KNOWLEDGE-BASED ENVIRONMENT POTENTIALS

FOR PROTEIN STRUCTURE PREDICTION

By

Elizabeth Ashley Durham

Thesis

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

MASTER OF SCIENCE

in

Biomedical Informatics

August, 2008

Nashville, Tennessee

Approved:

Jens Meiler

David Tabb

Dan Masys

ACKNOWLEDGEMENTS

This work was funded by the National Library of Medicine Vanderbilt Biomedical Informatics training grant, 2-T15 LM07450-06. Without their financial support, this work would not have been possible.

I would like to thank the members of my Thesis committee, Dr. Dan Masys, Dr. David Tabb, and Dr. Jens Meiler for their time, ideas, and encouragement throughout this project.

I have learned more from my mentor and advisor, Jens Meiler, than I have from any teacher before. Nils Woetzel patiently oriented me to the Biochemical Library and development environment used in the Meiler Lab. Rene Staritzbichler provided guidance and developed the Overlapping Spheres Algorithm. Brent Dorr provided support and developed the Neighbor Vector Algorithm. I am grateful for the stimulating discussions and encouragement from all members of the Meiler Lab and the Department of Biomedical Informatics.

I would like to thank my friends and family for their love and invaluable support. Most importantly, I thank my Lord and Savior Jesus Christ; “In God we make our boast all day long, and we will praise your name forever.” (Psalm 44:8)

TABLE OF CONTENTS

Page

Chapter

LIST OF TABLES

Table Page

LIST OF FIGURES

LIST OF ABBREVIATIONS

ANN	artificial neural network
CATH	Classification, Architecture, Topology, and Homologous superfamily
DNA	deoxyribonucleic acid
KBP	knowledge-based potential
MC	monte carlo
MSMS	maximal speed molecular surfaces
NC	neighbor count
NV	neighbor vector
OLS	overlapping spheres
PDB	protein data bank
PSP	protein structure prediction
RMSD	root mean square distance
RNA	ribonucleic acid
SAE	solvent-accessible exposure
SASA	solvent-accessible surface area

CHAPTER I

INTRODUCTION

Overview

This Master's Thesis project had as its objectives: (1) to optimize algorithms for solvent-accessible surface area (SASA) approximation to develop an environment free energy knowledge-based potential; and, (2) to assess the knowledge-based environment free energy potentials for *de novo* protein structure prediction.

Protein Structure

The Importance of Protein Structure

The central dogma of biology states that DNA is transcribed to RNA, which is then translated into a protein[1]. While DNA is often described as the blueprint for life[2], proteins are the molecular machinery actually built from this DNA blueprint. Some important functions of proteins are maintenance of cellular structure, signaling, catalysis, immune defense, cellular defense, and molecular transportation.

A foundational belief in molecular biology is that the structure of a protein determines its function. Proteins are three-dimensional molecules that function by binding to molecules with a complementary interface. Therefore, elucidating the structure of the protein also provides information about the function of the protein. This knowledge is very important in drug and enzyme design.

Introduction to Protein Structure

Proteins are polymers of amino acids joined together by peptide bonds. Protein structure can be decomposed into four distinct aspects – primary through quaternary structure. The sequence of the amino acids present in a protein is called the primary structure of the protein. Local hydrogen bonding interactions between amino acids can cause the amino acid chain to form secondary structural elements. The common forms of secondary structure are ∝-helices and β-strands (represented pictorially as a coil and arrow respectively). The way in which these secondary structural elements come together in three-dimensional space defines the tertiary structure of the protein. Multiple protein subunits can bind to one another to form the quaternary structure of the protein. The native conformation, often referred to simply as “the native”, is the conformation that the protein naturally assumes. Aspects of protein structure are presented in Figure 1.

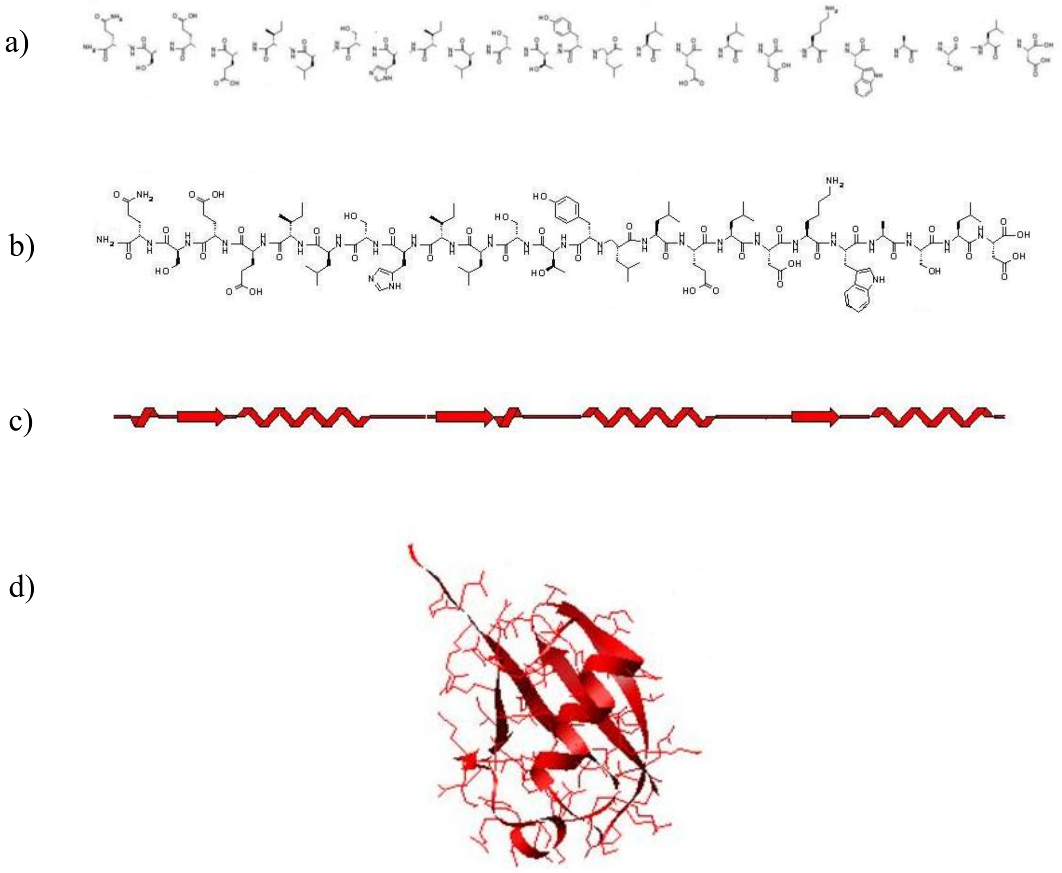


Figure 1: Aspects of protein structure. a) amino acids b) primary structure c) secondary structure d) tertiary structure

Computational Protein Structure Prediction

Genome sequencing has provided a wealth of information about the amino acid sequence of proteins, or the primary structure. While experimental methods provide structural information about proteins, these techniques are laborious and are not feasible for use on all proteins[3]. In particular, membrane proteins, which comprise up to 35% of all proteins[4] and 50% of all drug targets[5], are difficult to analyze with experimental techniques.

Therefore, there has been an increased demand for computational methods to predict the native conformation for such proteins and to assist in protein structure elucidation from sparse or low-resolution experimental data. The goal of *de novo* computational protein structure prediction (PSP) is to determine the native tertiary structure of a protein given only its primary structure.

Monte Carlo in Protein Structure Prediction

Levinthal's paradox states that even if you consider a simplified model of a small protein, the amount of time to conduct a brute force search of all possible conformations that the protein is able to assume is far greater than the time the universe has existed[6]. While there are a vast number of physical conformations available for a given protein sequence, it is believed that the native structure is the conformation with the lowest free energy[7]. Therefore, the protein folding problem becomes a search for the global minimum on the free energy landscape of all possible conformations of the amino acid sequence. An efficient procedure for searching conformational space and an energy function that approximates free energy are, hence, key components of *de novo* PSP techniques[3].

The sampling technique used for PSP in this work is a Monte Carlo (MC) algorithm with Metropolis criteria in a simulated annealing environment. *De novo* PSP begins by predicting secondary structure[8-16] and other properties of a given sequence such as β-hairpins[17], disorder[18,19], non-local contacts[20], domain boundaries[21,22], and domain interactions[23,24]. Secondary structural prediction methods are used to assemble a pool of candidate secondary structural elements. The task at hand is therefore to properly place the secondary structural elements together in three-dimensional space to determine the tertiary structure. The MC algorithm begins with a random three-dimensional placement of candidate secondary structural elements. A new model is generated by randomly perturbing the secondary structural elements in the existing model.

The energy of each model generated is determined by an energy evaluation function. If this new model is lower in energy than the current model, the new model is accepted as the current model. If this new model is higher in energy than the current model, the new, higher-energy model is accepted with a temperature-dependent probability given by the Boltzmann distribution. This is called the Metropolis criteria and allows the MC search to avoid entrapment in local minima[25]. The probability, P, that represents the likelihood with which a model is accepted as the current model is described by the following equation where E represents the energy of the model and T represents the pseudo-temperature.

$$P = \min\,(1, e^{-\frac{E_{new-}\,E_{current}}{T}})$$

The higher the pseudo-temperature, the more likely is the acceptance of a higher-energy model. Throughout the folding process, the temperature is lowered, allowing a focus on low-energy models. This is the simulated annealing principle. After an established number of new models are rejected, the folding process concludes. Such stochastic optimization methods are commonly used in protein folding[26] but can require over a million energy calculations[27]. Therefore, a computationally efficient energy evaluation function is needed.

A PSP program that uses the MC algorithm has been implemented in the Meiler Lab and is able to fold proteins to native-like conformations. However, the energy evaluation function can be improved by an accurate environment energy potential.

Knowledge-Based Potentials

Whereas molecular mechanics force fields seek to describe the energy directly associated with known physical interactions, knowledge-based potentials (KBPs) are derived from statistics generated from known three-dimensional protein structures. Hence, they approximate the overall free energy more generally, and frequently encompass multiple classical energy terms associated with a physical interaction. KBPs relate the likelihood of a conformation occurring, to the energy associated with that conformation. KBPs have been successful tools in predicting protein structure[28-32], predicting protein-protein interactions[33,34], predicting protein-ligand interactions[35-39], and in protein design[40,41]. Further details on the generation of a KBP are given in Chapter V.

Model Complexity

In order to effectively and efficiently search the conformational space of an amino acid sequence, a reduced representation of amino acid side chains is used. The algorithms presented in this work take as input protein models with a reduced side chain representation where all side chain atoms with the exception of the C_β atom are removed. (For glycine, a pseudo- C_β atom is used.) These aspects of the model are simplified in order to decrease the computational complexity of the task in order to make it more efficient.

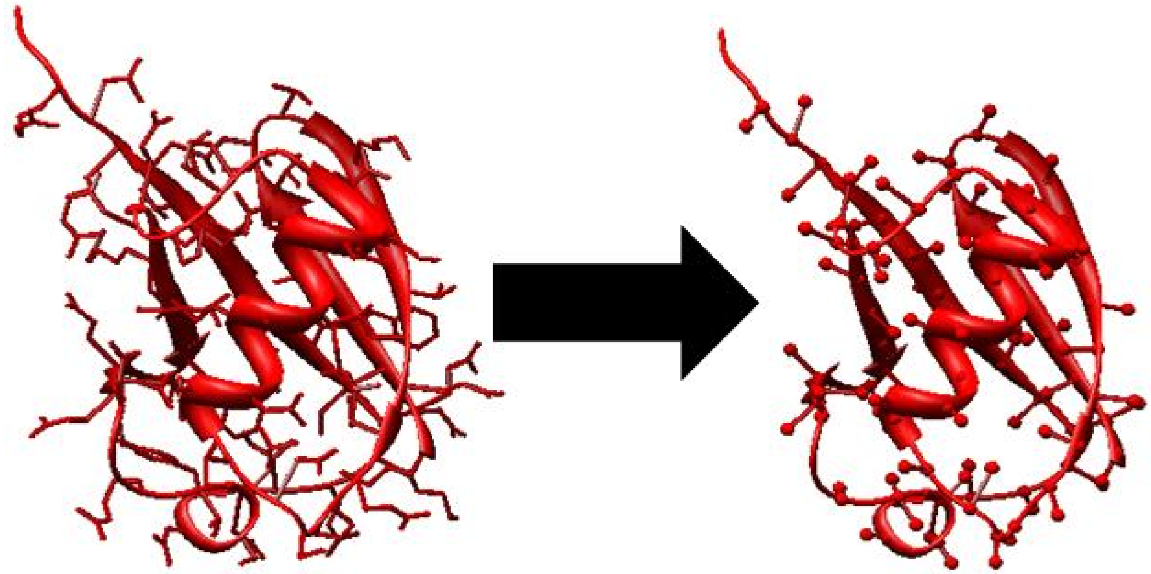


Figure 2: Reduced amino acid side chain representation.

Environment Free-Energy

The extent to which an amino acid interacts with its environment is naturally proportional to the degree to which the amino acid is exposed to solvent. The solvent-accessible surface area (SASA) is a geometric measure of this exposure.

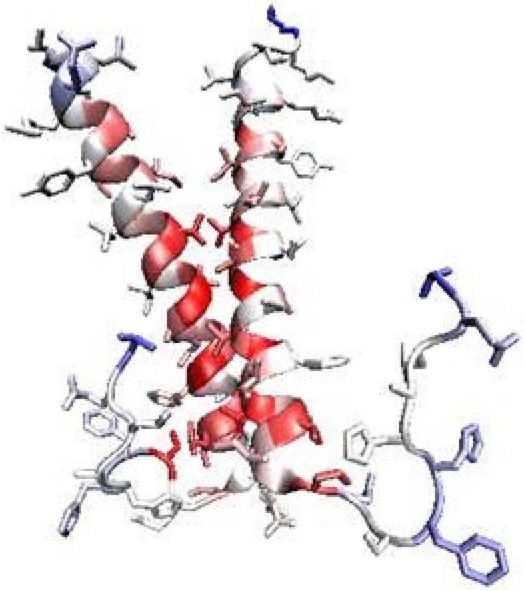


Figure 3: Protein colored by solvent-accessibility where red represents burial and blue represents exposure.

The environment free energy is approximately proportional to the SASA of the amino acid. In fact, several energy evaluation functions assume a strictly linear relationship between SASA and the environment free energy[42,43], neglecting the non-linear complexities of this relationship, due to the computational complexity of a precise SASA calculation. This work seeks to efficiently capture these non-linear complexities.

Energetic terms, such as hydrogen bonding, electrostatics, and van der Waals forces, contribute to the interaction between atoms within a protein[44] and also govern interactions between protein and solvent. However, an explicit calculation of these interactions is computationally complex and is therefore often omitted[27]. Nevertheless, to correctly evaluate the free energy of a protein in solution, an accurate description of its interaction with the solvent is imperative[45,46]. Otherwise, observed effects like surface area minimization, burial of hydrophobic side chains, and strength of hydrogen bonds cannot be described.

An effective free energy evaluation function is an essential part of *de novo* structure prediction methods. An important piece of such a function is the inclusion of an energetic evaluation of the environment free energy. This work provides an environment free energy evaluation function based on the SASA of amino acids in the protein model.

CHAPTER II

RELATED WORK

Wodak and Janin define the accessible surface area as “the area on the surface over which a water molecule can be placed while making van der Waals contact with this atom and not penetrating any other protein atom”[47]. Lee and Richards presented the first algorithm for calculating the solvent-accessible surface area of a molecular surface[48]. Their method involved the extension of the van der Waals radius for each atom by 1.4 Å (the radius of a water molecule) and the calculation of the surface area of these expanded-radius atoms. The Shrake and Rupley algorithm[49] involves the testing of points on an atom’s van der Waals surface for overlaps with points on the van der Waals surface of neighboring atoms. Many additional methods followed including spherical probing methods[50,51] and geometric fitting approaches[52].

While these methods provide a very accurate SASA measure, they are also computationally intensive. Many approximations have been developed, such as statistical approximations based on atom distances[47], lattice approximations[53,54], and spline approximations[55]. Other methods attempt to create a pairwise-decomposable method of SASA approximation[56]. One of the more efficient algorithms is the Maximal Speed Molecular Surfaces (MSMS) algorithm which fits spherical and toroidal patches onto the surfaces of atoms based on which points on the atom are accessible to a spherical probe that approximates a solvent molecule[57]. The SASA calculated by the MSMS algorithm is used as a reference standard.

In addition to an efficient method, the method should also calculate the SASA for each residue, rather than for the protein as a whole. This is necessary in order to take advantage of the knowledge-based technique. The method should also be able to accurately calculate the SASA on the reduced side chain representation described in Figure 3. While many of the methods described in the previous paragraph provide an accurate SASA calculation, these methods are not able to calculate a per-residue SASA in a manner efficient enough for use in MC folding programs.

Similar work has been done by the Baker Lab and is implemented in a folding program called Rosetta[32,58]. However, the measure they use for the calculation of SASA is limited and fails to accurately describe exposure in some cases (shown below). Novel elements of the work described in this thesis include a comprehensive, optimized geometric measure of SASA and the evaluation of different approximation methods.

CHAPTER III

ALGORITHM DEVELOPMENT

Reference Standard

The Visual Molecular Dynamics (VMD)[59] molecular visualization package implements the MSMS algorithm. The SASA as calculated by the VMD implementation of MSMS is used as a reference standard and serves as a basis of comparison for the SASA approximation algorithms developed. The probe radius used is 1.4 Å, the radius of a water molecule and an accepted value for solvent probe size[48]. All hydrogens were removed from the structural information for consistency as hydrogen coordinates are not always available.

The SASA is converted from an area measured in Å to a relative exposure that can range from 0.0 (completely buried) to 1.0 (complexity exposed). This relative exposure will be referred to as the solvent-accessible exposure (SAE). In creation of the reference standard, SASA is converted to SAE by dividing the SASA calculated for an amino acid in a protein model by the SASA for that amino acid alone in space (i.e. all atoms in the protein that are not constituent atoms of the given amino acid are removed and the SASA is calculated). The conversion from SASA to SAE facilitates comparison between amino acids of various sizes and allows a more general prediction of SASA in the absence of side chain coordinates.

Introduction to Algorithms

Neighbor Count

The central idea behind the Neighbor Count algorithm is that the number of neighboring amino acids is inversely proportional to exposure. A previous method defines neighboring amino acids as those whose C_β atom is within a distance of 10 Å of the C_β of the amino acid of interest[58]. The definition of a "neighbor" is expanded in this work by assigning a weight between 0.0 and 1.0 to all other amino acids in the protein based on their proximity to the amino acid of interest. A lower boundary and an upper boundary are chosen such that all amino acids at a distance less than or equal to the lower boundary are assigned a neighbor weight of 1 (i.e. they are a complete neighbor), amino acids at a distance greater than the upper boundary are assigned a neighbor weight of 0 (i.e. they are not neighbors at all), and amino acids at a distance between the lower and upper bounds are assigned a weight between 0.0 and 1.0 (i.e. they are partial neighbors). This allows amino acids that are spatially close to the amino acid of interest to have a greater input in determining the neighbor count.

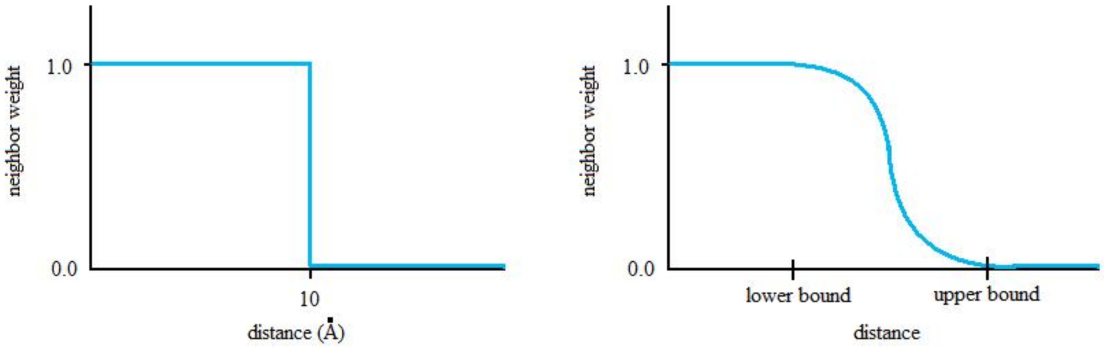


Figure 4: Neighbor weight functions. a) basic function b) expanded function

$$NeighborWeight = \begin{cases} 1, & dist \leq lower\ bound \\ \dfrac{\left(cos\left(\frac{(distance - lower_{bound})}{(upper_{bound} - lower_{bound})} * \Pi\right) + 1.0\right)}{2.0}, & lower\ bound < dist < upper\ bound \\ 0, & dist\ \geq upper\ bound \end{cases}$$

The Neighbor Count for each amino acid is then found by summing the neighbor weights of all other amino acids in the protein.

$$NeighborCount(aa_{\mathrm{i}}) =$$

$$\sum_{j \neq i} NeighborWeight(dist(aa_{\mathrm{i}}, aa_{\mathrm{j}}), lower\ bound, upper\ bound)$$

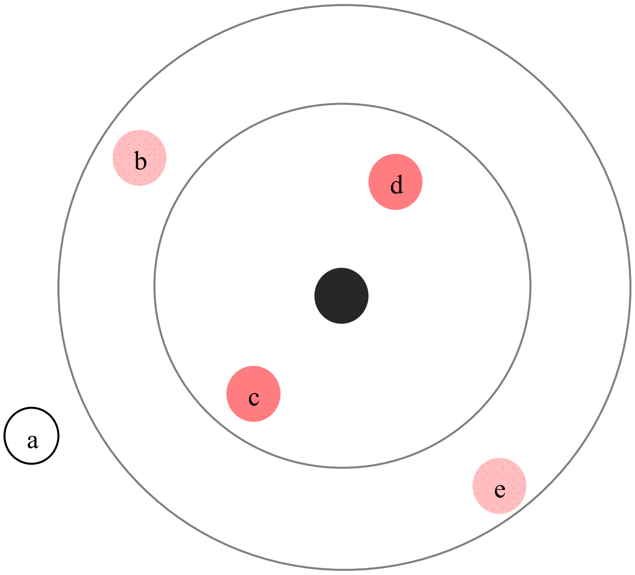


Figure 5: The neighbor count algorithm. The inner and outer gray rings represent the lower and upper bounds respectively. All other spheres represent the C_{β} atoms of amino acids. The black circle represents the amino acid of interest. Amino acids a and f are assigned a neighbor weight of 0 because they are outside of the upper bound. Amino acids a and e are assigned a weight between 0 and 1 because they lie between the upper and lower bounds. Amino acids c and d are counted as one complete neighbor each because they lie within the lower bound.

A shortcoming of the Neighbor Count algorithm is that it does not take into account the spatial distribution of its neighbors. For an example of this behavior, consider the scenarios shown in Figure 7 which represent different exposures yet return the same Neighbor Count.

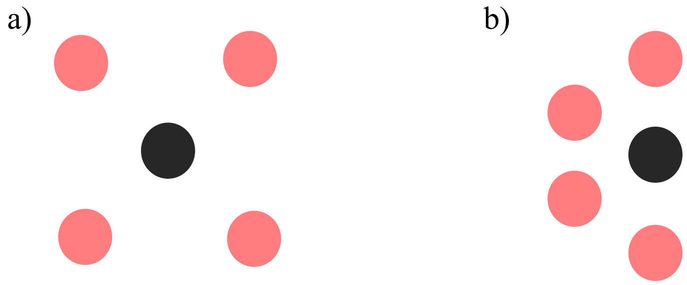


Figure 6: Shortcoming of the neighbor count algorithm. The scenarios depicted in a) and b) return the same Neighbor Count but represent different exposures.

Neighbor Vector

The Neighbor Vector algorithm is an extension of the Neighbor Count algorithm that takes into account the spatial orientation of neighboring amino acids when determining the exposure.

$$NeighborVector(aa_i) =$$

$$\left\| \frac{\sum_{j\neq i} \left(\overrightarrow{Vector_{i,j}} \,/\, \|\overrightarrow{Vector_{i,j}}\|\right) * NeighborWeight(dist(i,j), lower\ bound, upper\ bound)}{NeighborCount(aa_i)} \right\|$$

The neighbor vector is a vector associated with each amino acid whose length can range from 0.0 to 1.0. A neighbor vector of length ≅ 1 implies high exposure whereas a neighbor vector of length ≅ 0 implies low exposure (i.e. burial).

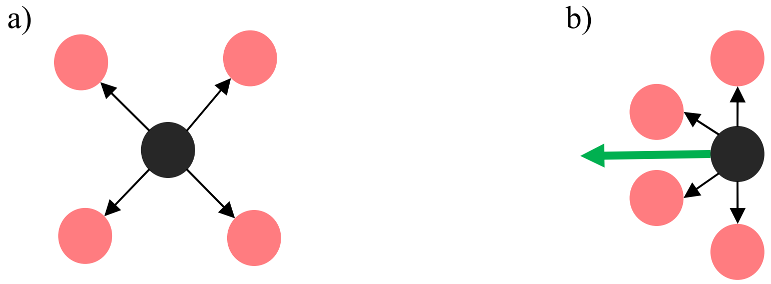


Figure 7: The Neighbor Vector algorithm is able to distinguish between the scenarios presented previously between which the Neighbor Count algorithm could not distinguish. Arrows drawn to all neighbors are shown in black. The summation of these vectors is the Neighbor Vector and is shown in green. a) The vectors drawn to all neighbors essentially cancel out when summed and yield a Neighbor Vector of magnitude ≅ 0. b) The vectors drawn to all neighbors yield a Neighbor Vector with a large magnitude when summed.

However, there are still scenarios between which the Neighbor Vector algorithm cannot distinguish.

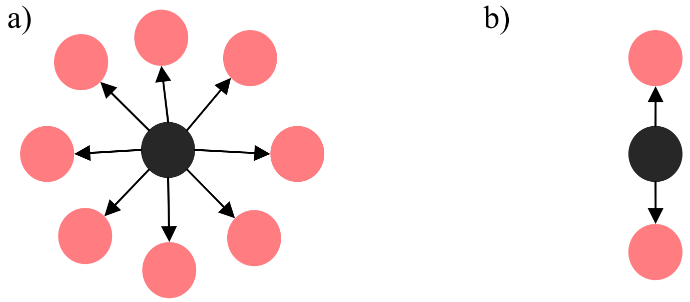


Figure 8: Shortcoming of the neighbor vector algorithm. The scenarios depicted in a) and b) return the same Neighbor Vector (≅0) but represent different exposures.

Artificial Neural Network

In order to distinguish between the scenarios for which the Neighbor Vector algorithm returns an ambiguous Neighbor Vector, an artificial neural network (ANN) is used to predict SAE. An additional term is introduced as an input to the Artificial Neural Network (ANN): the dot product of the ($C_\propto - C_\beta$) vector with the Neighbor Vector ($NV \bullet (C_\propto - C_\beta)$). Recall that the side chain atoms (which have been removed in these reduced models) extend from the C_β atom. Therefore, this dot product provides additional information about the position of the side chain with respect to the neighboring amino acids.

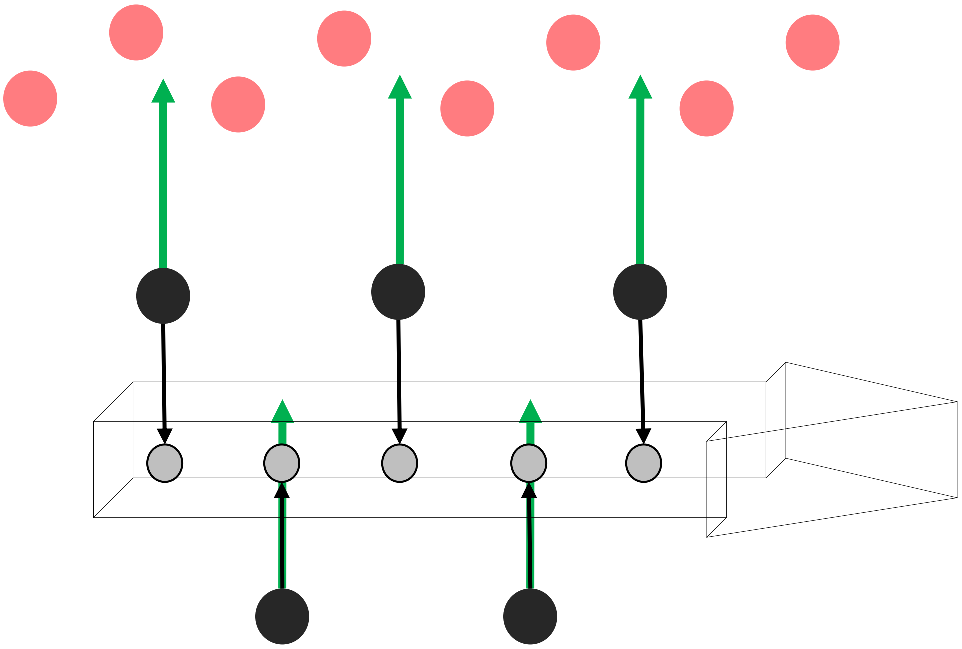


Figure 9: The ($C_\propto$ - C_β) vector gives additional information about the orientation of side chain atoms with respect to neighboring amino acids. The long, flat arrow represents a β-sheet, the red balls represent the C_β atoms of neighboring amino acids, the black balls the C_β atoms of the amino acids of the β-sheet, the gray balls the $C_\propto$ atoms of the amino acids of the β-sheet, the green arrows the Neighbor Vectors, and the black arrows the ($C_\propto$ - C_β) vectors.

The output of the Neighbor Count algorithm, the output of the Neighbor Vector algorithm, and the result of $NV \bullet (C_{\propto} - C_{\beta})$ are provided as inputs to the ANN. Note that the NV used in the dot product calculation is not normalized by NeighborCount_i as shown in the Neighbor Count formula because the ANN is already receiving information about the Neighbor Count.

Overlapping Spheres

The Overlapping Spheres algorithm is a variant of the Shrake and Rupley algorithm[49] for calculating molecular surfaces with the exception that spheres surround only the C_β of each amino acid rather than each atom. In this algorithm, a sphere is placed around each C_β and points are placed on the surface of the sphere surrounding the amino acid of interest. The points on this sphere that do not overlap with the spheres surrounding any neighboring amino acids are used as a measure of exposure.

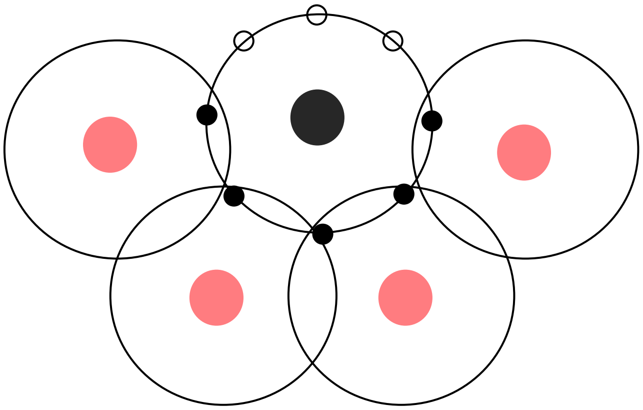


Figure 10: The Overlapping Spheres algorithm. The black ball represents the C_β atom of the amino acid of interest and the red balls represent the C_β atoms of neighboring amino acids. Each C_β is surrounded by a sphere (shown in two dimensions as a ring here). The small, filled spheres represent points that overlap with the spheres surrounding neighboring amino acids. The small, unfilled spheres represent points that do not overlap with the spheres surrounding neighboring amino acids and are used as an approximation for exposure. The Overlapping Spheres algorithm would determine that the amino acid of interest is 37.5% (3/8) exposed.

Parameter Optimization

In order to determine the optimal parameters for each algorithm, all parameters in a reasonable range were systematically test. The parameter set that produced exposures correlated most with exposures produced by the MSMS reference standard was selected as optimal.

Table 1: Optimal parameters for SAE algorithms.

Algorithm	**Optimal Parameters**
Neighbor Count	lower bound: 4.0 Å, upper bound: 11.4 Å
Neighbor Vector	lower bound: 3.3 Å, upper bound: 11.1 Å
Artificial Neural Network	nine inputs are provided to the ANN:
	- NC(2.0, 9.4), NV(1.3, 9.1), & NV(1.3,9.1) • ($C_\propto - C_\beta$)
	- NC(4.0, 11.4), NV(3.3, 11.1), & NV(3.3, 11.1) • ($C_\propto - C_\beta$)
	- NC(6.0, 13.4), NV(5.3, 13.1), & NV(5.3, 13.1)•($C_\propto - C_\beta$)
Overlapping Spheres	sphere radius: 4.75 Å

CHAPTER IV

ALGORITHM EVALUATION

Comparison to Reference Standard

In order to determine algorithm accuracy, the SAEs produced by each algorithm are compared to the SAEs produced by the MSMS referenced standard.

Table 2: Correlation of each SAE algorithm with the MSMS reference standard.

Algorithm	**Correlation with MSMS Reference Standard**
Neighbor Count	-0.846
Neighbor Vector	0.889
Artificial Neural Network	0.904
Overlapping Spheres	0.900

Note that the correlation for the Neighbor Count algorithm is negative due to the fact that the number of neighbors is inversely proportional to SAE. As expected, as the complexity of the algorithm increases, the quality of SAEs produced also increases.

Runtime Analysis

Table 3: Runtime of SAE Algorithms.

Algorithm	**Runtime on 58 Proteins (seconds)**
MSMS	12,195
Neighbor Count	63
Neighbor Vector	63
Artificial Neural Network	342
Overlapping Spheres	551

As expected, as the complexity of the algorithm increases, the run time also increases.

CHAPTER V

GENERATION OF KNOWLEDGE-BASED POTENTIALS

Establishment of Representative Protein Database

Experimentally-determined protein structure information is stored in the Protein Data Bank (PDB)[60,61]. Proteins used for the generation of the KBP are selected from the Dunbrack database[62,63], a subset of the PDB that is selected to represent high quality, non-repetitive structures.

Table 4: Protein database used to generate KBPs.

protein group	**# proteins**	**# amino acids**	**# ∝-helices**	**# β-strands**
membrane proteins	58	47,635	1,545	1,454
soluble proteins	1,795	884,529	32,075	32,641

Generating Potentials for Soluble Proteins

A histogram for each amino acid was generated for each of the algorithms by running the algorithms over the protein structures in the representative protein database described in Table 4. The following equation describes how histograms are generated for each amino acid type.

$$histogram\_aa_{\mathrm{i}}[j] = \frac{\left[1 + \sum_{aa_{\mathrm{i}}}^{n} equal\ exposure(aa_{i,}\ e_{j})\right]}{\sum_{e_{\mathrm{k}}}^{m} histogram\_aa_{\mathrm{i}}[k]} * \ m$$

$$equal\ exposure(aa_i, e_j) = \begin{cases} 1, e(aa_{\mathrm{i}}) = e_{\mathrm{j}} \\ 0, e(aa_{\mathrm{i}}) \neq e_{\mathrm{j}} \end{cases}$$

where aa_i is amino acid type i, n is the number of amino acids of type i in the database, e_j is the range of exposure values j associated with that bin, and m is the number of exposure values (we allowed 20 exposure values). Prior to multiplication by the number of exposure values, the values in each bin are probabilities ($0 \leq$ probability ≤ 1). Multiplying by the number of exposure values converts these probabilities to propensities ($0 \leq$ propensity $\leq$ number of exposure values). Propensities are then converted to energies according to the following equation:

$$energy(e_{\mathrm{j}}|aa_{\mathrm{i}}) = -ln(histogram[i][j])$$

Table 5: Relationship between probabilities, propensities, and energies. Each exposure value is assumed to be equally likely, therefore P_{random} = 1 / (number of possible exposure values).

Probability	**Propensity**	**Energy**
Probability > P_{random}	Propensity > 1	Energy < 0 ("low energy")
Probability = P_{random}	Propensity = 1	Energy = 0 ("neutral energy")
Probability < P_{random}	0 < Propensity < 1	Energy > 0 ("high energy")

Essentially exposure values that are seen rarely in native proteins are associated with a high energy whereas exposure values that are seen often in native proteins are associated with a lower energy. A spline is used to smooth the histogram bins into a differentiable potential. The addition of a pseudocount of 1 to each bin is necessary because $-\ln(0) = \infty$ and while exposure values not seen in native proteins should be "penalized" with high energy scores, they should not be explicitly forbidden.

a) Counts

	neighbor count = 1	neighbor count = 2	total
ala	0	8	8
tyr	4	4	8
lys	74	4	78
total	78	16	94

add a pseudo-count of 1 to each bin

b) Counts + Pseudo-counts

	neighbor count = 1	neighbor count = 2	total
	1	9	10
tyr	5	5	10
lys	75	5	80
total	81	19	100

divide by the number of amino acids of that type

c) Conditional Probabilities

	neighbor count = 1	neighbor count = 2	total
ala	0.1	0.9	1.0
tyr	0.5	0.5	1.0
lys	0.9375	0.0625	1.0
total	1.5375	1.4625	3.0

multiply by the number of possible exposure values

d) Conditional Propensities

	neighbor count = 1	neighbor count = 2	total
ala	0.2	1.8	2.0
tyr	1.0	1.0	2.0
lys	1.875	0.125	2.0
total	3.075	2.925	6.0

take the –ln of each bin

e) Energies

	neighbor count = 1	neighbor count = 2
ala	1.6	-0.49
tyr	0.0	0.0
lys	-0.63	20.8

Figure 11: Simplified example of KBP generation. Assume that there are only two possible exposure values and only three amino acids.

Generating Potentials for Membrane Proteins

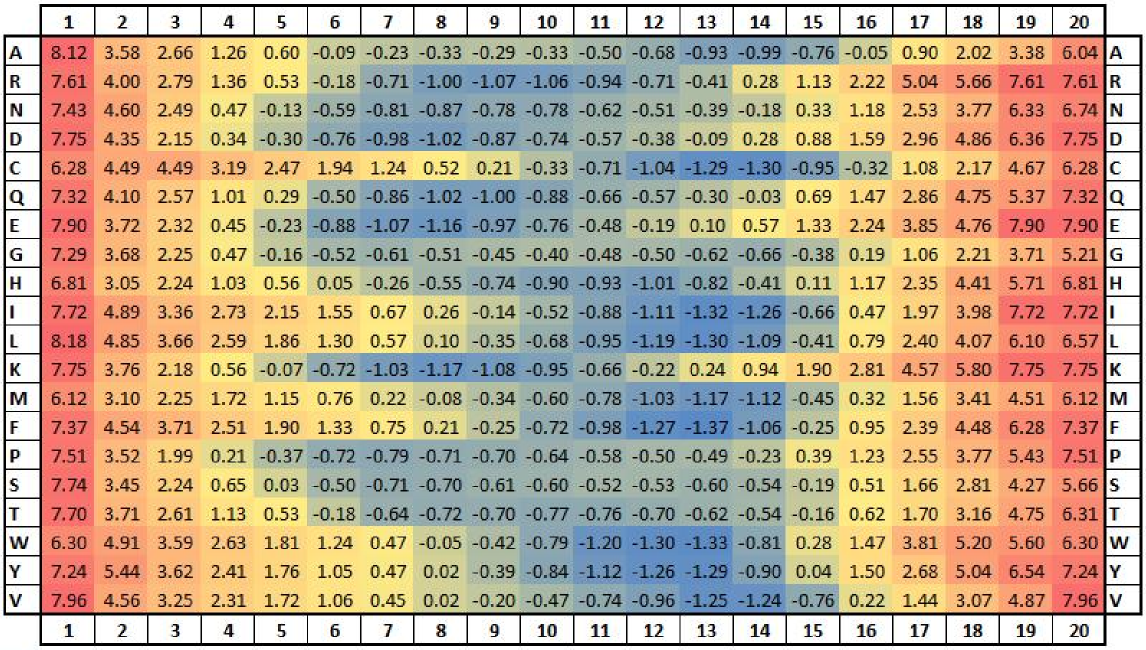
A similar procedure is used to generate KBPs for membrane proteins. However, additional considerations are taken into account for membrane proteins. Specifically, membrane proteins come into contact with three distinct regions: the hydrophobic interior of the membrane core, the highly charged transition region containing fatty acid head groups, and the polar solution on either side of the membrane. A potential is generated for each of these distinct regions as the energies associated with a given exposure value differ in these three environments.

CHAPTER VI

EVALUATION OF KNOWLEDGE-BASED POTENTIALS

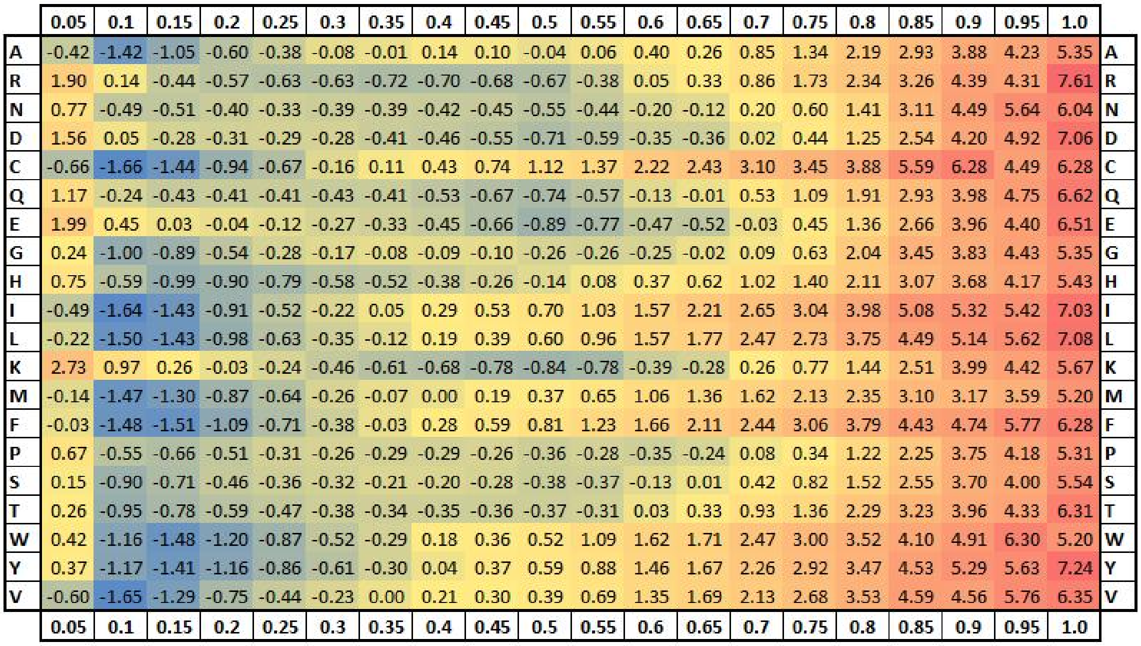
Visualization of Knowledge-Based Potentials

A visual inspection of the KBPs allows us to verify that the potentials agree with expectations. For example, it is expected that hydrophobic amino acids in solution prefer burial. This is in fact what is seen. Consider the preference of hydrophobic amino acids, such as valine (V), methionine (M), and phenylalanine (F) for a large number of neighbors (Figure 12), a small neighbor vector magnitude (Figure 13), and small SAEs (Figures 14 and 15). Additionally, it is expected that hydrophilic amino acids prefer a high degree of exposure in solution. This is also the case. Consider the preference of the hydrophilic amino acids lysine (K), asparagine (N), and glutamine (Q) for low neighbor counts (Figure 12), a large neighbor vector magnitude (Figure 13), and large SAEs (Figures 14 and 15). The KBPs for soluble proteins generated by each algorithm are shown in Figures 12 – 15.



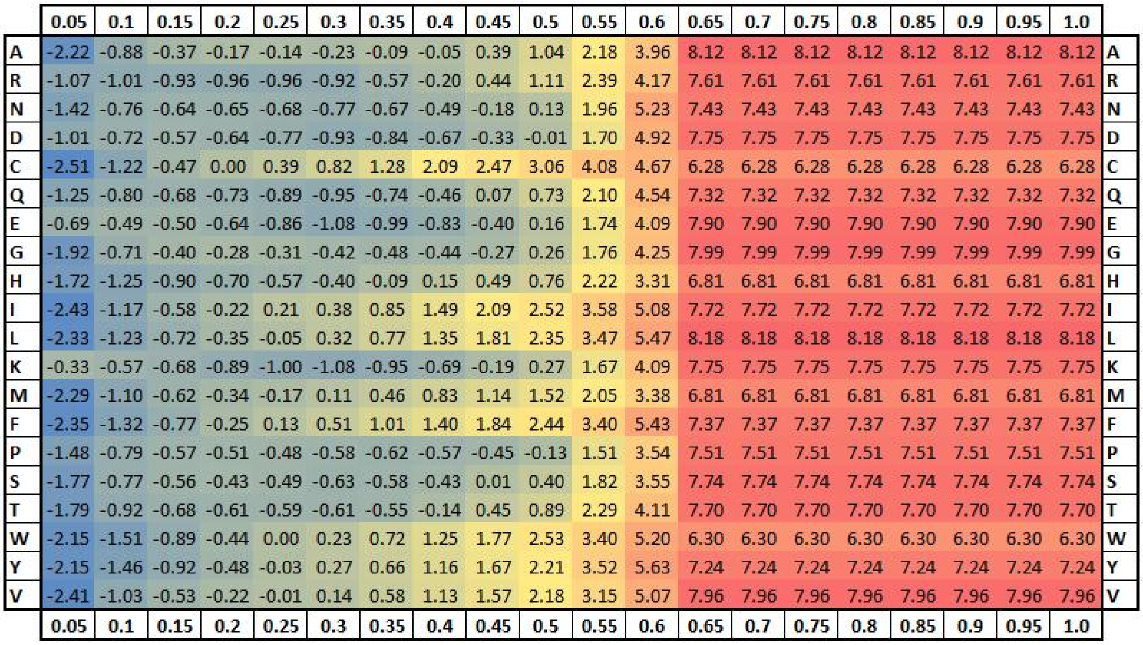
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
A	8.12	3.58	2.66	1.26	0.60	-0.09	-0.23	-0.33	-0.29	-0.33	-0.50	-0.68	-0.93	-0.99	-0.76	-0.05	0.90	2.02	3.38	6.04	A
R	7.61	4.00	2.79	1.36	0.53	-0.18	-0.71	-1.00	-1.07	-1.06	-0.94	-0.71	-0.41	0.28	1.13	2.22	5.04	5.66	7.61	7.61	R
N	7.43	4.60	2.49	0.47	-0.13	-0.59	-0.81	-0.87	-0.78	-0.78	-0.62	-0.51	-0.39	-0.18	0.33	1.18	2.53	3.77	6.33	6.74	N
D	7.75	4.35	2.15	0.34	-0.30	-0.76	-0.98	-1.02	-0.87	-0.74	-0.57	-0.38	-0.09	0.28	0.88	1.59	2.96	4.86	6.36	7.75	D
C	6.28	4.49	4.49	3.19	2.47	1.94	1.24	0.52	0.21	-0.33	-0.71	-1.04	-1.29	-1.30	-0.95	-0.32	1.08	2.17	4.67	6.28	C
Q	7.32	4.10	2.57	1.01	0.29	-0.50	-0.86	-1.02	-1.00	-0.88	-0.66	-0.57	-0.30	-0.03	0.69	1.47	2.86	4.75	5.37	7.32	Q
E	7.90	3.72	2.32	0.45	-0.23	-0.88	-1.07	-1.16	-0.97	-0.76	-0.48	-0.19	0.10	0.57	1.33	2.24	3.85	4.76	7.90	7.90	E
G	7.29	3.68	2.25	0.47	-0.16	-0.52	-0.61	-0.51	-0.45	-0.40	-0.48	-0.50	-0.62	-0.66	-0.38	0.19	1.06	2.21	3.71	5.21	G
H	6.81	3.05	2.24	1.03	0.56	0.05	-0.26	-0.55	-0.74	-0.90	-0.93	-1.01	-0.82	-0.41	0.11	1.17	2.35	4.41	5.71	6.81	H
I	7.72	4.89	3.36	2.73	2.15	1.55	0.67	0.26	-0.14	-0.52	-0.88	-1.11	-1.32	-1.26	-0.66	0.47	1.97	3.98	7.72	7.72	I
L	8.18	4.85	3.66	2.59	1.86	1.30	0.57	0.10	-0.35	-0.68	-0.95	-1.19	-1.30	-1.09	-0.41	0.79	2.40	4.07	6.10	6.57	L
K	7.75	3.76	2.18	0.56	-0.07	-0.72	-1.03	-1.17	-1.08	-0.95	-0.66	-0.22	0.24	0.94	1.90	2.81	4.57	5.80	7.75	7.75	K
M	6.12	3.10	2.25	1.72	1.15	0.76	0.22	-0.08	-0.34	-0.60	-0.78	-1.03	-1.17	-1.12	-0.45	0.32	1.56	3.41	4.51	6.12	M
F	7.37	4.54	3.71	2.51	1.90	1.33	0.75	0.21	-0.25	-0.72	-0.98	-1.27	-1.37	-1.06	-0.25	0.95	2.39	4.48	6.28	7.37	F
P	7.51	3.52	1.99	0.21	-0.37	-0.72	-0.79	-0.71	-0.70	-0.64	-0.58	-0.50	-0.49	-0.23	0.39	1.23	2.55	3.77	5.43	7.51	P
S	7.74	3.45	2.24	0.65	0.03	-0.50	-0.71	-0.70	-0.61	-0.60	-0.52	-0.53	-0.60	-0.54	-0.19	0.51	1.66	2.81	4.27	5.66	S
T	7.70	3.71	2.61	1.13	0.53	-0.18	-0.64	-0.72	-0.70	-0.77	-0.76	-0.70	-0.62	-0.54	-0.16	0.62	1.70	3.16	4.75	6.31	T
W	6.30	4.91	3.59	2.63	1.81	1.24	0.47	-0.05	-0.42	-0.79	-1.20	-1.30	-1.33	-0.81	0.28	1.47	3.81	5.20	5.60	6.30	W
Y	7.24	5.44	3.62	2.41	1.76	1.05	0.47	0.02	-0.39	-0.84	-1.12	-1.26	-1.29	-0.90	0.04	1.50	2.68	5.04	6.54	7.24	Y
V	7.96	4.56	3.25	2.31	1.72	1.06	0.45	0.02	-0.20	-0.47	-0.74	-0.96	-1.25	-1.24	-0.76	0.22	1.44	3.07	4.87	7.96	V
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	

Figure 12: KBP for soluble proteins produced by the Neighbor Count algorithm. Each row shows the potential for an amino acid (labeled on left and right). Each column represents the neighbor count (labeled on top and bottom). Red represents high energy while blue represents low energy.



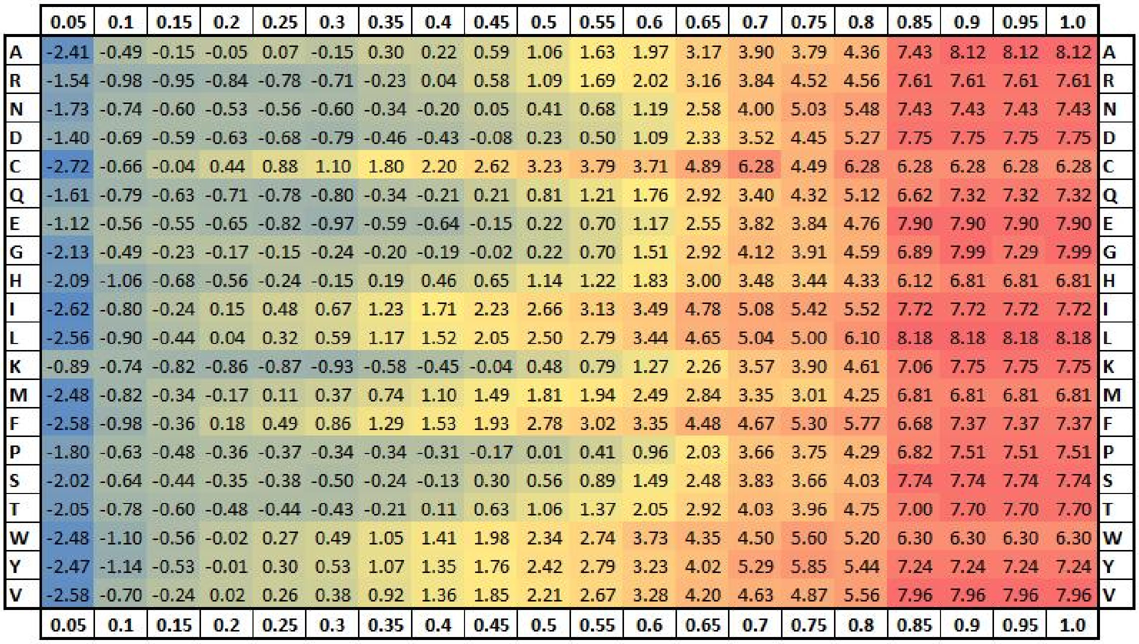
	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	0.55	0.6	0.65	0.7	0.75	0.8	0.85	0.9	0.95	1.0	
A	-0.42	-1.42	-1.05	-0.60	-0.38	-0.08	-0.01	0.14	0.10	-0.04	0.06	0.40	0.26	0.85	1.34	2.19	2.93	3.88	4.23	5.35	A
R	1.90	0.14	-0.44	-0.57	-0.63	-0.63	-0.72	-0.70	-0.68	-0.67	-0.38	0.05	0.33	0.86	1.73	2.34	3.26	4.39	4.31	7.61	R
N	0.77	-0.49	-0.51	-0.40	-0.33	-0.39	-0.39	-0.42	-0.45	-0.55	-0.44	-0.20	-0.12	0.20	0.60	1.41	3.11	4.49	5.64	6.04	N
D	1.56	0.05	-0.28	-0.31	-0.29	-0.28	-0.41	-0.46	-0.55	-0.71	-0.59	-0.35	-0.36	0.02	0.44	1.25	2.54	4.20	4.92	7.06	D
C	-0.66	-1.66	-1.44	-0.94	-0.67	-0.16	0.11	0.43	0.74	1.12	1.37	2.22	2.43	3.10	3.45	3.88	5.59	6.28	4.49	6.28	C
Q	1.17	-0.24	-0.43	-0.41	-0.41	-0.43	-0.41	-0.53	-0.67	-0.74	-0.57	-0.13	-0.01	0.53	1.09	1.91	2.93	3.98	4.75	6.62	Q
E	1.99	0.45	0.03	-0.04	-0.12	-0.27	-0.33	-0.45	-0.66	-0.89	-0.77	-0.47	-0.52	-0.03	0.45	1.36	2.66	3.96	4.40	6.51	E
G	0.24	-1.00	-0.89	-0.54	-0.28	-0.17	-0.08	-0.09	-0.10	-0.26	-0.26	-0.25	-0.02	0.09	0.63	2.04	3.45	3.83	4.43	5.35	G
H	0.75	-0.59	-0.99	-0.90	-0.79	-0.58	-0.52	-0.38	-0.26	-0.14	0.08	0.37	0.62	1.02	1.40	2.11	3.07	3.68	4.17	5.43	H
I	-0.49	-1.64	-1.43	-0.91	-0.52	-0.22	0.05	0.29	0.53	0.70	1.03	1.57	2.21	2.65	3.04	3.98	5.08	5.32	5.42	7.03	I
L	-0.22	-1.50	-1.43	-0.98	-0.63	-0.35	-0.12	0.19	0.39	0.60	0.96	1.57	1.77	2.47	2.73	3.75	4.49	5.14	5.62	7.08	L
K	2.73	0.97	0.26	-0.03	-0.24	-0.46	-0.61	-0.68	-0.78	-0.84	-0.78	-0.39	-0.28	0.26	0.77	1.44	2.51	3.99	4.42	5.67	K
M	-0.14	-1.47	-1.30	-0.87	-0.64	-0.26	-0.07	0.00	0.19	0.37	0.65	1.06	1.36	1.62	2.13	2.35	3.10	3.17	3.59	5.20	M
F	-0.03	-1.48	-1.51	-1.09	-0.71	-0.38	-0.03	0.28	0.59	0.81	1.23	1.66	2.11	2.44	3.06	3.79	4.43	4.74	5.77	6.28	F
P	0.67	-0.55	-0.66	-0.51	-0.31	-0.26	-0.29	-0.29	-0.26	-0.36	-0.28	-0.35	-0.24	0.08	0.34	1.22	2.25	3.75	4.18	5.31	P
S	0.15	-0.90	-0.71	-0.46	-0.36	-0.32	-0.21	-0.20	-0.28	-0.38	-0.37	-0.13	0.01	0.42	0.82	1.52	2.55	3.70	4.00	5.54	S
T	0.26	-0.95	-0.78	-0.59	-0.47	-0.38	-0.34	-0.35	-0.36	-0.37	-0.31	0.03	0.33	0.93	1.36	2.29	3.23	3.96	4.33	6.31	T
W	0.42	-1.16	-1.48	-1.20	-0.87	-0.52	-0.29	0.18	0.36	0.52	1.09	1.62	1.71	2.47	3.00	3.52	4.10	4.91	6.30	5.20	W
Y	0.37	-1.17	-1.41	-1.16	-0.86	-0.61	-0.30	0.04	0.37	0.59	0.88	1.46	1.67	2.26	2.92	3.47	4.53	5.29	5.63	7.24	Y
V	-0.60	-1.65	-1.29	-0.75	-0.44	-0.23	0.00	0.21	0.30	0.39	0.69	1.35	1.69	2.13	2.68	3.53	4.59	4.56	5.76	6.35	V
	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	0.55	0.6	0.65	0.7	0.75	0.8	0.85	0.9	0.95	1.0	

Figure 13: KBP for soluble proteins produced by the Neighbor Vector algorithm. Each row shows the potential for an amino acid (labeled on left and right). Each column represents the magnitude of the neighbor vector (labeled on top and bottom). Red represents high energy while blue represents low energy.



	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	0.55	0.6	0.65	0.7	0.75	0.8	0.85	0.9	0.95	1.0	
A	-2.22	-0.88	-0.37	-0.17	-0.14	-0.23	-0.09	-0.05	0.39	1.04	2.18	3.96	8.12	8.12	8.12	8.12	8.12	8.12	8.12	8.12	A
R	-1.07	-1.01	-0.93	-0.96	-0.96	-0.92	-0.57	-0.20	0.44	1.11	2.39	4.17	7.61	7.61	7.61	7.61	7.61	7.61	7.61	7.61	R
N	-1.42	-0.76	-0.64	-0.65	-0.68	-0.77	-0.67	-0.49	-0.18	0.13	1.96	5.23	7.43	7.43	7.43	7.43	7.43	7.43	7.43	7.43	N
D	-1.01	-0.72	-0.57	-0.64	-0.77	-0.93	-0.84	-0.67	-0.33	-0.01	1.70	4.92	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	D
C	-2.51	-1.22	-0.47	0.00	0.39	0.82	1.28	2.09	2.47	3.06	4.08	4.67	6.28	6.28	6.28	6.28	6.28	6.28	6.28	6.28	C
Q	-1.25	-0.80	-0.68	-0.73	-0.89	-0.95	-0.74	-0.46	0.07	0.73	2.10	4.54	7.32	7.32	7.32	7.32	7.32	7.32	7.32	7.32	Q
E	-0.69	-0.49	-0.50	-0.64	-0.86	-1.08	-0.99	-0.83	-0.40	0.16	1.74	4.09	7.90	7.90	7.90	7.90	7.90	7.90	7.90	7.90	E
G	-1.92	-0.71	-0.40	-0.28	-0.31	-0.42	-0.48	-0.44	-0.27	0.26	1.76	4.25	7.99	7.99	7.99	7.99	7.99	7.99	7.99	7.99	G
H	-1.72	-1.25	-0.90	-0.70	-0.57	-0.40	-0.09	0.15	0.49	0.76	2.22	3.31	6.81	6.81	6.81	6.81	6.81	6.81	6.81	6.81	H
I	-2.43	-1.17	-0.58	-0.22	0.21	0.38	0.85	1.49	2.09	2.52	3.58	5.08	7.72	7.72	7.72	7.72	7.72	7.72	7.72	7.72	I
L	-2.33	-1.23	-0.72	-0.35	-0.05	0.32	0.77	1.35	1.81	2.35	3.47	5.47	8.18	8.18	8.18	8.18	8.18	8.18	8.18	8.18	L
K	-0.33	-0.57	-0.68	-0.89	-1.00	-1.08	-0.95	-0.69	-0.19	0.27	1.67	4.09	7.75	7.75	7.75	7.75	7.75	7.75	7.75	7.75	K
M	-2.29	-1.10	-0.62	-0.34	-0.17	0.11	0.46	0.83	1.14	1.52	2.05	3.38	6.81	6.81	6.81	6.81	6.81	6.81	6.81	6.81	M
F	-2.35	-1.32	-0.77	-0.25	0.13	0.51	1.01	1.40	1.84	2.44	3.40	5.43	7.37	7.37	7.37	7.37	7.37	7.37	7.37	7.37	F
P	-1.48	-0.79	-0.57	-0.51	-0.48	-0.58	-0.62	-0.57	-0.45	-0.13	1.51	3.54	7.51	7.51	7.51	7.51	7.51	7.51	7.51	7.51	P
S	-1.77	-0.77	-0.56	-0.43	-0.49	-0.63	-0.58	-0.43	0.01	0.40	1.82	3.55	7.74	7.74	7.74	7.74	7.74	7.74	7.74	7.74	S
T	-1.79	-0.92	-0.68	-0.61	-0.59	-0.61	-0.55	-0.14	0.45	0.89	2.29	4.11	7.70	7.70	7.70	7.70	7.70	7.70	7.70	7.70	T
W	-2.15	-1.51	-0.89	-0.44	0.00	0.23	0.72	1.25	1.77	2.53	3.40	5.20	6.30	6.30	6.30	6.30	6.30	6.30	6.30	6.30	W
Y	-2.15	-1.46	-0.92	-0.48	-0.03	0.27	0.66	1.16	1.67	2.21	3.52	5.63	7.24	7.24	7.24	7.24	7.24	7.24	7.24	7.24	Y
V	-2.41	-1.03	-0.53	-0.22	-0.01	0.14	0.58	1.13	1.57	2.18	3.15	5.07	7.96	7.96	7.96	7.96	7.96	7.96	7.96	7.96	V
	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	0.55	0.6	0.65	0.7	0.75	0.8	0.85	0.9	0.95	1.0	

Figure 14: KBP for soluble proteins produced by the ANN algorithm. Each row shows the potential for an amino acid (labeled on left and right). Each column represents the SAE predicted by the ANN (labeled on top and bottom). Red represents high energy while blue represents low energy.



	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	0.55	0.6	0.65	0.7	0.75	0.8	0.85	0.9	0.95	1.0	
A	-2.41	-0.49	-0.15	-0.05	0.07	-0.15	0.30	0.22	0.59	1.06	1.63	1.97	3.17	3.90	3.79	4.36	7.43	8.12	8.12	8.12	A
R	-1.54	-0.98	-0.95	-0.84	-0.78	-0.71	-0.23	0.04	0.58	1.09	1.69	2.02	3.16	3.84	4.52	4.56	7.61	7.61	7.61	7.61	R
N	-1.73	-0.74	-0.60	-0.53	-0.56	-0.60	-0.34	-0.20	0.05	0.41	0.68	1.19	2.58	4.00	5.03	5.48	7.43	7.43	7.43	7.43	N
D	-1.40	-0.69	-0.59	-0.63	-0.68	-0.79	-0.46	-0.43	-0.08	0.23	0.50	1.09	2.33	3.52	4.45	5.27	7.75	7.75	7.75	7.75	D
C	-2.72	-0.66	-0.04	0.44	0.88	1.10	1.80	2.20	2.62	3.23	3.79	3.71	4.89	6.28	4.49	6.28	6.28	6.28	6.28	6.28	C
Q	-1.61	-0.79	-0.63	-0.71	-0.78	-0.80	-0.34	-0.21	0.21	0.81	1.21	1.76	2.92	3.40	4.32	5.12	6.62	7.32	7.32	7.32	Q
E	-1.12	-0.56	-0.55	-0.65	-0.82	-0.97	-0.59	-0.64	-0.15	0.22	0.70	1.17	2.55	3.82	3.84	4.76	7.90	7.90	7.90	7.90	E
G	-2.13	-0.49	-0.23	-0.17	-0.15	-0.24	-0.20	-0.19	-0.02	0.22	0.70	1.51	2.92	4.12	3.91	4.59	6.89	7.99	7.29	7.99	G
H	-2.09	-1.06	-0.68	-0.56	-0.24	-0.15	0.19	0.46	0.65	1.14	1.22	1.83	3.00	3.48	3.44	4.33	6.12	6.81	6.81	6.81	H
I	-2.62	-0.80	-0.24	0.15	0.48	0.67	1.23	1.71	2.23	2.66	3.13	3.49	4.78	5.08	5.42	5.52	7.72	7.72	7.72	7.72	I
L	-2.56	-0.90	-0.44	0.04	0.32	0.59	1.17	1.52	2.05	2.50	2.79	3.44	4.65	5.04	5.00	6.10	8.18	8.18	8.18	8.18	L
K	-0.89	-0.74	-0.82	-0.86	-0.87	-0.93	-0.58	-0.45	-0.04	0.48	0.79	1.27	2.26	3.57	3.90	4.61	7.06	7.75	7.75	7.75	K
M	-2.48	-0.82	-0.34	-0.17	0.11	0.37	0.74	1.10	1.49	1.81	1.94	2.49	2.84	3.35	3.01	4.25	6.81	6.81	6.81	6.81	M
F	-2.58	-0.98	-0.36	0.18	0.49	0.86	1.29	1.53	1.93	2.78	3.02	3.35	4.48	4.67	5.30	5.77	6.68	7.37	7.37	7.37	F
P	-1.80	-0.63	-0.48	-0.36	-0.37	-0.34	-0.34	-0.31	-0.17	0.01	0.41	0.96	2.03	3.66	3.75	4.29	6.82	7.51	7.51	7.51	P
S	-2.02	-0.64	-0.44	-0.35	-0.38	-0.50	-0.24	-0.13	0.30	0.56	0.89	1.49	2.48	3.83	3.66	4.03	7.74	7.74	7.74	7.74	S
T	-2.05	-0.78	-0.60	-0.48	-0.44	-0.43	-0.21	0.11	0.63	1.06	1.37	2.05	2.92	4.03	3.96	4.75	7.00	7.70	7.70	7.70	T
W	-2.48	-1.10	-0.56	-0.02	0.27	0.49	1.05	1.41	1.98	2.34	2.74	3.73	4.35	4.50	5.60	5.20	6.30	6.30	6.30	6.30	W
Y	-2.47	-1.14	-0.53	-0.01	0.30	0.53	1.07	1.35	1.76	2.42	2.79	3.23	4.02	5.29	5.85	5.44	7.24	7.24	7.24	7.24	Y
V	-2.58	-0.70	-0.24	0.02	0.26	0.38	0.92	1.36	1.85	2.21	2.67	3.28	4.20	4.63	4.87	5.56	7.96	7.96	7.96	7.96	V
	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	0.55	0.6	0.65	0.7	0.75	0.8	0.85	0.9	0.95	1.0	

Figure 15: KBP for soluble proteins produced by the OLS algorithm. Each row shows the potential for an amino acid (labeled on left and right). Each column represents the SAE as given by the OLS algorithm (labeled on top and bottom). Red represents high energy while blue represents low energy.

Benchmark Proteins

In order to determine the effectiveness of the knowledge-based potentials, their ability to distinguish between native-like and nonnative-like protein models is evaluated. In other words, are they able to tell that the good models are in fact good, and that the bad models are in fact bad? Nineteen benchmark proteins were selected for analysis. Many protein models (i.e. random placements of the secondary structural elements) were selected for each benchmark protein.

Whereas root mean square distance (*rmsd*) is a metric commonly used to quantify the degree of similarity between two structures, a normalized form of *rmsd* called *rmsd*100 that is independent of the number of amino acids in the protein[64] is used in this analysis. *Rmsd*100 is considered a more robust means of structural comparison.

$$rmsd100 = \frac{\sqrt{\frac{\sum_i d_i^2}{n}}}{1 + \ln\sqrt{\frac{n}{100}}}$$

The higher the *rmsd*100, the greater is the deviation from the native structure. The term native-like is used to describe protein models having an *rmsd*100 less than 5 Å whereas protein models having an *rmsd*100 greater than or equal to 5 Å are described as nonnative-like.

Protein models were selected such that the percentage of proteins with an *rmsd*100 below 5 Å constitute 10% of the decoy set. This is similar to the distribution of native-like and nonnative-like models that are generated during a MC folding program.

Classification, Architecture, Topology, and Homologous superfamily (CATH) is a hierarchical classification of protein domain structures[65]. In order to examine the performance on a variety of protein domains, proteins from multiple CATH classifications were selected (shown in Table 6). Additionally, models of various sizes were selected to ensure a robust benchmark set (also shown in Table 6).

Table 6: Benchmark proteins.

PDB ID	CATH classification	# amino acids	# models with *rmsd*100 < 5 Å	# models
1ail	mainly alpha	70	11	110
1e6i	mainly alpha	136	7	70
1enh	mainly alpha	54	48	480
1r69	mainly alpha	69	11	110
1a19	alpha beta	180	57	570
1iib	alpha beta	212	68	680
1acf	alpha beta	125	103	1030
1bm8	alpha beta	99	72	720
1cc8	alpha beta	73	71	710
1ctf	alpha beta	74	90	900
1hz6	alpha beta	216	45	450
1opd	alpha beta	85	95	950
1tig	alpha beta	94	17	170
1b3a	mainly beta	134	64	640
1bq9	mainly beta	54	12	120
1c9o	mainly beta	132	49	490
1fna	mainly beta	91	67	670
1shf	mainly beta	118	13	130
1scj	alpha beta	346	11	110

Enrichment Analysis

In order to determine how well the KBPs distinguish between native-like and nonnative-like protein models, the metric enrichment is defined as follows:

$$enrichment = \frac{\left(\frac{\#\ models\ in\ lowest\ 10\%\ of\ energy\ score\ with\ rmsd100 < 5\ \text{Å}}{\#\ models\ with\ rmsd100 < 5\ \text{Å}}\right)}{percentage\ of\ models\ with\ rmsd100 < 5\ \text{Å}}$$

Enrichment is a measure of how well the KBP identifies native-like models as good models by assigning them a favorable energy score. The maximum enrichment possible is 10.0 and would occur if all of the models in the lowest 10% of energy scores have an *rmsd*100 less than 5 Å (i.e. it identified all of the native-like models as energetically favorable (low energy)) and all models in the highest 90% of energy scores have an *rmsd*100 greater than 5 Å (i.e. it identified all of the nonnative-like models as energetically unfavorable (high energy)). A random enrichment of 1.0 is expected. Therefore, an enrichment greater than 1.0 is better than random.

Table 7: Average enrichment over benchmark proteins.

algorithm	**weighted average enrichment ± weighted standard deviation**
MSMS	3.49 ± 2.99
Neighbor Count	2.69 ± 2.69
Neighbor Vector	3.07 ± 2.63
ANN	3.44 ± 3.00
OLS	3.30 ± 2.87

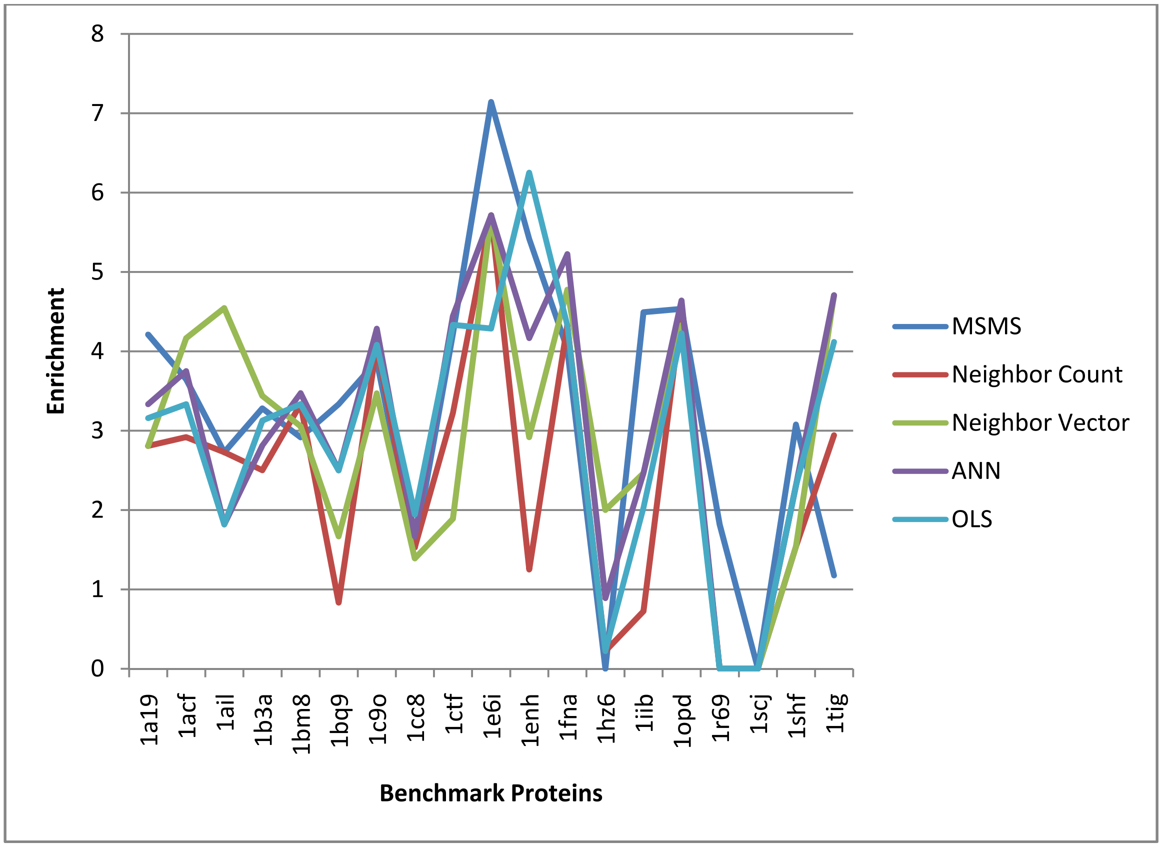


Figure 16: Enrichment over benchmark proteins.

Evaluating the Energy of Benchmark Proteins

A more detailed analysis of KBPs is presented for selected representative benchmark proteins.

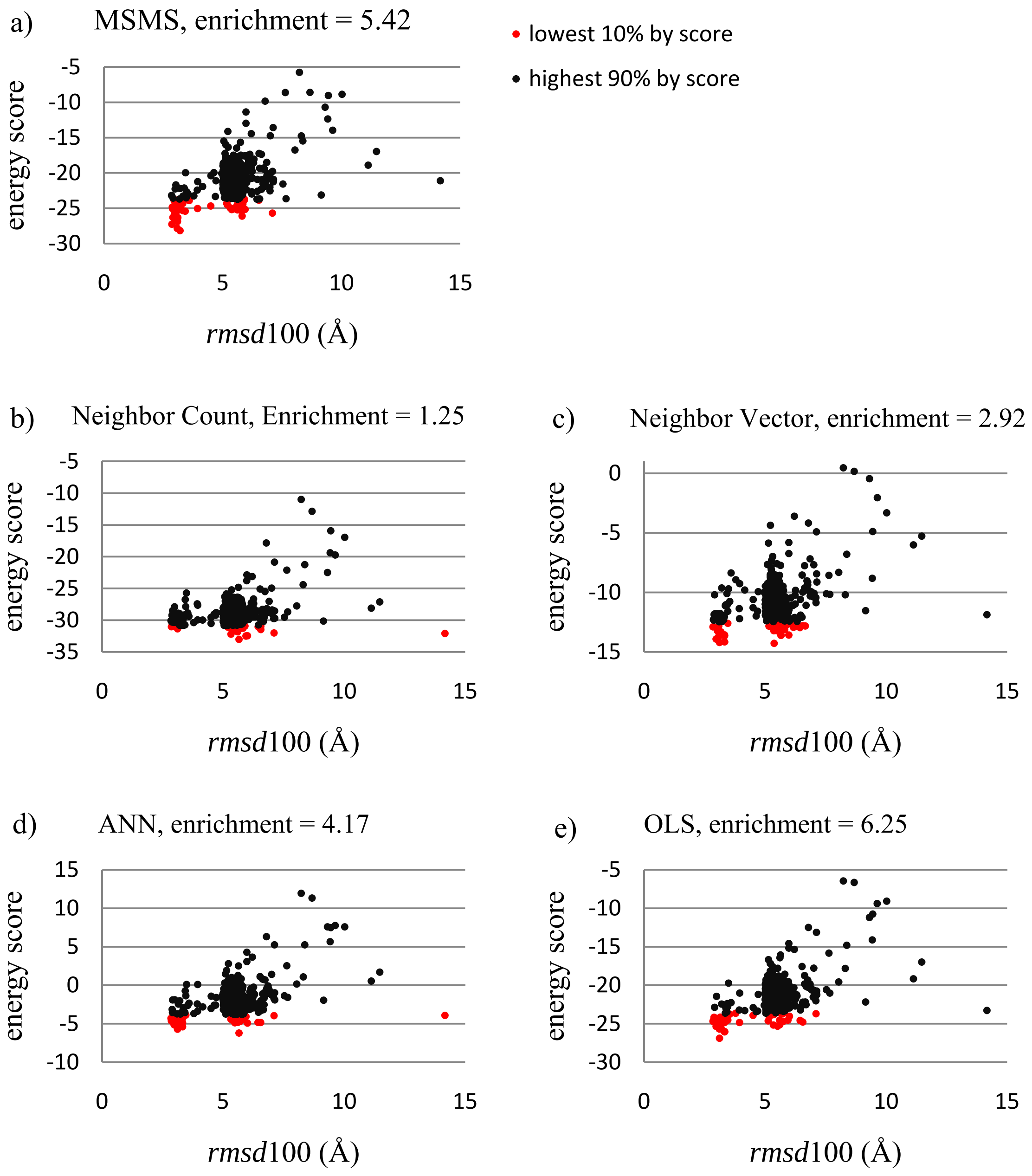


Figure 17: Energy scores for 1enh decoys.

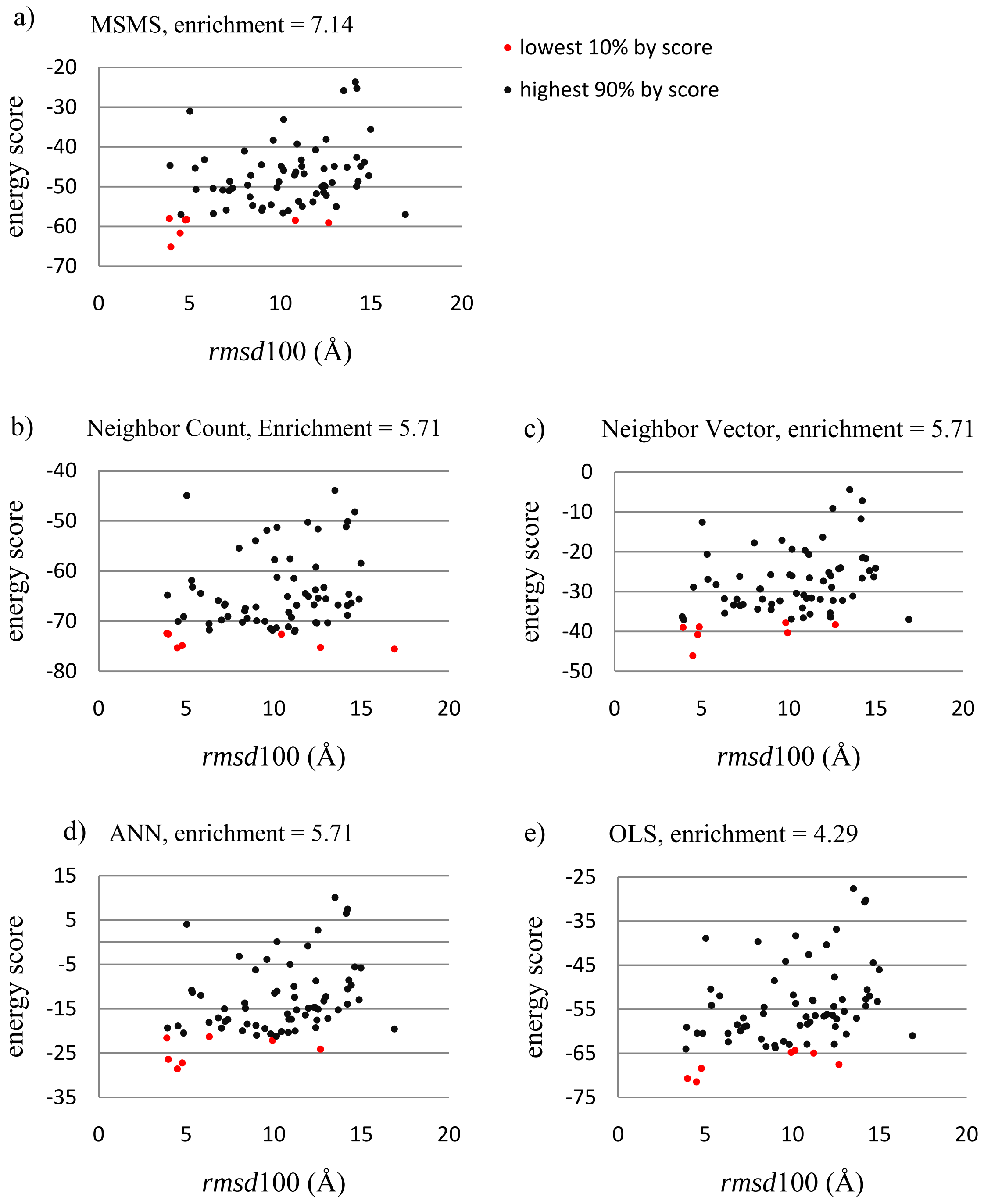


Figure 18: Energy scores for 1e6i decoys.

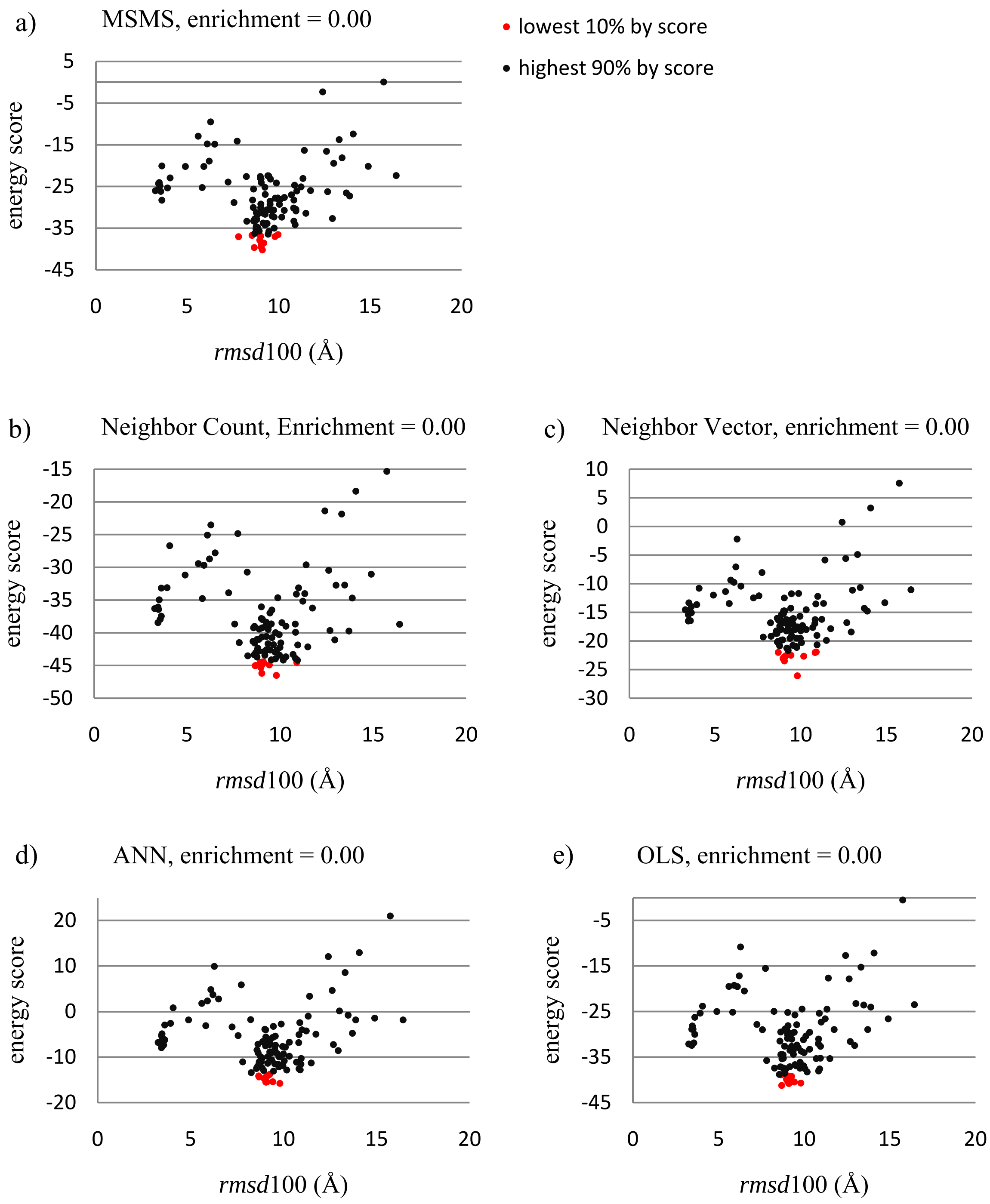


Figure 19: Energy scores for 1scj decoys.

CHAPTER VII

DISCUSSION

Neighbor Count is the simplest measure of SAE and, as expected, achieves the lowest average enrichment. Also as expected, as the algorithms increase in complexity, they are able to achieve a higher enrichment. In other words, they are more effectively able to distinguish between native-like and nonnative-like protein models. The ANN is particularly effective at this task and achieves enrichments on reduced-complexity models that are nearly as high as the enrichments achieved by the high-resolution MSMS algorithm on full-atom models.

As is seen in Figure 16 and indicated by the large standard deviations shown in Table 7, the degree to which the algorithms are able to recognize native-like protein varies widely. Consider the high enrichments produced for the protein 1e6i. In this case, the algorithms are fairly effectively able to distinguish between native-like and nonnative-like protein models. (See Figure 18.) However, there are proteins that are "hard," for example 1scj. All algorithms produced an enrichment of 0.0. (See Figure 19.)

In all cases, the maximum possible enrichment of 10.0 was not achieved by any algorithm, including the reference standard MSMS. This indicates that the environment free energy contains a limited amount of information and additional energy terms should be considered in order to achieve maximum enrichment. This is in fact the case and multiple additional energy evaluation functions are implemented in the Meiler Lab and are used in evaluating the energy of models generated throughout the course of a MC folding run.

CHAPTER IIX

CONCLUSION

Summary

Four SAE approximation algorithms of varying complexities are presented. The least complex algorithms are the most efficient in terms of runtime; however, they are the most limited in terms of their ability to distinguish between native-like and nonnative-like protein models.

The ANN is able to distinguish between native-like and nonnative-like protein models of reduced complexity very quickly yet nearly as well as the high-resolution reference standard MSMS.

Study Limitations

Further study is needed to determine how effective the algorithms and KBPs are for membrane proteins.

Future Work

Experiments have indicated that the backbone atoms are not as essential in determining SAE as are the side chains of other amino acids. These backbone atoms can produce noise and obstruct the signal in determining exposure. One way to increase the signal-to-noise ratio is to extend the C_β atom further into space away from the backbone. Experiments are currently being conducted to determine if this enhancement increases the ability of the SAE algorithms to distinguish between native-like and nonnative-like proteins.

BIBLIOGRAPHY

1. Crick FH. On protein synthesis. Symp Soc Exp Biol 1958;12:138-163.

2. Feitelson DG, Treinin M. The blueprint for life? Computer 2002;35(7):34-+.

3. Baker D, Sali A. Protein structure prediction and structural genomics. Science 2001;294(5540):93-96.

4. Wiener MC. A pedestrian guide to membrane protein crystallization. Methods 2004;34(3):364-372.

5. Fang Y, Frutos AG, Lahiri J. Membrane protein microarrays. J Am Chem Soc 2002;124(11):2394-2395.

6. Levintha.C. Are There Pathways for Protein Folding. Journal De Chimie Physique Et De Physico-Chimie Biologique 1968;65(1):44-&.

7. White SH, Wimley WC. Membrane protein folding and stability: Physical principles. Annual Review of Biophysics and Biomolecular Structure 1999;28:319-365.

8. Chandonia JM, Karplus M. New methods for accurate prediction of protein secondary structure. Proteins 1999;35(3):293-306.

9. Guermeur Y, Geourjon C, Gallinari P, Deleage G. Improved performance in protein secondary structure prediction by inhomogeneous score combination. Bioinformatics 1999;15(5):413-421.

10. Karplus K, Sjolander K, Barrett C, Cline M, Haussler D, Hughey R, Holm L, Sander C. Predicting protein structure using hidden Markov models. Proteins 1997;Suppl 1:134-139.

11. Meiler J, Baker D. Coupled prediction of protein secondary and tertiary structure. Proc Natl Acad Sci U S A 2003;100(21):12105-12110.

12. Meiler J, Muller M, Zeidler A, Schmaschke F. Generation and evaluation of dimension-reduced amino acid parameter representations by artificial neural networks. Journal of Molecular Modeling 2001;7(9):360-369.

13. Rost B. PHD: predicting one-dimensional protein structure by profile-based neural networks. Methods Enzymol 1996;266:525-539.

14. Rost B, Sander C. Improved prediction of protein secondary structure by use of sequence profiles and neural networks. Proc Natl Acad Sci U S A 1993;90(16):7558-7562.

15. Rost B, Sander C. Prediction of protein secondary structure at better than 70% accuracy. J Mol Biol 1993;232(2):584-599.

16. Ward JJ, McGuffin LJ, Buxton BF, Jones DT. Secondary structure prediction with support vector machines. Bioinformatics 2003;19(13):1650-1655.

17. Kuhn M, Meiler J, Baker D. Strand-loop-strand motifs: Prediction of hairpins and diverging turns in proteins. Proteins-Structure Function and Genetics 2004;54(2):282-288.

18. Jones DT, Ward JJ. Prediction of disordered regions in proteins from position specific score matrices. Proteins-Structure Function and Genetics 2003;53(6):573-578.

19. Linding R, Jensen LJ, Diella F, Bork P, Gibson TJ, Russell RB. Protein disorder prediction: Implications for structural proteomics. Structure 2003;11(11):1453-1459.

20. Grana O, Baker D, MacCallum RM, Meiler J, Punta M, Rost B, Tress ML, Valencia A. CASP6 assessment of contact prediction. Proteins 2005;61 Suppl 7:214-224.

21. Galzitskaya OV, Melnik BS. Prediction of protein domain boundaries from sequence alone. Protein Sci 2003;12(4):696-701.

22. Liu J, Rost B. Comparing function and structure between entire proteomes. Protein Sci 2001;10(10):1970-1979.

23. Ben-Hur A, Noble WS. Kernel methods for predicting protein-protein interactions. Bioinformatics 2005;21 Suppl 1:i38-46.

24. Valencia A, Pazos F. Computational methods for the prediction of protein interactions. Curr Opin Struct Biol 2002;12(3):368-373.

25. Metropolis N, Teller E. Equation of State Calculations by Fast Computing Machines. J Chem Phys 1953;21(6):1087.

26. Korostelev A, Bertram R, Chapman M. Simulated-annealing real-space refinement as a tool in model building. Acta Crystallogr D Biol Crystallogr 2002;58(5):7617.

27. Pokala N, Handel TM. Energy functions for protein design I: efficient and accurate continuum electrostatics and solvation. Protein Sci 2004;13(4):925-936.

28. Casadio R, Fariselli P, Martelli PL, Tasco G. Thinking the impossible: how to solve the protein folding problem with and without homologous structures and more. Methods Mol Biol 2007;350:305-320.

29. Chen CT, Lin HN, Sung TY, Hsu WL. HYPLOSP: a knowledge-based approach to protein local structure prediction. J Bioinform Comput Biol 2006;4(6):1287-1307.

30. Ferrada E, Melo F. Nonbonded terms extrapolated from nonlocal knowledge-based energy functions improve error detection in near-native protein structure models. Protein Sci 2007;16(7):1410-1421.

31. Lu H, Skolnick J. Application of statistical potentials to protein structure refinement from low resolution ab initio models. Biopolymers 2003;70(4):575-584.

32. Yarov-Yarovoy V, Schonbrun J, Baker D. Multipass membrane protein structure prediction using Rosetta. Proteins 2006;62(4):1010-1025.

33. Audie J, Scarlata S. A novel empirical free energy function that explains and predicts protein-protein binding affinities. Biophys Chem 2007;129(2-3):198-211.

34. Darnell SJ, Page D, Mitchell JC. An automated decision-tree approach to predicting protein interaction hot spots. Proteins 2007;68(4):813-823.

35. Evers A, Gohlke H, Klebe G. Ligand-supported homology modelling of protein binding-sites using knowledge-based potentials. J Mol Biol 2003;334(2):327-345.

36. Evers A, Klebe G. Ligand-supported homology modeling of g-protein-coupled receptor sites: models sufficient for successful virtual screening. Angew Chem Int Ed Engl 2004;43(2):248-251.

37. Gohlke H, Hendlich M, Klebe G. Knowledge-based scoring function to predict protein-ligand interactions. J Mol Biol 2000;295(2):337-356.

38. Grzybowski BA, Ishchenko AV, Shimada J, Shakhnovich EI. From knowledge-based potentials to combinatorial lead design in silico. Acc Chem Res 2002;35(5):261-269.

39. Roche O, Kiyama R, Brooks CL, 3rd. Ligand-protein database: linking protein-ligand complex structures to binding data. J Med Chem 2001;44(22):3592-3598.

40. Isogai Y, Ito Y, Ikeya T, Shiro Y, Ota M. Design of lambda Cro fold: solution structure of a monomeric variant of the de novo protein. J Mol Biol 2005;354(4):801-814.

41. Poole AM, Ranganathan R. Knowledge-based potentials in protein design. Curr Opin Struct Biol 2006;16(4):508-513.

42. Gordon DB, Marshall SA, Mayo SL. Energy functions for protein design. Curr Opin Struct Biol 1999;9(4):509-513.

43. Ooi T, Oobatake M, Nemethy G, Scheraga HA. Accessible surface areas as a measure of the thermodynamic parameters of hydration of peptides. Proc Natl Acad Sci U S A 1987;84(10):3086-3090.

44. Vizcarra CL, Mayo SL. Electrostatics in computational protein design. Curr Opin Chem Biol 2005;9(6):622-626.

45. Macdonald JR, Johnson WC, Jr. Environmental features are important in determining protein secondary structure. Protein Sci 2001;10(6):1172-1177.

46. Kinoshita M, Okamoto Y, Hirata F. First-principle determination of peptide conformations in solvents: Combination of Monte Carlo simulated annealing and RISM theory. Journal of the American Chemical Society 1998;120(8):1855-1863.

47. Wodak SJ, Janin J. Analytical approximation to the accessible surface area of proteins. Proc Natl Acad Sci U S A 1980;77(4):1736-1740.

48. Lee B, Richards FM. The interpretation of protein structures: estimation of static accessibility. J Mol Biol 1971;55(3):379-400.

49. Shrake A, Rupley JA. Environment and exposure to solvent of protein atoms. Lysozyme and insulin. J Mol Biol 1973;79(2):351-371.

50. Connolly ML. Analytical molecular surface calculation. J Appl Cryst 1983;16:548-558.

51. Richmond TJ, Richards FM. Packing of alpha-helices: geometrical constraints and contact areas. J Mol Biol 1978;119(4):537-555.

52. Connolly ML. Solvent-accessible surfaces of proteins and nucleic acids. Science 1983;221(4612):709-713.

53. Pearl LH, Honegger A. Generation of molecular surfaces for graphic display. J Mol Graph 1983;1(1):9-12.

54. You T, Bashford D. An analytical algorithm for the rapid determination of the solvent-accessibility of points in a three-dimensional lattice around a solute molecule. J Comp Chem 1994;16(6):743-757.

55. Colloc'h N, J-P. M. A new tool for the qualitative and quantitative analysis of protein surfaces using B-spline and density of surface neighborhood. J Mol Graph 1990;8:133-140.

56. Street AG, Mayo SL. Pairwise calculation of protein solvent-accessible surface areas. Fold Des 1998;3(4):253-258.

57. Sanner MF, Olson AJ, Spehner JC. Reduced surface: An efficient way to compute molecular surfaces. Biopolymers 1996;38(3):305-320.

58. Simons KT, Kooperberg C, Huang E, Baker D. Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions. J Mol Biol 1997;268(1):209-225.

59. Humphrey W, Dalke A, Schulten K. VMD: visual molecular dynamics. J Mol Graph 1996;14(1):33-38, 27-38.

60. Berman HM, Battistuz T, Bhat TN, Bluhm WF, Bourne PE, Burkhardt K, Feng Z, Gilliland GL, Iype L, Jain S, Fagan P, Marvin J, Padilla D, Ravichandran V, Schneider B, Thanki N, Weissig H, Westbrook JD, Zardecki C. The Protein Data Bank. Acta Crystallogr D Biol Crystallogr 2002;58(Pt 6 No 1):899-907.

61. Bernstein FC, Koetzle TF, Williams GJ, Meyer EF, Jr., Brice MD, Rodgers JR, Kennard O, Shimanouchi T, Tasumi M. The Protein Data Bank. A computer-based archival file for macromolecular structures. Eur J Biochem 1977;80(2):319-324.

62. Dunbrack RL, Jr. Rotamer libraries in the 21st century. Curr Opin Struct Biol 2002;12(4):431-440.

63. Dunbrack RL, Jr., Karplus M. Backbone-dependent rotamer library for proteins. Application to side-chain prediction. J Mol Biol 1993;230(2):543-574.

64. Carugo O, Pongor S. A normalized root-mean-square distance for comparing protein three-dimensional structures. Protein Sci 2001;10(7):1470-1473.

65. Orengo CA, Michie AD, Jones S, Jones DT, Swindells MB, Thornton JM. CATH--a hierarchic classification of protein domain structures. Structure 1997;5(8):1093-1108.