INVESTIGATING BIAS IN PROTEIN PROPERTIES INFERRED VIA ANCESTRAL SEQUENCE RECONSTRUCTION

by

ABRAHAM J. RICKETT

A THESIS

Presented to the Department of Chemistry & Biochemistry
and the Robert D. Clark Honors College
in partial fulfillment of the requirements for the degree of
Bachelor of Science

June 2017

An Abstract of the Thesis of

Abraham Rickett for the degree of Bachelor of Science
in the Department of Chemistry & Biochemistry to be taken June 2017

Title: Investigating bias in protein properties inferred via ancestral sequence reconstruction.

Approved: ____

Michael J. Harms

Ancestral Sequence Reconstruction (ASR) is a powerful technique used by researchers to study ancient proteins and their evolution. It is, however, an approximation based on incomplete information and simplifying assumptions about the evolutionary process. It is therefore important to understand and control sources of error in ASR studies.

One possible source of error is electrostatics. We formulate and analyze the electrostatics of 14 distinct protein families, each containing PDB structures of reconstructed ancestral proteins and their modern descendants.

We observe electrostatic abnormalities in some ancestral families, however these abnormalities are not statistically significant in the context of the entire sample. Simulated evolution analyses suggest that reconstruction can generate sequences with less extreme electrostatic character compared to the known original sequence. High charge optimality is not easily recoverable through posterior probability sampling. Existence of electrostatic bias is ultimately not disproven, and should be explored further with larger samples and more rigorous analytical methods.

Acknowledgements

Firstly, I would like to thank Dr. Michael Harms and the members of the Harms Lab at the University of Oregon, for helping me to fully examine this topic and consider the various perspectives and contexts related to this subject matter. I cannot thank them enough for their indispensable guidance and support throughout this strenuous process. Secondly, I thank Dr. Tim Williams and the faculty of the Robert D. Clark Honors College for their academic passion, as well as their dedication to furthering the success of undergraduate students. They have pushed me to excel both academically and personally, and I am extremely grateful. Finally, my sincerest thanks to my mother and father, my sisters, and my friends for their consistent emotional support throughout these past four years. I would not have made it through without them.

Table of Contents

List of Accompanying Materials

List of Figures

Introduction

General Background

Gaining a clearer understanding of the past is an integral part of any academic discipline, and biochemistry is no exception. In biochemistry, history is determined by the evolution of proteins and other biomolecules. Protein evolution involves changes which occur at the microscopic level. Analyzing the biological and physiological outcomes of these specific changes can give researchers information applicable to various other biological contexts, aiding in the development of highly specialized biological molecules for use in the medical field. In this way, our understanding of protein evolution is directly linked to our ability to recognize and efficiently fix biological problems which affect humans across the world.

Proteins are essential to the survival of any organism. They break down ingested toxins, help extract chemical energy from food, join together in the cell to form structural elements, send signals from cell to cell, and much more. Every protein sequence defines a unique 3-dimensional structure which a protein assumes through a process called folding. Differing sequences will encode different structures and functional properties, subsequently generating diversity in function.

Proteins are an extremely diverse group of molecules that consist of chains of smaller building-block molecules called amino acids. There are 20 different amino acids which can appear at any place in a protein sequence. Since every amino acid in a sequence can interact with one another, as well as other molecules present in the environment, the unique amino acid sequence of a protein directly encodes its structure and thus the function it will perform within a cell.

Proteins in different organisms that share a common ancestor accumulate different mutations and thus diverge in sequence over time. Evolutionary biochemists seek to understand this process. A key tool in their kit is ancestral sequence reconstruction.

Ancestral Sequence Reconstruction

Ancestral sequence reconstruction (ASR) is a technique used by biochemists to infer the sequence of an ancestral protein using known sequences of that protein's modern descendants (Yang, Kumar, and Nei 1995; Zuckercandl and Pauling 1965; Harms and Thornton 2010; Thornton 2004). Through ASR, researchers are essentially back-calculating the evolution of proteins using a specific series of statistical calculations. The resulting reconstructed sequence can provide important information about how sequence change correlates with altered function, and thus helps reveal how the sequence of a protein determines its physical structure and biological function. This process has been used to dissect medically important proteins (Eick et al. 2012; Lynch et al. 2008; Ortlund et al. 2007; Field et al. 2006; Bloom, Gong, and Baltimore 2010), make predictions about changes in viral RNA (Shapiro et al. 2006), and uncover the molecular basis and timeline of fluorescent protein color (Kelmanson and Matz 2003). ASR has even helped to unravel the mechanism of an important drug (Wilson et al. 2015). For these reasons, ASR is a very applicable and important technique with the potential to revolutionize the way we think about human disease, as well as proteins and their properties.

One area of interest is understanding the inaccuracies in ASR. The method is, at its core, an approximation. Previous studies have investigated possible problems in

ASR. These include inflicted bias (Williams et al. 2006), robustness to uncertain trees (Hanson-Smith, Kolaczkowski, and Thornton 2010), and random error (Hart et al. 2014; Eick et al. 2017).

Experimentation performed by Williams, Pollock, Blackburne, and Goldstein is perhaps the most comparable to the goals of this investigation. The authors use simulated evolution of protein structures and use maximum parsimony, maximum likelihood, and a Bayesian method to infer an ancestral sequence from the evolved proteins. They find that both maximum parsimony and maximum likelihood tend to select amino acids which overestimate the thermostability of the ancestral sequence (2006). These results support the idea that bias is induced on sequences reconstructed using ASR.

To determine whether the accuracy of a reconstructed ancestral sequence is improved by the inclusion of phylogenetic uncertainty in the reconstruction algorithm, Hanson-Smith, Kolaczkowski, and Thornton use simulated evolution to compare reconstructions calculated using the ML approach and the Bayesian method, which incorporates statistical uncertainty into its inferences. They find that the ML approach results in an accurate reconstruction, even when there exists uncertainty in the phylogenetic model, thus deeming the Bayesian method unnecessary and ineffective in improving reconstruction accuracy (2010).

The basis for the uncertainty of a reconstruction lies in the mechanistic limitations of ASR itself. Widely used phylogenetic models allow each site in a protein to evolve separately. This characteristic makes models of protein evolution tractable. Conversely, if the identity of a certain site is defined as dependent on all other sites in the sequence, the number of model parameters becomes extremely large. A model with a large number of parameters is much more difficult to use, as computations will be much more time consuming.

However, models based on site-independence are an approximation of reality. Amino acids within a protein do physically interact, and thus are known to co-evolve (Gloor et al. 2005; Hopf et al. 2015; Yip et al. 2008; Süel et al. 2003; Socolich et al. 2005).

Most reconstructed sequences are derived by assigning each amino acid according to the highest probability candidate within a phylogenetic model, a technique which provides the maximum likelihood (ML) sequence. Since these probabilities are calculated under the assumption of site-independence, the method itself could bias the sequences it predicts, and thus misrepresent the co-evolutionary properties of ancestral proteins. For this reason, making inferences about an ancestral sequence’s properties using ASR could be problematic.

Electrostatics as a source of bias

Interactions between charged amino acids are an important aspect of protein structure and function. However, electrostatics has not yet been investigated as a source of bias in ASR. Despite this, there is interest in the electrostatics of ancestral proteins. For instance, one investigation studied the net surface charge of marine mammal myoglobin as it relates to evolutionary change in diving capacity (Mirceta et al. 2013). A protein’s electrostatics depends on interactions ignored by ASR, and thus could cause bias in reconstructions, as well as inaccuracies in inferences made about the electrostatics of reconstructed sequences.

The electrostatics of proteins are clearly well tuned, as shown by several previous investigations. Wada and Nakamura provide a physics-oriented investigation on the nature of charge distribution in known protein sequences. While the investigation does not analyze properties of reconstructed proteins, it does reveal important trends which motivate an investigation of electrostatic bias in ancestral sequences. The authors provide an analysis of 14 proteins with 44,000 charge pairs and conclude that charged atoms are, on average, surrounded by charges of the opposite sign. They also find that charged atoms are evenly distributed across the surface of the studied proteins (Wada and Nakamura 1981). Some aspects of the computational analyses within our investigation will be similar to Wada and Nakamura's (namely their analyses of same-sign and opposite-sign charged neighbors), but reconstructed ancestral sequences will also be included, and directly compared to modern sequences.

Previous investigators have proposed that electrostatics is a source of possible bias when studying protein evolution. Haq, Andrec, Morozov, and Levy study the electrostatic effects of mutating charged amino acids in HIV protease, using a "coarse-grained energy model" in conjunction with a Potts model for statistical inference of charge states. The authors conclude that mutations of charged residues have a significant effect on protein stability, and "uncorrelated [electrostatic] mutations would strongly destabilize the enzyme" (2012). Haq and colleagues provide evidence suggesting that electrostatics is a crucial component in the co-evolution of protein mutations.

Due to the co-evolutionary nature of protein electrostatics, as well as the finely tuned electrostatic characteristics of known protein sequences, we were motivated to investigate whether ASR generates sequences with altered electrostatics.

Experimental questions

In our exploration of electrostatic bias in protein sequences inferred via ASR, we will attempt to answer the following questions:

- Are the electrostatics of ancestral proteins accurately inferred by evolutionary models which assume site-independence?
- If bias in electrostatics exists, where does it come from?
- Is there a way to better account for electrostatics in evolutionary models?

Results

Tools to quantify electrostatic optimality

We first set out to develop metrics to quantify the electrostatics of ancestral and modern proteins. At its most basic level, our model of electrostatic energy calculates interactions between pairs of charges. Using Coulomb's law as well as the Debye-Hückel theory, we developed a script which reads in any protein structure from the Protein Data Bank (PDB), calculates the energy of every possible interaction between two charged atoms within the structure, and sums all of these energy values, outputting a total Coulomb energy for the structure's charge distribution.

Additionally, we designed a script which outputs an "optimality score" for given structure's charge distribution. This value is calculated by randomly shuffling the signs of all charged atoms in the structure and recalculating the total Coulomb energy. Thousands of these "shuffled energies" are recorded, and a z-score is generated to relate the mean of the shuffled energies to the original structure's energy. A positive optimality score signifies a more optimal charge distribution than expected by chance (see Figure 1, Methods, and S2 for more information).

Some reconstructed proteins have altered electrostatics

The first stage of our investigation of electrostatic bias in ASR involved an analysis of 35 reconstructed ancestral protein crystal structures and 66 modern sequences matched the ancestral sequences via Protein BLAST. We searched the public "Protein Data Bank" database for reconstructed ancestral proteins that had published 3D structures. We found 35 such structures. We then searched for structures of modern members of these ancestral proteins using BLAST. Related modern proteins were sourced from a variety of organisms when possible, and all structures were sorted into 14 unique families according to function and relatedness (see S1).

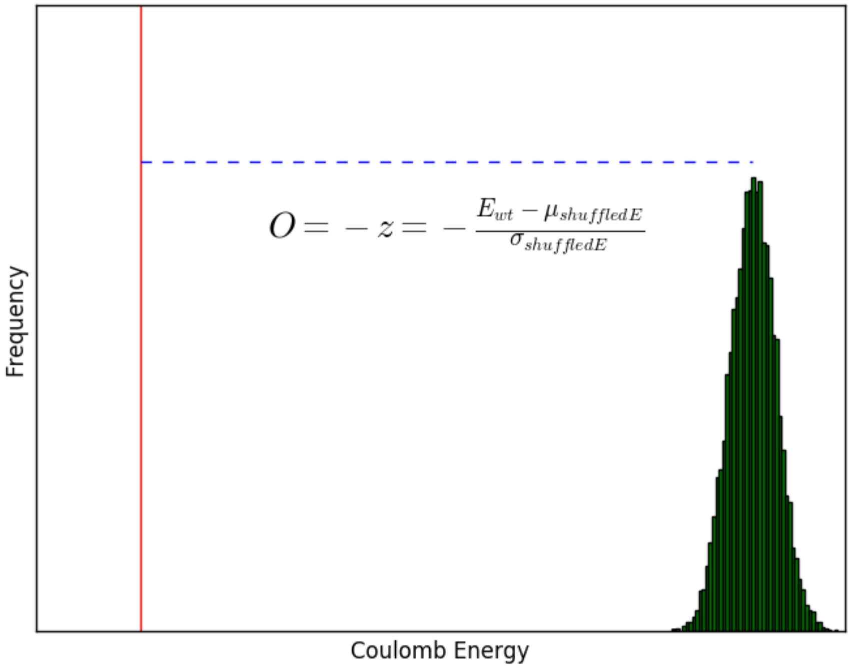


Figure 1. Visual example calculation of an electrostatic optimality score (O).

Plot shows a histogram of Coulomb energies calculated for 10,000 shuffled charge distributions of a single protein (green bars). The Coulomb energy of the original charge distribution is indicated with the red line. By calculating the mean and standard deviation of the distribution, we can calculate a z-score for the Coulomb energy of the original charge distribution. The z-score is multiplied by -1 to calculate the optimality score.

We performed several analyses to characterize the electrostatic characteristics of our selected ancestral and modern protein files. We first measured charge optimality. Our shuffled charge analysis calculates the energy of both the original protein structure and 10,000 structures with randomized charge distribution, and outputs a z-score for the original structure which we call an “optimality score”. A large, positive optimality score indicates that the distribution of charges in the protein is much better than expected by a random distribution of charges on the surface. The calculation of an optimality score is visualized in Figure 1.

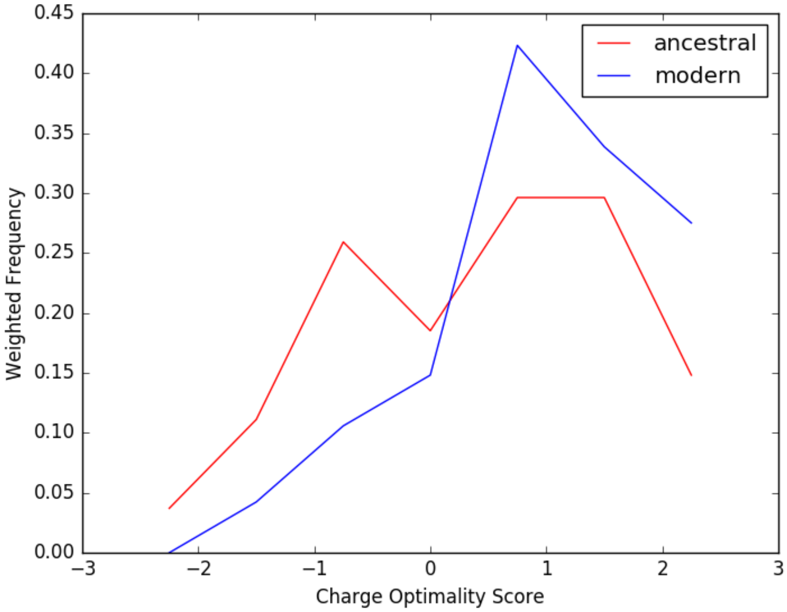


Figure 2. General distribution of modern and ancestral optimality scores.

Plot shows optimality scores which were obtained from a sample of 10,000 shuffled Coulomb energy values for each sequence. We collected data from 35 ancestral (red) and 66 modern (blue) sequences. Histogram frequencies have been normalized. Ion pairs excluded from shuffling (see Methods).

For a structure with a more optimal charge distribution, we would expect the shuffling procedure to (on average) destabilize the protein, indicated by an increase in the shuffled Coulomb energy and a higher, more positive optimality score.

We then performed this analysis on all ancestral and modern proteins in our dataset. Figure 2 displays two normalized distributions of optimality scores for ancestral and modern structures across 14 protein families. A peak on the negative end of the ancestral distribution indicates a group of ancestral structures with non-optimal charge distribution.

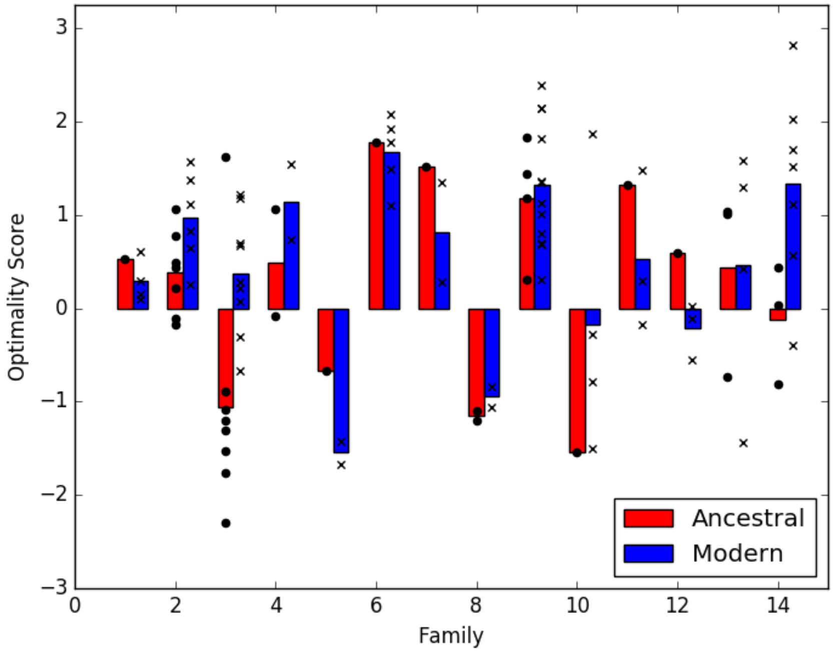


Figure 3. Summarized optimality data for 14 protein families.

Plot shows individual optimality scores which are plotted as dots (ancestral) and crosses (modern), in addition to colored bars indicating mean optimalities for both modern (blue) and ancestral (red) family datasets. Values were obtained from a sample of 10,000 shuffled Coulomb energy values for each sequence. Ion pairs excluded from shuffling.

We next broke out the analysis by family. Of the 14 families, 7 have ancestral sequences with lower average optimality scores (Family 2, 3, 4, 8, 9, 10, 14) 1 family showed about the same average optimality between ancestral and modern (Family 13), and 6 families had ancestral average optimalities higher than the average modern sequence (Family 1, 5, 6, 7, 11, 12).

Figure 3 summarizes the collected optimality data. The tendency for modern proteins to have more optimal charge distributions makes sense when we consider ASR’s intrinsic ignorance of electrostatic characteristics. Since a modern protein’s electrostatic character is highly regulated and extreme compared to the entire distribution of reconstruction candidate sequences, we would expect ASR to select sequences closer to the mean of that distribution, thus producing a sequence with less extreme charge character.

A closer look at the identities of the structures included in Figures 2 and 3 show that 47% of the non-optimal scores belong to reconstructed proteins from the thioredoxin family. Due to the small sample size, it is difficult to make statistical inferences on the data, but it is evident that some families show ancestral sequences with much less optimal charge distributions (Family 3, 14) and some with less extreme charge characteristics (Family 2). This variability in charge character may be partially due to differences in structural and functional properties between families.

From these data, we conclude that while not universally evident, electrostatic bias cannot be ruled out as a source of error in ASR. A wider range of ancestral crystal structures could give a more conclusive and representative result.

We next compared the distribution of charged neighbors for the modern and ancestral proteins. Our charged neighbor analysis iterates over every charged atom in a given structure and counts the neighboring charged atoms, generating average numbers of same-charge and opposite-charge neighbors for the whole structure. The results of the charged neighbor analysis are similarly inconclusive in their comparison of ancestral and modern structures.

Figure 4 shows a set of normalized values to create directly comparable proportions of same versus opposite charge neighbors in the thioredoxin family of proteins. In general, both modern and ancestral structures tend to have a higher number of opposite-charge neighbors than same-charge neighbors, which aligns with experimentally observable characteristics of proteins to have more favorable, non-repulsive charge interactions on their surface (Wada and Nakamura 1981).

Some ancestral structures, such as the thioredoxin 2YOI.pdb (A4 in Figure 4), have an increased average of same-charge neighbors, which could define their charge distribution as being "non-optimal". However, these differences are not widespread throughout the entire dataset, and thus it is difficult construct a general statement about the prevalence of electrostatic bias in ASR.

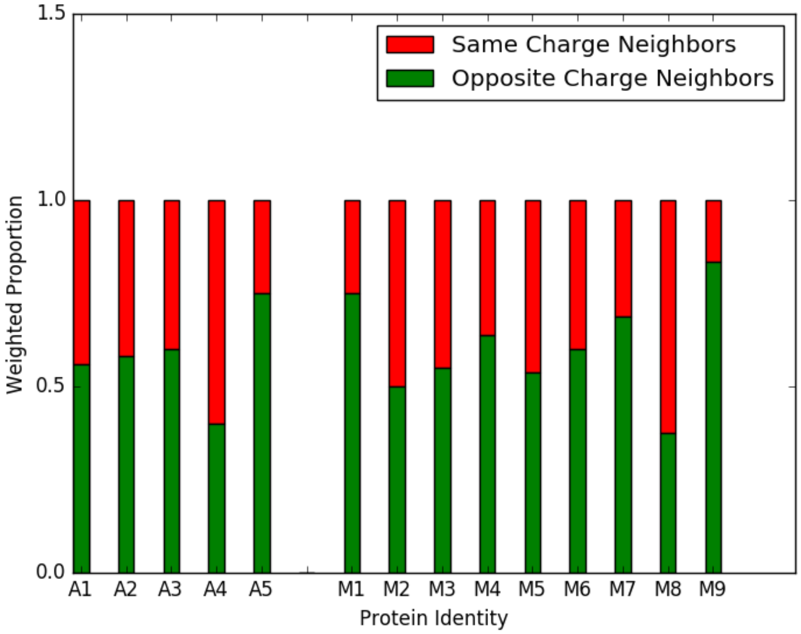


Figure 4. Charged neighbor data for ancestral and modern thioredoxin proteins.

Plot shows normalized proportions of same-charge (red bars) and opposite-charge (green bars) neighbors within 6 Angstroms of all charged atoms within a given protein structure (M=modern, A=ancestral).

Evolutionary simulations further characterize electrostatic bias of ancestral sequence reconstruction

We next sought to investigate whether ignoring electrostatics led to bias in simulated ancestral reconstruction studies. A series of scripts allows a known starting sequence to be evolved along randomly generated trees. Following evolution, the original sequence is inferred using ancestral sequence reconstruction. The reconstructed sequence's electrostatic character is then compared to that of the actual starting sequence. By doing this for many, many trees, we can determine whether there are trends in the electrostatics of reconstructed sequences.

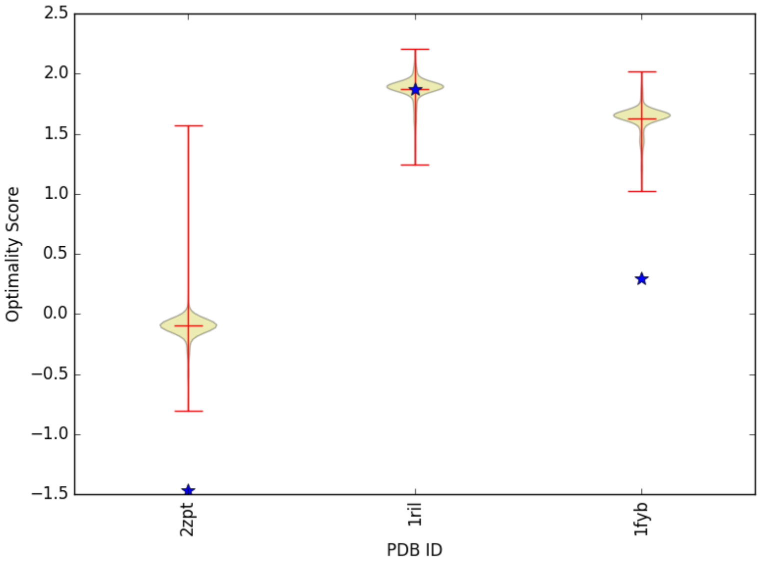


Figure 5. Optimality scores of reconstructions using no electrostatic constraint on simulated evolution.

Plot shows distributions of reconstructed optimality scores (yellow) for three modern PDB files. These specific distributions exhibit the highest drift away from an optimality score of zero, the closest drift towards zero, and the smallest drift overall compared to the optimality score of the original sequence (blue stars). Each distribution contains over 300 distinct values.

Figure 5 is a violin plot representing the distributions of optimality scores collected from reconstructions. The blue stars mark the optimality scores of the known original sequences. The three PDB files displayed in Figure 5 were chosen to illustrate the range of the data collected. Some reconstructions yielded much more extreme charge distributions than the wild-type sequences (1FYB.pdb), others became less extreme and drifted toward a neutral optimality score (2ZPT.pdb), and still others generated reconstructions with comparable optimality scores to the original sequence (1RIL.pdb). The range of the reconstructed optimality scores varied as well. Data for 2ZPT.pdb indicates far-removed outliers on the more optimal end of the distribution, while data for 1FYB.pdb and 1RIL.pdb show less extreme outliers.

To ask if there was a global trend, we collected similar data from all modern proteins in the dataset. Figure 6 shows a single distribution of distances from the mean of each reconstructed optimality distribution to the optimality score of the original sequence.

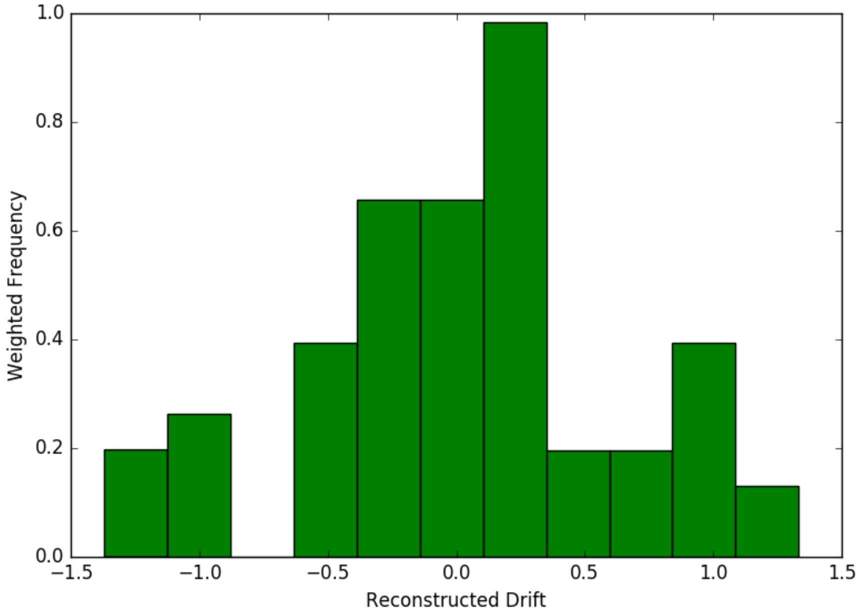


Figure 6. Distribution of reconstructed optimality drift for 65 modern proteins.

Plot shows a distribution of reconstructed drift values which were obtained by subtracting absolute value of wild type optimality from absolute value of mean reconstructed optimality. A negative value indicates a drift toward zero after reconstruction. Reconstructed optimality distributions contain over 300 distinct values.

We observe the distribution to be centered around zero, however, there are a large amount of sequences whose reconstructed optimalities deviated from that of the wild-type sequence. Again, evidence for electrostatic bias is not disproven, but its effects are not widespread according to this sample of modern PDB structures.

Sampling maximum likelihood distribution to detect electrostatic bias

To explore the relationship between charge distribution optimality and raw reconstruction data which generates non-optimal ancestral sequences, we sampled the posterior probabilities of reconstructed sequences from the previously outlined simulations.

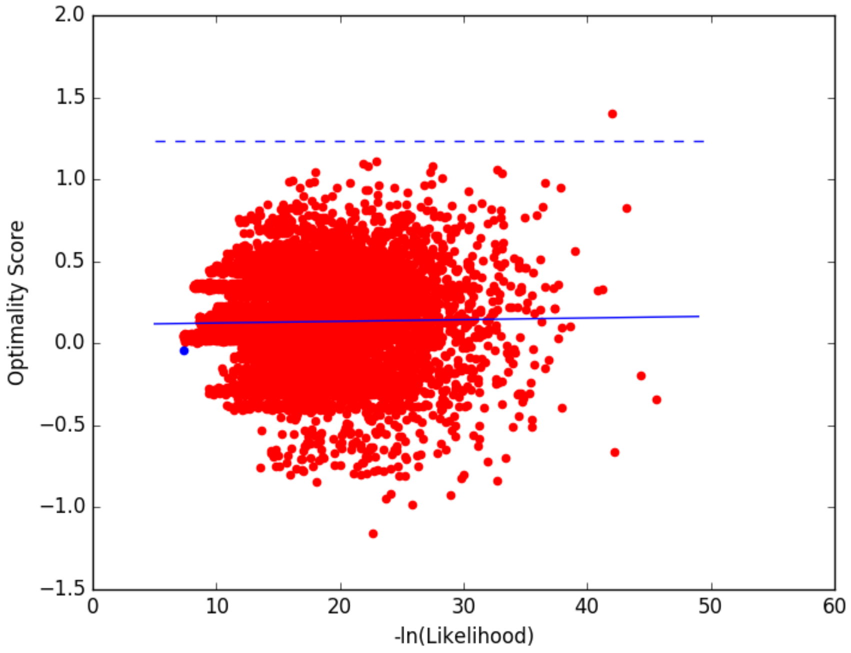


Figure 7. Likelihood of sequence vs. optimality score sampled from posterior probability values.

Plot shows optimality and likelihood of over 10,000 sampled sequences (red dots) from the posterior probabilities of a reconstruction of 2O7K.pdb. The maximum likelihood (ML) sequence is plotted in blue. Wild-type optimality indicated by dashed blue line. Linear fit indicated by solid blue line (R = 0.0204).

Every reconstruction generates a matrix of posterior probabilities, containing the numerical probability of any given amino acid to be at each site in the protein, as determined by the phylogenetic model used. After sampling these probabilities, we then mapped the charge distribution of the sampled sequences onto the original structure and generated optimality scores for every sample.

Figure 7 plots the likelihood of 10,000 samples against their optimality scores, with the maximum likelihood (ML) reconstructed sequence plotted in blue and a dashed line indicating the z-score of the original structure. Here, the normalization effect of the reconstruction process evident, due to very few samples with comparable optimality scores to the original structure. Another notable result is the fact that the likelihood of the sampled sequence does not predict its optimality score. This suggests there may be challenges involved in reducing the effects of electrostatic bias for a given sequence. However, there are some samples which have both high likelihood relative to the ML sequence and improved charge distribution optimality.

We then considered that a sample sequence's Hamming distance (number of sites difference from original sequence) might correlate differently with likelihood. Figure 8 plots the Hamming distance of the same 10,000 samples against their respective likelihoods.

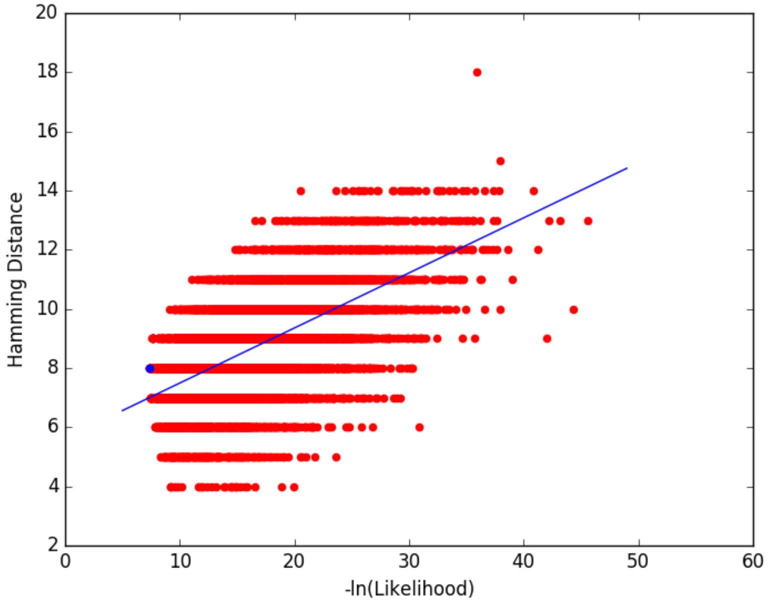


Figure 8. Likelihood of sequence vs. Hamming distance sampled from posterior probability values.

Plot shows Hamming distance and likelihood of over 10,000 sampled sequences (red dots) from the posterior probabilities of a reconstruction of 2O7K.pdb. The maximum likelihood (ML) sequence is plotted in blue. Linear fit indicated by solid blue line (R = 0.5891).

We observe that in general, an increase in Hamming distance corresponds with a decrease in likelihood, displayed by the positive slope of the linear fit equation. This result indicates that a reconstruction candidate could be selected due to a high likelihood and/or a low Hamming distance, but that does not tell us anything about the sequence’s electrostatic character. A reconstructed sequence could be biased (see Figure 3, Family 3) or not biased (see Figure 3, Family 9), but the sequence’s likelihood value cannot provide this information.

Discussion

Implications

The results of this investigation suggest that while electrostatic bias does not affect the reconstructions of all proteins, it could pose a problem for certain types of sequences. The abnormal electrostatic characteristics of specific proteins suggest that the effects of electrostatic bias in ancestral sequence reconstruction could be more significant in certain families of proteins. Our shuffled-charge analysis indicated that several families of proteins contained reconstructed sequences less optimal charge distributions than their modern descendants (Figure 3). The properties of structures with more radical charge distribution could be analyzed further in vitro, using metrics like thermodynamic stability. If a sequence is found to have a low electrostatic optimality compared to a modern descendant, we would expect the structure to be less stable thermodynamically.

Overall, our findings do not provide significant evidence for the existence of widespread electrostatic bias in ASR. However, abnormal findings for specific sequences suggest that this form of bias should not be disregarded. Electrostatic energy is a quantitative trait, and one that is essential to a protein’s stability and function. The existence of this sort of bias in ASR suggests that the method is more effective at reconstructing qualitative traits (like binding behavior) than quantitative properties. Thus, the intricacies of protein electrostatics can become lost in the process of reconstruction, a problem which is especially evident among certain proteins.

This notion is important to keep in mind when reconstructing proteins with especially sensitive electrostatic character, such as the thioredoxins (Family 3) and the oxidoreductases (Family 14) displayed in Figures 2 and 3.

Solutions and future directions

As mentioned previously, this investigation found significant evidence of bias only in certain groups of sequences. It would therefore be beneficial to identify what types of sequences are more susceptible to electrostatic bias and explore methods by which to reduce the bias, thus generating more accurate reconstructions.

The first step in achieving this would be to develop alternative analysis tools to electrostatically characterize specific sequences. Our model for the shuffled-charge analysis only manipulated the charges of atoms which have a non-zero charge in the wild-type structure, and did not exclude charges which may have been present in the inner parts of the protein. Perhaps a more accurate and comprehensive analysis would allow for the charge-shuffling of any site on the surface of the protein, and would disallow the shuffling of any charges beneath the surface of the structure. This shuffling method could be applied to both wild-type optimality calculations as well as optimality calculations for reconstructed and simulated sequences as well.

With multiple ways to represent and quantify the electrostatics of different proteins, we can begin to understand what features make certain types of protein sequences (such as the thioredoxin and oxidoreductase families indicated previously) more susceptible to reconstructive bias, thus allowing us to predict when electrostatic considerations are most important in performing an accurate reconstruction.

Another limitation of this investigation was the approximation of the position of charges on evolved sites. Creating scripts to adaptively map charge positions would add more accuracy to energy calculations of evolved sequences. Finally, to make more statistically-backed conclusions, it would be necessary to sample a larger group of reconstructed protein structures. The varied electrostatic character of the structures sampled in this investigation may have simply been due to a small sample size and limited quantity of solved ancestral structures accessible from the Protein Data Bank. Collecting a wider range of sequences would allow for more direct statistical analysis, and offer more insight into the full extent of ASR's bias.

After gaining a better understanding of the nature and prevalence of bias in ASR, methods could be explored to correct specific aspects of the bias in reconstructed sequences. One option which could help researchers find more electrostatically optimal would be to use a Bayesian method for reconstruction, sampling a wide variety of candidates and testing their electrostatic properties. However, as shown in Figure 8, reconstruction likelihood does not necessarily predict the charge distribution optimality of the candidate, so (while worth exploring) the utility of this method could be limited.

Another option would be to reformulate the entire process of reconstruction, implementing structure-aware parameters to the reconstruction model. This solution seems more logical, but would be much more difficult in practice. Assumptions would have to be made about the spatial arrangement of sites in an ancestral structure, and certain aspects of the calculations would need to adjust depending on the type of sequence under study.

In summary, the existence of bias inherent in ancestral sequence reconstruction is observable in some cases, and should be explored further from novel and varied perspectives. While ancestral sequence reconstruction is not perfect, an increased awareness of its limitations is essential, and will lead us to create more accurate methods by which to explore the intricacies of protein evolution.

Materials and Methods

Our investigation of bias in reconstructed protein sequences relies on a series of distinct computational analyses. These analyses were conducted to quantify the severity and ubiquity of electrostatic bias which exists in previously reconstructed sequences, to determine whether the bias can be systematically and/or reliably reproduced, and to analyze the relationship between a sequence's reconstructive likelihood and its electrostatic properties.

Protein families

We searched the Protein Data Bank (PDB) to compile a group of 3D structures of reconstructed ancestral sequences ("RCSB Protein Data Bank - RCSB PDB" 2016). After recording descriptions of each ancestral sequence, we used the Protein BLAST (Basic Local Alignment Search Tool) to identify a pool of modern descendant structures for each ancestral reconstruction (Altschul et al. 1990). Selected modern structures were sourced from a variety of different organisms when possible, and then consolidated into 14 protein families containing ancestral and modern structures.

Using the "pdb-tools" Python scripts, we isolated the specific chain subsets within each protein that were matched by the BLAST from the rest of the structure and placed them in a separate directory for computational analysis (Harms 2016). Analyzing these matched chains allowed for a more accurate and direct comparison between modern and ancestral sequences.

Shuffled-charge analysis

Several scripts were written in the Python programming language to analyze the electrostatic properties of the collected set of PDB files. The starting point was a program which simply reads a PDB file and calculates its Coulomb energy in kcal/mol, incorporating the Debye-Hückel theory of electrolytes, as shown below:

$$E_C = \frac{332\ q_1 q_2}{r_{1,2}\varepsilon \cdot e^{-50.29\sqrt{I/(\varepsilon \cdot T)}}}$$

where q_1 and q_2 are the signs of two charged atoms, $r_{1,2}$ is the distance between the two atoms in Angstroms, ε is the dielectric constant, I is the ionic strength in molar, and T is the temperature in K. A default environment was defined for all calculations with a dielectric constant of 20.0, an ionic strength of 0.1 M, and a temperature of 300K. The total energy of a structure was calculated by performing the above calculation between every unique pair of charged atoms in the structure, and summing the results.

After calculating the Coulomb energy for each isolated chain, we developed a script to compare the optimality of charge distributions. The updated script was written to shuffle the charge signs on a PDB file a defined number of times, calculating a new total energy for each shuffle. The script then takes the distribution of "shuffled" energies and calculates a z-score using the wild-type energy collected earlier. Multiplying this z-score by -1 provided our "electrostatic optimality score" (see Figure 1 and S2). A negative value indicates a non-optimal charge distribution, and a positive value indicates an optimal distribution.

Additionally, charges involved in ion pairs (defined as charge pairs yielding -3.5 kcal/mol or less) were excluded from shuffling. Changing charge signs involved in ion pairs can result in a large change in the total energy. The magnitude of this change would be more significant for structures with more ion pairs, thus skewing certain optimality scores to be more negative. Optimality scores for all ancestral and modern sequences were recorded using a distribution of 10,000 unique shuffles.

Charge neighbor analysis

As a complement to the charge distribution optimality data, we developed an independent analysis of the identities of charged neighbors of charged atoms within each protein structure. A new Python script was written to calculate the average number of same-charge neighbors and opposite-charge neighbors within 6 Angstroms for each PDB structure. Key ancestral sequences with significantly different optimality scores and/or average neighbor identities were identified, along with any trends found among protein families.

Simulated evolution

To determine whether electrostatic bias could be reproduced, we ran evolutionary simulations with customized electrostatic constraints. PyVolve was used as a tool to evolve individual sequences along a given phylogenetic tree (Spielman and Wilke 2015). The program's evolver was edited to monitor the difference in Coulomb energy between the known starting sequence and the evolved candidate, using a percent-difference threshold. At each node in a randomized phylogenetic tree, the modified evolver checks the energy of all evolved sequences and verifies that the percent change in energy from the original sequences is below the set threshold. We define a lower threshold value as more constrained, since the percent change accepted is lower. Trials were run at a variety of thresholds (0.05, 0.1, 0.2, 0.4, 0.5, 0.6, 0.8, 1.0, and 100.0) to observe the differences in results as the threshold became more narrow, further manipulating the evolution process.

To allow for more consistent charge distribution calculations as random mutagenesis occurs, we coded all charges to be mapped to the β-carbon atoms of each residue's sidechain. Differences in energy calculations between the β-carbon mapping and wild-type mapping were compared beforehand and were shown to be statistically negligible (see S3). A single round of analysis consisted of evolution, reconstruction, and shuffled z-score comparison. The evolved sequences generated by the PyVolve evolver were used to "back-calculate" the starting sequence via ASR. We used the LG phylogeny model to perform both the evolution and reconstruction for each trial. Following reconstruction, the optimality score of both the known starting sequence and the reconstructed sequence were compared. Every collected modern chain sequence was used as a starting sequence for this analysis. Each round of simulations incorporated between 30 and 60 randomized Newick trees, performing 10 replicates for each tree. This resulted in at least 300 evolutions, reconstructions, and z-scores for each modern sequence.

Electrostatic characterization and sampling of reconstruction candidates

Finally, we explored methods by which to visualize the relationship between electrostatic properties and the process of ancestral reconstruction. The posterior probabilities of an ancestral reconstruction from the simulations described above were utilized as the input to a specialized Python script. These posterior probabilities were sourced from a trial done with no electrostatic constraint on evolution. By randomly sampling from the probabilities, we generated over 10,000 reconstruction candidates, ultimately providing a distribution of likelihood (calculated by multiplying together the sampled probability for every site in the sequence), versus charge distribution optimality (calculated using a modified version of the shuffled charge analysis). These data were graphed along with the same data for the reconstruction’s maximum likelihood sequence and the wild-type optimality score. We also constructed charts plotting likelihood against each sample’s hamming distance. A linear fit was calculated for each plot to summarize the trend and relationship between these variables.

Glossary

amino acid: an organic molecule containing both an amine ($-NH_3^+$) and carboxylic acid ($-COO^-$) functional group, as well as a variable sidechain. Multiple amino acids join together by forming peptide bonds between amine and carboxylic acid groups, thus producing an amino acid chain, or polypeptide.

β-carbon: The first carbon which makes up an amino acid’s sidechain. All 20 biological amino acids found in proteins have a β-carbon except glycine, which has a single hydrogen atom as its sidechain.

Coulomb energy: The energy of an interaction or series of interactions between charged atoms, expressed in units of energy (Joules or calories, for example).

crystal structure: A 3-dimensional structure of a protein determined using a technique called X-ray crystallography. A purified protein is put into solution so that the molecules align and crystallize, at which point the crystal can be exposed to a beam of X-rays and characterized using an algorithm which analyzes the specific diffraction of the X-rays. The majority of the structures in the Protein Data Bank were solved using this technique.

electrostatics: The study of static electrical charges (as opposed to moving electric currents), like those present on the surface of a protein.

in vitro: In vitro studies involve analyses of biological matter (cells, molecules) outside of a normal biological environment.

phylogenetic model: An evolutionary “tree” which displays the inferred relationships between a group of species and their common ancestor.

Accompanying Materials

S1. Table of Ancestral protein families.

ANCESTRAL PROTEINS	MODERN PROTEINS
FAMILY 1 – Protease/Hydrolase Inhibitors	
1CE3.pdb	1OYV.pdb
	4SGB.pdb
	1FYB.pdb
	2JZM.pdb
FAMILY 2 – Corticoid Receptors	
2Q3Y.pdb	2AA6.pdb
3GN8.pdb	3VHU.pdb
3RY9.pdb	2AA2. pdb
4E2J.pdb	3MNE.pdb
	1M2Z.pdb
	3MNO.pdb
FAMILY 3 - Thioredoxins	
2YJ7.pdb	2FCH.pdb
2YN1.pdb	2O7K.pdb
2YNX.pdb	2TRX.pdb
3ZIV.pdb	3HHV.pdb
2YOI.pdb	4TN8.pdb
	2E0Q.pdb

ANCESTRAL PROTEINS	**MODERN PROTEINS**
	1SYR.pdb
	2FA4.pdb
	1QUW.pdb
FAMILY 4 - Congerins	
3AJY.pdb	1C1F.pdb
3AK0.pdb	1WLD.pdb
FAMILY 5 - Sulfotransferases	
3QVU.pdb	1LS6.pdb
	2ZPT.pdb
FAMILY 6 - Transferases	
3VVT.pdb	2CWK.pdb
	1WKJ.pdb
	2ZUA.pdb
	2VU5.pdb
	2AZ1.pdb
FAMILY 7 - Hydrolases	
3ZDJ.pdb	3N4I.pdb
	4IBX.pdb
FAMILY 8 - Hydrolases	
4C6Y.pdb	3W4Q.pdb
	1YLP.pdb
FAMILY 9 – Luminescent/Fluorescent Proteins	

ANCESTRAL PROTEINS	**MODERN PROTEINS**
4DXM.pdb	4JC2.pdb
4DXN.pdb	3ADF.pdb
	2Z6X.pdb
	2VZX.pdb
	4IZN.pdb
	3S05.pdb
	4HQ8.pdb
	4HQ8.pdb
	1ZUX.pdb
	2GW3.pdb
	1XSS.pdb
	3LS3.pdb
	3ADF.pdb
	2OTB.pdb
FAMILY 10 - Hydrolases	
4LY7.pdb	1RBR.pdb
	3AA4.pdb
	2E4L.pdb
	1RIL.pdb
FAMILY 11 – Transport Proteins	
4M1V.pdb	3W9V.pdb

ANCESTRAL PROTEINS	**MODERN PROTEINS**
	2V3Q.pdb
	2Q9T.pdb
FAMILY 12 – Transcription/DNA Binding	
4OLN.pdb	1HCQ.pdb
	4AA6.pdb
	1R4I.pdb
FAMILY 13 – Transcription/Unknown Function	
4P3K.pdb	4P82.pdb
	1NON.pdb
	1UFR.pdb
	1W30.pdb
FAMILY 14 - Oxidoreductases	
4PLF.pdb	3CZM.pdb
4PLW.pdb	1OC4.pdb
	1CEQ.pdb
	3GVH.pdb
	4ROR.pdb
	2HJR.pdb
	3P7M.pdb

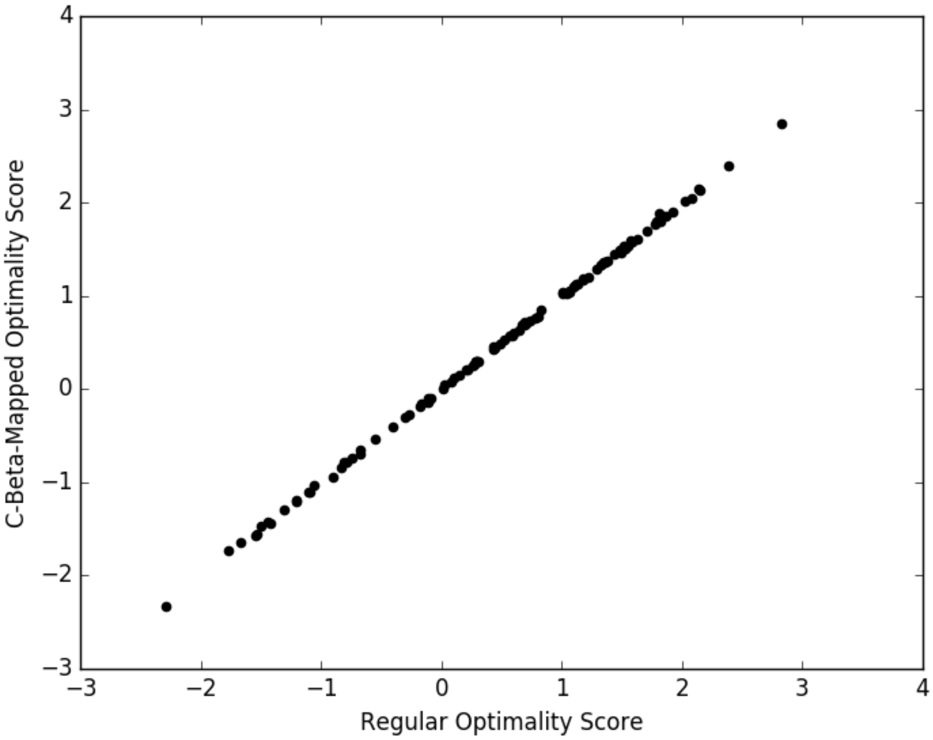
S2. Coulomb shuffling script.

```
#!/usr/bin/env python

"""
calcCoulomb3.py
A script to analyze the electrostatic coulomb energy (kcal/mol) of a protein in a PDB file.

Can provide the following values:
        - a protein's wild-type energy
        - energies of all charge-charge interactions
        - energy of a protein with its charge arrangement shuffled (excludes charges
        involved in ion pairs)
        - the z-score of a WT protein given a user-specified number of the shuffled proteins
        described above

User inputs (these are the parameters the user may change):
        PDB_FILE: pdb file containing structure
        NUM_REPS: number of charge shufflings to calculate
        USE_CB: True: bring charges down to position of beta carbons
             False: use actual positions of charged atoms
        DIELEC_CONST: dielectric constant of the system
        IONIC_STR: ionic strength of the system in molar
        TEMP_K: temperature of the system in Kelvin
"""

# ************************************************************** #
#                    User inputs                    # #
************************************************************** #

from sys import argv

PDB_FILE = argv[1]

try:
    NUM_REPS = int(argv[2])
except IndexError:
    NUM_REPS = 10000

try:
    USE_CB = argv[3]
except IndexError:
    USE_CB = False

import core

# ************************************************************** #   #                    Function
definitions                    #   # ************************************************************** #

from math import sqrt
from random import shuffle
import numpy as np

def calcPotential(coord,charge):
    """
    Calculates the energy of a structure given the coordinates of each charged atom,
    their fractional charge, and the dielectric constant of the system.

    E = 332*sum_i[sum_j[q_i*q_j/(r_ij * dielec_const)]] (kcal/mol)
    """

    # Initialize variables
    num_groups = len(coord)
    energy = 0.
    shuffle_pool = range(num_groups)
    potential = [[0.0 for j in range(num_groups)] for i in range(num_groups)]

    # Calculate energy of interaction of every ij interaction
    # (making sure not to double count; note we start j at i+1).
    for i in range(num_groups):
        for j in range(i+1,num_groups):
```

```
                                     # Calculate distance between atom i and atom j
                        r = (coord[i][0] - coord[j][0])**2
                        r = r + (coord[i][1] - coord[j][1])**2
                        r = r + (coord[i][2] - coord[j][2])**2
                        r = sqrt(r)
                        temp_energy = core.calcCoulomb(charge[i],charge[j],r)

                        # Print all individual interaction energies
                        #print [i],[j],temp_energy

                                      # Add the energy of each interaction to the total # energy
                                      energy = temp_energy + energy

                                      # Remove groups involved in ion pair interactions # from the
                                     shuffle_pool
                        if temp_energy < -3.5:
                            try:
                                shuffle_pool.remove(i)
                            except ValueError:
                                pass
                            try:
                                shuffle_pool.remove(j)
                            except ValueError:
                                pass

                                     # Set up matrix of potentials
                                     potential[i][j] = core.calcCoulombPotential(r)
                                     potential[j][i] = potential[i][j]

            # Create variable for the protein's wild-type energy
            wtenergy = energy

            # Return energy
            return potential, shuffle_pool, wtenergy

def calcShuffledE(shuffle_pool,coord,potential):
```

```
"""
Builds a matrix of shuffled charges based on shuffle_pool. Multiplies each charge pair to its
assocaited potential. Calculates final shuffled energy via summation of matrix items
"""

        # Initialize variables and lists
        num_groups = len(coord)
        pdb_q = charge[:]
        new_q = pdb_q[:]
        shuf2 = shuffle_pool[:]
        finalE_list = []

        # Shuffle appropriate charges for user-defined number of reps
        for x in xrange(NUM_REPS):
                shuffle(shuf2)

                # Align and replace original charges with shuffled
                # charges in new_q
            for i in range(len(shuffle_pool)):
                new_q[shuf2[i]] = pdb_q[shuffle_pool[i]]

            # Initalize a variable for total shuffled energy
                finalE = 0.




                # Multiply charge pairs by potential and add up total
                # energy
            for i in range(num_groups):
                for j in range(i+1,num_groups):
                  finalE += new_q[i]*new_q[j]*potential[i][j]

             # Add to list of shuffled energies
             finalE_list.append(finalE)
        return finalE_list

def calcZscore(wtenergy,finalE_list):
        """
        Calculates the z-score for the wild-type energy, given the wild-type energy and a list
        of energies with shuffled charge arrangements
        """

        return (wtenergy-np.mean(finalE_list))/np.std(finalE_list)

#********************************************************************** # #                    Main code
# # ********************************************************************* #

# Read in coordinates
coord, pka, charge, residue = core.readPDB(PDB_FILE, use_cb=False)
potential, shuffle_pool, wtenergy = calcPotential(coord,charge)

# if you're using cb, re-read the potentials and coord, but *keep* # the old shuffle pool so
we don't mess with ion pairs
if USE_CB == True:
        coord, pka, charge, residue = core.readPDB(PDB_FILE, use_cb=True)
        potential, shuffle_pool_junk, wtenergy = calcPotential(coord,charge)

finalE_list = calcShuffledE(shuffle_pool,coord,potential)
zscore = calcZscore(wtenergy,finalE_list)
```

S3. C-beta mapping vs. normal mapping optimality scores.



Each point on this plot represents one PDB structure. We calculate its optimality score normally (x-axis) and with charges mapped to their respective C-beta atoms (y-axis). The plot shows almost no change in optimality after mapping charges to C-beta atoms. Thus, our approximation of charge location is acceptable for this type of calculation.

S4. Simulated evolution and reconstruction data for modern sequences.

PDB ID	Original Optimality	Mean Reconstructed Optimality	Standard Deviation	Drift Distance
1c1f.pdb	0.735855339	1.65085313323	0.141194521557	0.914997794228
1ceq.pdb	2.036642405	1.77203274388	0.151604419619	-0.264609661116
1fyb.pdb	0.296905812	1.62926526697	0.118948197865	1.33235945497
1ls6.pdb	-1.690689768	-0.397344237785	0.241106393557	-1.29334553022
1m2z.pdb	1.599905666	0.630077826241	0.170030890803	-0.969827839759
1non.pdb	0.476632452	0.146851471429	0.143283627948	-0.329780980571
1oc4.pdb	2.906685646	2.4783420309	0.131921628063	-0.428343615103

PDB ID	Original Optimality	Mean Reconstructed Optimality	Standard Deviation	Drift Distance
1oyv.pdb	0.095075546	0.386607496585	0.198684197248	0.291531950585
1quw.pdb	0.19789134	0.0260401644316	0.194180720046	-0.171851175568
1r4i.pdb	-0.152930202	0.380494221255	0.130758077809	0.227564019255
1rbr.pdb	-0.880080358	-0.481123844675	0.177475360936	-0.398956513325
1ril.pdb	1.870247948	1.87181784951	0.114435998109	0.00156990151396
1syr.pdb	0.065016709	0.91552561878	0.145538758379	0.85050890978
1vs1.pdb	1.513404104	1.21222817091	0.125435952038	-0.301175933093
1w30.pdb	1.315208565	1.55967538605	0.137511760832	0.244466821053
1wkj.pdb	1.87561024	1.52154790978	0.107160274415	-0.35406233022
1wld.pdb	1.490290881	1.5814063823	0.113753369893	0.0911155013036
1xss.pdb	1.84699462	2.03562041423	0.106049306441	0.188625794234
1ylp.pdb	-0.993099804	1.90866109131	0.211339884063	0.915561287306
2aa2.pdb	0.835872262	0.81978749485	0.146279233575	-0.01608476715
2aa6.pdb	1.051158786	0.637150005552	0.174175164995	-0.414008780448
2az1.pdb	1.473945038	1.58063928325	0.132132557824	0.106694245254
2cwk.pdb	1.906394008	2.04529038308	0.117140283683	0.138896375076
2e0q.pdb	1.199922841	1.52700979241	0.112603046271	0.327086951412
2e4l.pdb	-1.422292153	-0.818391444014	0.312028289069	-0.603900708986
2fa4.pdb	-0.227022714	0.433527843452	0.180278146838	0.206505129452
2fch.pdb	0.730382671	1.10302787869	0.13094621261	0.372645207694
2gw3.pdb	1.108388506	1.80628076899	0.161208032216	0.69789226299
2hjr.pdb	1.088667314	0.739664991113	0.217868372613	-0.349002322887
2jzm.pdb	0.728532229	0.607346328388	0.147290957107	-0.121185900612
2o7k.pdb	1.237772528	0.138600680962	0.183539885408	-1.09917184704
2otb.pdb	0.742261375	0.650186847793	0.142299227527	-0.0920745272067
2q9t.pdb	-0.123417392	1.08929136046	0.087924849633	0.965873968459
2trx.pdb	0.300425391	0.479069705728	0.173928593992	0.178644314728
2v3q.pdb	0.285846596	0.810789331801	0.196781334102	0.524942735801
2vu5.pdb	1.094634917	0.880236631358	0.198365262299	-0.214398285642
2vzx.pdb	0.693132726	1.24128701212	0.17352362988	0.548154286122
2z6x.pdb	1.289831407	2.18181298497	0.155299338253	0.891981577969
2zpt.pdb	-1.465352542	-0.0952872780953	0.182083877487	-1.3700652639
2zua.pdb	2.167975732	1.86798907566	0.139145462015	-0.299986656336
3aa4.pdb	-0.316298952	-0.0917707619837	0.220022580131	-0.224528190016

PDB ID	Original Optimality	Mean Reconstructed Optimality	Standard Deviation	Drift Distance
3adf.pdb	0.985075009	1.99171764178	0.10016439491	1.00664263278
3czm.pdb	1.500803133	1.5171763249	0.12751376056	0.0163731919019
3gn8.pdb	0.244981529	0.392068611964	0.178253027019	0.147087082964
3gvh.pdb	1.666370463	1.89686331561	0.14732403032	0.230492852606
3hhv.pdb	0.696426072	0.348080603693	0.148953591023	-0.348345468307
3ls3.pdb	0.271590806	0.607990595403	0.116168428048	0.336399789403
3mne.pdb	1.38535283	0.36075487474	0.210966337681	-1.02459795526
3mno.pdb	0.24509891	0.303931769418	0.187549709302	0.0588328594184
3n4i.pdb	1.383110086	1.54280640763	0.183456058943	0.159696321631
3s05.pdb	2.133307159	2.79534885149	0.0974061273478	0.662041692491
3vhu.pdb	0.630345819	0.860724024822	0.189632984159	0.230378205822
3w4q.pdb	-0.85928373	-0.771521863819	0.267536175055	-0.0877618661814
3w9v.pdb	1.466659501	0.210315146079	0.0524889498399	-1.25634435492
4grs.pdb	1.731164575	1.73673869661	0.112239912691	0.0055741216093
4hq8.pdb	1.381442944	-0.332337574814	0.197464618886	-1.04910536919
4ibx.pdb	0.286504612	1.45958057143	0.113184584847	1.17307595943
4izn.pdb	0.773655409	-0.313554703757	0.248014064243	-0.460100705243
4jc2.pdb	2.169223864	2.34720479836	0.129554567645	0.177980934357
4p82.pdb	-1.400098617	1.35348604167	0.139687117022	-0.0466125753259
4ror.pdb	0.512943613	1.28431725041	0.046088344454	0.771373637409
4sgb.pdb	0.649321382	-0.0652888589327	0.223486881329	-0.584032523067

Table shows data from modern protein sequences evolved and then reconstructed with no electrostatic constraint. Mean and standard deviation values were collected from distributions containing 300 or more unique reconstructed sequences.

Bibliography

Altschul, S. F., W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. 1990. “Basic Local Alignment Search Tool.” *Journal of Molecular Biology* 215 (3): 403–10. doi:10.1016/S0022-2836(05)80360-2.

Bloom, Jesse D., Lizhi Ian Gong, and David Baltimore. 2010. “Permissive Secondary Mutations Enable the Evolution of Influenza Oseltamivir Resistance.” *Science* 328 (5983): 1272–75. doi:10.1126/science.1187816.

Eick, Geeta N., Jamie T. Bridgham, Douglas P. Anderson, Michael J. Harms, and Joseph W. Thornton. 2017. “Robustness of Reconstructed Ancestral Protein Functions to Statistical Uncertainty.” *Molecular Biology and Evolution* 34 (2): 247–61. doi:10.1093/molbev/msw223.

Eick, Geeta N., Jennifer K. Colucci, Michael J. Harms, Eric A. Ortlund, and Joseph W. Thornton. 2012. “Evolution of Minimal Specificity and Promiscuity in Steroid Hormone Receptors.” *PLOS Genet* 8 (11): e1003072. doi:10.1371/journal.pgen.1003072.

Field, Steven F., Maria Y. Bulina, Ilya V. Kelmanson, Joseph P. Bielawski, and Mikhail V. Matz. 2006. “Adaptive Evolution of Multicolored Fluorescent Proteins in Reef-Building Corals.” *Journal of Molecular Evolution* 62 (3): 332–39. doi:10.1007/s00239-005-0129-9.

Gloor, Gregory B., Louise C. Martin, Lindi M. Wahl, and Stanley D. Dunn. 2005. “Mutual Information in Protein Multiple Sequence Alignments Reveals Two Classes of Coevolving Positions.” *Biochemistry* 44 (19): 7156–65. doi:10.1021/bi050293e.

Hanson-Smith, Victor, Bryan Kolaczkowski, and Joseph W. Thornton. 2010. “Robustness of Ancestral Sequence Reconstruction to Phylogenetic Uncertainty.” *Molecular Biology and Evolution* 27 (9): 1988–99. doi:10.1093/molbev/msq081.

Haq, Omar, Michael Andrec, Alexandre V. Morozov, and Ronald M. Levy. 2012. “Correlated Electrostatic Mutations Provide a Reservoir of Stability in HIV Protease.” *PLOS Comput Biol* 8 (9): e1002675. doi:10.1371/journal.pcbi.1002675.

Harms, Michael J. 2016. “Harmslab/pdbtools.” *GitHub*. Accessed June 8. https://github.com/harmslab/pdbtools.

Harms, Michael J., and Joseph W. Thornton. 2010. “Analyzing Protein Structure and Function Using Ancestral Gene Reconstruction.” *Current Opinion in Structural Biology* 20 (3): 360–66. doi:10.1016/j.sbi.2010.03.005.

Hart, Kathryn M., Michael J. Harms, Bryan H. Schmidt, Carolyn Elya, Joseph W. Thornton, and Susan Marqusee. 2014. “Thermodynamic System Drift in Protein Evolution.” *PLOS Biol* 12 (11): e1001994. doi:10.1371/journal.pbio.1001994.

Hopf, Thomas A., Satoshi Morinaga, Sayoko Ihara, Kazushige Touhara, Debora S. Marks, and Richard Benton. 2015. “Amino Acid Coevolution Reveals Three-Dimensional Structure and Functional Domains of Insect Odorant Receptors.” *Nature Communications* 6 (January): 6077. doi:10.1038/ncomms7077.

Kelmanson, Ilya V., and Mikhail V. Matz. 2003. “Molecular Basis and Evolutionary Origins of Color Diversity in Great Star Coral Montastraea Cavernosa (Scleractinia: Faviida).” *Molecular Biology and Evolution* 20 (7): 1125–33. doi:10.1093/molbev/msg130.

Lynch, Vincent J., Andrea Tanzer, Yajun Wang, Frederick C. Leung, Birgit Gellersen, Deena Emera, and Gunter P. Wagner. 2008. “Adaptive Changes in the Transcription Factor HoxA-11 Are Essential for the Evolution of Pregnancy in Mammals.” *Proceedings of the National Academy of Sciences* 105 (39): 14928–33. doi:10.1073/pnas.0802355105.

Mirceta, Scott, Anthony V. Signore, Jennifer M. Burns, Andrew R. Cossins, Kevin L. Campbell, and Michael Berenbrink. 2013. “Evolution of Mammalian Diving Capacity Traced by Myoglobin Net Surface Charge.” *Science* 340 (6138): 1234192. doi:10.1126/science.1234192.

Ortlund, Eric A., Jamie T. Bridgham, Matthew R. Redinbo, and Joseph W. Thornton. 2007. “Crystal Structure of an Ancient Protein: Evolution by Conformational Epistasis.” *Science* 317 (5844): 1544–48. doi:10.1126/science.1142819.

“RCSB Protein Data Bank - RCSB PDB.” 2016. Accessed June 8. http://www.rcsb.org/pdb/home/home.do.

Shapiro, Beth, Andrew Rambaut, Oliver G. Pybus, and Edward C. Holmes. 2006. “A Phylogenetic Method for Detecting Positive Epistasis in Gene Sequences and Its Application to RNA Virus Evolution.” *Molecular Biology and Evolution* 23 (9): 1724–30. doi:10.1093/molbev/msl037.

Socolich, Michael, Steve W. Lockless, William P. Russ, Heather Lee, Kevin H. Gardner, and Rama Ranganathan. 2005. “Evolutionary Information for Specifying a Protein Fold.” *Nature* 437 (7058): 512–18. doi:10.1038/nature03991.

Spielman, Stephanie J., and Claus O. Wilke. 2015. “Pyvolve: A Flexible Python Module for Simulating Sequences along Phylogenies.” *PLOS ONE* 10 (9): e0139047. doi:10.1371/journal.pone.0139047.

Süel, Gürol M., Steve W. Lockless, Mark A. Wall, and Rama Ranganathan. 2003. "Evolutionarily Conserved Networks of Residues Mediate Allosteric Communication in Proteins." *Nature Structural Biology* 10 (1): 59–69. doi:10.1038/nsb881.

Thornton, Joseph W. 2004. "Resurrecting Ancient Genes: Experimental Analysis of Extinct Molecules." *Nature Reviews Genetics* 5 (5): 366–75. doi:10.1038/nrg1324.

Wada, Akiyoshi, and Haruki Nakamura. 1981. "Nature of the Charge Distribution in Proteins." *Nature* 293 (5835): 757–58. doi:10.1038/293757a0.

Williams, Paul D., David D. Pollock, Benjamin P. Blackburne, and Richard A. Goldstein. 2006. "Assessing the Accuracy of Ancestral Protein Reconstruction Methods." *PLOS Comput Biol* 2 (6): e69. doi:10.1371/journal.pcbi.0020069.

Wilson, C., R. V. Agafonov, M. Hoemberger, S. Kutter, A. Zorba, J. Halpin, V. Buosi, et al. 2015. "Using Ancient Protein Kinases to Unravel a Modern Cancer Drug's Mechanism." *Science* 347 (6224): 882–86. doi:10.1126/science.aaa1823.

Yang, Z., S. Kumar, and M. Nei. 1995. "A New Method of Inference of Ancestral Nucleotide and Amino Acid Sequences." *Genetics* 141 (4): 1641–50.

Yip, Kevin Y., Prianka Patel, Philip M. Kim, Donald M. Engelman, Drew McDermott, and Mark Gerstein. 2008. "An Integrated System for Studying Residue Coevolution in Proteins." *Bioinformatics* 24 (2): 290–92. doi:10.1093/bioinformatics/btm584.

Zuckercandl, E, and L Pauling. 1965. "Evolutionary Divergence and Convergence in Proteins." In *Evolving Genes and Proteins*, edited by V Bryson and HJ Vogel, 97–165. Academic Press.