is termed surface potential dampening and will be discussed in detail with the concept of hydration later. If solvent were to have no dampening effect on the surface potential, the ζ and surface potential would be equivalent. However, this case (referred to as the "slip" boundary condition) is only seen on randomly-oriented chains in this work, but it is expected to occur for any small protein [11, p. 259]. The main point here is the ζ is just as dependent on the solvent as it is dependent on the charged surface defining it. Thus, as described in **APPENDIX C**, the solvent properties must be accurately modeled in order to accurately model the ζ and convert it to a kinetic value for experimental comparison (e.g. electrophoretic mobility).

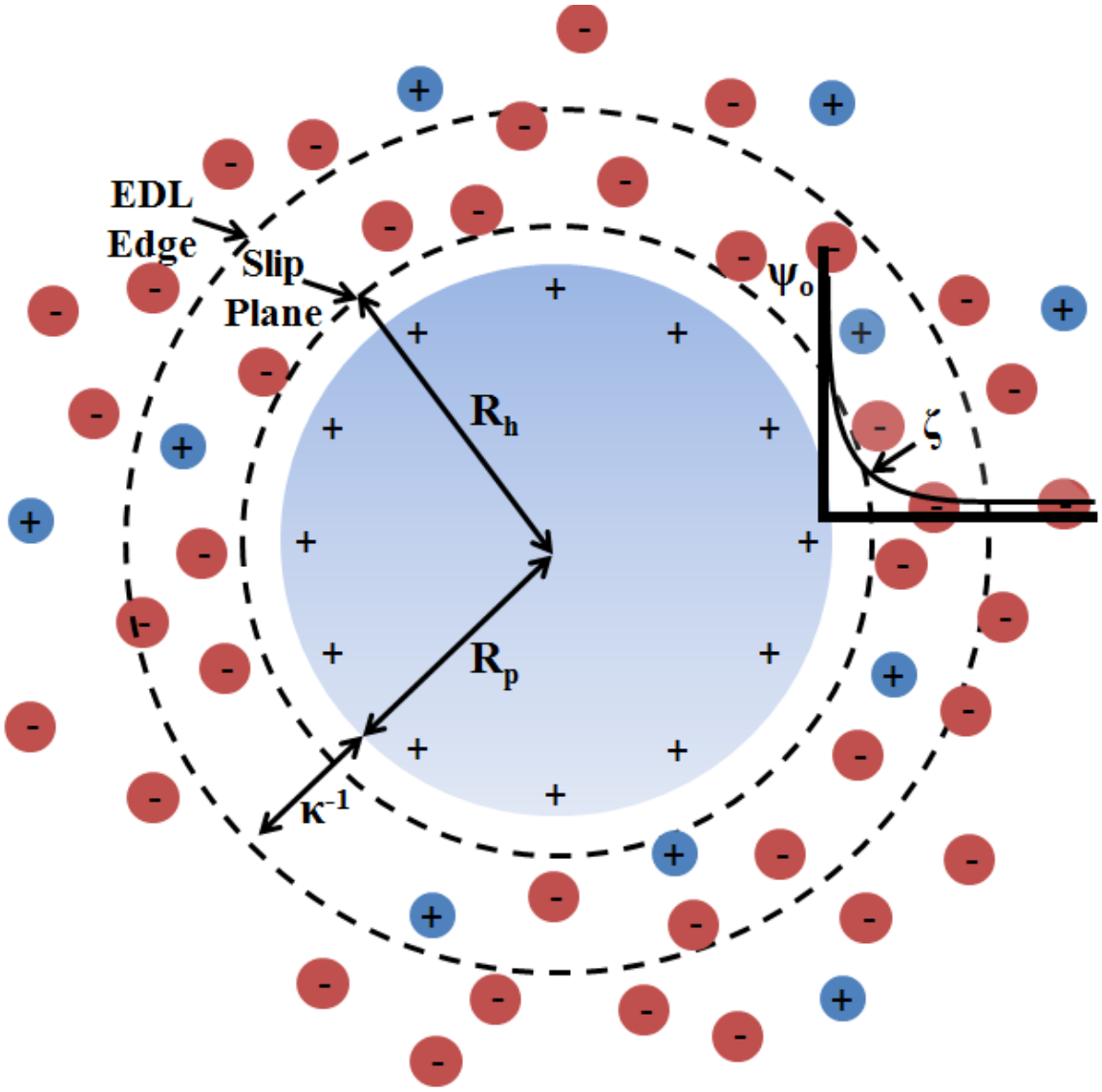

Figure 0-1. Idealized Electric Double Layer Around a Spherical Cation. Plotted on the top right, the surface potential (ψ_o) propagates outward into the cloud of ions treated as point charges immersed in a solvent with a constant relative dielectric. The zeta potential (ζ) is located at the slip plane, which is proposed to coincide with the hydrodynamic radius (R_h). R_p is the protein radius and κ^{-1} is the Debye length.

To visualize the ζ, the protein-solvent interface is modeled as an electric double layer (EDL), which is the collection of solvation layers that form around a protein in an attempt to neutralize its charge. Gouy and Chapman [12] developed an EDL model containing a molecule with a uniform surface charge that is neutralized by a region of

diffusing ions that encompass the molecular surface. The propagation of the surface potential and ion concentrations from the surface are defined by the Poisson-Boltzmann equation (PBE) (see **Eq. B4**). **Figure 0-1** depicts key features of the EDL surrounding an idealized cationic spherical protein, and the electrostatic potential distribution extending into solution from the protein surface. A hydration layer extends from the surface to the slip plane position (X_{SP}), similar to the Stern layer of the Gouy-Chapman-Stern (GCS) EDL. The depicted EDL assumes that the propagation of electrostatic potential and ion concentrations within the hydration layer are defined by the nonlinear PBE, unlike the GCS EDL, which applies a modified PBE to consider ion size constraints [13]. The ζ is typically weaker than the surface potential (ψ_o) and located at the X_{SP}, which is somewhere in the cloud of diffusive ions less than a Debye length (κ^{-1}) away from the surface. It is important to note the X_{SP} is only an imaginary position defined for mathematical convenience. The existence of a sharp cutoff between the hydration layer and bulk solvent is physically unrealistic [14, 15]. It is more appropriate to think of the hydration layer as a region of solvent encompassing a protein, where solvent motion is significantly hindered relative to the bulk, causing an apparent increase in the size of the protein. This region is only a few molecular-sized layers thick [16] (in general about 2 to 4 Å [17]), and ions adsorbed within it can cause "specific-ion effects" [1, p. 233] that can be modeled with the GCS EDL [13, 18]. Nonetheless, the X_{SP} defines the thickness of the hydration layer and thus it is more meaningful to think of it as the distance from the protein surface, over which surface potential dampening occurs.

The solvent effect of surface potential dampening within the hydration layer depends on the charge and geometry of the foundation on which the hydration layer sits.

Thus, it is possible for different proteins, even of the same net charge in the same solution conditions, to experience different extents of surface potential dampening (see **Fig. 0-2**). The computational protocol developed in this thesis (called ZPRED) assumes the solvent's dampening effect is defined by the non-linear PBE. Based on the experimental accuracy achieved by ZPRED, this apparently is not a bad assumption for the proteins studied. ZPRED defines the distribution of hydration over a molecular structure following the "uniform hydration layer model", which is believed to be physically appropriate [17].

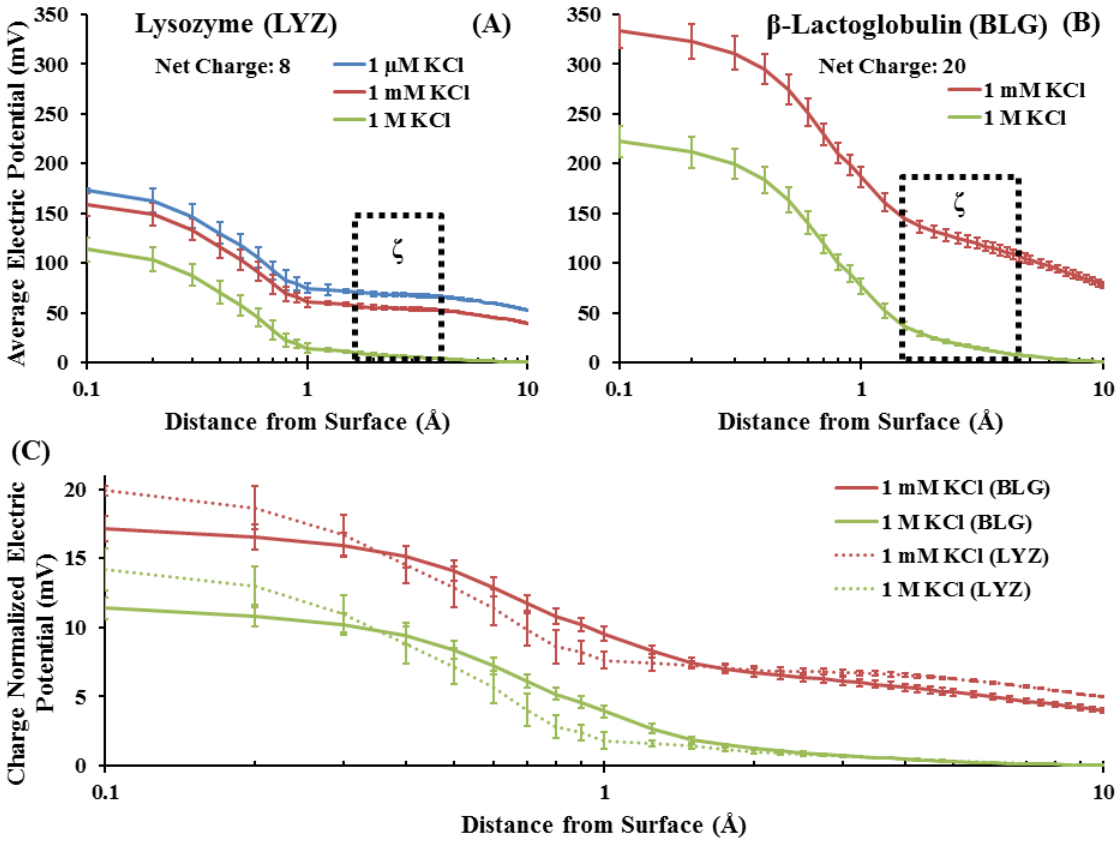

Figure 0-2. Computed Surface Potential Dampening of Hen Egg White Lysozyme and Bovine β-Lactoglobulin with KCl. (**A**,**B**). The surface potentials and potential dampening with distance from the surface were computed for each protein (PDB id: 6lyz for LYZ and 3blg for BLG) using ZPRED. Dashed boxes containing the zeta potential symbol (ζ) represent the distance range, where the ζ could be defined, with the minimum being the radius of water (1.42 Å [68]) and the maximum being the difference between the HYDROPRO computed hydrodynamic radius and protein radius. The magnitude of the surface potential and its dampening depends directly on the protein net charge. Thus to compare the two protein geometries, potentials must be divided by the net charge valence. (**C**) Comparison of the averaged electric potentials propagating from the surface divided by net charge. The dampening on the charge normalized potentials provides a comparison of the affinity each protein's surface geometry has for the solvent. As can be seen, the Cl^- ion more effectively dampens the surface potential of lysozyme relative to β-lactoglobulin with increasing distance.

What is hydration?

Hydration is a "solvent effect" [19, p. 24] referring to the process of developing a slowly diffusing layer of solvent over a molecular surface [15]. In other words, it refers to the formation of ordered structures of solvent at the molecular surface. Hydration is a

complicated phenomenon depending on the flow of solvent relative to the molecule [14, p. 302], ion type [20], ion concentration [21], temperature [14, p. 244, 22], pH [23], and pressure [24]. A further complication arises from the fact that solvent can either "slip" or "stick" [11, p. 259] to the protein surface depending on the surface residues [14, p. 326] and size of the protein [11, p. 259]. In the past, hydration was quantified as the ratio of the mass of bound water to the mass of the protein. With atomic resolution structures now available [25], hydration can also be quantified by the hydration layer thickness (i.e. the X_{SP}) [17]. There are two models for defining the distribution of hydration over a molecular surface: the uniform expansion model and uniform hydration layer model [17] (see **Fig. 0-3**).

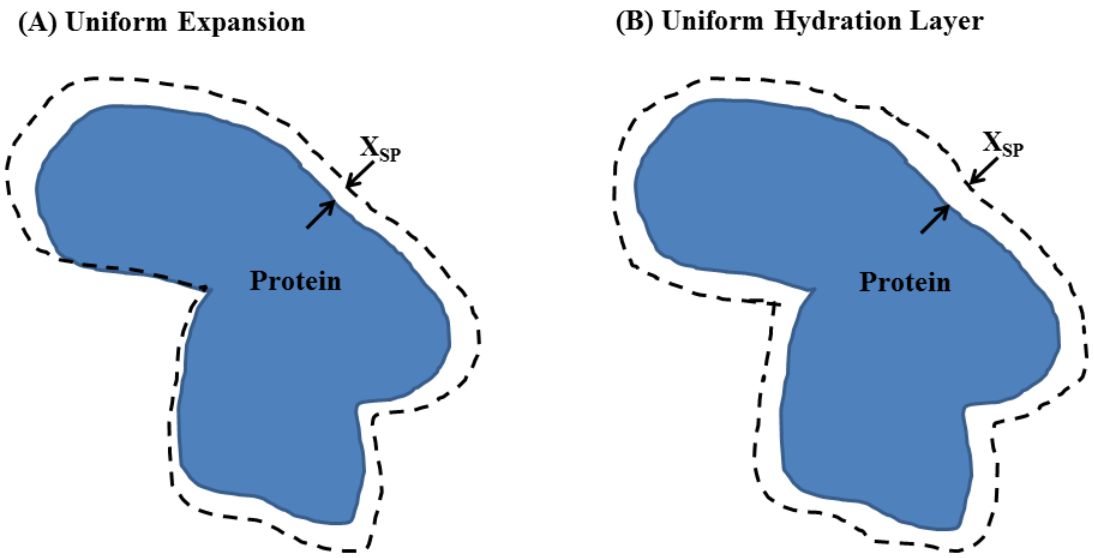

Figure 0-3. Comparison of Hydration Distribution Models. (**A**) The uniform expansion model results in a variable hydration layer thickness (X_{SP}) based on the dimensions of the protein. (**B**) The uniform hydration model applies a uniform hydration layer thickness as its name implies.

In the uniform expansion model (**Fig. 0-3(A)**), the hydration layer is believed to extend from the protein surface based on the protein dimensions relative to its center. This results in a hydration layer of variable thickness (unless you are dealing with a perfect sphere) proportioned to the distribution of mass within the protein. This model is unrealistic for fibrillar proteins as their distribution of mass causes excessive elongation of the hydration layer along the length of the protein. On the other hand, there is the uniform hydration layer model, which assumes hydration maintains a constant thickness (i.e. the X_{SP}) over the protein surface (**Fig. 0-3(B)**). Based on the accuracy achieved by

ZPRED and HYDROPRO, this distribution model is believed to be “the proper, physically significant way to account for hydration” [17].

As previously mentioned, hydration is a complex phenomenon resulting from multi-body interactions depending on the way solvent moves relative to the protein and the solution conditions. The effect of solvent motion relative to the protein is discussed later, where the connection between diffusion and electrophoresis is made. Because the change in pressure necessary for altering hydration is extreme [24], a study on the effect of pressure on hydration is outside the scope of this thesis, which works at normal atmospheric pressure. However, the other parameters (ionic strength, temperature and pH) still require discussion. The impact ions hold on the hydration layer is complex [20, 14] and can have a number of effects. In general, ions displace water from or bring water into the hydration layer, possibly by the mechanism depicted in **Fig. 0-4**. The presence of ions alters the structure of bulk water effecting solution properties (e.g. the relative dielectric, viscosity, and density), not to mention it can alter the structure of the protein as well [14, p. 281]. Thus, a complete definition of the protein-solvent interface should not only include protein-ion and protein-water interactions, but also ion-ion and ion-water interactions [26, 19]. Temperature, being a measure of internal energy, has the obvious effect of altering the rate of diffusing water and thus the rate of motion within and outside of the hydration layer [27]. In addition, temperature affects the solubility and water activity of ions, which can have significant effects on the solution properties (e.g. viscosity, density, relative dielectric). Thus, temperature can also affect hydration by changing the behavior of the ions interacting with the hydration layer. Hydration dependence on pH is an interesting concept considering the ions defining the pH

specifically adsorb to the protein inside its hydration layer to define it net charge. The hydration of an ion strongly depends on its surface charge density [26]. So as the ions defining the pH build the hydration layer foundation, they are also defining the surface charge density of the protein, which also affects the hydration layer. pH, being the power of the hydrogen ion concentration, is "the fundamental potential determining ion concentration scale" [1, p. 224]. Hunter refers to protons and hydroxides as a particular kind of specifically adsorbing ion, called a potential determining ion, that creates the surface charge on a protein [1, p. 223]. Thus, pH affects hydration by defining the charge foundation on which the hydration layer sits [28, 23]. Having covered the factors influencing hydration, let's take a look at different mechanisms that can alter the hydration layer via ionic interactions.

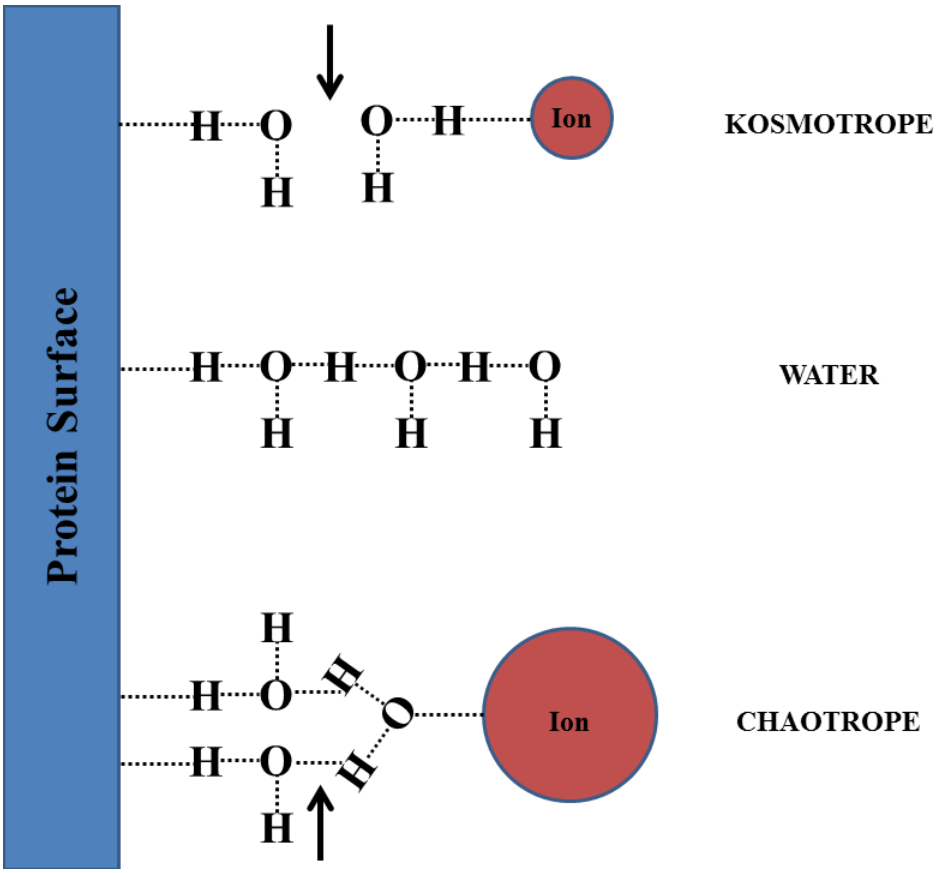

Figure 0-4. The Effect of Kosmotropes Versus Chaotropes on Polar Protein Surfaces. (**Top**) Kosmotropes (i.e. ions with relatively high surface charge density) tend to disrupt the hydration layer as they compete with it for binding water. (**Middle**) Kim Collins defines the interfacial region between a protein and solvent as three molecular-sized layers of solvent extending from the surface [26]. (**Bottom**) Chaotropes (i.e. ions with relatively low surface charge density) tend to build the hydration layer as they hold low affinity for water.

Depicted in **Fig. 0-4**, the first mechanism discussed is related to the lyotropic, or direct Hofmeister, series [29, 26, 30, 31, 32, 33, 34], which is believed to exist as a result of complex ion-water-protein interactions [20]. As water adheres to a charged surface, oppositely charged ions attracted to the surface interact with neighboring waters that attempt to form orientation-dependent hydrogen bonds.

> "Highly directional polar hydrogen bonding interactions dominate in aqueous solution, and a water molecule cannot achieve the maximum pairwise enthalpy of interaction with each of its immediate neighbors simultaneously; it will therefore "choose" to interact most strongly with the neighbor for which it has the most

> favorable pairwise enthalpy of interaction." "When a strongly hydrated anion or neutral solute (kosmotrope) is inserted into the third water layer, the second water layer is "busy" solvating the kosmotrope and cannot help the first layer solvate the protein surface [top of **Fig. 0-4**]; the solution thus becomes a poorer solvent and the protein attempts to minimize its solvent exposed surface area by becoming more compact (and rigid). In contrast, when a weakly hydrated anion (chaotrope) is inserted into the third water layer, the second water layer is freed up to help out the first water layer solvate the protein surface more effectively [bottom of **Fig. 0-4**]; the solution thus becomes a better solvent and the protein attempts to maximize its solvent exposed surface area by unfolding." [26]

The mechanism proposed above is one possible way ions at the protein-solvent interface (i.e. edge of the hydration layer) can influence a protein structure and behavior in solution. Another mechanism that can alter the hydration layer arises from the extension and folding of functional groups at the protein surface in response to changes in ion concentration (**Fig. 0-5**). This behavior is commonly seen with particles that can be modeled by the "Hairy Layer EDL", such as polystyrene beads with chemically treated surfaces [35]. Speaking of ion concentration, the next concept worth discussing is that of ionic strength.

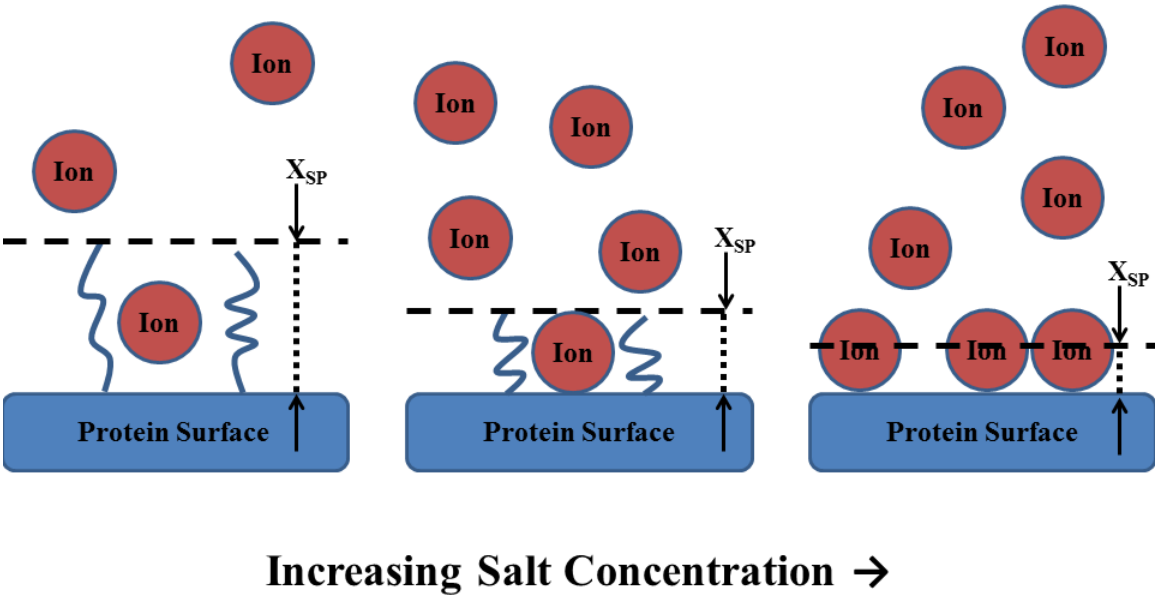

Figure 0-5. An Example of Slip Plane Motion with Increasing Ion Concentration in the Hairy EDL Model. (From left to right) Extended chains on the molecular surface fold in response to an increase in ion concentration reducing the hydration layer thickness.

What is ionic strength?

It is a common belief that the ionic strength of a general solution can be simply defined by half of the sum of the products of each constituent ion's molal concentration and squared valence [22]. However, this definition is only applicable for strongly dissociating electrolytes at low ion concentrations [36, 37] as it is based on the principle: "in dilute solutions the activity coefficient of a given strong electrolyte is the same in all solutions of the same ionic strength" [22, p. 1141]. A more complete definition of ionic strength must consider ion-water interactions [11, p. 30] and the fact that ions (even of the same valence) are not the same. For example, anions are more hydrated than cations

of the same charge density [38]. Kiriukhin et Collins claim “the ratio of the strength of ion-water interactions to the strength of water-water interactions defines the behavior of ions [in solution]” [37]. The main point here is ions are not the same as they hold different sizes and surface charge densities that influence their dissociation in water. Thus, to assess the charge-quenching power of an ion, it must be compared to a standard ion on the same charge foundation and in the same solution conditions. Both K and Cl are only weakly chaotropic and are often used as a standard of comparison in the Hofmeister series [33, 11, p. 35]. Thus, KCl was used as a standard of comparison in this work as well. Keeping in mind it is counter-ions that predominately accumulate within the EDL [1]; the counter-ions are expected to define the solvent behavior within the EDL at the hydration layer [39]. This fact is supported from a previous study [40] that measured the electrophoretic mobility of negatively charged polystyrene beads as function of KF, KCl, KBr and KI concentrations to study the effect of co-ion adsorption. The mobility behavior was found to be the same for each electrolyte indicating no effect from co-ion type [30]. Considering the effect from the Hofmeister series is more pronounced in anions [29, 30], it seems worthwhile to study the influence of counter-ion adsorption on a positively charged protein-solvent interface [30], which is the objective of Aim 1 in this body of work. However, first it is necessary to introduce two analysis methods.

What is the hydrodynamic radius?

In this work, two types of radii are analyzed to assess the effect of different perturbations on the hydration layer: the Stokes-Einstein hydrodynamic radius and the electrophoretic radius. Theoretically, the Stokes-Einstein hydrodynamic radius (R_h) was

derived as the radius of an uncharged sphere (or equivalently, a solvated sphere obeying the "no slip" boundary condition) undergoing diffusive motion [41]. The X_{SP} and R_h differ in their theoretical definitions, with the X_{SP} being the position of the ζ during electrokinetic phenomena (e.g. electrophoresis) [1] and the R_h being a radius pertaining to the edge of solvation during diffusion [41], defined by the Stokes-Einstein equation (**Eq. 0-1**).

$$R_h = \frac{k_b T}{6\pi\eta D} \quad \textbf{(0-1)}$$

where k_b is the **Boltzmann constant**, T is the absolute temperature, η is the pure solvent viscosity and D is the single particle/protein diffusivity

Einstein originally derived **Eq. 0-1** from the Navier-Stokes equation (see **APPENDIX F**) for dilute, non-charged spheres [41]. However, the proteins of this work are charged, and it is necessary to identify the effect that bearing a charge has on the R_h. At low ion concentrations, diffusivity shows marked increases [42, 43, 44, 45] that lead to a R_h being smaller than the physical size of the molecule itself – a hyper-diffusive regime (see **Fig. 3-1**). This increase in diffusivity is believed to result from long-range charge repulsion that accelerates diffusion as the κ^{-1} increases [46]. In order to use experimental R_h appropriately, it is necessary to establish the Stokes-Einstein regime – a range of ionic strengths where **Eq. 0-1** is valid for a charge-bearing particle. This is done through a combined analysis of the R_h and the other radius of this work, the electrophoretic radius.

The electrophoretic radius (R_e) is the hydrodynamic radius during electrophoresis. **Eq. 0-2a** shows the relation between the R_e, protein radius (R_p) and X_{SP}. Henry derived an equation for electrophoretic mobility (u_e) accounting for the electrophoretic effect,

called electrophoretic retardation (see **Ch. I** for definitions of all electrophoretic effects), from the Poisson-Boltzmann and Navier-Stokes equations while assuming the ionic atmosphere surrounding the charged particle to remain in its equilibrium state [47] (**Eq. 0-2b**). Henry's equation has been experimentally tested on nanometer to micron-scale polystyrene, gamboge and silica spheres [48, 49]. **Equation 0-2c** expresses this relationship in terms of the R_e as a function of u_e, the net valence of the protein (Q), and the pure solvent viscosity (η) [50, 51]. Q is determined from controlling the solution pH and knowing the pKa values of the charged surface residues [52, 53, 54, 55]. The pure solvent viscosity (η) can be measured by a rheometer (see **APPENDIX C**; **Fig. C2**); however, much data already exists on the viscosity of aqueous electrolyte solutions [56] and thus empirical models can be used [56, 57, 58, 59, 60]. The Henry correction factor for electrophoretic retardation on a sphere ($f_s(\kappa R_e)$) varies between 1 and 1.5 and is formally defined in **Ch. I** (see **Eq. 1-2a**). However, in this work the approximation shown in **Eq. 0-2d** [61] was used for globular proteins (lysozyme, bovine serum albumin, β-lactoglobulin, and green fluorescent protein) and **Eq. 0-2f** [62] was used in **Eq. 0-2e** for randomly oriented cylindrical proteins ($[(PPG)_{10}]_3$).

$$R_e = R_p + X_{SP} \qquad \textbf{(0-2a)}$$

Sphere:
$$u_e = \frac{2\varepsilon_o\varepsilon_r\zeta f_s(\kappa R_e)}{3\eta\left(\frac{f}{f_o}\right)} \qquad \textbf{(0-2b)}$$

$$R_e = \frac{Qef(\kappa R_e)}{6\pi\eta u_e(1+\kappa R_e)\left(\frac{f}{f_o}\right)} \qquad \textbf{(0-2c)}$$

Sphere:
$$f_s(\kappa R_e) \approx 1 + \frac{1}{2\left(1+\frac{\delta}{\kappa R_e}\right)^3},\ \delta = \frac{2.5}{1+2e^{-\kappa R_e}} \qquad \textbf{(0-2d)}$$

Cylinder:
$$u_e = \frac{1}{3}u_{e,\parallel} + \frac{2}{3}u_{e,\perp} = \frac{\varepsilon_o\varepsilon_r\zeta}{3\eta}\left(1 + 2f_c(\kappa R_e)\right) \qquad \textbf{(0-2e)}$$

Cylinder: $$f_c(\kappa R_e) \approx \frac{1}{2}\left[1 + \frac{1}{\left(1+\frac{\delta}{\kappa R_e}\right)^2}\right], \delta = \frac{2.55}{1+e^{-\kappa R_e}} \quad \textbf{(0-2f)}$$

where R_p is the protein radius (see **Eq. 2-1**), X_{SP} is the slip plane position, u_e is electrophoretic mobility, ε_o is **vacuum permittivity**, ε_r is the relative dielectric of the solution, ζ is the zeta potential, η is the pure solvent viscosity (i.e. no protein), Q is the protein net valence, e is the **charge on an electron**, κ is the inverse Debye length (see **APPENDIX B**), $\left(\frac{f}{f_o}\right)$ is a shape factor (1.17 for lysozyme, 1.29 for bovine serum albumin, and 1.18 for β-Lactoglobulin [50]), $u_{e,\parallel}$ is the electrophoretic mobility of a cylinder with a parallel orientation to the electric field and $u_{e,\perp}$ is the mobility of a cylinder with a perpendicular orientation

The idea that the X_{SP} coincides with R_h has been previously considered [63, 64, 65], but to my knowledge, has not been experimentally validated for proteins. Thus, the experiments in this work compare experimentally determined R_e and R_h values to assess the hydration during electrophoresis and diffusion. As will be shown later (see **Fig. 3-5**), only R_e values are applicable for assessing hydration at all ion concentrations. Nonetheless, the common coincidence of these two radii in this work strongly supports the claim that hydration remains the same during diffusion and electrophoresis.

Is hydration the same during diffusion and electrophoresis?

This question is addressed by the central hypothesis of this thesis. As previously discussed, hydration is dependent on a number of factors, one of which being the flow of solvent relative to the molecule [14]. Diffusion involves both rotational and translational motions; however, for a globular protein, rotation is significantly slower (on the order of

10^{-7} seconds [14]) than translation which is a necessary criterion for the accurate analysis of dynamic light scattering measurements [66, p. 248]. In electrophoresis, translational motion is dominant as a charged molecule is accelerated towards an oppositely charged electrode. This means globular proteins are expected to experience predominately the same type of flow during diffusion and electrophoresis. Thus, if the solution conditions are the same during diffusive and electrophoretic measurements, hydration should be equivalent with any deviation representing the difference in flow perturbations at the protein-solvent interface (i.e. the slip plane position*).* The experimental hydration layer thickness during diffusion (Δ_D) defined by single particle translational diffusivity is given below [17]:

$$\Delta_D = \frac{k_b T}{6\pi\eta D} - R_p \qquad \textbf{(0-3)}$$

Similarly for electrophoresis, the experimental hydration layer thickness during electrophoresis ($\boldsymbol{\Delta_E}$) can be defined using the Henry equation as shown.

$$\Delta_E = \frac{Qef(\kappa R_e)}{6\pi\eta u_e(1+\kappa R_e)\left(\frac{f}{f_o}\right)} - R_p \qquad \textbf{(0-4)}$$

It is important to note, the light scattering methods used in this work are lower resolution relative to other methods for studying hydration [14]. However, this work holds the advantage of knowing the protein structures. This allows an excellent opportunity for EDL modeling as the protein charged surfaces are known [25], and changes in conformation can be studied experimentally or through simulation [67]. An obstacle to using atomic structure of proteins to estimate ζ is the lack of general criteria for the location of the X_{SP} [1, 64, 65]. Other studies [6] have used the EDL edge defined by the **Debye length** (κ^{-1}) as the X_{SP} for calculating the ζ. However, the κ^{-1} is typically used to represent the EDL thickness, and thus likely resulted in an underestimate of their

calculated ζ values. As will be seen by the collection of evidence in this body of work, a more accurate placement of the X_{SP} is the radius of hydration (**Fig. 0-1**).

CHAPTER I

ZPRED: Zeta Potential Prediction in a General Aqueous Electrolyte Solution

Chapter Abstract

This chapter covers electrophoresis theory, and the zeta potential prediction protocol (ZPRED) in qualitative detail. The protocol has been written for a general molecular structure, and it works by modeling an EDL over the molecular structure (e.g. the modified Gouy-Chapman EDL shown in **Fig. 0-1**) and then averages the electric potentials at an estimated slip plane position within the EDL to compute the zeta potential.

COMPUTATIONAL METHODS OF ZPRED

Theoretically, the zeta, or electrokinetic, potential (ζ) is the electric potential governing all electrokinetic phenomena [64]. However, this thesis only covers electrophoresis due to the availability of a Zetasizer Nano ZS (Malvern Instruments, Worcestershire, U.K.), which performed light scattering analysis of electrophoretic measurements (see **Ch. II** for experimental details). Electrophoresis has been well-studied for about a century [47, 69, 70, 71, 72, 73, 74, 61, 62, 75] [76, 77] and is one of the oldest methods for determining the ζ of a particle in solution. More recently, work has been done to calculate the electrophoretic mobility (the measurement of electrophoresis) of a protein from its structure [65, 63, 78, 79]. These boundary element method based approaches are similar to ZPRED in that they can determine the ζ of a structure through conversion of the modeled mobility value. However, ZPRED is an improvement over the others as it directly determines the ζ through EDL modeling, which should be applicable to all electrokinetic phenomena. ZPRED is applied to a molecular structure through six primary steps shown in **Fig. 1-1**.

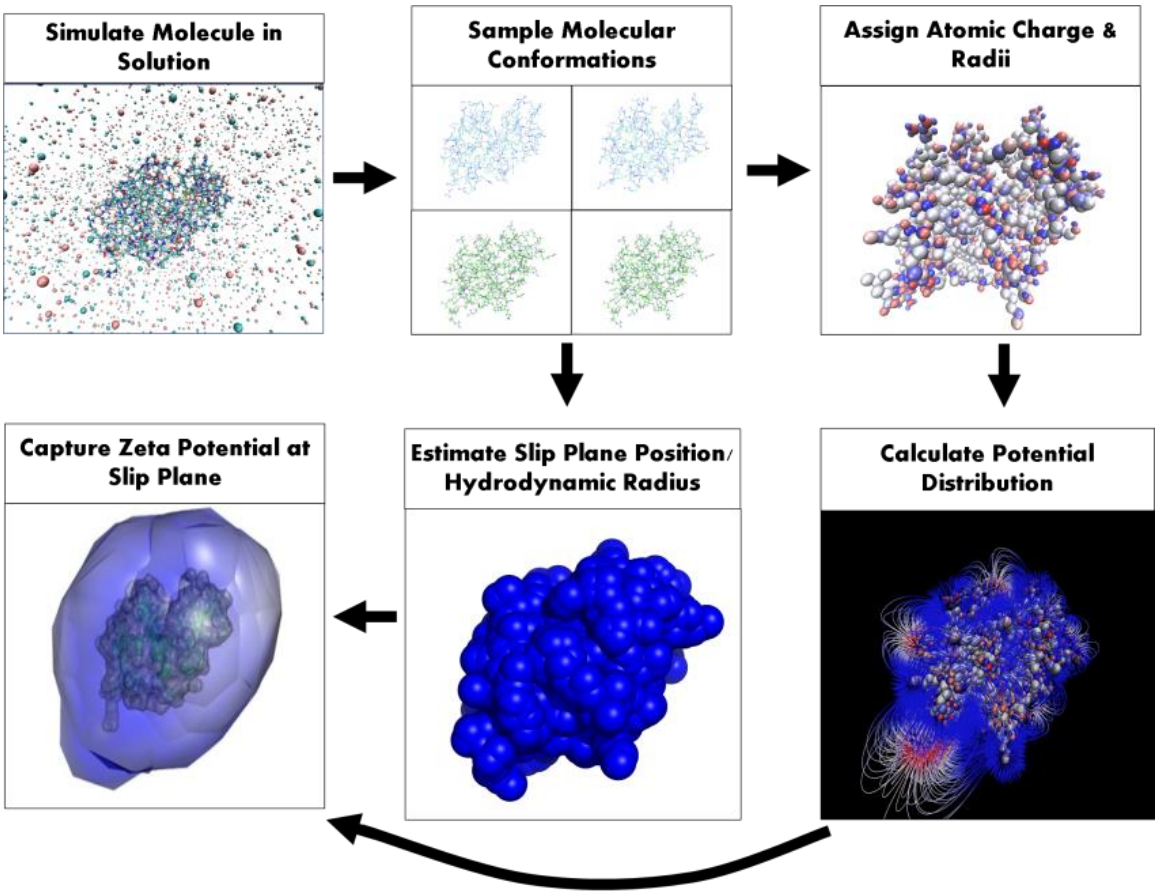

Figure 1-1. Flow Diagram of ZPRED Outlining the Steps of Computation of the Zeta Potential of a Molecule. In the diagram, computation is applied to the structure of hen egg white lysozyme (PDB id: 6lyz).

The six steps of ZPRED are:

1) sample molecular conformations of the molecular structure

2) estimate the slip plane position of each conformation

3) assign atomic charges and radii to each conformation

4) calculate electric potentials from each conformation propagating into solvent

5) average electric potentials at the estimated slip plane to calculate the zeta potential for each conformation

6) calculate the zeta potential of the molecule by averaging zeta potentials from the different conformations

1) Sample Molecular Conformations

The first step uses molecular dynamics software (e.g. Amber 2015 [67]) to simulate the structural motions of the molecule in solvent. In general, this step is comprised of four parts listed below:

1a) prepare the molecular structure for a molecular dynamics simulation

1b) energetically minimize the molecular structure

1c) thermally excite the molecular structure

1d) simulate the molecular structure in explicit solvent and sample conformations at a structural steady-state

In step 1a, atomic coordinates of a molecular structure (e.g. a crystal structure from the protein data bank (PDB) [80]) are prepared by removing any water molecules (e.g. using the Amber tool, pdb4amber [67, p. 193]) and protonating the structure (e.g. using the Amber tool, reduce [67, p. 196]). Prepared structures are loaded into a molecular dynamics simulation (e.g. as an UNIT object manipulated by the Amber program, teLeap [67, p. 197]) specifying force field [81] parameters and generating initial topology and coordinates of the atoms of the prepared structure in a specified volume of solvent molecules. In general, globular (spherical) molecules are housed in water boxes extending 20 Å from the molecular surface and fibrillar (cylindrical) molecules are housed in water boxes extending 30 Å away. Step 1b takes the generated topology and coordinate files and performs a molecular dynamics simulation (e.g. using either sander [67, p. 296] or pmemd [67, p. 345] of Amber) to energetically minimize the structure, which optimizes its geometry in solution. The coordinates of the optimized structure provide a starting point for the simulation of Step 1c. This step gradually heats the

structure from 0 K to a specified temperature, inducing thermal motion of the solvent and the molecule. Step 1d is the main molecular dynamics simulation and uses the coordinates of the prepared heated structure as input. This simulation is run until the structure reaches a steady-state based on the root mean squared displacement of the primary structural chain (e.g. a protein backbone). Structural steady-state should be assessed after centering the entire trajectory of the solvent and atomic coordinates around the molecule's center of mass (e.g. using the Amber tool, cpptraj [67, p. 517]). Once a structural steady-state is reached, the molecule should switch between a limited number of molecular conformations, which are sampled based on the variation in the root mean squared displacement. For consistency in this thesis, simulations were run for 100 nanoseconds and twenty conformations were sampled from the last 20 nanoseconds, one per each nanosecond. Selected conformation coordinates can then converted into an appropriate file format for hydrodynamic and electrostatic calculations (e.g. the PDB format using the Amber tool, ambpdb [67, p. 515], with the bres flag).

2) Estimate the Slip Plane Position

In the second step, the position of the slip plane relative to the molecular surface must be either determined from experimental data or estimated computationally. As shown in this work, the Stokes-Einstein hydrodynamic radius (R_h) determined from measured diffusivities, and the electrophoretic radius (R_e) determined from measured electrophoretic mobilities provide reasonable representations of the slip plane position. ZPRED assumes hydration is distributed based on the "uniform hydration layer model" [17]. Thus, the slip plane position (X_{SP}) can be estimated experimentally by subtracting

the molecular radius from a measured solvated radius, which should always be greater than or equal to the molecular radius [17]. It is important to note the Stokes-Einstein equation (**Eq. 0-1**) is limited to relatively high salt concentrations [45, 42, 43, 82], and thus, other methods for determining molecular size must be used, such as the R_e determined from electrophoretic mobility measurements (**Eq. 0-2c**).

Estimating the X_{SP} computationally requires estimating the molecular radius and R_h. For globular molecules, the molecular radius can be calculated as the average distance between the center of mass and the solvent-excluded surface of the structure under assessment (**Eq. 2-1**). As shown in **Eq. 0-1**, R_h depends on temperature, which is controlled by the user; leaving pure solvent viscosity and single particle/protein diffusivity to be defined. A number of empirical relationships have been developed for the pure solvent viscosity of many different salt solutions at varying temperatures and can be obtained from the literature [56] (see **APPENDIX C**). If values cannot be found, the viscosity of pure water (**Eq. C3** in **APPENDIX C**) can be used as an estimate since added salt only affects viscosity at higher salt concentrations. Single particle diffusivity can be computed with available hydrodynamics software (e.g. HYDROPRO [83] or HullRad [84]). HYDROPRO requires the protein structure, its specific volume, its molecular weight, temperature, pure solvent viscosity and pure solvent density as inputs. Molecular weight is computed from the summation of the molecular weights of residues present. Specific volume of each structure requires calculating the volume of the structure (using MSMS [85]) and then dividing by the mass of structure. Pure solvent density is estimated by an empirical model [86] for various aqueous electrolyte solutions. If values cannot be found, the density of pure water (**Eq. C13b** in **APPENDIX C**) can be used as

an estimate. Once viscosity and diffusivity values are obtained, the R_h can be calculated by **Eq. 0-1**. Alternatively, R_e can be used instead if the electrophoretic mobility is easier to acquire than the diffusivity. Computation of the electrophoretic mobility from a PDB structure has been done by Allison [63]; however, he was unwilling to share his code for his method. Anyway, just like experimental values, the estimated X_{SP} is calculated by subtracting the molecular radius from the estimated R_h or R_e. Once a X_{SP} is determined, it is stored for later use in the fifth step of ZPRED.

3) Assign Atomic Charges and Radii

The protocol's third step assigns a charge and radius to each atomic coordinate of the molecular structure using available software. For example, the software, PDB2PQR [54], converts a PDB formatted coordinates file into a PQR formatted file for use in electrostatics software. This involves checking the integrity of the structure (e.g. whether heavy atoms are missing or not) and then protonating it based on a pKa predictor (e.g. PROPKA [55]) at a specified pH. It's worth noting, PROPKA has proven to be a more accurate method for pKa prediction among other pKa predictors (MCCE, MEAD, and UHBD) [87]. Following protonation, the position of hydrogens are determined by Monte Carlo optimization based on the global H-bonding network of the structure considering charge residue side chains and water-molecule interactions. Once atomic charges and radii have been assigned, the structure is ready for electrostatic calculations.

4) Calculate Electric Potential Distribution

In the fourth step, the molecule's distribution of electrostatic potentials is computed by solving the Poisson-Boltzmann equation (PBE) (see **Eq. B4** in **APPENDIX B**) over the structure (e.g. using the adaptive Poisson-Boltzmann solver [88]). This can involve use of an adaptive finite element method, which solves the PBE by iteratively adjusting the discretization of subsections of the problem domain. Subsections are allocated based on the error predicted from larger encompassing subsections initially starting with the entire problem domain. To solve the PBE, the problem domain is divided into two regions of different dielectrics: the molecule (e.g. proteins hold a dielectric from 2 to 4 [89]) and solvent (dielectric based on solvent and temperature [89], see **Eq. C1** in **APPENDIX C**). The two regions are separated by a solvent-accessible surface generated over the molecular structure using the largest ion in the solvent. Thus, ionic radii are needed as an input (see **APPENDIX C**). This step of ZPRED connects EDL theory and application as the PBE models the diffuse region of the EDL [1, p. 44]. By solving either the complete non-linear PBE (**Eq. B4**) or the linear version (**Eq. B6**) over the molecular structure, a Gouy-Chapman EDL model encompassing the molecule is generated. To model specific ion effects, a Gouy-Chapman-Stern EDL model should be used. Generating a Gouy-Chapman-Stern EDL requires modifying the molecular surface to include a stagnant layer of specifically sized ions holding some dielectric and then solving the PBE from the stagnant layer into the diffuse region of the EDL holding a different dielectric. This is referred to as the Stern-layer-modified PBE or the size-modified PBE. Once the electric potentials are generated in the EDL model, it is now time to capture the potentials at the slip plane to define the zeta potential.

5) Average Electric Potentials at Slip Plane

The fifth step of ZPRED involves generating a solvent-excluded surface (SES) over the molecular structure (e.g. using MSMS [85]). The SES generated is composed of Cartesian coordinates and their normal vectors directed away from the molecular surface. The SES is inflated to the X_{SP} by translating its initial coordinates along their respective normal vector by the estimated slip plane distance calculated in the second step. Then the calculated electric potentials from the fourth step at the inflated coordinates are captured (e.g. using the APBS tool, multivalue). Using multivalue requires converting the inflated coordinates into a comma separated vector (CSV) file format, which is simply done by writing each coordinate on its own line and delimiting by commas in a text file. A zeta potential value for each conformation is computed by averaging the captured potentials at the inflated SES.

6) Calculate the Zeta Potential of a Molecule

The sixth step completes the zeta potential prediction protocol by averaging the zeta potentials determined from each conformation. The resulting zeta potential value represents what would be expected from the molecular structure in solution. In order to compare predicted zeta potential values to experimental values, an appropriate electrokinetic model must be used.

EXPERIMENTAL VALIDATION OF ZPRED.

The ζ is not directly measurable, but must be determined by an electrokinetic model relating it to a measurable quantity, such as the electrophoretic mobility of electrophoresis. The oldest method for getting at the ζ is electrophoresis; however, conversion of measured electrophoretic mobilities into a ζ value can be complicated depending on the effective forces acting on the EDL when an electric field is perturbing it. Electrokinetic models for converting an electrophoretic mobility (u_e) into a ζ are shown in **Fig. 1-2**, and each account for different electrophoretic effects, which arise under different solution conditions. In **Fig. 1-2**, the dimensionless electrophoretic mobility (**Eq. 1-1**) is plotted against the dimensionless electrokinetic radius (hydrodynamic radius divided by **Debye length**) to map the landscape, in which different effects arise.

$$E_m = \frac{3\eta e|u_e|}{2\varepsilon_r \varepsilon_o k_b T} \qquad \textbf{(1-1)}$$

where η is the pure solvent viscosity, u_e is electrophoretic mobility, and the other terms hold their usual significance (see **APPENDIX A** for details)

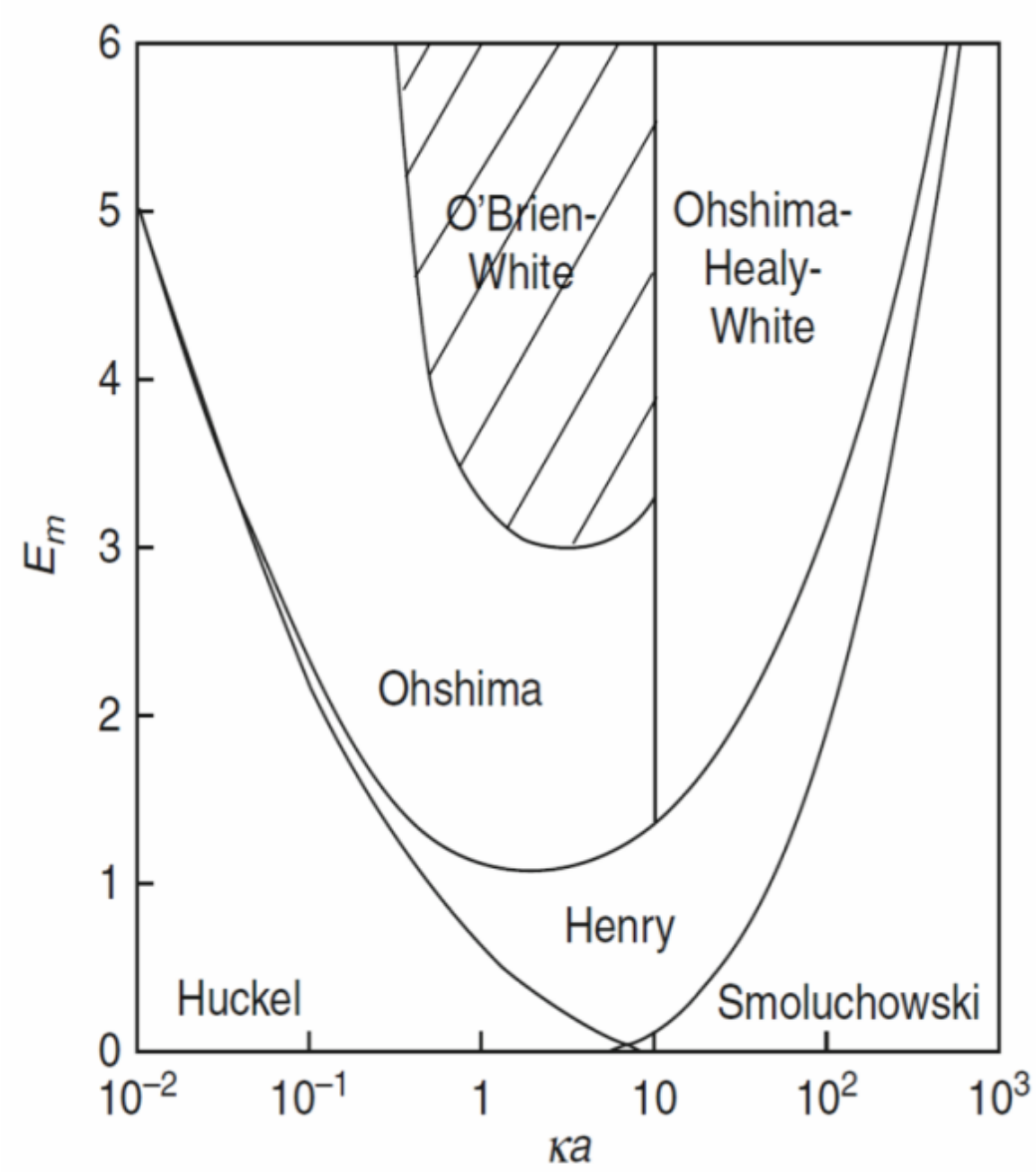

Figure 1-2. Appropriate Selection of Electrophoretic Mobility - Zeta Potential Relations (adapted from [90, p. 1431]**)**

Electrophoretic Retardation

Electrophoretic retardation is one of the oldest effects to be modeled and is a viscous shear stress passed to the molecular surface from oppositely moving counter-ions in the diffuse layer, which hinders electrophoretic motion. This effect becomes more

pronounced as ion concentration increases. As shown in **Fig. 1-2**, the Huckel equation (defined below with $f_{ER} = 1$) accounts for the case of no electrophoretic retardation, and the Smoluchowski equation accounts for electrophoretic retardation at its maximum effect (defined below with $f_{ER} = \frac{3}{2}$). The Henry equation [47] accounts for the transition between no and maximum electrophoretic retardation with his electrophoretic retardation correction factor (f_{ER}) formally defined in **Eq. 1-2a**. However, a few accurate approximations have been developed [61, 62, 75] and are typically used.

$$u_e = \frac{2\varepsilon_r\varepsilon_o\zeta f_{ER}}{3\eta} \quad \textbf{(1-2)}$$

$$f_{ER} = \frac{3}{2}(1 - e^{\kappa a}[5E_7(\kappa a) - 2E_5(\kappa a)]) \quad \textbf{(1-2a)}$$

where κ is inverse Debye length (see **APPENDIX B**) and E_n is the n-th order exponential integral (see **APPENDIX A** for definition and modeling approximation)

Relaxation Effect

Another effect present predominate with highly charged molecules is the relaxation effect. This effect refers to the distortion and effective polarization of the EDL that slightly neutralizes the electrokinetic charge reducing its attractive propulsion in the electric field, and thus hindering electrophoretic motion. The Ohshima approximation [77] for Overbeek's expression for symmetrical electrolytes (see **Eq. 1-3**) accounts for the case of combined electrophoretic retardation and relaxation (see region of **Fig. 1-2**).

$$u_e = \frac{2\varepsilon_r\varepsilon_o\zeta f}{3\eta} \quad \textbf{(1-3)}$$

$$f = f_{ER} - \left(\frac{ze\zeta}{k_bT}\right)^2 \left[f_3 + \left(\frac{m_+ + m_-}{2}\right) f_4\right] \quad \textbf{(1-3a)}$$

$$f_3 = \frac{\kappa a\left(\kappa a + 1.3e^{(-0.18\kappa a)} + 2.5\right)}{2\left(\kappa a + 1.2e^{(-7.4\kappa a)} + 4.8\right)^3} \quad \textbf{(1-3b)}$$

$$f_4 = \frac{9\kappa a\left(\kappa a + 5.2e^{(-3.9\kappa a)} + 5.6\right)}{8\left(\kappa a - 1.55e^{(-0.32\kappa a)} + 6.02\right)^3} \quad \textbf{(1-3c)}$$

$$m_{\pm} = \frac{2\varepsilon_r\varepsilon_o k_b T}{3\eta z^2 e^2}\lambda_{\pm} \quad \textbf{(1-3d)}$$

where $\lambda_{\pm}$ is the ionic drag coefficient of cations and anions, which can be defined by either their limiting conductivities [77] or their ionic radii ($\lambda_{\pm} = 6\pi\eta R_i$, where R_i is ionic radius (see **APPENDIX C** for values))

Surface Conductance

Another effect that can arise with high ion concentrations is surface conductance in the diffuse layer. This effect refers to the excessive conductivity (relative to bulk solution) resulting from ion motion in the diffuse layer that distorts the applied electric field near the protein surface. The combined effects of electrophoretic retardation, relaxation and surface conductance can be accurately modeled through solving the standard electrokinetic model, which is a system of coupled partial differential equations (specifically, the Navier-Stokes, Nernst-Planck, Poisson-Boltzmann, and Continuity equations). Ohshima-Healy-White approximation [73] solves the standard electrokinetic model by series expansion approximations and is only applicable at relatively high salt concentrations (see region of **Fig. 1-2**).

$$u_e = \frac{2\varepsilon_r\varepsilon_o\zeta f}{3\eta} \quad \textbf{(1-4)}$$

$$f = 1 - \frac{2AB}{\tilde{\zeta}(1+A)} + \frac{1}{\tilde{\zeta}\kappa a}\{W - X + Y - Z\} \quad \textbf{(1-4a)}$$

$$W = \frac{10A}{1+A}\left(t + \frac{7t^2}{20} + \frac{t^3}{9}\right) - 12C\left(t + \frac{t^3}{9}\right) \quad \textbf{(1-4b)}$$

$$X = 4D\left(1 + \frac{2\varepsilon_r\varepsilon_o k_b T}{\eta e z_{co}^2 |u_{e,co}|}\right)\left[1 - e^{-\left(\frac{\tilde{\zeta}}{2}\right)}\right] \quad \textbf{(1-4c)}$$

$$Y = \frac{8AB}{(1+A)^2} + \frac{6\tilde{\zeta}}{1+A}\left(\frac{2\varepsilon_r\varepsilon_o k_b T D}{3\eta e z_{co}^2|u_{e,co}|} + \frac{2\varepsilon_r\varepsilon_o k_b T B}{3\eta e z_{ctr}^2|u_{e,ctr}|}\right) \quad \textbf{(1-4d)}$$

$$Z = \frac{24A}{1+A}\left(\frac{2\varepsilon_r\varepsilon_o k_b T D^2}{3\eta e z_{co}^2|u_{e,co}|} + \frac{2\varepsilon_r\varepsilon_o k_b T B^2}{3\eta e z_{ctr}^2|u_{e,ctr}|(1+A)}\right) \quad \textbf{(1-4e)}$$

$$A = \frac{2}{\kappa a}\left(1 + \frac{2\varepsilon_r\varepsilon_o k_b T}{\eta e z_{ctr}^2|u_{e,ctr}|}\right)\left[e^{\left(\frac{\tilde{\zeta}}{2}\right)} - 1\right] \quad \textbf{(1-4f)}$$

$$B = \ln\left(1 + e^{\left(\frac{\tilde{\zeta}}{2}\right)}\right) - \ln(2) \quad \textbf{(1-4g)}$$

$$C = 1 - \frac{25}{3(\kappa a+10)} e^{-\left(\frac{\kappa a\tilde{\zeta}}{6(\kappa a+6)}\right)} \quad \textbf{(1-4h)}$$

$$D = \ln\left(1 + e^{-\left(\frac{\tilde{\zeta}}{2}\right)}\right) - \ln(2) \quad \textbf{(1-4i)}$$

$$t = \tanh\left(\frac{\tilde{\zeta}}{4}\right) \quad \textbf{(1-4j)}$$

$$\tilde{\zeta} = \frac{z_{ctr} e|\zeta|}{k_b T} \quad \textbf{(1-4k)}$$

where z_{co} is the co-ion valence, $u_{e,co}$ is the co-ion mobility, z_{ctr} is the counter-ion valence, and $u_{e,ctr}$ is the counter-ion mobility

The O'Brien-White algorithm [72] numerically solves the standard electrokinetic model accounting for all effects described above and is applicable for the entire electrophoretic landscape shown in **Fig. 1-2**. However, it is cumbersome to use, which motivated the development of the analytical approximations shown above. It is worth noting, all the models described here assume a stagnant layer of ions and solvent surround the molecule (in other words, stagnant layer conductance/mobile Stern layer is not considered). Fortunately, for proteins this is not a common occurrence, and in fact, most cases of protein electrophoresis can be interpreted with the Henry equation accounting for electrophoretic retardation alone. Having covered the standard

electrophoretic effects in moderate to high ion concentrations, the last effect to be discussed only occurs in environments of very dilute ion concentrations.

Counter-Ion Condensation

Counter-ion condensation refers to the tight packing of available counter-ions in the hydration layer of a highly charged particle [90, p. 1435]. When the particle charge is low enough to allow the infinitely dilute particle assumption to be applicable, the appropriate electrokinetic model is the Huckel equation (see **Eq. 1-2**). However, if it is too high and induces counter-ion condensation the electrophoretic mobility becomes dependent on the volume fraction and no longer depends on the charge as shown below.

$$u_e = \begin{cases} \frac{2\varepsilon_r\varepsilon_o\zeta}{3\eta}, & Q^* \leq \ln(1/\phi) \\ \frac{2\varepsilon_r\varepsilon_o}{3\eta}\frac{ze}{k_bT}\ln(1/\phi), & Q^* > \ln(1/\phi) \end{cases} \quad \textbf{(1-5)}$$

where $Q^* = \frac{Q}{4\pi\varepsilon_r\varepsilon_o R_h}\left(\frac{ze}{k_bT}\right)$ and is called the scaled particle surface charge, ϕ is the particle volume fraction

CONCLUSIONS

ZPRED models the ζ of a molecular structure and achieves accurate electrophoretic mobility values to compare to experiment through consideration of electrophoretic effects and of the variation in solution properties (relative dielectric, viscosity and density) with solution conditions. Unlike other models [65, 63, 78, 79], ZPRED models an EDL around a structure to determine the ζ, truly defining the potential by its theoretical definition.

CHAPTER II

Controlling Proteins During Light Scattering Measurements

Chapter Abstract

Due to the delicate nature of proteins and the common misusage of the light scattering apparatus (Zetasizer) used in this work, a chapter has been dedicated to the study and performance of good light scattering measurements. This chapter covers necessary considerations and important findings regarding the electrophoretic and diffusive measurement methods used with the Zetasizer Nano ZS. It is important to note the Zetasizer does not measure the size or zeta potential of a molecule directly. Size is determined from diffusivity measurements using the Stokes-Einstein equation (**Eq. 0-1**), and zeta potential is determined from electrophoretic mobility measurements using an electrokinetic model (for the Zetasizer only the Huckel and Helmholtz-Smoluchowski equations are available (see **Eq. 1-2**)). In order to get the accuracy achieved in this work (e.g. see **Figs. 3-1** and **3-3**), it is essential to perform analysis on the measured values of the Zetasizer and not its converted values. In addition to covering the experimental protocols used in this work, a detailed description of each protein studied and the reason for its selection is also provided.

INTRODUCTION

In this chapter, methods for the creation of protein colloids that make "good" light scattering solutions are described. A "good" colloid is one that allows for a "good" light scattering measurement, satisfying the necessary experimental assumptions for an accurate analysis. In general for light scattering, the important assumptions include:

(i) a constant and uniform solution temperature

(ii) a protein concentration dilute enough to minimize protein-protein interactions

(iii) a protein concentration concentrated enough to produce a signal at least two and half times greater than pure solvent

(iv) a monodisperse population of stably separated protein structures

(v) a monodisperse population of structurally stable protein structures [66]

For proteins, the last two assumptions are the most difficult to achieve and require carefully designing the solution conditions to allow for both dispersion and structural stability. In order to evade issues with these last two assumptions, this work used well-studied proteins with high melting (i.e. unfolding) temperatures and known aggregation behavior dependent on solution conditions.

In general, "ideal proteins" were selected that behave like "ideal colloids" holding sturdy, spherical structures insensitive to variation in solution conditions and an appropriate dispersion stability allowing for good light scattering measurements. For all the globular proteins considered in this work, both monomeric and dimeric structures were assessed computationally to account for the possibility of dimerization. Twenty conformations sampled from molecular dynamics (described in **Ch. I**) were used for each

protein structure. Structural shape descriptors were calculated for each ensemble of protein structures and are presented in **Tables 2-1, 2-2,** and **2-3**.

For globular proteins, the protein radius (R_p) is defined as the average distance from the surface to the center of the protein structure, where the center coordinates of the protein were defined as the average of the atomic coordinates of the PDB structure under assessment.

$$R_p = \frac{1}{N}\sum_{i=1}^{N}\left(\sqrt{\left(S_{x,i} - C_x\right)^2 + \left(S_{y,i} - C_y\right)^2 + \left(S_{z,i} - C_z\right)^2}\right) \quad \textbf{(2-1)}$$

$$C_x = \frac{1}{M}\sum_{j=1}^{M} x_j \qquad C_y = \frac{1}{M}\sum_{j=1}^{M} y_j \qquad C_z = \frac{1}{M}\sum_{j=1}^{M} z_j$$

where R_p is the computed radius of a protein crystal structure, N is the number of surface coordinates calculated by MSMS [85], $S_{x,y,z,i}$ are the i-th surface Cartesian coordinates calculated by MSMS [85], $C_{x,y,z}$ are the Cartesian coordinates of the center of the protein crystal structure, M is the number of atoms in the protein crystal structure, x_j, y_j, and z_j are the j-th atomic coordinates of the protein crystal structure

The other descriptors (gyration radius (R_g), asphericity (A), and shape parameter (S_p)) were calculated from the gyration tensor (or inertial tensor as it was called in [91]) of the protein crystal structures as defined below.

$$T = \begin{bmatrix} T_{xx} & T_{xy} & T_{xz} \\ T_{yx} & T_{yy} & T_{yz} \\ T_{zx} & T_{yz} & T_{zz} \end{bmatrix} =$$

$$\begin{bmatrix} \left[\frac{1}{M}\sum_{j=1}^{M}(x_j - C_x)(x_j - C_x)\right] & \left[\frac{1}{M}\sum_{j=1}^{M}(x_j - C_x)(y_j - C_y)\right] & \left[\frac{1}{M}\sum_{j=1}^{M}(x_j - C_x)(z_j - C_z)\right] \\ \left[\frac{1}{M}\sum_{j=1}^{M}(y_j - C_y)(x_j - C_x)\right] & \left[\frac{1}{M}\sum_{j=1}^{M}(y_j - C_y)(y_j - C_y)\right] & \left[\frac{1}{M}\sum_{j=1}^{M}(y_j - C_y)(z_j - C_z)\right] \\ \left[\frac{1}{M}\sum_{j=1}^{M}(z_j - C_z)(x_j - C_x)\right] & \left[\frac{1}{M}\sum_{j=1}^{M}(z_j - C_z)(y_j - C_y)\right] & \left[\frac{1}{M}\sum_{j=1}^{M}(z_j - C_z)(z_j - C_z)\right] \end{bmatrix}$$

$$R_g = \sqrt{T_{xx}^2 + T_{yy}^2 + T_{zz}^2} \quad \textbf{(2-2)}$$

$$A = \frac{3}{2}\frac{\left((T_{xx}-\lambda)^2 + \left(T_{yy}-\lambda\right)^2 + (T_{zz}-\lambda)^2\right)}{\left(T_{xx}+T_{yy}+T_{zz}\right)^2} \quad \textbf{(2-3)}$$

$$S_p = 27\frac{(T_{xx}-\lambda)(T_{yy}-\lambda)(T_{zz}-\lambda)}{\left(T_{xx}+T_{yy}+T_{zz}\right)^3} \quad \textbf{(2-4)}$$

where $\lambda = \frac{T_{xx}+T_{yy}+T_{zz}}{3}$

The gyration radius (R_g) provides a measure of the size of the structure based on its distribution of mass from its center. Asphericity (A) ranges from zero (for a perfect sphere) to one (for a perfect cylinder) and is measure of the crystal structure's deviation from a sphere. The shape parameter (S_p) ranges from -0.25 to 2 and provides a measure of the ellipticity of the crystal structure. Negative values indicate the structure is similar to an oblate ellipsoid (i.e. disc), positive values indicate the structure is similar to a prolate ellipsoid (i.e. football), and zero is indicative of a perfect sphere. More detail on the assessment of shape can be found elsewhere [91].

Based on the values in **Tables 2-1**, **2-2**, and **2-3**, monomers of each protein can be treated as spheres. This supports the method employed for the experimental estimation of the hydration layer thickness (i.e. the X_{SP}) being the difference between the measured solvated radius and the protein radius. In addition to their spherical shapes, each protein held other features making it attractive to this work.

PROTEIN SELECTION

Hen Egg White Lysozyme (LYZ)

LYZ is a well-studied protein [63, 42, 92, 15, 52, 93] that holds a melting temperature of about 70°C [94] and, based on its hydrogen ion titration curve (**Fig. 2-1**), an isoelectric point of about pH 11.2 [52]. It is known to make tetragonal crystal lattices in solution [95] that remain structurally intact along with its monomeric structure [96]. This made filtering the protein a bit more of a challenge relative to the others. However, once filtered (about 6 hours for 5mL at about 10 $^{mg}/_{mL}$ through a 20 nm filter), lysozyme can make a good colloid (i.e. a light scattering solution) for about 24 hours. As a structure, the lysozyme monomer (PDB id: 6lyz [97]) is highly spherical holding asphericity and shape parameter values indicative that the molecule can be represented by a sphere (see **Table 2-1** (0 is a perfect sphere for both values)) [91]. This supports the use of the uniform hydration layer model [17]. Also, the protein is positive at neutral pH holding a charge that is independent of ion concentration [52], which allows the surface charge distribution to provide a comparable EDL foundation for the different ionic strengths [30]. This is why LYZ is commonly used for the assessing different salts in solution [30, 98]. Studies on the diffusion [42, 99], electrophoretic mobility [100] and hydration [15] of LYZ have been performed. These studies in addition to others [52, 101] show LYZ to be an excellent candidate for the studies of this work. Values for the LYZ shape descriptors are shown in **Table 2-1** and are in agreement with experimental findings [92].

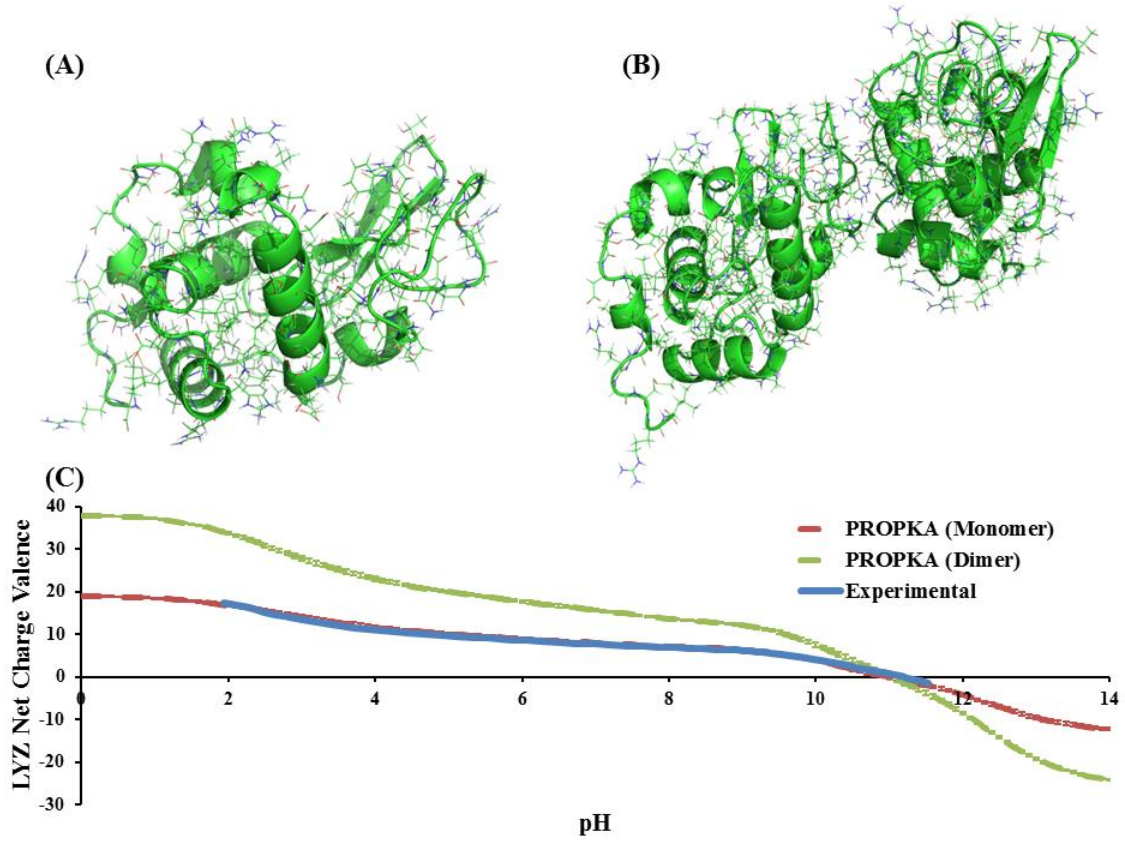

Figure 2-1. Structures of the Hen Egg White Lysozyme (LYZ) Monomer (**A**; PDB id: 6lyz) **and Dimer** (**B**; PDB id: 4r0f) **and Their Hydrogen Ion Titration Curves (C).** Experimental values came from [52]. Structures were obtained from the PDB [80] and their charges were calculated using PROPKA 3.0 [55] on the ensemble of structures sampled from molecular dynamics.

	LYZ Monomer (6lyz)	LYZ Dimer (4r0f)
Protein Radius (Å)	16.496 ± 0.181	22.182 ± 0.259
Asphericity	0.051443 ± 0.0305	0.11523 ± 0.0744
Shape Parameter	0.019561 ± 0.0240	0.051934 ± 0.0949
Gyration Radius (Å)	14.259 ± 0.130	21.081 ± 0.205

Table 2-1. Calculated Shape Descriptors for the Hen Egg White Lysozyme (LYZ) Monomer and Dimer

Bovine Serum Albumin (BSA)

One of the most commonly studied proteins [102, 103, 104, 105, 106, 107, 108], BSA holds a melting temperature of about 69°C [109] and, based on its hydrogen ion titration (**Fig. 2-2**), holds an isoelectric point around pH 5.4 [103]. Its structural stability as a function of ion concentration [110, 111, 112] and pH [113] make it a good structure for assessment under variation of these solution conditions. In addition to the bovine variant, horse serum albumin has also been studied electrophoretically [51]. Calculated shape descriptors of BSA are shown below in **Table 2-2** and are in agreement with previous work [114, 107].

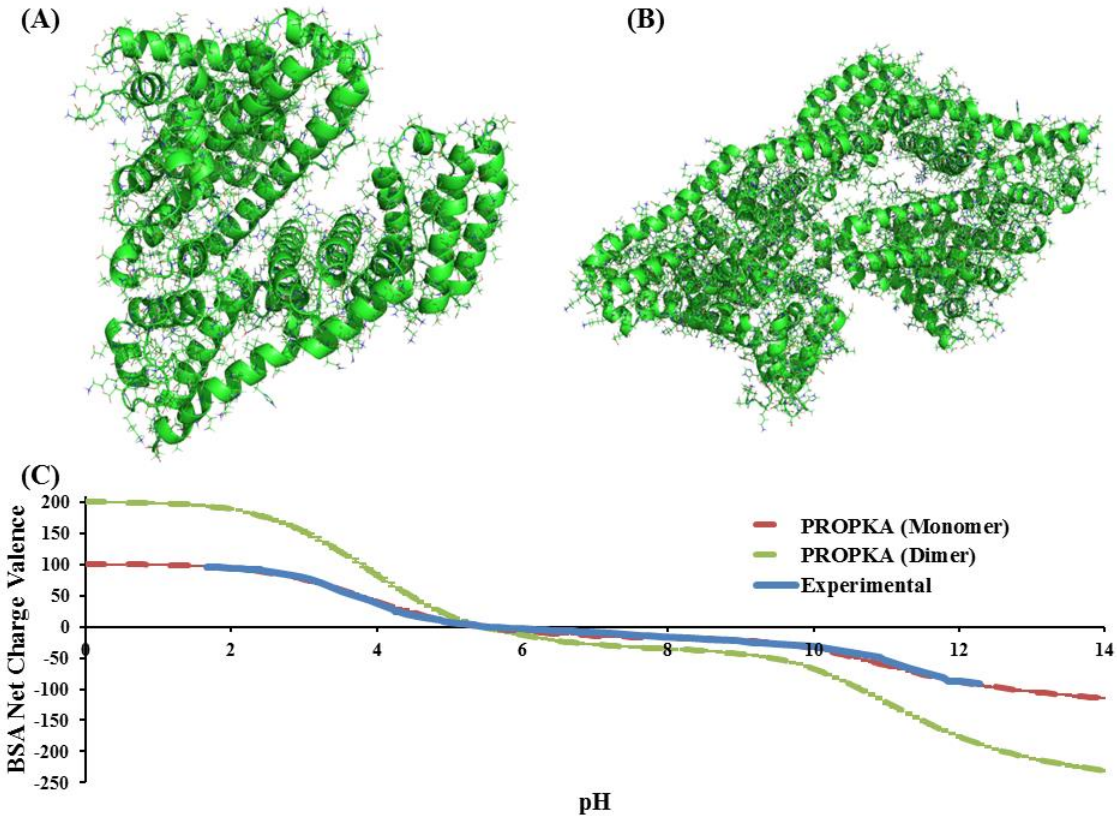

Figure 2-2. Structures of Bovine Serum Albumin (BSA) Monomer (**A**; PDB id: 4f5s chain A) **and Dimer** (**B**; PDB id: 4f5s) **and Their Hydrogen Ion Titration Curves (C)**. Experimental values came from [103]. Crystal structures were obtained from the PDB [80] and their charges were calculated using PROPKA 3.0 [55] on the ensemble of structures sampled from molecular dynamics.

	BSA Monomer (4f5s chain A)	BSA Dimer (4f5s)
Protein Radius (Å)	27.378 ± 0.140	37.967 ± 0.552
Asphericity	0.015326 ± 0.00530	0.060112 ± 0.0222
Shape Parameter	-0.0033021 ± 0.00240	0.0028467 ± 0.0192
Gyration Radius (Å)	26.963 ± 0.108	39.403 ± 0.590

Table 2-2. Calculated Shape Descriptors for the Bovine Serum Albumin (BSA) Monomer and Dimer

Bovine β-Lactoglobulin AB (BLG)

Many relevant studies have been done on BLG [115, 116, 117, 118, 119, 120, 121, 122]. It is a highly stable structure with a melting temperature of 63°C [123] and an isoelectric point around pH 5.4 [116] as shown in **Fig. 2-3**. Studies have been performed on its structural stability [113], electrophoretic mobility [124] and diffusion [119]. Its calculated shape descriptors are in the **Table 2-3**.

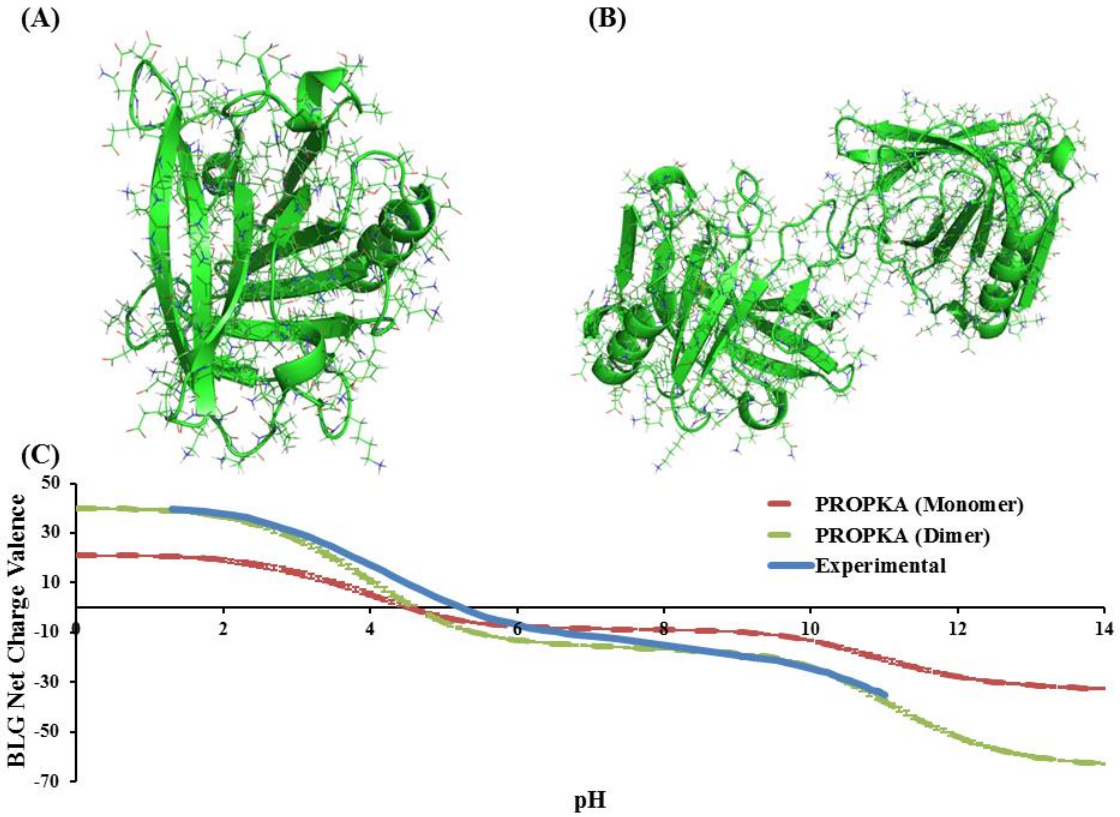

Figure 2-3. Structures of Bovine β-Lactoglobulin (BLG) Monomer (**A**; PDB id: 3blg) **and Dimer** (**B**; PDB id: 1beb) **and Their Hydrogen Ion Titrations Curves (C).** Experimental values came from [116]. Structures were obtained from the PDB [80] and their charges were calculated using PROPKA 3.0 [55] on the ensemble of structures sampled from molecular dynamics.

	BLG Monomer (3blg)	BLG Dimer (1beb)
Protein Radius (Å)	17.611 ± 0.165	25.888 ± 0.441
Asphericity	0.0032574 ± 0.00149	0.29151 ± 0.00599
Shape Parameter	0.00016353 ± 0.000241	0.30500 ± 0.0156
Gyration Radius (Å)	15.083 ± 0.0478	26.0959 ± 0.431

Table 2-3. Calculated Shape Descriptors for the Bovine β-Lactoglobulin (BLG) Monomer and Dimer

MATERIALS AND METHODS

Salt-free LYZ was obtained from Worthington Biochemical Corporation, Lakewood, NJ (LYSF). BSA was obtained from Sigma-Aldrich (CAS: 9048-46-8). BLG from bovine milk was ordered from Sigma Aldrich (CAS: 9045-23-2). Before designing experiments, literature (cited above) on each protein was reviewed to determine solution conditions allowing for monodispersity and structural stability. This work employed light scattering methods to assess hydration of the different proteins in a wide range of solution conditions. Notes on each method and procedures to apply them are below.

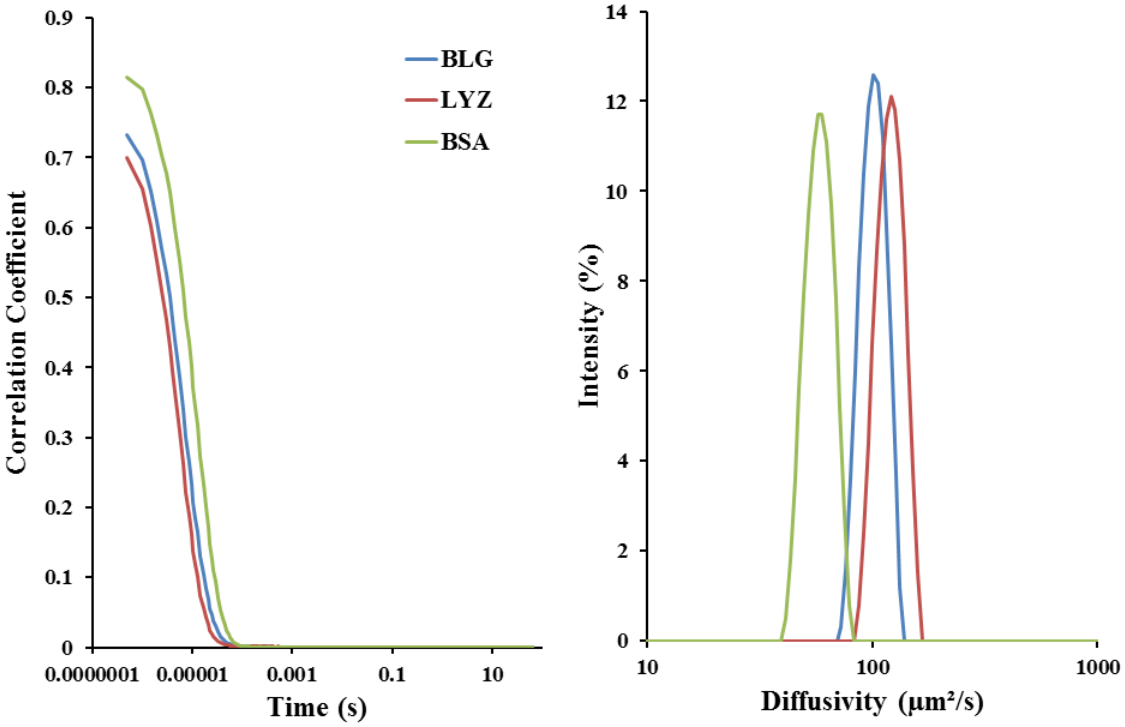

Figure 2-4. DLS Diffusivity Measurements are Based on the Measured Decay Rate of the Autocorrelation Function of the Protein Motion with Itself. In general, larger particles move slower than smaller particles, and thus larger particles hold slower decay rates. This can be seen above (on the left) considering bovine serum albumin (BSA) is the largest, followed by bovine β-Lactoglobulin (BLG), and then hen egg white lysozyme (LYZ) is the smallest. Diffusion data above was taken at the same temperature (25°C), same salt concentration (0.1 M NaCl) and relatively low protein concentrations (<5 mg/mL) to allow for comparison between proteins.

Dynamic Light Scattering (DLS)

DLS measures the intensity of light scattered by a molecule in solution and autocorrelates the measurements over time with the initial measurement. As Brownian motion of molecules is random, the autocorrelation inevitably decays to zero over time. The decay rate of the autocorrelation holds physical significance as the reciprocal of the time scale for the diffusion of a molecule in a specific solution (see **Fig. 2-4**). As diffusion involves both translational and rotational motions, two decay rates can occur if

both motions are significant (e.g. in cylindrical proteins [66, p. 248]). This work avoids this issue through use of globular proteins, which typically only hold significant translational diffusion [66, p. 259]. Decay rates are determined by fitting a polynomial expression to the measured correlogram (autocorrelation plotted against time, **see Fig. 2-4** for example) and converted to a diffusivity value by dividing by the squared magnitude of the scattering wave vector used. Diffusivity dependence on ion concentration, protein concentration, and temperature is shown in **Figs. 2-5**, **2-6**, and **2-7**, respectively. To ensure measurements reflected protein diffusion and not the pure solvent, the sample count rate of a protein solution should be at least 2.5 times greater than the protein-free solution [66, p. 240]. [66, p. Ch. 5]. In this work, the pure solvent was verified to be pure, containing no bacterial contamination post-filtering based on the inability of DLS to generate a correlogram.

$$|G_1| = \sqrt{\frac{G_2 - 1}{\beta}} \quad \textbf{(2-5a)}$$

$$\ln(G_1) = a - bt + \frac{1}{2}ct^2 - \frac{1}{6}dt^3 \quad \textbf{(2-5b)}$$

$$PDI = \frac{c}{b^2} \quad \textbf{(2-5c)}$$

$$D = \frac{b}{K^2} \quad \textbf{(2-5d)}$$

$$K = \left(\frac{4\pi n}{\lambda}\right)\sin\left(\frac{\theta}{2}\right) \quad \textbf{(2-5e)}$$

where β is the instrument efficiency coefficient (related to the measurement signal to noise ratio), a accounts for the y-intercept, b is the first moment and defines the diffusivity (D) (**Eq. 2-5d**), t is time, c is the second moment and defines the polydispersity index (**Eq. 2-5c**) and its square root provides the standard deviation of the diffusivity value, d is the third moment and describes the distribution asymmetry, K is

the wave vector interacting with the particles, n is refractive index, λ is wavelength, and θ is the scattering angle

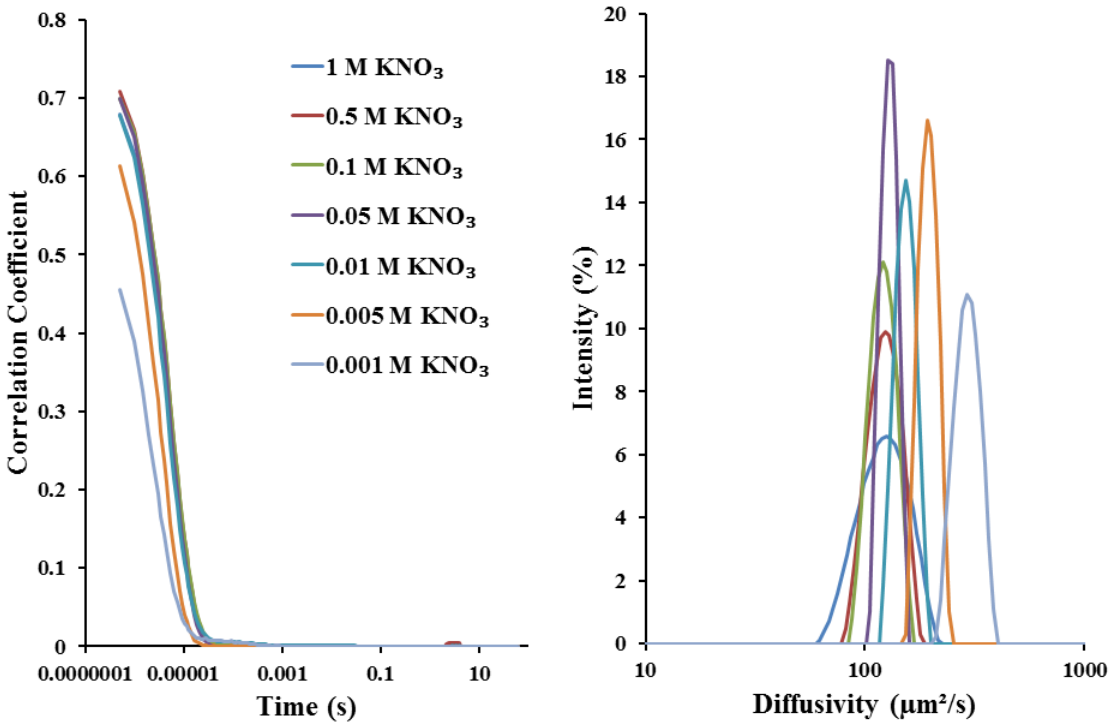

Figure 2-5. Diffusivity of Hen Egg White Lysozyme as a Function of KNO_3 Concentration at pH 6.55 and 25°C. Measured correlograms and respective diffusivities of lysozyme in a series of KNO_3 concentrations at 25°C show the general effect of decreasing diffusivity with increase in ion concentration.

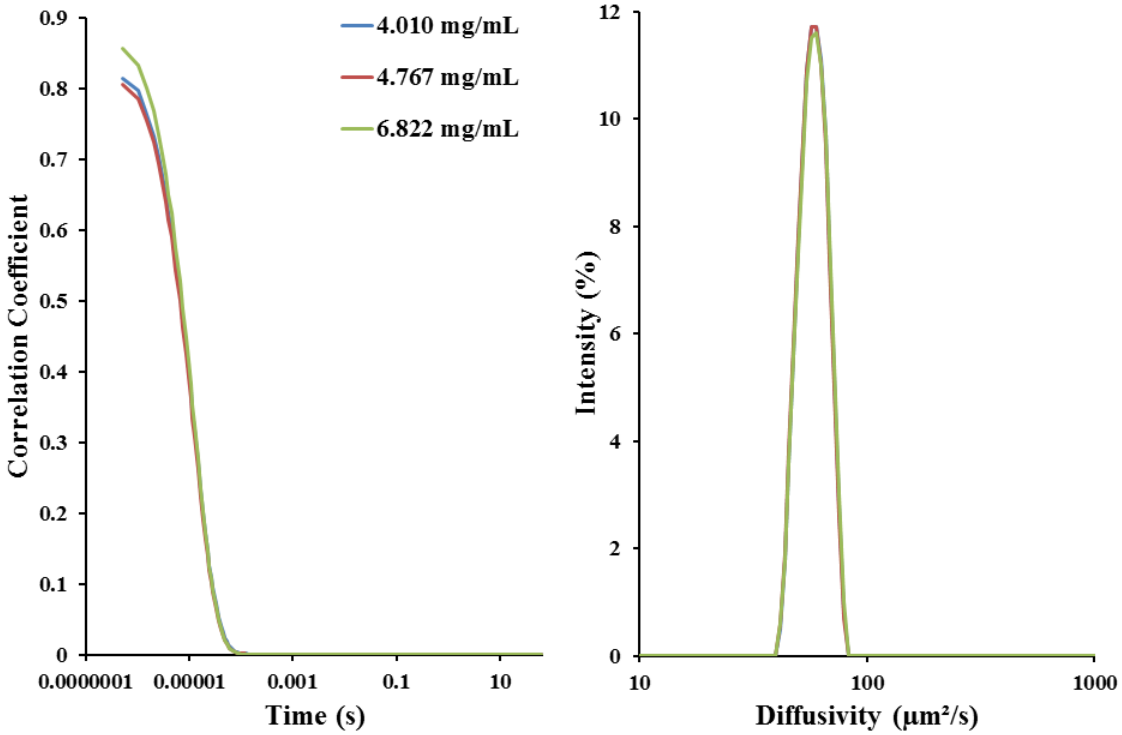

Figure 2-6. Diffusivity of Bovine Serum Albumin in 0.1M NaCl at 25°C as a Function of Protein Concentration. Due to its rarity, the constant diffusivity with protein concentration of BSA was presented. Most measurements in this thesis either increased or decreased with increase in protein concentration. Measured correlograms and respective diffusivities of BSA in 0.1 M NaCl at 25°C and various protein concentrations are shown above. There is no simple, general trend for diffusivity's dependence on protein concentration. It can increase, stay constant (as shown), or decrease with increase in protein concentration.

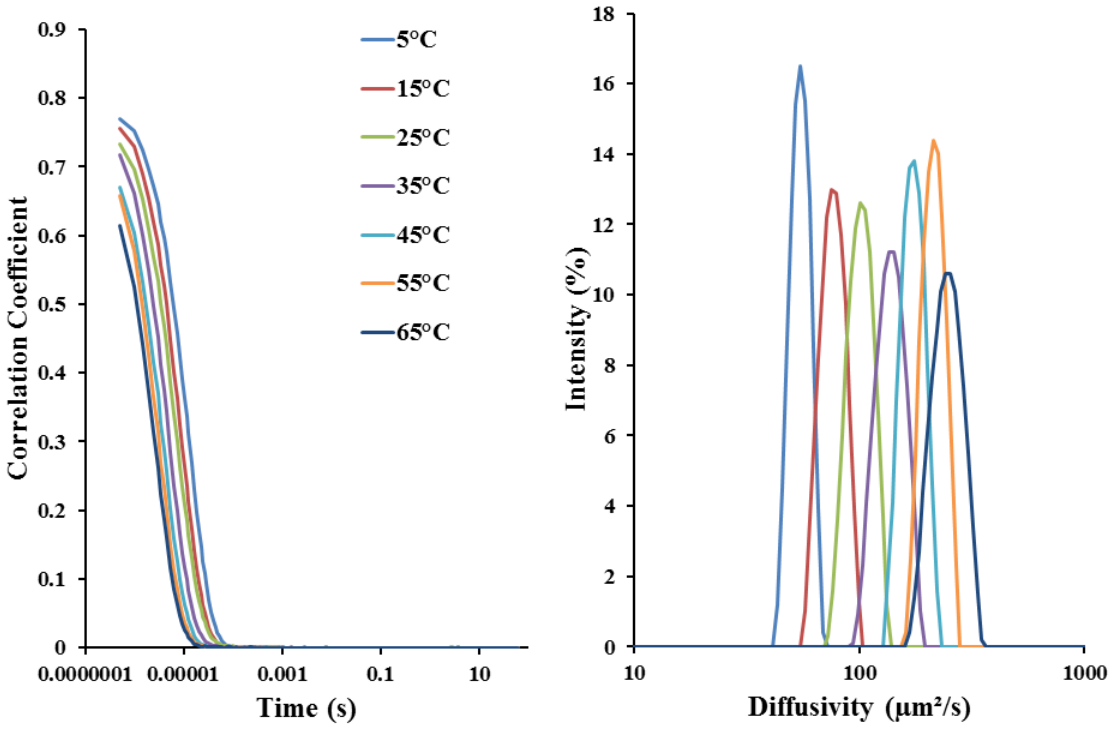

Figure 2-7. Diffusivity of Bovine β-Lactoglobulin in 0.1 M HCl as Function of Temperature. Measured correlograms and respective diffusivities of 3.585 $^{g}/_{L}$ bovine β-Lactoglobulin in 0.1 M HCl (pH=1.14 pre-mixing with protein) at various temperatures shows the general trend of increasing diffusivity with increase in temperature.

Electrophoretic and Phase Analysis Light Scattering

Electrophoretic light scattering (ELS) measures electrophoretic motion by the Doppler effect from a pulsing laser beam directed between active electrodes that generate an electric field. The illuminating beam pulse is frequency-shifted with respect to a reference to define an initial Doppler shift and strikes molecules at a scattering angle relative to the photodetector. The pulse of scattered illumination entering the detector either increases or decreases relative to the initial Doppler shift indicating the molecule's speed and direction (electrophoretic velocity). The electrophoretic velocity is converted into an electrophoretic mobility value by dividing by the magnitude of the applied

electric field. This allows comparison of electrophoretic motion under different electric fields. [2, p. Ch. 6] To use ELS with the Zetasizer, "General" analysis mode should be selected. Phase analysis light scattering (PALS) is very similar to ELS with the main difference being PALS analyzes a phase shift while ELS analyzes a frequency shift. To use PALS with the Zetasizer, "Monomodal" analysis mode should be selected.

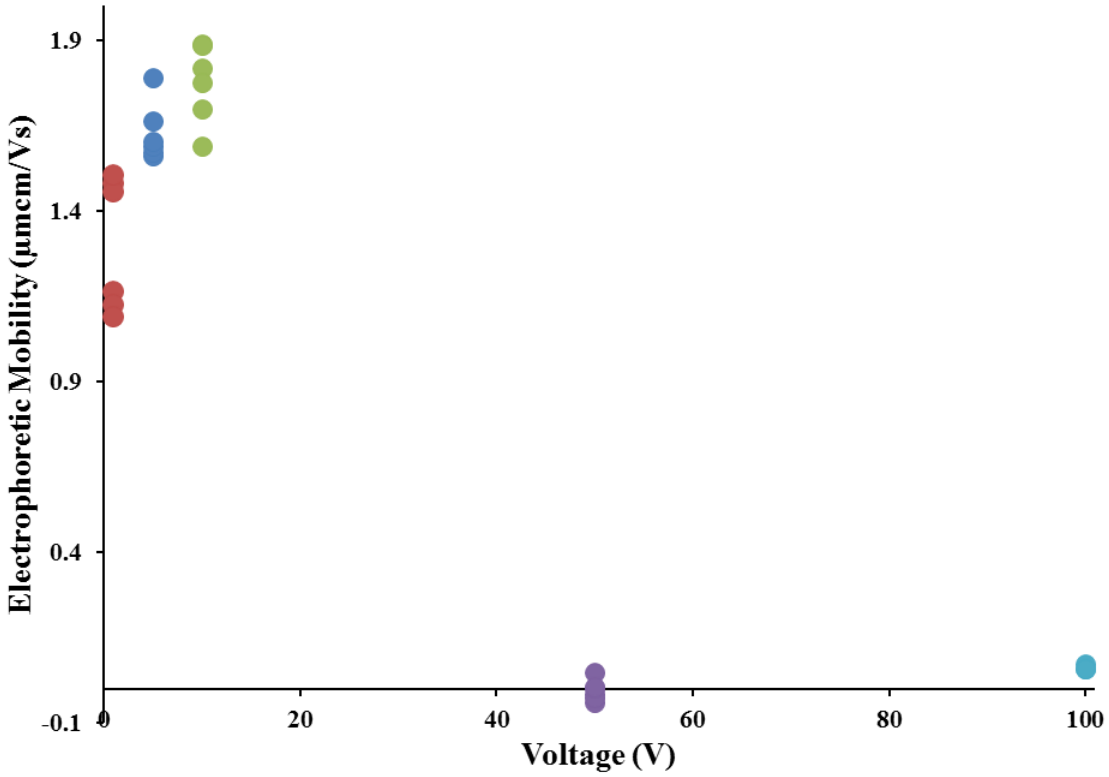

Figure 2-8. Electrophoretic Mobility of Hen Egg White Lysozyme Dependence on Applied Voltage Strength. Mobility is often assumed to be linear with applied voltage; however, this relationship assumes the molecular structure undergoing electrophoresis is sturdy enough for whatever voltages are being applied. As shown above, even a tough protein like lysozyme can only handle voltages of 10 V or less (applied between two palladium electrodes spaced 2 mm apart) before unfolding, then aggregating and causing unexpectedly low mobility measurements.

Proteins are delicate dispersions, and thus careful electric field perturbation is necessary to get them to move at the right rate without falling apart and without

aggregating. The basic procedure for measuring the electrophoretic mobility of a delicate monomeric dispersion requires an appropriate selection of:

(i) protein concentration

(ii) light attenuation

(iii) electric field voltage

(iv) proper application of voltage

Selecting a protein concentration involves considering the balance between allowing the solutions to remain concentrated enough for accurate light scattering measurements but dilute enough to ensure negligible protein-protein interactions [66]. An estimate of the maximum protein concentration allowing for negligible protein-protein interactions can be obtained by dividing a volume fraction of 0.01 [125] with the protein's specific volume. For example, the specific volume of lysozyme is about 0.000703 $^{L}/_{g}$ [126] and so protein concentrations below 14.225 $^{g}/_{L}$ should be prepared. It is important to note, this general rule is only applicable at relatively high salt concentrations (the Stokes-Einstein regime). To experimentally verify protein-protein interactions are negligible, simply measure the electrophoretic mobility at different protein concentrations and make sure they are consistent [1, p. 115]. This thesis worked in the dilute regime, and with the exception of changes in pH, no dependence on protein concentration was found in the measurements. Selection of light attenuation is based on the fact small particles scatter less light than larger particles; therefore, smaller particles need more light to capture a signal dominated by their scattering. The Zetasizer attenuation was set to the minimum setting (11) to allow as much light into the solution as possible. Selecting electric field voltage is determined iteratively through

experimentation and typically involves a loss of monomer as the voltage that induces minimal aggregation in the solution and at the electrodes is found (see **Fig. 2-8**). This is done through a combination of dynamic light scattering (DLS) and phase analysis/electrophoretic light scattering. Basically, keep lowering the voltage until DLS post-electrophoresis looks the same as DLS pre-electrophoresis. In general, 5V or less was sufficient for maintaining this criterion. However, going below 1V reduces the measurement resolution (R defined below [1, p. 170]) causing measured values to be all over the place as the electrophoretic force competes with thermal energy to control protein motion. Finally, to properly apply the voltage, induce single electric pulses and take measurements from these pulses (see **Fig. 2-9**). If a steady-state value is not achieved, try a new sample and perform pulses with an excessive time delay of 10 seconds between pulses to ensure the electrophoretic motion in the protein solution settles before the next measurement. When performed correctly measured electrophoretic mobilities should approach a steady state value that can be predicted by the Henry equation (see **Eq. 0-2c**). However, after a number of measurements depending on both the structural and dispersion stability of the protein, the electrophoretic mobility will decrease towards zero as aggregates and unfolded proteins begin to dominate the signal.

$$R = \frac{u_e E \lambda}{2\pi n D \theta} \tag{2-6}$$

where E is the applied electric field strength (i.e. voltage divide by distance applied), λ is the wavelength of light used, n is the solution refractive index, D is the measured diffusivity, and θ is the light scattering angle.

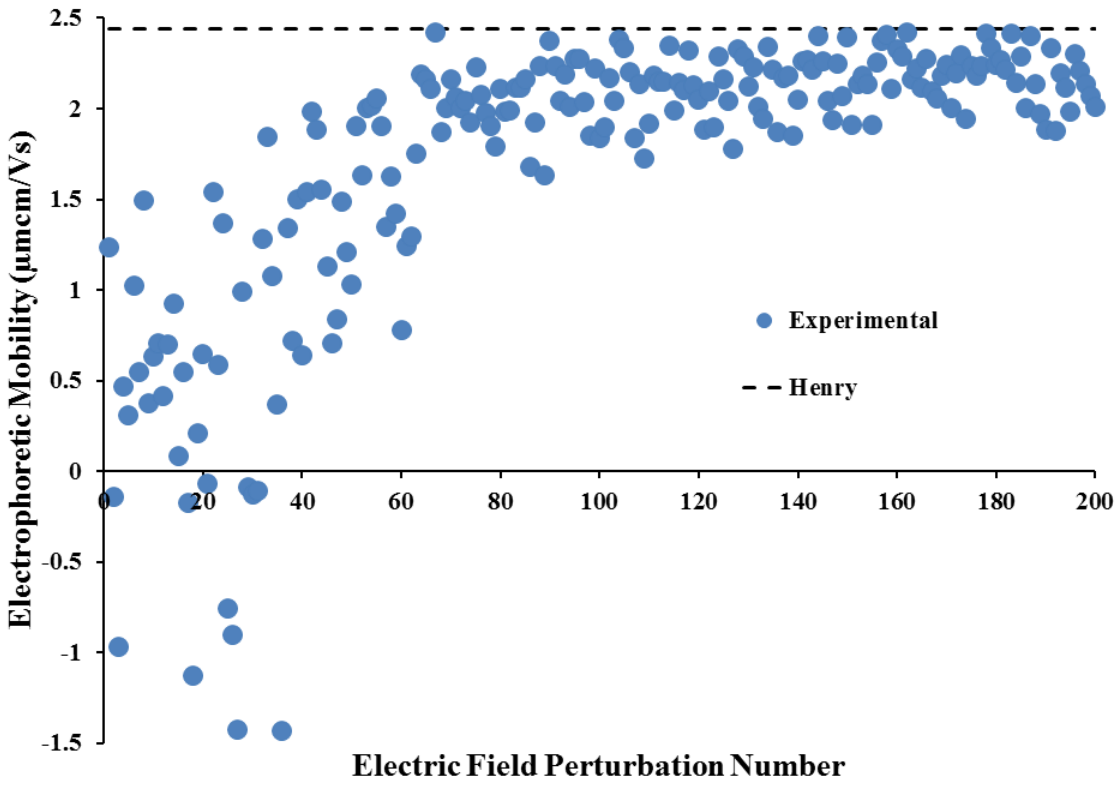

Figure 2-9. Electrophoretic Mobility of β-Lactoglobulin Dependence on Electric Field Application. Measured mobilities for bovine β-Lactoglobulin in 0.1M HCl (pH=1.14) at 35°C with an applied voltage of 5V between 2 mm spaced electrodes are shown above for each electric field perturbation (called a run by the Zetasizer). The modeled value comes from the Henry equation (**Eq. 0-2c**) assuming the electrophoretic radius is equal to the hydrodynamic radius (2.534 nm) under these conditions.

Protein Dispersity

Protein solutions were ensured to be monodisperse to satisfy both theoretical and experimental constraints [66, p. 245, 41]. As this work explores the effect of the solution conditions on solvation, it is necessary to isolate individual protein monomers for accurate measurement and analysis of EDLs. Monodispersity was obtained by filtering each protein solution with a 20 nm pore size Anotop syringe filter (GE Whatman, Pittsburgh, PA) immediately before measurements. Solutions were confirmed to be monodisperse by dynamic light scattering.

Protein Concentration

Protein solutions held a sufficiently dilute volume fraction (less than 0.01 [125]) to satisfy both theoretical and experimental constraints [66, p. 245, 41, 127]. This involved a balance between allowing the solution to remain concentrated enough for accurate light scattering measurements but dilute enough to make protein-protein interactions negligible. Volume fraction ($\boldsymbol{\phi}$) can be related to protein concentration ($\boldsymbol{C}$ in units of g/L) using the protein's specific volume ($\boldsymbol{\nu}$ in L/g) by the following relation:

$$\boldsymbol{\phi}=\boldsymbol{\nu C} \tag{2-7}$$

For example, lysozyme holds a specific volume of 0.000703 L/g [126], so a concentration below 14.225 g/L should be initially prepared. Protein concentrations were measured with an Aviv Model 14DS spectrophotometer (Lakewood, NJ) by UV absorption at 280 nm. Absorptivities at 280 nm for LYZ, BSA, and BLG are 37932 $M^{-1}cm^{-1}$, 43623 $M^{-1}cm^{-1}$, and 17550 $M^{-1}cm^{-1}$, respectively [128].

Diffusivity Measurements

A Zetasizer Nano ZS (Malvern Instruments, Worcestershire, U.K.) performed dynamic light scattering (DLS) to measure the diffusivity of filtered proteins under the different solution conditions. In general, samples were allowed five to ten minutes of thermal equilibration before measurement. For each sample, six autocorrelation functions with acquisition times of about 150 s were measured with a 4 mW He-Ne laser at λ=633 nm using back-scattering at an angle of 173°. Single particle/protein diffusivity values were determined by plotting the measured diffusivities versus protein concentration and

extrapolating the value of the y-intercept, where protein concentration is zero. Hydrodynamic radii were calculated using the Stokes-Einstein equation (**Eq. 0-1**) and pure solvent viscosity values (see **APPENDIX G** for values and **APPENDIX C** for calculation).

Electrophoretic Mobility Measurements

A Zetasizer Nano ZS (Malvern Instruments, Worcestershire, U.K.) performed primarily phase analysis light scattering (PALS) to measure the electrophoretic mobility of filtered proteins under the different solution conditions. In general, samples were allowed five to ten minutes of thermal equilibration before measurement. For each sample, a minimum of 20 technical replicates were taken using forward scattering of a 4 mW He-Ne laser at λ=633 nm at an angle of 17° through a 2 mm spaced palladium plated dip cell. Samples were checked for monodispersity before and after measurements by DLS. Samples were discarded after measurement due to aggregation induced during electrophoresis. Electrophoretic radii were calculated using the Henry equation (**Eq. 0-2c**) and pure solvent viscosity values (see **APPENDIX G** for values and **APPENDIX C** for calculation).

pH Measurements

The pH was measured with a model 14002-850 sympHony pH electrode (VWR, Randnor, PA) following calibration.

Computational Methods

The details of the computational protocol for predicting the zeta potential of a molecular structure can be found in **Ch. I**. In short, the computation determines the zeta potential of a molecular structure through six primary steps (see **Fig. 1-1**) that essentially models an EDL over the structure and averages the electric potentials at the slip plane to define the zeta potential. Descriptions and code for controlling its component software (MSMS [85], APBS [88], HYDROPRO [83], PDB2PQR [54, 53], and PROPKA [55]) are included in **APPENDIX D**.

CONCLUSIONS

This chapter described necessary considerations and the performance of experimental procedures used throughout this body of work. With these standard operating procedures, experiments were significantly simplified allowing for accurate measurements of the size and mobility of the different proteins, which were in agreement with previous works and theory. This agreement is shown in the experimental results contained in the remaining four chapters of this thesis.

CHAPTER III

The Impact of Ionic Strength on Hydration

Chapter Abstract

This chapter assessed hydration with varying ionic strength, which held two parts: ion type and ion concentration. Experimental work tested the hypothesis that hydration remains the same during diffusion and electrophoresis using hen egg white lysozyme (LYZ) and bovine β-lactoglobulin (BLG) in different electrolyte solutions of varying ion concentration. Ion type was assessed with the following four salts along the Hofmeister series: KH_2PO_4, KCl, KNO_3, and $KClO_4$. This involved measuring the diffusivity and electrophoretic mobility by light scattering in the different ionic environments to determine experimental radii during diffusion (R_h) and electrophoresis (R_e). For LYZ with KCl and $KClO_4$, comparison of the R_h and R_e showed hydration to remain the same indicating the X_{sp} coincides with the R_h for the LYZ-KCl and LYZ-$KClO_4$ interfaces. However, deviation was found for KNO_3 and KH_2PO_4, which may be the result of specific ionic interactions with LYZ. For BLG with KCl, comparison of the R_h and R_e indicated the X_{sp} coincided with the R_h. ZPRED accurately predicted the electrophoretic mobilities of both proteins in all variations of ionic strength using a constant, computed X_{SP}.

INTRODUCTION

Positively charged LYZ is a model system [30, 98] for assessing the effect of various anionic counter-ions along the Hofmeister series (specifically KH_2PO_4, KCl, KNO_3, and $KClO_4$). In addition, positively charged BLG with HCl and KCl was used for assessing the combined impact of change in ion concentration and temperature. As the impact of the Hofmeister series is more pronounced with its anions than cations [30], positive proteins were necessary for this aim. LYZ was used due to its positive charge at neutral pH [52], high monodispersity, and sturdy structure insensitive to low and high ion concentrations [99]. With these attributes, the effects of the different anionic counter-ions were easily measured by simply adding them to solution with LYZ. Note: the highest ion concentrations that could be acquired for the mixing procedure with LYZ in KH_2PO_4 was 0.75 M and in $KClO_4$ was 0.05 M due to the ions' solubilities in water at 25°C. Three different protein concentrations of LYZ were prepared. LYZ at twice the desired protein concentration was dissolved in deionized water and filtered (described in **Ch. II**) to separate monomers from aggregates. The pH of the water pre-mixing with protein was 6.55. LYZ concentrations were determined post-filtration as described in **Ch. II**. All measurements were taken immediately after mixing with salt. 0.5 mL of the filtered protein solutions (containing twice the desired protein concentration) were mixed with 0.5 mL of each salt solution (containing twice the desired ion concentration) yielding a 1 mL solution containing desired concentrations of both protein and salt. Experimental results for the mixed solutions are discussed below.

RESULTS AND DISCUSSION

Diffusion of LYZ in Various Salts

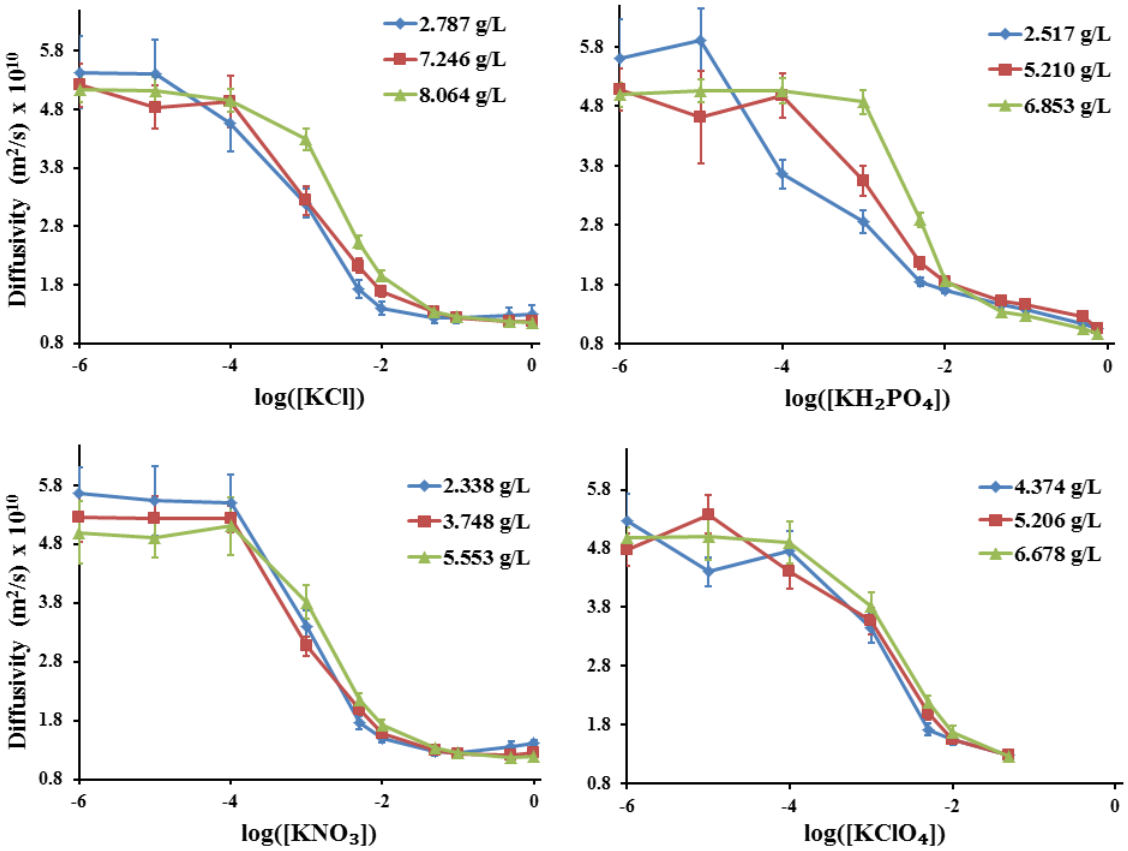

Figure 3-1. Measured Diffusivities of LYZ with Various Electrolyte Solutions. Experimental solution conditions employed during measurement can be found in **APPENDIX G**.

In **Fig. 3-1,** the measured diffusivities of LYZ in the presence of all the salts showed the same trend, which is a transition between two different regimes of diffusion: the hyper-diffusive and Stokes-Einstein regimes. In general, it appears the onset of the transition from the hyper-diffusive regime is protein concentration dependent. This makes sense being the higher the presence of protein, the more ions it takes to initiate the neutralization of the electrostatic enhancement on diffusion. The trend for LYZ with KH_2PO_4 differs slightly from the others, which may be the result of a specific interaction

between lysozyme and H_2PO_4, or possibly from the fact the H_2PO_4 ion is a kosmotrope. Diffusivity values were converted to a single protein diffusivity value for each ion concentration by plotting the diffusivities versus protein concentration and acquiring the y-intercept (where the protein concentration is zero) by extrapolation. Single protein diffusivities were used in **Eq. 0-1** along with pure solvent viscosities, which can be found in **APPENDIX G** and were calculated based on **APPENDIX C**, to determine the hydrodynamic radii during diffusion (R_h).

R_h values are plotted in **Fig. 3-2**, and show the Stokes-Einstein regime (i.e. ion concentrations above ~0.005M) is the only range of ion concentrations at which the R_h can be defined to give physically realistic values. In this regime, all electrolytes except KH_2PO_4 show a similar trend in approaching a maximum and then decreasing due to EDL contraction. EDL contraction refers to the disintegration of the outer solvation layers with increasing ionic strength. This effect can be theoretically quantified with the **Debye length**, representing the EDL edge from the protein surface [1]. The maximum in the hydrodynamic radius is most likely a result of counter-ions saturating the LYZ surface [98]. The rate at which this happens may be the related to the size of the ions [98] (see **APPENDIX C** for ionic radii values).

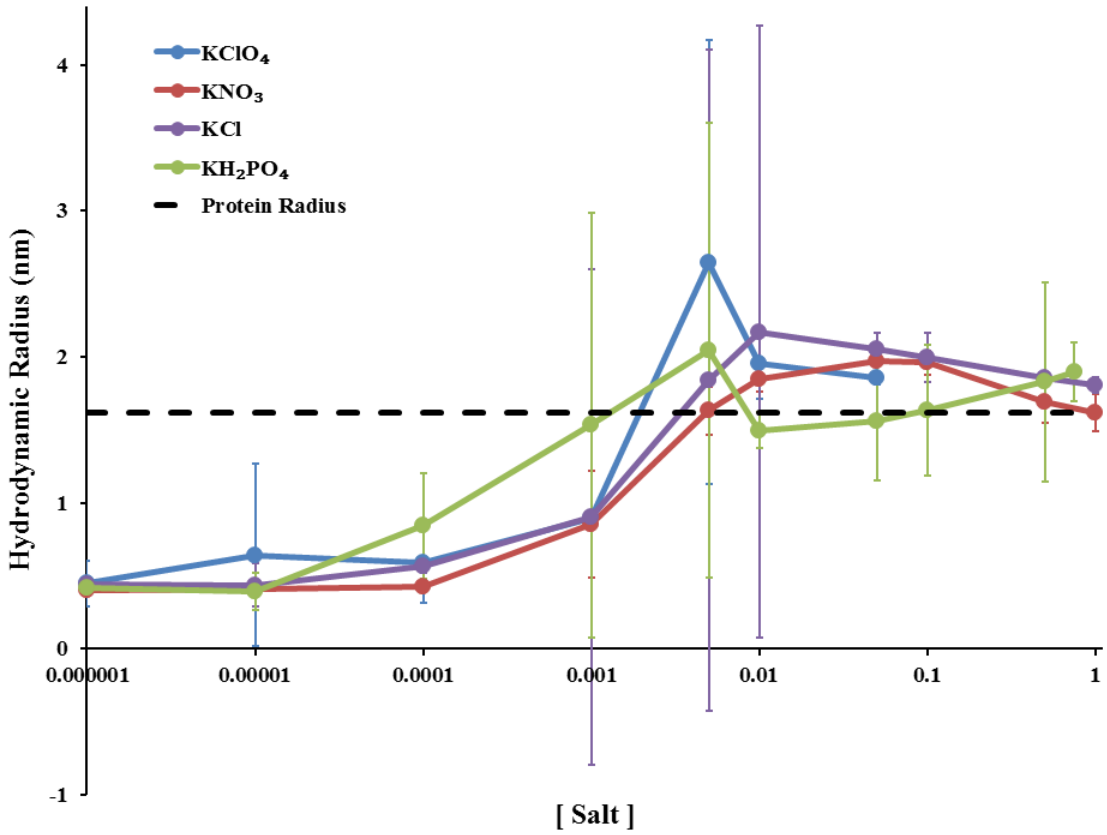

Figure 3-2. Comparison of Hydrodynamic Radii of LYZ with Different Electrolytes

Electrophoresis of LYZ in Various Salts

Electrophoresis of LYZ in the increasing presence of the different salts yielded similar results as shown in **Fig. 3-3**. Experimental results are compared to two different models: the Henry equation and ZPRED. The Henry equation [47] (**Eq. 0-2c** rearranged for electrophoretic mobility) assumed the electrophoretic radius (R_e) to be equal to the radius of LYZ (1.64 nm) plus the diameter of water (0.284 nm [68]). As can be seen, this proves to be a fairly accurate representation of the R_e value among the different salts used here. ZPRED was applied to twenty of each of the lysozyme monomer (PDB id: 6lyz [97]) and dimer (PDB id: 4r0f [129]) conformations sampled from molecular dynamics simulations (see **Ch. I** for the details). Their shape descriptors can be found in **Table 2-1**.

By comparison with experimental values, it appears monomers were predominately present throughout the entire range of ion concentrations with the possibility of dimers at the higher ion concentrations, which is in agreement with previous findings [101]. Nonetheless, it was assumed all mobility values represented the monomeric state for determining R_e values. Experimental electrophoretic mobilities were used to calculate experimental R_e values using **Eq. 0-2c** with pure solvent viscosity values, which can be found in **APPENDIX G** and were calculated as described in **APPENDIX C**.

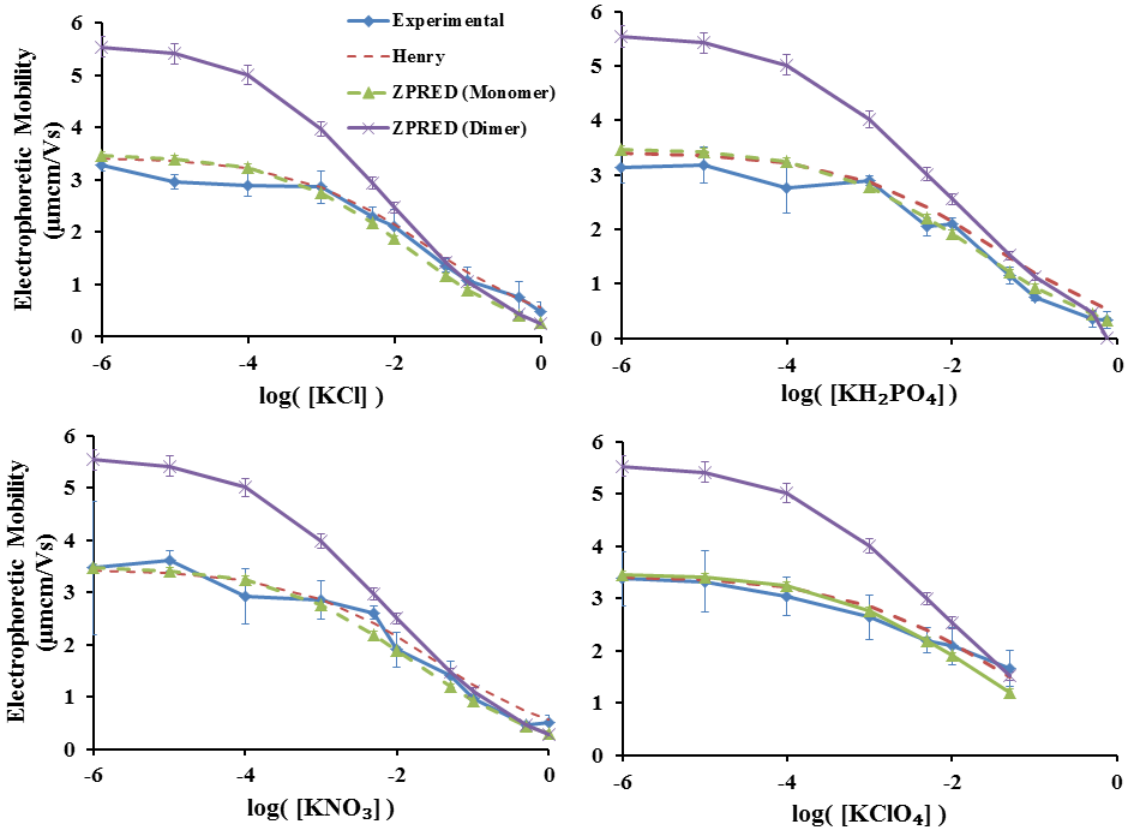

Figure 3-3. Measured Electrophoretic Mobilities of LYZ in Various Electrolytes. Experimental solution conditions for each salt can be found in **APPENDIX G**.

Calculated experimental electrophoretic radii (R_e) are shown in **Fig. 3-4**. R_e values were all found to be similar until the higher ion concentrations, which maybe the

result of the dimerization of LYZ. As shown in **Fig. 3-4**, values portray a hydration layer approximately equal to the diameter of a water molecule (0.284 nm [68]).

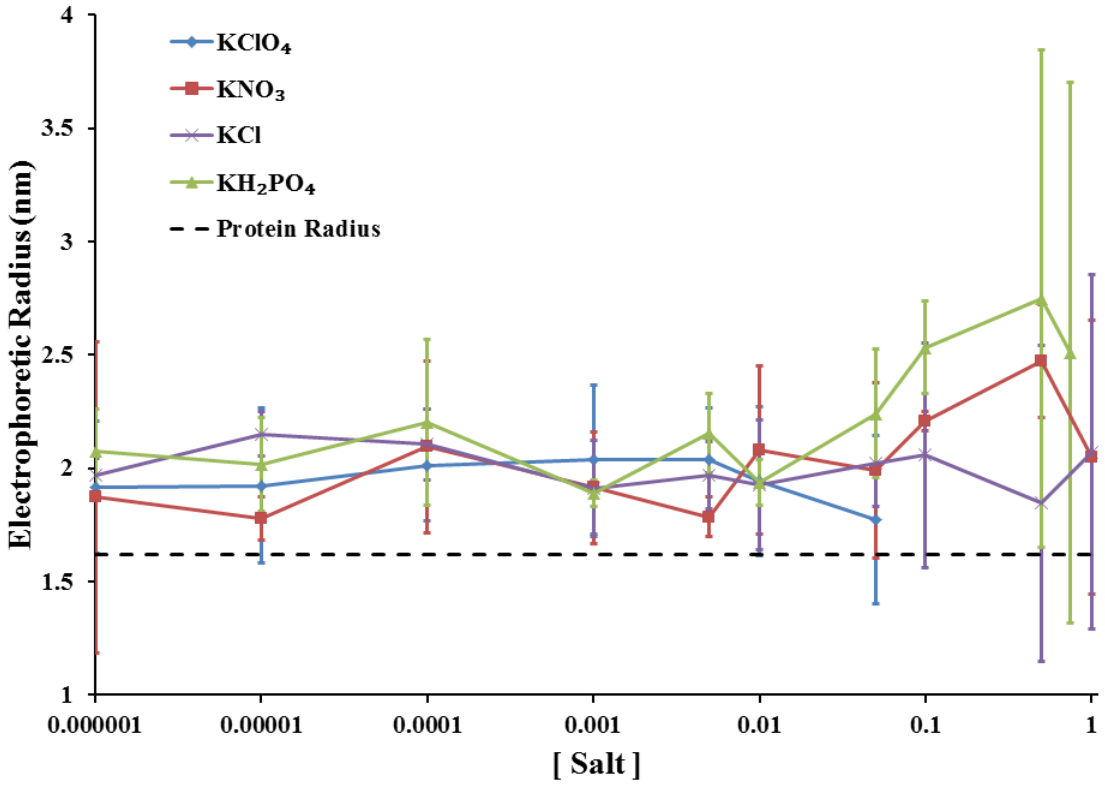

Figure 3-4. Comparison of Electrophoretic Radii of LYZ in Different Electrolytes

Analysis/Discussion of LYZ in Various Salts

As the hyper-diffusive regime is the result of an electrostatic enhancement from neighboring proteins, the ion concentration at which the Stokes-Einstein regime begins provides a direct measure of the ion's ability to neutralize the overall EDL surrounding the protein allowing molecules to appear "neutral". The ion concentration that effectively quenches this charge-driven enhancement provides a measure of the ion's "charge-quenching power" and may be useful for assessing the difference each ion holds on the effect of surface potential dampening. This ion concentration, at which the Stokes-

Einstein regime begins, is estimated experimentally by interpolating the ion concentration where the electrophoretic and hydrodynamic radii first meet.

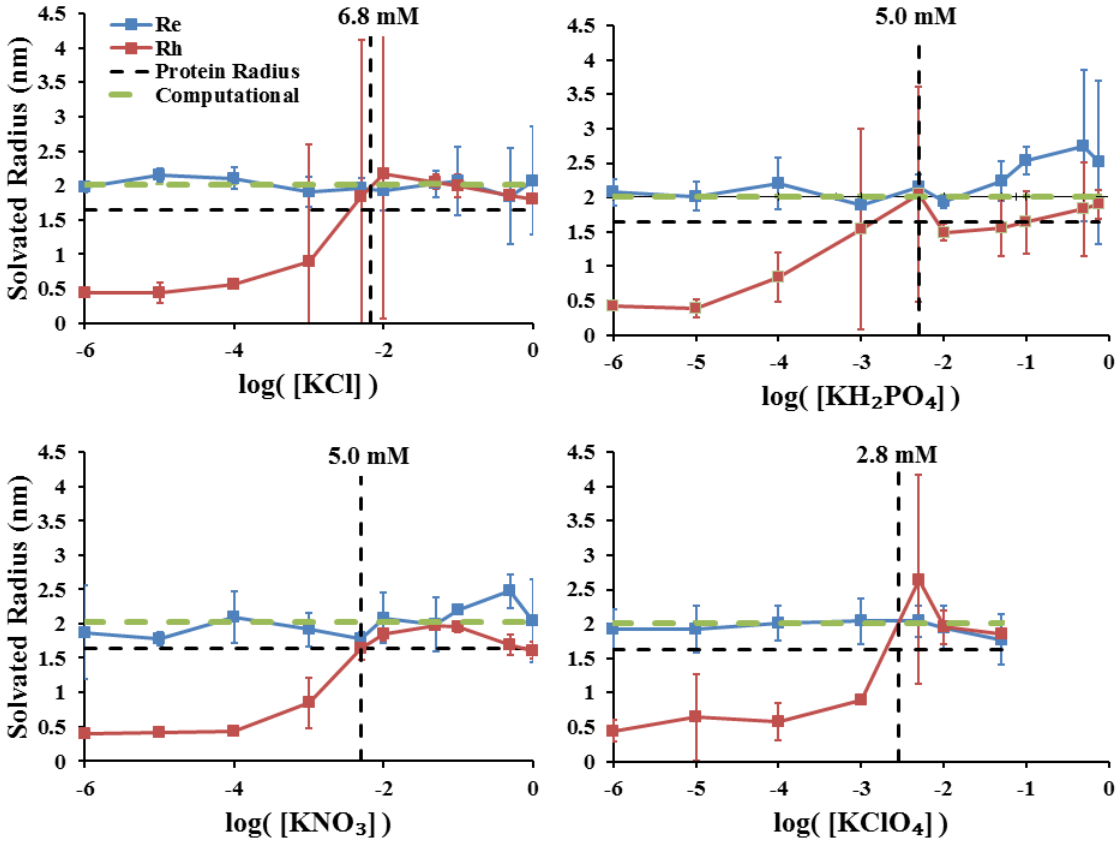

Figure 3-5. Comparison of Experimental Solvated Radii of LYZ in Different Salts. Comparison of the onset of the Stokes-Einstein regime provides an assessment of the effective charge quenching power of each ion by its ability to quench the electrostatic enhancement on diffusion.

As shown in **Fig. 3-5**, the Stokes-Einstein regime (i.e. ion concentrations approximately above 0.005 M) is the only range of ion concentrations at which the R_h can be defined to give physically realistic values using **Eq. 0-1**. Though it is interesting to note the R_h values at the lowest ion concentration (1 μM) came out to practically the same value for all ions used (0.425 ± 0.02 nm). Thus, it seems these ions all become ineffective at 1 μM. In the Stokes-Einstein regime (marked by the dashed vertical line in **Fig. 3-5**), most of the calculated radii are within error indicating agreement in the

methods for determining molecular size [41, 51] and thus the hydration layer thickness. However, significant deviations between the two were found at the higher ion concentrations for KNO_3 and KH_2PO_4. This could be the result of dimerization of LYZ during electrophoresis. Unfortunately, it seems to be the nature of proteins to resist remaining monodisperse. Comparing the onset of the Stokes-Einstein regime among the ions, it seems they can be ranked in the following order of decreasing "charge-quenching power": $KClO_4$ (2.8 mM), KH_2PO_4 (5 mM), KNO_3 (5 mM) and KCl (6.8 mM). This seems to be correlated with the size of the counter-ions, which are 2.28 Å for ClO_4^- [68], 2.38 Å for $H_2PO_4^-$ [37], 2.03 Å for NO_3^- [68] and 1.76 Å for Cl^- [68]. This analysis may be a bit flawed due to the large variance in the measurements, and thus a higher resolution measurement would be useful for clarifying this observation. Nonetheless, this connection in KCl and $KClO_4$ indicates the EDL of lysozyme is the same under both electrophoretic and diffusive conditions, supporting the central hypothesis that the X_{SP} coincides with the R_h. Values for the different radii are presented in **Table 3-1**. To estimate the slip plane position from experimental, the protein radius (R_p = 1.64 nm) was subtracted from the measured R_h values in the Stokes-Einstein regime and R_e values outside of this regime (**Eq. 3-1**). Note this equation runs parallel to **Eqs. 0-3** and **0-4**.

$$X_{SP} = \begin{cases} R_e - R_p, & C_i < C^{SE} \\ R_h - R_p, & C_i \geq C^{SE} \end{cases} \quad \textbf{(3-1)}$$

where X_{SP} is the slip plane position relative to the protein surface, R_e is the electrophoretic radius (defined in **Eq. 0-2c**), R_p is protein radius (defined in **Eq. 2-1**), C_i is the ion concentration, C^{SE} is the ion concentration at which the Stokes-Einstein regime begins, and R_h is the hydrodynamic radius (defined in **Eq. 0-1**).

	KCl		KH_2PO_4		KNO_3		$KClO_4$	
log [Salt]	**R_e** (nm)	**R_h** (nm)	**R_e** (nm)	**R_h** (nm)	**R_e** (nm)	**R_h** (nm)	**R_e** (nm)	**R_h** (nm)
-6	1.969 (0.1)	0.438 (0.01)	2.074 (0.2)	0.414 (0.03)	1.873 (0.6)	0.403 (0.1)	1.917 (0.3)	0.446 (0.2)
-5	2.152 (0.1)	0.436 (0.2)	2.015 (0.2)	0.390 (0.1)	1.776 (0.1)	0.410 (0.01)	1.923 (0.3)	0.641 (0.6)
-4	2.106 (0.2)	0.562 (0.04)	2.203 (0.4)	0.841 (0.4)	2.094 (0.4)	0.423 (0.02)	2.012 (0.3)	0.586 (0.3)
-3	1.909 (0.2)	0.899 (1.7)	1.890 (0.1)	1.532 (1.5)	1.914 (0.2)	0.852 (0.4)	2.038 (0.3)	0.896 (0.1)
-2.30	1.968 (0.1)	1.841 (2.3)	2.155 (0.2)	2.048 (1.6)	1.785 (0.1)	1.632 (0.2)	2.037 (0.2)	2.648 (1.5)
-2	1.928 (0.3)	2.169 (2.1)	1.938 (0.1)	1.494 (0.1)	2.081 (0.4)	1.844 (0.1)	1.941 (0.3)	1.953 (0.2)
-1.30	2.024 (0.2)	2.056 (0.1)	2.242 (0.3)	1.555 (0.4)	1.990 (0.4)	1.970 (0.01)	1.773 (0.4)	1.851 (0.1)
-1	2.059 (0.5)	1.997 (0.2)	2.534 (0.2)	1.635 (0.4)	2.207 (0.1)	1.959 (0.1)	--	--
-0.30	1.846 (0.7)	1.857 (0.01)	2.748 (1.1)	1.828 (0.7)	2.473 (0.2)	1.692 (0.1)	--	--
-0.13	--	--	2.510 (1.2)	1.898 (0.2)	--	--	--	--
0	2.072 (0.8)	1.808 (0.1)	--	--	2.049 (0.6)	1.613 (0.1)	--	--

Table 3-1. Experimental Hydrated Radii of LYZ in Various Electrolytes. Values are shown for each radii (above) with their standard deviation in parenthesis.

For all ions, the R_e is approximately equal to the radius of LYZ plus a water molecule (1.64+0.284 = 1.924 nm), indicating a constant hydration layer. However, the R_h is seen to vary with ion concentration in the Stokes-Einstein regime, implying a variable hydration layer. This begs the question: is the X_{SP} constant or variable with regard to ionic strength? To assess this question, ZPRED was applied to LYZ using both a constant and a variable slip plane position.

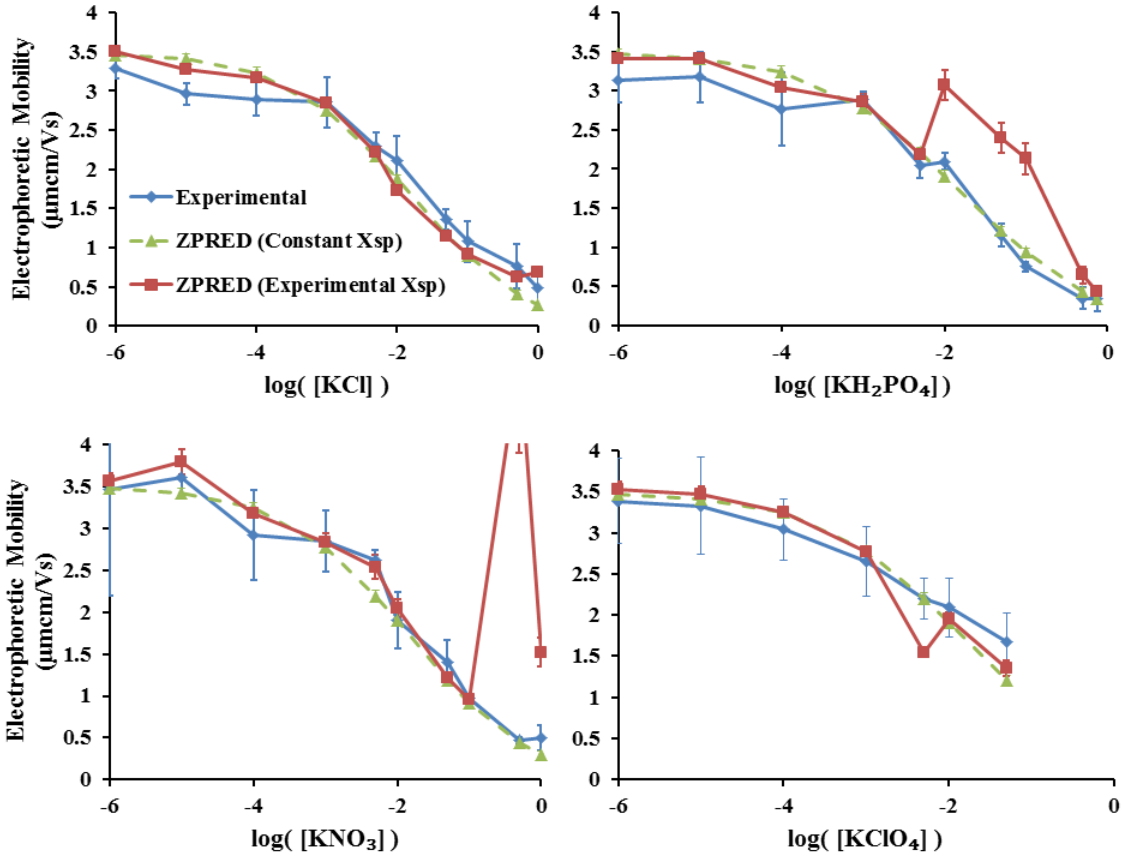

Figure 3-6. Zeta Potential Analysis of the Slip Plane Estimates for LYZ in the Presence of Different Electrolytes

Based on the zeta potential analysis in **Fig. 3-6**, it is clear using X_{SP} values experimentally determined from R_h values can result in significant deviation from experimental values while use of a constant hydration layer adequately describes the electrokinetic behavior of lysozyme in all the different salts. The LYZ-KCl interface was the only interface that could represent its hydration layer thickness by the difference in R_h and R_p for all ion concentrations. Considering these hydration layer thicknesses were determined from diffusion behavior, the electrophoretic rate of motion shown is that expected of LYZ with an equivalent interface. Thus, deviations from values using a constant X_{SP} show a theoretical electrophoretic enhancement associated with the loss of hydration. Or alternatively, it can be thought of as the theoretical electrophoretic loss

associated with hydration. Nonetheless, electrophoretic mobilities computed with the experimental X_{SP} values became unrealistic in the Stokes-Einstein regime, where specific ion effects may have been in effect. This indicates experimentally determined R_h values are not always realistic, and experimentally determined R_e values seem to provide a more realistic depiction of the solvation of LYZ. In conclusion, it seems the hydration layer remains constant, approaching the diameter of a water molecule, for lysozyme in the different electrolytes.

Diffusion of BLG in 0.005 M HCl and Increasing [KCl]

Shown in **Fig. 3-7**, BLG also experiences a taste of the hyper-diffusive regime like LYZ in KCl (see **Fig. 3-1**). Though since HCl was used to drop the pH, the experiment was restricted to a minimum Cl concentration of 0.005 M. Considering LYZ (at pH 6.55) holds a net valence of 8 and BLG (at pH 2.32) holds a valence of 19.5 as calculated by PROPKA [55], the hyper-diffusive regime was expected to exist at higher ion concentrations relative to LYZ. This can be seen in **Fig. 3-8**.

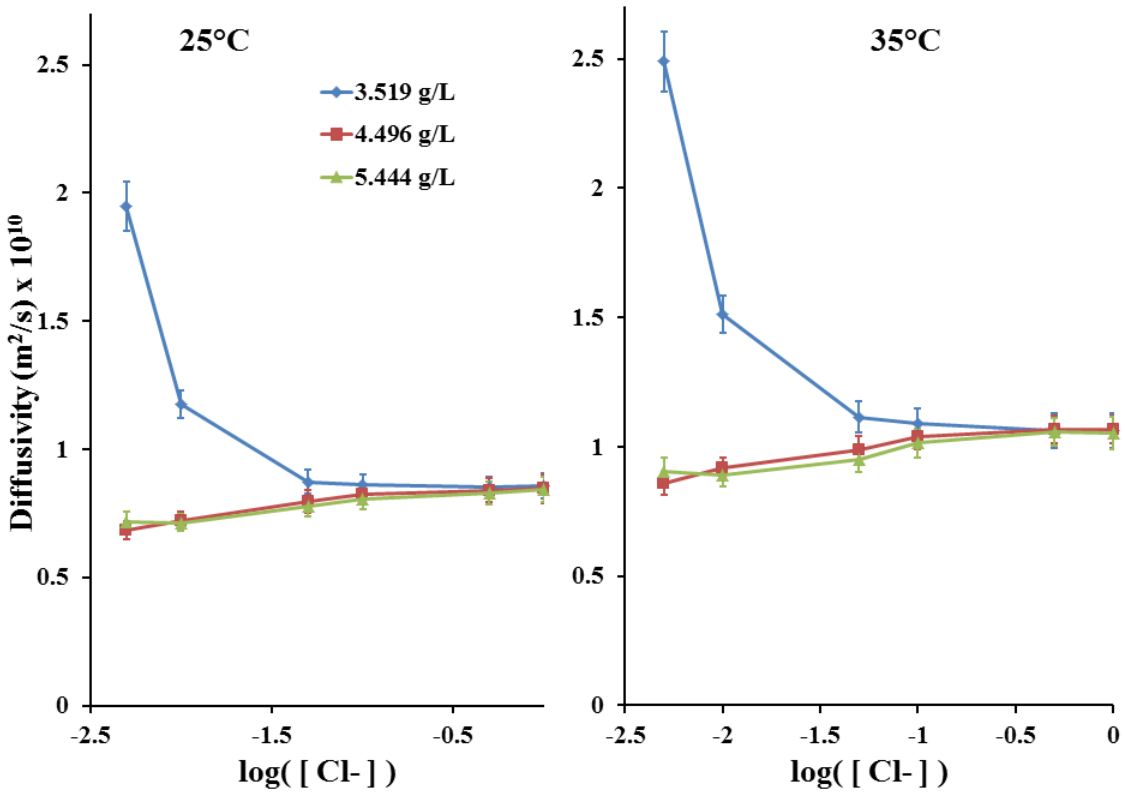

Figure 3-7. Measured Diffusivities of BLG in 0.005 M HCl and Increasing KCl Concentrations. Values are plotted against the counter-ion concentration. The experimental solution conditions can be found in **APPENDIX G**.

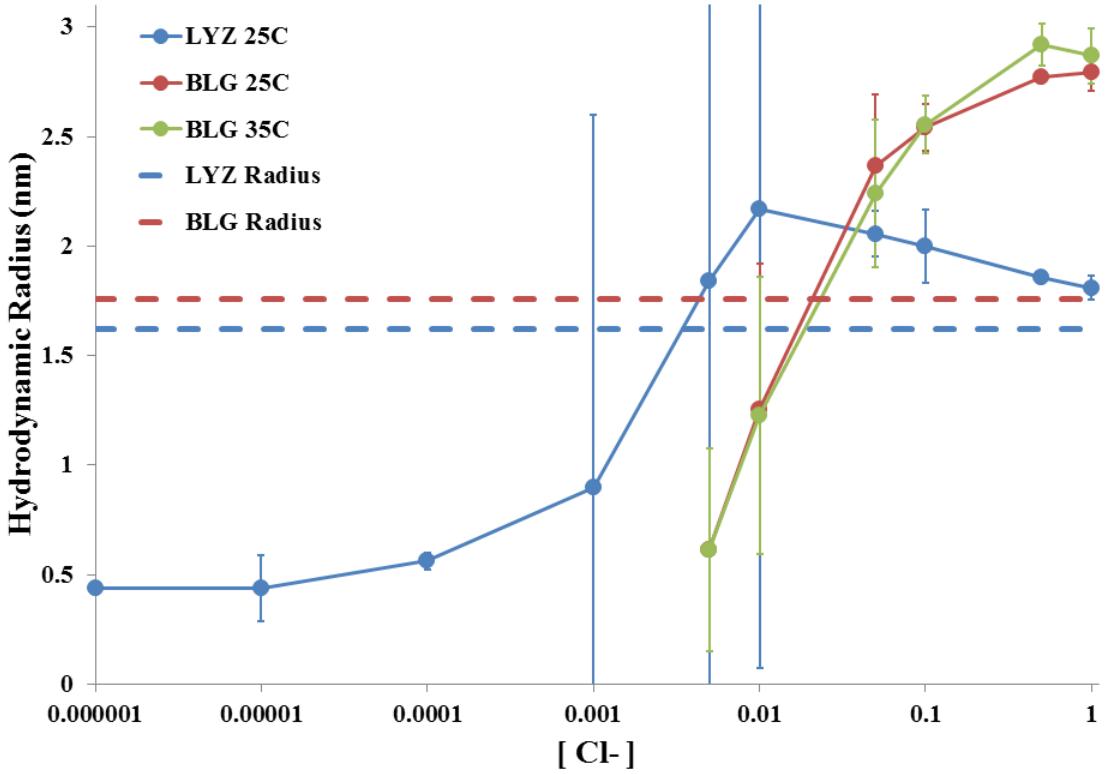

Figure 3-8. Comparison of Hydrodynamic Radii of LYZ and BLG with Increasing KCl Concentration

There are two differences that might be the cause of the deviation in behavior seen by LYZ and BLG in **Fig. 3-8**. First, BLG holds a valence (19.5 as calculated by PROPKA at pH 2.32 and supported by the literature (19.2 at pH 2.5 [117, 116])) almost three times as large as LYZ (8 as calculated by PROPKA and supported by the literature [52]). Thus, this charge difference may be one cause of the shifted maximum at higher ion concentrations. Another possible difference is BLG may be more sensitive to dimerization than LYZ [118, 101]. All measurements were taken immediately after mixing with salt; however, BLG was given two hours to dissociate at pH 1.99 (in a 0.1M HCl solution) based on a previous procedure [115] and then the pH was re-measured (pH 2.32) before diffusivity and electrophoretic mobility measurements were taken. This two

hour delay in 0.1 M HCl may have induced dimerization, which became more apparent as the KCl concentration was increased to neutralize the hyper-diffusivity effect. Nonetheless, despite what may have happened, the electrophoretic mobility measurements reflect this change in effective size as well.

Electrophoresis of BLG in 0.005 M HCl and Increasing [KCl]

The electrophoretic mobilities of BLG at 25°C and 35°C show the expected trend of decreasing mobility with increasing ion concentration (**Fig. 3-9**). Similarly to **Fig. 3-3**, experimental data are compared to two models: the Henry equation and ZPRED. The Henry equation [47] (**Eq. 0-2c** rearranged for electrophoretic mobility) assumed the electrophoretic radius (R_e) to be equal to the constant hydrodynamic radius computed by HYDROPRO (2.210 nm) on the ensemble of BLG structures sampled from molecular dynamics for both temperatures. As can be seen in **Fig. 3-9**, this seems to adequately represent the R_e. ZPRED was applied to both the monomeric (PDB id: 3blg) and dimeric (PDB id: 1beb) structures of BLG. Unfortunately, it seems as the ion concentration increases, the mobility of the monomer and dimer coincide, making the two structures indistinguishable by electrophoresis. Overall, the experimental data better fits the computed trend for the dimer (1beb) at the two different temperatures. Therefore, it seems the two hour delay in 0.1M HCl may have indeed induced dimerization of the BLG structure.

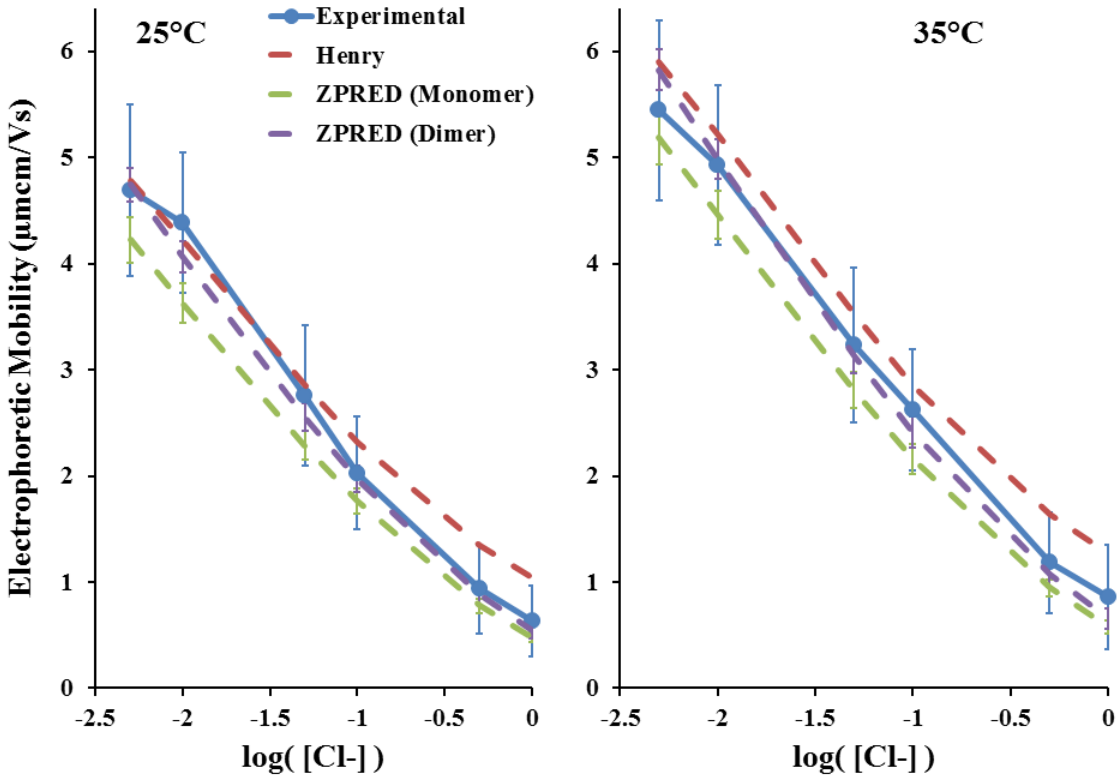

Figure 3-9. Electrophoretic Behavior of BLG in 0.005 M HCl and Increasing KCl Concentration at Different Temperatures

Experimentally determined R_e values (**Fig. 3-10**) show a continuously increasing size with increasing ion concentration; providing further evidence of BLG dimerization. Interestingly, both the diffusive and electrophoretic measurements caught this behavior, and as shown in **Fig. 3-11**, it appears the hydration over BLG remains the same during both.

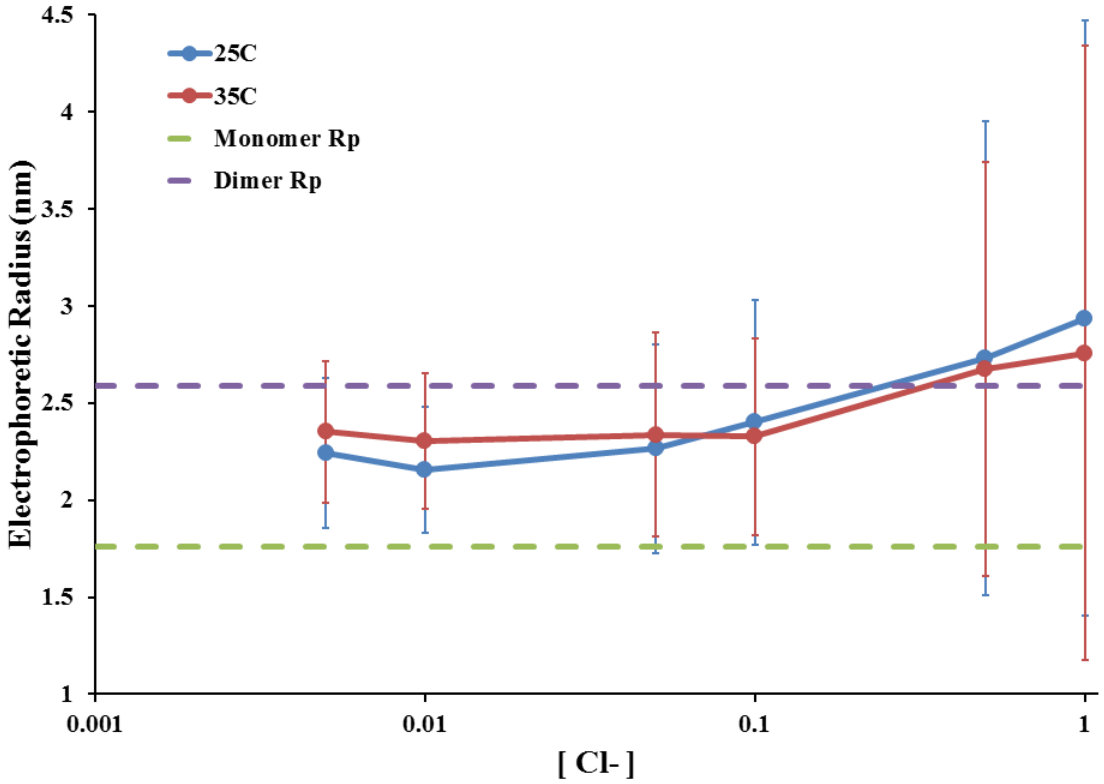

Figure 3-10. Electrophoretic Radii of BLG in KCl at Different Temperatures

Analysis/Discussion of BLG in 0.005 M HCl and Increasing [KCl]

In **Fig. 3-11**, the R_h of BLG is shown to coincide with its R_e even throughout what might have been a dimerization event. This provides further support for the central hypothesis that the X_{SP} coincides with the R_h showing it is valid for more than what was expected.

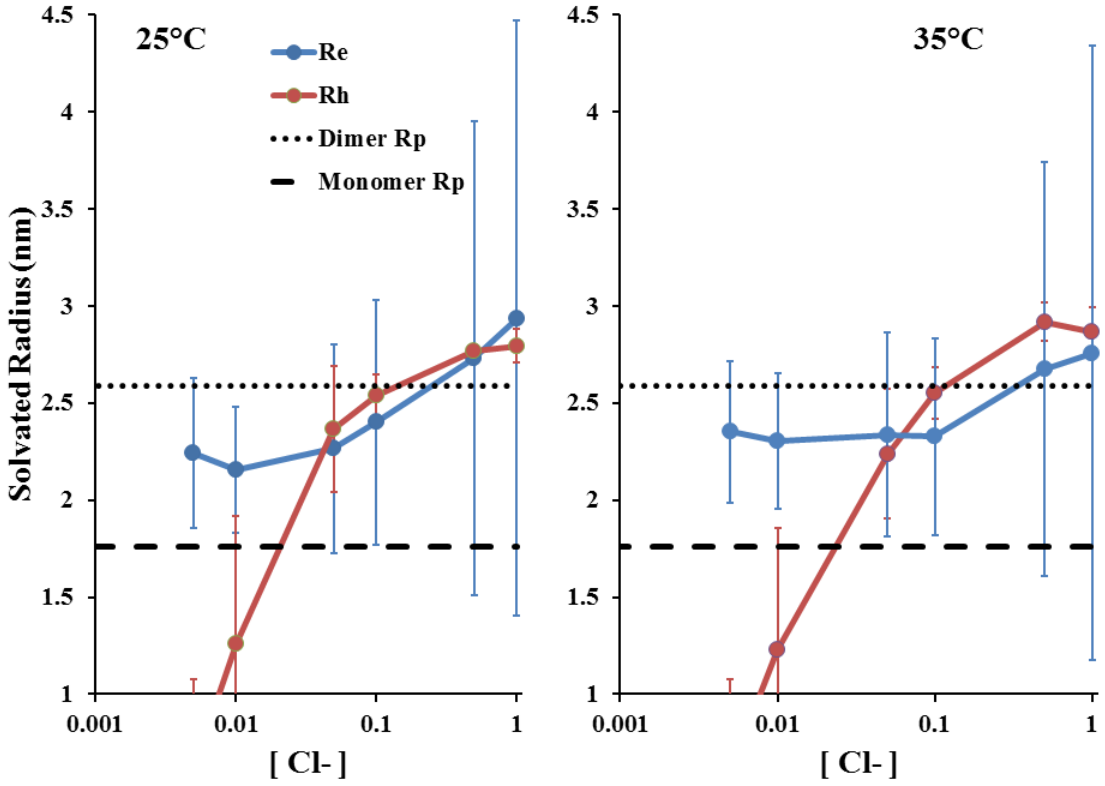

Figure 3-11. Comparison of Experimental Radii of BLG in 0.005 M HCl and Various KCl Concentrations

CONCLUSIONS

Experimental results for LYZ (**Fig. 3-5**) and BLG (**Fig. 3-11**) both indicate the X_{sp} coincides with the R_h for the protein-KCl interface. Due to its properties [33, 11, p. 35], KCl provides a salt standard for comparison and represents the case of no specific ion interactions modeled by the Hofmeister series. Thus, it can be concluded the X_{sp} coincides with the R_h for an indifferent electrolyte (i.e. an electrolyte that experiences purely Coulombic interaction, holding no specific interaction with the protein surface) despite change in ion concentration. In addition, the LYZ-$KClO_4$ interface and most ion concentrations for the LYZ-KNO_3 interface also showed the X_{SP} to coincide with the R_h. These findings indicate the X_{SP} coincides with the R_h when the ion type happens to be a chaotrope and no specific ionic interactions are in effect. Future work will test the coincidence of the X_{SP} and R_h with other chaotropes and kosmotropes as well as solutions holding different relative dielectrics to further explore the effect ionic interactions hold on the protein-solvent interface.

CHAPTER IV

The Impact of Temperature on Hydration

Chapter Abstract

In this chapter, the effect of varying temperature on hydration is assessed. Experimental work tested the hypothesis that hydration remains the same during diffusion and electrophoresis with varying temperature using bovine serum albumin (BSA). In addition, the electrophoretic behavior of the collagen-like triple helix, $[(PPG)_{10}]_3$, was modeled by ZPRED. This involved the same measurements and analysis as **Ch. III**. For BSA, comparison of R_h and R_e showed hydration to remain the same, indicating the X_{sp} coincides with the R_h for the BSA-NaCl interface at various temperatures up to its melting point. For $[(PPG)_{10}]_3$, ZPRED accurately predicted its electrokinetic behavior assuming a X_{SP} equal to the radius of water (1.42 Å [68]) and was able to predict the un-folded $(PPG)_{10}$ chains electrokinetic behavior as well assuming a X_{SP}) equal to zero.

INTRODUCTION

Temperature increases the diffusion of molecules in solution, which may alter the way water lingers in the hydration layer. In addition, temperature affects the solubility and water activity of ions, which can have significant effects on the solution properties (e.g. viscosity [56], density [86], relative dielectric [130], etc.). Thus, it seems possible temperature could affect hydration by changing the behavior of the ions interacting with the hydration layer as well. As temperature also affects the structural stability of proteins, measurements were taken around the melting temperatures of the proteins. The same experiments were conducted on BLG at three different pH values as is shown later in **Ch. V** (see **Fig. 5-6**). Three different protein concentrations of BSA were prepared. BSA at twice the desired protein concentration was dissolved in deionized water and filtered as previously described to separate monomers from aggregates. The pH of the water pre-mixing with protein was 6.67. BSA concentrations were determined post-filtration as described in **Ch. II**. Measurements were taken immediately after mixing with salt. 0.5 mL of the filtered protein solutions (containing twice the desired protein concentration) were mixed with 0.5 mL of a 0.2M NaCl solution (containing twice the desired salt concentration) yielding a 1 mL solution containing desired concentrations of both protein and salt. Experimental results for the mixed solutions are discussed below.

RESULTS AND DISCUSSION

Diffusion of BSA in 0.1 M NaCl at Various Temperatures

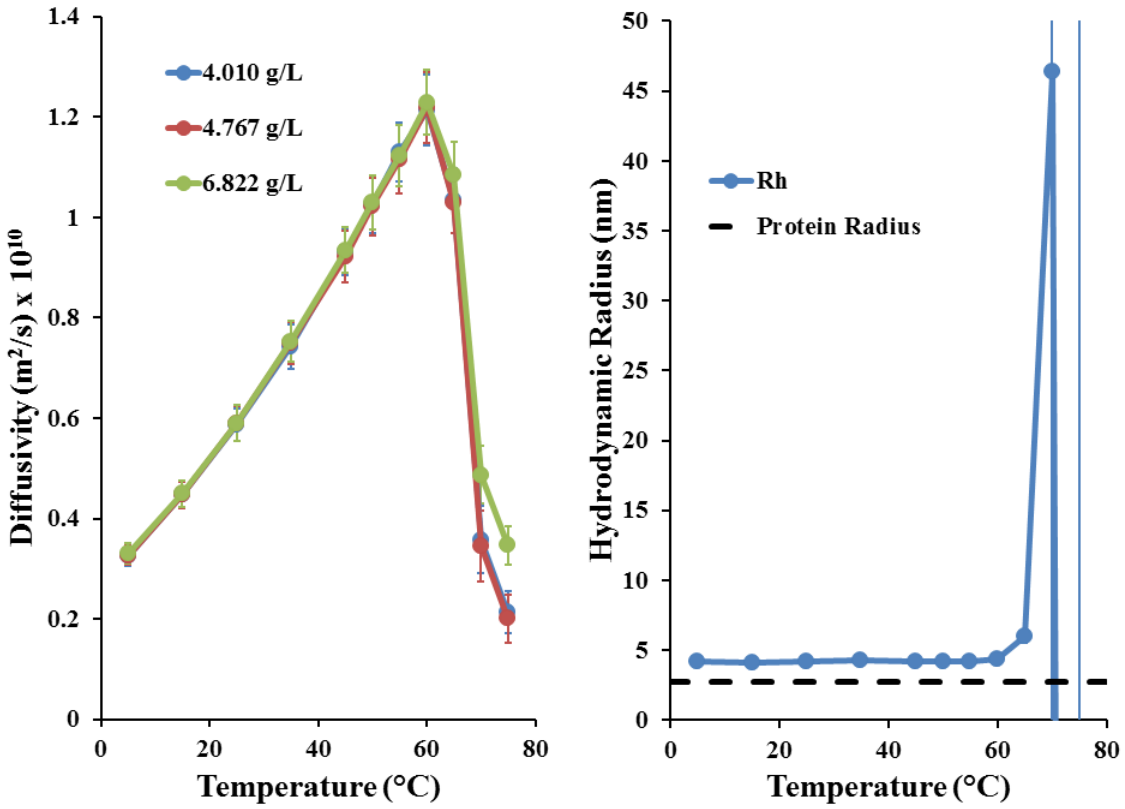

Figure 4-1. Measured Diffusivities and R_h values of BSA in 0.1 M NaCl at Various Temperatures

As shown on the left of **Fig. 4-1**, diffusion of BSA increases with increasing temperature as expected (see **Fig. 2-7**) up to its melting point temperature (~68°C). At 65°C, there is a noticeable decline in diffusion as the structure of BSA changes. This is further evident when viewing the right subplot of **Fig. 4-1**, showing the change in R_h with respect to temperature. Once again, R_h was calculated at each temperature from single protein diffusivities obtained from the y-intercept of the measured diffusivities versus the protein concentration. It is interesting to note, the measured diffusivities did not show

much variation with change in protein concentration, and thus, the measured diffusivities shown are nearly equivalent to the single protein diffusivities at each temperature.

Electrophoresis of BSA in 0.1 M NaCl at Various Temperatures

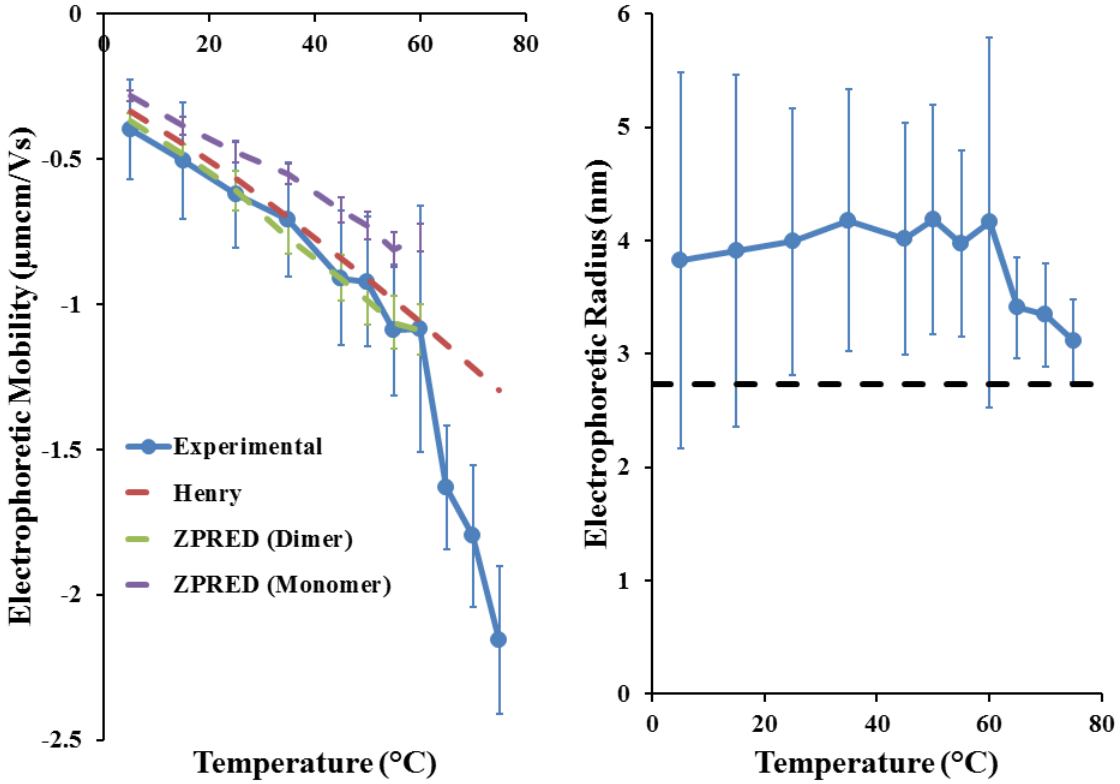

Figure 4-2. Electrophoretic Mobilities and R_e values of BSA in 0.1 M NaCl at Various Temperatures

On the left of **Fig. 4-2**, experimental electrophoretic mobilities are compared (as usual) to two models: the Henry equation and ZPRED. The Henry equation (**Eq. 0-2c** rearranged for electrophoretic mobility) used the average of the measured R_h values from 5°C to 60°C (4.218 nm) for the electrophoretic radius with a charge of -15.5, which was the average charge valence calculated using PROPKA on the ensemble of BSA conformations sampled from molecular dynamics (see **Fig. 2-2**). As can be seen, this

seems to adequately represent BSA in solution. ZPRED was applied to both the monomeric (4f5s A) and dimeric (4f5s) structures of BSA. Computed mobilities of both structures are within experimental error with the dimeric form holding smaller deviation. This is surprising as the Henry equation, which also has comparably small deviation from experimental values, models the monomer. Nonetheless, ZPRED was not able to distinguish the two structures outside of experimental error. On the right of **Fig. 4-2**, R_e values are plotted as a function of temperature and show a decrease in effective size beyond the melting point temperature. Large error bars for the electrophoretic radius are the result of relatively high error to mobility ratio at the lower temperatures (e.g. at 5°C, the electrophoretic mobility was -0.396 ± 0.172, while at 55°C, the mobility was -1.089 ± 0.225).

Analysis/Discussion of BSA in 0.1 M NaCl at Various Temperatures

Comparison of the R_e and R_h of BSA across the temperature range shown in **Fig. 4-3** indicates the X_{SP} coincides with the R_h with variation in temperature, proving the hypothesis of this chapter. Future work will involve computationally modeling the unfolded protein based on previous work [131]. It seems no discernable variation in hydration occurs with variation in temperature. The same outcome was found for BLG as well (see **Fig. 5-6**). This is most likely due to the low resolution of the light scattering methods employed in this work. However, it can be concluded that if there is a change in hydration, it is so small that it makes no difference on the level of a macroscopic measurement.

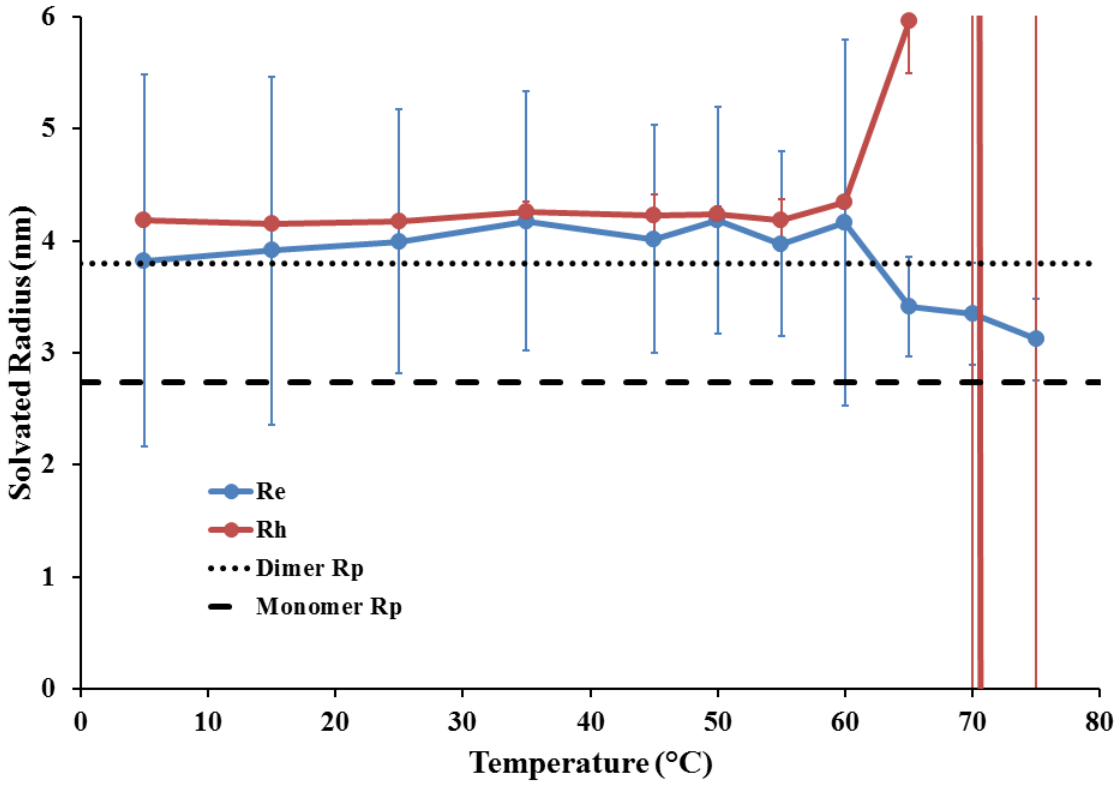

Figure 4-3. Comparison of Experimental Radii of BSA in 0.1 M NaCl at Various Temperatures. Experimental solution conditions can be found in **APPENDIX G**.

Electrophoresis of $[(PPG)_{10}]_3$ in Citrate Phosphate Buffer at Various Temperatures

Experimental electrophoretic mobilities for the melting collagen-like triple helix $[(PPG)_{10}]_3$ were measured in citrate phosphate buffer at pH 7.00 after dialysis of the protein. Measurements were taken over a wide temperature range around the triple helix's melting point (~24°C) to capture the transition in electrophoretic motion of the relatively rigid triple helices and the flexible PPG_{10} chains. Computed values were determined from predicted zeta potentials using the Henry equation for 1k6f (**Eq. 0-2e**) and the Smoluchowski equation (**Eq. 1-2**) for 1k6f A. Please see **Ch. I** for a detailed description of the different electrokinetic models and their use. ZPRED was applied to the $[(PPG)_{10}]_3$ crystal structure (PDB id: 1k6f) at temperatures below 24°C and to an

individual $(PPG)_{10}$ chain at higher temperatures. As the hydrodynamic radii were not determined for fibrillar proteins, a X_{SP} value equal to the radius of a water molecule (1.42 Å [68]) was used for $[(PPG)_{10}]_3$ and a value of zero was used for the $(PPG)_{10}$ chains.

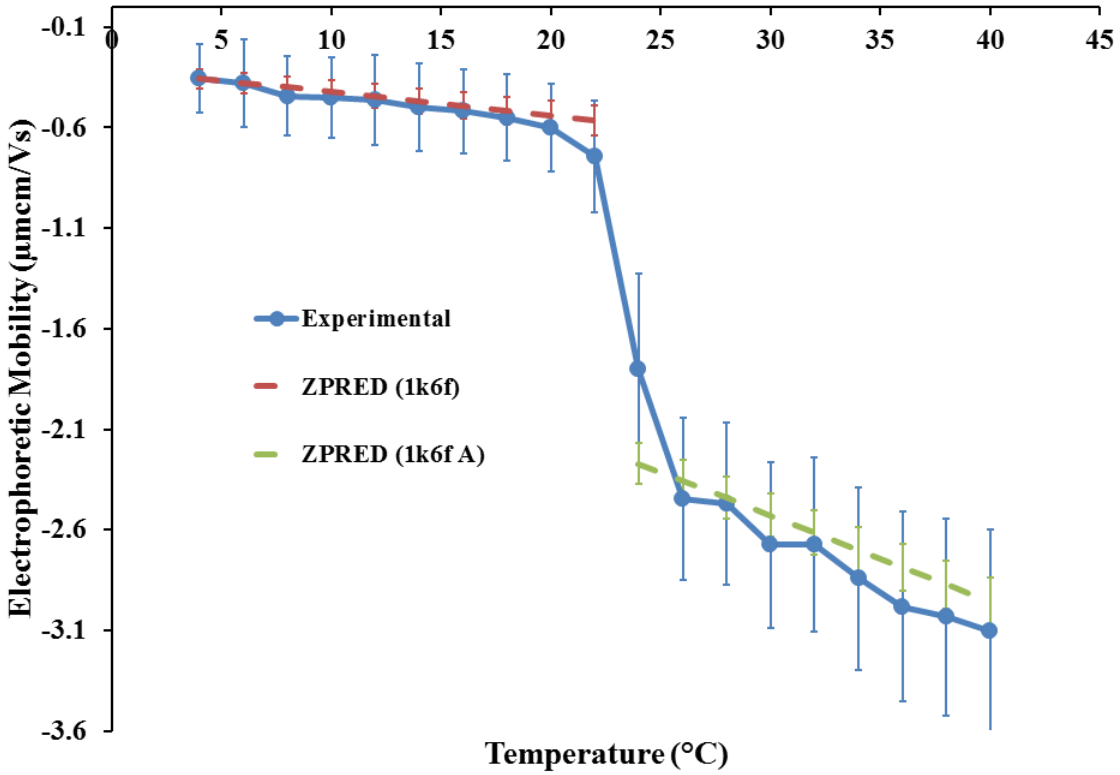

Figure 4-4. . Comparison of Experimental and Computed Electrophoretic Mobilities of the Collagen-like Triple Helix $[(PPG)_{10}]_3$

CONCLUSIONS

With the low resolution of light scattering, hydration appears to remain constant with variation in temperature. The X_{SP} coincides with the R_h for the BSA-NaCl interface for a wide range of temperatures up to its melting point, proving the hypothesis of this chapter.

CHAPTER V

The Impact of pH on Hydration

Chapter Abstract

This chapter assesses hydration with varying pH. Experimental work tested the hypothesis that hydration remains the same during diffusion and electrophoresis using BSA with varying pH at constant ionic strength and temperature. In addition, the combined effects of variation in pH and temperature at constant ionic strength were assessed with BLG. This involved the same measurements and analysis as in the previous two chapters (**Ch. III** and **IV**). Variation in hydration with respect to pH was not discernable due to structural changes of the proteins. Despite both proteins exhibiting changes in structure with variation in pH, the R_h and R_e coincided when transient structures did not contaminate the measurements.

INTRODUCTION

pH is the solution property governing the charge of proteins. As pH changes, the specific adsorption of protons at the protein surface changes and alters the charge of the protein functional groups. Changes in hydration based on hydrogen adsorption alone are most likely too small to measure with the light scattering methods employed in this work. However, pH could affect hydration by changing the charge foundation on which the hydration layer sits. Alterations in charge foundation will change the protein's affinity for ions and water. A wide range of pH values were selected encompassing the isoelectric point of each protein to assess both a positive and negative charge foundation. In addition, it seemed worthwhile to explore the combined effect of pH and temperature. As increasing the temperature increases the internal energy of all the molecules in solution, it seems a combination of altering pH and temperature would exacerbate any alterations that occur in the hydration layer.

RESULTS AND DISCUSSION

Diffusion of BSA in HCl/NaOH and NaCl at Various pH

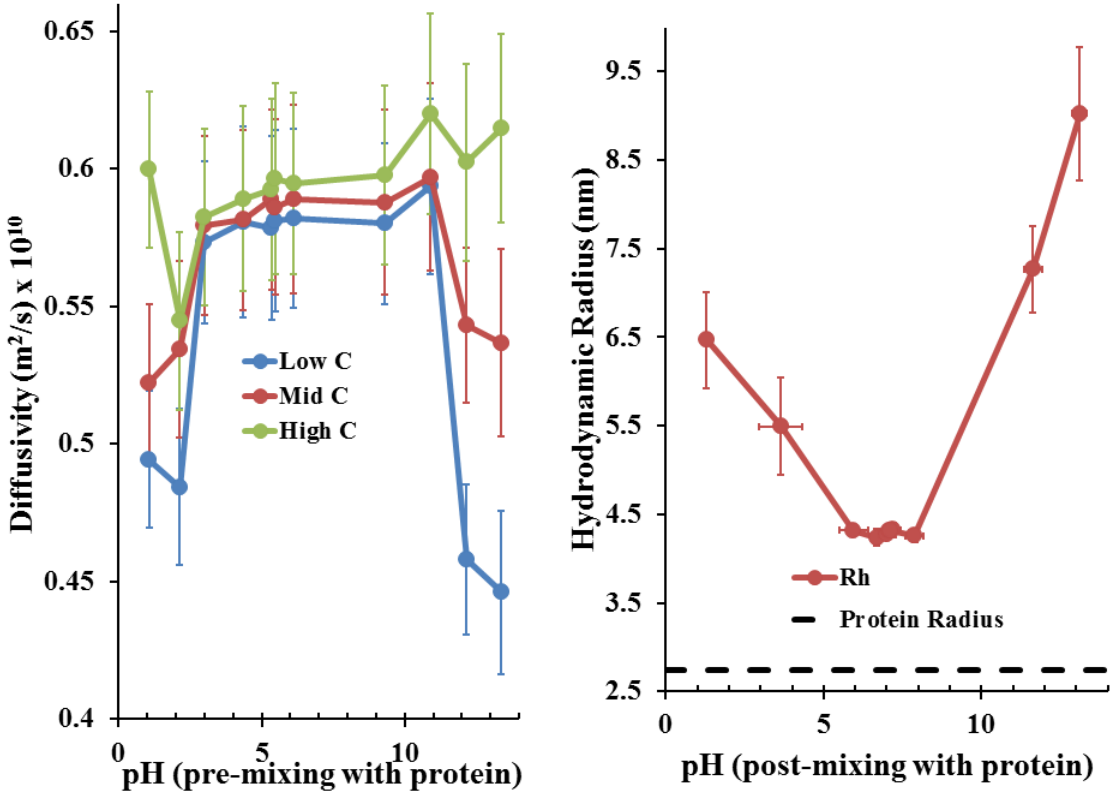

Figure 5-1. . Diffusion and Hydrodynamic Radii of BSA as a Function of pH. (**Left**) Diffusivity is plotted against the measured solution pH that BSA was dissolved in. (**Right**) Hydrodynamic radii are plotted against the re-measured pH of BSA solutions right before taking electrophoretic mobility measurements. Although dilute the protein significantly buffered the pH of the solution as can be seen.

Measured diffusivities of BSA (**Fig. 5-1**) are in agreement with previous experimental results [104, 105, 107]. There appears to be a change in structure at the extreme pH values, but much work has shown the structure of BSA at low pH with the salt concentrations used (0.1 M) to remain monomeric [104, 132, 133]. It is believed the structure becomes partially unfolded (or expanded) causing an increase in the friction experienced at its interface. The nature of the BSA structure and its interface will be

discussed further following the presentation of its electrophoretic behavior, which also captured this structural change.

Electrophoresis of BSA in HCl/NaOH and NaCl at Various pH

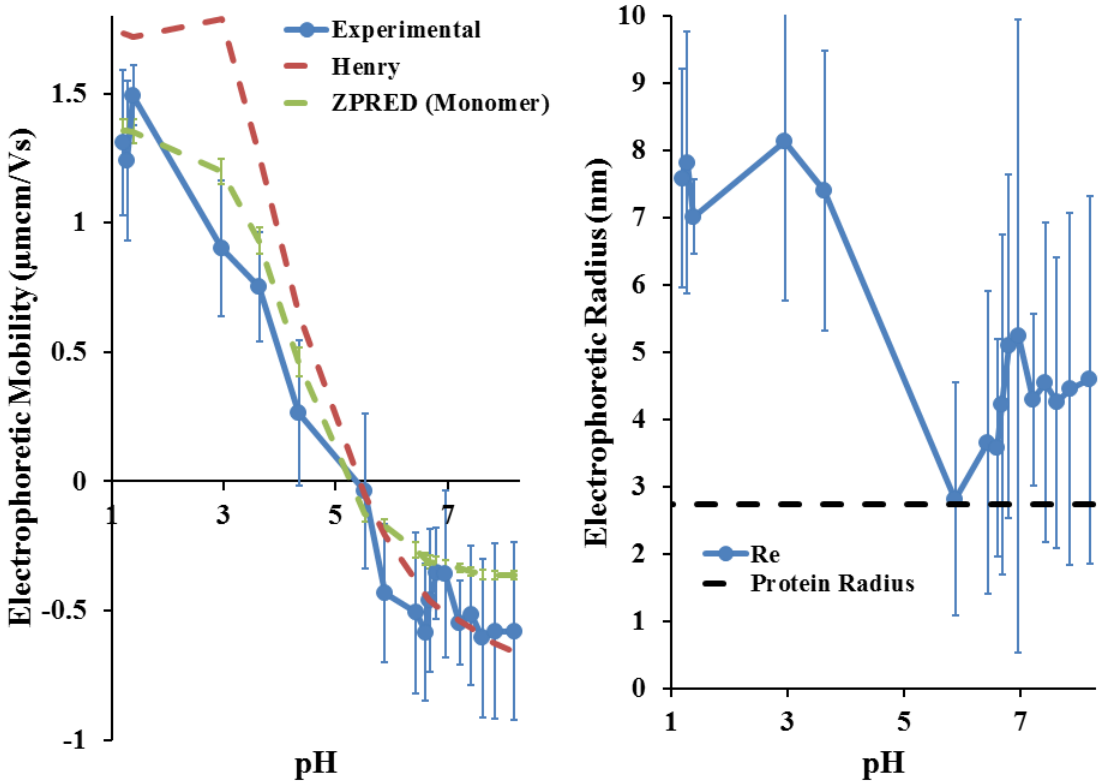

Figure 5-2. Electrophoretic Mobilities and Electrophoretic Radii of BSA as a Function of pH. Experimental values are plotted against BSA solution pH values that were measured immediately before mobility measurements. Measured pH values can be found in **APPENDIX G** with the experimental solution conditions.

With an isoelectric point around 5.4 (see **Fig. 2-2**), both positive and negative charge foundations were assessed with the pH range of this study. As shown on the left of **Fig. 5-2**, experimental mobilities are compared to two different models as usual. The Henry equation (**Eq. 0-2c** re-arranged for electrophoretic mobility) used R_e value equal to the measured R_h values plotted in **Fig. 5-1**. ZPRED was applied to the A chain of 4f5s

and yielded good agreement with experimental results when using a X_{SP} defined by HYDROPRO (3.6067 - 2.7378 nm). This was surprising considering how large the experimental R_h values were. Nonetheless, ZPRED output supports previous work indicating the BSA structure to remain monomeric at low pH and relatively high salt concentrations [104, 132, 133]. It seems the ensemble of folded structures used by ZPRED hold a similar interface as the actual structure in these conditions.

Analysis/Discussion of BSA in HCl/NaOH and NaCl at Various pH

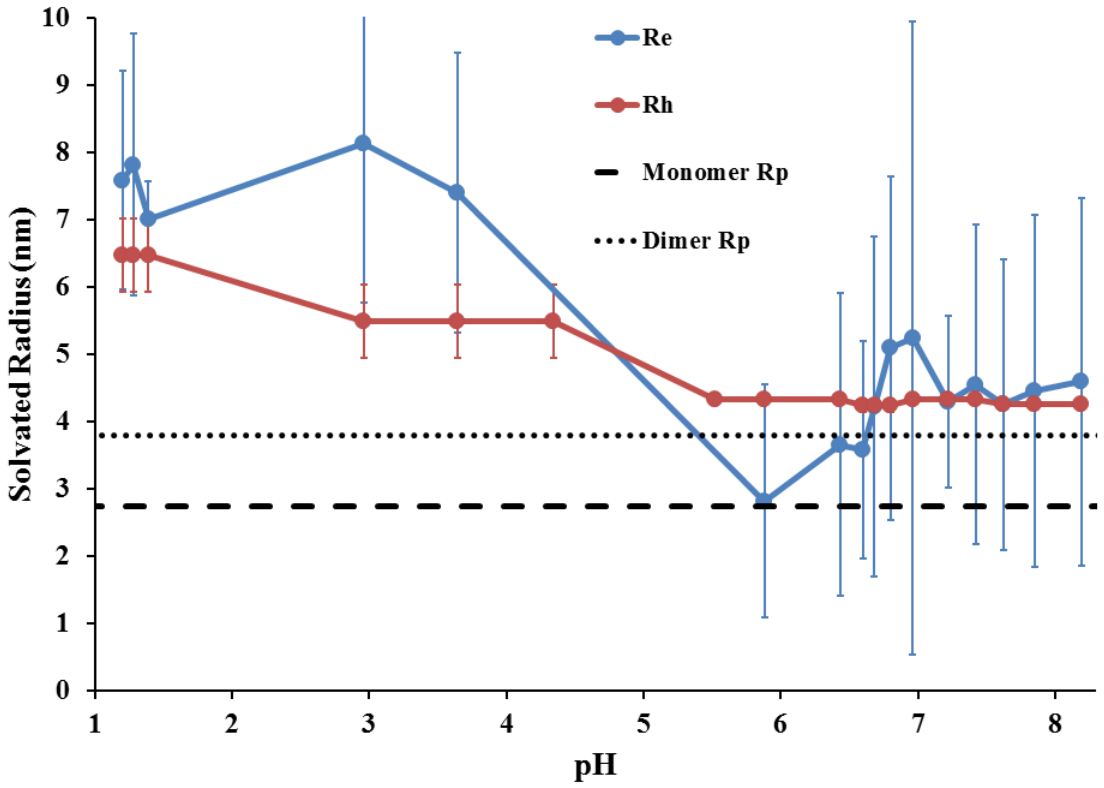

Figure 5-3. Comparison of Experimental Radii of BSA as a Function of pH. Values are plotted against BSA solution pH values that were measured immediately before mobility measurements.

Figure 5-3 shows the R_h and R_e coincide for most of the pH range studied. The increase in apparent size at the lower pH values (or equivalently increase in friction coefficient) has been well documented [104, 132, 133] and is most likely the result of a partially unfolded state. Nonetheless, the state of this structure was apparently captured during both diffusive and electrophoretic measurements.

Diffusion of BLG in HCl and NaCl at Various pH and Temperatures

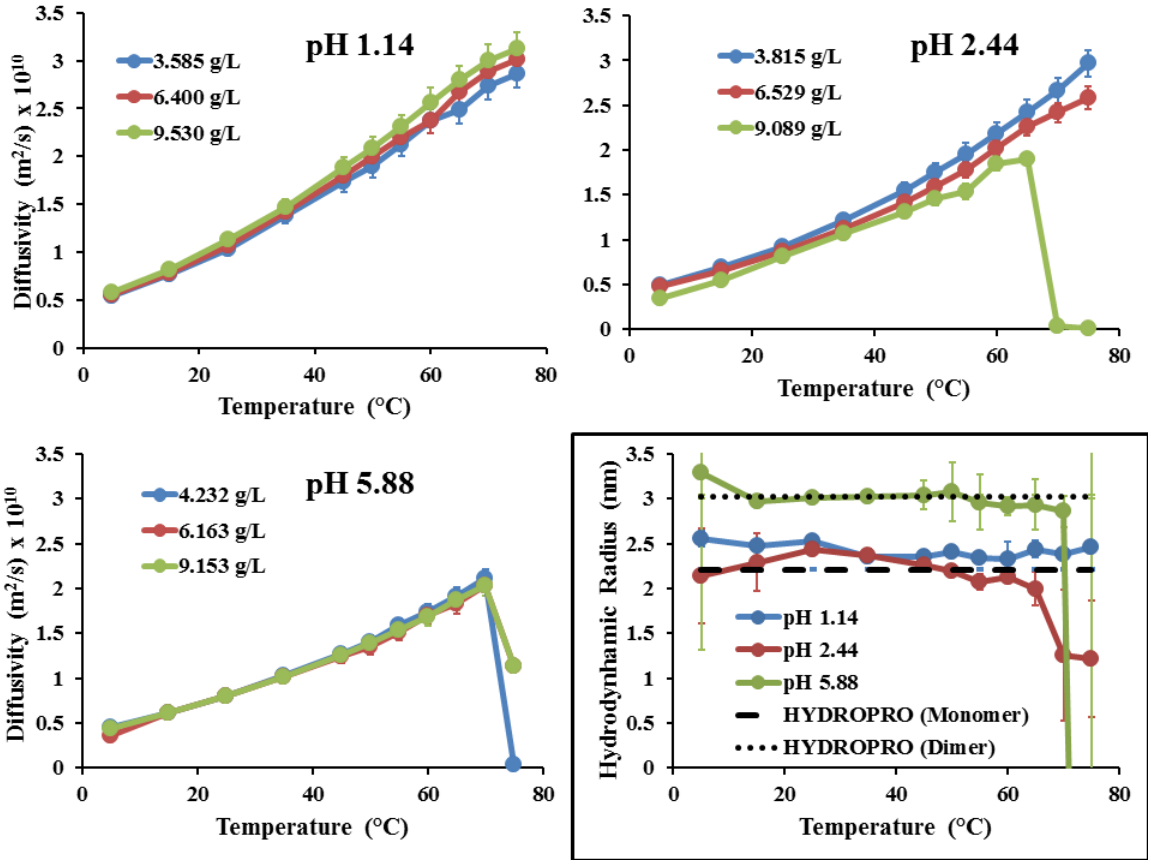

Figure 5-4. Measured Diffusivities and Hydrodynamic Radii of BLG as a Function of Temperature and pH. pH values shown are the measured pH of solution post-mixing with protein. These final equilibrium pH values were measured right before measurement of the electrophoretic mobility approximately two hours after dissolving BLG in solution.

As expected, diffusion of BLG (**Fig. 5-4**) increases with increasing temperature up to its melting point just like with BSA (see **Fig. 4-1**). Comparing the R_h values

obtained at the different pH values shows a structural change from monomer to dimer as the pH was raised past the isoelectric point to 5.88. At the two lower pH values (1.14 and 2.44), it seems the BLG monomer is more structurally stable at pH 1.14 as it maintained a constant R_h across the temperature range while at pH 2.44 a noticeable decrease occurs as the temperature approaches its melting temperature.

Electrophoresis of BLG in HCl and NaCl at Various pH and Temperatures

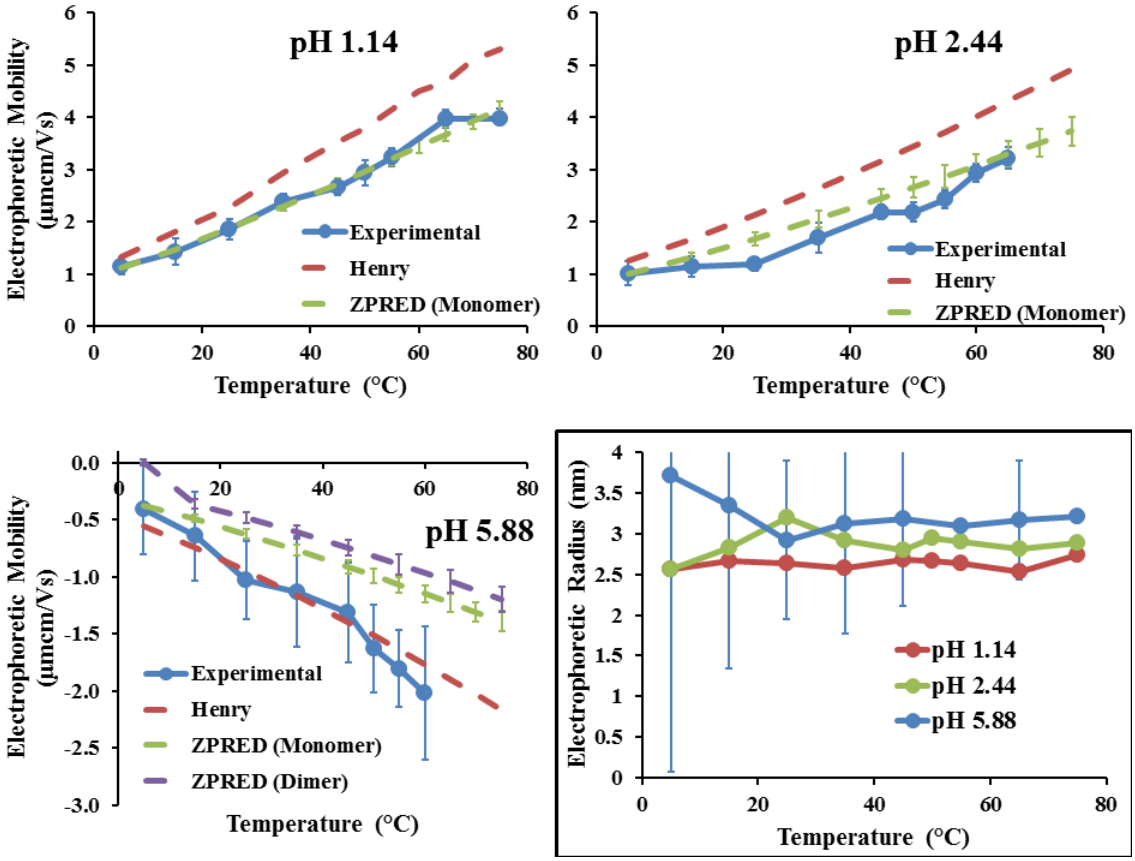

Figure 5-5. Electrophoretic Mobilities and Electrophoretic Radii of BLG as a Function of Temperature and pH. pH values were measured right before measurement of the electrophoretic mobility approximately two hours after dissolving BLG in a 0.1M HCl solution.

With an isoelectric point of 5.39 in water [116], the span of pH used in this experiment captured the electrophoretic behavior of both a positive and negative BLG

interface. **Figure 5-5** shows the expected trend of an increasing electrophoretic mobility magnitude with increasing temperature.

Analysis/Discussion of BLG in HCl and NaCl at Various pH and Temperatures

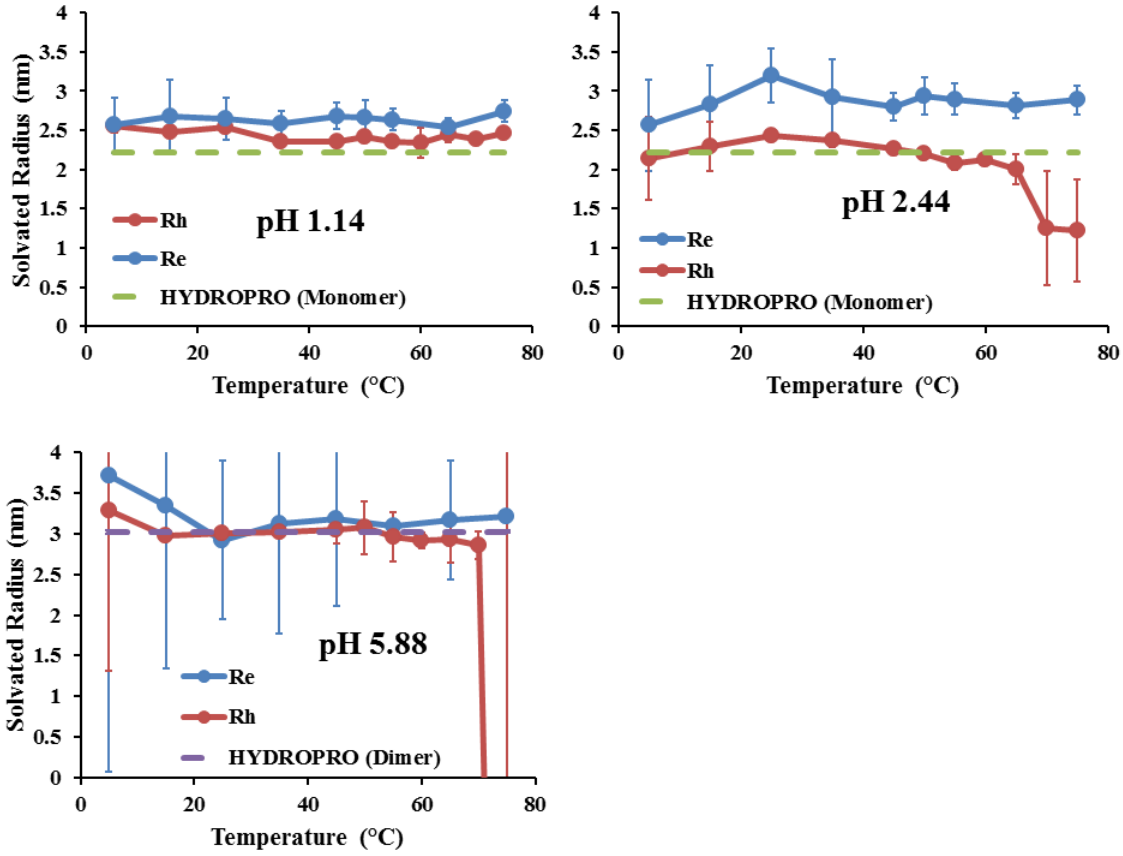

Figure 5-6. Comparison of Experimental Radii of BLG in HCl and NaCl at Various pH and Temperatures

In **Fig. 5-6**, comparison of the experimentally determined radii shows much coincidence of the X_{SP} and R_h across the temperature span at pHs 1.14 and 5.88. However, at pH 2.44 the two radii mostly differ (except at the lower temperatures) with the R_e showing a greater hydration layer thickness the R_h. There is much work that has been done in assessing the aggregation behavior of BLG around these low pH values [118] and thus this difference in hydration may be the result of temporary dimers that

form and fall apart contaminating the measurement signals. At pH 5.88, both diffusion and electrophoresis exhibit signals dominated by the presence of dimers, which allowed the coincidence of the R_e and R_h to be clearly seen. As a function of pH, the hydration of BLG seems to only be equivalent during diffusion and electrophoresis at the lower temperatures.

CONCLUSIONS

This chapter has demonstrated the X_{SP} coincides with the R_h with variation in pH for monodisperse stable structures. Due to structural changes with BSA, it was not possible to discern the variation that may occur in the hydration layer with varying pH. Future work will involve use of a higher resolution instrument on polystyrene beads, which do not experience a structural change in the charge foundation with change in pH.

CHAPTER VI

Additional Experimental Validation and Applications of ZPRED

Chapter Abstract

This chapter includes additional experimental data for further validation of ZPRED. Electrophoretic mobility data from external sources were acquired using a graph digitizer. In addition, to show the utility of ZPRED in protein design, experimental electrophoretic mobilities of mutated structures of green fluorescent protein were compared to computed values. ZPRED was capable of predicting the mobilities of most of the data sets with deviation most likely resulting from the presence of aggregates in the experimental data.

INTRODUCTION

To obtain further validation of the accuracy of ZPRED, electrophoretic mobility data from external sources were obtained. Only sources providing their experimental solution conditions could be modeled.

RESULTS AND DISCUSSION

Electrophoresis of BSA in Citrate Phosphate Buffer at Various pH

In **Fig. 6-1**, the electrophoretic mobilities of an un-filtered BSA solution are shown. Unfortunately, I measured this data before perfecting my electrophoresis procedure (see **Ch. II** for details). As larger particles scatter more light than smaller particles, the mobility measurements are most likely contaminated by the presence of aggregates. Nonetheless, ZPRED accurately computed the electrophoretic mobilities measured above the isoelectric point (IEP) of BSA. This indicates BSA maintained a monomeric structure above its IEP.

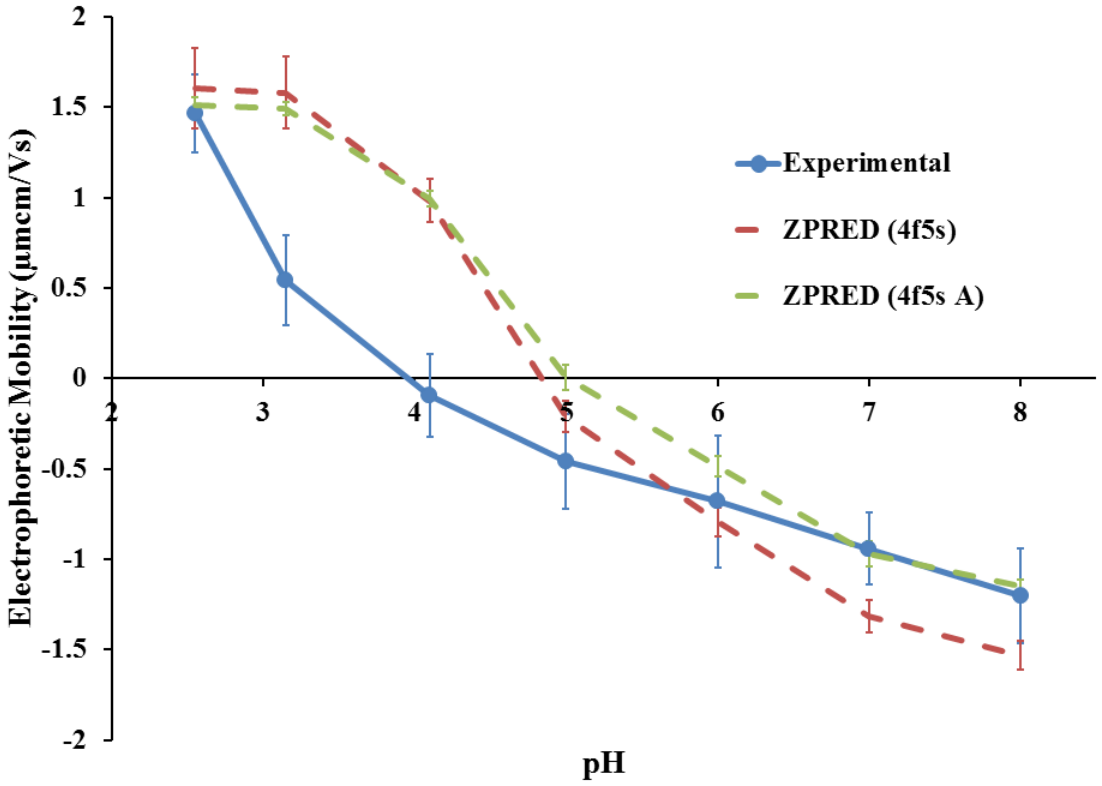

Figure 6-1. Comparison of Experimental and Computed Electrophoretic Mobilities of BSA in Citrate Phosphate Buffer at 20°C. Experimental values for bovine serum albumin (BSA) (PDB id: 4f5s) at a concentration of 5 mg/mL and 20°C show the typical trend of decreasing mobility with increasing pH. Sodium citrate phosphate buffer (see **APPENDIX G** for solution conditions) was used to tailor the pH and may have induced a specific ion effect as shown by the shifted isoelectric point (IEP). The actual IEP of BSA is 4.68 in water; whereas it was found to be about 4.00 in our buffer and calculated to be 4.69 by ZPRED, which predicts the isoelectric point when determined mobilities are less than or equal to the variance from the different structures. Computed values were determined by ZPRED using experimental hydrodynamic radii for 4f5s and HYDROPRO computed values for 4f5s A. At pH below the IEP, BSA seems to experience a specific ion effect with phosphate, citrate or possibly both and deviates from computed values being the computation does not account for these effects. At pH above the IEP, BSA is negative and no longer experiences a specific ion effect with the anions. In this pH range, agreement between computed and experimental values can be seen.

Electrophoresis of BLG at Various pH

Figure 6-2 compares experiment to ZPRED computation using two different X_{SP} values: the diameter of a water molecule and that calculated by HYDROPRO. Deviation

between experimental and computed values occurs around the isoelectric point of BLG, where higher order aggregates most likely produced the dominate light scattering signal. The external data source [124] did not filter their protein solution before measurement, and consequently, the mobility measured is most likely contaminated to some extent by aggregation, especially around the IEP. This is further supported by the deviation in charge valence seen in **Fig. 2-3.** Nonetheless, at pH values far from the IEP, agreement between computed and experimental values is achieved; indicating the BLG dimeric structure becomes predominately monodisperse in solution and makes up the majority of the measurement signal during electrophoresis.

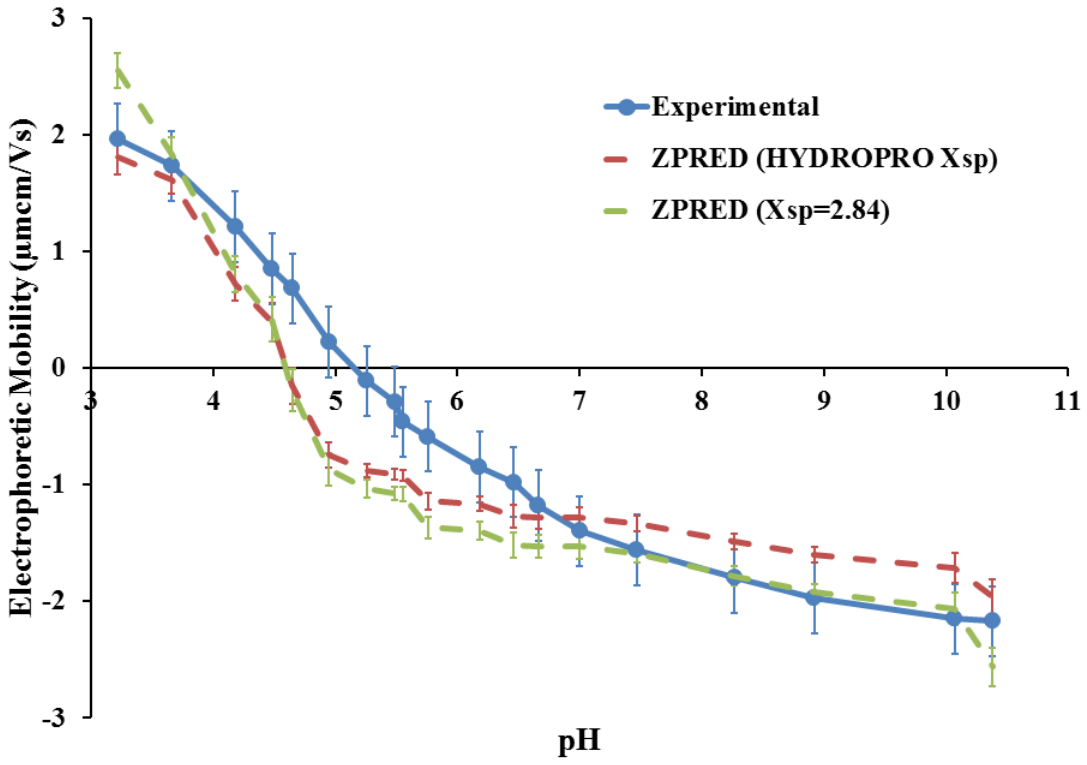

Figure 6-2. Computational Versus Experimental Electrophoretic Mobilities of Bovine β-Lactoglobulin (Data Source: [124]). Experimental measurements were conducted at 20°C. Values for the standard deviation error were determined as the average error associated with the compilation of BLG data in KCl (see **Fig. 3-9**).

Electrophoresis of LYZ at Various pH

The experimental data shown in **Fig. 6-3** also comes from an un-filtered protein solution [100]. Aggregates most likely dominated the measurement signal across most of the pH range [96], and as a result, ZPRED predicted electrophoretic mobilities that were not seen experimentally.

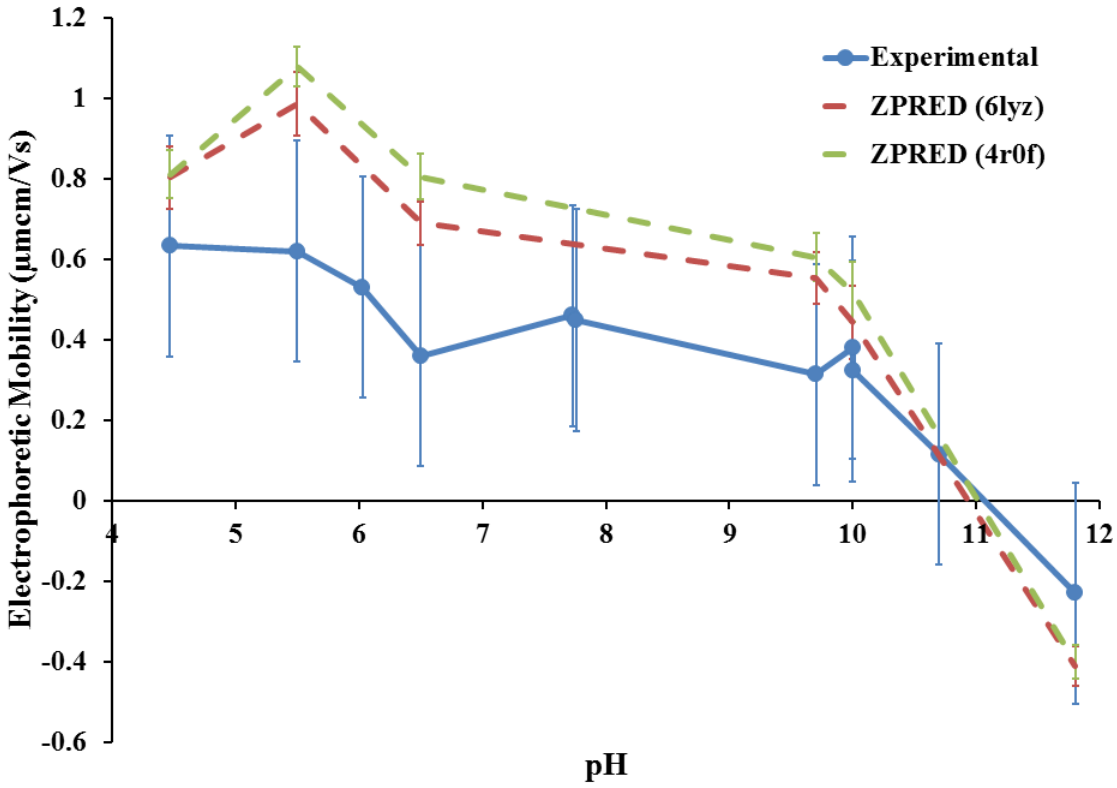

Figure 6-3. Computational Versus Experimental Electrophoretic Mobilities of Hen Egg White Lysozyme (Data source: [100]). Experimental measurements were conducted at 20°C. Values for the standard deviation error were determined as the average error associated with the compilation of LYZ data in various salts (see **Fig. 3-3**). For the data point at pH 10, the small concentration of glycine (0.02M) was substituted by $NaCH_3COO$ instead due to lack of solution properties data.

Application for Protein Design: Prediction of Mutant Behavior

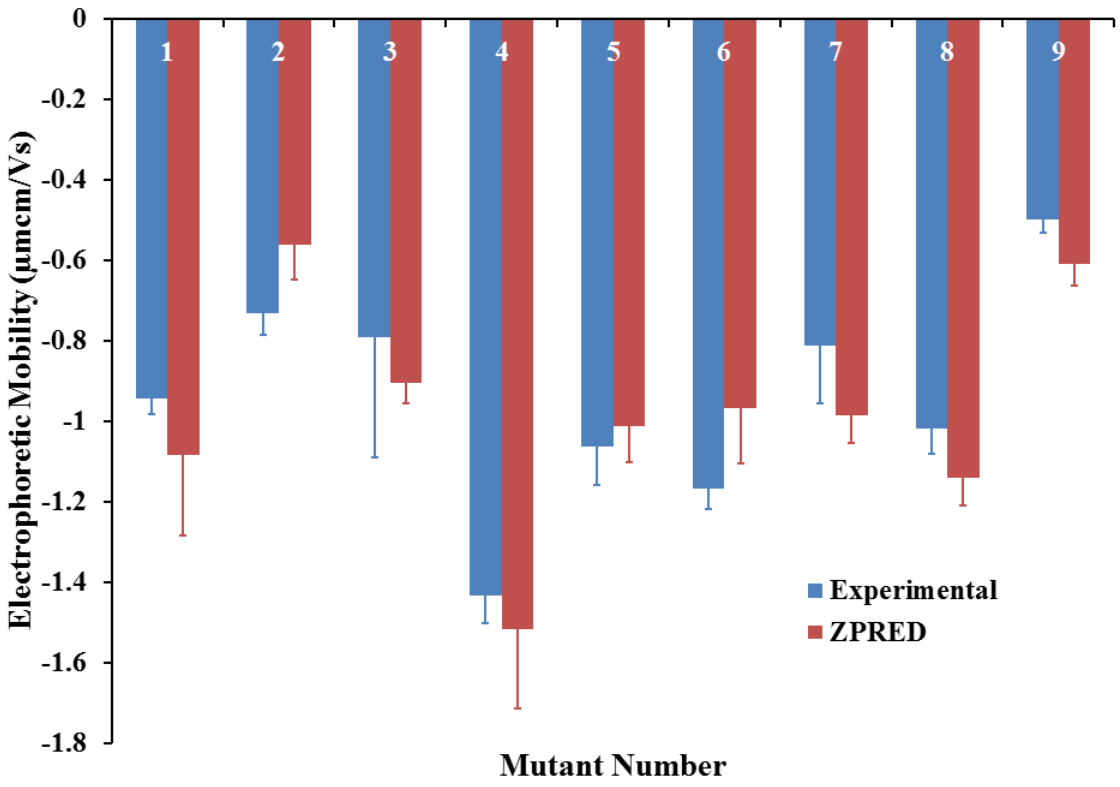

Figure 6-4. Comparison of Experimental and Computational Electrophoretic Mobilities of Green Fluorescent Protein. Experimental mobilities were measured at 25°C in a 50mM Na3Citrate buffer at a pH of 6.00. A key showing the residue mutations for each mutant number is in **APPENDIX E**. Computed values were determined from predicted zeta potentials using appropriate electrokinetic models and computed hydrodynamic radii. The zeta potential prediction protocol was applied to mutated structures generated from the wild type crystal structure (PDB id: 2y0g). As shown, prediction within experimental error can be achieved.

In **Fig. 6-4**, experimental mobilities of green fluorescent protein (GFP) mutants are compared to ZPRED computed values, which show good agreement within error. Thus, ZPRED seems ready for protein design.

CONCLUSIONS

ZPRED was unable to compute electrophoretic mobilities in agreement with experimental for the proteins solutions that were most likely polydisperse. Polydispersity is a well-known issue for proteins, especially around their isoelectric points, where charge repulsion is significantly diminished. However, at pH values outside of the IEP range, agreement between computation and experiment was achieved for BSA and BLG. As higher order aggregates are known to contaminate LYZ solutions [95], experimental values coming from an un-filtered LYZ solution were expected to deviate from the computation of ZPRED, which was applied to the monomeric structure (6lyz). Based on the accuracy achieved in predicting the mobilities of GFP mutants, it seems ZPRED is ready to be applied to protein design.

FINAL CONCLUSIONS & FUTURE DIRECTIONS

This work has demonstrated that a good approximation of the slip plane position defining the zeta potential of a globular protein is the solvation edge defined by the Stokes-Einstein hydrodynamic radius. Based on the $(PPG)_{10}$ data (see **Fig. 4-4**), the X_{SP} of a rigid cylindrical protein (the collagen-like triple helix, $[(PPG)_{10}]_3$) can be approximated as the radius of a water molecule (0.142 nm [17]), while more flexible fibrillar proteins (i.e. the $(PPG)_{10}$ chains) have a slip plane position coincident with the protein surface. With knowledge of the slip plane position, a protocol for predicting the zeta potential entirely from structure has been developed and experimentally verified by this work. This protocol is dependent on the EDL model employed, which for the case of all the data presented in this work was a modified Gouy-Chapman EDL (**Fig. 0-1**). Thus, future work will involve testing the ζ protocol with a Gouy-Chapman-Stern EDL [76, 18] to model specific ion effects like the shift in the isoelectric point of BSA shown in **Fig. 6-1**. Additional future work will involve testing coagulation theories (specifically the Eilers and Korff rule [134]) to assess how accurately the ζ truly defines the dispersion stability of an electrostatically stabilized solution.

At pH 2, β-lactoglobulin provides a good example of a dimerization process to model. Experimentally, this would involve performing DLS in acidic conditions and increasing the KCl concentration from 0.01 M to 1.0 M (similarly to **Fig. 3-7**). However, instead of repeating my experiment, a smaller KCl concentration increment (maybe 0.05 M) should be used to allow for a more accurate analysis. Once experimental correlograms are obtained at each ion concentration, they can be analyzed by principal component analysis (PCA) to extract the monomer and dimer signal components from the average

decay rate [135]. Analysis of the monomer and dimer contributions will allow for determination of the ion concentration, which induces dimerization of BLG (i.e. the critical coagulation concentration). With the critical coagulation concentration (CCC) defined, the Hamaker constant of the BLG colloidal system can be determined and then used to analyze the system in terms of DLVO theory [1, p. 240].

PUBLICATIONS

D. Grisham and V. Nanda, "Hydrodynamic Radius Coincides with the Slip Plane Position in Electrokinetic Behavior of Lysozyme", PROTEINS: Structure, Function and Bioinformatics, Jan. 2018 [ACCEPTED; Prot-00383-2017.R1]

A.S. Parmar, J.K. James, D.R. Grisham, D.H. Pike et V. Nanda, "Dissecting Electrostatic Contributions to Folding and Self-Assembly Using Designed Multicomponent Peptide Systems", Journal of the American Chemical Society, 138, 4362-4367, 2016

K.E. Drzewiecki, D.R. Grisham, A.S. Parmar, V. Nanda et D.I. Shreiber, "Circular Dichroism Spectroscopy of Collagen Fibrillogenesis: A New Use for an Old Technique", Biophysical Journal, 111, 2377-2386, 2016

PUBLICATIONS IN PROGRESS

D. Grisham and V. Nanda, "Anions of the Hofmeister Series Affect Diffusion Not Solvation", PROTEINS: Structure, Function and Bioinformatics [IN PROGRESS]

D. Grisham and V. Nanda, "A Protocol for Predicting the Zeta Potential of Proteins from their Crystal Structures", Journal of the American Chemical Society, [IN PROGRESS]

D. Grisham and V. Nanda, "A Test of the Eilers and Korff Rule: the Dimerization of β-Lactoglobulin", Journal of Colloid and Interface Science, [IN PROGRESS]

APPENDIX A. Mathematical Expressions, Physical Constants & Vector Operators

The contents of this appendix are self-explanatory and used by ZPRED (see **Ch. I**).

- **Avogadro's number** [136, p. Backcover]

$$N_A = 6.022140857 \times 10^{23} mol^{-1}$$

- **Boltzmann constant** [136, p. Backcover]

$$k_b = \frac{R}{N_A} = 1.38064852 \times 10^{-23} \frac{J}{K}$$

- **electron charge** [137, p. 712]

$$e = -1.6021766208 \times 10^{-19} C$$

- **Exponential Integral** [138, 139, 140]. Due to its importance in electrokinetic modeling, its definition and approximation is presented here.

$$E_n(x) = \int_1^\infty \frac{e^{-xt}}{t^n} dt$$

$$E_{n+1}(x) = \frac{1}{n}[e^{-x} - xE_n(x)]$$

  - Approximation

$$E_1(x) \cong \begin{cases} a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4 + a_5 x^5 - \ln(x) & 0 \le x \le 1 \\ \dfrac{e^{-x}(x^4 + b_1 x^3 + b_2 x^2 + b_3 x + b_4)}{x(x^4 + c_1 x^3 + c_2 x^2 + c_3 x + c_4)} & 1 \le x < \infty \end{cases}$$

$$\begin{array}{lll} a_0 = -0.57721566 & & \\ a_1 = 0.99999193 & b_1 = 8.5733287401 & c_1 = 9.5733223454 \\ a_2 = -0.24991055 & b_2 = 18.0590169730 & c_2 = 25.6329561486 \\ a_3 = 0.05519968 & b_3 = 8.6347608925 & c_3 = 21.0996530827 \\ a_4 = -0.00976004 & b_4 = 0.2677737343 & c_4 = 3.9584969228 \\ a_5 = 0.00107857 & & \end{array}$$

The deviation error for $0 \le x \le 1$ is 2×10^{-7}, and for $1 \le x < \infty$ is 2×10^{-8}.

- **Faraday's constant** [141, p. 201]

$$F = eN_A$$

- **gas constant** [137, p. 1371]

$$R = 8.3144598\ {}^{J}/_{K \cdot mol}$$

- **permeability of free space** [137, p. 1371]

$$\mu_o = 4\pi \times 10^{-7}\ {}^{T \cdot m}/_{A}$$

- **π**

$$\pi = 3.14159265358979323846$$

- **Reynolds Number** [72]

$$Re = \frac{\rho R_H v}{\eta}$$

ρ is mass density of the solution [kg/L]

R_H is the effective hydrodynamic radius [m]

v is velocity of particle [m/s]

η is the solution viscosity [Pa s]

- **speed of light (in vacuum)** [136, p. Backcover]

$$c = 299792458\,\frac{m}{s}$$

- **vacuum permittivity (permittivity of free space, electric constant)** [142, p. 160] [137, pp. 712, 1371]

$$\varepsilon_o = \frac{1}{\mu_o c^2} = 8.85418781762 \times 10^{-12}\,\frac{F}{m}$$

Vector Operators

The following definitions use the Cartesian, Cylindrical, and Spherical vectors, respectively:

$$F(x, y, z) = F_x\hat{\imath} + F_y\hat{\jmath} + F_z\hat{k}$$

$$F(r,\theta,z) = F_r\hat{r} + F_\theta\hat{\theta} + F_z\hat{z}$$

$$F(r,\theta,\varphi) = F_r\hat{r} + F_\theta\hat{\theta} + F_\varphi\hat{\varphi}$$

- **Curl (or Circulation)**

$$\nabla \times F(x,y,z) = \begin{vmatrix} \hat{\imath} & \hat{\jmath} & \hat{k} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ F_x & F_y & F_z \end{vmatrix} = \left(\frac{\partial F_z}{\partial y} - \frac{\partial F_y}{\partial z}\right)\hat{\imath} + \left(\frac{\partial F_x}{\partial z} - \frac{\partial F_z}{\partial x}\right)\hat{\jmath} + \left(\frac{\partial F_y}{\partial x} - \frac{\partial F_x}{\partial y}\right)\hat{k}$$

$$\nabla \times F(r,\theta,z) = \begin{vmatrix} \hat{r} & \hat{\theta} & \hat{z} \\ \frac{\partial}{\partial r} & \frac{\partial}{\partial \theta} & \frac{\partial}{\partial z} \\ F_r & F_\theta & F_z \end{vmatrix} = \left(\frac{\partial F_z}{\partial \theta} - \frac{\partial F_\theta}{\partial z}\right)\hat{r} + \left(\frac{\partial F_r}{\partial z} - \frac{\partial F_z}{\partial x}\right)\hat{\theta} + \left(\frac{\partial F_\theta}{\partial r} - \frac{\partial F_r}{\partial \theta}\right)\hat{z}$$

$$\nabla \times F(r,\theta,\varphi) = \begin{vmatrix} \hat{r} & \hat{\theta} & \hat{\varphi} \\ \frac{\partial}{\partial r} & \frac{\partial}{\partial \theta} & \frac{\partial}{\partial \varphi} \\ F_r & F_\theta & F_\varphi \end{vmatrix} = \left(\frac{\partial F_\varphi}{\partial \theta} - \frac{\partial F_\theta}{\partial \varphi}\right)\hat{r} + \left(\frac{\partial F_r}{\partial \varphi} - \frac{\partial F_\varphi}{\partial x}\right)\hat{\theta} + \left(\frac{\partial F_\theta}{\partial r} - \frac{\partial F_r}{\partial \theta}\right)\hat{\varphi}$$

- **Divergence**

$$\nabla \cdot F(x,y,z) = \frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y} + \frac{\partial F_z}{\partial z}$$

$$\nabla \cdot F(r,\theta,z) = \frac{1}{r}\frac{\partial (rF_r)}{\partial x} + \frac{1}{r}\frac{\partial F_\theta}{\partial \theta} + \frac{\partial F_z}{\partial z}$$

$$\nabla \cdot F(r,\theta,\varphi) = \frac{1}{r^2}\frac{\partial (r^2F_r)}{\partial x} + \frac{1}{r\sin(\theta)}\frac{\partial (\sin(\theta)\,F_\theta)}{\partial \theta} + \frac{1}{r\sin(\theta)}\frac{\partial F_\varphi}{\partial \varphi}$$

- **Gradient**

$$\nabla F(x,y,z) = \frac{\partial F}{\partial x} + \frac{\partial F}{\partial y} + \frac{\partial F}{\partial z}$$

$$\nabla F(r,\theta,z) = \frac{\partial F}{\partial r} + \frac{1}{r}\frac{\partial F}{\partial \theta} + \frac{\partial F}{\partial z}$$

$$\nabla F(r,\theta,\varphi) = \frac{\partial F}{\partial r} + \frac{1}{r}\frac{\partial F}{\partial \theta} + \frac{1}{r\sin(\theta)}\frac{\partial F}{\partial \varphi}$$

- **Laplacian**

$$\nabla \cdot \nabla F(x,y,z) = \nabla^2 F(x,y,z) = \frac{\partial^2 F}{\partial x^2} + \frac{\partial^2 F}{\partial y^2} + \frac{\partial^2 F}{\partial z^2}$$

$$\nabla \cdot \nabla F(r,\theta,z) = \nabla^2 F(r,\theta,z) = \frac{1}{r}\frac{\partial\left(r\frac{\partial F}{\partial r}\right)}{\partial r} + \frac{1}{r^2}\frac{\partial^2 F}{\partial \theta^2} + \frac{\partial^2 F}{\partial z^2}$$

$$\nabla \cdot \nabla F(r,\theta,\varphi) = \nabla^2 F(r,\theta,\varphi) = \frac{1}{r^2}\frac{\partial\left(r^2\frac{\partial F}{\partial r}\right)}{\partial r} + \frac{1}{r^2\sin(\theta)}\frac{\partial\left(\sin(\theta)\frac{\partial F}{\partial \theta}\right)}{\partial \theta} + \frac{1}{r^2\sin^2(\theta)}\frac{\partial^2 F}{\partial \varphi^2}$$

APPENDIX B. The Debye Length

Due to its common use in electrokinetics, a complete definition of the Debye length including its derivation is presented in this appendix. To define the Debye length (κ^{-1}), the molecule-solvent interface is modeled as an electric double layer (EDL). An EDL is the collection of solvation layers that form around a charged molecule in an attempt to neutralize the molecule's charge. A number of EDL models have been proposed over the past century [143]. The simplest and oldest model was developed by Helmholtz and referred to by Perrin in 1904 [144]. It involves two stagnant, charged layers in parallel planes forming a molecular condenser [1, p. 22]. Although the Helmholtz EDL model is inadequate for electrokinetic interfaces in solution, it was the starting point of EDL modeling. The EDL model pertinent to the definition of the κ^{-1} is the Gouy-Chapman (GC) EDL. Gouy in 1910 and Chapman in 1913 [12] developed a model, where a particle with a uniform surface charge is neutralized by a diffuse region of ions encompassing the molecular surface. At equilibrium, the particle's surface potential ($\boldsymbol{\psi_o}$) propagates outward into the cloud of ions treated as point charges immersed in a solvent with a constant relative dielectric ($\boldsymbol{\varepsilon_r}$). The zeta potential of the particle is typically less than the surface potential and is located at the slip plane, which is somewhere in the diffusive cloud of ions less than a Debye length away from the surface. The propagation of the surface potential into solvent is defined by Poisson's equation (**Eq. B1**)

$$\nabla^2\boldsymbol{\psi} = -\frac{\boldsymbol{\rho_c}}{\boldsymbol{\varepsilon_o\varepsilon_r}} \qquad \textbf{(B1)}$$

$$\boldsymbol{\rho_c} = \sum_{i=1}^{N} \boldsymbol{n_i z_i e} \qquad \textbf{(B1a)}$$

where $\boldsymbol{\psi}$ is electrostatic potential, $\boldsymbol{\rho_c}$ is the volume charge density defined in **Eq. B1a**, $\boldsymbol{\varepsilon_o}$ is **vacuum permittivity**, $\boldsymbol{N}$ is the number of ions in solution, $\boldsymbol{n_i}$ is the number of i-th ions per unit volume, $\boldsymbol{z_i}$ is the i-th ion valence, $\boldsymbol{e}$ is **electron charge**

At equilibrium the electric and diffusion forces acting on each ion must balance (i.e. the electrochemical potential ($\boldsymbol{\mu_i}$) of each ion is constant everywhere).

$$\nabla \boldsymbol{\mu_i} + \boldsymbol{z_i e} \nabla \boldsymbol{\psi} = 0 \qquad \textbf{(B2)}$$

$$\boldsymbol{\mu_i} = \boldsymbol{\mu_i^o} + \boldsymbol{k_b T} \ln(\boldsymbol{n_i}) \qquad \textbf{(B2a)}$$

where $\boldsymbol{\mu_i^o}$ is the bulk chemical potential of the i-th ion, $\boldsymbol{k_b}$ is **Boltzmann constant**, $\boldsymbol{T}$ is absolute temperature

Substituting **Eq. B2a** into **Eq. B2** and integrating in the bulk solution where $\boldsymbol{\psi} = 0$ and $\boldsymbol{n_i} = \boldsymbol{n_i^o}$ derives the Boltzmann equation (**Eq. B3**), which defines the local concentration of each ion in the diffuse EDL:

$$\boldsymbol{n_i} = \boldsymbol{n_i^o} e^{\left(\frac{-z_i e \psi}{k_b T}\right)} = \boldsymbol{N_A C_i^o} e^{\left(\frac{-z_i e \psi}{k_b T}\right)} \qquad \textbf{(B3)}$$

where $\boldsymbol{n_i^o}$ is the bulk number of i-th ions per unit volume, $\boldsymbol{N_A}$ is **Avogadro's number**, and $\boldsymbol{C_i^o}$ is the bulk concentration of the i-th ion

Using **Eq. B3**, the volume charge density of **Eq. B1a** can be defined converting Poisson's equation into the Poisson-Boltzmann equation:

$$\nabla^2 \boldsymbol{\psi} = -\frac{\mathbf{1}}{\boldsymbol{\varepsilon_o \varepsilon_r}} \sum_{i=1}^{N} \boldsymbol{z_i e\, N_A C_i^o} e^{\left(\frac{-z_i e \psi}{k_b T}\right)} \qquad \textbf{(B4)}$$

In order to solve the PBE, the Debye-Huckel approximation is applied. If the potential is sufficiently small in the EDL so that $\boldsymbol{z_i e\psi \ll k_b T}$, the exponential term in **Eq. B4** can be expanded based on the Taylor series ($e^{-x} = 1 - x - \cdots$) to yield a linearized form:

$$\nabla^2\boldsymbol{\psi} = -\frac{\mathbf{1}}{\boldsymbol{\varepsilon_o \varepsilon_r}}\left\{\sum_{i=1}^{N} \boldsymbol{z_i e\, N_A C_i^o} - \sum_{i=1}^{N} \frac{\boldsymbol{z_i^2 e^2 N_A C_i^o \psi}}{\boldsymbol{k_b T}}\right\} \qquad \textbf{(B5)}$$

Since $\sum_{i=1}^{N} \boldsymbol{z_i e\, N_A C_i^o} = 0$ due to the condition of electroneutrality in the bulk electrolyte, the linearized Poisson-Boltzmann equation reduces to:

$$\nabla^2\boldsymbol{\psi} = \frac{\boldsymbol{e^2 N_A \sum_{i=1}^{N} z_i^2 C_i^o}}{\boldsymbol{\varepsilon_o \varepsilon_r k_b T}}\boldsymbol{\psi} = \boldsymbol{\kappa^2\psi} \qquad \textbf{(B6)}$$

This approximation is adequate for particles with low surface potential in a solution of dilute indifferent ions (ions that do not hold a special interaction with the molecular surface [1, p. 19]) [1, p. 22]. Extracting the variable ($\boldsymbol{\kappa^2}$), and re-arranging for the Debye length (κ^{-1}) [m] yields the following:

$$\boldsymbol{\kappa^{-1}} = \sqrt{\frac{\boldsymbol{\varepsilon_o \varepsilon_r k_b T}}{\boldsymbol{N_A e^2 \sum_{i=1}^{N} 1000 C_i z_i^2}}} \qquad \textbf{(B7)}$$

where N_A is **Avogadro's constant**, R is the **universal gas constant**, k_b is the **Boltzmann constant**, e is the **electron charge**, μ is the permeability of free space, c is the **speed of light**, ε_o is **vacuum permittivity**, T is absolute temperature, ε_r is the solvent relative dielectric constant, N is the number of ionic species in solution, C_i is the bulk molar ion concentration of the i-th ionic species, z_i is the valence of the i-th ionic species

APPENDIX C. Computation of Solution Properties.

This appendix provides relationships for calculating the solution properties (specifically, relative dielectric, viscosity, and density) for the zeta potential prediction protocol (ZPRED). In general, all properties are defined by first calculating the respective property of pure water as a specified temperature and then correcting the value to consider the presence of salts. Many of the relationships rely on the electrolyte molality instead of the molarity, and so the conversion from molarity to molality is presented below for reference. The molality (m) of any electrolyte can be calculated from its mass fraction (x_m) in the equation below. Values between brackets represent the molar concentration of the substance (e.g. [KCl] represents the molar concentration of KCl).

$$x_m = \frac{MW_{Ion}[Ion]}{\rho_{H2O}\left({}^{1000g}/_{1kg}\right)+MW_{Ion}[Ion]}$$

$$m = \frac{x_m\left({}^{1000g}/_{1kg}\right)}{(1-x_m)MW_{Ion}}$$

where MW_{Ion} is the ion molecular weight

Relative Dielectric of Solution. The relative dielectric of an aqueous solution decreases in the presence of salts due the effect of kinetic polarization and the formation of water solvation shells around the ions [145]. In general, ZPRED assumes the value of the relative dielectric of pure water to adequately represent the solution relative dielectric. Theoretically under low ion concentrations, this is not a bad assumption; however, with increasing ion concentration the relative dielectric can deviate significantly from that of water [130]. The relative dielectric of pure water a function of temperature is defined below in **Eq. C1**.

- **Pure water relative dielectric** [146]

$$\varepsilon_{r,H2O} = A + BT + CT^2 + DT^3 = 78.285\ (at\ 25°C) \quad \textbf{(C1)}$$

where $A = 87.740$, $B = -0.4008$, $C = 0.0009398$, $D = -0.00000141$, and T is the temperature in Centigrade

The relative dielectric of a KCl solution as a function of the KCl concentration is shown below in **Eq. C2**. As can be seen, increasing KCl concentration decreases the overall dielectric of the solution.

- **KCl Solution Relative Dielectric** [130]

$$\varepsilon_{r,KCl} = \varepsilon_{r,H2O}(1 - 0.15[KCl]) \quad \textbf{(C2)}$$

As shown in **Fig. C1**, the inclusion of the corrected relative dielectric of solution (**Eq. C2**) holds very little impact on the modeled output and does not decrease deviation from experiment. Thus, inclusion of a correction for the depression of the relative dielectric with the increasing presence of salt was not included in this body of work.

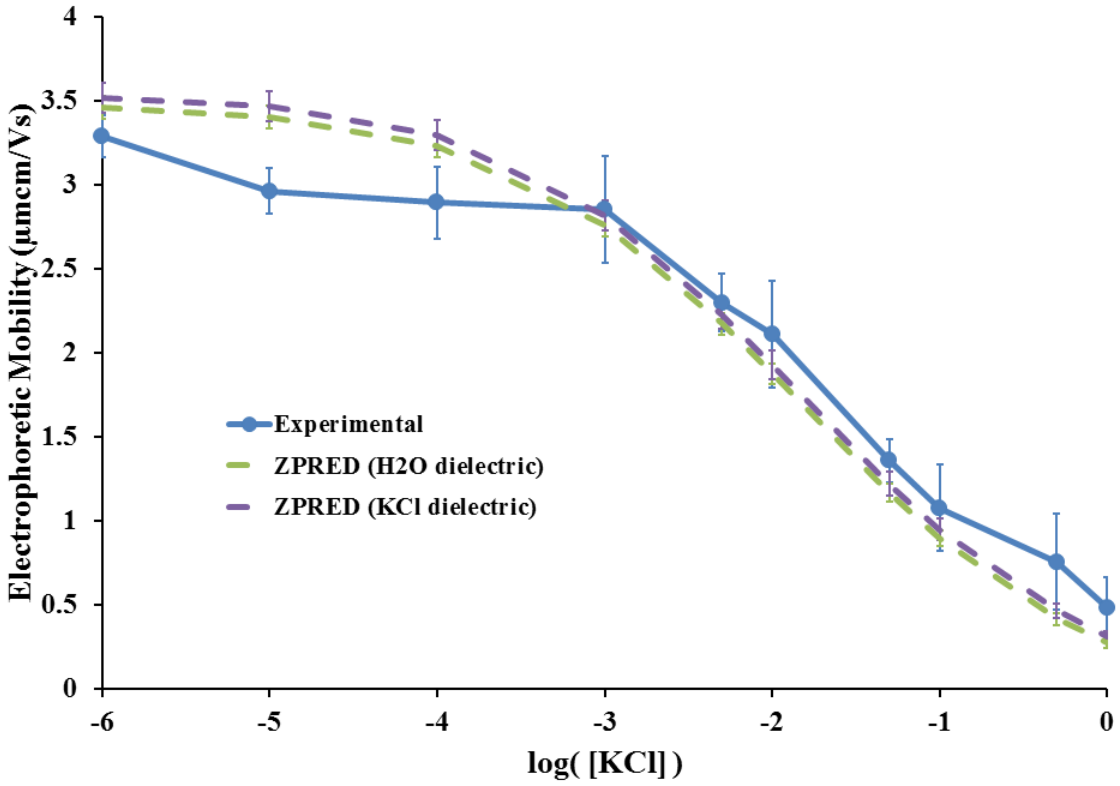

Figure C-1. Comparison of ZPRED Computation Using the Relative Dielectric of Pure Water (Eq. C1) Versus the Relative Dielectric of a KCl Solution (Eq. C2)

Viscosity of Solution. The viscosity of an aqueous solution can either increase or decrease in the presences of salts [59]. ZPRED uses a generalized viscosity model for aqueous electrolyte solutions developed by Laliberte [56]. The model applies the following mixing rule to define the viscosity (η) of an aqueous electrolyte solution.

$$\eta = (\eta_{H2O})^{x_{H2O}} \prod (\eta_i)^{x_i}$$

where η_{H2O} is the viscosity of pure water, x_{H2O} is the mass fraction of water, η_i is the i-th solute viscosity, and x_i is the i-th solute mass fraction

The viscosity of pure water is given below as a function of temperature and plotted below in **Fig. C2**.

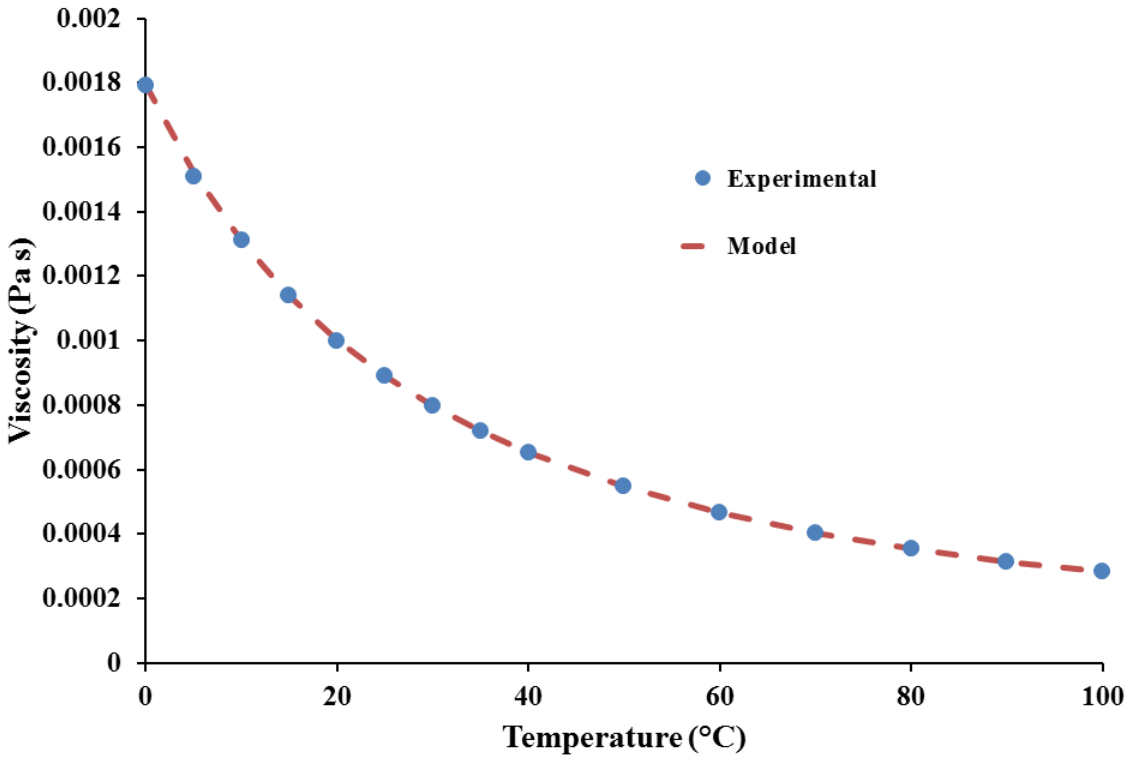

Figure C-2. Validation of Laliberte's Water Viscosity Model (Experimental values from [147])

- **Pure Water Viscosity** [Pa s] [56]

$$\eta_{H2O} = \left(\frac{T+246}{(0.05594T+5.2842)T+137.37}\right)\left({}^{1\ Pa\ s}/_{1000mPa\ s}\right) = 0.000890166 \quad \textbf{(C3)}$$

where T is the temperature in Centigrade

Three different aqueous electrolytes (KCl, KH_2PO_4 and KNO_3) are plotted showing the accuracy of the empirical relations compared to experimentally measured viscosities containing the protein, hen egg white lysozyme. All relations output viscosity in units of Pa s. Viscosities were measured with a Malvern Kinexus ultra+ rotational rheometer (Malvern Instruments, Worcestershire, U.K.). During measurement, 25 µL of solution was held between a 20 mm diameter flat plate upper geometry and 65-mm-diameter flat

plate lower geometry spaced with a 0.1 mm gap size. A shear rate of 11980 1/s was used as it was found to provide reasonably close values of pure water at 25°C. After allowing 5 minutes between the plates to reach thermal equilibrium, viscosities were measured for 60 seconds. Only the last 55 seconds were averaged to allow a 5 second delay to reach the machine's steady state.

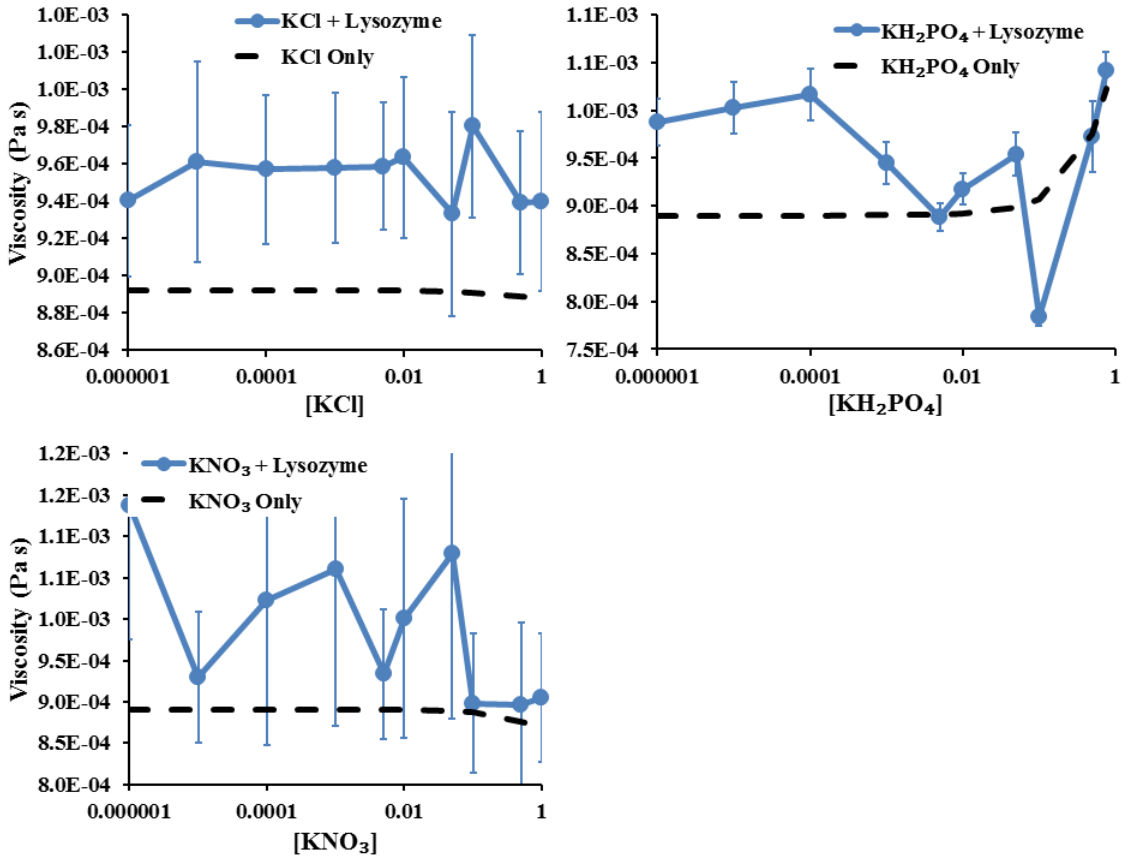

Figure C-3. Measured Viscosities Versus Empirical Pure Solvent Relationships

- **KCl Solution Viscosity** at 25°C [57]

$$\eta_{KCl} = \eta_{H2O} + Am + Bm^2 + Cm^3 \qquad \textbf{(C4)}$$

where $A = -0.0000101$, $B = 0.00000651$, $C = -0.000000249$, and $\boldsymbol{m}$ is molality of KCl in [mol/kg]

- **KH_2PO_4 Solution Viscosity** [56]

$$\eta_{KH2PO4} = (\eta_{H2O})^{(1-x_{KH_2PO_4})}\left(exp\left(\frac{A(x_{KH_2PO_4})^B+C}{(DT+1)(E(x_{KH_2PO_4})^F+1)}\right)\right)^{x_{KH2PO4}} \quad \textbf{(C5)}$$

where $A = 1358.1$, $B = 3.8539$, $C = 1.6617$, $D = 0.012129$, $E = -1.0516$, $F = 4.1301$, x_{KH2PO4} is the mass fraction of KH_2PO_4 and T is the temperature in Centigrade

- **KNO_3 Solution Viscosity** at 25°C [58]

$$\eta_{KNO3} = \eta_{H2O}\left(1 + A\sqrt{m} + Bm + Cm^2 + Dm^3\right) \quad \textbf{(C6)}$$

where $A = 0.005$, $B = -0.053$, $C = 0.0277$, $D = -0.0023$, and m is molality of KNO_3 in [mol/kg]

- **$KClO_3$ Solution Viscosity** at 25°C [59]

$$\eta_{KClO_3} = \eta_{H2O}\left(1 + 0.005\sqrt{[KClO_3]} - 0.0309[KClO_3]\right) \quad \textbf{(C7)}$$

- **NaCl Solution Viscosity** [56]

$$\eta_{NaCl} = (\eta_{H2O})^{(1-x_{NaCl})}\left(exp\left(\frac{A(x_{NaCl})^B+C}{(DT+1)(E(x_{NaCl})^F+1)}\right)\right)^{x_{NaCl}} \quad \textbf{(C8)}$$

where $A = 16.222$, $B = 1.3229$, $C = 1.4849$, $D = 0.0074691$, $E = 30.78$, $F = 2.0583$, x_{NaCl} is the mass fraction of NaCl, and T is temperature in Centigrade

- **HCl and KCl Solution Viscosity** [56]

$$\eta_{HCl/NaCl} = (\eta_{H2O})^{(x_{H2O})}\left(exp\left(\frac{A(1-x_{H2O})^B+C}{(DT+1)(E(1-x_{H2O})^F+1)}\right)\right)^{x_{HCl}}\left(exp\left(\frac{G(1-x_{H2O})^H+J}{(KT+1)(L(1-x_{H2O})^M+1)}\right)\right)^{x_{KCl}} \quad \textbf{(C9)}$$

where x_{H2O} is the mass fraction of water, $A = 7.124$, $B = 1.1919$, $C = 1.6648$, $D = 0.00096271$, $E = 22.185$, $F = 1.479$, x_{HCl} is the mass fraction of HCl, G = 6.4883, H = 1.3175, J = −0.77785, K = 0.092722, L = −1.300, M = 2.0811, x_{KCl} is the mass fraction of KCl, and T is the temperature in Centigrade

- **HCl and NaCl Solution Viscosity** [56]

$$\eta_{HCl/NaCl} = (\eta_{H2O})^{(x_{H2O})}\left(exp\left(\frac{A(1-x_{H2O})^{B}+C}{(DT+1)(E(1-x_{H2O})^{F}+1)}\right)\right)^{x_{HCl}}\left(exp\left(\frac{G(1-x_{H2O})^{H}+J}{(KT+1)(L(1-x_{H2O})^{M}+1)}\right)\right)^{x_{NaCl}} \quad \textbf{(C10)}$$

where x_{H2O} is the mass fraction of water, $A = 7.124$, $B = 1.1919$, $C = 1.6648$, $D = 0.00096271$, $E = 22.185$, $F = 1.479$, x_{HCl} is the mass fraction of HCl, G = 16.222, H = 1.3229, J = 1.4849, K = 0.0074691, L = 30.78, M = 2.0583, x_{NaCl} is the mass fraction of NaCl, and T is the temperature in Centigrade

- **NaOH and NaCl Solution Viscosity** [56]

$$\eta_{NaOH/NaCl} = (\eta_{H2O})^{(x_{H2O})}\left(exp\left(\frac{A(1-x_{H2O})^{B}+C}{(DT+1)(E(1-x_{H2O})^{F}+1)}\right)\right)^{x_{NaOH}}\left(exp\left(\frac{G(1-x_{H2O})^{H}+J}{(KT+1)(L(1-x_{H2O})^{M}+1)}\right)\right)^{x_{NaCl}} \quad \textbf{(C11)}$$

where x_{H2O} is the mass fraction of water, $A = 440.2$, $B = 0.0089764$, $C = -423.67$, $D = 0.015949$, $E = 107.6$, $F = 4.6489$, x_{NaOH} is the mass fraction of NaOH, G = 16.222, H = 1.3229, J = 1.4849, K = 0.0074691, L = 30.78, M = 2.0583, x_{NaCl} is the mass fraction of NaCl, and T is the temperature in Centigrade

- **Citrate Phosphate Solution Viscosity** [148]

$$\eta_{Cit/Phs} = (\eta_{H2O})^{x_{H2O}}(\eta_{Cit})^{x_{Cit}}(\eta_{Phs})^{x_{Phs}} \quad \textbf{(C12)}$$

where x_{H2O} is the mass fraction of water, η_{Cit} is the viscosity of pure citric acid, x_{Cit} is the mass fraction of citric acid, η_{Phs} is the viscosity of pure Na_2HPO_4, and x_{Phs} is the mass fraction of Na_2HPO_4

Density of Solution. The density of an aqueous solution typically increases in the presence of salts. ZPRED uses a generalized density model for aqueous electrolyte solutions developed by Laliberte et Cooper [86]. The model applies the following mixing rule to define the density (ρ) of aqueous electrolyte solutions.

$$\rho = \frac{1}{\frac{x_{H2O}}{\rho_{H2O}}+\sum\frac{x_i}{\rho_i}}$$

where ρ_{H2O} is the density of water, x_{H2O} is the mass fraction of water, ρ_i is the density of the i-th solute, and x_i is the i-th solute mass fraction

Two empirical relationships for the density of pure water as function of temperature are given below. Although both are accurate, ZPRED uses the second relationship (**Eq. C13b**) to define the pure water viscosity out of respect for Laliberte. This relationship along with experimental values are shown below in **Fig. C4.**

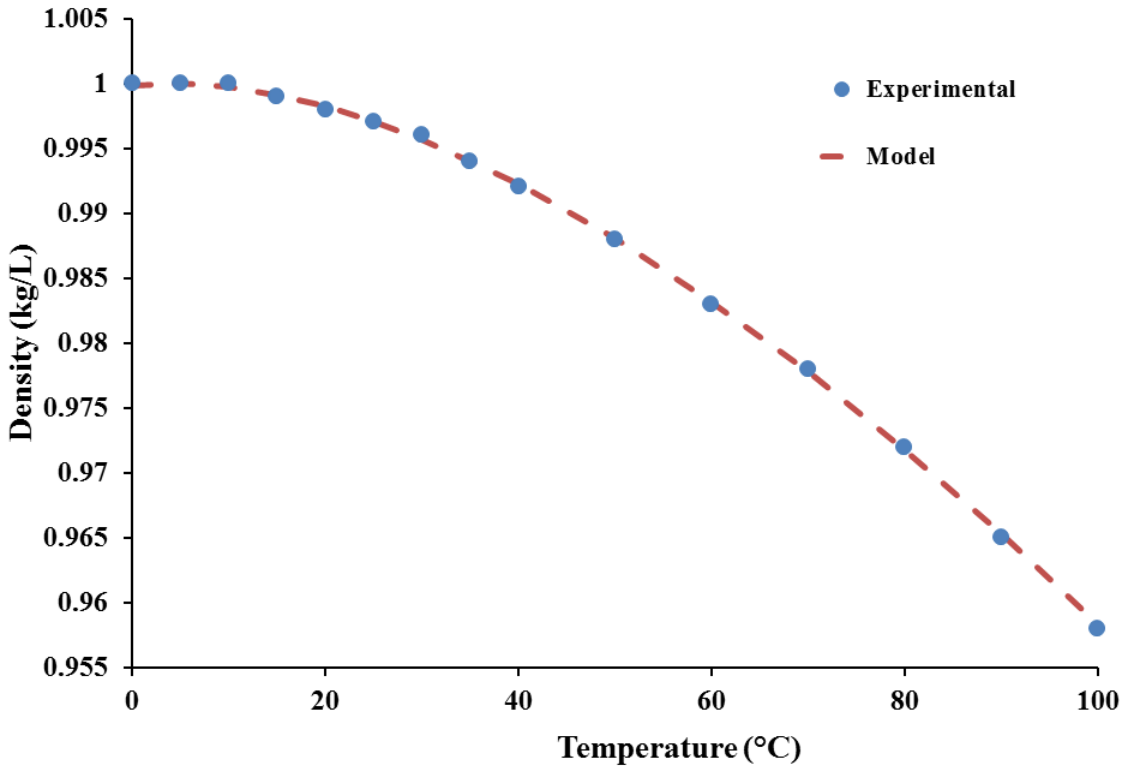

Figure C-4. Validation of Laliberte's Density Model (Experimental from [147])

- Pure Water Density [kg/L] [149]

$$\rho_{H2O} = 0.9999727 + AT + BT^2 + CT^3 + DT^4 = 0.997066\ (at\ 25°C) \quad \textbf{(C13a)}$$

where $A = 4.035198 \times 10^{-5}$, $B = -7.090436 \times 10^{-6}$, $C = 3.554779 \times 10^{-8}$, $D = -1.0027098 \times 10^{-10}$, and T is the temperature in Centigrade

- **Pure Water Density** [kg/L] [86]

$$\rho_{H2O} = \frac{\left(\left(\left(\left((AT+B)T+C\right)T+D\right)T+E\right)T+F\right)}{1+GT}\left({}^{1\ kg}/_{1000\ g}\right) = 0.997045(25°C) \text{ (C13b)}$$

where $A = -2.8054253 \times 10^{-10}$, $B = 1.0556302 \times 10^{-7}$, $C = -4.6170461 \times 10^{-5}$, $D = -0.0079870401$, $E = 16.945176$, $F = 999.83952$, $G = 0.01687985$ and T is the temperature in Centigrade

- **KCl Solution Density** at 25°C [42]

$$\rho_{KCl} = \rho_{H2O} + 0.0470[KCl] \quad \textbf{(C14)}$$

In addition to this empirical relationship, interpolation on experimental densities can also be used [57].

- **KH_2PO_4 Solution Density** at 20°C [148, pp. 8-68]

$$\rho_{KH2PO4} = \rho_{H2O} + 0.095182[KH2PO4] \quad \textbf{(C15)}$$

Note: equation was fit to experimental data (density vs ion concentration)

- **KNO_3 Solution Density** [149]

$$\rho_{KNO3} = \frac{\rho_{H2O}}{1-\rho_{H2O}\left(Ax_m+Bx_m^2\right)} \quad \textbf{(C16)}$$

where $A = 0.61564$, $B = -0.1390$, and x_m is the mass fraction of KNO_3

- **$KClO_4$ Solution Density** at 25°C

$$\rho_{KClO4} = \frac{1}{\frac{x_{H2O}}{\rho_{H2O}}+\frac{x_{KClO4}}{2523.9}} \quad \textbf{(C17)}$$

where x_{KClO4} is the mass fraction of $KClO_4$

- **NaCl Solution Density** [86]

$$\rho_{NaCl} = \frac{1}{\frac{x_{H2O}}{\rho_{H2O}}+x_{NaCl}\left(\frac{x_{NaCl}+C+DT}{\left(Ax_{NaCl}+B\right)exp\left(0.000001(T+E)^2\right)}\right)} \quad \textbf{(C18)}$$

where x_{H2O} is the mass fraction of water, x_{NaCl} is the mass fraction of NaCl, $A = -0.00433$, $B = 0.06471$, $C = 1.01660$, $D = 0.014624$, $E = 3315.6$, and T is the temperature in Centigrade

- **HCl and KCl Solution Density** [86]

$$\rho_{HCl/KCl} = \frac{1}{\frac{x_{H2O}}{\rho_{H2O}} + x_{HCl}\left(\frac{x_{HCl}+C+DT}{(Ax_{HCl}+B)exp(0.000001(T+E)^2)}\right) + x_{KCl}\left(\frac{x_{KCl}+H+JT}{(Fx_{KCl}+G)exp(0.000001(T+K)^2)}\right)} \quad \textbf{(C19)}$$

where x_{H2O} is the mass fraction of water, x_{HCl} is the mass fraction of HCl, x_{KCl} is the mass fraction of KCl, $A = -80.061$, $B = 255.42$, $C = 118.42$, $D = 1.0164$, $E = 2619.5$, $F = -0.46782$, $G = 4.30800$, $H = 2.3780$, $J = 0.022044$, $K = 2714.0$, and T is the temperature in Centigrade

- **HCl and NaCl Solution Density** [86]

$$\rho_{HCl/NaCl} = \frac{1}{\frac{x_{H2O}}{\rho_{H2O}} + x_{HCl}\left(\frac{x_{HCl}+C+DT}{(Ax_{HCl}+B)exp(0.000001(T+E)^2)}\right) + x_{NaCl}\left(\frac{x_{NaCl}+H+JT}{(Fx_{NaCl}+G)exp(0.000001(T+K)^2)}\right)} \quad \textbf{(C20)}$$

where x_{H2O} is the mass fraction of water, x_{HCl} is the mass fraction of HCl, x_{NaCl} is the mass fraction of NaCl, $A = -80.061$, $B = 255.42$, $C = 118.42$, $D = 1.0164$, $E = 2619.5$, $F = -0.00433$, $G = 0.06471$, $H = 1.01660$, $J = 0.014624$, $K = 3315.6$, and T is the temperature in Centigrade

- **NaOH and NaCl Solution Density** [86]

$$\rho_{NaOH/NaCl} = \frac{1}{\frac{x_{H2O}}{\rho_{H2O}} + x_{NaOH}\left(\frac{x_{NaOH}+C+DT}{(Ax_{NaOH}+B)exp(0.000001(T+E)^2)}\right) + x_{NaCl}\left(\frac{x_{NaCl}+H+JT}{(Fx_{NaCl}+G)exp(0.000001(T+K)^2)}\right)} \quad \textbf{(C21)}$$

where x_{H2O} is the mass fraction of water, x_{NaOH} is the mass fraction of NaOH, x_{NaCl} is the mass fraction of NaCl, $A = 385.55$, $B = 753.47$, $C = -0.10938$, $D = 0.0006953$, $E = 542.88$, $F = -0.00433$, $G = 0.06471$, $H = 1.01660$, $J = 0.014624$, $K = 3315.6$, and T is the temperature in Centigrade

- **Citrate Phosphate Solution Density** [148]

$$\rho_{Cit/Phs} = \frac{1}{\frac{x_{H2O}}{\rho_{H2O}} + \frac{x_{Cit}}{\rho_{Cit}} + \frac{x_{Phs}}{\rho_{Phs}}} \quad \textbf{(C22)}$$

where x_{H2O} is the mass fraction of water, x_{Cit} is the mass fraction of citric acid, ρ_{Cit} is the density of pure citric acid, x_{Phs} is the mass fraction of Na_2HPO_4, and ρ_{Phs} is the density of pure Na_2HPO_4

Ion Radii in Solution. A good review covering concepts of ionic radii and measurement was given by Marcus [68]. All ionic radii values came from this reference except for those indicated by a superscript letter.

Ion Name	Ion Symbol	Ionic Radius (Å)		Ion Name	Ion Symbol	Ionic Radius (Å)
Aluminum	Al^{+3}	0.460		Acetate	CH_3COO^-	2.180
Ammonium	NH_4^+	1.180		Borate[C]	$B(OH)_4^-$	2.440
Barium	Ba^{+2}	1.480		Bromide	Br^-	1.950
Beryllium	Be^{+2}	0.330		Carbonate[A]	CO_3^{-2}	1.780
Cadmium	Cd^{+2}	0.880		Carboxylate	COO^-	1.280
Calcium	Ca^{+2}	1.000		Chloride	Cl^-	1.760
Cesium	Cs^+	1.710		Fluoride	F^-	1.450
Chromium (III)	Cr^{+3}	0.540		Hydroxide[A]	OH^-	1.330
Cobalt	Co^{+2}	0.680		Methanoate[B]	HCO_2^-	1.310
Copper (II)	Cu^{+2}	0.980		Nitrate	NO_3^-	2.030
Hydrogen[A]	H^+	0.250		Perchlorate	ClO_4^-	2.280
Hydronium	H_3O^+	1.330		Phosphate[B]	$H_2PO_4^-$	2.380
Indium	In^{+3}	0.730		Phosphate[B]	HPO_4^{-2}	2.300
Iron (II)	Fe^{+2}	0.690		Phosphate[B]	PO_4^{-3}	2.230
Iron (III)	Fe^{+3}	0.610		Selenate	SeO_4^-	2.530
Lithium	Li^+	0.660		Sulfate	SO_4^-	2.390
Magnesium	Mg^{+2}	0.670		Water	H_2O	1.420
Manganese	Mn^{+2}	0.770				
Mercury (II)	Hg^{+2}	1.000				
Methylammonium	$CH_3NH_3^+$	2.280				
Nickel	Ni^{+2}	0.640				
Potassium	K^+	1.370				
Rhodium (III)	Rh^{+3}	0.610				

Rubidium	Rb^{+}	1.500				
Silver	Ag^{+}	0.990				
Sodium	Na^{+}	0.930				
Strontium	Sr^{+2}	1.220				
Thallium (III)	Tl^{+3}	0.810				
Thorium	Th^{+4}	1.110				
Tin (II)	Sn^{+2}	1.200				
Yttrium	Y^{+3}	0.925				
Zinc	Zn^{+2}	0.670				

Table C-1. Ionic Radii Used by ZPRED. [A] [150], [B] [37], [C] [151]

APPENDIX D. ZPRED Source Code

This appendix provides source code for the zeta potential prediction protocol (ZPRED) allowing for its re-creation if so desired.

SECTION D1. Molecular Dynamics Scripts

This section contains example codes for running the molecular dynamics simulation component of the zeta potential prediction protocol. Indented code is separated from the main text by lines.

Amber 2015 Molecular Dynamics Software Installation. Due to the complicated nature of the installation of the Amber 2015 molecular dynamics software package, a thorough installation procedure for the 64-bit Ubuntu 14.04 operating system is provided here. The first step requires installation of various libraries Amber depends on.

```
sudo apt-get install freeglut3-dev build-essential libx11-dev libxmu-dev libxi-dev libgl1-mesa-glx libglu1-mesa libglu1-mesa-dev flex tcsh gfortran g++ xorg-dev libbz2-dev git vim libopenmpi-dev openmpi-bin python-tk python-dev autoconf libtool python-mpi4py grace gnuplot ssh bkchem chemtool netpbm libnetcdf-dev python-numpy python-scipy mpich2 libmpich2-dev gksu
```

The next step requires updating the blacklist default graphics modules, which can be opened with the command shown below.

```
gksu gedit /etc/modprobe.d/blacklist.conf
```

Once opened, add the following lines to the end of the file and save it.

```
blacklist vga16fb
blacklist nouveau
blacklist rivafb
blacklist nvidiafb
blacklist rivatv
```

Now, update the .bashrc file, which can be opened with the command shown below.

```
gedit ~/.bashrc
```

Once opened, add the following lines to the end of the file and save it.

```
export AMBERHOME=/*Insert file path where AMBER and AMBERTOOLS are extracted to*/
export CUDA_HOME=/usr/local/cuda-6.5
export LD_LIBRARY_PATH=$CUDA_HOME/lib64:$LD_LIBRARY_PATH\:$AMBERHOME/lib
export LD_LIBRARY_PATH=/usr/local/cuda-6.5/lib:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH=/usr/local/cuda-6.5/lib64:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH=/usr/lib/nvidia-current:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH=/usr/local/lib:$LD_LIBRARY_PATH
export CUDA_ROOT=/usr/local/cuda-6.5/bin
export PATH=$PATH:$AMBERHOME:$AMBERHOME/bin:$AMBERHOME/lib:$MPI_HOME:$CUDA_HOME:$CUDA_HOME/bin:$CUDA_HOME/lib64:$CUDA_HOME/lib:$LD_LIBRARY_PATH:$CUDA_ROOT:$CUDA_HOME/include:$PROTCADDIR:$PROTCADDIR/bin
```

The next step requires installation of an appropriate NVIDIA driver and CUDA toolkit. The newest NVIDIA graphics driver for your GPU (e.g. GeForce GTX 480) can be downloaded from the following website:

http://www.nvidia.com/Download/index.aspx?lang=en-us.

However, it is recommended to use the graphics driver included in this work (NVIDIA-Linux-x86_64-340.98.run) as it is known to be compatible with the included CUDA toolkit (cuda_6.5.14_linux_64.run). Though other versions of the CUDA toolkit run file can be obtained at: https://developer.nvidia.com/cuda-downloads. If another version is used, you must update the .bashrc file (currently it is updated for the cuda 6.5). Once both

run files for the NVIDIA graphics driver and CUDA toolkit are obtained, remove any old NVIDIA programs and drivers with the command below.

```
sudo apt-get remove --purge nvidia*
```

Now reboot the computer, and enter Ctrl+Alt+F1 to login to terminal only mode once you are at the login screen. Go to the directory, where your run files are located and run the following commands for the NVIDIA driver installation.

```
sudo service lightdm stop
chmod +x NVIDIA-Linux*.run
sudo ./NVIDIA-Linux*.run
sudo reboot
```

Once complete, login normal, open a terminal (Ctrl+Alt+T) and cd to the directory where your cuda run file is located then run the following commands. Be sure to install the toolkit only, because the driver and samples are not necessary.

```
chmod +x cuda_6.5*.run
sudo ./cuda_6.5*.run
```

Now we are ready to install Amber and Ambertools. First, download both Amber14 and Ambertools15 into the directory where you want your Amber installation to occur. Note this is the file path you set to define AMBERHOME in the .bashrc file. Open a terminal and run the following commands.

```
tar xvfj Amber14*.tar.bz2
tar xvfj AmberTools15.tar.bz2
```

Your Amber files will be extracted and ready for installation. Now, patch Amber with the following commands.

```
cd $AMBERHOME
./update_amber –update
```

Run the commands above repeatedly until it reports that Amber and AmberTools are up to date. Once finished, we will install sander first using the following commands in the $AMBERHOME directory.

```
./configure gnu
make -j6 install
```

Second, we will install pmemd. Though this first requires updating a configuration file as shown below.

```
sudo vim /usr/local/cuda/include/host_config.h
```

In line 80 of this file, change the 8 to a 9:

#if _GNUC_ > 4 || (_GNUC_ == 4 && _GNUC_MINOR_ > 8)

change to

#if _GNUC_ > 4 || (_GNUC_ == 4 && _GNUC_MINOR_ > 9)

Once updated, save and exit the file then run the following commands in the $AMBERHOME directory.

```
./configure -cuda gnu
make -j6 install
```

This completes the installation of the Amber 2015 Molecular Dynamics software package.

LEaP Input Command File. A command file for *tleap* corresponding to **Step 1a** of our protocol is shown below. Prior to running the commands with *tleap*, the Amber tool, *pdb4amber*, is used to prepare the crystal structure (e.g. in shell script, pdb4amber -i $pdbFile -o $newPdbFile --dry --reduce).

```
source leaprc.ff14SB
protein=loadpdb $newPdbFile
solvateBox protein TP3 20
check protein
saveamberparm protein $leapTopFile $leapCrdFile
quit
```

To run the commands, save the script as a file ($leapInputFile) and call *tleap* as follows:

```
tleap -f $leapInputFile
```

Amber Molecular Dynamics: Energy Minimization Control File. A control file for Amber molecular dynamics corresponding to **Step 1b** of our protocol is shown below. Text following exclamation points are comments.

```
Energy Minimization
&cntrl
    imin=1,      ! 0 = No minimization, 1 = Perform minimization
    maxcyc=3000, ! Maximum number of cycles
    ntmin=1,     ! 0,1,2,3,or 4 specifies minimization method
    ncyc=1500,   ! Switch from steepest descent to conjugate gradient after this many
cycles when ntmin=1
    ntpr=200,    ! prints energy information every ntpr steps to mdout and mdinfo
    iwrap=1,     ! wrap coordinates into primary box
```

```
&end
```

After running tleap (see LEaP Input Command File), a topology file ($leapTopFile) and an initial coordinates file ($leapCrdFile) are generated that are necessary inputs for performing this step. To run the control commands, save the script as a file ($emCtrlFile) and call pmemd as follows. Note it is necessary to specify which GPU is to be used.

```
export CUDA_VISIBLE_DEVICES=1
pmemd.cuda -O -i $emCtrlFile -o $emOutputFile -p $leapTopFile -c $leapCrdFile -r
$emRestartFile
```

Amber Molecular Dynamics: Thermal Excitation Control File. A control file for Amber molecular dynamics corresponding to **Step 1c** of the protocol is shown below. This example shows how to heat up a crystal structure and its explicit water environment from 0 K to 298 K over a time span of 100 picoseconds. Text following exclamation points are comments.

```
Thermal Excitation (100 ps)
&cntrl
    imin=0,      ! 0 = No minimization, 1 = Perform minimization
    irest=0,     ! 0=No restart, run new Sim,1=restart Simulation
    ntx=1,       ! 1 or 5 read initial coordinates
    ntr=0,       ! 0=no atoms restrained,>0 restrains atoms by restraint_wt
    nmropt=0,    ! 0=No NMR analysis,1 or 2
    iwrap=1,     ! wrap coordinates into primary box
    ntb=1,       ! 0=no periodicity,1=constant vol,2=constant pressure
    ntp=0,       ! 0=no pressure scaling,1=isotropic,2=anisotropic,3=semi-isotropic
    nstlim=50000,! Num of MD steps to be performed
    dt=.002,     ! Time step (psec)
    ntc=2,       ! 1=No SHAKE,2=H-bonds constrained,3=all bonds constrained
    ntf=2,       ! 1,2,3,4,5,6,7,or 8 Force evaluation
    ntt=3,       ! 0,1,2,3,or 9 temp scaling switch
    ig=-1,       ! random number generator seed (-1 based on time)
    tempi=0.0,   ! Initial temp
    temp0=298.0, ! Reference temp system is to be kept
    gamma_ln=3.0,! Collision frequency (1/psec)
```

```
    ntpr=10000,  ! prints energy information every ntpr steps to mdout and mdinfo
    ntwx=10000,  ! writes coordinates to mdcrd (-x output) every ntwx steps
    ntwr=50000,  ! writes restart file every ntwr steps
    cut=8.0,     ! non-bonded cutoff in Angstroms
&end
```

The restart coordinates file ($emRestartFile) from **Step 1b** is a necessary input for this step. To run the control commands, save the script as a file ($thrmCtrlFile) and call *pmemd* as follows. Note it is necessary to specify which GPU is to be used.

```
export CUDA_VISIBLE_DEVICES=1
pmemd.cuda -O -i $thrmCtrlFile -o $thrmOutputFile -p $leapTopFile -c
$emRestartFile -r $thrmRestartFile -x $thrmCrdFile
```

Amber Molecular Dynamics: Simulation in Solution. A control file for Amber molecular dynamics corresponding to **Step 1d** of our protocol is shown below. Text following exclamation points are comments.

```
Main MD Simulation (100 ns)
&cntrl
    imin=0,      ! 0 = No minimization, 1 = Perform minimization
    irest=0,     ! 0=No restart, run new Sim,1=restart Simulation
    ntx=1,       ! 1 or 5 read initial coordinates
    ntr=0,       ! 0=no atoms restrained,>0 restrains atoms by restraint_wt
    nmropt=0,    ! 0=No NMR analysis,1 or 2
    iwrap=1,     ! wrap coordinates into primary box
    ntb=1,       ! 0=no periodicity,1=constant vol,2=constant pressure
    ntp=0,       ! 0=no pressure scaling,1=isotropic,2=anisotropic,3=semi-isotropic
    nstlim=50000000,! Num of MD steps to be performed
    dt=.002,     ! Time step (psec)
    ntc=2,       ! 1=No SHAKE,2=H-bonds constrained,3=all bonds constrained
    ntf=2,       ! 1,2,3,4,5,6,7,or 8 Force evaluation
    ntt=1,       ! 0,1,2,3,or 9 temp scaling switch
    temp0=298.0,  ! Reference temp system is to be kept
    tautp=10.0,   ! Pressure relaxation time (psec)
    ntpr=10000,   ! prints energy information every ntpr steps to mdout and mdinfo
    ntwx=10000,  ! writes coordinates to mdcrd every ntwx steps
    ntwr=10000,  ! writes restart file every ntwr steps
```

```
    cut=8.0,      ! non-bonded cutoff in Angstroms
&end
```

The restart coordinates file ($thrmRestartFile) from **Step 1c** is a necessary input for this step. To run the control commands, save the script as a file ($mdCtrlFile) and call *pmemd* as follows. Note it is necessary to specify which GPU is to be used.

```
export CUDA_VISIBLE_DEVICES=1
pmemd.cuda -O -i $mdCtrlFile -p $leapTopFile -c $thrmRestartFile -o $mdOutputFile -r $mdRestartFile -x $mdCrdFile
```

Amber Molecular Dynamics: Post Simulation Processing. A control file for the Amber tool, cpptraj, corresponding to **Step 1d** of ZPRED is shown below.

```
trajin $mdCrdFile
unwrap :1-$numResidues
center mass :1-$numResidues
image center familiar
rms first mass :1-$numResidues
```

The main molecular dynamics simulation coordinates file ($mdCrdFile) from **Step 1d** is a necessary input for this step. To run the commands, save the script as a file ($cpptrajCtrlFile) and call *cpptraj* as follows. The output is a centered coordinates file ($cpptrajCrdFile) that can be used to analyze the molecular dynamics simulation.

```
cpptraj -p $leapTopFile -i $cpptrajCtrlFile -x $cpptrajCrdFile
```

Amber Molecular Dynamics: Convert Coordinates to PDB Format. The commands necessary for using the Amber tool, *ambpdb,* to convert simulation coordinates ($mdRestartFile) into PDB format ($mdPdbFile) is shown below.

```
ambpdb -bres -p $leapTopFile < $mdRestartFile > $mdPdbFile
```

SECTION D2. Automation of MSMS

MSMS [85] is used to generate solvent-excluded surface coordinates and their normal vectors propagating from the surface of a PDB structure file. The shell script below is called each time to generate the different surfaces necessary for the protocol.

Run_MSMS.sh – Command file for executing MSMS

```
#!/bin/bash
pdbFile=$Fldr"PDB_Format/"$PDB".pdb"
msmsFldr=$Fldr"msms/"
pdb2xyzrn=$msmsFldr"pdb_to_xyzrn"
xFile=$msmsFldr"msms.x86_64Linux2.2.6.1"

sphereInFile=$msmsFldr$PDB".xyzrn"
triSurfOutFile=$msmsFldr$PDB"_surface"
areaFile=$msmsFldr$PDB"_area"
outputFile=$Fldr"Surface/"$PDB$fldrExt"fromMSMS/msmsOutput_SR"$5".txt"

# Select 5th Variable
probeRad=$5
surfPointDensity=1.000000
surfPointHiDensity=3.000000

# PDB_to_XYZRN
cd $msmsFldr
$pdb2xyzrn $pdbFile > $sphereInFile

# MSMS
$xFile\
 -if $sphereInFile\
 -of $triSurfOutFile\
 -af $areaFile\
 -probe_radius $probeRad\
 -density $surfPointDensity\
 -hdensity $surfPointHiDensity\
 -no_header > $outputFile
```

```
mv -f $msmsFldr$PDB"_surface.face"
$Fldr"Surface/"$PDB$fldrExt"fromMSMS/SR"$probeRad".face"
mv -f $msmsFldr$PDB"_surface.vert"
$Fldr"Surface/"$PDB$fldrExt"fromMSMS/SR"$probeRad".vert"
mv -f $msmsFldr$PDB"_area.area"
$Fldr"Surface/"$PDB$fldrExt"fromMSMS/SR"$probeRad".area"
```

SECTION D3. Automation of HYDROPRO

HYDROPRO [83] is used to calculate the single particle diffusivity of a PDB structure file as well as its Einstein coefficient. To automate HYDROPRO, C++ codes were developed for writing HYDROPRO input files and reading HYDROPRO output files. Automation of HYDROPRO is controlled by the shell script shown below.

Run_HYDROPRO.sh – Command file for executing HYDROPRO

```
#!/bin/bash

PDB=$1
Fldr=$2
fldrExt=$3

# Copy PDB File into HYDROPRO Folder
original_pdbFile=$Fldr"PDB_Format/"$PDB".pdb"
hydroproFldr=$Fldr"hydropro10/"
cd $hydroproFldr
pdbFile=$hydroproFldr$PDB".pdb"
cp $original_pdbFile $pdbFile

inputFile=$hydroproFldr"hydropro.dat"
xFile=$hydroproFldr"hydropro10-lnx.exe"

$xFile $inputFile $pdbFile

# Remove Copied PDB File from HYDROPRO Folder
rm -f $pdbFile

# Move Files to Hydrodynamic Properties Folder
mv -f $hydroproFldr$PDB"-pri.bea"
$Fldr"Hydrodynamic_Properties/"$PDB$fldrExt$PDB"-pri.bea"
```

```
mv -f $hydroproFldr$PDB"-pri.vrml" $Fldr"Hydrodynamic_Properties/"$PDB$fldrExt$PDB"-pri.vrml"
mv -f $hydroproFldr$PDB"-res.txt" $Fldr"Hydrodynamic_Properties/"$PDB$fldrExt$PDB"-res.txt"
mv -f $hydroproFldr$PDB"-sol.txt" $Fldr"Hydrodynamic_Properties/"$PDB$fldrExt$PDB"-sol.txt"
mv -f $hydroproFldr"hydropro-fit.txt" $Fldr"Hydrodynamic_Properties/"$PDB$fldrExt"hydropro-fit.txt"
mv -f $hydroproFldr"hydropro-sum.txt" $Fldr"Hydrodynamic_Properties/"$PDB$fldrExt"hydropro-sum.txt"

# Return to Main Directory
cd $Fldr
```

SECTION D4. Automation of APBS

APBS [88] is used to generate the electric potential distribution propagating outward from the surface of a PQR structure file into implicit solvent. To automate APBS, the APBS python script, inputgen.py, was heavily edited to allow for more generalized control of the APBS executable and is shown below (inputgen_edited.py). Upon generating an input file, APBS is executed by calling the executable with the input file (e.g. in shell script, $APBS_bin $apbsInputFile).

---

inputgen_edited.py – Generates an APBS input file when called from the shell script, IN_File_Editer.sh

```
""" inputgen class
   Create an APBS input file using psize data

   ---------------------------
   Version:  $Id$
   ---------------------------
"""

# User - Definable Variables: Default values
# cfac = 1.7              # Factor by which to expand mol dims to
                          # get coarse grid dims
# fadd = 20               # Amount to add to mol dims to get fine
                          # grid dims
```

```
# space = 0.50              # Desired fine mesh resolution
# gmemfac = 200             # Number of bytes per grid point required
                            # for sequential MG calculation
# gmemceil = 400            # Max MB allowed for sequential MG
                            # calculation.  Adjust this to force the
                            # script to perform faster calculations (which
                            # require more parallelism).
# ofrac = 0.1               # Overlap factor between mesh partitions
# redfac = 0.25             # The maximum factor by which a domain
                            # dimension can be reduced during focusing

import string, sys
import psize
import pickle

class Elec:
    """
        An object for the ELEC section of an APBS input file
    """
    def __init__(self, pqrpath, size, method, asyncflag, istrng=0, potdx=0):
        """
            Initialize the variables that can be set in this object
            Users can modify any of these variables (that's why
            they're here!)
        """

        # If this is an async or parallel calc, we want to use
        # the per-grid dime rather than the global dime.

        self.dime = size.getFineGridPoints()
        gmem = 200.0 * self.dime[0] * self.dime[1] * self.dime[2] / 1024.0 / 1024.0
        if method == "": # method not named - use ceiling
            if gmem > size.getConstant("gmemceil"): method = "mg-para"
            else: method = "mg-auto"

        if method == "mg-para":
            self.dime = size.getSmallest()

        self.method = method
        self.istrng = istrng
        self.glen = size.getCoarseGridDims()
        self.cglen = size.getCoarseGridDims()
        self.fglen = size.getFineGridDims()
        self.pdime = size.getProcGrid()

        self.label = ""
```

```
        self.nlev = 4
        self.ofrac = 0.1
        self.async = 0
        self.asyncflag = asyncflag
        self.cgcent = "mol 1"
        self.fgcent = "mol 1"
        self.gcent = "mol 1"
        self.mol = 1
        self.lpbe = 1
        self.npbe = 0
        self.bcfl = "sdh"
#         self.ion = [[-1,1.815],[1,1.875]] # Multiple ions possible
        self.ion = [[-1,0.150,1.815],[1,0.150,1.875]] # Multiple ions possible
        self.pdie = 2.0
        self.sdie = 78.54
        self.srfm = "smol"
        self.chgm = "spl2"
        self.sdens = 10.0
        self.srad = 1.4
        self.swin = 0.3
        self.temp = 298.15
        self.gamma = 0.105
        self.calcenergy = "total"
        self.calcforce = "no"

        if potdx == 1:
            self.write = [["pot", "dx", pqrpath]] # Multiple write statements possible
        else:
                                #if numWrite == 0:
                                #       self.write = ""
                                #elif numWrite >= 1:
                                        self.write = [["pot", "dx", "pot1"]]

    def __str__(self):
        """
            Return the elec statement as a string. Check the method
            to see which keywords to use.
        """
        text = "elec %s\n" % self.label
        text += "    %s\n" % self.method
        text += "    dime %i %i %i\n" % (self.dime[0], self.dime[1], self.dime[2])
        if self.method == "mg-manual":
            text += "    nlev %i\n" % self.nlev
            text += "    glen %.3f %.3f %.3f\n" % (self.glen[0], self.glen[1], self.glen[2])
            text += "    gcent %s\n" % self.gcent
        elif self.method == "mg-auto":
```

```
            text += "    cglen %.4f %.4f %.4f\n" % (self.cglen[0], self.cglen[1], self.cglen[2])
            text += "    fglen %.4f %.4f %.4f\n" % (self.fglen[0], self.fglen[1], self.fglen[2])
            text += "    cgcent %s\n" % self.cgcent
            text += "    fgcent %s\n" % self.fgcent
        elif self.method == "mg-para":
            text += "    pdime %i %i %i\n" % (self.pdime[0], self.pdime[1], self.pdime[2])
            text += "    ofrac %.1f\n" % self.ofrac
            text += "    cglen %.4f %.4f %.4f\n" % (self.cglen[0], self.cglen[1], self.cglen[2])
            text += "    fglen %.4f %.4f %.4f\n" % (self.fglen[0], self.fglen[1], self.fglen[2])
            text += "    cgcent %s\n" % self.cgcent
            text += "    fgcent %s\n" % self.fgcent
            if self.asyncflag == 1:
                text += "    async %i\n" % self.async
        text += "    mol %i\n" % self.mol
        if self.lpbe: text += "    lpbe\n"
        else: text += "    npbe\n"
        text += "    bcfl %s\n" % self.bcfl
        if self.istrng > 0:
            for ion in self.ion:
#                text += "    ion charge %.2f conc %.3f radius %.4f\n" % (ion[0], self.istrng,
ion[1])
                text += "    ion charge %.2f conc %.10f radius %.3f\n" % (float(ion[0]),
float(ion[1]), float(ion[2]))
        text += "    pdie %.4f\n" % self.pdie
        text += "    sdie %.4f\n" % self.sdie
        text += "    srfm %s\n" % self.srfm
        text += "    chgm %s\n" % self.chgm
        text += "    sdens %.2f\n" % self.sdens
        text += "    srad %.2f\n" % self.srad
        text += "    swin %.2f\n" % self.swin
        text += "    temp %.2f\n" % self.temp
        text += "    calcenergy %s\n" % self.calcenergy
        text += "    calcforce %s\n" % self.calcforce
        for write in self.write:
            text += "    write %s %s %s\n" % (write[0], write[1], write[2])
        text += "end\n"
        return text

class Input:
    """
        The input class.  Each input object is one APBS input file.
    """

    def __init__(self, pqrpath, size, method, asyncflag, pdie, sdie, srfm, chgm, sdens, srad,
swin, temp, ion, Frmt, saltFldrExt, unfrmGrdSpcng, dime_x, dime_y, dime_z, charge=0,
```

```
atompot=0, dielx=0, diely=0, dielz=0, pot=0, vdw=0, ivdw=0, edens=0, smol=0, sspl=0,
kappa=0, lap=0, ndens=0, qdens=0, istrng=0, potdx=0, npbe=0):
        """
            Initialize the input file class.  Each input file contains
            a PQR name, a list of elec objects, and a list of strings
            containing print statements.  For starters assume two
            ELEC statements are needed, one for the inhomgenous and
            the other for the homogenous dielectric calculations.

            Users can edit the elec statements and the print statements.

            This assumes you have already run psize, either by
                size.runPsize(/path/to/pqr) or

                size.parseString(string)
                size.setAll()

            Parameters
               pqrpath:   The path to the PQR file (string)
               size:      The Psize object (psize)
               method:    The method (para, auto, manual, async) to use
               asyncflag: 1 if async is desired, 0 otherwise
        """
        self.pqrpath = pqrpath
        self.asyncflag = asyncflag
        modelFldrLoc = string.find(pqrpath,"PQR_Format")
        modelFilePath=pqrpath[0:modelFldrLoc]
        extrctdFilePath=self.pqrpath[modelFldrLoc+len("PQR_Format"):len(self.pqrpath)]
        slashLoc=string.find(extrctdFilePath,"/",1)
        pdbId=self.pqrpath[modelFldrLoc+11:modelFldrLoc+11+slashLoc-1]
        period=string.find(pqrpath,".pqr")
        Nm=pqrpath[modelFldrLoc+len("PQR_Format/")+len(pdbId)+1:period]
        #print(Frmt.upper())
        dxpath=modelFilePath+Frmt.upper()+"_Format/"+pdbId+saltFldrExt
        # Initialize variables to default elec values
        numWrite=0
        if atompot != 0: numWrite=numWrite+1
        if charge != 0: numWrite=numWrite+1
        if dielx != 0: numWrite=numWrite+1
        if diely != 0: numWrite=numWrite+1
        if dielz != 0: numWrite=numWrite+1
        if edens != 0: numWrite=numWrite+1
        if ivdw != 0: numWrite=numWrite+1
        if lap != 0: numWrite=numWrite+1
        if ndens != 0: numWrite=numWrite+1
        if qdens != 0: numWrite=numWrite+1
```

```
if smol != 0: numWrite=numWrite+1
if sspl != 0: numWrite=numWrite+1
if pot != 0: numWrite=numWrite+1
if kappa != 0: numWrite=numWrite+1
if vdw != 0: numWrite=numWrite+1

elec1 = Elec(pqrpath, size, method, asyncflag, istrng, potdx)
if npbe != 0:
   setattr(elec1,"lpbe",0)
   setattr(elec1,"npbe",1)
#print(elec1.glen)
#if potdx == 0:
   #elec2 = Elec(pqrpath, size, method, asyncflag, saltFldrExt, istrng, potdx)
   #setattr(elec2, "sdie", 2.0)
   #setattr(elec2, "write", [])
#else:

if unfrmGrdSpcng=="uniform":
   #val=max(elec1.cglen)
   #print(elec1.glen)
   val=307.8030
   setattr(elec1,"cglen",(val,val,val))
   #val=max(elec1.fglen)
   val=142.6010
   setattr(elec1,"fglen",(val,val,val))

if chgm != "": setattr(elec1,"chgm",chgm)
if pdie != "": setattr(elec1,"pdie",pdie)
if sdie != "": setattr(elec1,"sdie",sdie)
if srfm != "": setattr(elec1,"srfm",srfm)
if sdens != "": setattr(elec1,"sdens",sdens)
if srad != "": setattr(elec1,"srad",srad)
if temp != "": setattr(elec1,"temp",temp)
if swin != "": setattr(elec1,"swin",swin)
if dime_x != "" and dime_y != "" and dime_z != "":
   bldDim=[int(dime_x),int(dime_y),int(dime_z)]
   setattr(elec1,"dime",bldDim)

if istrng != 0:
   numIon=string.count(ion,",")/3
   ions=string.split(ion,",")
   bld=[[ions[0],ions[1],ions[2]]]
   for ii in range(1,numIon):
      bld=bld+[[str(ions[ii*3]),str(ions[ii*3+1]),str(ions[ii*3+2])]]
   #print(elec1.ion)
   setattr(elec1,"ion",bld)
```

```
#if unfrmGrdSpcng=="uniform":
   #val=elec1.cglen[0]
   #print(elec1.glen)
   #bldCglen=[val,val,val]
   #setattr(elec1,"cglen",bldCglen)
   #val=max(elec1.fglen)
   #setattr(elec1,"fglen",(val,val,val))
   #val=max(elec1.fglen)
   #setattr(elec1,"flgen",[[val,val,val]])
sR=str(elec1.srad)
period=string.find(sR,".")
test=len(sR[period:len(sR)])
if test<4:
   for i in range(0,4-test):
         sR=sR+"0"

#dxFilePath=dxpath+Nm+"T"+str(elec1.temp)+"SR"+str(elec1.srad)
dxFilePath=dxpath+Nm+"T"+str(elec1.temp)+"SR"+sR
Count=0
bld=""
if atompot != 0:
   #dxFilePath=dxpath+"atompot/"+Nm+"T"+str(elec1.temp)+"SR"+str(elec1.srad)
   dxFilePath=dxpath+"atompot/"+Nm+"T"+str(elec1.temp)+"SR"+sR
   if Count>0:
         bld=bld+[["atompot",Frmt,dxFilePath]]
   else:
         bld=[["atompot",Frmt,dxFilePath]]
         Count=Count+1
if charge != 0:
   #dxFilePath=dxpath+"charge/"+Nm+"T"+str(elec1.temp)+"SR"+str(elec1.srad)
   dxFilePath=dxpath+"charge/"+Nm+"T"+str(elec1.temp)+"SR"+sR
   if Count>0:
         bld=bld+[["charge",Frmt,dxFilePath]]
   else:
         bld=[["charge",Frmt,dxFilePath]]
         Count=Count+1
if dielx != 0:
   #dxFilePath=dxpath+"dielx/"+Nm+"T"+str(elec1.temp)+"SR"+str(elec1.srad)
   #dxFilePath=dxpath+"dielx/"+Nm+"T"+str(elec1.temp)+"SR"+str(elec1.srad)
   dxFilePath=dxpath+"dielx/"+Nm+"T"+str(elec1.temp)+"SR"+sR
   if Count>0:
         bld=bld+[["dielx",Frmt,dxFilePath]]
   else:
         bld=[["dielx",Frmt,dxFilePath]]
         Count=Count+1
```

```
if diely != 0:
   #dxFilePath=dxpath+"diely/"+Nm+"T"+str(elec1.temp)+"SR"+str(elec1.srad)
   dxFilePath=dxpath+"diely/"+Nm+"T"+str(elec1.temp)+"SR"+sR
   if Count>0:
         bld=bld+[["diely",Frmt,dxFilePath]]
   else:
         bld=[["diely",Frmt,dxFilePath]]
         Count=Count+1
if dielz != 0:
   #dxFilePath=dxpath+"dielz/"+Nm+"T"+str(elec1.temp)+"SR"+str(elec1.srad)
   dxFilePath=dxpath+"dielz/"+Nm+"T"+str(elec1.temp)+"SR"+sR
   if Count>0:
         bld=bld+[["dielz",Frmt,dxFilePath]]
   else:
         bld=[["dielz",Frmt,dxFilePath]]
         Count=Count+1
if edens != 0:
   #dxFilePath=dxpath+"edens/"+Nm+"T"+str(elec1.temp)+"SR"+str(elec1.srad)
   dxFilePath=dxpath+"edens/"+Nm+"T"+str(elec1.temp)+"SR"+sR
   if Count>0:
         bld=bld+[["edens",Frmt,dxFilePath]]
   else:
         bld=[["edens",Frmt,dxFilePath]]
         Count=Count+1
if ivdw != 0:
   #dxFilePath=dxpath+"ivdw/"+Nm+"T"+str(elec1.temp)+"SR"+str(elec1.srad)
   dxFilePath=dxpath+"ivdw/"+Nm+"T"+str(elec1.temp)+"SR"+sR
   if Count>0:
         bld=bld+[["ivdw",Frmt,dxFilePath]]
   else:
         bld=[["ivdw",Frmt,dxFilePath]]
         Count=Count+1
if lap != 0:
   #dxFilePath=dxpath+"lap/"+Nm+"T"+str(elec1.temp)+"SR"+str(elec1.srad)
   dxFilePath=dxpath+"lap/"+Nm+"T"+str(elec1.temp)+"SR"+sR
   if Count>0:
         bld=bld+[["lap",Frmt,dxFilePath]]
   else:
         bld=[["lap",Frmt,dxFilePath]]
         Count=Count+1
if ndens != 0:
   #dxFilePath=dxpath+"ndens/"+Nm+"T"+str(elec1.temp)+"SR"+str(elec1.srad)
   dxFilePath=dxpath+"ndens/"+Nm+"T"+str(elec1.temp)+"SR"+sR
   if Count>0:
         bld=bld+[["ndens",Frmt,dxFilePath]]
   else:
```

```
            bld=[["ndens",Frmt,dxFilePath]]
            Count=Count+1
if kappa != 0:
   dxFilePath=dxpath+"kappa/"+Nm+"T"+str(elec1.temp)+"SR"+str(elec1.srad)
   if Count>0:
            bld=bld+[["kappa",Frmt,dxFilePath]]
   else:
            bld=[["kappa",Frmt,dxFilePath]]
            Count=Count+1
if pot != 0:
   #dxFilePath=dxpath+"pot/"+Nm+"T"+str(elec1.temp)+"SR"+str(elec1.srad)
   dxFilePath=dxpath+"pot/"+Nm+"T"+str(elec1.temp)+"SR"+sR
   if Count>0:
            bld=bld+[["pot",Frmt,dxFilePath]]
   else:
            bld=[["pot",Frmt,dxFilePath]]
            Count=Count+1
if qdens != 0:
   #dxFilePath=dxpath+"qdens/"+Nm+"T"+str(elec1.temp)+"SR"+str(elec1.srad)
   dxFilePath=dxpath+"qdens/"+Nm+"T"+str(elec1.temp)+"SR"+sR
   if Count>0:
            bld=bld+[["qdens",Frmt,dxFilePath]]
   else:
            bld=[["qdens",Frmt,dxFilePath]]
            Count=Count+1
if smol != 0:
   #dxFilePath=dxpath+"smol/"+Nm+"T"+str(elec1.temp)+"SR"+str(elec1.srad)
   dxFilePath=dxpath+"smol/"+Nm+"T"+str(elec1.temp)+"SR"+sR
   if Count>0:
            bld=bld+[["smol",Frmt,dxFilePath]]
   else:
            bld=[["smol",Frmt,dxFilePath]]
            Count=Count+1
if sspl != 0:
   #dxFilePath=dxpath+"sspl/"+Nm+"T"+str(elec1.temp)+"SR"+str(elec1.srad)
   dxFilePath=dxpath+"sspl/"+Nm+"T"+str(elec1.temp)+"SR"+sR
   if Count>0:
            bld=bld+[["sspl",Frmt,dxFilePath]]
   else:
            bld=[["sspl",Frmt,dxFilePath]]
            Count=Count+1
if vdw != 0:
   #dxFilePath=dxpath+"vdw/"+Nm+"T"+str(elec1.temp)+"SR"+str(elec1.srad)
   dxFilePath=dxpath+"vdw/"+Nm+"T"+str(elec1.temp)+"SR"+sR
   if Count>0:
            bld=bld+[["vdw",Frmt,dxFilePath]]
```

```
            else:
                    bld=[["vdw",Frmt,dxFilePath]]
                    Count=Count+1

        if Count == 0:
            bld=""

        setattr(elec1,"write",bld)

        elec2 = ""
        self.elecs = [elec1, elec2]

        i = string.rfind(pqrpath, "/") + 1
        self.pqrname = pqrpath[i:]
        self.prints=[]

        #if potdx == 0:
        #    self.prints = ["print elecEnergy 2 - 1 end"]
        #else:
                    #self.prints = []

    def __str__(self):
        """
            Return the text of the input file
        """
        text  = "read\n"
        text += "    mol pqr %s\n" % self.pqrpath
        text += "end\n"
        for elec in self.elecs:
            text += str(elec)
        for prints in self.prints:
            text += prints
        text += "\nquit\n"
        return text

    def printInputFiles(self,saltFldrExt,temp,srad,pdie,sdie):
        """
            Make the input file(s) associated with this object
        """
        modelFldrLoc=string.find(self.pqrpath,"PQR_Format")
        modelFilePath=self.pqrpath[0:modelFldrLoc]
        extrctdFilePath=self.pqrpath[modelFldrLoc+len("PQR_Format"):len(self.pqrpath)]
        slashLoc=string.find(extrctdFilePath,"/",1)
        #print(slashLoc)
#       pdbId=self.pqrpath[modelFldrLoc+11:modelFldrLoc+15]
        pdbId=self.pqrpath[modelFldrLoc+11:modelFldrLoc+11+slashLoc-1]
```

```
        #print(pdbId)
        #inFilePath=self.pqrpath[modelFldrLoc+11:modelFldrLoc+11+slashLoc-1]
        period=string.find(self.pqrpath,".pqr")
        inFilePath=self.pqrpath[modelFldrLoc+len("PQR_Format/")+len(pdbId)+1:period]
        #print(inFilePath)
        inFileOutput=modelFilePath+"IN_Format/"+pdbId+saltFldrExt+inFilePath

        if temp != "":
           inFileOutput=inFileOutput+"T"+str(temp)
        if srad != "":
           solvRadius=str(srad)
           period=string.find(solvRadius,".")
           test=len(solvRadius[period:len(solvRadius)])
           if test<4:
                   for i in range(0,4-test):
                           solvRadius=solvRadius+"0"
           inFileOutput=inFileOutput+"SR"+solvRadius
        #if pdie != "":
        #  inFileOutput=inFileOutput+"PD"+str(pdie)
        if sdie != "":
           inFileOutput=inFileOutput+"SD"+str(sdie)

        inFileOutput=inFileOutput+".in"
        outname=inFileOutput
        #print(inFileOutput)
                   #dxpath=modelFilePath+"DX_Format"+pdbId
                   #period=40
#                  outname=inFileOutput

        if self.asyncflag == 1:
           #outname = self.pqrpath[0:period] + "-para.in"

           # Temporarily disable async flag
           for elec in self.elecs:
              elec.asyncflag = 0
           file = open(outname, "w")
           file.write(str(self))
           file.close()

           # Now make the async files
           elec = self.elecs[0]

           nproc = elec.pdime[0] * elec.pdime[1] * elec.pdime[2]
           for i in range(int(nproc)):
              #outname = self.pqrpath[0:period] + "-PE%i.in" % i
              for elec in self.elecs:
```

```
                elec.asyncflag = 1
                elec.async = i
            file = open(outname, "w")
            file.write(str(self))
            file.close()

        else:
            #if period > 0:
                #outname = self.pqrpath[0:period] + ".in"
            #else:
                #outname = self.pqrpath + ".in"
            file = open(outname, "w")
            file.write(str(self))
            file.close()
            #print(outname)

    def dumpPickle(self):
        """
            Make a Python pickle associated with the APBS input parameters
        """
        period = string.find(self.pqrpath,".")
        if period > 0:
            outname = self.pqrpath[0:period] + "-input.p"
        else:
            outname = self.pqrpath + "-input.p"
        pfile = open(outname, "w")
        pickle.dump(self, pfile)
        pfile.close()

def splitInput(filename):
    """
        Split the parallel input file into multiple async file names

        Parameters
            filename:  The path to the original parallel input
                       file (string)
    """
    nproc = 0
    file = open(filename, 'rU')
    text = ""
    while 1:
        line = file.readline()
        if line == "": break
        text += line
        line = string.strip(line)
        if line.startswith("pdime"): # Get # Procs
```

```
            words = string.split(line)
            nproc = int(words[1]) * int(words[2]) * int(words[3])

    if nproc == 0:
        sys.stderr.write("%s is not a valid APBS parallel input file!\n" % filename)
        sys.stderr.write("The inputgen script was unable to asynchronize this file!\n")
        sys.exit(2)

    period = string.find(filename,".")
    for i in range(nproc):
        outname = filename[0:period] + "-PE%i.in" % i
        outtext = string.replace(text, "mg-para\n","mg-para\n    async %i\n" % i)
        outfile = open(outname, "w")
        outfile.write(outtext)
        outfile.close()

def usage():
    """
        Display the usage information for this script
    """
    size = psize.Psize()
    usage = "\n"
    usage = usage + "Use this script to generate new APBS input files or split an
existing\n"
    usage = usage + "parallel input file into multiple async files.\n\n"
    usage = usage + "Usage: inputgen_edited.py [opts] <filename>\n"
    usage = usage + "Optional Arguments:\n"
    usage = usage + "  --help             : Display this text\n"
    usage = usage + "  --split            : Split an existing parallel input file to multiple\n"
    usage = usage + "                       async input files.\n"
    usage = usage + "  --potdx            : Create an input to compute an electrostatic
potential map.\n"
    usage = usage + "  --method=<value>     : Force output file to write a specific APBS\n"
    usage = usage + "                         ELEC method.  Options are para (parallel), auto\n"
    usage = usage + "                         (automatic), manual (manual), or async
(asynchronous).\n"
    usage = usage + "                         solve.  async will result in multiple input files.\n"
    usage = usage + "  --cfac=<value>       : Factor by which to expand molecular\n"
    usage = usage + "                         dimensions to get coarse grid dimensions.\n"
    usage = usage + "                         [default = %g]\n" % size.getConstant("cfac")
    usage = usage + "  --fadd=<value>       : Amount to add to molecular dimensions to
get\n"
    usage = usage + "                         fine grid dimensions.\n"
    usage = usage + "                         [default = %g]\n" % size.getConstant("fadd")
    usage = usage + "  --space=<value>      : Desired fine mesh resolution\n"
    usage = usage + "                         [default = %g]\n" % size.getConstant("space")
```

```
    usage = usage + "  --gmemfac=<value>    : Number of bytes per grid point required\n"
    usage = usage + "                       for sequential MG calculation\n"
    usage = usage + "                       [default = %g]\n" % size.getConstant("gmemfac")
    usage = usage + "  --gmemceil=<value>   : Max MB allowed for sequential MG\n"
    usage = usage + "                       calculation.  Adjust this to force the\n"
    usage = usage + "                       script to perform faster calculations (which\n"
    usage = usage + "                       require more parallelism).\n"
    usage = usage + "                       [default = %g]\n" % size.getConstant("gmemceil")
    usage = usage + "  --ofrac=<value>      : Overlap factor between mesh partitions\n"
    usage = usage + "                       [default = %g]\n" % size.getConstant("ofrac")
    usage = usage + "  --redfac=<value>     : The maximum factor by which a domain\n"
    usage = usage + "                       dimension can be reduced during focusing\n"
    usage = usage + "                       [default = %g]\n" % size.getConstant("redfac")
#    usage = usage + "  --istrng=<value>     : Ionic strength (M). Na+ anc Cl- ions will
be\n"
#    usage = usage + "                        used.\n"
    usage = usage + "  --istrng=<value>     : (value=0) ions NOT included. (value=1)
ions\n"
    usage = usage + "                       included. [default = 0]\n"
    usage = usage + "  --pdie=<value>           : Protein Dielectric Constant. This is
usually\n"
    usage = usage + "                       a value between 2 to 20, where lower values\n"
    usage = usage + "                       consider only electronic polarization and\n"
    usage = usage + "                       higher values consider additional\n"
    usage = usage + "                       polarization due to intramolecular motion.\n"
    usage = usage + "                       [default = 2.0]\n"
    usage = usage + "  --sdie=<value>           : Solvent Dielectric Constant. Bulk water
at\n"
    usage = usage + "                       biologically-relevant temperatures is usually\n"
    usage = usage + "                       modeled with a value of 78 to 80.\n"
    usage = usage + "                       [default = 78.5]\n"
    usage = usage + "  --temp=<value>           : Temperature.\n"
    usage = usage + "                       [default = 298.15]\n"
    usage = usage + "  --Frmt=<value>           : APBS Output Data Format. A string
that\n"
    usage = usage + "                       specifies the format for writing out the data\n"
    usage = usage + "                       Possible values are listed:\n"
    usage = usage + "                       dx\n"
    usage = usage + "                       avs\n"
    usage = usage + "                       uhbd\n"
    usage = usage + "APBS Output Data Types  : APBS Output Data Types each hold
their own\n"
    usage = usage + "                        specific flag.\n"
    usage = usage + "  --atompot         : Electrostatic Potential at each atom.\n"
    usage = usage + "  --charge                  : Charge Distribution.\n"
    usage = usage + "  --dielx                   : Dielectric Map shifted 1/2 grid spacing.\n"
```

```
    usage = usage + "                   in x-direction.\n"
    usage = usage + "  --diely                  : Dielectric Map shifted 1/2 grid spacing.\n"
    usage = usage + "                   in y-direction.\n"
    usage = usage + "  --dielz                  : Dielectric Map shifted 1/2 grid spacing.\n"
    usage = usage + "                   in z-direction.\n"
    usage = usage + "  --edens                  : Energy Density.\n"
    usage = usage + "  --ivdw                   : Infalted van der Waals.\n"
    usage = usage + "  --kappa                  : Ion-accessibility Kappa Map.\n"
    usage = usage + "  --lap             : Laplacian.\n"
    usage = usage + "  --ndens                  : Total Mobile Ion Density.\n"
    usage = usage + "  --pot             : Electrostatic Potential Distribution.\n"
    usage = usage + "  --qdens                  : Total Mobile Charge Density.\n"
    usage = usage + "  --smol                   : Solvent Excluded Surface.\n"
    usage = usage + "  --sspl                   : Spline-based Solvent Excluded Surface.\n"
    usage = usage + "  --vdw                    : van der Waals (Molecular) Surface.\n"
    sys.stderr.write(usage)
    sys.exit(2)

def main():

    import getopt
    filename = ""
    saltFldrExt=""
    shortOptList = ""
    longOptList =
["help","split","potdx","method=","cfac=","fadd=","space=","gmemceil=","gmemfac=","
ofrac=","redfac=","istrng=","pdie=","sdie=","srfm=","chgm=","sdens=","srad=","swin=",
"temp=","charge","atompot","dielx","diely","dielz","pot","vdw","ivdw","edens","smol","
sspl","kappa","lap","ndens","qdens","unfrmGrdSpcng","Frmt=","ion=","dime-
x=","dime-y=","dime-z=","npbe"]

    try:
        opts, args = getopt.getopt(sys.argv[1:], shortOptList, longOptList)
    except getopt.GetoptError, details:
        sys.stderr.write("Option error (%s)!\n" % details)
        usage()

    if len(args) != 2:
        sys.stderr.write("Invalid argument list!\n")
        usage()
    else:
        filename = args[0]
        saltFldrExt = args[1]

    method = ""
    size = psize.Psize()
```

```
async = 0
split = 0
istrng = 0
potdx = 0
pdie=""
sdie=""
srfm=""
chgm=""
sdens=""
srad=""
swin=""
temp=""
Frmt="dx"
unfrmGrdSpcng=""
charge=0
atompot=0
dielx=0
diely=0
dielz=0
pot=0
vdw=0
ivdw=0
edens=0
smol=0
sspl=0
kappa=0
lap=0
ndens=0
qdens=0
ion=""
dime_x=""
dime_y=""
dime_z=""
npbe=0

for o, a in opts:
    if o == "--help":
        usage()
    if o == "--split": split = 1
    if o == "--potdx": potdx = 1
    if o == "--method":
        if a == "para":
            sys.stdout.write("Forcing a parallel calculation\n")
            method = "mg-para"
        elif a == "auto":
            sys.stdout.write("Forcing a sequential calculation\n")
```

```
            method = "mg-auto"
        elif a == "async":
            sys.stdout.write("Forcing an asynchronous calculation\n")
            method = "mg-para"
            async = 1
        elif a == "manual":
            sys.stdout.write("Forcing a manual calculation\n")
            method = "mg-manual"
        else:
            sys.stdout.write("Incorrect method argument: %s\n" % a)
            sys.stdout.write("Defaulting to memory dependent result\n")
    if o == "--cfac":
        size.setConstant("cfac", float(a))
    if o == "--fadd":
        size.setConstant("fadd", float(a))
    if o == "--space":
        size.setConstant("space", float(a))
    if o == "--gmemfac":
        size.setConstant("gmemfac", int(a))
    if o == "--gmemceil":
        size.setConstant("gmemceil",  int(a))
    if o == "--ofrac":
        size.setConstant("ofrac", float(a))
    if o == "--redfac":
        size.setConstant("redfac", float(a))
    if o == "--istrng":
        istrng = float(a)
    if o == "--pdie":
        if a != "": pdie=float(a)
    if o == "--sdie":
        if a != "": sdie=float(a)
    if o == "--srfm":
        if a != "": srfm=str(a)
    if o == "--chgm":
        if a != "": chgm=float(a)
    if o == "--sdens":
        if a != "": sdens=float(a)
    if o == "--srad":
        if a != "": srad=float(a)
    if o == "--swin":
        if a != "": swin=float(a)
    if o == "--temp":
        if a != "": temp=float(a)
    if o == "--charge": charge=1
    if o == "--atompot": atompot=1
    if o == "--dielx": dielx=1
```

```
        if o == "--diely": diely=1
        if o == "--dielz": dielz=1
        if o == "--pot": pot=1
        if o == "--vdw": vdw=1
        if o == "--ivdw": ivdw=1
        if o == "--edens": edens=1
        if o == "--smol": smol=1
        if o == "--sspl": sspl=1
        if o == "--kappa": kappa=1
        if o == "--lap": lap=1
        if o == "--ndens": ndens=1
        if o == "--qdens": qdens=1
        if o == "--npbe": npbe=1
        if o == "--unfrmGrdSpcng": unfrmGrdSpcng="uniform"
        if o == "--Frmt": Frmt=str(a)
        if o == "--ion": ion=str(a)
        if o == "--dime-x": dime_x=str(a)
        if o == "--dime-y": dime_y=str(a)
        if o == "--dime-z": dime_z=str(a)

    if split == 1:
        splitInput(filename)
    else:
        size.runPsize(filename)
        input = Input(filename, size, method, async, pdie, sdie, srfm, chgm, sdens, srad,
swin, temp, ion, Frmt, saltFldrExt, unfrmGrdSpcng, dime_x, dime_y, dime_z, charge,
atompot, dielx, diely, dielz, pot, vdw, ivdw, edens, smol, sspl, kappa, lap, ndens, qdens,
istrng, potdx, npbe)
        input.printInputFiles(saltFldrExt,temp,srad,pdie,sdie)

if __name__ == "__main__": main()
```

IN_File_Editer.sh – Command file for generating APBS input files

```
#!/bin/bash

memMax=975
cfac=3
fadd=40
Frmt=dx
unfrmGrdSpcng=false
pdie=4.0
pdbFile=$Fldr"PDB_Format/"$PDB".pdb"
pqrFldr=$Fldr"PQR_Format/"$PDB"/"
funFldr=$Fldr"programs/APBS-1.4-linux-static-x86_64/share/apbs/tools/manip/"
inputgen=$funFldr"inputgen_edited.py"
```

```
method=auto
# psize.py inputs
flags="--cfac="$cfac
flags=$flags" --fadd="$fadd
flags=$flags" --gmemceil="$memMax
flags=$flags" --method="$method

# APBS inputs
flags=$flags" --pdie="$pdie
flags=$flags" --Frmt="$Frmt
flags=$flags" --istrng="$istrng
flags=$flags" --dime-x="$dime_x
flags=$flags" --dime-y="$dime_y
flags=$flags" --dime-z="$dime_z
flags=$flags" --unfrmGrdSpcng"

# Flag write types
for P in ${WT_List[@]}; do
        if [ ${P} == "atompot" ]
                then
                        flags=$flags" --atompot"
        fi
        if [ ${P} == "charge" ]
                then
                        flags=$flags" --charge"
        fi
        if [ ${P} == "dielx" ]
                then
                        flags=$flags" --dielx"
        fi
        if [ ${P} == "diely" ]
                then
                        flags=$flags" --diely"
        fi
        if [ ${P} == "dielz" ]
                then
                        flags=$flags" --dielz"
        fi
        if [ ${P} == "edens" ]
                then
                        flags=$flags" --edens"
        fi
        if [ ${P} == "ivdw" ]
                then
                        flags=$flags" --ivdw"
        fi
```

```
        if [ ${P} == "lap" ]
                then
                        flags=$flags" --lap"
        fi
        if [ ${P} == "ndens" ]
                then
                        flags=$flags" --ndens"
        fi
        if [ ${P} == "kappa" ]
                then
                        flags=$flags" --kappa"
        fi
        if [ ${P} == "pot" ]
                then
                        flags=$flags" --pot"
        fi
        if [ ${P} == "qdens" ]
                then
                        flags=$flags" --qdens"
        fi
        if [ ${P} == "smol" ]
                then
                        flags=$flags" --smol"
        fi
        if [ ${P} == "sspl" ]
                then
                        flags=$flags" --sspl"
        fi
        if [ ${P} == "vdw" ]
                then
                        flags=$flags" --vdw"
        fi
done

if [ $npbe == "true" ]
        then
                flags=$flags" --npbe"
fi

numFiles=1

let Counter=0
for (( i=0; i<$numFiles; i++ )) do
        if [ $istrng == 1 ]
                then
                flags=$flags" --ion="$ionData
```

```
        fi
        dynamicflags=$flags" --temp="$temperature
        dynamicflags=$dynamicflags" --srad="$maxRadius
        dynamicflags=$dynamicflags" --sdie="$sdie
        pqrFile=$pqrFldr"pH"$pH".pqr"
        python $inputgen $dynamicflags $pqrFile $fldrExt
done
```

SECTION D5. Automation of PROPKA and PDB2PQR

PROPKA [55] and PDB2PQR [54, 53] assign atomic charges and radii to a PDB structure file and prepares it for the electrostatic computation of APBS. Automation of these programs is controlled by the shell script below.

run_PROPKA_PDB2PQR.sh – Command file for executing PROPKA and PDB2PQR

```
#!/bin/bash

PDB_Input_File=$Fldr"PDB_Format/"$PDB".pdb"
PDB2PQR_File=$Fldr"programs/pdb2pqr-1.8/main.py"
force_field="PARSE"

PQR_Output_File=$Fldr"PQR_Format/"$PDB"/pH"$pH".pqr"
PROPKA_Output_File=$Fldr"PROPKA_Format/"$PDB"/pH"$pH".propka"

if [ $whiteSpace == true ]
        then
                python $PDB2PQR_File --ff=$force_field --with-ph=$pH --whitespace -v
$PDB_Input_File $PQR_Output_File
else
        python $PDB2PQR_File --ff=$force_field --with-ph=$pH -v $PDB_Input_File
$PQR_Output_File
fi

mv $Fldr"PQR_Format/"$PDB"/pH"$pH".propka" $PROPKA_Output_File

cd $Fldr
```

APPENDIX E. Green Fluorescent Protein (GFP) Mutations

The table below shows the mutations applied to green fluorescent protein (GFP). To assess the utility of ZPRED for drug/protein design, 9 mutant structures of GFP (PDB id: 2y0g) were synthesized with the displayed mutations to their surface residues.

Valence ->	-21	-22	-10	-22	-12	-21	-12	-10	-6
	Mutant Number								
Mutated AA	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
3		**K->N**							
6		**E->Q**	**E->Q**						
11	**V->E**							**V->S**	**V->E**
15	**L->S**					**L->R**	**L->R**	**L->K**	**L->R**
21		**D->N**							
26		**K->E**		**K->E**		**K->E**	**K->R**		
28									**S->R**
30							**S->R**	**S->E**	
41		**K->D**	**K->D**	**K->D**	**K->D**	**K->D**	**K->D**		
43	**T->D**					**T->D**		**T->R**	**T->R**
44				**L->N**	**L->N**				
47	**I->S**							**I->K**	
52		**K->L**	**K->N**						
72	**A->S**	**A->S**	**A->S**	**A->S**	**A->S**	**A->S**	**A->S**	**A->S**	**A->S**
76		**D->N**	**D->N**						

93	**V->D**					**V->K**	**V->K**	**V->S**	**V->R**
99	**F->E**		**F->N**			**F->D**	**F->D**	**F->D**	**F->D**
105	**N->D**							**N->E**	**N->E**
107		**K->E**		**K->E**		**K->E**	**K->E**		
113		**K->D**	**K->D**	**K->D**	**K->D**				
117		**D->Q**							
118	**T->K**							**T->K**	**T->K**
120	**V->E**							**V->R**	**V->R**
122		**R->E**		**R->E**		**R->D**	**R->D**		
126		**K->D**	**K->D**	**K->D**	**K->D**	**K->R**	**K->R**		
128	**I->K**								
131		**K->N**	**K->N**						
133		**D->N**	**D->N**						
140		**K->N**	**K->N**						
147	**S->D**							**S->D**	**S->D**
151						**Y->E**	**Y->E**		
153	**M->E**	**M->D**						**M->E**	**M->E**
155		**D->Q**							
158						**K->E**	**K->R**		
162		**K->E**		**K->E**		**K->D**	**K->D**		
164	**N->D**					**N->D**	**N->D**	**N->D**	**N->K**
166		**K->D**	**K->D**	**K->D**	**K->D**				

176	**V->T**			**V->S**	**V->S**	**V->S**			
178	**L->E**			**L->N**	**L->N**	**L->E**	**L->E**	**L->E**	**L->E**
183									**Q->E**
184	**Q->D**					**Q->E**	**Q->K**	**Q->D**	**Q->R**
186	**T->E**							**T->E**	**T->E**
190		**D->N**	**D->N**						
200	**Y->D**		**Y->T**				**Y->S**	**Y->D**	**Y->D**
202	**S->E**					**S->D**	**S->D**	**S->R**	
204	**Q->D**					**Q->E**	**Q->R**	**Q->E**	**Q->E**
208	**S->E**							**S->D**	**S->D**
214		**K->E**		**K->E**					
219	**V->D**	**V->E**		**V->S**	**V->S**	**V->R**		**V->E**	**V->E**
221	**L->A**	**L->E**		**L->N**	**L->N**				
223	**F->E**					**F->D**	**F->D**	**F->K**	**F->K**
225	**T->D**							**T->E**	
227	**A->E**							**A->D**	**A->K**

APPENDIX F. Derivation of the Helmholtz-Smoluchowski Equation

Mathematically deriving the ζ aids in understanding the physical constraints (i.e. model assumptions) involved in the conversion of this potential into a rate of motion. The key assumption is that the ζ exists as the slip plane position, which for this derivation of the Helmholtz-Smoluchowski equation is assumed to satisfy the "no slip" or "stick" boundary condition [11, p. 259].

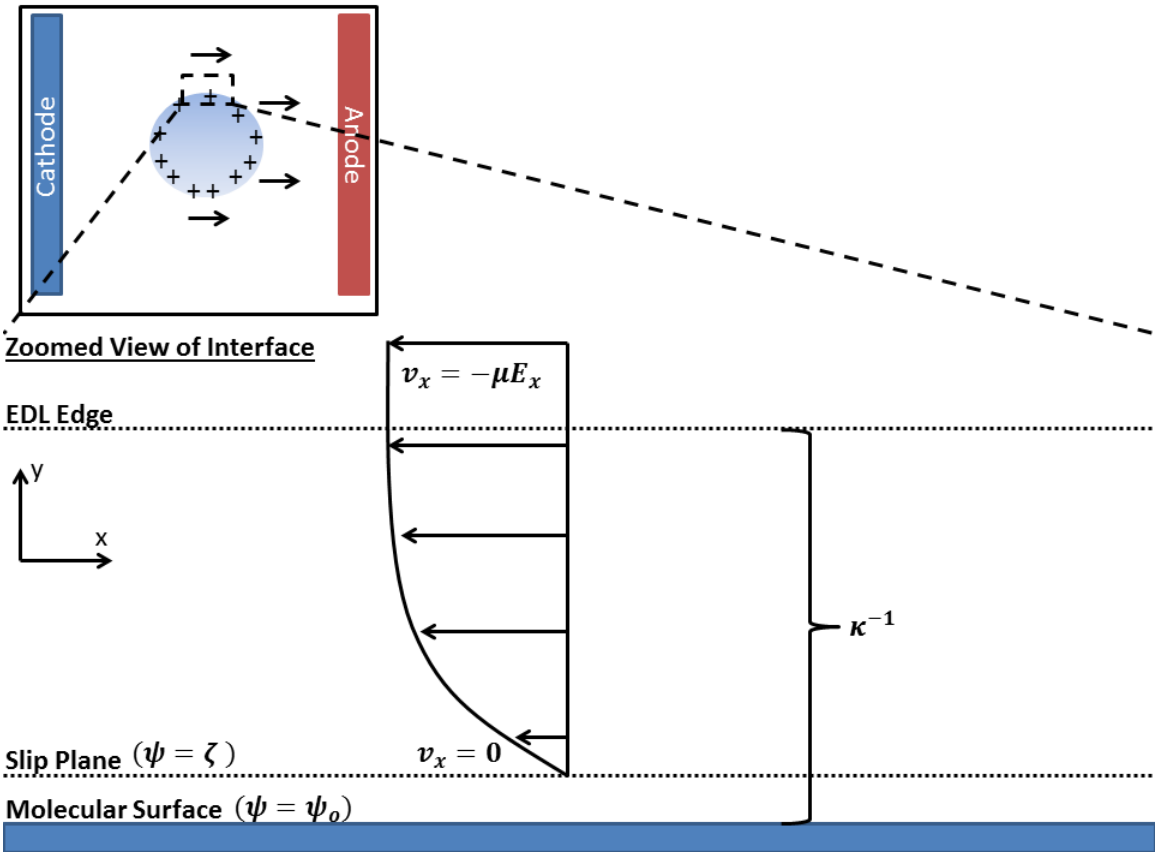

Figure F-1. The Zeta Potential and Electrophoretic Motion.

First, start with the Navier-Stokes equation (NSE) [141, p. 438], which defines the momentum transfer (and thus fluid flow) from the molecular surface to bulk solvent. The NSE can be written in **vector notation**:

$$\rho\left(\frac{\partial v}{\partial t} + v \cdot \nabla v\right) = -\nabla P + \eta\nabla^2 v + F_{ext}$$

Or equivalently, in its vector components for Cartesian coordinates,

X-component: $$\rho\left(\frac{\partial v_x}{\partial t}+v_x\frac{\partial v_x}{\partial x}+v_y\frac{\partial v_x}{\partial y}+v_z\frac{\partial v_x}{\partial z}\right)=-\frac{\partial P}{\partial x}+\eta\left(\frac{\partial^2 v_x}{\partial x^2}+\frac{\partial^2 v_x}{\partial y^2}+\frac{\partial^2 v_x}{\partial z^2}\right)+F_{x,ext}$$

Y-component: $$\rho\left(\frac{\partial v_y}{\partial t}+v_x\frac{\partial v_y}{\partial x}+v_y\frac{\partial v_y}{\partial y}+v_z\frac{\partial v_y}{\partial z}\right)=-\frac{\partial P}{\partial y}+\eta\left(\frac{\partial^2 v_y}{\partial x^2}+\frac{\partial^2 v_y}{\partial y^2}+\frac{\partial^2 v_y}{\partial z^2}\right)+F_{y,ext}$$

Z-component: $$\rho\left(\frac{\partial v_z}{\partial t}+v_x\frac{\partial v_z}{\partial x}+v_y\frac{\partial v_z}{\partial y}+v_z\frac{\partial v_z}{\partial z}\right)=-\frac{\partial P}{\partial z}+\eta\left(\frac{\partial^2 v_z}{\partial x^2}+\frac{\partial^2 v_z}{\partial y^2}+\frac{\partial^2 v_z}{\partial z^2}\right)+F_{z,ext}$$

where ρ is solution density, v is the solution velocity vector (holding x (v_x),y (v_y) and z (v_z) components), t is time, P is pressure, η is the solution viscosity, and F_{ext} is an external force (holding x ($F_{x,ext}$), y ($F_{y,ext}$) and z ($F_{z,ext}$) components) acting on the volume of mass under consideration

Assuming the fluid to be incompressible (i.e. uniform density), which is physically realistic for liquids, allows the Continuity equation to be applied as a constraint:

$$\nabla\cdot v=\frac{\partial v_x}{\partial x}+\frac{\partial v_y}{\partial y}+\frac{\partial v_z}{\partial z}=0$$

Assuming steady-state (no variables change with time, so $\frac{\partial v}{\partial t}=0$), laminar flow (i.e. low **Reynolds number**), Newtonian liquid (i.e. constant viscosity), incompressible fluid (constant density), no pressure gradient ($\nabla P=0$), and unidirectional flow along the x-axis ($v_y=v_z=0$) the NSE is significantly simplified to the following form:

$$\eta\frac{d^2 v_x}{dy^2}=-F_{x,ext}$$

Considering the particle is undergoing electrophoresis, it is necessary to take into account this external force.

$$F_{ext}=\rho_e E_x$$

where ρ_e is charge density and E_x is the applied electric field along the x-axis

To define the charge density, we use the Poisson equation

$$\rho_e = -\nabla \cdot (\varepsilon_r \varepsilon_o \nabla \psi) = \varepsilon_r \varepsilon_o \left(\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} + \frac{\partial^2 \psi}{\partial z^2}\right)$$

where ε_r is the solution relative dielectric, ε_o is **vacuum permittivity**, ψ is the electric potential

Assuming a constant solution dielectric, the Poisson equation can be substituted into the NSE to represent the charge density being acted on by the applied electric field.

$$\eta \frac{d^2 v_x}{dy^2} = \varepsilon_r \varepsilon_o \frac{d^2 \psi}{dy^2} E_x$$

Integrating the equation:

$$\eta \int \frac{d^2 v_x}{dy^2} dy = \varepsilon_r \varepsilon_o E_x \int \frac{d^2 \psi}{dy^2} dy$$

$$\eta \frac{dv_x}{dy} = \varepsilon_r \varepsilon_o E_x \frac{d\psi}{dy} + C$$

Then applying the boundary conditions ($\frac{dv_x}{dy} = \frac{d\psi}{dy} = 0 \; at \; y \to \infty$) yields:

$$\eta \frac{dv_x}{dy} = \varepsilon_r \varepsilon_o E_x \frac{d\psi}{dy}$$

Integrating the equation once again:

$$\eta \int \frac{dv_x}{dy} dy = \varepsilon_r \varepsilon_o E_x \int \frac{d\psi}{dy} dy$$

$$\eta v_x = \varepsilon_r \varepsilon_o E_x \psi + C$$

Then applying the boundary conditions:

$$\psi = \zeta, \;\; v_x = 0, \;\; y = 0$$

$$v_x = \frac{\varepsilon_r \varepsilon_o E_x \zeta}{\eta}$$

Thus through derivation of the Helmholtz-Smoluchowski equation, the zeta potential can be perceived to be the effective driving force of electrophoretic motion of a solvated molecule.

APPENDIX G. Calculated Solution Properties

This appendix contains solution property values and is organized by electrolyte name in alphabetical order. Chemical names inside brackets represent their molar concentration (e.g. [KCl] represents the molar concentration of KCl). The symbol ($\boldsymbol{\varepsilon_r}$) represents the solution relative dielectric constant. Although $\boldsymbol{\varepsilon_r}$ is well known to decrease with the increasing concentration of electrolytes, ZPRED assumes it is independent of the ion concentration (for validation of this assumption see **APPENDIX C**; **Fig. C1**).

Table G-1. Citrate Phosphate Solution Properties. Density was calculated using **Eq. C22.** Viscosity was calculated using **Eq. C12**. Values for 20°C were obtained from [148] and extrapolated to other temperatures.

		20°C (ε_r=80.089)		
[$H_3C_6H_5O_7$]	**[Na_2HPO_4]**	**pH (pre-mixing w/ protein)**	**Density (kg/L)**	**Viscosity (mPa s)**
0.00962556	0.000392080	2.55	0.998337	1.00283
0.00660753	0.00339911	3.15	0.998341	1.00279
0.00461247	0.00540067	4.10	0.998389	1.00293
0.00339764	0.00710233	5.00	0.998450	1.00317
0.00229992	0.00768706	6.00	0.998536	1.00353
0.000986842	0.00899123	7.00	0.998724	1.00437
0.000224903	0.00960949	8.00	0.998214	1.00216

[$H_3C_6H_5O_7$]	**[Na_2HPO_4]**	**Temperature (°C)**	**ε_r**	**Density (kg/L)**	**Viscosity (mPa s)**
0.000986842	0.00899123	4	86.152	1.00049	1.554350
0.000986842	0.00899123	6	85.369	1.00044	1.462030
0.000986842	0.00899123	8	84.593	1.00033	1.378110
0.000986842	0.00899123	10	83.825	1.00018	1.301610
0.000986842	0.00899123	12	83.063	0.999976	1.231670
0.000986842	0.00899123	14	82.309	0.999728	1.167570
0.000986842	0.00899123	16	81.562	0.999436	1.108680
0.000986842	0.00899123	18	80.822	0.999101	1.054440
0.000986842	0.00899123	20	80.089	0.998724	1.004370
0.000986842	0.00899123	22	79.362	0.998306	0.958050
0.000986842	0.00899123	24	78.643	0.997849	0.915116
0.000986842	0.00899123	26	77.930	0.997354	0.875240
0.000986842	0.00899123	28	77.223	0.996822	0.838135
0.000986842	0.00899123	30	76.524	0.996255	0.803547

0.000986842	0.00899123	32	75.831	0.995653	0.771249
0.000986842	0.00899123	34	75.144	0.995017	0.741041
0.000986842	0.00899123	36	74.463	0.994349	0.712743
0.000986842	0.00899123	38	73.789	0.993649	0.686195
0.000986842	0.00899123	40	73.121	0.992919	0.661252

Table G-2. HCl and KCl Solution Properties. Density was calculated from **Eq. C19** [86]. Viscosities were calculated from **Eq. C9** [56]**.**

		25°C (ε_r=78.285)		**35°C** (ε_r=74.803)	
[HCl]	**[KCl]**	**Density (kg/L)**	**Viscosity (mPa s)**	**Density (kg/L)**	**Viscosity (mPa s)**
0.005	0	0.997133	0.890450	0.994119	0.719215
0.005	0.005	0.997367	0.890411	0.994350	0.719241
0.005	0.045	0.999229	0.890101	0.996196	0.719448
0.005	0.095	1.001546	0.889726	0.998493	0.719714
0.005	0.495	1.019693	0.887450	1.016480	0.722278
0.005	0.995	1.041423	0.886664	1.038022	0.726829

Table G-3. HCl/NaOH and NaCl Solution Properties. Density was calculated for mixtures of HCl and NaCl using **Eq. C20** and for mixtures of NaOH and NaCl using **Eq. C21**. Viscosity was calculated for mixtures of HCl and NaCl using **Eq. C10** and for mixtures of NaOH and NaCl using **Eq. C11**.

			25°C (ε_r=78.285)		
[HCl]	**[NaOH]**	**[NaCl]**	**pH (pre-mixing w/ protein)**	**Density (kg/L)**	**Viscosity (mPa s)**
0.1	0	0	1.06	0.998806	0.895833
0.01	0	0.09	2.14	1.000873	0.897207
0.001	0	0.099	3.00	1.001076	0.897343
0.0001	0	0.0999	4.35	1.001096	0.897356
0.00001	0	0.09999	5.34	1.001098	0.897357
0.000001	0	0.099999	5.47	1.001099	0.897358
0	0.00001	0.09999	6.12	1.001099	0.897356
0	0.0001	0.0999	9.30	1.001100	0.897342
0	0.001	0.099	10.90	1.001102	0.897203
0	0.01	0.09	12.16	1.001133	0.895790
0	0.1	0	13.39	1.001383	0.878553

[HCl]	**[NaCl]**	**Temperature (°C)**	$\boldsymbol{\varepsilon_r}$	**Density (kg/L)**	**Viscosity (mPa s)**
0.1	0	5	85.759	1.001806	1.526317
0.1	0	15	81.935	1.000895	1.144919
0.1	0	25	78.285	0.998806	0.895833
0.1	0	35	74.803	0.995770	0.724074
0.1	0	45	71.479	0.991938	0.600414
0.1	0	50	69.873	0.989758	0.551124

0.1	0	55	68.304	0.987416	0.508229
0.1	0	60	66.771	0.984921	0.470646
0.1	0	65	65.271	0.982280	0.437513
0.1	0	70	63.805	0.979500	0.408138
0.1	0	75	62.372	0.976585	0.381958
0.01	0.09	5	85.759	1.003953	1.528348
0.01	0.09	15	81.935	1.002998	1.146569
0.01	0.09	25	78.285	1.000873	0.897207
0.01	0.09	35	74.803	0.997808	0.725243
0.01	0.09	45	71.479	0.993954	0.601430
0.01	0.09	50	69.873	0.991766	0.552077
0.01	0.09	55	68.304	0.989416	0.509126
0.01	0.09	60	66.771	0.986915	0.471494
0.01	0.09	65	65.271	0.984269	0.438317
0.01	0.09	70	63.805	0.981485	0.408903
0.01	0.09	75	62.372	0.978568	0.382688
0.001	0.099	5	85.759	1.004165	1.528552
0.001	0.099	15	81.935	1.003205	1.146733
0.001	0.099	25	78.285	1.001076	0.897343
0.001	0.099	35	74.803	0.998009	0.725358
0.001	0.099	45	71.479	0.994153	0.601529
0.001	0.099	50	69.873	0.991963	0.552170
0.001	0.099	55	68.304	0.989613	0.509214
0.001	0.099	60	66.771	0.987111	0.471576
0.001	0.099	65	65.271	0.984465	0.438396
0.001	0.099	70	63.805	0.981680	0.408977
0.001	0.099	75	62.372	0.978763	0.382759

Table G-4. KCl Solution Properties. Density was calculated using Laliberte's model [56]. Viscosities were calculated using **Eq. C4** [57].

	25°C (ε_r=78.285)	
[KCl]	**Density (kg/L)**	**Viscosity (mPa s)**
0.000001	0.997045	0.892000
0.00001	0.997045	0.892000
0.0001	0.997049	0.891999
0.001	0.997092	0.891990
0.005	0.997278	0.891950
0.01	0.997511	0.891899
0.05	0.999372	0.891510
0.1	1.001688	0.891052
0.5	1.019823	0.888541
1.0	1.041539	0.888167

Table G-5. $KClO_4$ Solution Properties. Due to the difficulty in finding $KClO_4$ solution property data, data for $KClO_3$ was used instead. Density was calculated using **Eq. C17**. Viscosities were calculated from **Eq. C7** [59]

	25°C (ε_r=78.285)	
[$KClO_4$]	**Density (kg/L)**	**Viscosity (mPa s)**
0.000001	0.997045	0.890171
0.00001	0.997046	0.890180
0.0001	0.997053	0.890208
0.001	0.997129	0.890280
0.005	0.997464	0.890343
0.01	0.997883	0.890336
0.05	1.001224	0.889786

Table G-6. KH_2PO_4 Solution Properties. Density was calculated using Laliberte's model [86]. Viscosities were calculated using **Eq. C5** [56]**.**

	25°C (ε_r=78.285)	
[KH_2PO_4]	**Density (kg/L)**	**Viscosity (mPa s)**
0.000001	0.997048	0.890166
0.00001	0.997048	0.890168
0.0001	0.997058	0.890183
0.001	0.997143	0.890335
0.005	0.997524	0.891011
0.01	0.997999	0.891856
0.05	1.001807	0.898601
0.1	1.006566	0.907002
0.5	1.044639	0.974532
0.75	1.068435	1.023320

Table G-7. KNO_3 Solution Properties. Density was calculated using **Eq. C16** [149]**.** Viscosity values were calculated from **Eq. C6** [58]**.**

	25°C (ε_r=78.285)	
[KNO_3]	**Density (kg/L)**	**Viscosity (mPa s)**
0.000001	0.997045	0.890171
0.00001	0.997046	0.890180
0.0001	0.997051	0.890206
0.001	0.997107	0.890260
0.005	0.997356	0.890246
0.01	0.997667	0.890141
0.05	1.000146	0.888859
0.1	1.003227	0.887090
0.5	1.027211	0.875602
1.0	1.055613	0.870043

Table G-8. NaCl Solution Properties. Density was calculated using **Eq. C18**. [86] Viscosity was calculated using **Eq. C8.** [56]

[NaCl]	Temperature (°C)	ε_r	Density (kg/L)	Viscosity (mPa s)
0.1	5	85.759	1.004188	1.528574
0.1	15	81.935	1.003228	1.146751
0.1	25	78.285	1.001099	0.897357
0.1	35	74.803	0.998031	0.725371
0.1	45	71.479	0.994175	0.601540
0.1	50	69.873	0.991985	0.552180
0.1	55	68.304	0.989635	0.509223
0.1	60	66.771	0.987133	0.471585
0.1	65	65.271	0.984486	0.438404
0.1	70	63.805	0.981702	0.408985
0.1	75	62.372	0.978784	0.382766

REFERENCES

[1] R. J. Hunter, ZETA POTENTIAL IN COLLOID SCIENCE Principles and Applications, 3rd ed., San Diego, CA: ACADEMIC PRESS, 1988.

[2] T. M. Riddick, "Control of colloid stability through zeta potential," *Zeta Meter Corp*, 1968.

[3] G. A. Thiagarajan, "A comparison of biophysical characterization techniques in predicting monoclonal antibody stability," *MAbs*, vol. 8, pp. 1088-1097, 2016.

[4] A. S. Parmar, J. K. James, D. R. Grisham, D. H. Pike and V. Nanda, "Dissecting Electrostatic Contributions to Folding and Self-Assembly Using Design Multicomponent Peptide Systems," *Journal of the American Chemical Society*, vol. 138, pp. 4362-4367, 2016.

[5] M. T. Schermeyer, A. K. Woll, B. Kokke, M. Eppink and J. Hubbuch, "Characterization of highly concentrated antibody solution - A toolbox for the description of protein long-term solution stability," *MAbs*, pp. 1-17, 2017.

[6] W. F. Long and P. Labute, "Calibrative approaches to protein solubility modeling of a mutant series using physicochemical descriptors," *Journal of Computer Aided Molecular Design*, vol. 24, pp. 907-916, 2010.

[7] P. M. Buck, S. Kumar, X. Wang, N. J. Agrawal, B. L. Trout and S. K. Singh, "Computational methods to predict therapeutic protein aggregation," *Therapeutic proteins: Methods and protocols*, pp. 425-415, 2012.

[8] J. M. Perchiacca, C. C. Lee and P. M. Tessier, "Optimal charged mutations in the complementarity-determining regions that prevent domain antibody aggregation are dependent on the antibody scaffold," *Protein Engineering, Design & Selection*, vol. 27, pp. 29-39, 2014.

[9] C. N. Haarmeyer, M. D. Smith, S. P. Chundawat, D. Sammond and T. A. Whitehead, "Insights into cellulase-lignin non-specific binding revealed by computational redesign of the surface of green fluorescent protein," *Biotechnol Bioeng*, vol. 114, pp. 740-750, 2017.

[10] M. Lundqvist, J. Stigler, G. Elia, I. Lynch, T. Cedervall and K. A. Dawson,

"Nanoparticle size and surface properties determin the protein corona with possible implications for biological impacts," *Proc Natl Acad Sci USA,* vol. 105, pp. 14265-14270, 2008.

[11] I. N. Serdyuk, N. R. Zaccai and J. Zaccai, Methods in Molecular Biophysics Structure, Dynamics, Function, Cambridge University Press, 2007.

[12] D. L. Chapman, "LI. A contribution to the theory of electrocapillarity," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science,* vol. 25, no. 148, pp. 475-481, 1913.

[13] M. A. Brown, G. V. Bossa and S. May, "Emergence of a Stern Layer from the Incorporation of Hydration Interactions into the Gouy-Chapman Model of the Electrical Double Layer," *Langmuir,* vol. 31, pp. 11477-11483, 2015.

[14] I. D. Kuntz and W. Kauzmann, "Hydration of Proteins and Polypeptides," *Advances in Protein Chemistry,* vol. 28, pp. 239-345, 1974.

[15] J. A. Rupley and G. Careri, "Protein hydration and function," *Advances in Protein Chemistry,* vol. 41, pp. 37-172, 1991.

[16] J. N. Israelachvili and G. E. Adams, "Measurement of forces between two mica surfaces in aqueous electrolyte solutions in the range 0-100 nm," *Journal of the Chemical Society, Faraday Transactions 1,* vol. 74, pp. 975-1001, 1978.

[17] J. G. de la Torre, "Hydration from hydrodynamics. General considerations and applications of bead modeling to globular proteins," *Biophysical Chemistry,* vol. 93, pp. 159-170, 2001.

[18] O. Stern, "The theory of the electric double layer," *Z. Elektrochem. Angew. Phys. Chem.,* vol. 30, pp. 508-516, 1924.

[19] J. N. Israelachvili, Intermolecular and Surface Forces, San Diego, CA: Academic Press, 2011.

[20] H. B. Bull and K. Breese, "Water and Solute Binding by Proteins 1. Electrolytes," *Archives of Biochemistry and Biophysics,* vol. 137, pp. 299-305, 1970.

[21] H. B. Bull and K. Breese, "Protein Hydration I. Binding Sites," *Archives of Biochemistry and Biophysics,* vol. 128, pp. 488-496, 1968.

[22] G. N. Lewis and M. Randall, "The activity coefficient of strong electrolytes,"

Journal of the American Chemical Society, vol. 43, no. 5, pp. 1112-1154, 1921.

[23] T. G. Pedersen and J. B. Ifft, "Buoyant titration of ovalbumin in four alkali halides. Hydration and ion binding," *Carlsberg Research Communications,* vol. 43, pp. 65-76, 1978.

[24] C. E. Kundrot and F. M. Richards, "Crystal Structure of Hen Egg-White Lysozyme at a Hydrostatic Pressure of 1000 Atmospheres," *Journal of Molecular Biology,* vol. 193, no. 1, pp. 157-170, 1987.

[25] F. C. Bernstein, T. F. Koetzle, G. J. Williams, E. E. Meyer, M. D. Brice, J. R. Rodgers, O. Kennard, T. Shimanouchi and M. Tasumi, "The Protein Data Bank: A Computer-based Archival File For Macromolecular Structures," *Journal of Molecular Biology,* vol. 112, p. 535, 1977.

[26] K. D. Collins, "Ions from the Hofmeister series and osmolytes: effects on proteins in solution and in the crystallization process," *Methods,* vol. 34, pp. 300-311, 2004.

[27] E. Y. Lau and V. V. Krishnan, "Temperature dependence of protein-hydration hydrodynamics by molecular dynamics simulations," *Biophysical Chemistry,* vol. 130, pp. 55-64, 2007.

[28] S. Ebbinghaus, S. J. Kim, M. Heyden, X. Yu, M. Gruebele, D. M. Leitner and M. Havenith, "Protein Sequence and pH Dependent Hydration Probed by Terahertz Spectroscopy," *Journal of the American Chemical Society,* vol. 130, pp. 2374-2375, 2008.

[29] Y. Zhang and P. S. Cremer, "Interactions between macromolecules and ions: the Hofmeister series," *Current Opinion in Chemical Biology,* vol. 10, pp. 658-663, 2006.

[30] Y. Zhang and P. S. Cremer, "Chemistry of Hofmeister Anions and Osmolytes," *Annual Review of Physical Chemistry,* vol. 61, pp. 63-83, 2010.

[31] R. L. Baldwin, "How Hofmeister Ion Interactions Affect Protein Stability," *Biophysical Journal,* vol. 71, pp. 2056-2063, 1996.

[32] K. D. Collins and M. W. Washabaugh, "The Hofmeister effect and the behaviour of water at interfaces," *Quarterly Review of Biophysics,* vol. 18, no. 4, pp. 323-422, 1985.

[33] Z. Yang, "Hofmeister effects: an explanation for the impact of ionic liquids on

biocatalysis," *Journal of Biotechnology,* vol. 144, pp. 12-22, 2009.

[34] H. Donald, B. Jenkins and Y. Marcus, "Viscosity B-Coefficients of Ions in Solution," *Chemical Reviews,* vol. 95, no. 8, pp. 2695-2724, 1995.

[35] R. Hidalgo-Alvarez, A. Martin, A. Fernandez, D. Bastps, F. Martinez and F. J. de las Nieves, "Electrokinetic properties, colloidal stability and aggregation kinetics of polymer colloids," *Advances in Colloid and Interface Science,* vol. 67, pp. 1-118, 1996.

[36] R. Leberman, "The Hofmeister series and ionic strength," *Federation of European Biochemical Societies Letters,* vol. 284, no. 2, pp. 293-294, 1991.

[37] M. Y. Kiriukhin and K. D. Collins, "Dynamic hydration numbers for biologically important ions," *Biophysical Chemistry,* vol. 99, pp. 155-168, 2002.

[38] K. D. Collins, "Sticky ions in biological systems," *Proceedings of the National Academy of Sciences,* vol. 92, pp. 5553-5557, 1995.

[39] W. G. Eversole and W. W. Boardman, "The Effect of Electrostatic Forces on Electrokinetic Potentials," *Journal of Chemical Physics,* vol. 9, pp. 798-801, 1941.

[40] B. R. Midmore and R. J. Hunter, "The effect of electrolyte concentration and co-ion type on the ζ-potential of polystyrene latices," *Journal of Colloid and Interface Science,* vol. 122, no. 2, pp. 521-529, 1988.

[41] A. Einstein, "A New Determination of Molecular Dimensions," *University of Zurich Thesis Dissertation,* pp. 1-21, 1905.

[42] A. D. Cadman, R. Fleming and R. H. Guy, "Diffusion of lysozyme chloride in water and aqueous potassium chloride solutions," *Biophysical Journal,* vol. 37, no. 3, pp. 569-574, 1981.

[43] P. Doherty and G. B. Benedek, "The effect of electric charge on the diffusion of macromolecules," *The Journal of Chemical Physics,* vol. 61, no. 12, pp. 5426-5434, 1974.

[44] M. J. Stephen, "Spectrum of light scattered from charged macromolecules in solution," *The Journal of Chemical Physics,* vol. 55, no. 8, pp. 3878-3883, 1971.

[45] R. Schor and E. N. Serrallach, "Theoretical studies of the translational diffusion coefficient of charged macromolecules applied to bovine serum albumin," *The*

Journal of Chemical Physics, vol. 70, pp. 3012-3015, 1979.

[46] M. Muthukumar, "Ordinary-extraordinary transition in dynamics of solutions of charged macromolecules," *Proceedings of the National Academy of Sciences,* vol. 113, no. 45, pp. 12627-12632, 2016.

[47] D. C. Henry, "The Cataphoresis of Suspended Particles. Part I. The Equation of Cataphoresis," *Proceedings of the Royal Society A,* vol. 133, pp. 106-129, 1931.

[48] I. Kemp and E. K. Rideal, "On the relation between particle size and cataphoretic mobility," *Transactions of the Faraday Society,* vol. 31, pp. 1347-1357, 1935.

[49] R. H. Ottewill and J. N. Shaw, "Electrophoretic Stuides on Polystyrene Latices," *Journal of Electroanalytical Chemistry and Interfacial Electrochemistry,* vol. 37, no. 1, pp. 133-142, 1972.

[50] S. K. Basak and M. R. Ladisch, "Correlation of Electrophoretic Mobilities of Proteins and Peptides with Their Physicochemical Properties," *Analytical Biochemistry,* vol. 226, pp. 51-58, 1995.

[51] H. A. Abramson, M. H. Gorin and L. S. Moyer, "THE POLAR GROUPS OF PROTEIN AND AMINO ACID SURFACES IN LIQUIDS," *Chemical Reviews,* vol. 24, no. 2, pp. 345-366, 1939.

[52] D. E. Kuehner, J. Engmann, F. Fergg, M. Wernick, H. W. Blanch and J. M. Prausnitz, "Lysozyme Net Charge and Ion Binding in Concentrated Aqueous Electrolyte Solutions," *The Journal of Physical Chemistry B,* vol. 103, no. 8, pp. 1368-1374, 1999.

[53] T. J. Dolinsky, P. Czodrowski, H. Li, J. E. Nielsen, J. H. Jensen, G. Klebe and N. A. Baker, "PDB2PQR: expanding and upgrading automated preparation of biomolecular structures for molecular simulations," *Nucleic Acids Research,* vol. 35, 2007.

[54] T. J. Dolinsky, J. E. Nielsen, J. A. McCammon and N. A. Baker, "PDB2PQR: an automated pipeline for the setup of Poisson-Boltzmann electrostatics calculations," *Nucleic Acids Research,* vol. 32, pp. W665-W667, 2004.

[55] M. H. M. Olsson, C. R. Sondergaard, M. Rostkowski and J. H. Jensen, "PROPKA3: Consistent Treatment of Internal and Surface Residues in Empirical pKa Predictions," *Journal of Chemical Theory and Computation,* vol. 7, pp. 525-

537, 2011.

[56] M. Laliberte, "Model for Calculating the Viscosity of Aqueous Solutions," *Journal of Chemical and Engineering Data,* vol. 52, no. 2, pp. 321-335, 2007.

[57] F. A. Goncalves and J. Kestin, "The viscosity of NaCl and KCl solutions in the range 25 - 50C," *Berichte der Bunsengesellschaft fur physikalische chemie,* vol. 81, no. 11, pp. 1156-1161, 1977.

[58] T. H. Doan and J. Sangster, "Viscosities of Concentrated Aqueous Solutions of Some 1:1, 2:1, and 3:1 Nitrates at 25C," *Journal of Chemical and Engineering Data,* vol. 26, no. 2, pp. 141-144, 1981.

[59] G. Jones and S. K. Talley, "The Viscosity of Aqueous Solutions as a Function of the Concentration," *Journal of the American Chemical Society,* vol. 55, no. 2, pp. 624-642, 1933.

[60] F. Chenlo, R. Moreira, G. Pereira and M. J. Vazquez, "Viscosity of Binary and Ternary Aqueous Systems of NaH2PO4, Na2HPO4, Na3PO4, KH2PO4, K2HPO4, and K3PO4," *Journal of Chemical and Engineering Data,* vol. 41, no. 4, pp. 906-909, 1996.

[61] H. Ohshima, "A Simple Expression for Henry's Function for the Retardation Effect in Electrophoresis of Spherical Colloidal Particles," *Journal of Colloid and Interface Science,* vol. 168, pp. 269-271, 1994.

[62] H. Ohshima, "Henry's Function for Electrophoresis of a Cylindrical Colloidal Particle," *Journal of Colloid and Interface Science,* vol. 180, no. 1, pp. 299-301, 1996.

[63] S. A. Allison and V. T. Tran, "Modeling the Electrophoresis of Rigid Polyions: Application to Lysozyme," *Biophysical Journal,* vol. 68, pp. 2261-2270, 1995.

[64] A. V. Delgado, F. Gonzalez-Caballero, R. J. Hunter, L. K. Koopal and J. Lyklema, "Measurement and interpretation of electrokinetic phenomena," *Journal of Colloid and Interface Science,* vol. 309, pp. 194-224, 2007.

[65] K. S. Chae and A. M. Lenhoff, "Computation of the Electrophoretic Mobility of Proteins," *Biophysical Journal,* vol. 68, no. 3, pp. 1120-1127, 1995.

[66] R. Xu, Particle Characterization: Light Scattering Methods, New York: Kluwer Academic Publishers, 2002.

[67] D. A. Case, J. T. Berryman, R. M. Betz, D. S. Cerutti, T. E. Cheatham III, T. A. Darden, R. E. Duke, T. J. Giese, H. Gohlke, A. W. Goetz, N. Homeyer, S. Izadi, P. Janowski, J. Kaus, A. Kovalenko, T. S. Lee, S. LeGrand, P. Li, T. Luchko, R. Luo, B. Madej, K. M. Merz, G. Monard, P. Needham, H. Nguyen, H. T. Nguyen, I. Omelyan, A. Onufriev, D. R. Roe, A. Roitberg, R. Salomon-Ferrer, C. L. Simmerling, W. Smith, J. Swails, R. C. Walker, J. Wang, R. M. Wolf, X. Wu, D. M. York and P. A. Kollman, "AMBER 2015," San Francisco, 2015.

[68] Y. Marcus, "Ionic radii in aqueous solutions," *Chemical Reviews,* vol. 88, no. 8, pp. 1475-1498, 1988.

[69] J. T. G. Overbeek, "Theorie der Elektrophorese - Der Relaxationseffekt," *Kolloid-Beihefte,* vol. 54, pp. 287-364, 1943.

[70] F. Booth, "The cataphoresis of spherical, solid non-conducting particles in a symmetrical electrolyte," *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences,* vol. 203, no. 1075, pp. 514-533, 1950.

[71] P. H. Wiersema, A. L. Loeb and J. T. G. Overbeek, "Calculation of the Electrophoretic Mobility of a Spherical Colloid Particle," *Journal of Colloid and Interface Science,* vol. 22, pp. 78-99, 1966.

[72] R. W. O'Brien and L. R. White, "Electrophoretic Mobility of a Spherical Colloidal Particle," *Journal of the Chemical Society, Faraday Transactions 2: Molecular and Chemical Physics,* vol. 74, pp. 1607-1626, 1978.

[73] H. Ohshima, T. W. Healy and L. R. White, "Approximate Analytical Expressions for the Electrophoretic Mobility of Spherical Colloidal Particles and the Conductivity of their Dilute Suspensions," *Journal of the Chemical Society, Faraday Transactions 2,* vol. 79, no. 11, pp. 1613-1628, 1983.

[74] C. S. Mangelsdorf and L. R. White, "Effects of Stern-layer Conductance on Electrokinetic Transport Properties of Colloidal Particles," *Journal of the Chemical Society, Faraday Transactions,* vol. 86, no. 16, pp. 2859-2870, 1990.

[75] S. R. Deshiikan and K. D. Papadopoulos, "Modified Booth equation for the calculation of zeta potential," *Colloid and Polymer Science,* vol. 276, pp. 117-124, 1998.

[76] C. S. Mangelsdorf and L. R. White, "The dynamic double layer Part 1 Theory of a mobile Stern layer," *Journal of the Chemical Society, Faraday Transactions,* vol.

94, no. 16, pp. 2441-2452, 1998.

[77] H. Ohshima, "Approximate Analytic Expression for the Electrophoretic Mobility of a Spherical Colloidal Particle," *Journal of Colloid and Interface Science,* vol. 239, pp. 587-590, 2001.

[78] S. A. Allison, "Modeling the Electrophoresis of Rigid Polyions. Inclusion of Ion Relaxation," *Macromolecules,* vol. 29, pp. 7391-7401, 1996.

[79] S. Mazur, C. Chen and S. A. Allison, "Modeling the Electrophoresis of Short Duplex DNA: Counterions K+ and Tris+," *The Journal of Physical Chemistry B,* vol. 105, pp. 1100-1108, 2001.

[80] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov and P. E. Bourne, "The Protein Data Bank," *Nucleic Acids Research,* vol. 28, no. 1, pp. 235-242, 2000.

[81] J. A. Maier, C. Martinez, K. Kasavajhala, L. Wickstrom, K. E. Hauser and C. Simmerling, "ff14SB: Improving the accuracy of protein side chain and backbone parameters from ff99SB," *Journal of Chemical Theory and Computation,* vol. 11, pp. 3696-3713, 2015.

[82] G. D. Phillies, "Effects of intermacromolecular interactions on diffusion. I. Two-component solutions," *The Journal of Chemical Physics,* vol. 60, pp. 976-982, 1974.

[83] J. G. de la Torre, M. L. Huertas and B. Carrasco, "Calculation of hydrodynamic properties of globular proteins from their atomic-level structure," *Biophysical Journal,* vol. 78, pp. 719-730, 2000.

[84] P. J. Fleming and K. G. Fleming, "HullRad: Fast Calculations of Folded and Disordered Protein and Nucleic Acid Hydrodynamic Properties," *Biophysical Journal,* vol. 114, pp. 856-869, 2018.

[85] M. F. Sanner, A. J. Olson and J.-C. Spehner, "Reduced Surface: An Efficient Way to Compute Molecular Surfaces," *Biopolymers,* vol. 38, no. 3, pp. 305-320, 1996.

[86] M. Laliberte and W. E. Cooper, "Model for calculating the Density of Aqueous Electrolyte Solutions," *Journal of Chemical Engineering Data,* vol. 49, pp. 1141-1151, 2004.

[87] M. N. Davies, C. P. Toseland, D. S. Moss and D. R. Flower, "Benchmarking pKa

prediction," *BMC Biochemistry,* vol. 7, no. 18, pp. 1-12, 2006.

[88] N. A. Baker, D. Sept, S. Joseph, M. J. Holst and J. A. McCammon, "Electrostatics of nanosystems applications to microtubules and the ribosome," *Proceedings of the National Academy of Sciences,* vol. 98, no. 18, pp. 10037-10041, 2001.

[89] M. T. Neves-Petersen and S. B. Petersen, "Protein electrostatics: A review of the equation and methods used to model electrostatic equations in biomolecules - Applications in biotechnology," *Biotechnology Annual Review,* vol. 9, pp. 315-395, 2003.

[90] Encyclopedia of Colloid and Interface Science, New York, NY: Springer, 2013, pp. 1421-1432.

[91] N. Rawat and P. Biswas, "Shape, flexibility and packing of proteins and nucleic acids in complexes," *Phys. Chem. Chem. Phys.,* vol. 13, pp. 9632-9643, 2011.

[92] A. J. Sophianopoulos, C. K. Rhodes, D. N. Holcomb and K. E. van Holde, "Physical Studies of Lysozyme," *The Journal of Biological Chemistry,* vol. 237, no. 4, pp. 1107-1112, 1962.

[93] D. Schneidman-Duhovny, M. Hammel, J. A. Tainer and A. Sali, "Accurate SAXS Profile Computation and its Assessment by Contrast Variation Experiments," *Biophysical Journal,* vol. 105, pp. 962-974, 2013.

[94] S. Venkataramani, J. Truntzer and D. R. Coleman, "Thermal stability of high concentration lysozyme across varying pH: a Fourier transform infrared study," *Journal of Pharmacy and BioAllied Sciences,* vol. 5, no. 2, pp. 148-153, 2013.

[95] H.-M. Lee, Y. W. Kim and J. K. Baird, "Electrophoretic mobility and zeta-potential of lysozyme crystals in aqueous solutions of some 1:1 electrolytes," *Journal of Crystal Growth,* vol. 232, pp. 294-300, 2001.

[96] J. Jacob, C. Krafft, K. Welfle and W. Saenger, "Melting Points of Lysozyme and Ribonuclease A Crystals Correlated with Protein Unfolding: a Raman Spectroscopic Study," *Acta Crystallographica Section D,* vol. D54, pp. 74-80, 1998.

[97] R. Diamond, "Real-space refinement of the structure of hen egg-white lysozyme," *Journal of Molecular Biology,* vol. 82, no. 3, pp. 371-374, 1974.

[98] Y. Zhang and P. S. Cremer, "The inverse and direct Hofmeister series for

lysozyme," *Proceedings of the National Academy of Sciences,* vol. 106, no. 36, pp. 15249-15253, 2009.

[99] L. L. Sorret, M. A. DeWinter, D. K. Schwartz and T. W. Randolph, "Challenges in Predicting Protein-Protein Interactions from Measurements of Molecular Diffusivity," *Biophysical Journal,* vol. 111, pp. 1831-1824, 2016.

[100] G. Alderton, W. H. Ward and H. L. Fevold, "Isolation of Lysozyme from Egg White," *Journal of Biological Chemistry,* vol. 157, pp. 43-58, 1945.

[101] O. G. Hampe, "Conformation of Lysozyme in Aqueous Solution: Effect of Ionic Strength and Protein Concentration," *European Journal of Biochemistry,* vol. 31, pp. 32-37, 1972.

[102] U. Bohme and U. Scheler, "Effective charge of bovine serum albumin determined by electrophoresis NMR," *Chemical Physics Letters,* vol. 435, pp. 342-345, 2007.

[103] C. Tanford, S. A. Swanson and W. S. Shore, "Hydrogen Ion Equilibria of Bovine Serum Albumin," *Journal of the American Chemical Society,* vol. 77, no. 24, pp. 6414-6421, 1955.

[104] W. F. Harrington, P. Johnson and R. H. Ottewill, "Bovine Serum Albumin and its Behavior in Acid Solution," *Biochemical Journal,* vol. 62, no. 4, pp. 569-582, 1956.

[105] J. M. Creeth, "The use of the Gouy diffusiometer with dilute protein solutions. An assessment of the accuracy of the method," *Biochemical Journal,* vol. 51, no. 1, pp. 10-17, 1952.

[106] C. Tanford and J. G. Buzzell, "The Viscosity of Aqueous Solutions of Bovine Serum Albumin between pH 4.3 and 10.5," *The Journal of Physical Chemistry,* vol. 60, no. 2, pp. 225-231, 1956.

[107] K. S. Schmitz and M. Lu, "Effect of titration charge on the diffusion of bovine serum albumin," *Proceedings of the National Academt of Sciences,* vol. 80, pp. 425-429, 1983.

[108] Y. S. Oh and C. S. Johnson, "The wave vector dependence of diffusion coefficients in photon correlation spectroscopy of protein solutions," *The Journal of Chemical Physics,* vol. 74, no. 5, pp. 2717-2720, 1981.

[109] A. Michnik, "Thermal Stability of Bovine Serum Albumin DSC Study," *Journal of*

Thermal Analysis and Calorimetry, vol. 71, pp. 509-519, 2003.

[110] S. Gumpen, P. O. Hegg and H. Martens, "Thermal stability of fatty acid serum albumin complexes studied by differential scanning calorimetry," *Biochimica et Biophysica Acta,* vol. 574, pp. 189-196, 1979.

[111] M. Yamasaki, H. Yano and K. Aoki, "Differential scanning calorimetric studies on bovine serum albumin: I. Effects of pH and ionic strength," *International Journal of Biological Macromolecules,* vol. 12, no. 4, pp. 263-268, 1990.

[112] C. Giancola, C. D. Sena, D. Fessas, G. Graziano and G. Barone, "DSC studies on bovine serum albumin denaturation Effects of ionic strength and SDS concentration," *International Journal of Biological Macromolecules,* vol. 20, pp. 193-204, 1997.

[113] M. Paulsson, P. O. Hegg and H. B. Castberg, "Thermal stability of whey proteins studied by differential scanning calorimetry," *Thermochimica Acta,* vol. 95, pp. 435-440, 1985.

[114] P. Doty and R. F. Steiner, "Macro-Ions. I. Light Scattering Theory and Experiments with Bovine Serum Albumin," *The Journal of Chemical Physics,* vol. 20, no. 1, pp. 85-94, 1952.

[115] S. N. Timasheff and R. Townend, "Molecular Interactions in B-Lactoglobulin. V. The Association of the Genetic Species of B-Lactoglublin below the Isoelectric Point," *Journal of the American Chemical Society,* vol. 83, no. 2, pp. 464-469, 1961.

[116] Y. Nozaki, L. G. Bunville and C. Tanford, "Hydrogen Ion Titration Curves of B-Lactoglobulin," *Journal of the American Chemical Society,* vol. 81, no. 21, pp. 5523-5529, 1959.

[117] K. R. Cannan, A. H. Palmer and A. C. Kibrick, "The Hydrogen Ion Dissociation Curve of B-Lactoglobulin," *Journal of Biological Chemistry,* vol. 142, pp. 803-822, 1942.

[118] M. Gottschalk, H. Nilsson, H. Roos and B. Halle, "Protein self-association in solution: The bovine B-lactoglobulin dimer and octamer," *Protein Science,* vol. 12, pp. 2404-2411, 2003.

[119] P. R. Majhi, R. R. Ganta, R. P. Vanam, E. Seyrek, K. Giger and P. L. Dubin, "Electrostatically Driven Protein Aggregation: B-Lactoglobulin at Low Ionic

Strength," *Langmuir,* vol. 22, pp. 9150-9159, 2006.

[120] D. Mercadante, L. D. Melton, G. E. Norris, T. S. Loo, M. A. K. Williams, R. C. J. Dobson and G. B. Jameson, "Bovine B-Lactoglobulin Is Dimeric Under Imitative Physiological Conditions: Dissociation Equilibrium and Rate Constants over the pH Range of 2.5-7.5," *Biophysical Journal,* vol. 103, pp. 303-312, 2012.

[121] C. Tanford and Y. Nozaki, "Physico-Chemical Comparison of B-Lactoglobulins A and B," *The Journal of Biological Chemistry,* vol. 234, no. 11, pp. 2874-2877, 1959.

[122] S. Brownlow, J. H. M. Cabral, R. Cooper, D. R. Flower, S. J. Yewdall, I. Polikarpov, A. C. T. North and L. Sawyer, "Bovine B-lactoglobulin at 1.8 A resolution - still an enigmatic lipocalin," *Structure,* vol. 5, no. 4, pp. 481-495, 1997.

[123] S. Prabakaran and S. Damodaran, "Thermal Unfolding of B-Lactoglobulin: Characterization of Initial Unfolding Events Responsible for Heat-Induced Aggregation," *Journal of Agricultural and Food Chemistry,* vol. 45, no. 11, pp. 4303-4308, 1997.

[124] K. O. Pedersen, "Ultracentrifugal and electrophoretic studies on the millk proteins. II. The lactoglobulin of palmer," *Biochemical Journal,* vol. 30, no. 6, pp. 961-970, 1936.

[125] F. J. Rubio-Hernandez, F. Carrique and E. Ruiz-Reina, "The primary electroviscous effect in colloidal suspensions," *Advances in Colloid and Interface Science,* vol. 107, pp. 51-60, 2004.

[126] A. S. Parmar and M. Muschol, "Hydration and Hydrodynamic Interactions of Lysozyme: Effects of Chaotropic and Kosmotropic Ions," *Biophysical Journal,* vol. 97, no. 2, pp. 590-598, 2009.

[127] B. E. Conway and A. Dobry-Cuclaux, RHEOLOGY Theory and Applications Chapter 3 VISCOSITY OF SUSPENSIONS OF ELECTRICALLY CHARGED PARTICLES AND SOLUTIONS OF POLYMERIC ELECTROLYTES, vol. 3rd, F. R. Eirich, Ed., New York, NY: ACADEMIC PRESS, 1960, pp. 83-120.

[128] S. C. Gill and P. H. von Hippel, "Calculation of protein extinction coefficients from amino acid sequence data," *Analytical Biochemistry,* vol. 182, pp. 319-326, 1989.

[129] P. Sharma, N. Verma, P. K. Singh, S. Korpole and Ashish, "Characterization of heat induced spherulites of lysozyme reveals new insight on amyloid initiation,"

Scientific Reports, vol. 6, p. 22475, 2016.

[130] E. Glueckauf, "Bulk Dielectric Constant of Aqueous Electrolyte Solutions," *Transactions of the Faraday Society,* vol. 60, pp. 1637-1645, 1964.

[131] N. C. Fitzkee and G. D. Rose, "Reassessing random-coil statistics in unfolded proteins," *Proceedings of the Naional Academy of Sciences,* vol. 101, no. 34, pp. 12497-12502, 2004.

[132] V. Bloomfield, "The Structure of Bovine Serum Albumin at Low pH," *Biochemistry,* vol. 5, no. 2, pp. 684-689, 1966.

[133] M. E. Reichmann and P. A. Charlwood, "Light-scattering and sedimentation studies of bovine serum albumin at low pH," *Canadian Journal of Chemistry,* vol. 32, no. 12, pp. 1092-1099, 1954.

[134] H. Eilers and J. Korff, "The Significance of the Phenomenon of the Electrical Charge on the Stability of Hydrophobic Dispersions," *Transactions of the Faraday Society,* vol. 35, pp. 229-241, 1940.

[135] U. Katzel, "Dynamic Light Scattering for the Characterization of Polydisperse Fractal Systems by the Example of Pyrogenic Silica," der Technischen Universitat Dresden, 2007.

[136] D. L. Nelson and M. M. Cox, Lehninger Principles of Biochemistry, 5th ed., New York, NY: W. H. Freeman and Company, 2008.

[137] R. A. Serway and J. W. Jewett, Physics for Scientists and Engineers with Modern Physics, Cengage Learning, 2007.

[138] M. Abramowitz and I. A. Stegun, Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, 10th ed., National Bureau of Standard Applied Mathematics Series 55, 1972.

[139] E. E. Allen, "Note 169," *MTAC,* vol. 8, p. 240, 1954.

[140] W. J. Cody and H. C. Thacher, "Rational Chebyshev Approximations for the Exponential Integral E1(x)," *Mathematics of Computation,* vol. 22, pp. 641-649, 1968.

[141] B. J. Kirby, Micro- and Nanoscale Fluid Mechanics Transport in Microfluidic Devices, New York, NY: Cambridge University Press, 2010.

[142] C. A. Miller and P. Neogi, Interfacial Phenomena Equilibrium and Dynamic Effects, 2nd ed., Boca Raton, FL,: Taylor & Francis Group, LLC, 2008.

[143] R. Qiao and P. Wu, "Concept of double layers," Virginia Tech, [Online]. Available: https://sites.google.com/a/g.clemson.edu/ionic-liquids/home/concept-of-double-layers.

[144] J. Perrin, *J. Chim. Phys.*, vol. 2, p. 601, 1904.

[145] B. Maribo-Mogensen, G. M. Kontogeorgis and K. Thomsen, "Modeling of Dielectric Properties of Aqueous Salt Solutions with an Equation of State," *The Journal of Physical Chemistry B*, vol. 117, pp. 10523-10533, 2013.

[146] C. G. Malmberg and A. A. Maryott, "Dielectric Constant of Water from 0 to 100 C," *Journal of Research of the National Bureau of Standards,* vol. 56, no. 1, pp. 1-8, 1956.

[147] C. T. Crowe, D. F. Elger, B. C. Williams and J. A. Roberson, "Chapter 2 Viscosity," in *Engineering Fluid Mechanics*, 9th ed., Hoboken, NJ: John Wiley & Sons, Inc., 2009.

[148] D. R. Lide, Ed., CRC Handbook of Chemistry and Physics, 86th ed., Boca Raton, FL: Taylor and Francis, 2006.

[149] A. Apelblat, "Effect of temperature on compressibility properties of 0.1, 0.5 and 1.0 molal solutions of alkali metal nitrates. Part 2. Aqueous solutions of lithium nitrate, sodium nitrate and potassium nitrate in the 287.15K to 353.15K temperature range," *Journal of Molecular Liquids,* vol. 242, pp. 1236-1247, 2017.

[150] "Wolfram Alpha," [Online]. Available: WolframAlpha.com.

[151] H. Corti, R. Crovetto and R. Fernandez-Prini, "Properties of the borate ion in dilute aqueous solutions," *Journal of the Chemical Society, Faraday Transactions I: Physical Chemistry in Condensed Phases,* vol. 76, pp. 2179-2186, 1980.