Title
Inferring interactions and functions of proteins using evolutionary information

Permalink
https://escholarship.org/uc/item/91x95301

Author
Kim, Yohan

Publication Date
2007

Peer reviewed|Thesis/dissertation
UNIVERSITY OF CALIFORNIA, SAN DIEGO

Inferring interactions and functions of proteins using evolutionary information

A dissertation submitted in partial satisfaction of the requirements for the degree Doctor of Philosophy

in

Chemistry

by

Yohan Kim

Committee in charge:
Professor Shankar Subramaniam, Co-chair
Professor Douglas Magde, Co-chair
Professor Philip E. Bourne
Professor Alexander Hoffmann
Professor Susan S. Taylor

2007
The dissertation of Yohan Kim is approved, and it is acceptable in quality and form for publication on microfilm:

Co-chair

Co-chair

University of California, San Diego

2007
TABLE OF CONTENTS

Signature Page ........................................................................................................ iii
Table of Contents ........................................................................................................ iv
List of Figures ............................................................................................................. vi
List of Tables ................................................................................................................ vii
Acknowledgments ....................................................................................................... vii
Vita, Publications, and Fields of Study ....................................................................... ix
Abstract ...................................................................................................................... x

I Introduction ............................................................................................................... 1
   A. Domain co-evolution ............................................................................................ 2
   B. Local backbone structures .................................................................................. 2
   C. Protein design ...................................................................................................... 2
   D. Aligning protein-protein interaction networks .................................................. 3

II Inferring interactions and functions of proteins from domain co-evolution ........ 4
   A. Introduction ........................................................................................................ 4
   B. Computational Methods and Algorithms ............................................................ 5
      1. Constructing Phylogenetic Profiles .................................................................. 6
      2. Computing Coevolutionary Matrices .............................................................. 10
      3. Deriving Phylogenetic Similarity Scores ......................................................... 11
   C. Results ................................................................................................................ 13
      1. Comparison of Coevolutionary-Matrix and Single-Profile Methods ............... 14
      2. Examples of Domain Co-evolution .................................................................. 20
   D. Discussion ........................................................................................................... 27
   E. Concluding Remarks ............................................................................................ 28

III Energy function derived from statistical analysis of backbone dihedral angles
   of proteins .................................................................................................................. 29
   A. Introduction ........................................................................................................ 29
   B. Methods ............................................................................................................... 30
      1. Energy Functions Tested .................................................................................. 31
      2. Multi-Dimensional Scaling ............................................................................... 34
      3. Threading Experiments ................................................................................... 34
   C. Results ................................................................................................................ 38
      1. Distribution of shapes of mono-peptide and tri-peptide specific PDFs .......... 38
      2. Threading Experiments ................................................................................... 39
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>II.1</td>
<td>Binary phylogenetic profiles</td>
<td>6</td>
</tr>
<tr>
<td>II.2</td>
<td>Limitations of the Single-Profile method</td>
<td>8</td>
</tr>
<tr>
<td>II.3</td>
<td>Limitations of the Multiple-Profile method</td>
<td>9</td>
</tr>
<tr>
<td>II.4</td>
<td>Algorithm for the Coevolutionary-Matrix method</td>
<td>12</td>
</tr>
<tr>
<td>II.5</td>
<td>Positive Predictive Value (PPV) vs. Coverage</td>
<td>17</td>
</tr>
<tr>
<td>II.6</td>
<td>Co-evolutionary score distributions</td>
<td>18</td>
</tr>
<tr>
<td>II.7</td>
<td>P-values of scores for the set of protein pairs known to interact</td>
<td>19</td>
</tr>
<tr>
<td>II.8</td>
<td>Coevolutionary matrix for the PTS proteins</td>
<td>22</td>
</tr>
<tr>
<td>II.9</td>
<td>Coevolutionary matrix for the chemotaxis proteins</td>
<td>24</td>
</tr>
<tr>
<td>II.10</td>
<td>Coevolutionary-matrix for the Kdp proteins</td>
<td>26</td>
</tr>
<tr>
<td>III.1</td>
<td>Normalized energies with varying number of random sequences</td>
<td>36</td>
</tr>
<tr>
<td>III.2</td>
<td>Distribution of normalized energies</td>
<td>37</td>
</tr>
<tr>
<td>III.3</td>
<td>A map of shapes of tri-peptide specific PDFs</td>
<td>40</td>
</tr>
<tr>
<td>III.4</td>
<td>A sample of PDF shapes</td>
<td>41</td>
</tr>
<tr>
<td>III.5</td>
<td>A map of PDFs for the five set of tri-peptides</td>
<td>42</td>
</tr>
<tr>
<td>III.6</td>
<td>The effect of residue position on the backbone angles</td>
<td>43</td>
</tr>
<tr>
<td>III.7</td>
<td>Ungapped threading results</td>
<td>45</td>
</tr>
<tr>
<td>III.8</td>
<td>Threading of distant homologues</td>
<td>46</td>
</tr>
<tr>
<td>III.9</td>
<td>Fractions of foldable sequences vs. sequence lengths</td>
<td>48</td>
</tr>
<tr>
<td>III.10</td>
<td>Example proteins with high/low values of $-\log(P(x))$</td>
<td>49</td>
</tr>
<tr>
<td>IV.1</td>
<td>Superimposed structural homologues</td>
<td>60</td>
</tr>
<tr>
<td>IV.2</td>
<td>Conservation profile for $\alpha$-amylase</td>
<td>68</td>
</tr>
<tr>
<td>IV.3</td>
<td>Conservation profile for RRM</td>
<td>69</td>
</tr>
<tr>
<td>IV.4</td>
<td>Conservation profile for SH3</td>
<td>70</td>
</tr>
<tr>
<td>V.1</td>
<td>Sample execution of MULE</td>
<td>75</td>
</tr>
<tr>
<td>V.2</td>
<td>Interaction patterns for TFIID</td>
<td>76</td>
</tr>
<tr>
<td>V.3</td>
<td>Small nuclear ribonucleoprotein complex</td>
<td>77</td>
</tr>
<tr>
<td>V.4</td>
<td>Actin-related protein Arp2/3 complex</td>
<td>77</td>
</tr>
</tbody>
</table>
LIST OF TABLES

Table II.1: Number of predicted interactions at various thresholds. . . . . . . 16
Table II.2: Number of overlapping true interactions between the methods. 17
Table II.3: Top 20 interacting partners of a PTS protein. . . . . . . . . . . . 23
Table II.4: Top 20 interacting partners of a chemotaxix protein. . . . . . . 25
Table II.5: Top 10 interacting partners of a Kdp protein. . . . . . . . . . . . 26

Table IV.1: Information on the structural homologues. . . . . . . . . . . . . . . . . 60
Table IV.2: Protein design results for α-amylase. . . . . . . . . . . . . . . . . . 61
Table IV.3: Protein design results for RRM. . . . . . . . . . . . . . . . . . . . . . 62
Table IV.4: Protein design results for SH3 domain. . . . . . . . . . . . . . . . 63
Table IV.5: Fold recognition results for α-amylase. . . . . . . . . . . . . . . . 65
Table IV.6: Fold recognition results for RRM. . . . . . . . . . . . . . . . . . . . . 66
Table IV.7: Fold recognition results for SH3. . . . . . . . . . . . . . . . . . . . . 67

Table V.1: Statistics of Mined PPI Networks. . . . . . . . . . . . . . . . . . . . . 76
ACKNOWLEDGMENTS

I would like to thank my parents, my brother, and friends for providing encouragements and support throughout these years. I am very grateful to Professor Shankar Subramaniam for allowing me to pursue various ideas in the field of structural biology and genomics in his laboratory. I also would like to thank Mehmet Koyutürk, Umut Topkara, and Professor Ananth Grama for being wonderful collaborators on a number of projects.

Section II, in full, is a reprint of the article "Inferring functional information from domain co-evolution" by Y. Kim, M. Koyutürk, U. Topkara, A. Grama, and S. Subramaniam, published in Bioinformatics, volume 22, pages 40-49, 2006. Y. Kim was the primary researcher and author for this work. This work has been reprinted with permissions from co-authors and Oxford University Press.

Section V uses some of the figures and table from the article "Detecting conserved interaction patterns in biological networks" by M. Koyutürk, Y. Kim, S. Subramaniam, W. Szpankowski, and A. Grama, published in J. Comput. Biol., volume 13, pages 1299-1322, 2006. Y. Kim contributed to the analysis of the frequently occurring interaction patterns detected by the method. This work has been reprinted with permissions from co-authors and Mary Ann Liebert Inc., publishers.
VITA

1996 Bachelor of Science, Emory University
2000–2001 Teaching Assistant, University of California at San Diego
2003 Master of Science, University of California at San Diego
2001–2007 Research Assistant, University of California at San Diego
2007 Doctor of Philosophy, University of California at San Diego

PUBLICATIONS

Kim Y. and Subramaniam S. Energy functions derived from statistics of local backbone dihedral angles of proteins and their applications in threading. (to be submitted)


FIELDS OF STUDY

Major Field: Chemistry
Studies in Computational Biology
Professor Shankar Subramaniam
ABSTRACT OF THE DISSERTATION

Inferring interactions and functions of proteins using evolutionary information

by

Yohan Kim
Doctor of Philosophy in Chemistry
University of California, San Diego, 2007
Professor Shankar Subramaniam, Co-chair
Professor Douglas Magde, Co-chair

The dissertation describes computational methods that can be used to predict structure, function, and interactions of proteins by exploiting the wealth of biological databases. The dissertation is divided into four sections. In the first section, a protein interaction prediction algorithm is developed that extends the original protein phylogenetic profiling algorithm. The new algorithm broadens the idea of co-evolution between two whole proteins to include that of their sub-regions. This extension results in identification of co-evolving protein domains that are validated to be interacting.

In the second section, the effects of nearest neighboring residues on the backbone-angle propensities of the middle residue are mapped out. An energy function incorporating these effects is tested using threading experiments, and its performance is compared with those of other widely used energy functions.

In the third section, the utility of using known structural homologues of a protein fold as a model for backbone flexibility in a protein design algorithm is investigated. Using the design algorithm, 100 protein sequences are designed for each of the structural
homologues. Multiple sequence alignments (MSAs) are then derived from the designed sequences for each structure as well as for the entire structural homologues of a protein family. These MSAs are then compared with those of naturally occurring sequences using fold recognition experiments and conservation profiles.

Finally, in the fourth section, a method is described that can be used to mine large-scale protein-protein interaction data. By searching protein-protein interaction networks of nine eukaryotic organisms for frequently occurring interaction patterns, previously known protein complexes and pathways are discovered.
I

Introduction

Genome sequencing marked a new era in biology by generating vast amounts of sequence data. Sequencing projects involving hundreds of species and a number of large-scale environmental samples have been completed, and more sequencing projects will likely to continue at faster rate in the future. Similarly, due to initiatives in structural genomics, the number of structures solved each year is also growing at faster rate. Additionally, new types of biological data such as protein-protein interaction networks of various organisms have been recently generated by high-throughput experiments. Thus developing computational methods that process a large quantity of data and that provide insights into their origin will become more important in biology.

In Chapter II, one such computational method is described that uses comparisons of complete genome sequences to predict protein-protein interactions on a genome-scale. In Chapter III, a large number of solved protein structures are used to develop energy functions based on the statistics of their local backbone angles. In Chapter IV, known structural homologues are used to evaluate a protein design algorithm. Finally, in Chapter V, a method is described that mines large protein-protein interaction networks for frequently occurring interaction patterns. Thus mining of the biological databases to develop better methods for predicting structure, function, and interactions of proteins is the unifying theme of this dissertation. In the following sections, motivations driving the development of each of these methods are presented.
I.A Domain co-evolution

One of the forces that may influence how one protein is interacting with another is co-evolutionary constraints. Specifically, two proteins necessary to carry out a function should be present together in a genome of an organism. However, if this organism later has no need for such function, then there is no evolutionary pressure to maintain the presence of the two proteins in its genome. By capturing such co-occurrences of proteins among genomes using phylogenetic profiles, Pellegrini et al. in 1999 demonstrated that co-evolution of two proteins can be measured and correlated with their likelihood of interactions.

Chapter II describes a method that broadens this idea to include co-evolution of protein domains. Genome-scale predictions of protein-protein interactions made for E. coli using the method are benchmarked against a set of known interacting proteins. In the chapter, it is demonstrated that co-evolution of domains can be captured by the method, resulting in detections of interactions missed by earlier approaches.

I.B Local backbone structures

Motivated by growing number of solved protein structures that are available, statistics involving nearest-neighboring residues and backbone angles, phi and psi, of their middle residues were collected and analyzed. Chapter III presents probability density functions (PDFs) derived from these backbone-angle statistics and a 2-dimensional map that provides insights into relationships between these PDFs. Energy functions based on these PDFs are also described, as well as their performances with respect to those of a number of widely used energy functions.

I.C Protein design

Chapter IV investigates how well a protein design algorithm can capture evolutionary forces shaping the sequence space of known protein families. Given a fixed
protein backbone, a protein design algorithm searches the sequence space to find compatible sequences. One way to test a protein design algorithm is to ask how similar are multiple sequence alignments (MSAs) of designed sequences to MSAs of known protein families. To answer this, fold recognition experiments using profiles derived from these MSAs are were carried out, and their results are presented in the chapter. In addition, conservation profiles of these MSAs are compared.

I.D Aligning protein-protein interaction networks

Soon after complete genome sequences of model organisms became available, results from high-throughput experimental characterizations of their protein interaction networks were reported, motivating the development of the method described in Chapter V. The method mines large protein interaction networks of different organisms for frequently occurring interaction patterns. The method is an extension of the idea of aligning sequences. This extension immediately implies that conserved interaction patterns among different organisms may have functional significance. Some of the detected frequently occurring patterns are presented and discussed with respect to known biological examples.
II

Inferring interactions and functions of proteins from domain co-evolution

II.A Introduction

Identifying interacting pairs of proteins encoded in a genome is an important step towards understanding how a cell works. Towards this eventual goal, several computational and experimental techniques have been developed in recent years. For instance, recent developments in high-throughput experiments yielded protein interaction data on a very large scale [2, 3, 4, 5, 6]. High-throughput methods, however, are prone to errors in terms of both false negatives and positives [7]. Hence, computational methods must be developed in parallel to complement experimental techniques. Indeed, integrating in-silico analysis with experimental information provides more comprehensive and reliable understanding of functional association between proteins [8].

Computational methods that predict protein interactions have gained impetus from recently available databases of complete genome sequences. Using genome data, researchers have inferred functions of numerous proteins by comparing genomes across species [9, 10, 11, 12]. One way of exploiting evolutionary pressure to understand function is quantifying the conservation of gene neighborhoods across genomes, which has been shown to correlate with their function [9, 11]. Another approach is comparing pro-
tein phylogenetic profiles, where each profile is a vector indicating presence or absence of a protein across genomes [10]. The similarity of two phylogenetic profiles, which captures the degree of co-evolution between the two corresponding proteins, has been shown to correlate with their functions [10]. Subsequent work has shown that many of the known pathways can be reconstructed using such methods [13, 14].

In earlier work, we presented a simple extension to a method based on protein phylogenetic profiles [15]. By taking into account the multi-domain nature of proteins, our method detected several known interactions missed by earlier methods. This extension, referred to as the Multiple-Profile method, simply partitioned a protein sequence into overlapping segments (e.g., 30 residues) of fixed length (e.g., 120 residues) and constructed separate phylogenetic profiles for each of these segments. Because large and fixed-length segments are used, boundaries of domains with different evolutionary histories cannot be cleanly resolved. Consequently it is difficult to assess whether these co-evolving segments correspond to true domains. In addition, there exists a possibility of introducing false phylogenetic profiles as artifacts of segmentation. This may occur when one of the segments covers two domains having different evolutionary histories.

This paper improves on existing techniques by using a novel method for identifying co-evolving regions precisely, thus reducing the number of false phylogenetic profiles. With this new tool, we show a number of examples from the E. coli proteome where the identified co-evolving regions correspond to biochemically characterized and functionally associated domains.

II.B Computational Methods and Algorithms

Our method, Coevolutionary-Matrix, is designed to assign phylogenetic similarity scores to each pair of proteins under consideration (e.g., all E. coli proteins) to predict functional associations between these proteins. Similar to other phylogenetic-profile based interaction prediction methods, our method uses the amino acid sequences of proteins and a set of completely sequenced genomes belonging to different species.
Figure II.1: An example illustrating binary phylogenetic profiles. Symbols $\bullet$ and $\circ$ indicate presence and absence, respectively, of a protein in a genome.

The method consists of three major steps:

1. Constructing detailed phylogenetic profiles for all proteins,

2. Using these profiles, constructing coevolutionary matrices for all protein pairs,

3. Assigning phylogenetic similarity scores to all protein pairs based on these matrices.

The following sections describe each of these steps in detail.

II.B.1 Constructing Phylogenetic Profiles

Protein Phylogenetic Profiles.

A phylogenetic profile of a protein is a vector, where each entry quantifies the existence of the protein in a genome. An example for phylogenetic profiles is shown in Figure 1. In this example, only two symbols $\bullet$ and $\circ$ are used to indicate the presence or absence of a protein in a genome, respectively. Each row in the figure is the binary phylogenetic profile of the respective protein. Observe that the proteins $P_1$ and $P_3$ in the figure are likely to share a particular function as their phylogenetic profiles suggest that they have followed a similar evolutionary trajectory.

Conventional methods [10, 14], hereon referred to as Single-Profile methods, rely on a single phylogenetic profile associated with each protein. Given a set of proteins $P = \{P_1, P_2, ..., P_n\}$ and genomes $G = \{G_1, G_2, ..., G_m\}$, the phylogenetic profile $\psi_i$ for protein $P_i$ is a vector defined as

$$profile_i(j) = -1/\log(E_{ij}), \ 1 \leq j \leq m \quad (II.1)$$
where $E_{ij}$ is the minimum (i.e., most significant) BLAST [16] $E$-value of local alignments between $P_i$ and $G_j$. Each profile element is thus a real value that quantifies our confidence of knowing whether a protein exists in a genome. To avoid the logarithm-induced artifacts, the maximum value that a phylogenetic profile element can take is set to 1, indicating the absence of the protein in the corresponding genome. This corresponds to an $E$-value cutoff of 0.5 if $\log_2$ is used. This threshold was used to faithfully replicate the method of [14] so that our method can be compared with a well-known implementation of the Single-Profile method. As was noted in the same study, using real values instead of booleans for profile elements offers the advantage of capturing degrees of sequence divergence, providing greater information than booleans.

For assessing the similarity between two phylogenetic profiles, mutual information provides a useful measure that takes into account co-existence and co-absence of proteins together. Indeed, it has been shown to be reliable and used successfully for predicting protein interactions [14]. The mutual information $I(X,Y)$ of a pair of random variables $X$ and $Y$ is defined as follows:

$$I(X,Y) = H(X) + H(Y) - H(X,Y),$$  

where $H(X)$ is the Shannon entropy of $X$, which is defined as:

$$H(X) = -\sum_{x \in \mathcal{X}} p_x \log(p_x).$$  

Here, $\mathcal{X}$ is the set of possible values taken by $X$ and $p_x = \Pr\{X = x\}$. Similarly, $H(X,Y)$ is the joint entropy of $X$ and $Y$.

A probability distribution for a phylogenetic profile $\psi_i$ is computed by quantizing profile elements into a certain number of bins and estimating the relative frequency of each bin. Then, the phylogenetic similarity between $\psi_i$ and $\psi_j$ is computed as

$$\mu_S(P_i, P_j) = I(\psi_i, \psi_j)$$  

by the Single-Profile method. In the example of Figure 1, the mutual information between the profiles of $P_1$ and $P_3$ is 1, while it is 0 between $P_1$ and $P_2$. Intuitively, as $P_2$
Figure II.2: (a) An example illustrating that the Single-Profile method does not capture domain-level evolutionary histories. Protein \( P_2 \) contains two domains, shown by thick lines on its sequence. While domain \( D_1 \) on protein \( P_2 \) follows an evolutionary trajectory similar to that of proteins \( P_1 \) and \( P_3 \) of Figure 1, the phylogenetic profile of \( P_2 \) does not reveal this information as it combines the independent evolutionary histories of \( D_1 \) and \( D_2 \). (b) Dividing \( P_2 \) into fixed-size segments, we can capture the phylogenetic similarity between proteins \( P_1 \) and \( P_2 \) since \( \mu_M(P_1, P_2) = \max_s I(\psi_1, \psi_2^s) = I(\psi_1, \psi_1^1) = 1 \).

exists in all genomes, its co-existence with \( P_1 \) in some genomes does not provide any information on the functional association of these proteins.

**Segment Phylogenetic Profiles.**

While providing a useful computational method for predicting interactions between proteins, Single-Profile methods may miss many existing interactions. This is because domains within a single protein may have followed very different evolutionary trajectories. Since there are numerous multi-domain proteins in both prokaryotes and eukaryotes, such occurrence may be quite frequent. This point is illustrated by a simple example in Figure 2(a). To capture domain-level co-evolution, the Multiple-Profile method [15] chops each protein \( P_i \) into overlapping segments \( S_i^1, S_i^2, \ldots, S_i^k \) of fixed size
and computes the phylogenetic similarity between two proteins as

$$\mu_M(P_i, P_j) = \max_{s,t} I(\psi^s_i, \psi^t_j),$$

where $\psi^s_i$ denotes the phylogenetic profile for segment $S^s_i$ of a protein $P_i$. The Multiple-Profile method, illustrated in Figure 2(b), is shown to perform better than the Single-Profile method in identifying functional associations between proteins accurately.

**Residue Phylogenetic Profiles.**

While the Multiple-Profile method can detect known interactions missed by the Single-Profile methods by emphasizing on domain-level co-evolution, it still has flaws in capturing the underlying domain information. A scenario where the Multiple-Profile method fails to accurately identify co-evolving domains is shown in Figure 3. The figure illustrates that the Multiple-Profile method may miss potentially informative domains because segments have fixed lengths and their placements are pre-determined.

In this study, to capture the underlying domain information accurately, we further extend the phylogenetic profile based methods by computing residue phylogenetic profiles for each protein. Our approach relies on the fact that a significant local alignment between two proteins corresponds to the unusual similarity between two contiguous portions of the two proteins rather than entire sequences. Therefore, while aligning
a protein with a genome, instead of regarding a significant local alignment as the indicator of existence of the entire protein, we attribute this existence to the residues that are covered in the alignment. This allows fine-grain analysis of sequence conservation at the domain level.

Let \( A(P_i, G_j) \) be the set of significant local alignments between a protein \( P_i \) and a genome \( G_j \). Each alignment \( A \in A(P_i, G_j) \) is associated with a contiguous interval \( T(A) = [r_b, r_e] \) of residues on \( P_i \) and a BLAST \( E \)-value \( E(A) \). Then, for each amino acid residue \( r \) on \( P_i \), we define phylogenetic profile \( \psi^r_i \) as follows:

\[
\psi^r_i(j) = \min_{A \in A_r} -1/\log(E(A)), \quad 1 \leq j \leq m
\]  

II.6

Here, \( A_r = \{ A \in A(P_i, G_j) : r \in T(A) \} \) is the set of local alignments that contain \( r \). In equation 6, the most significant of \( E \)-values for a residue was chosen because we want to know whether the region of a protein covering the residue is present in a genome. Choosing less significant \( E \)-values would mean dampening the signals needed to detect the presence of this region of a protein.

Note that the phylogenetic profile of a single residue does not correspond to its conservation since the alignment can contain mismatches and gaps. However, analyzing residue-level phylogenetic profiles defined in this way provides information on the conservation of a particular portion of the protein. Specifically, if the phylogenetic profiles of a contiguous group of residues are similar, this group might indeed correspond to a conserved domain on the protein. In terms of the co-evolution of two proteins, this corresponds to the co-evolution of such contiguous regions on each protein. In the following sections, we discuss how residue profiles can be used to identify these co-evolved regions.

II.B.2 Computing Coevolutionary Matrices

To capture the co-evolution of proteins at the domain-level, we construct a coevolutionary matrix for each pair of proteins. For a pair of proteins \( P_i \) and \( P_j \) let \( l_i \) and \( l_j \) denote their respective lengths. The coevolutionary matrix \( M_{ij} \) of \( P_i \) and \( P_j \) is an
A \( l_i \times l_j \) rectangular matrix, where each entry corresponds to the mutual information score between a pair of residues each from one protein, \( i.e., \)

\[
M_{ij}(r, s) = I(\psi^r_i, \psi^s_j),
\]

(II.7)

for \( 1 \leq r \leq l_i \) and \( 1 \leq s \leq l_j \). Each entry of the matrix quantifies the residue-level co-evolution between the two proteins. If the proteins contain a co-evolved domain, this appears as a contiguous block of high mutual information scores. Sample coevolutionary matrices for the \( E. coli \) proteins that are shown in Figures II.8 and II.9 illustrate this point.

Note that the computation of full coevolutionary matrices might be infeasible in practice. Given a set of \( n \) proteins and \( m \) genomes, it is necessary to compute \( O(n^2) \) matrices. If the longest protein consists of \( l \) residues, the overall time complexity is \( O(ml^2n^2) \). Since conserved regions are usually fairly long, considering all pairs of residues on them is redundant. Therefore, by downsampling the coevolutionary matrix, we can avoid the complexity penalty without significantly impacting the sensitivity of the algorithm. Using a downsampling factor of \( f \), the size of the largest coevolutionary matrix is reduced to \( l^2/f^2 \). In general, \( f \) can set to be large enough so that \( l/f \) is bounded by a constant. Note that the complexity of an algorithm that does not consider individual residues is \( O(mn^2) \). In this manner, the simplification reduces the overhead of residue-profile based algorithm to a constant factor, \( l^2/f^2 \).

### II.B.3 Deriving Phylogenetic Similarity Scores

A coevolutionary matrix contains information about which regions from two proteins have co-evolved. It is important to note that there might be spurious (large) entries in the matrix due to artifacts created while compiling BLAST outputs. To identify co-evolved regions accurately, we use a filtering scheme. Our algorithm is based on the intuition that co-evolved regions of the two proteins must be \textit{sufficiently} large to be considered as significant ones. In terms of the coevolutionary matrix, there must be a sufficiently large sub-matrix such that all entries in that sub-matrix are consistently high.
**Algorithm** Compute Coevolutionary-Matrix

**Input:** Protein sequences $P = \{P_i\}$

**Input:** Genomes $G = \{G_j\}$

**Output:** Phylogenetic similarity scores $\mu_C(P_i, P_j)$ for each protein pair

for each $P_i \in P$

for each $G_j \in G$

$A \leftarrow$ BLAST($P_i, G_j$)

▷ $A$ is the set of all local alignments $A = \{\text{Interval} \ T(A), \ E-value \ E(A)\}$

for each residue $r$ on $P_i$

$E_r(j) \leftarrow \min_{A \in A | r \in T(A)} E(A)$

▷ $E_r(j)$ is the E-value of most significant alignment that contains $r$

$\psi^r_i(j) \leftarrow -1/\log(E_r(j))$

▷ $\psi^r_i$ is the phylogenetic profile of residue $r$ of protein $P_i$

for each $P_i, P_j \in P \times P$

$M_{ij} \leftarrow \{I(\psi^r_i, \psi^s_j) | r \in P_i, s \in P_j\}$

▷ $M_{ij}$ is the coevolutionary matrix of all residue pairs in $P_i \times P_j$

▷ In the implementation, these entries are downsampled

for each $r \in P_i, s \in P_j$

$m(r, s) \leftarrow \min_{r \leq a < r + W, s \leq b < s + W} M_{ij}(a, b)$

▷ $m(r, s)$ is the mutual information of contiguous regions of size $W$ starting at residues $r$ on $P_i$

▷ and $s$ on $P_j$

$\mu_C(P_i, P_j) \leftarrow \max_{r \in P_i, s \in P_j} m(r, s)$

Figure II.4: Algorithm for computing Coevolutionary-Matrix.
Clearly, the sub-matrix with the maximum consistently high mutual information score provides the degree of co-evolution between the two proteins. Hence, we formulate the phylogenetic similarity between proteins $P_i$ and $P_j$ as follows:

$$\mu_C(P_i, P_j) = \max_{1 \leq r \leq l_i, r \leq s < r + W} \min_{1 \leq s \leq l_j, s \leq b < s + W} M_{ij}(a, b)$$ \hspace{1cm} (II.8)$$

Here, $W$ is the window parameter that quantifies the sufficiency of the size of a region on a protein to be considered as a conserved domain. The overall algorithm for computing the Coevolutionary-Matrix based phylogenetic similarity between each pair of proteins is shown in Figure II.4.

**II.C Results**

We implemented the proposed method and tested on 4311 *E. coli* proteins. We used 152 genomes to construct phylogenetic profiles. Although some genomes are “redundant” in the sense that they share a large fraction of their proteins, our collection of genomes is diverse enough to cover the three branches of life (131 Bacteria, 17 Archaea, and 4 Eukaryota). The complete list of genomes is at [http://genome.ucsd.edu/CoevolutionaryMatrix/list-152.txt](http://genome.ucsd.edu/CoevolutionaryMatrix/list-152.txt). Using a default setting, we ran BLAST (i.e., blastp program) for each one of 4311 *E. coli* proteins against each one of 152 genomes. For the Single-Profile, only the most significant $E$-value was kept for each protein. For the Coevolutionary-Matrix, the same BLAST run was carried out except that all matched region information and corresponding $E$-values meeting the threshold were kept.

To reduce the time and memory requirements associated with the filtering algorithm, we downsampled the coevolutionary matrix by a factor of $f = 30$. For two proteins with $l_i$ and $l_j$ amino acid residues, the dimensions of their coevolutionary matrix is $(l_i/30) \times (l_j/30)$. In addition, the parameter $W = 2$ was chosen. The use of the downsampling factor $f$ of 30 and $W$ of 2 translate to dividing proteins into overlapping segments that are 60 residues long. Since an average domain size is around 100 residues, current values for the $f$ and $W$ are reasonable.
Using this implementation of the Coevolutionary-Matrix method and an implementation of the Single-Profile method proposed by [14], we compared their performances. Since homologous proteins should have similar phylogenetic profiles and thus have high mutual information scores, we excluded them from our analysis. To compare mutual information scores under the two methods, we converted them into \( p \)-values. Here, the \( p \)-value of a protein pair is defined as the fraction of non-homologous protein pairs in \( E. coli \) that have higher mutual information score than the one in question. In other words,

\[
p(\mu(P_i, P_j)) = \frac{|\{(P_a, P_b) \in N : \mu(P_a, P_b) > \mu(P_i, P_j)\}|}{|N|},
\]  

where \( N \) is the set of all non-homologous protein pairs. Here, \( \mu \) denotes the phylogenetic similarity score assigned by the Single-Profile \( (\mu_S) \) or Coevolutionary-Matrix \( (\mu_C) \) method.

We used a set of reference protein interactions that we derived from the KEGG database [17] to test and compare the Single-Profile and Coevolutionary-Matrix methods. We use the term “interactions between proteins” to imply a broad range of interactions, from physical binding to functional association. In this respect, proteins participating in different steps of a biochemical pathway are considered interacting. Consequently, we define a reference interaction as a pair of proteins that share a KEGG pathway assignment. To generate this set of reference interactions, for each \( E. coli \) pathway retrieved from KEGG, we formed a “clique” of proteins that participate in the corresponding pathway. The final reference set consists of 1282 proteins and 43,331 interacting protein pairs derived from these proteins after excluding homologous pairs (BLAST \( E \)-value < 1.0).

II.C.1 Comparison of Coevolutionary-Matrix and Single-Profile Methods

Both the Coevolutionary-Matrix and Single-Profile methods are used to predict interactions between \( E. coli \) proteins by setting a threshold on the phylogenetic similarity score. In other words, proteins \( P_i \) and \( P_j \) are predicted as interacting partners.
if $\mu(P_i, P_j) > \mu^*$. For each value of $\mu^*$, coverage is defined as the sum of true positives (TP) and false positives (FP). Both are numbers of protein pairs that meet the threshold. Furthermore, proteins in each pair are represented in the KEGG data set. The difference is that true positive protein pairs are interacting in the KEGG data set but not false positive pairs. In addition, Positive Predictive Value (PPV) is defined as $TP/(TP+FP)$.

PPV versus coverage plots for the Coevolutionary-Matrix and Single-Profile methods are shown in Figure II.5. A similar plot for the Multiple-Profile method is also shown for comparison. It is evident from the figure that the Coevolutionary-Matrix method has about 1.5 fold greater coverage at PPV of 0.7 than that of the Single-Profile method. ROC curves for the methods, which plot Sensitivity against (1-Specificity), also indicate that the Coevolutionary-Matrix performs better than the Single-Profile, although the difference is rather small (figure not shown). Sensitivity is defined as $TP/(TP+FN)$, and Specificity is $TN/(FP+TN)$, where TN and FN are true negatives and false negatives, respectively. Both TN and FN are numbers of protein pairs that do not meet the threshold. In addition, proteins in each pair are represented in the KEGG data set. Their difference is that true negative protein pairs are not interacting in the KEGG data set while false negative pairs are.

To have a closer look at the performances of the two methods, we show PPV, Specificity, and Sensitivity for the three different sets of predicted pairs by each method in Table II.1. At same number of predicted pairs, the Coevolutionary-Matrix method is again shown to perform better than the Single-Profile both in terms of PPV and Sensitivity. In Table II.2, we also show PPV for both overlapping and non-overlapping areas between the sets of predicted pairs in Table II.1. In section A in Table II.2, percent overlap between the two sets is 55% and the PPV for this overlap is 0.75, which is higher than any one of the methods alone. Furthermore, PPV of the Coevolutionary-Matrix method alone is higher than that of the Single-Profile method alone (0.57 versus 0.33). Similar observations are made for the sections B and C in Table II.2. These results indicate that the Coevolutionary-Matrix method predicts a significantly different set of interactions from those of the Single-Profile with greater number of true positives.
Table II.1: Number of predicted interactions at various mutual information score thresholds. MIS = mutual information score threshold; # of PP = number of predicted protein pairs; Covg. = coverage; TP = true positives; PPV = Positive Predictive Value.

**Single-Profile**

<table>
<thead>
<tr>
<th>MIS</th>
<th># of PP</th>
<th>Covg.</th>
<th>TP</th>
<th>PPV</th>
<th>Specificity</th>
<th>Sensitivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.68702</td>
<td>15340</td>
<td>1239</td>
<td>789</td>
<td>0.637</td>
<td>0.99942</td>
<td>0.01821</td>
</tr>
<tr>
<td>0.72673</td>
<td>8549</td>
<td>855</td>
<td>620</td>
<td>0.725</td>
<td>0.99970</td>
<td>0.01431</td>
</tr>
<tr>
<td>0.76461</td>
<td>4971</td>
<td>617</td>
<td>499</td>
<td>0.809</td>
<td>0.99985</td>
<td>0.01152</td>
</tr>
</tbody>
</table>

**Coevolutionary-Matrix**

<table>
<thead>
<tr>
<th>MIS</th>
<th># of PP</th>
<th>Covg.</th>
<th>TP</th>
<th>PPV</th>
<th>Specificity</th>
<th>Sensitivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.60500</td>
<td>15339</td>
<td>1620</td>
<td>1091</td>
<td>0.673</td>
<td>0.99932</td>
<td>0.02518</td>
</tr>
<tr>
<td>0.64350</td>
<td>8548</td>
<td>1043</td>
<td>792</td>
<td>0.759</td>
<td>0.99968</td>
<td>0.01828</td>
</tr>
<tr>
<td>0.68200</td>
<td>4970</td>
<td>720</td>
<td>610</td>
<td>0.847</td>
<td>0.99986</td>
<td>0.01408</td>
</tr>
</tbody>
</table>

To determine which KEGG pathways are most represented in top scoring protein pairs using each method, we took top 2000 pairs out of all non-homologous pairs for the *E. coli* proteome and counted number of those that belong to each pathway (data not shown). Phylogenetic similarity score thresholds used to generate these sets are 0.835 ($p < 2.2 \times 10^{-4}$) for the Single-Profile and 0.741 ($p < 2.2 \times 10^{-4}$) for the Coevolutionary-Matrix. These thresholds are considered to be strict and hence should yield high-confidence predictions.

Top-scoring protein pairs predicted by both methods are from a wide range of pathways, whose rankings based on the number of protein pairs falling into each are similar (36 pathways for the Coevolutionary-Matrix and 29 for the Single-Profile method out of total of 131 KEGG pathways). Some of the highly populated pathways shared by both methods include flagellar assembly, phosphotransferase system, ABC transporters, oxidative phosphorylation, ubiquinone biosynthesis, and histidine metabolism. In the set of 2000 pairs for the Single-Profile method, there are 339 pairs that share at least a KEGG pathway. For the Coevolutionary-Matrix method, there are 391 pairs with 281 of them overlapping with those of the Single-Profile method. Despite this relatively high number of shared pairs, the overall percent overlap between the two sets of 2000 pairs is 61.5%.
Figure II.5: Positive Predictive Value (PPV) versus Coverage plots for the Single-Profile, Multiple-Profile, and Coevolutionary-Matrix methods using the KEGG interaction data set. As mutual information score threshold is varied, coverage and PPV at that threshold are plotted. Each dot represents such pair.

Table II.2: Number of true interactions in the overlapping and non-overlapping sets of predicted interactions between the Single-Profile (SP) and Coevolutionary-Matrix (CM) methods. Symbols ∧ and ∼ indicate “logical AND” and “logical negation,” respectively; # of PP = number of predicted protein pairs; Covg. = coverage; TP = true positives; PPV = Positive Predictive Value.

<table>
<thead>
<tr>
<th></th>
<th># of PP</th>
<th>Covg.</th>
<th>TP</th>
<th>PPV</th>
<th># of PP</th>
<th>Covg.</th>
<th>TP</th>
<th>PPV</th>
<th># of PP</th>
<th>Covg.</th>
<th>TP</th>
<th>PPV</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
<td>B</td>
<td>C</td>
<td></td>
<td>A</td>
<td>B</td>
<td>C</td>
<td></td>
<td>A</td>
<td>B</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>SP∧(~CM)</td>
<td>6932</td>
<td>340</td>
<td>111</td>
<td>0.33</td>
<td>3731</td>
<td>233</td>
<td>94</td>
<td>0.40</td>
<td>2073</td>
<td>160</td>
<td>83</td>
<td>0.52</td>
</tr>
<tr>
<td>CM∧(~SP)</td>
<td>6931</td>
<td>721</td>
<td>413</td>
<td>0.57</td>
<td>3730</td>
<td>421</td>
<td>266</td>
<td>0.63</td>
<td>2072</td>
<td>263</td>
<td>194</td>
<td>0.74</td>
</tr>
<tr>
<td>SP∧CM</td>
<td>8408</td>
<td>899</td>
<td>678</td>
<td>0.75</td>
<td>4818</td>
<td>622</td>
<td>526</td>
<td>0.85</td>
<td>2896</td>
<td>457</td>
<td>416</td>
<td>0.91</td>
</tr>
</tbody>
</table>
Figure II.6: Mutual information score distributions under the Single-Profile and Coevolutionary-Matrix methods. From the KEGG interaction set, 10,000 interacting proteins are randomly selected. Then another 10,000 protein pairs, without requiring them to be interacting, are randomly selected from the proteins present in the KEGG interaction set.
Figure II.7: *P*-values of mutual information scores of interacting proteins under the Single-Profile method versus those under the Coevolutionary-Matrix method. The same set of 10,000 interacting protein pairs used in Figure II.6 is used here. Each circle represents two proteins that are known to interact.
In Figure II.6, we show mutual information score distributions of sets of interacting and random protein pairs calculated with each method. At zero mutual information score, the Single-Profile method shows a peak for the distribution of interacting protein pairs while the Coevolutionary-Matrix method does not have this peak. Based on this observation, it appears that a significant portion of the interacting protein pairs with very low mutual information scores under the Single-Profile gained higher (and potentially meaningful) scores under the Coevolutionary-Matrix. To investigate this further, Figure II.7 shows how \( p\)-values of mutual information scores of the interacting protein pairs are correlated between the two methods. Although there is a rough correlation, there are many outliers. Some of these have \( p\)-value differences as high as four orders of magnitude. The presence of these outliers indicate that the two methods can make very different predictions for some proteins.

II.C.2 Examples of Domain Co-evolution

In this section, we show three examples of domain-level co-evolution from the \textit{E. coli} cellular systems. We then hypothesize how co-evolved domains detected by the Coevolutionary-Matrix method fit with existing biochemical data. Finally, top interacting partners predicted by the two methods for these proteins are compared.

**Phosphotransferase System (PTS)**

The Phosphotransferase System is the major pathway through which translocation of sugars across the bacterial inner membrane is coupled with phosphorylation [18]. The cytoplasmic protein IIAB transfers a phosphoryl group from the cytoplasmic proteins I and HPr to substrates through interactions with the membrane proteins IIC and IID in the case of mannose-specific PTS [18]. The co-evolving region detected for the proteins IIAB and IIC is shown in Figure II.8.

Figure II.8 clearly captures two different regions of IIAB. In fact, these regions correspond to the domains IIA (residues 1-170) and IIB (residues 170-320). Figure II.8 also captures the notion that the domain IIB co-evolved with IIC instead of the domain...
IIA. This can be explained in light of how the task assigned to IIAB as a whole is divided between IIA and IIB. Within the PTS, domain IIA has the role of receiving the phosphoryl group from proteins I and HPr and passing it to domain IIB. Domain IIB then passes the phosphoryl group to membrane proteins (in this case, IIC and IID) and then to sugars. Physical interaction between the domain IIB and protein IIC likely drives their co-evolution. This is shown for the mannitol-specific protein domains IIB and IIC, where domains IIA, IIB, and IIC are fused to form a single protein [19]. Similar observation has been made by another group of researchers based on a different study [20].

In Table II.3, top 20 predicted interacting partners of the protein IIAB under the Single-Profile and Coevolutionary-Matrix methods are shown. Although both methods pick out the proteins IIC (ManY) and IID (ManZ) as their top scoring proteins, those under the Coevolutionary-Matrix show greater number of proteins that are involved in PTS. Four PTS proteins (ManY, ManZ, AgaC, and AgaD) are found using the Single-Profile method and eight PTS proteins (ManY, ManZ, AgaD, AgaC, AgaW, CelC, CelB, and CelA) are found with the Coevolutionary-Matrix method.

**Chemotaxis**

Chemotaxis signaling pathway allows a bacterium to sense the state of its external environment and determine its swimming behavior accordingly. CheA is a multi-domain protein whose domains carry out different functions in this system [21]. A plot of the coevolutionary matrix for CheA and CheB, another chemotaxis component, is shown in Figure II.9.

Figure II.9 suggests that the N-terminus and C-terminus regions of CheA (residues 1-200 and 540-670, respectively) co-evolved with the C-terminus region of CheB (residues 170-340). Although there is a biochemical evidence that CheB binds to the N-terminus region [22], none exists for the binding of CheB to the C-terminus region. However, it is known that CheW, a chemotaxis component, binds to the C-terminus region [23]. The sequence region from residues 200 to 350 of CheA, which
Figure II.8: Coevolutionary matrix plot for the *E. coli* proteins mannose-specific IIAB and IIC (ManY). Darker color indicates higher mutual information score. The matrix shown here is the downsampled version of the original matrix. Each bin is 10 residues wide.
Table II.3: Top 20 interacting partners predicted by the Single-Profile and Coevolutionary-Matrix methods for the *E. coli* mannose-specific IIAB. Predicted interacting proteins are ranked based on their mutual information scores (MI) and shown here with their NCBI GenBank Identifiers (GI) and names. GI number of mannose-specific IIAB is 16129771. Each entry represents a single protein and more than one name is provided if available. Known PTS components are highlighted.

<table>
<thead>
<tr>
<th>Rank</th>
<th>Single Profile</th>
<th>Coevolutionary-Matrix</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MI</td>
<td>pValue</td>
</tr>
<tr>
<td>1</td>
<td>0.728</td>
<td>0.0009</td>
</tr>
<tr>
<td>2</td>
<td>0.690</td>
<td>0.0016</td>
</tr>
<tr>
<td>3</td>
<td>0.652</td>
<td>0.0028</td>
</tr>
<tr>
<td>4</td>
<td>0.642</td>
<td>0.0033</td>
</tr>
<tr>
<td>5</td>
<td>0.623</td>
<td>0.0046</td>
</tr>
<tr>
<td>6</td>
<td>0.608</td>
<td>0.0058</td>
</tr>
<tr>
<td>7</td>
<td>0.598</td>
<td>0.0069</td>
</tr>
<tr>
<td>8</td>
<td>0.594</td>
<td>0.0072</td>
</tr>
<tr>
<td>9</td>
<td>0.588</td>
<td>0.0080</td>
</tr>
<tr>
<td>10</td>
<td>0.582</td>
<td>0.0089</td>
</tr>
<tr>
<td>11</td>
<td>0.581</td>
<td>0.0091</td>
</tr>
<tr>
<td>12</td>
<td>0.574</td>
<td>0.0101</td>
</tr>
<tr>
<td>13</td>
<td>0.569</td>
<td>0.0109</td>
</tr>
<tr>
<td>14</td>
<td>0.565</td>
<td>0.0115</td>
</tr>
<tr>
<td>15</td>
<td>0.565</td>
<td>0.0116</td>
</tr>
<tr>
<td>16</td>
<td>0.561</td>
<td>0.0124</td>
</tr>
<tr>
<td>17</td>
<td>0.560</td>
<td>0.0126</td>
</tr>
<tr>
<td>18</td>
<td>0.558</td>
<td>0.0130</td>
</tr>
<tr>
<td>19</td>
<td>0.555</td>
<td>0.0136</td>
</tr>
<tr>
<td>20</td>
<td>0.555</td>
<td>0.0137</td>
</tr>
</tbody>
</table>
Figure II.9: Coevolutionary matrix for the *E. coli* proteins CheA and CheB. Darker color indicates higher mutual information score. The matrix shown here is the downsampled version of the original matrix. Each bin is 10 residues wide.

shows weaker co-evolution, corresponds to the dimerization domain [24]. Another region of CheA (residues 355-540) that does not seem to co-evolve with CheB corresponds to the kinase domain. The co-evolving regions of CheA identified from the matrix are essentially the same for Mcp’s, CheW, CheR, and CheB proteins.

In Table II.4, top 20 predicted interacting partners of CheA using each method are shown. Under the Single-Profile method, only Mcp3 is known to participate in chemotaxis. In contrast, Mcp3, Mcp2, CheW, Mcp4, CheR, and CheB are known to participate in chemotaxis under the Coevolutionary-Matrix method [21]. In the same list, Aer is involved in aerotaxis; and FlgC, MotB, MotA, FlgF, and FlgL are likely picked because the chemotaxis signaling pathway is coupled to the flagellar motor system. As a note, Mcp2, Mcp3, Mcp4, and Aer are homologous (BLAST *E*-value < 1.0).

**Kdp system**

In *E. coli*, KdpD and KdpE regulate expression of the kdpFABC operon, which encodes a high affinity *K*⁺ transport ATPase [25]. KdpD is a multi-domain protein which consists of an N-terminal cytoplasmic domain (residues 1-395), four transmembrane domains, and a cytoplasmic C-terminal transmitter domain [26].
Table II.4: Top 20 interacting partners predicted by the Single-Profile and Coevolutionary-Matrix methods for the *E. coli* CheA. Predicted interacting proteins are ranked based on their mutual information scores (MI) and shown here with their NCBI GenBank Identifiers (GI) and names. GI number of CheA is 16129840. Each entry represents a single protein and more than one name is provided if available. Known chemotaxis components are highlighted.

<table>
<thead>
<tr>
<th>Rank</th>
<th>Single-Profile</th>
<th>Coevolutionary-Matrix</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MI</td>
<td>pValue</td>
</tr>
<tr>
<td>1</td>
<td>0.897</td>
<td>0.000097</td>
</tr>
<tr>
<td>2</td>
<td>0.871</td>
<td>0.000136</td>
</tr>
<tr>
<td>3</td>
<td>0.844</td>
<td>0.000192</td>
</tr>
<tr>
<td>4</td>
<td>0.830</td>
<td>0.000228</td>
</tr>
<tr>
<td>5</td>
<td>0.778</td>
<td>0.000451</td>
</tr>
<tr>
<td>6</td>
<td>0.777</td>
<td>0.000456</td>
</tr>
<tr>
<td>7</td>
<td>0.770</td>
<td>0.000497</td>
</tr>
<tr>
<td>8</td>
<td>0.769</td>
<td>0.000504</td>
</tr>
<tr>
<td>9</td>
<td>0.765</td>
<td>0.000528</td>
</tr>
<tr>
<td>10</td>
<td>0.757</td>
<td>0.000591</td>
</tr>
<tr>
<td>11</td>
<td>0.757</td>
<td>0.000594</td>
</tr>
<tr>
<td>12</td>
<td>0.756</td>
<td>0.000599</td>
</tr>
<tr>
<td>13</td>
<td>0.744</td>
<td>0.000716</td>
</tr>
<tr>
<td>14</td>
<td>0.742</td>
<td>0.000737</td>
</tr>
<tr>
<td>15</td>
<td>0.740</td>
<td>0.000766</td>
</tr>
<tr>
<td>16</td>
<td>0.733</td>
<td>0.000841</td>
</tr>
<tr>
<td>17</td>
<td>0.710</td>
<td>0.001163</td>
</tr>
<tr>
<td>18</td>
<td>0.709</td>
<td>0.001177</td>
</tr>
<tr>
<td>19</td>
<td>0.709</td>
<td>0.001181</td>
</tr>
<tr>
<td>20</td>
<td>0.702</td>
<td>0.001312</td>
</tr>
</tbody>
</table>

Coevolutionary-Matrix method clearly delineates three corresponding domains in Figure II.10.

Figure II.10 suggests that the N-terminal domain of KdpD co-evolved with KdpC. Supporting this hypothesis, a recent study has shown that this N-terminal domain alone triggers semi-constitutive expression of the kdpFABC operon through interactions with KdpE [26]. Interaction between KdpD and KdpC is therefore of functional dependence rather than physical. Top 10 interacting partners predicted by the Single-Profile include only KdpE from this system while those of the Coevolutionary-Matrix include KdpE, KdpA, and KdpC. Mutual information scores of KdpC and KdpA with respect to KdpD using the Single-Profile method are 0.1829 and 0.2496, respectively. These very low mutual information scores suggest that the Single-Profile method cannot detect co-evolution between KdpC/KdpA and KdpD.
Figure II.10: Coevolutionary matrix for the *E. coli* proteins KdpD and KdpC. Darker color indicates higher mutual information score. The matrix shown here is the downsampled version of the original matrix. Each bin is 10 residues wide.

Table II.5: Top 10 interacting partners predicted by the Single-Profile and Coevolutionary-Matrix methods for the *E. coli* KdpD. Predicted interacting proteins are ranked based on their mutual information scores (MI) and shown here with their NCBI GenBank Identifiers (GI) and names. GI number of KdpD is 16128671. Each entry represents a single protein. Known Kdp system components are highlighted.
II.D Discussion

The results shown in this paper strongly suggest that co-evolution of proteins should be captured at the domain-level. As indicated by the coevolutionary matrices shown in Figures II.8, II.9, and II.10, sequence regions with conflicting evolutionary histories can co-exist within a single protein. By representing protein co-evolution at the domain-level, the Coevolutionary-Matrix method can assign very different phylogenetic similarity scores to proteins when compared with the Single-Profile method (Figure II.7). In turn, these differences have substantial effect on the performances of the two methods (Tables II.3, II.4, and II.5).

Others have also noted the importance of including domain information when predicting protein interactions. By incorporating interaction profile of domains in their method, [27] reported increased performance in inferring protein interaction of one organism from the interaction network of another. Similar in spirit to our approach, [28] improved upon the phylogenetic profiling method by using domains defined with the Pfam database [29]. Although the coverage of their method is limited by that of the Pfam database, it has the advantage of requiring less computing time and having a simple update procedure as more genomes are used.

Interestingly, similar to databases such as Pfam [29], the Coevolutionary-Matrix method can delineate “domains” within a protein. Because of the way parameters were chosen, the co-evolving regions detected with our method are required to have sizes of at least 60 residues. The size requirement ensures that it is in the range of independently folding protein domains, excluding those of loops (i.e., less than 20 residues).

Motivated by the performance of the Coevolutionary-Matrix method, we explored the idea of whether co-evolving domains captured indeed are involved in interactions at the domain-level. For the PTS proteins IIAB and IIC, physical interaction between the domain IIB and protein IIC seem plausible based on available evidence. However, for some proteins such as those involved in the chemotaxis pathway, it ap-
pears that much of the co-evolution between the domains were driven by their functional dependence. For example, the Coevolutionary-Matrix method identified that the N-terminus and C-terminus regions of CheA co-evolved with CheB. The method indicates that these same regions also co-evolved with Mcp’s, CheW, and CheR proteins. Most likely all these proteins do not physically interact with the same two regions of CheA. Likewise, the N-terminus domain of KdpD in the Kdp system does not physically interact with KdpC or KdpA but is needed to drive the expression of the latter two proteins.

II.E Concluding Remarks

Since evolution and functions of proteins are coupled, greater understanding of the former can reveal much about the latter. By capturing co-evolution of proteins at the domain-level, regions that are important for supporting both functional and physical interactions between these proteins are detected. With examples from the cellular systems of the *E. coli* bacterium, we showed that these regions correspond to biochemically characterized protein domains.

Acknowledgments

Section II, in full, is a reprint of the article "Inferring functional information from domain co-evolution" by Y. Kim, M. Koyuturk, U. Topkara, A. Grama, and S. Subramaniam, published in *Bioinformatics*, volume 22, pages 40-49, 2006. Y. Kim was the primary researcher and author for this work. This work has been reprinted with permissions from co-authors and Oxford University Press.
III

Energy function derived from statistical analysis of backbone dihedral angles of proteins

III.A Introduction

Backbone angles of proteins, $\phi$ and $\psi$, are important variables that characterize protein structures. Thus, it is not surprising that they have been studied for quite some time, one of the most famous examples being that of the Ramachandran plot [30]. Traditionally constructed for 20 amino acid types, each Ramachandran plot contains a distribution of $\phi$ and $\psi$ angles that reflect physical properties of the corresponding amino acid. For instance, because of their unique side-chain structures, glycines and prolines have distributions that are easily distinguished from those of the remaining 18 residues. Although backbone-angle propensities have been studied for a number of decades, physical details of how they arise are still incomplete [31, 32].

To capture sequence to structure relationships without explicitly modeling these physical details, investigators have developed what are called Statistical Effective Energy Function (SEEF) [33]. Assuming that the database of protein structures contains structures with minimum free energies, statistics of chosen variables are collected and
converted into energy-like quantities. One of the major advantages of this approach is that a delicate balance of various types of energy terms can be captured without explicitly representing them. The torsion-angle based energy functions presented in this study are also constructed based on this idea.

Statistics of local backbone dihedral angles of proteins have been studied by numerous groups [34, 35, 36, 37, 38, 39, 40]. For many of these works, mono- and di-peptides have been studied using various types of data mining and filtering techniques to deal with scarcity of data. More recently, others have also studied tri-peptide specific PDFs [41, 42]. For instance, by optimizing an information measure, PDFs for tri-peptides and longer peptides using a reduced set of amino acids have been constructed [41]. In another paper, energy functions of different resolutions were derived from tri-peptide specific PDFs, and they were tested in protein folding simulations [42]. These papers have shown that indeed nearest neighboring residues influence structural preferences of the middle residue, and that incorporating this information into energy functions results in higher performing energy functions.

Extending these works, we further characterize the tri-peptide and mono-peptide specific PDFs to better illustrate the relationships between sequences of tri-peptides and their structures. Also we test the energy functions derived from these PDFs by carrying out threading experiments, where they will be used widely, and compare their performances with those of other energy functions. Finally, fractions of foldable sequences for a large number of proteins are indirectly measured using these energy functions.

### III.B Methods

Details of mapping sequence-structure relationships of tri-peptides onto a 2-dimensional map are presented in this section. Also presented are details of constructing mono-peptide specific $TPDF_{\text{mono}}$ and tri-peptide specific $TPDF_{\text{tri}}$ energy functions as well as other energy functions built with different emphasis. Lastly, the threading experiments used to benchmark and compare these energy functions are described.
III.B.1 Energy Functions Tested

The five energy functions tested in the present study are $TPDF_{mono}$, $TPDF_{tri}$, $DPDF_{local}$, $SipplPair$, and $Solv$ (See Equations III.1, III.2, III.3 and III.4). Of the five energy functions, $SipplPair$ and $Solv$ have been extensively used in both threading and protein folding algorithms [43, 44]. Except for $DPDF_{local}$, all other energy functions treat each residue as a point in space. Also, all of these energy functions use the Boltzmann relation (i.e. $energy = -\ln(P(x))$) to convert probabilities into energy-like quantities. Because these energy functions capture different physical properties of protein structures, they should behave differently.

$$TPDF(x) = -\sum_{i=2}^{M-1} \ln(1.0 + P(\phi, \psi|peptide_i))$$  \hspace{0.5cm} (III.1)

$$DPDF_{local}(x) = -\sum_{v=1}^{N} \sum_{w>v}^{N} \ln(1.0 + P(d|s, atom_v, atom_w))$$  \hspace{0.5cm} (III.2)

$$SipplPair(x) = -\sum_{i=1}^{M} \sum_{j>i}^{M} \ln \left[1.0 + \frac{P(d|residue_i, residue_j)}{P(d)}\right]$$  \hspace{0.5cm} (III.3)

$$Solv(x) = -\sum_{i=1}^{M} \ln \left[1.0 + \frac{P(countNeighbor(i)|residue_i)}{P(countNeighbor(i))}\right]$$  \hspace{0.5cm} (III.4)

where $x$ represents coordinates of the input structure; $M$ is a length of the protein sequence; $i$ and $j$ are residue indexes; $N$ is a total number of backbone atoms (i.e. N, Cα, C, and O); $v$ and $w$ are atom indexes; $d$ is a distance in Angstrom; $s$ is a sequence separation.

Torsion-angle Probability Density Functions

The torsion-angle Probability Density Functions (PDFs) and the energy functions derived from them, $TPDF_{mono}$ and $TPDF_{tri}$, are defined in Equation III.1. For a given structure, $x$, a PDF $P(\phi, \psi|peptide_k)$ returns a conditional probability of seeing a pair of backbone angles, $\phi$ and $\psi$, given $peptide_k$. The variable $peptide_k$ can be either $residue_i$ or $(residue_{i-1}, residue_i, residue_{i+1})$ if $TPDF_{mono}$ or $TPDF_{tri}$ is being
constructed, respectively. For the tri-peptide, the backbone angles are for the middle residue (i.e. residue$_i$).

Backbone angles were extracted from a filtered list of protein structures, and PDFs, $P(\phi, \psi|\text{peptide}_i)$, were constructed from them. Instead of the usual histogram technique, kernel density estimation technique [45] was used to convert a list of $\phi$ and $\psi$ angles into a PDF for each peptide. The kernel density estimation technique uses Equation III.6.

Using this technique, a probability density of a given data point is calculated by finding out how many data points are around it. The more neighboring data points are around the data point in question, its probability density is greater. How each neighboring data point contributes to the final probability density is determined by the kernel function in Equation III.5. Euclidean distance metric was used to calculate distances between pairs of $\phi$-$\psi$ angles. The distance metric takes into account the circular nature of angles (i.e. -180 and +180 degrees are same). In addition, there is a parameter, $h$, that controls the amount of smoothing present in the PDF. The parameter, $h$, was set to 18. Each PDF was represented as a $20 \times 20$ matrix, whose entries all sum to 1.0.

The list of protein structures used to derive PDFs was downloaded from the PISCES server [46], which provides lists of filtered structures from the Protein Data Bank [47] based on various requirements. The list consisted of 4332 structures with cutoffs of 50% sequence identity, 2.0 Angstrom resolution, and 0.25 R-factor. Tri-peptide specific PDFs with not enough data points (i.e. $< 30$) were substituted with corresponding mono-peptide specific PDFs when using them as energy functions. Out of 8000 possible tri-peptides, 6821 had sufficient number of data points based on this cutoff.

$$Kernel(d) = \frac{1}{2\pi} \exp \left[ -\frac{d^2}{2h^2} \right] \quad \text{(III.5)}$$

where $d$ is a distance between the input and the observed $\phi$-$\psi$ angles, and $h$ is a smoothing parameter.
\[ KDE(x) = \frac{1}{N} \sum_{i=1}^{N} \text{Kernel}(x - x_i) \]  

where \( x \) is an input \( \phi-\psi \) angles, \( N \) is a total number of observed data points, and \( (x - x_i) \) is a distance between the input and observed \( \phi-\psi \) angles.

**Distance Probability Density Functions**

Distance Probability Density Functions, \( DPDF \), has been already described [48]. Briefly, it is a collection of probability density functions of distances between atoms. To use \( DPDF \) for the threading experiments, \( DPDF_{local} \) was derived from \( DPDF \) such that only pair-wise interactions between backbone atoms were considered. Also interactions between atoms with residue separations more than 4 residues were not considered, emphasizing local interactions. The energy function is defined in Equation III.2.

**SipplPair**

The energy function \( SipplPair \) was derived from statistics of pair-wise distances of residues and follows the normalization scheme in [49]. \( SipplPair \) is defined in Equation III.3. \( C_\beta \) atoms were used as centers of residues except glycine, where \( C_\alpha \) atom was used instead.

**Solvation**

To capture the effect of excluding solvent atoms, \( Solv \) collects statistics of number of neighboring residues within a sphere of size 10 Angstrom around each residue, converting them into PDFs. Each PDF then quantifies how much each residue prefers to be surrounded by neighboring residues. In Equation III.4, the function \( P(countNeighbor(i)|residue_i) \) returns a probability of observing a certain number of residues around a given residue. The function \( P(countNeighbor(i)) \) is essentially the
same except that the probability is for all 20 residue types combined, and it is used for normalization purpose.

### III.B.2 Multi-Dimensional Scaling

Multi-dimensional scaling technique was used to map pair-wise distances of torsion-angle PDFs described in the earlier section onto a plane. This approach is a method of visualizing PDFs such that similar PDFs are placed near each other on the 2-dimensional map. Likewise, PDFs with large distances are placed far apart on the map.

For all pairs of PDFs, root mean square deviations (RMSD) were calculated to represent distances. Each torsion-angle PDF is represented as a $20 \times 20$ matrix, whose elements all sum to 1. Then the multi-dimensional technique [1] was applied to the matrix of all pair-wise PDF distances, and the coordinate system of the matrix was reoriented such that first $n$ dimensional axes capture most information about the data points. In this study, the first two largest components were used as axes.

### III.B.3 Threading Experiments

To measure the performances of the energy functions, two types of ungapped threading experiments were carried out. The first threading experiment assigns normalized energies to native sequences using the five energy functions. This experiment should determine how well the energy functions are able to detect native sequences from randomly generated sequences. The second experiment, on the other hand, threads distant homologues of the template structures, testing how well the different energy functions can detect remote homologues.

For these threading experiments, normalized energies were extensively used to compare performances between energy functions. The normalized energy is actually a z-score for a given sequence calculated with respect to a set of randomly generated sequences. A z-score is defined as $z-score = \frac{x - \overline{x}}{\sigma}$, where $x$ is the energy of the given sequence, $\overline{x}$ is the average energy of the randomly generated sequences, and $\sigma$ is the
standard deviation of energies of the same random set. Thus a normalized energy indicates the number of standard deviations away from the average energy of the randomly generated sequences. In addition, large negative z-score values indicate that energies of native sequences are favorable and well separated from those of the random sequences.

To determine the sufficient size of randomly generated sequences needed to calculate reliable normalized energies, Figure III.1 was plotted, where normalized energies for an SH3 domain calculated with the five energy functions are shown as functions of the number of randomly generated sequences. Figure III.1 indicates that using 1000 random sequences is enough to calculate normalized energies that are sufficiently close to the converged ones. In Figure III.2, five distributions of normalized energies, one for each energy function, are also shown. Each distribution was derived from 1000 random sequences. All of the distributions shown in the figure are normal. Based on these results, the probability of a sequence with certain energy can be calculated and compared across different energy functions.

The list of protein structures used in threading of native sequences was derived from the SCOP database to broadly sample various types of protein folds [50]. When preparing the list, only the first entry in each fold category was included. Furthermore, small and designed protein categories were not considered. Additional filtering of those structures that had unresolved or non-standard amino acid residues resulted in the final list containing 1603 unique proteins.

For the second threading experiment where distant homologues were used, the list of structures and their structurally aligned distant homologues were downloaded from the DALI server [51]. First 1600 entries were used for the analysis, after pruning away those sequences with non-standard amino acids.
Figure III.1: Normalized energies for the native sequence of an SH3 domain (PDB code = 1YP5) as a function of the number of random sequences used to calculate them. For each set of \( n \) random sequences generated, the sequences were threaded onto the structure, and their energies were calculated. Then the normalized energy (i.e. z-score) with respect to this set of random sequences was calculated for the native sequence.
Figure III.2: Distribution of normalized energies for the random sequences threaded onto an SH3 domain (PDB code = 1YP5) using the five energy functions. The energy functions used are $TPDF_{mono}$, $TPDF_{tri}$, $DPDF_{local}$, $SipplPair$, and $Solv$. Five normalized energy distributions are shown in this figure. Each distribution was generated from 1000 random sequences. It is apparent that the distributions are normal.
III.C Results

III.C.1 Distribution of shapes of mono-peptide and tri-peptide specific PDFs

Recently a number of papers have shown that nearest neighboring residues influence backbone angle preferences of the middle residue [41, 52, 42]. Here, we take this idea further and construct a 2-dimensional map which illustrates how tri-peptides specifying similar backbone angle preferences are clustered together (Figure III.3). For comparisons, mono-peptide specific PDFs are also represented in the figure with single letters of their residue types.

The map shown in Figure III.3 captures well known physical characteristics of amino acid residues, supporting its use as a tool for further analysis. For instance, the map is able to distinguish leucine from isoleucine by placing isoleucine far from leucine but close to valine. This is because both isoleucine and valine have branching C$\beta$ atoms while leucine does not. The presence of branching C$\beta$ atom imposes greater repulsive forces between neighboring residues, and thus there is a greater preference for beta-strand regions in the $\phi$-$\psi$ maps of isoleucine and valine.

Another observation that can be made about Figure III.3 is that mono-peptide specific PDFs are placed on the inner part of the map. This is because mono-peptide specific PDFs are averages of tri-peptide specific PDFs. The placement of mono-peptide specific PDFs on the map naturally leads one to wonder how the tri-peptides sharing the same middle residue are distributed on the map and what their PDF shapes are.

First, to illustrate which PDF shapes correspond to which regions on the map, PDFs of numerically labeled regions on the map in Figure III.3 are shown in Figure III.4. The sampled PDFs show a wide range of possible shapes, ranging from exclusively left-handed alpha-helix to variations of beta-strands as well as those in between. Also regions that are close together on the map display more similarities in PDF shapes than those that do not.

Second, to find out how the PDF shapes of tri-peptides sharing the same middle residue are distributed on the map, Figure III.5 was plotted for tri-peptides X-Ala-X,
X-Gly-X, X-Asn-X, X-Pro-X, and X-Val-X, where X represents any residue. As was expected, shapes of PDFs for each of these subsets of tri-peptides are loosely clustered in one region of the map. This observation demonstrates that the identities of middle residues largely specify which set of PDF shapes are sampled. In addition, we observe that some of the PDF shapes sampled by one subset of tri-peptides overlap with those of different subsets. For instance, PDF shapes of X-Ala-X tri-peptides (colored red) generally show preferences for a left-handed alpha-helix (e.g. subplots 1 and 2 in Figure III.4), but they can sample relatively a wide range of other shapes depending on the identities of their nearest neighboring residues. The overlap of PDF shapes between different subsets of tri-peptides reflects the observation that sequences with insignificant similarities can fold to the same structure [51].

We also studied distributions of PDF shapes for tri-peptides with either fixed first or third residue (e.g. Ala-X-X and X-X-Ala). Compared with the PDF shapes of tri-peptides with fixed middle residues (e.g. X-Ala-X), the distributions are more spread apart (data not shown), indicating that the middle residue has greater influence on specifying the backbone angle preferences than its nearest neighbors. An exception to this are tri-peptides with proline as their third residue (i.e. X-X-Pro). Interestingly, the distributions of PDF shapes for X-Pro-X and X-X-Pro appear to be well separated (Figure III.6). Although it was earlier observed that the placement of proline at the C-terminal positions of tri-peptides has much effect on the backbone angle preferences of the middle residue [42], Figure III.6 illustrates how extensive this influence can be as well as indicating which PDF shapes are preferred.

To validate the expanded set of shapes of tri-peptide specific PDFs with respect to those of mono-peptides, threading experiments were carried out, and the results are presented in the following section.

III.C.2 Threading Experiments

Two different threading experiments were carried out to determine whether an energy function derived from tri-peptide specific PDFs, $TPDF_{tri}$, performs better than
Figure III.3: A map of shapes of tri-peptide specific PDFs. Pair-wise distances for all pairs of PDFs were calculated and the relationships derived were mapped onto a plane using multi-dimensional scaling technique implemented in R statistical language [1]. In this figure, 6821 tri-peptides and 20 mono-peptide specific PDFs are represented as points. The remaining tri-peptides specific PDFs (i.e. 8000-6821 = 1179) are not represented because they had less than 30 data points, making them unreliable. Root Mean Square Distance was used as the metric for calculating distances for pairs of PDFs.
Figure III.4: A sample of PDF shapes for backbone dihedral angles of tri-peptides. Numbers labeling each subplot correspond to regions in Figure III.3. X- and y-axes represent $\phi$ and $\psi$ dihedral angles, respectively. Contour levels are fixed for all subplots in this figure to make comparisons between subplots meaningful.
Figure III.5: A map of shapes of tri-peptide specific PDFs with Alanine, Glycine, Asparagine, Proline, or Valine as their middle residues.
Figure III.6: The effect of residue position on the backbone angle preferences of the middle residue for proline at three different positions. The numbers of tri-peptides represented for Pro-X-X, X-Pro-X, and X-X-Pro are 356, 357, and 363, respectively.
the one derived from mono-peptide specific PDFs, $TPDF_{mono}$. In the first threading experiment, normalized energies were calculated for the native sequences to determine how well each energy function can separate native sequences from randomly generated sequences. To put the performances of these energy functions in perspective, those of other energy functions have been also measured using the same set of experiment. In Figure III.7, results from this threading experiment are shown.

As expected $TPDF_{tri}$ performs better than $TPDF_{mono}$ (i.e. many more data points lie below the diagonal), indicating that the nearest neighboring residues influence the middle residue’s backbone angle preferences. However, the increase in performance is, although significant, less than expected, yielding only -0.87 in average differences of energies (See $f$ in Figure III.7), for the two energy functions. Surprisingly, the figure also indicates that the remaining three energy functions perform worse than $TPDF_{mono}$. In addition, the normalized energies calculated by the five energy functions are highly correlated with respect to $TPDF_{mono}$ (i.e. $r > 0.7$).

Although Figure III.7 indicates that $TPDF_{mono}$ and $TPDF_{tri}$ are capable of distinguishing native sequences from randomly generated sequences, the results might be different in a more realistic threading setting where the goal is to detect distant homologues instead. To address this issue, structurally aligned distant homologues (i.e. < 35% sequence identity) were downloaded from the DALI server [51] and threaded onto their corresponding template structures. Results from this second threading experiment are shown in Figure III.8 for $TPDF_{mono}$, $TPDF_{tri}$, and other energy functions.

Figure III.8 also indicates that $TPDF_{tri}$ performs better than $TPDF_{mono}$, with a similar level of performance gain observed in the first threading experiment. However, in this second threading experiment, the performance of SipplPair improved to a level comparable to that of $TPDF_{mono}$. In addition, performances of $DPDF_{local}$ and $Solv$ are slightly better with respect to that of $TPDF_{mono}$.

Another change from the results of the first threading experiment is that there are much less correlations between energy functions. For SipplPair and Solv, their correlation coefficients with respect to $TPDF_{mono}$ are 0.31 and 0.46, respectively. These
Figure III.7: Results from the ungapped threading experiments using the five energy functions. For each energy function, an ungapped threading experiment was carried out using a list of template structures. The list of 1603 template structures was derived from the SCOP database. Normalized energies calculated with the energy functions shown on the y-axis were compared to those calculated with $TPDF_{mono}$ on the x-axis. More negative normalized energy indicates better performance. Thus for each subplot, the energy function shown on the y-axis is performing better than the one on the x-axis if greater number of points lie below the diagonal. This is indicated by a variable $f$, which is an average of differences of normalized energies from the two energy functions.
Figure III.8: Threading of distant homologues using structural alignments. Structural alignments of distant homologues from the DALI server were used to thread and assign normalized energies to them using the five energy functions. All of the distant homologues had < 35% sequence identity to the corresponding native sequences.
low levels of correlations indicate that there are certain groups of proteins for which $TPDF_{tri}$ detects more homologues than SipplPair or Solv, and vice versa. These results highlight the potential gains that can be achieved through the use of $TPDF_{tri}$ in genome-scale threading methods such as mGenTHREADER [53].

III.C.3 Estimating the fraction of foldable protein sequences

In this section, we further characterize $TPDF$ energy functions by estimating fractions of foldable sequences for a large number of protein structures. As a criterion for a sequence to fold, we simply require that the sequence has lower energy than that of the native sequence. This requirement is based on the observation that sequences with lower energies are more likely to adopt a given structure.

Because calculating the absolute number of foldable sequences is difficult due to the exponentially growing sequence space (i.e. $20^n$, where $n$ is the sequence length), fractions of foldable sequences are instead estimated. This is done by calculating the probability of seeing a sequence with energy comparable to that of the native sequence for a given structure and using this as an estimate of the fraction. Assuming that the distribution of energies of all sequences available to a structure is well approximated by a normal distribution, the probability is simply $P(z) = \frac{1}{\sqrt{2\pi}} \exp(-0.5z^2)$, where $z$ is a z-score. Since a normalized energy is in fact a z-score, normalized energies calculated for the first threading experiment were converted into probabilities using this equation. As shown in Figure III.2, estimating probabilities using a normal distribution function appears to be well supported. In addition, by plotting these probabilities for various proteins against their sequence lengths, fractions of foldable sequences can be compared between energy functions, as shown in Figure III.9.

Figure III.9 suggests two important properties of the estimated fractions. First, it suggests that the probability of observing a sequence with energy comparable to that of the native sequence exponentially decreases as sequence length increases for all five energy functions tested. This is probably not surprising since as sequence length increases, the growth of the number of unfavorable pair-wise residue interactions is much
Figure III.9: Correlations between estimated fractions of foldable sequences and sequence lengths for the five energy functions. Normalized energies calculated from the threading experiments were converted into log-scaled probabilities and plotted against sequence lengths of the corresponding proteins. For each subplot, a best fit line was drawn for the energy function indicated on the y-axis (solid line). Also, best fit line for $TPDF_{mono}$ is shown for comparison (dotted line) without showing its data points. Slopes for the energy functions indicated on the y-axis are also shown.
Figure III.10: Six proteins of length 100 residues with low and high low values of $-\log(P(x))$ are presented. The top row contains proteins with high $-\log(P(x))$ values (PDB codes = 1M2D, 1LN4, and 1QC7). The bottom row contains proteins with low $-\log(P(x))$ values (PDB codes = 1J2M, 1LRI, and 1QBH).

faster than that of the favorable interactions.

Second, Figure III.9 suggests that for protein structures with similar sequence lengths, their fractions vary widely. For example, sequences of lengths around 100 residues can have values of $-\log(P(x))$ that range from 30 to near 0 using any one of the five energy functions. Of course, a protein with $-\log(P(x))$ of near 0 means that the corresponding energy function cannot distinguish the native from random sequences. An examination of those structures with low and high values of $-\log(P(x))$ suggests that structures with $-\log(P(x))$ near zero tend to be those with metals and ligands as integral parts of the structure (See Figure III.10). Because the energy functions used in the present study do not take into account interactions with such external factors, structures containing them should have less favorable energies. Therefore, at least some of the spread of $-\log(P(x))$ at fixed sequence lengths probably resulted from inherent limitations of these energy functions.
III.D Discussion

The 2-dimensional map of PDF shapes presented in this study clearly illustrates the relationships between the sequences of tri-peptides and the backbone angles of their middle residues. Although only the first two components from the multidimensional scaling analysis were used to construct the map, a number of features of the map readily indicate that the map is a valid tool to study sequence-structure relationships for tri-peptides. For instance, the relative distances between mono-peptide specific PDFs reflect known physical properties of the 20 amino acid residues. In addition, it is easy to see that nearest neighboring residues influence backbone angle preferences of the middle residues, also observed by others [54, 41, 52, 42]. For instance, the map captures earlier findings that flanking residues with bulky side chains (i.e. Val, Ile, Tyr, and Phe) shift backbone angle preferences of the middle residue away from the helix regions [38]. In short, the map of PDF shapes presented in this paper is a concise visualization of the mapping between local sequences and their structures for tri-peptides. It would be worthwhile to determine how much agreement is there between the map and simulations of peptides using detailed energy functions [32].

To further validate the expanded set of PDF shapes of tri-peptides, two different threading experiments were carried out. Results of these experiments presented earlier demonstrated that tri-peptide specific PDFs capture information missing in mono-peptide specific PDFs, enabling them to perform better in detecting both native sequences and distant homologues. In addition, despite their simple forms, $TPDF_{\text{mono}}$ and $TPDF_{\text{tri}}$ perform at least as well as the three other energy functions tested in this study. Between the two different threading experiments, we noticed that there are much less correlations between energy functions for the detection of distant homologues. Also, in general, the energies of distant homologues threaded onto their template structures are lower than those of the native sequences (data not shown). These results indicate that a “fold” is in fact an ensemble of similar structures and that each one of these structures is sequence specific [55]. Accurately capturing these structural
variations is an area for improvement to develop higher performing fold recognition and refinement methods.

Lastly, as an exploratory study, the five energy functions were used to indirectly measure the fractions of foldable sequences for a large number of structures. Before these quantities were measured, however, a number of assumptions had to be made to deal with the exponentially growing sequence space. It was assumed that the fractions are correlated with the probabilities derived from normalized energies. Also, it was assumed that the fixed template structures represent their corresponding “folds”. In light of these approximations, it was observed that the fractions of foldable sequences decrease exponentially as sequence lengths increase for all five energy functions used.

Recently, others have estimated related quantity called protein designability [56, 57]. The designability of a protein is defined as the number of sequences out of all possible sequences that uniquely fold to the given structure [58]. Their estimated protein designability scores were presented without considering sequence lengths. The results shown earlier suggest that sequence length is an important determinant of protein designability.

III.E Conclusion

Tri-peptide specific PDFs and mono-peptide specific PDFs have been characterized using multi-dimensional scaling and threading experiments. The map of PDF shapes constructed in this study enabled the visualization of sequence-structure relationships for tri-peptides. Using this map, it was illustrated how extensively the middle residues determine which PDF shapes are sampled.

As a validation of the expanded PDF shapes of tri-peptides, two different threading experiments were carried out to benchmark $TPDF_{tri}$, $TPDF_{mono}$, and three other energy functions. Based on the correlations of energies between energy functions, $TPDF_{tri}$ performs better than $TPDF_{mono}$ by about one standard deviation on average.

Lastly, fractions of foldable sequences were estimated for proteins from the
SCOP database using the five energy functions. The measured quantities indicate that as sequence lengths increase, fractions of foldable sequence decrease exponentially. Furthermore, surprisingly wide distributions of these fractions were observed for proteins of same lengths.
Using known structural homologues to model backbone flexibility in a protein design algorithm

IV.A  Introduction

In recent years, the field of protein design has enjoyed a number of astounding successes. By using an all-atom energy function and searching for sequences compatible with a given protein backbone, a number of groups have demonstrated that previously unobserved proteins with new properties can be constructed. Some of the notable examples include a zinc fold that does not require the presence of a zinc [59], right-handed coiled-coil oligomers with a superhelical twist [60], and de novo design of a novel backbone structure named Top7 [61].

Another growing uses of protein design algorithms have been in predicting various properies of naturally occurring protein sequences. For instance, a number of investigators have shown that the designed sequences share significant sequence identities with their corresponding native sequences at both whole protein and core residues levels ([62, 63]), yielding 30% and 50%, respectively. The high native sequence recapitulation rates suggest that thermodynamic is one of the major forces shaping pro-
tein sequence space [63]. In addition, using only profiles generated from the designed sequences for various structures, a number of investigators have shown that native sequences as well as their homologues can be detected with high confidence ([62, 64, 55]). For instance, the large-scale fold recognition study by Larson et al. [55] achieved significant native sequence recognitions for about half of the 264 small protein structures they used. These results point toward a direction where rational protein design algorithms can be used to simulate evolution of proteins and shed insights into how existing protein sequences have resulted.

Despite these successes, existing protein design algorithms still do not faithfully capture some of the forces that shape protein sequence space. For instance, predicting conserved residues using designed sequences is still difficult due to complications rising from the effects of both structural and functional requirements [65, 66]. In addition, predicting the volume of sequence space for protein structures is at an early stage [55, 67].

To develop better performing protein design algorithms, a number of studies indicate that inclusion of backbone flexibility into the algorithms is necessary [55, 67]. In Larson et al. [55], backbone flexibility was introduced by using an ensemble of perturbed structures around the native structure. The protein design algorithm was then applied to each structure from the ensemble. In Saunders et al. [67], iterations of backbone perturbation and sequence design steps were implemented.

In the present study, backbone flexibility is also introduced into a protein design algorithm. However, unlike previous algorithms, backbone flexibility is modeled by designing sequences for known structural homologues of a protein fold. This approach is motivated by an idea that if a protein design algorithm can identify those residues important in determining a structure, designed sequences for its known structural homologues as a whole should encode amplified signals of these important residues.

Furthermore, using known structural homologues to model backbone flexibility has the obvious benefit of bypassing the difficult task of sampling structural perturbations around the native structure. In addition, the ensemble of structures used already
includes gaps and insertions that are hallmarks of naturally occurring sequences. Thus
known structural homologues may perhaps be a more realistic model of backbone flex-
ibility in a protein design algorithm.

The hypothesis of the present study is that designed sequences for known
structural homologues should be more "native-like" than those for an individual struc-
ture. We measure native-likeness by carrying out fold recognition experiments with
Multiple Sequence Alignments (MSAs) of designed sequences. Then we characterize
these MSAs by comparing their conservation profiles with those of naturally occurring
sequences.

IV.B Methods

Protein sequences were designed for each structure from the sets of homolo-
gous proteins shown in Figure IV.1. The superimposed structures and their structurally
aligned sequences were downloaded from the HOMSTRAD database [68]. Multiple
sequence alignments of designed sequences for each template structure and those of
corresponding natural counterparts were later characterized. In the following, the de-
tails of the protein design protocol and the sequence analysis techniques are provided.

IV.B.1 Protein Design Protocol

The protein design protocol consists of two parts: (1) an all-atom energy func-
tion and (2) sampling of sequence space. To limit the search space, discrete represen-
tations of sidechains were used [69]. In addition, the backbone of the protein structure
was fixed, further reducing the search space. Guided by an all-atom energy function,
simulated annealing [70] was used to search for the set of sidechain rotamers (and thus
the sequence) compatible with the backbone structure. For each template structure, 100
sequences were designed. In the following sections, details on the energy function and
the construction of energy lookup tables crucial for an efficient run of a design process
are described.
All-atom Energy Function

The energy function shown in Equation IV.1 was used to calculate an energy for a set of rotamers placed onto a fixed backbone of a structure. The energy function was derived from that of Kuhlman et al.[61].

\[
E_{\text{protein}} = W_{\text{atr}}E_{\text{atr}} + W_{\text{rep}}E_{\text{rep}} + W_{\text{solv}}E_{\text{solv}} + W_{\text{hb}}E_{\text{hb}} + W_{\text{tpdf}}E_{\text{tpdf}} + W_{\text{pair}}E_{\text{pair}} + E_{\text{ref}} \tag{IV.1}
\]

\[
E_{\text{atr}} = \sum \sum \left[ \left( \frac{r_{ij}}{d_{ij}} \right)^{12} - \left( \frac{r_{ij}}{d_{ij}} \right)^{6} \right] \tag{IV.2}
\]

\[
E_{\text{rep}} = \sum \sum y_{\text{intercept}} - d_{ij}slope \tag{IV.3}
\]

\[
slope = -12e_{ij}(1.33^{13} - 1.33^{7})(1/r_{ij}) \]

\[
y_{\text{intercept}} = -slope \frac{r_{ij}}{1.33} + e_{ij}(1.33^{12} - 2(1.33)^6) \]

\[
E_{\text{solv}} = \sum \sum \left[ \frac{-2\Delta G_{i}^{\text{free}}}{4\pi\sqrt{\pi}\lambda_{i}r_{ij}^{2}} \exp(-d_{ij}^{2})V_{j} + \frac{-2\Delta G_{j}^{\text{free}}}{4\pi\sqrt{\pi}\lambda_{j}r_{ij}^{2}} \exp(-d_{ji}^{2})V_{i} \right] \tag{IV.4}
\]

\[
E_{\text{tpdf}} = \sum \log(1 + \text{prob}(\phi, \psi|\text{tripeptide}(i))) \tag{IV.5}
\]

\[
E_{\text{pair}} = \sum \sum \log(\text{prob}(d_{ij}|\text{aa}(i), \text{aa}(j))) \tag{IV.6}
\]

\[
E_{\text{ref}} = \sum W_{\text{ref}}(\text{aa}(i)) \tag{IV.7}
\]

The energy function consists of seven energy terms with corresponding weights. The energy terms are composed of attractive portion of the Lennard-Jones function and a linear steric interaction term in Equations IV.2 and IV.3, Lazaridis-Karplus solvation model [71] in Equation IV.4, orientation-dependent hydrogen-bonding, \(E_{\text{hb}}\), as described in [72], torsion-angle Probability Density Function in Equation IV.5, residue-residue pair-wise energy in Equation IV.6, and reference energies for the 20 residues in Equation IV.7. The residue reference energy, \(E_{\text{ref}}\), controls the residue composition of the designed sequences. Without it, residues with more atoms (e.g. arginine and
tryptophan) tend to dominate in the final designed sequences.

**Training the Energy Function**

The weights of the energy terms were optimized using the technique described in [63] and a training set of 20 high-resolution structures. Briefly, conjugate gradient [73] was used to maximize the cost function defined in Equation IV.8 over the weight space for each residue position of a structure in the training set. The nominator term, \( \exp(-E_{\text{native}}) \), is an exponential of an energy of the native sequence. The denominator term, \( \sum_i \exp(-E_i) \), is a sum of exponentials of energies of the rotamer substituted sequences of the native sequence. By maximizing the cost function for each residue position, energy weights that favor native sequences with their rotamers were generated.

\[
P = \frac{\exp(-E_{\text{native}})}{\sum_i \exp(-E_i)}
\]  

(IV.8)

**Energy Lookup Tables**

Energy lookup tables, which store pre-calculated values of pair-wise energy terms, were used during the design process to achieve orders of magnitude of speed improvements. Construction of energy lookup tables are possible due to discrete representation of sidechain rotamers and the pair-wise decomposability of the energy terms in Equation IV.1. For a protein with 80 residues, a single protein sequence can be designed within 5 minutes on a single processor on a standard workstation. This would have taken a few hours if energy lookup tables were not used.

**Optimizing Sequence-Structure Relationships**

Simulated annealing [70] was used to optimize sequence-structure compatibility using the energy function in Equation IV.1. Starting with the temperature at 100, an exponential scheduling was used to gradually decrease the temperature. A move consisted of randomly choosing a residue position to mutate, followed by randomly choos-
ing a rotamer. The best solution found during a simulated annealing run was returned as
the final solution.

**IV.B.2 Preparing the Multiple Sequence Alignments**

Four different types of multiple sequence alignments (MSAs) were characterized in this work. They are (1) Designed, (2) Combined Designed, (3) Naturally occurring, and (4) Structurally aligned sequences.

An MSA of (1) Designed sequences refers to that of 100 designed sequences for a single structure. Since all these sequences were generated from the same structure, the designed sequences were already aligned.

An MSA of (2) Combined Designed sequences refers to all of the designed sequences of structural homologues for a given protein fold. All of these designed sequences were aligned to each other based on the structurally aligned native sequences of the homologous structures.

An MSA of (3) Naturally occurring sequences refers to those collected using 2 rounds of PSI-BLAST using the native sequence as the probe. The resulting set of sequences were then aligned using the program MUSCLE [74].

Finally, an MSA of (4) Structurally aligned sequences refers to that of the structural homologues shown in Figure IV.1 and Table IV.1.

**IV.B.3 Fold Recognition Using PSI-BLAST**

Fold recognition experiments were carried out using PSI-BLAST with -B option and an MSA prepared as described in the previous section. PSI-BLAST generates a Position Specific Scoring Matrix (PSSM) from the given MSA and uses this PSSM to search for any significant matches [16].

**IV.B.4 Conservation Profiles for Designed and Naturally Occurring Sequences**

Sequence analysis method developed by Lockless and Ranganathan [75] was used to derive conservation profiles of MSAs of the designed sequences and their nat-
urally occurring counterparts. Given an MSA, its conservation profile is defined by Equation IV.9. In a conservation profile, a value $\Delta G_{i}^{\text{stat}}$ is associated with a residue position, indicating how much the distribution of 20 residue types, $P_{i}^{x}$, at the residue position $i$ is different from that (i.e. $P_{MSA}^{x}$) of the Swiss-Prot protein sequence database. Higher value indicates, higher degree of conservation. Thus each conservation profile is a measure of how the sequence space is shaped by an underlying source.

The probability of each residue type was calculated using the binomial density function in Equation IV.10. The variable $N$ is the total number of sequences in the given MSA, $n_{x}$ is the number of sequences with amino acid $x$, $p_{x}$ is the probability of the same amino acid observed in the Swiss Prot sequence database. To compare conservation profiles of MSAs with different number of sequences, their variables $N$ and $n_{x}$ were scaled to those of an MSA with total of 100 sequences.

$$\Delta G_{i}^{\text{stat}} = kT \sum_{x} \left( \ln \frac{P_{i}^{x}}{P_{MSA}^{x}} \right)^{2}$$  \hspace{1cm} (IV.9)

$$P(x) = \frac{N!}{n_{x}!(N-n_{x})!}p_{x}^{n_{x}}(1-p_{x})^{N-n_{x}}$$  \hspace{1cm} (IV.10)

### IV.C Results

Protein sequences were designed for the structural homologues presented in Figure IV.1 and Table IV.1 using the design protocol described earlier. The three sets of homologous structures were downloaded from the HOMSTRAD database [68]. They were chosen because they were relatively small, did not have metal ions as integral parts of their structure, and had relatively a large number of structural homologues. For each structure, 100 designed sequences were generated. In the following sections, various properties of MSAs derived from these designed sequences are described and compared with MSAs of their natural counterparts.
Table IV.1: The set of homologous protein structures used for designing proteins.

<table>
<thead>
<tr>
<th>Name</th>
<th>No. Struct.</th>
<th>Avg. Length</th>
<th>Avg. % ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>α-amylase C</td>
<td>21</td>
<td>85</td>
<td>21</td>
<td>breaks down sugars</td>
</tr>
<tr>
<td>RRM/RBP/RNP</td>
<td>20</td>
<td>87</td>
<td>23</td>
<td>RNA binding protein</td>
</tr>
<tr>
<td>SH3</td>
<td>20</td>
<td>63</td>
<td>32</td>
<td>interaction module</td>
</tr>
</tbody>
</table>

IV.C.1 Sequence identities shared between the designed and the native sequences of the structural homologues

For a large number of structures, their native sequence recapitulation rates are similar to those reported in [63, 62]. As shown in Tables IV.2, IV.3, and IV.4, native sequence recapitulation rates can be around 30% and 50% for whole protein and core residues, respectively. These results indicate that the performance of the protein design algorithm used here is comparable to the ones reported earlier.

Interestingly, structure-dependent effects on the recapitulation rates of native sequences are seen for all three sets of structural homologues, as shown in Tables IV.2, IV.3, and IV.4. The structural homologues of α-amylase had much better native sequence recapitulation rates than those of RRM and SH3 domains. Even among structural homologues of a protein fold, significant differences in native sequence recapitulation rates can be seen. For instance, native sequence recapitulation rates at the level of whole protein vary from 13% - 32% for the RRM domain and from 12% - 32% for the SH3 domain.
Table IV.2: Protein design results for the C-terminal domain of α-amylase. For each structure labeled with a Protein Data Bank ID, % sequence identities shared with the native sequences at the levels of whole protein and core residues are shown in the second and third columns, respectively. The fourth column, %ID P., is average pair-wise % sequence identities, indicating how similar each designed sequence is to each other. Also shown in fifth and sixth columns are fraction of polar and nonpolar residues for naturally occurring sequences and designed sequences. Finally, fold recognition results for Natural MSAs and Designed MSAs are shown in the last two columns.

<table>
<thead>
<tr>
<th>PDB ID</th>
<th>%ID W.</th>
<th>%ID C.</th>
<th>%ID P.</th>
<th>polar/nonpolar Nat.</th>
<th>polar/nonpolar Des.</th>
<th>foldRecog. Nat.</th>
<th>foldRecog. Des.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1HVXA</td>
<td>32</td>
<td>38</td>
<td>60</td>
<td>0.67/0.33</td>
<td>0.45/0.55</td>
<td>5/21</td>
<td>2/21</td>
</tr>
<tr>
<td>1VJS</td>
<td>33</td>
<td>47</td>
<td>58</td>
<td>0.66/0.34</td>
<td>0.47/0.53</td>
<td>6/21</td>
<td>2/21</td>
</tr>
<tr>
<td>1QHPA</td>
<td>27</td>
<td>36</td>
<td>59</td>
<td>0.62/0.38</td>
<td>0.38/0.62</td>
<td>10/21</td>
<td>6/21</td>
</tr>
<tr>
<td>1PAMA</td>
<td>32</td>
<td>38</td>
<td>57</td>
<td>0.59/0.41</td>
<td>0.40/0.60</td>
<td>10/21</td>
<td>6/21</td>
</tr>
<tr>
<td>1D3CA</td>
<td>28</td>
<td>33</td>
<td>57</td>
<td>0.58/0.42</td>
<td>0.42/0.58</td>
<td>10/21</td>
<td>6/21</td>
</tr>
<tr>
<td>1CGT</td>
<td>28</td>
<td>36</td>
<td>57</td>
<td>0.64/0.36</td>
<td>0.37/0.63</td>
<td>10/21</td>
<td>6/21</td>
</tr>
<tr>
<td>1CYG</td>
<td>29</td>
<td>25</td>
<td>54</td>
<td>0.63/0.37</td>
<td>0.43/0.57</td>
<td>11/21</td>
<td>6/21</td>
</tr>
<tr>
<td>1BAG</td>
<td>33</td>
<td>50</td>
<td>62</td>
<td>0.64/0.36</td>
<td>0.41/0.59</td>
<td>12/21</td>
<td>3/21</td>
</tr>
<tr>
<td>1SMD</td>
<td>29</td>
<td>43</td>
<td>57</td>
<td>0.63/0.37</td>
<td>0.41/0.59</td>
<td>12/21</td>
<td>4/21</td>
</tr>
<tr>
<td>1HX0A</td>
<td>30</td>
<td>48</td>
<td>59</td>
<td>0.63/0.37</td>
<td>0.44/0.56</td>
<td>11/21</td>
<td>4/21</td>
</tr>
<tr>
<td>1JAE</td>
<td>30</td>
<td>48</td>
<td>58</td>
<td>0.67/0.33</td>
<td>0.43/0.57</td>
<td>11/21</td>
<td>5/21</td>
</tr>
<tr>
<td>1G94A</td>
<td>22</td>
<td>29</td>
<td>52</td>
<td>0.64/0.36</td>
<td>0.45/0.55</td>
<td>13/21</td>
<td>3/21</td>
</tr>
<tr>
<td>2AAA</td>
<td>25</td>
<td>20</td>
<td>57</td>
<td>0.64/0.36</td>
<td>0.32/0.68</td>
<td>3/21</td>
<td>2/21</td>
</tr>
<tr>
<td>1AVAA</td>
<td>31</td>
<td>24</td>
<td>60</td>
<td>0.54/0.46</td>
<td>0.33/0.67</td>
<td>2/21</td>
<td>1/21</td>
</tr>
<tr>
<td>7TAA</td>
<td>25</td>
<td>30</td>
<td>56</td>
<td>0.64/0.36</td>
<td>0.33/0.67</td>
<td>5/21</td>
<td>2/21</td>
</tr>
<tr>
<td>1GJWA</td>
<td>27</td>
<td>47</td>
<td>60</td>
<td>0.59/0.41</td>
<td>0.37/0.63</td>
<td>1/21</td>
<td>3/21</td>
</tr>
<tr>
<td>1UOK</td>
<td>30</td>
<td>44</td>
<td>59</td>
<td>0.54/0.46</td>
<td>0.33/0.67</td>
<td>14/21</td>
<td>2/21</td>
</tr>
<tr>
<td>1G5AA</td>
<td>27</td>
<td>49</td>
<td>59</td>
<td>0.54/0.46</td>
<td>0.30/0.70</td>
<td>7/21</td>
<td>2/21</td>
</tr>
<tr>
<td>1BVZA</td>
<td>18</td>
<td>24</td>
<td>56</td>
<td>0.64/0.36</td>
<td>0.32/0.68</td>
<td>10/21</td>
<td>1/21</td>
</tr>
<tr>
<td>1CIU</td>
<td>30</td>
<td>33</td>
<td>55</td>
<td>0.61/0.39</td>
<td>0.39/0.61</td>
<td>10/21</td>
<td>6/21</td>
</tr>
</tbody>
</table>
Table IV.3: Protein design results for RNA recognition motif domain. For each structure labeled with a Protein Data Bank ID, % sequence identities shared with the native sequences at the levels of whole protein and core residues are shown in the second and third columns, respectively. The fourth column, %ID P., is average pair-wise % sequence identities, indicating how similar each designed sequence is to each other. Also shown in fifth and sixth columns are fraction of polar and nonpolar residues for naturally occurring sequences and designed sequences. Finally, fold recognition results for Natural MSAs and Designed MSAs are shown in the last two columns.

<table>
<thead>
<tr>
<th>PDB ID</th>
<th>%ID W.</th>
<th>%ID C.</th>
<th>%ID P.</th>
<th>polar/nonpolar Nat.</th>
<th>polar/nonpolar Des.</th>
<th>foldRecog. Nat.</th>
<th>foldRecog. Des.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1FJEB1</td>
<td>17</td>
<td>14</td>
<td>61</td>
<td>0.59/0.41</td>
<td>0.56/0.44</td>
<td>20/20</td>
<td>0/20</td>
</tr>
<tr>
<td>1IQTA</td>
<td>13</td>
<td>9</td>
<td>59</td>
<td>0.64/0.36</td>
<td>0.42/0.58</td>
<td>20/20</td>
<td>0/20</td>
</tr>
<tr>
<td>2U2FA</td>
<td>18</td>
<td>15</td>
<td>53</td>
<td>0.56/0.44</td>
<td>0.48/0.52</td>
<td>20/20</td>
<td>0/20</td>
</tr>
<tr>
<td>1CVJA1</td>
<td>28</td>
<td>48</td>
<td>59</td>
<td>0.54/0.46</td>
<td>0.36/0.64</td>
<td>20/20</td>
<td>8/20</td>
</tr>
<tr>
<td>2U1A</td>
<td>14</td>
<td>22</td>
<td>54</td>
<td>0.49/0.51</td>
<td>0.44/0.56</td>
<td>20/20</td>
<td>0/20</td>
</tr>
<tr>
<td>1QM9A2</td>
<td>15</td>
<td>15</td>
<td>59</td>
<td>0.61/0.39</td>
<td>0.46/0.54</td>
<td>20/20</td>
<td>0/20</td>
</tr>
<tr>
<td>1QM9A1</td>
<td>21</td>
<td>43</td>
<td>60</td>
<td>0.57/0.43</td>
<td>0.38/0.62</td>
<td>20/20</td>
<td>0/20</td>
</tr>
<tr>
<td>1U2FA</td>
<td>19</td>
<td>20</td>
<td>55</td>
<td>0.52/0.48</td>
<td>0.45/0.55</td>
<td>20/20</td>
<td>0/20</td>
</tr>
<tr>
<td>1FJEB2</td>
<td>18</td>
<td>21</td>
<td>55</td>
<td>0.67/0.33</td>
<td>0.42/0.58</td>
<td>20/20</td>
<td>0/20</td>
</tr>
<tr>
<td>1OO0B</td>
<td>26</td>
<td>42</td>
<td>57</td>
<td>0.62/0.38</td>
<td>0.35/0.65</td>
<td>20/20</td>
<td>6/20</td>
</tr>
<tr>
<td>1FXLA1</td>
<td>32</td>
<td>49</td>
<td>60</td>
<td>0.60/0.40</td>
<td>0.36/0.64</td>
<td>20/20</td>
<td>6/20</td>
</tr>
<tr>
<td>1FXLA2</td>
<td>30</td>
<td>49</td>
<td>61</td>
<td>0.65/0.35</td>
<td>0.38/0.62</td>
<td>20/20</td>
<td>9/20</td>
</tr>
<tr>
<td>1L3KA1</td>
<td>29</td>
<td>37</td>
<td>63</td>
<td>0.62/0.38</td>
<td>0.33/0.67</td>
<td>20/20</td>
<td>1/20</td>
</tr>
<tr>
<td>2MSTA</td>
<td>15</td>
<td>16</td>
<td>58</td>
<td>0.63/0.37</td>
<td>0.55/0.45</td>
<td>20/20</td>
<td>2/20</td>
</tr>
<tr>
<td>1HD1A</td>
<td>16</td>
<td>17</td>
<td>56</td>
<td>0.64/0.36</td>
<td>0.49/0.51</td>
<td>20/20</td>
<td>0/20</td>
</tr>
</tbody>
</table>
Table IV.4: Protein design results for SH3 domain. For each structure labeled with a Protein Data Bank ID, % sequence identities shared with the native sequences at the levels of whole protein and core residues are shown in the second and third columns, respectively. The fourth column, %ID P., is average pair-wise % sequence identities, indicating how similar each designed sequence is to each other. Also shown in fifth and sixth columns are fraction of polar and nonpolar residues for naturally occurring sequences and designed sequences. Finally, fold recognition results for Natural MSAs and Designed MSAs are shown in the last two columns.

<table>
<thead>
<tr>
<th>PDB ID</th>
<th>%ID W.</th>
<th>%ID C.</th>
<th>%ID P.</th>
<th>polar/nonpolar Nat.</th>
<th>polar/nonpolar Des.</th>
<th>foldRecog. Nat.</th>
<th>foldRecog. Des.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1AWJ</td>
<td>12</td>
<td>23</td>
<td>65</td>
<td>0.66/0.34</td>
<td>0.57/0.43</td>
<td>18/20</td>
<td>1/20</td>
</tr>
<tr>
<td>1SHG</td>
<td>21</td>
<td>33</td>
<td>54</td>
<td>0.60/0.40</td>
<td>0.28/0.72</td>
<td>19/20</td>
<td>0/20</td>
</tr>
<tr>
<td>1CSKA</td>
<td>23</td>
<td>20</td>
<td>56</td>
<td>0.61/0.39</td>
<td>0.38/0.62</td>
<td>18/20</td>
<td>0/20</td>
</tr>
<tr>
<td>1CKAA</td>
<td>23</td>
<td>36</td>
<td>64</td>
<td>0.61/0.39</td>
<td>0.34/0.66</td>
<td>18/20</td>
<td>1/20</td>
</tr>
<tr>
<td>1GRIA2</td>
<td>21</td>
<td>12</td>
<td>58</td>
<td>0.61/0.39</td>
<td>0.34/0.66</td>
<td>17/20</td>
<td>2/20</td>
</tr>
<tr>
<td>2HSP</td>
<td>17</td>
<td>26</td>
<td>62</td>
<td>0.63/0.37</td>
<td>0.50/0.50</td>
<td>18/20</td>
<td>0/20</td>
</tr>
<tr>
<td>1ARK</td>
<td>17</td>
<td>21</td>
<td>57</td>
<td>0.55/0.45</td>
<td>0.58/0.42</td>
<td>17/20</td>
<td>0/20</td>
</tr>
<tr>
<td>1AOJA</td>
<td>18</td>
<td>46</td>
<td>65</td>
<td>0.62/0.38</td>
<td>0.33/0.67</td>
<td>18/20</td>
<td>0/20</td>
</tr>
<tr>
<td>1PHT</td>
<td>32</td>
<td>50</td>
<td>62</td>
<td>0.69/0.31</td>
<td>0.43/0.57</td>
<td>2/20</td>
<td>3/20</td>
</tr>
<tr>
<td>1SHFA</td>
<td>23</td>
<td>22</td>
<td>55</td>
<td>0.63/0.37</td>
<td>0.36/0.64</td>
<td>17/20</td>
<td>3/20</td>
</tr>
<tr>
<td>2SRC</td>
<td>23</td>
<td>34</td>
<td>61</td>
<td>0.66/0.34</td>
<td>0.36/0.64</td>
<td>18/20</td>
<td>2/20</td>
</tr>
<tr>
<td>1LCKA</td>
<td>22</td>
<td>28</td>
<td>52</td>
<td>0.61/0.39</td>
<td>0.39/0.61</td>
<td>18/20</td>
<td>2/20</td>
</tr>
<tr>
<td>1QLYA</td>
<td>21</td>
<td>30</td>
<td>61</td>
<td>0.60/0.40</td>
<td>0.44/0.56</td>
<td>18/20</td>
<td>0/20</td>
</tr>
<tr>
<td>1GBRA</td>
<td>20</td>
<td>29</td>
<td>68</td>
<td>0.62/0.38</td>
<td>0.50/0.50</td>
<td>17/20</td>
<td>0/20</td>
</tr>
</tbody>
</table>
**IV.C.2 Fold recognition experiments**

Various types of Multiple Sequence Alignments (MSAs), including those of designed sequences, described in the Methods section were entered into PSI-BLAST, and fold recognition experiments were carried out. The number of known structural homologues detected are shown in the last two columns of Tables IV.2, IV.3, IV.4. Not surprisingly, MSAs of naturally occurring sequences (Natural MSAs) consistently detect greater number of homologues than those of designed sequences (Designed MSAs). Interestingly, Designed MSAs of $\alpha$-amylase had greater number of known homologues detected than those of RRM and SH3 domains. This is probably because the designed sequences of $\alpha$-amylase had much better native sequence recapitulation rates.

In Tables IV.5, IV.6, and IV.7, detailed results from fold recognition experiments for the three structures are presented. In addition to the fold recognition results for Designed and Natural MSAs, results for combined designed and structurally aligned sequences are also presented (Combined MSAs and Structure MSAs, respectively). As was expected, Designed MSAs can detect their corresponding native sequences with high confidence, except in the case of SH3 domain. However, only a small fraction of other homologues were detected by Designed MSAs. Combined MSAs did not do better, suggesting that the design algorithm is not correctly identifying important residues of protein folds. On the other hand, Natural MSAs detected many more homologues than either Designed or Combined MSAs. Finally, Structure MSAs did even better than Natural MSAs, indicating that structurally aligned MSAs contain more information specifying a protein fold.

**IV.C.3 Conservation profiles of the designed and naturally occurring sequences**

Conservation profiles for the designed and naturally occurring sequences of the three structures are shown in Figures IV.2, IV.3, and IV.4. Based on these plots, it is clear that the current implementation of protein design algorithm generates very different sets of sequences with respect to their natural counterparts. No correlation was found between the conservation profiles of designed and naturally occurring sequences.
Table IV.5: Fold recognition results for an $\alpha$-amylase C-terminal domain with PDB ID = 1BAG. Fold recognition results using Designed, Combined, Natural, and Structure MSAs are shown. The definitions of the four different types of MSAs are provided in the Methods section.

<table>
<thead>
<tr>
<th>PDB ID</th>
<th>% ID</th>
<th>Designed</th>
<th>Combined</th>
<th>Natural</th>
<th>Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>1hvx</td>
<td>18</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>3E-023</td>
</tr>
<tr>
<td>1vjs</td>
<td>12</td>
<td>-</td>
<td>4E-002</td>
<td>-</td>
<td>1E-021</td>
</tr>
<tr>
<td>2aaa</td>
<td>10</td>
<td>-</td>
<td>-</td>
<td>8.9</td>
<td>-</td>
</tr>
<tr>
<td>7taa</td>
<td>13</td>
<td>-</td>
<td>1E+002</td>
<td>17</td>
<td>3E-009</td>
</tr>
<tr>
<td>1gjw</td>
<td>13</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>6E-002</td>
</tr>
<tr>
<td>1sma</td>
<td>6</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>1uok</td>
<td>14</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>8E-005</td>
</tr>
<tr>
<td>1g5a</td>
<td>12</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>4E-007</td>
</tr>
<tr>
<td>1bvz</td>
<td>6</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>7E-010</td>
</tr>
<tr>
<td>1ciu</td>
<td>31</td>
<td>-</td>
<td>1E+000</td>
<td>3.1</td>
<td>7E-012</td>
</tr>
<tr>
<td>1qhp</td>
<td>24</td>
<td>6.2</td>
<td>1E+002</td>
<td>57</td>
<td>6E-009</td>
</tr>
<tr>
<td>1pam</td>
<td>26</td>
<td>-</td>
<td>2E+000</td>
<td>8.6</td>
<td>3E-011</td>
</tr>
<tr>
<td>1d3c</td>
<td>28</td>
<td>-</td>
<td>-</td>
<td>68</td>
<td>3E-010</td>
</tr>
<tr>
<td>1cgt</td>
<td>27</td>
<td>-</td>
<td>2E+002</td>
<td>-</td>
<td>2E-009</td>
</tr>
<tr>
<td>1cyg</td>
<td>27</td>
<td>19</td>
<td>-</td>
<td>1.3</td>
<td>3E-011</td>
</tr>
<tr>
<td>1bag</td>
<td>100</td>
<td>2E-009</td>
<td>-</td>
<td>4E-014</td>
<td>2E-002</td>
</tr>
<tr>
<td>1smd</td>
<td>22</td>
<td>-</td>
<td>3E+002</td>
<td>2E-023</td>
<td>9E-008</td>
</tr>
<tr>
<td>1hx0</td>
<td>21</td>
<td>-</td>
<td>-</td>
<td>7E-024</td>
<td>9E-009</td>
</tr>
<tr>
<td>1jae</td>
<td>22</td>
<td>-</td>
<td>-</td>
<td>9E-021</td>
<td>3E-008</td>
</tr>
<tr>
<td>1g94</td>
<td>22</td>
<td>-</td>
<td>-</td>
<td>1E-014</td>
<td>7E-009</td>
</tr>
<tr>
<td>1ava</td>
<td>9</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
Table IV.6: Fold recognition results for an RNA Recognition Motif domain with PDB ID = 1FXL. Fold recognition results using Designed, Combined, Natural, and Structure MSAs are shown. The definitions of the four different types of MSAs are provided in the Methods section.

<table>
<thead>
<tr>
<th>PDB ID</th>
<th>% ID</th>
<th>Designed</th>
<th>Combined</th>
<th>Natural</th>
<th>Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>1jmt</td>
<td>15</td>
<td>-</td>
<td>-</td>
<td>2E-001</td>
<td>2E-007</td>
</tr>
<tr>
<td>1qm9</td>
<td>19</td>
<td>-</td>
<td>-</td>
<td>1E-005</td>
<td>1E-017</td>
</tr>
<tr>
<td>1a9n</td>
<td>22</td>
<td>-</td>
<td>-</td>
<td>2E-005</td>
<td>5E-015</td>
</tr>
<tr>
<td>1nu4</td>
<td>21</td>
<td>-</td>
<td>-</td>
<td>1E-006</td>
<td>4E-015</td>
</tr>
<tr>
<td>2v1a</td>
<td>21</td>
<td>-</td>
<td>-</td>
<td>3E-002</td>
<td>2E-021</td>
</tr>
<tr>
<td>1qm9</td>
<td>24</td>
<td>-</td>
<td>-</td>
<td>1E-005</td>
<td>1E-017</td>
</tr>
<tr>
<td>1hd1</td>
<td>29</td>
<td>4E+000</td>
<td>-</td>
<td>1E-011</td>
<td>1E-013</td>
</tr>
<tr>
<td>1l3k</td>
<td>25</td>
<td>1E-001</td>
<td>-</td>
<td>6E-011</td>
<td>2E-014</td>
</tr>
<tr>
<td>2mst</td>
<td>23</td>
<td>-</td>
<td>-</td>
<td>1E-007</td>
<td>2E-012</td>
</tr>
<tr>
<td>1iqt</td>
<td>19</td>
<td>-</td>
<td>-</td>
<td>3E-005</td>
<td>9E-012</td>
</tr>
<tr>
<td>2v2f</td>
<td>25</td>
<td>-</td>
<td>-</td>
<td>3E-009</td>
<td>5E-016</td>
</tr>
<tr>
<td>1oo0</td>
<td>24</td>
<td>-</td>
<td>-</td>
<td>6E-009</td>
<td>1E-016</td>
</tr>
<tr>
<td>1cvj</td>
<td>29</td>
<td>-</td>
<td>-</td>
<td>3E-014</td>
<td>5E-017</td>
</tr>
<tr>
<td>1h6k</td>
<td>26</td>
<td>-</td>
<td>-</td>
<td>2E-008</td>
<td>2E-017</td>
</tr>
<tr>
<td>1fxl</td>
<td>37</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>1fxl</td>
<td>100</td>
<td>4E-010</td>
<td>-</td>
<td>2E-017</td>
<td>4E-016</td>
</tr>
<tr>
<td>1cvj</td>
<td>26</td>
<td>2E+001</td>
<td>-</td>
<td>3E-014</td>
<td>5E-017</td>
</tr>
<tr>
<td>1fje</td>
<td>18</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>1fje</td>
<td>17</td>
<td>-</td>
<td>-</td>
<td>9E-006</td>
<td>5E-011</td>
</tr>
<tr>
<td>1u2f</td>
<td>18</td>
<td>-</td>
<td>-</td>
<td>9E-004</td>
<td>1E-009</td>
</tr>
</tbody>
</table>
Table IV.7: Fold recognition results for an SH3 domain with PDB ID = 1SHF. Fold recognition results using Designed, Combined, Natural, and Structure MSAs are shown. The definitions of the four different types of MSAs are provided in the Methods section.

<table>
<thead>
<tr>
<th>PDB ID</th>
<th>% ID</th>
<th>Designed</th>
<th>Combined</th>
<th>Natural</th>
<th>Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>1awj</td>
<td>36</td>
<td>-</td>
<td>-</td>
<td>1E-009</td>
<td>4E-012</td>
</tr>
<tr>
<td>1shg</td>
<td>33</td>
<td>-</td>
<td>-</td>
<td>1E-012</td>
<td>6E-012</td>
</tr>
<tr>
<td>1shf</td>
<td>100</td>
<td>1.4</td>
<td>95</td>
<td>2E-022</td>
<td>4E-018</td>
</tr>
<tr>
<td>2src</td>
<td>75</td>
<td>2.4</td>
<td>-</td>
<td>2E-022</td>
<td>1E-018</td>
</tr>
<tr>
<td>1qcf</td>
<td>59</td>
<td>-</td>
<td>-</td>
<td>7E-020</td>
<td>3E-021</td>
</tr>
<tr>
<td>1lck</td>
<td>52</td>
<td>-</td>
<td>-</td>
<td>2E-018</td>
<td>5E-020</td>
</tr>
<tr>
<td>1qly</td>
<td>45</td>
<td>-</td>
<td>-</td>
<td>2E-015</td>
<td>2E-013</td>
</tr>
<tr>
<td>1bbz</td>
<td>45</td>
<td>-</td>
<td>-</td>
<td>7E-014</td>
<td>2E-012</td>
</tr>
<tr>
<td>1gri</td>
<td>34</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>4E-016</td>
</tr>
<tr>
<td>1gbr</td>
<td>32</td>
<td>-</td>
<td>-</td>
<td>2E-010</td>
<td>4E-017</td>
</tr>
<tr>
<td>1csk</td>
<td>33</td>
<td>55</td>
<td>-</td>
<td>1E-007</td>
<td>7E-014</td>
</tr>
<tr>
<td>1cka</td>
<td>34</td>
<td>-</td>
<td>-</td>
<td>2E-009</td>
<td>1E-012</td>
</tr>
<tr>
<td>1sem</td>
<td>29</td>
<td>-</td>
<td>-</td>
<td>5E-015</td>
<td>1E-014</td>
</tr>
<tr>
<td>1gri</td>
<td>32</td>
<td>-</td>
<td>-</td>
<td>4E-017</td>
<td>4E-016</td>
</tr>
<tr>
<td>1ycs</td>
<td>29</td>
<td>-</td>
<td>-</td>
<td>4E-009</td>
<td>2E-014</td>
</tr>
<tr>
<td>2hsp</td>
<td>36</td>
<td>-</td>
<td>-</td>
<td>2E-008</td>
<td>7E-015</td>
</tr>
<tr>
<td>1ark</td>
<td>32</td>
<td>-</td>
<td>-</td>
<td>1E-010</td>
<td>2E-014</td>
</tr>
<tr>
<td>1aoj</td>
<td>27</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>2E-013</td>
</tr>
<tr>
<td>1pht</td>
<td>31</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>3E-012</td>
</tr>
<tr>
<td>1bb9</td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>4E-011</td>
</tr>
</tbody>
</table>
Figure IV.2: Conservation profile for an $\alpha$-amylase protein domain with PDB ID = 1BAG.

Their correlation coefficient were below 0.1. This difference is more striking since for the two of structures (i.e. $\alpha$-amylase and RRM), native sequence recapitulation rates for their designed sequences are relatively high (30% at the level of whole protein). Furthermore, conservation profiles of designed sequences consistently indicated higher conservation scores than conservation profiles of the ntaurally occurring sequences. This observation is probably due to the fixed backbone model used, which restricts the sequence space sampled and thus artificially boost the conservation scores.

IV.D Discussion and Conclusion

The protein design algorithm used here has been shown to design sequences with rates of recapitulation of native sequences comparable to those reported earlier [62, 63]. In addition, it has been demonstrated that using the same set of designed se-
Figure IV.3: Conservation profile for an RNA Recognition Motif protein domain with PDB ID = 1FXL.
Figure IV.4: Conservation profile for an SH3 protein domain with PDB ID = 1SHF.
quences, the protein design algorithm with fixed backbone can detect close homologues. However, for those homologues near 30% sequence identity, Designed MSAs cannot detect them with significant confidence, whereas Natural MSAs and Structure MSAs can confidently detect most of them. Furthermore, Combined MSAs performed either worse than or similar to Designed MSAs for various fold recognition targets. These fold recognition results suggest that the current implementation of the protein design algorithm is not accurately identifying structurally important residues, whereas MSAs of naturally occurring sequences and those derived from aligned structures encode them.

The difference between Designed MSAs and Natural MSAs was further emphasized by their conservation profiles. Based on these profiles, Natural MSAs samples much larger sequence space than Designed MSAs. Conservation profiles of Designed MSAs show higher overall scores because the designed sequences are very close to each other (average pair-wise %ID of around 60). Despite these differences, Designed MSAs can still detect a few remote homologues, suggesting that more work needs to be done to identify what properties of MSAs are responsible for their detections.

Based the results presented above, recapitulation rates of native sequences alone should not be used to determine how well a protein design algorithm is modeling the properties of the Natural MSAs. Benefits of using known structural homologues in a protein design algorithm has not been entirely realized in this study. On a more positive note, known structural homologues appear to be a source of rigorous benchmarks for protein design algorithms.
V

Functional modules detected by searching for frequently occurring patterns in protein interaction networks of different organisms

V.A Introduction

A computational method, MULE, that can mine large-scale protein-protein interaction networks for frequently occurring subnetworks was developed by Mehmet Koyutürk and others\(^1\). Through such mining, it became possible to answer which sub-networks of interactions are conserved among different organisms. As in sequence alignments, highly conserved sub-networks may indicate functional importance. Consequently, the author took an active role in analyzing the frequently occurring protein interaction patterns returned by the method. The author then carried out extensive literature searches on the conserved sub-networks and validated these findings with functional modules and pathways that are described in the literature.

\(^1\)This chapter describes the contributions that the author of the dissertation had made in the article "Detecting conserved interaction patterns in biological networks" by M. Koyutürk, Y. Kim, S. Subramaniam, W. Szpankowski, and Grama A. J. Comput. Biol. 13, 1299-1322 (2006).
V.B Methods

The method MULE is fully described in [76]. To present the examples of detected functional modules in a proper context, an outline of the method is provided.

A simple execution of the method MULE is illustrated in Figure V.1. Given a set of four protein interaction networks, represented as graphs, shown (i.e. $G_1$, $G_2$, $G_3$, and $G_4$) in Figure V.1a, the method finds all interaction patterns that occur in at least three interaction networks. As a first step, those edges with at least three occurrences in the networks are determined. In the example shown in the figure, these edges are $ab$, $ac$, $de$, and $ea$. During the second step, all possible interaction patterns from combinations of this set of edges are explored efficiently, by traversing the tree shown in Figure V.1b. While traversing the tree, if an interaction pattern cannot be expanded further, the pattern is returned.

V.C Results

The method was applied to the nine eukaryotic protein-protein interaction networks from BIND [77] and DIP [78], shown in Table V.1.

In Figure V.2, one of the conserved interaction networks detected with a frequency threshold of four organisms is shown. The conserved network shown in Figure V.2 constitutes a portion of the basal transcription factor TFIID. In $S. cerevisiae$, TFIID consists of one TATA-Binding Protein (TBP) and at least 14 TATA-Associated Factors (TAFs) [79]; yet in the conserved subnetwork, only four are found. One hypothesis explaining this observation is that the TAFs present in the conserved network have greater role in promoting transcription than those TAFs that are absent.

Figures V.3 and V.4 show two of the conserved networks detected with a frequency threshold of three organisms. Fortunately, almost all proteins involved in these conserved subnetworks are well annotated for $S. cerevisiae$, allowing validation of the detected subnetwork. The subnetwork shown in Figure V.3 is a small nuclear ribonucleoprotein complex, and it is conserved in $D. melanogaster$, $C. elegans$, and $S. cerevisiae$. 
Proteins Lsm1-7 make up a complex that participates in mRNA degradation and splicing [80]. Proteins Smx3 and Smd2 are homologues of subunits in this complex.

In Figure V.4, the conserved subnetwork shown contains components of Actin-related protein Arp2/3 complex, involved in actin nucleation. Of the seven components known for Arp2/3 complex in S. cerevisiae, Arc18 is missing in the conserved subnetwork [81]. In the same study, it was shown that Arc40 is essential for viability, which may explain why Arc40 has greater number of interacting partners than the other proteins present in the conserved network.

V.D Conclusion

MULE is able to detect known functional modules from the eukaryotic interaction networks by exploiting their conservation among different organisms. Although our results are limited by the availability of the interaction data, it appears that the conservation of functional modules is a widespread phenomenon observed in numerous cellular activities. Interactions among subunits of protein complexes involved in transcription, mRNA degradation and splicing, and actin nucleation are significantly conserved in yeast and higher eukaryotes, such as humans. This suggests that as more interaction data becomes available, MULE can be used to automatically map functional organization of proteins of a query organism based on the interaction networks of others.

Acknowledgments

Section V uses some of the figures and table from the article "Detecting conserved interaction patterns in biological networks" by M. Koyutürk, Y. Kim, S. Subramaniam, W. Szpankowski, and A. Grama, published in J. Comput. Biol., volume 13, pages 1299-1322, 2006. Y. Kim contributed to the analysis of the frequently occurring interaction patterns detected by the method. This work has been reprinted with permissions from co-authors and Mary Ann Liebert Inc., publishers.
Figure V.1: Sample execution of MULE with a goal of finding patterns that are in at least 3 interaction networks. (a) Four protein interaction networks represented as graphs. (b) Efficiently expanding the set of edges to identify frequently occurring patterns.
Table V.1: Statistics of Mined PPI Networks and the Corresponding Ortholog-Contracted Graphs

<table>
<thead>
<tr>
<th>Organism</th>
<th>PPI network</th>
<th>Ortholog-contracted graph</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>No. of proteins</td>
<td>No. of interactions</td>
</tr>
<tr>
<td>A. thaliana</td>
<td>288</td>
<td>424</td>
</tr>
<tr>
<td>O. sativa</td>
<td>301</td>
<td>340</td>
</tr>
<tr>
<td>S. cerevisiae</td>
<td>5157</td>
<td>18,192</td>
</tr>
<tr>
<td>C. elegans</td>
<td>3345</td>
<td>5988</td>
</tr>
<tr>
<td>D. melanogaster</td>
<td>8577</td>
<td>28,829</td>
</tr>
<tr>
<td>H. sapiens</td>
<td>4541</td>
<td>8577</td>
</tr>
<tr>
<td>B. taurus</td>
<td>195</td>
<td>265</td>
</tr>
<tr>
<td>M. musculus</td>
<td>2479</td>
<td>2959</td>
</tr>
<tr>
<td>R. norvegicus</td>
<td>696</td>
<td>881</td>
</tr>
</tbody>
</table>

Figure V.2: Frequently occurring interaction pattern corresponds to the basal transcription factor TFIID ($p < 9e^{-51}$). Orthologous proteins are horizontally aligned.
Figure V.3: Frequently occurring interaction pattern corresponds to small nuclear ribonucleoprotein complex \((p < 2e-43)\). Orthologous proteins are horizontally aligned.

Figure V.4: Frequently occurring interaction pattern corresponds to actin-related protein Arp2/3 complex \((p < 9e-11)\). Orthologous proteins are horizontally aligned.
Bibliography


