21

COMPUTATIONAL APPROACHES TO PREDICT PROTEIN–PROTEIN AND DOMAIN–DOMAIN INTERACTIONS

RAJA JOTHI AND TERESA M. PRZYTYCKA

National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health, Bethesda, MD, USA

21.1 INTRODUCTION

Knowledge of protein and domain interactions provides crucial insights into their functions within a cell. Various high throughput experimental techniques such as mass spectrometry, yeast two hybrid, and tandem affinity purification have **[Q3]** generated a significant amount of large-scale high throughput protein interaction data [9,19,21,28,29,35,36,58]. Advances in experimental techniques are paralleled by the rapid development of computational approaches designed to detect protein–protein interactions [11,15,24,37,45,46,48,50]. These approaches complement experimental techniques and, if proven to be successful in predicting interactions, provide insights into principles governing protein interactions.

A variety of biological information (such as amino acid sequences, coding DNA sequences, three-dimensional structures, gene expression, codon usage, etc.) is used by computational methods to arrive at interaction predictions. Most methods rely on statistically significant biological properties observed among interacting proteins/domains. Some of the widely used properties include co-occurence, coevolution, co-expression, and co-localization of interacting proteins/domains.

Bioinformatics Algorithms: Techniques and Applications, Edited by Ion I. Măndoiu and Alexander Zelikovsky

This chapter is, by no account, a complete survey of all available computational approaches for predicting protein and domain interactions but rather a presentation of a bird's-eye view of the landscape of a large spectrum of available methods. For detailed descriptions, performances, and technical aspects of the methods, we refer the reader to the respective articles.

21.2 PROTEIN–PROTEIN INTERACTIONS

21.2.1 Phylogenetic Profiles

The patterns of presence or absence of proteins across multiple genomes (phylogenetic or phyletic profiles) can be used to infer interactions between proteins [18,50]. A phylogenetic profile for each protein i is a vector of length n that contains the presence or absence information of that protein in a reference set of n organisms. The presence or absence of protein i in organism j is recorded as $P_{ij} = 1$ or $P_{ij} = 0$, respectively, which is usually determined by performing a BLAST search [4] with an E-value threshold t. If the BLAST search results in a hit with E-value $< t$, then it is construed as an evidence for the presence of protein p in G. Otherwise, it is assumed that p is absent in G.

Proteins with identical or similar profiles are inferred to be functionally interacting under the assumption that proteins involved in the same pathway or functional system are likely to have been co-inherited during evolution [18,50] (Fig. 21.1a). Similarities between profiles can be measured using matrices such as Hamming distance, Jaccard coefficient, mutual information, among others. It has been shown that measuring profile similarity using mutual information rather than matrices such as Hamming distance results in a better prediction accuracy [22]. By clustering proteins based on their profile similarity scores, one can construct functional pathways and interaction network modules [12,22]. One of the main limitations of the profile comparison approach is the lineage-specific gains and losses of genes, thought to be more pervasive in microbial evolution [39], which could artificially decrease the similarity between functionally interacting genes.

Instead of using an ad hoc E-value threshold and binary values as originally proposed [50], recent studies have been using $P_{ij} = -1/\log E_{ij}$ to record the presence/absence information, where E_{ij} is the BLAST E-value of the top-scoring sequence alignment of protein i in organism j. To avoid algorithm-induced artifacts, $P_{ij} > 1$ are truncated to 1. Notice that a zero (or a one) entry in the profile now indicates the presence (absence, respectively) of a protein. It is being argued that using real values for P_{ij}, instead of binary values, captures varying degrees of sequence divergence, providing more information than the simple presence or absence of genes [12,33,37]. For a more comprehensive assessment of the phylogenetic profile comparison approach, we refer the reader to [33].

21.2.2 Gene Fusion Events

There are instances where a pair of interacting proteins in one genome is fused together into a single protein (referred to as the Rosetta Stone protein [37]) in another genome.

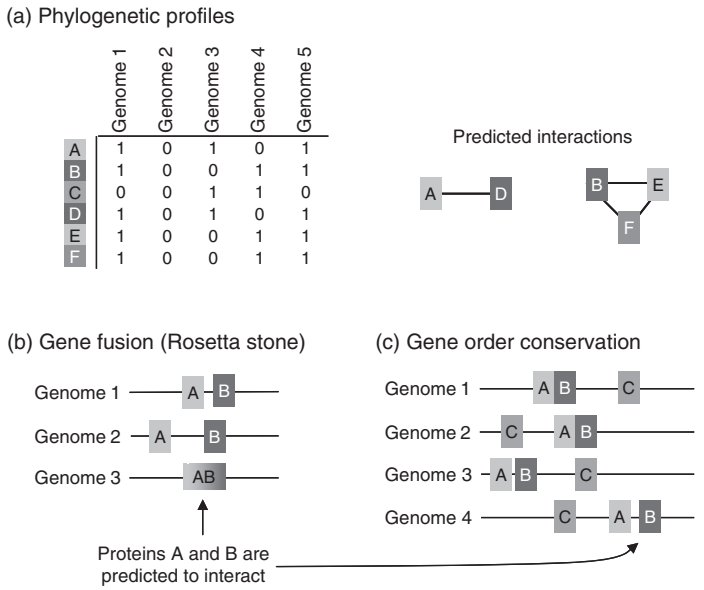


FIGURE 21.1 Computational approaches for predicting protein–protein interactions from genomic information. (**a**) Phylogenetic profiles [18,50]. A profile for a protein is a vector of 1s and 0s recording presence or absence, respectively, of that protein in a set of genomes. Two proteins are predicted to interact if their phylogenetic profiles are identical (or similar). (**b**) Gene fusion (Rosetta stone) [15,37]. Proteins *A* and *B* in a genome are predicted to interact if they are fused together into a single protein (Rosetta protein) in another genome. (**c**) Gene order conservation [11,45]. If the genes encoding proteins *A* and *B* occupy close chromosomal positions in various genomes, then they are inferred to interact. Figure adapted from [59].

For example, interacting proteins Gyr A and Gyr B in *Escherichia coli* are fused together into a single protein (topoisomerase II) in *Saccharomyces cerevisiae* [7]. Amino acid sequences of Gyr A and Gyr B align to different segments of the topoisomerase II. On the basis of such observations, methods have been developed [15,37] to predict interaction between two proteins in an organism based on the evidence that they form a part of a single protein in other organisms. A schematic illustration of this approach is shown in Fig. 21.1b.

21.2.3 Gene Order Conservation

The interactions between proteins can be predicted based on the observation that proteins encoded by conserved neighboring gene pairs interact (Fig. 21.1c). This idea is based on the notion that physical interaction between encoded proteins could be one of the reasons for evolutionary conservation of gene order [11]. Gene order conservation between proteins in bacterial genomes has been used to predict functional interactions [11,45]. This approach's applicability to bacterial genomes only, in which the genome order is a relevant property, is one of its main limitations [59]. Even within the bacteria, caution must be exercised while interpreting conservation of gene order

between evolutionarily closely related organisms (for example, *Mycoplasma genitalium* and *Mycoplasma pneumoniae*), as lack of time for genome rearrangements after divergence of the two organisms from their last common ancestor could be a reason for the observed gene order conservation. Hence, only organisms with relatively long evolutionary distances should be considered for such type of analysis. However, the evolutionary distances should be small enough in order to ensure that a significant number of orthologous genes are still shared by the organisms [11].

21.2.4 Similarity of Phylogenetic Trees

It is postulated that the sequence changes accumulated during the evolution of one of the interacting proteins must be compensated by changes in its interaction partner. Such correlated mutations have been subject of several studies [3,23,41,55]. Pazos et al. [46] demonstrated that the information about correlated sequence changes can distinguish right interdocking sites from incorrect alternatives. In recent years, a new method has emerged, which, rather than looking at coevolution of individual residues in protein sequences, measures the degree of coevolution of entire protein sequences by assessing the similarity between the corresponding phylogenetic trees [24,25,31,32,34,46–48,51,54]. Under the assumption that interacting protein sequences and their partners must coevolve (so that any divergent changes in one partner's binding surface are complemented at the interface by their interaction partner) [6,30,40,46], pairs of protein sequences exhibiting high degree of coevolution are inferred to be interacting.

In this section, we first describe the basic "mirror-tree" approach for predicting interaction between proteins by measuring the degree of coevolution between the corresponding amino acid sequences. Next, we describe an important modification to the basic mirror-tree approach that helps in improving its prediction accuracy. Finally, we discuss a related problem of predicting interaction specificity between two families of proteins (say, ligands and receptors) that are known to interact.

21.2.4.1 The Basic Mirror-Tree Approach This approach is based on the assumption that phylogenetic trees of interacting proteins are highly likely to be similar due to the inherent need for coordinated evolution [24,49]. The degree of similarity between two phylogenetic trees is measured by computing the correlation between the corresponding distance matrices that implicitly contains the evolutionary histories of the two proteins.

A schematic illustration of the mirror-tree method is shown in Fig. 21.2. The multiple sequence alignments (MSA) of the two proteins, for a common set of species, are constructed using one of the many available MSA algorithms such as ClustalW [57], MUSCLE [14], and T-Coffee [43]. The set of orthologous proteins for a MSA is usually obtained by one of the two following ways: (i) a stringent BLAST search with a certain *E*-value threshold, sequence identity threshold, alignment overlap percentage threshold or a combination thereof, or (ii) reciprocal (bidirectional) BLAST best-hits. In both approaches, orthologous sequences of a query protein q in organism Q is searched by performing a BLAST search of q against sequences in other organisms.

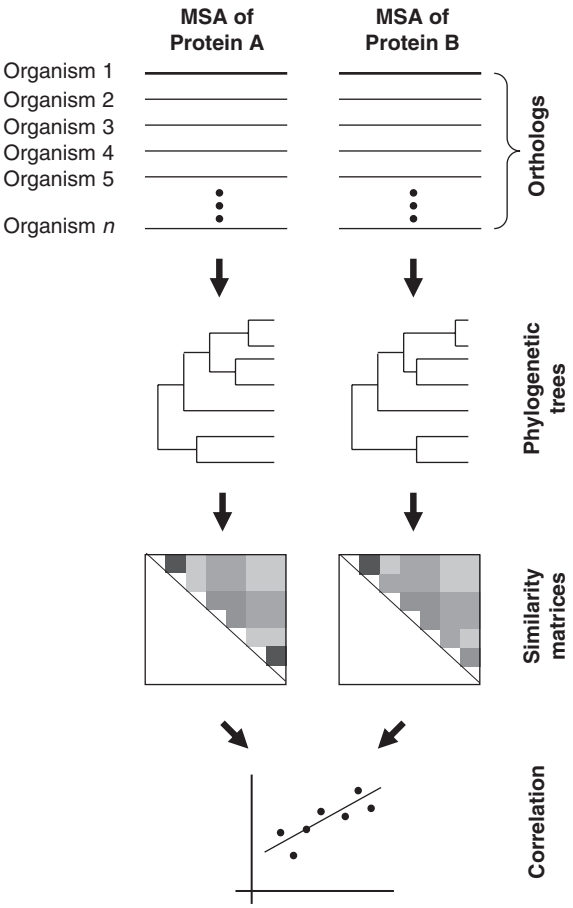


FIGURE 21.2 Schema of the mirror-tree method. Multiple sequence alignments of proteins *A* and *B*, constructed from orthologs of *A* and *B*, respectively, from a common set of species, are used to generate the corresponding phylogenetic trees and distance matrices. The degree of coevolution between *A* and *B* is assessed by comparing the corresponding distance matrices using a linear correlation criteria. Proteins *A* and *B* are predicted to interact if the degree of coevolution, measured by the correlation score, is high (or above a certain threshold).

In the former, q's best-hit h in organism H, with E-value $< t$, is considered to be orthologous to Q. In the latter, q's best-hit h in organism H (with no specific E-value threshold) is considered to be orthologous to q if and only if h's best-hit in organism Q is q. Using reciprocal best-hits approach to search for orthologous sequences is considered to be much more stringent than just using unidirectional BLAST searches with an E-value threshold t.

In order to be able to compare the evolutionary histories to two proteins, it is required that the two proteins have orthologs in at least a common set of n organisms. It is advised that n be large enough for the trees and that the corresponding

distance matrices contain sufficient evolutionary informatin. It is suggested that $n \geq 10$ [31,47,48]. Phylogenetic trees from MSA are constructed using standard tree construction algorithms (such as neighbor joining [53]), which are then used to construct the distance matrices (algorithms to construct trees and matrices from MSAs are available in the ClustalW suite).

The extent of agreement between the evolutionary histories of two proteins is assessed by computing the degree of similarity between the two corresponding distance matrices. The extent of agreement between matrices A and B can be measured using Pearson's correlation coefficient, given by

$$r_{AB} = \frac{\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} (A_{ij} - \overline{A})(B_{ij} - \overline{B})}{\sqrt{\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} (A_{ij} - \overline{A})^2 \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} (B_{ij} - \overline{B})^2}}, \tag{21.1}$$

where n is the number of organisms (number of rows or columns) in the matrices, A_{ij} and B_{ij} are the evolutionary distances between organisms i and j in the tree of proteins A and B, respectively, and $\overline{A}$ and $\overline{B}$ are the mean values of all A_{ij} and B_{ij}, respectively. The value of r_{AB} ranges from -1 to +1. The higher the value of r, the higher the agreement between the two matrices and thus the higher the degree of coevolution between A and B.

Pairs of proteins with correlation scores above a certain threshold are predicted to interact. A correlation score of 0.8 is considered to be a good threshold for predicting protein interactions [24,49]. Pazos et al. [49] estimated that about one third of the predictions by the mirror-tree method are false positives. A false positive in this context refers to a noninteracting pair that was predicted to interact due to their high correlation score. It is quite possible that the evolutionary histories of two noninteracting proteins are highly correlated due to their common speciation history. Thus, in order to truly assess the correlation of evolutionary histories of two proteins, one should first subtract the background correlation due to their common speciation history. Recently, it has been observed that subtracting the underlying speciation component greatly improves the predictive power of the mirror-tree approach by reducing the number of false positives. Refined mirror-tree methods that subtract the underlying speciation signal are discussed in the following subsection.

21.2.4.2 Accounting for Background Speciation As pointed at the end of the previous section, to improve the performance of the mirror-tree approach, the coevolution due to common speciation events should be subtracted from the overall coevolution signal. Recently, two approaches, very similar in techniques, have been proposed to address this problem [47,54].

For an easier understanding of the speciation subtraction process, let us think of the distance matrices used in the mirror-tree method as vectors (i.e., the upper right triangle of the distance matrices is linearized and represented as a vector), which will be referred to as the *evolutionary vectors* hereafter. Let $\overrightarrow{V_A}$ and $\overrightarrow{V_B}$ denote the evolutionary vector computed for a multiple sequence alignment of orthologs of proteins A and B, respectively, for a common set of species. Let $\overrightarrow{S}$ denote the canonical

evolutionary vector, also referred to as the *speciation vector*, computed in the same way but based on a multiple sequence alignment of 16S rRNA sequences for the same set of species. Speciation vector $\overrightarrow{S}$ approximates the interspecies evolutionary distance based on the set of species under consideration. The differences in the scale of protein and RNA distance matrices are overcome by rescaling the speciation vector values by a factor computed based on "molecular clock" proteins [47].

A pictorial illustration of the background speciation subtraction procedure is shown in Fig. 21.3. The main idea is to decompose evolutionary vectors $\overrightarrow{V_A}$ and $\overrightarrow{V_B}$ into two components: one representing the contribution due to speciation, and the other representing the contribution due to evolutionary pressure related to preserving the protein function (denoted by $\overrightarrow{C_A}$ and $\overrightarrow{C_B}$, respectively). To obtain $\overrightarrow{C_A}$ and $\overrightarrow{C_B}$, the speciation component $\overrightarrow{S}$ is subtracted from $\overrightarrow{V_A}$ and $\overrightarrow{V_B}$, respectively. Vectors $\overrightarrow{C_A}$ and $\overrightarrow{C_B}$ are expected to contain only the distances between orthologs that are not due to speciation but to other reasons related to function [47]. The degree of coevolution between A and B is then measured by computing the correlation between $\overrightarrow{C_A}$ and $\overrightarrow{C_B}$, rather than between $\overrightarrow{V_A}$ and $\overrightarrow{V_B}$ as in the basic mirror-tree approach.

The two speciation subtraction methods, by Pazos et al. [47] and Sato et al. [54], differ in how speciation subtraction is performed (see Fig. 21.3). An in-depth analysis of the pros and cons of two methods is provided in [34]. In a nutshell, Sato et al. attribute all changes in the direction of the speciation vector to the speciation process and thus assume that vector $\overrightarrow{C_A}$ is perpendicular to the speciation vector $\overrightarrow{S}$, whereas Pazos et al. assume that the speciation component in $\overrightarrow{V_A}$ is constant and independent on the protein family. As a result, Pazos et al. compute $\overrightarrow{C_A}$ to be the difference between $\overrightarrow{V_A}$ and $\overrightarrow{S}$, which explains the need to rescale RNA distances to protein distances in the vector $\overrightarrow{S}$. Interestingly, despite this difference, both speciation correction methods produce similar result [34]. In particular, Pazos et al. report that the speciation subtraction step reduces the number of false positives by about 8.5%.

The above-mentioned methods for subtracting the background speciation have recently been complemented by the work of Kann et al. [34]. Under the assumption that in conserved regions of the sequence alignment functional coevolution may be less concealed by speciation divergence, they demonstrated that the performance of the mirror-tree method can be improved further by restricting the coevolution analysis to the relatively highly conserved regions in the protein sequence [34].

21.2.4.3 Predicting Protein Interaction Specificity In this section, we address the problem of predicting interaction partners between members of two proteins families that are known to interact [20,32,51]. Given two families of proteins that are known to interact, the objective is to establish a mapping between the members of one family with the members of the other family.

To better understand the protein interaction specificity (PRINS) problem, let us consider an analogous problem, which we shall refer to as the *matching* problem. Imagine a social gathering attended by n married couples. Let $H = \{h_1, h_2, \ldots, h_n\}$ and $W = \{w_1, w_2, \ldots, w_n\}$ be the sets of husbands and wives attending the gathering. Given that husbands in set H are married to the wives in set W and that the marital

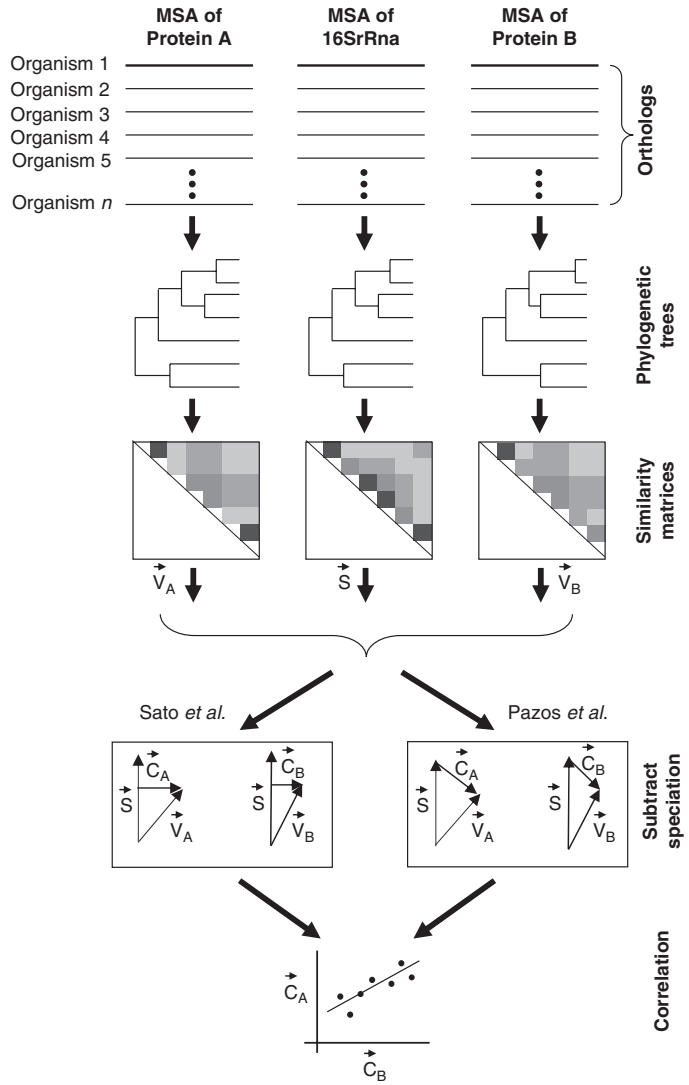


FIGURE 21.3 Schema of the mirror-tree method with a correction for the background speciation. Correlation between the evolutionary histories of two proteins could be due to (i) a need to coevolve in order to preserve the interaction and/or (ii) common speciation events. To estimate the coevolution due to the common speciation, a canonical tree-of-life is constructed by aligning the 16 S rRNA sequences. The rRNA alignment is used to compute the distance matrix representing the species tree. $\overrightarrow{V_A}$, $\overrightarrow{V_B}$, and $\overrightarrow{S}$ are the vector notations for the corresponding distance matrices. Vector $\overrightarrow{C_X}$ is obtained from $\overrightarrow{V_X}$ by subtracting it by the speciation component $\overrightarrow{S}$. The speciation component $\overrightarrow{S}$ is calculated differently based on the method being used. The degree of coevolution between A and B is then assessed by computing the linear correlation between $\overrightarrow{C_A}$ with $\overrightarrow{C_B}$. Proteins A and B are predicted to interact if the correlation between $\overrightarrow{C_A}$ and $\overrightarrow{C_B}$ is sufficiently high.

relationship is monogamous, the matching problem asks for a one-to-one mapping of the members in H to those in W such that each mapping (h_i, w_j) holds the meaning "h_i is married to w_j." In other words, the objective is to pair husbands and wives such that all n pairings are correct. The matching problem has a total of $n!$ possible mappings out of which only one is correct. The matching problem becomes much more complex if one were to remove the constraint that requires that the marital relationship is monogamous. Such a relaxation would allow the sizes of sets H and W to be different. Without knowing the number of wives (or husbands) each husband (wife, respectively) has, the problem becomes intractable.

The PRINS problem is essentially the same as the matching problem with the two sets containing proteins instead of husbands and wives. Let A and B be the two sets of proteins. Given that the proteins in A interact with those in B, the objective is to map proteins in A to their interaction partners in B. To fully appreciate the complexity of this problem, let us first consider a simpler version of the problem that assumes that the number of proteins in A is the same as that in B and the interaction between the members of A and B is one to one.

Protein interaction specificity (a protein binding to a specific partner) is vital to cell function. To maintain the interaction specificity, it is required that it persists through the course of strong evolutionary events, such as gene duplication and gene divergence. As genes are duplicated, the binding specificities of duplicated genes (paralogs) often diverge, resulting in new binding specificities. Existence of numerous paralogs for both interaction partners can make the problem of predicting interaction specificity difficult as the number of potential interactions grow combinatorially [51].

Discovering interaction specificity between the two interacting families of proteins, such as matching ligands to specific receptors, is an important problem in molecular biology, which remains largely unsolved. A naive approach to solve this problem would be to try out all possible mappings (assuming that there is an oracle to verify whether a given mapping is correct). If A and B contain n proteins each, then there are a total of $n!$ possible mappings between matrices A and B. For a fairly large n, it is computationally unrealistic to try out all possible mappings.

Under the assumption that interacting proteins undergo coevolution, Ramani and Marcotte [51] and Gertz et al. [20], in independent and parallel works, proposed the "column-swapping" method for the PRINS problem. A schematic illustration of the column-swapping approach is shown in Fig. 21.4. Matrices A and B in Fig. 21.4 correspond to distance matrices of families A and B, respectively. In this approach, a Monte Carlo algorithm [38] with simulated annealing is used to navigate through the search space in an effort to maximize the correlation between the two matrices. The Monte Carlo search process, instead of searching through the entire landscape of all possible mappings, allows for a random sampling of the search space in a hope to find the optimal mapping. Each iteration of the Monte Carlo search process, referred to as a "move," constitutes the following two steps.

1. Choose two columns uniformly at random and swap their positions (the corresponding rows are also swapped).

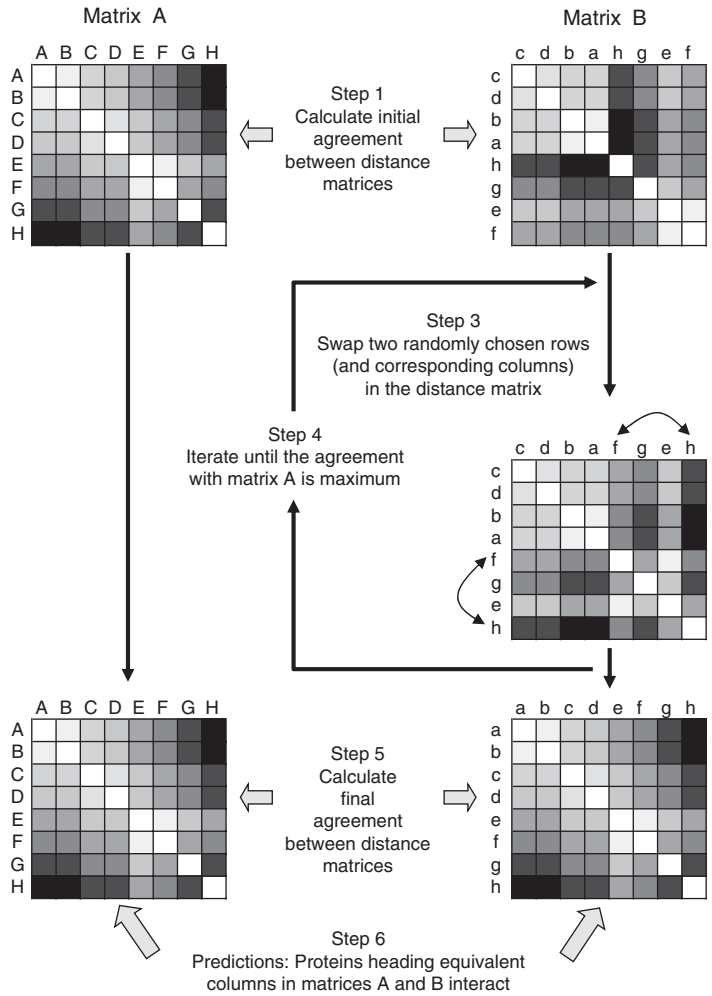


FIGURE 21.4 Schema of the column-swapping algorithm. Image reproduced from [51] with permission.

2. If, after the swap, the correlation between the two matrices has improved, the swap is kept. Else, the swap is kept with the probability $p = \exp(-\delta/T)$, where δ is the decrease in the correlation due to the swap, and T is the temperature control variable governing the simulation process.

Initially, T is set to a value such that $p = 0.8$ to begin with, and after each iteration the value of T is decreased by 5%. After the search process converges to a particular mapping, proteins heading equivalent columns in the two matrices are predicted to interact. As with any local search algorithm, it is difficult to say whether the final mapping is an optimal mapping or a local optima.

The main downside of the column-swapping algorithm is the size of search space ($n!$), which it has to navigate in order to find the optimal mapping. Since the size of the search space is directly proportional to search (computational) time, column-swapping algorithm becomes impractical even for families of size 30.

In 2005, Jothi et al. [32] introduced a new algorithm, called MORPH, to solve the PRINS problem. The main motivation behind MORPH is to reduce the search space of the column-swapping algorithm. In addition to using the evolutionary distance information, MORPH uses topological information encoded in the evolutionary trees of the protein families. A schematic illustration of the MORPH algorithm is shown in Fig. 21.5. While MORPH is similar to the column-swapping algorithm at the top level, the major (and important) difference is the use of phylogenetic tree topology to guide the search process. Each move in the column-swapping algorithm involves swapping two random columns (and the corresponding rows), whereas each move in MORPH involves swapping two isomorphic[1] subtrees rooted at a common node (and the corresponding sets of rows and columns in the distance matrix).

Under the assumption that the phylogenetic trees of protein families A and B are topologically identical, MORPH essentially performs a topology-preserving embedding(superimposition) of one tree onto the other. The complexity of the topology of the trees plays a key role in the number of possible ways that one could superimpose one tree onto another. Figure 21.6 shows three sets of trees, each of which has different number of possible mappings based on the tree complexity. For the set of trees in Fig. 21.6a, the search space (number of mappings) for the column-swapping algorithm is $4! = 24$, whereas it is only eight for MOPRH.

To apply MORPH, the phylogenetic trees corresponding to the two families of proteins must be isomorphic. To ensure that the trees are isomorphic, MORPH starts by contracting/shrinking these internal tree edges in both trees, with bootstrap score less than a certain threshold. It is made sure that equal number of edges are contracted on both trees. If, after the initial edge contraction procedure, the two trees are not isomorophic, additional internal edges are contracted on both trees (in increasing order of the edge bootstrap scores) until the trees are isomorphic. The benefits of edge contraction procedure is twofold: (i) ensure that the two trees are isomorphic to begin with and (ii) decrease the chances of less reliable edges (with low bootstrap scores) wrongly influencing the algorithm. Since MORPH relies heavily on the topology of the trees, it is essential that the tree edges are trustworthy. In the worst case, contracting all the internal edges on both trees will leave two star-topology trees (like those in Fig. 21.6c), in which case the number of possible mappings considered by MORPH will be the same as that considered by the column-swapping algorithm. Thus, in the worst case, MORPH's search space will be as big as that of the column-swapping algorithm.

After the edge contraction procedure, a Monte Carlo search process similar to that used in the column-swapping algorithm is used to find the best possible

[1]Two trees T_1 and T_2 are isomorphic if there is a one-to-one mapping between their vertices (nodes) such that there is an edge between two vertices in T_1 if and only if there is an edge between the two corresponding vertices in T_2.

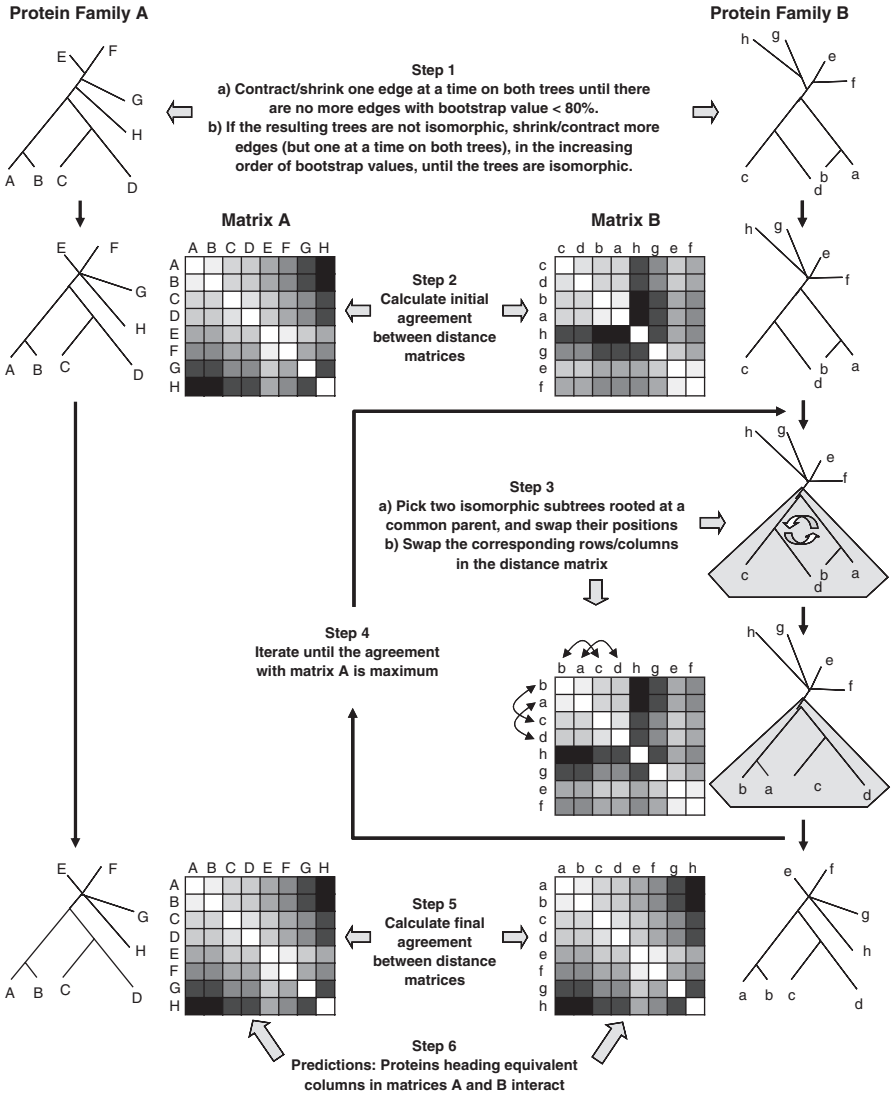


FIGURE 21.5 Schema of the MORPH algorithm. Image reprinted from [32] with permission.

superimposition of the two trees. As in the column-swapping algorithm, the distance matrix and the tree corresponding to one of the two families are fixed, and transformations are made to the tree and the matrix corresponding to the second family. Each iteration of the Monte Carlo search process constitutes the following two steps:

1. Choose two isomorphic subtrees, rooted at a common node, uniformly at random and swap their positions (and the corresponding sets of rows/columns)

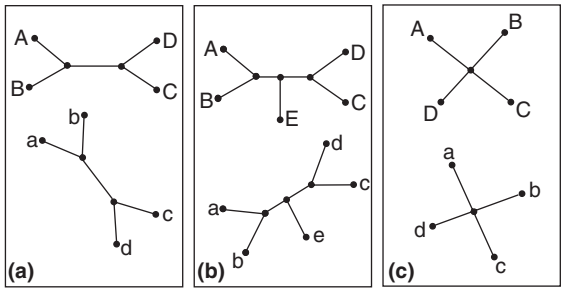


FIGURE 21.6 Three sets of topologically identical (isomorphic) trees. The number of topology preserving mappings of one tree onto another is (**a**) 8, (**b**) 8, and (**c**) 24. Despite the same number of leaves in (**a**) and (**c**), the number of possible mappings is different. This difference is due to the increased complexity of the tree topology in (**a**) when compared to that in (**c**). Image reprinted from [32] with permission.

2. If, after the swap, the correlation between the two matrices has improved, the swap is kept. Else, the swap is kept with the probability $p = \exp(-\delta/T)$.

Parameters δ and T are the same as those in the column-swapping algorithm. After the search process converges to a certain mapping, proteins heading equivalent columns in the two matrices are predicted to interact.

The sophisticated search process used in MORPH reduces the search space by multiple orders of magnitude in comparison to the column-swapping algorithm. As a result, MORPH can help solve larger instances of the PRINS problem. For more details on the column-swapping algorithm and MORPH, we refer the reader to [20,51] and [32], respectively.

21.3 DOMAIN–DOMAIN INTERACTIONS

Recent advances in molecular biology combined with large-scale high throughput experiments have generated huge volumes of protein interaction data. The knowledge gained from protein interaction networks has definitely helped to gain a better understanding of protein functionalities and inner workings of the cell. However, protein interaction networks by themselves do not provide insights into interaction specificity at the domain level. Most of the proteins are composed of multiple domains. It has been estimated that about two thirds of proteins in prokaryotes and about four fifths of proteins in eukaryotes are multidomain proteins [5,10]. Most often, the interaction between two proteins involves binding of a pair(s) of domains. Thus, understanding the interaction at the domain level is a critical step toward a thorough understanding of the protein–protein interaction networks and their evolution. In this section, we will discuss computational approaches for predicting protein domain interactions. We restrict our discussion to sequence- and network-based approaches.

21.3.1 Relative Coevolution of Domain Pairs Approach

Given a protein–protein interaction, predicting the domain pair(s) that is most likely mediating the interaction is of great interest. Formally, let protein P contain domains $\{P_1, P_2, \ldots, P_m\}$ and protein Q contain domains $\{Q_1, Q_2, \ldots, Q_n\}$. Given that P and Q interact, the objective is to find the domain pair $P_i Q_j$ that is most likely to mediate the interaction between P and Q. Recall that under the coevolution hypothesis, interacting proteins exhibit higher level of coevolution. On the basis of this hypothesis, it is only natural and logical to assume that interacting domain pairs for a given protein–protein interaction exhibit higher degree of coevolution than the noninteracting domain pairs. Jothi et al. [31] showed that this is indeed the case and, based on this, proposed the *relative coevolution of domain pairs* (RCDP) method to predict domain pair(s) that is most likely mediating a given protein–protein interaction.

Predicting domain interactions using RCDP involves two major steps: (i) make domain assignment to proteins and (ii) use mirror-tree approach to assess the degree of coevolution of all possible domain pairs. A schematic illustration of the RCDP method is shown in Fig. 21.7. Interacting proteins P and Q are first assigned with domains (HMM profiles) using HMMer [1], RPS-BLAST [2], or other similar tools. Next, MSAs for the two proteins are constructed using orthologous proteins from a common set of organisms (as described in Section 21.2.4.1). The MSA of domain P_i in protein P is constructed by extracting those regions in P's alignment that correspond

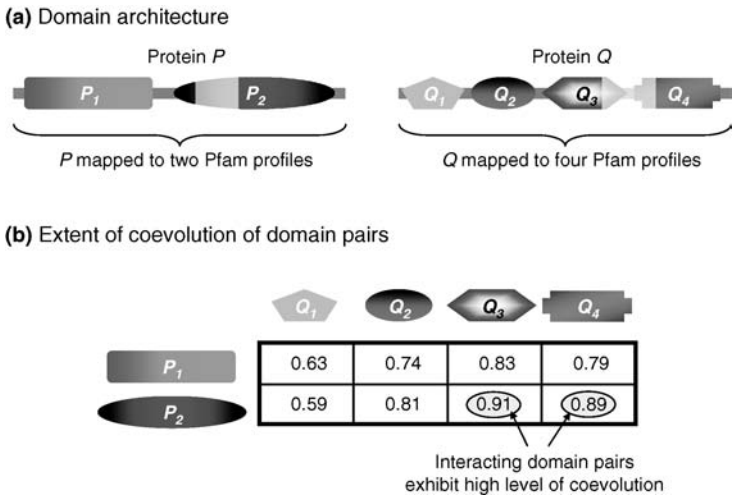
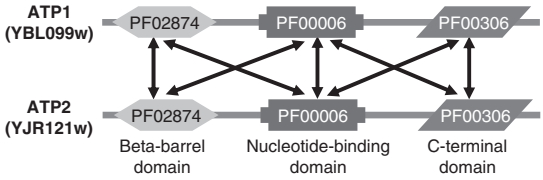


FIGURE 21.7 Relative coevolution of domain pairs in interacting proteins. (**a**) Domain assignments for interacting proteins P and Q. Interaction sites in P and Q are indicated by thick light-colored bands. (**b**) Correlation scores for all possible domain pairs between interacting proteins P and Q are computed using the mirror-tree method. The domain pair with the highest correlation score is predicted to be the one that is most likely to mediate the interaction between proteins P and Q. Figure adapted from [31].



YBL099w	YJR121w	Correlation	iPfam
PF00006	PF00006	0.95957039	Y
PF02874	PF00006	0.92390131	Y
PF00306	PF00306	0.89734590	Y
PF00006	PF02874	0.89692159	Y
PF02874	PF02874	0.88768393	Y
PF00006	PF00306	0.87369242	Y
PF00306	PF00006	0.86507957	Y
PF02874	PF00306	0.85735773	
PF00306	PF02874	0.84890155	

FIGURE 21.8 Protein–protein interaction between alpha (ATP1) and beta (ATP2) chains of F1-ATPase in *Saccharomyces cerevisiae*. Protein sequences YBL099w and YJR121w (encoded by genes ATP1 and ATP2, respectively) are annotated with three Pfam [17] domains each: beta-barrel domain (PF02874), nucleotide-binding domain (PF00006), and C-terminal domain (PF00306). The correlation scores of all possible domain pairs between the two proteins are listed (table on the right) in decreasing order. Interchain domain–domain interactions that are known to be true from PDB [8] crystal structures (as inferred in iPfam [16]) are shown using double arrows in the diagram and "Y" in the table. Interacting domain pairs between the two proteins have higher correlation than the noninteracting domain pairs. RCDP will correctly predict the top-scoring domain pair to be interacting. Figure adapted from [31].

to domain P_i. Then, using the mirror-tree method, the correlation (similarity) scores of all possible domain pairs between the two proteins are computed. Finally, the domain pair $P_i Q_j$ with the highest correlation score (or domain pairs, in case of a tie for the highest correlation score), exhibiting the highest degree of coevolution, is inferred to be the one that is most likely to mediate the interaction between proteins P and Q.

Figure 21.8 shows the domain-level interactions between alpha (YBL099w) and beta (YJR121w) chains of F1-ATPase in *Saccharomyces cerevisiae*. RCDP will correctly predict the top-scoring domain pair (PF00006 in YBL099w and PF00006 in YJR121w) to be interacting. In this case, there is more than one domain pair mediating a given protein–protein interaction. Since RCDP is designed to find only the domain pair(s) that exhibits highest degree of coevolution, it may not be able to identify all the domain level interactions between the two interacting proteins. It is possible that the highest scoring domain pair may not necessarily be an interacting domain pair. This could be due to what Jothi et al. refer to as the "uncorrelated set of correlated mutations" phenomenon, which may disrupt coevolution of proteins/domains. Since the underlying similarity of phylogenetic trees approach solely relies on coevolution principle, such disruptions can cause false predictions. RCDP's prediction accuracy was estimated to be about 64%. A naive random method that picks an arbitrary domain pair out of all possible domain pairs between the two interacting proteins is expected to have a prediction accuracy of 55% [31,44]. RCDP's prediction accuracy of 64% is significant considering the fact that Nye et al. [44] showed, using a different dataset, that the naive random method performs as well as Sprinzak and Margalit's association method [56], Deng et al.'s maximum likelihood estimation approach [13], and their own lowest p-value method, all of which are discussed in the following section. For a detailed analysis of RCDP and its limitations, we refer the reader to [31].

21.3.2 Predicting Domain Interactions from Protein–Protein Interaction Network

In this section, we describe computational methods to predict interacting domain pairs from an underlying protein–protein interaction network. To begin with, all proteins in the protein–protein interaction network are first assigned with domains using HMM profiles. Interaction between two proteins typically (albeit not always) involves binding of pair(s) of domains. Recently, several of computational method have been proposed that, based on the assumption that each protein–protein interaction is mediated by one or more domain–domain interactions, attempt to recover interacting domains.

We start by introducing the notations that will be used in this section. Let $\{P_1, \ldots, P_N\}$ be the set of proteins in the protein–protein interaction network and $\{D_1, \ldots, D_M\}$ be the set of all domains that are present in these interacting proteins. Let $\mathcal{I} = \{(P_{mn})|m, n = 1, \ldots, N\}$ be the set of protein pairs observed experimentally to interact. We say that the domain pair D_{ij} belongs to protein pair P_{mn} (denoted by $D_{ij} \in P_{mn}$) if D_i belongs to P_m and D_j belongs to P_n or vice versa. Throughout this section, we will assume that all domain pairs and protein pairs are unordered, that is, X_{ab} is the same as X_{ba}. Let N_{ij} denote the number of occurrences of domain pair D_{ij} in all possible protein pairs and let $\hat{N}_{ij}$ be the number of occurrences of D_{ij} in interacting protein pairs only.[2]

21.3.2.1 Association Method Sprinzak and Margalit [56] made the first attempt to predict domain–domain interactions from a protein–protein interaction network. They proposed a simple statistical approach, referred to as the *Association Method* (AM), to identify those domain pairs that are observed to occur in interacting protein pairs more frequently than expected by chance. Statistical significance of the observed domain pair is usually measured by the standard log-odds value A or probability α, given by

$$A_{ij} = \log_2 \frac{\hat{N}_{ij}}{N_{ij} - \hat{N}_{ij}}; \qquad \alpha_{ij} = \frac{\hat{N}_{ij}}{N_{ij}}. \tag{21.2}$$

The AM method is illustrated using a toy protein–protein interaction network in Fig. 21.9. It was shown that among high scoring pairs are pairs of domains that are know to interact, and a high α value can be used as a predictor of domain–domain interaction.

21.3.2.2 Maximum Likelihood Estimation Approach Following the work of Sprinzak and Margalit, several related methods have been proposed [13,42]. In particular, Deng et al. [13] extended the idea behind the association method and

[2]Not all the methods described in this section use unordered pairings. Some of them use ordered pairings, that is, X_{ab} is not the same as X_{ba}. Depending on whether one uses ordered or unordered pairing, the number of occurrences of a domain pair in a given protein pair is different. For example, let protein P_m contain domains D_x and D_y and let protein P_n contain domains D_x, D_y, and D_z. The number of occurrences of domain pair D_{xy} in protein pair P_{mn} is four if ordered pairing is used and two if unordered pairing is used.

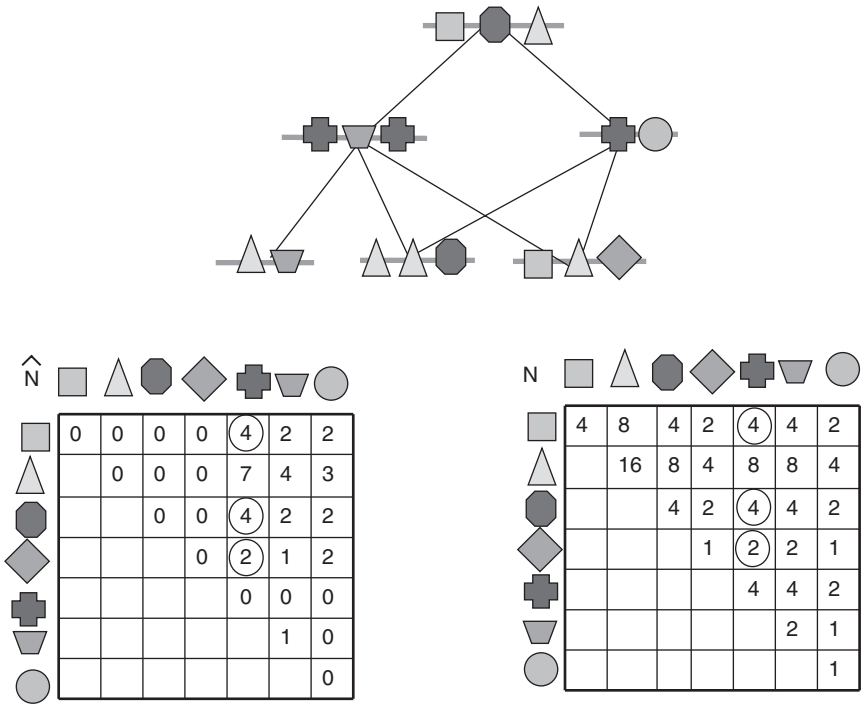


FIGURE 21.9 Schematic illustration of the association method. The toy protein–protein interaction network is given in the upper panel. The constituent domains of all the proteins in the network are represented using polygons of varying shapes. The lower panel shows domain pair occurrence tables $\hat{N}$ and N. Each entry $\hat{N}_{i,j}$ represents the number of times the domain pair (i, j) occurs in interacting protein pairs, and each entry $N_{i,j}$ represents the number of times (i, j) occurs in all protein pairs. A domain pair is counted only once even if it occurs more than once between a protein pair. Three domain pairs with maximum scores are encircled.

proposed a maximum likelihood approach to estimate the probability of domain–domain interactions. Their expectation maximization algorithm (EM) computes domain interaction probabilities that maximize the expectation of observing a given protein–protein interaction network $\mathcal{N}et$. An important feature of this approach is that it allows for an explicit treatment of missing and incorrect information (in this case, false negatives and false positives in the protein–protein interaction network).

In the EM method, protein–protein and domain–domain interactions are treated as random variables denoted by P_{mn} and D_{ij}, respectively. In particular, we let $P_{mn} = 1$ if proteins P_m and P_n interact with each other, and $P_{mn} = 0$ otherwise. Similarly, $D_{ij} = 1$ if domains D_i and D_j interact with each other, and $D_{ij} = 0$ otherwise. The probability that domains D_i and D_j interact is denoted by $\mathcal{P}r(D_{ij}) = \mathcal{P}r(D_{ij} = 1)$. The probability that proteins P_m and P_n interact is given by

$$\mathcal{P}r(P_{mn} = 1) = 1 - \prod_{D_{ij} \in P_{mn}} (1 - \mathcal{P}r(D_{ij})). \tag{21.3}$$

Random variable $\mathcal{O}_{mn}$ is used to describe the experimental observation of protein–protein interaction network. Here, $\mathcal{O}_{mn} = 1$ if proteins P_m and P_n were observed to interact (that is $P_{mn} \in \mathcal{I}$), and $\mathcal{O}_{mn} = 0$ otherwise. False negative rate is given by $f_n = \mathcal{P}r(\mathcal{O}_{mn} = 0 \mid P_{mn} = 1)$, and false positive rate is given by $f_p = \mathcal{P}r(\mathcal{O}_{mn} = 1 \mid P_{mn} = 0)$. Estimations of false positive rate and false negative rate vary significantly from paper to paper. Deng et al. estimated f_n and f_p to be 0.8 and $2.5E-4$, respectively.

Recall that the goal is to estimate $\mathcal{P}r(D_{ij}), \forall_{ij}$ such that the probability of the observed network $\mathcal{N}et$ is maximum. The probability of observing $\mathcal{N}et$ is given by

$$\mathcal{P}r(\mathcal{N}et) = \prod_{P_{mn} \mid \mathcal{O}_{mn}=1} \mathcal{P}r(\mathcal{O}_{mn} = 1) \prod_{P_{mn} \mid \mathcal{O}_{mn}=0} \mathcal{P}r(\mathcal{O}_{mn} = 0), \tag{21.4}$$

where

$$\mathcal{P}r(\mathcal{O}_{mn} = 1) = \mathcal{P}r(P_{mn} = 1)(1 - f_n) + (1 - \mathcal{P}r(P_{mn} = 1))f_n \tag{21.5}$$

$$\mathcal{P}r(\mathcal{O}_{mn} = 0) = 1 - \mathcal{P}r(\mathcal{O}_{mn} = 1). \tag{21.6}$$

The estimates of $\mathcal{P}r(D_{ij})$ are computed iteratively in an effort to maximize $\mathcal{P}r(\mathcal{N}et)$. Let $\mathcal{P}r(D_{ij}^t)$ be the estimation of $\mathcal{P}r(D_{ij})$ in the tth iteration and let D^t denote the vector of $\mathcal{P}r(D_{ij}^t), \forall_{ij}$ estimated in the tth iteration. Initially, values in D^0 can all be set the same, or those estimations obtained using the AM method. Note that each estimation of D^{t-1} defines $\mathcal{P}r(P_{mn} = 1)$ and $\mathcal{P}r(\mathcal{O}_{mn} = 1)$ using Equations 21.3 and 21.4. These values are, in turn, used to compute D^t in the current iteration as follows. First, for each domain pair D_{ij} and each protein pair P_{mn} the expectation that domain pair D_{ij} physically interacts in protein pair P_{mn} is estimated as

$$E(D_{ij}\text{interacts in } P_{mn}) = \begin{cases} \frac{\mathcal{P}r(D_{ij}^{t-1})(1-f_n)}{\mathcal{P}r(\mathcal{O}_{mn}=1)} & \text{if } P_{mn} \in \mathcal{I} \\ \frac{\mathcal{P}r(D_{ij}^{t-1})f_n}{\mathcal{P}r(\mathcal{O}_{mn}=0)} & \text{otherwise.} \end{cases} \tag{21.7}$$

The values of $\mathcal{P}r(D_{ij}^t)$ for the next iteration are then computed as

$$\mathcal{P}r(D_{ij}^t) = \frac{1}{N_{ij}} \sum_{P_{mn} \mid D_{ij} \in P_{mn}} E(D_{ij}\text{interacts in} P_{mn}). \tag{21.8}$$

Thus, similar to the AM method, the EM method provides a scoring scheme that measures the likelihood of a given domain pair interaction.

Since our knowledge of interacting domain pairs is limited (only a small fraction of interacting domains pairs have been inferred from crystal structures), it is not clear as to how any two methods predicting domain interactions can be compared. Deng et al. [13] compared the performance of their EM method to that of Sprinzak and Margalit's AM method [56] by assessing how well the domain–domain interaction predictions by the two methods can, in turn, be used to predict protein–protein interactions. For the AM method, $\mathcal{P}r(D_{ij})$ in Equation 21.3 is replaced by α_{ij}. Thus, rather than performing a direct comparison of predicted interacting domain pairs, they tested the method that leads to a more accurate prediction of protein–protein interactions. It was shown that the EM method outperforms the AM method significantly [13]. This result is not surprising considering the fact that the values of $\mathcal{P}r(D_{ij})$ in the EM method are computed so as to maximize the probability of observed interactions. Comparison of domain interaction prediction methods based on how well they predict protein–protein interaction is, however, not very satisfying. The correct prediction of protein interactions does not imply that the interacting domains have been correctly identified.

21.3.2.3 Domain Pair Exclusion Analysis (DPEA) An important problem in inferring domain interactions from protein interaction data using the AM and EM methods is that the highest scoring domain interactions tend to be nonspecific. The difference between specific and nonspecific interactions is illustrated in Fig. 21.10. Each of the interacting domains can have several paralogs within a given organism—several instances of the same domain. In a highly specific (nonpromiscuous) interaction, each such instance of domain D_i interacts with a unique instance of domain D_j (see Fig. 21.10a). Such specific interactions are likely to receive a low score by methods (AM and EM) that detect domain interactions by measuring the probability of interaction of corresponding domains. To deal with this issue, Riley et al. [52] introduced a new method called *domain pair exclusion analysis* (DPEA). The idea behind this method is to measure, for each domain pair, the reduction in the likelihood of the protein–protein interaction network if the interaction between this domain pair were to be disallowed. This is assessed by comparing the results of executing an expectation maximization protocol under the assumption that all pairs of domains can interact and that a given pair of domains cannot interact. The E-value is defined to be the ratio of the corresponding likelihood estimators. Figure 21.10b and c shows real-life examples with low θ scores and a high E-values.

The expectation maximization protocol used in DPEA is similar to that used in the EM method but performed under the assumption that the network is reliable (no false positives). The DPEA method has been compared to the EM and AM methods by measuring the frequency of retrieved (predicted) domain pairs that are known to interact (based on crystal structure evidence as inferred in iPFAM [16]). Riley et al. [52] showed that the DPEA method outperforms the AM and EM methods by a significant margin.

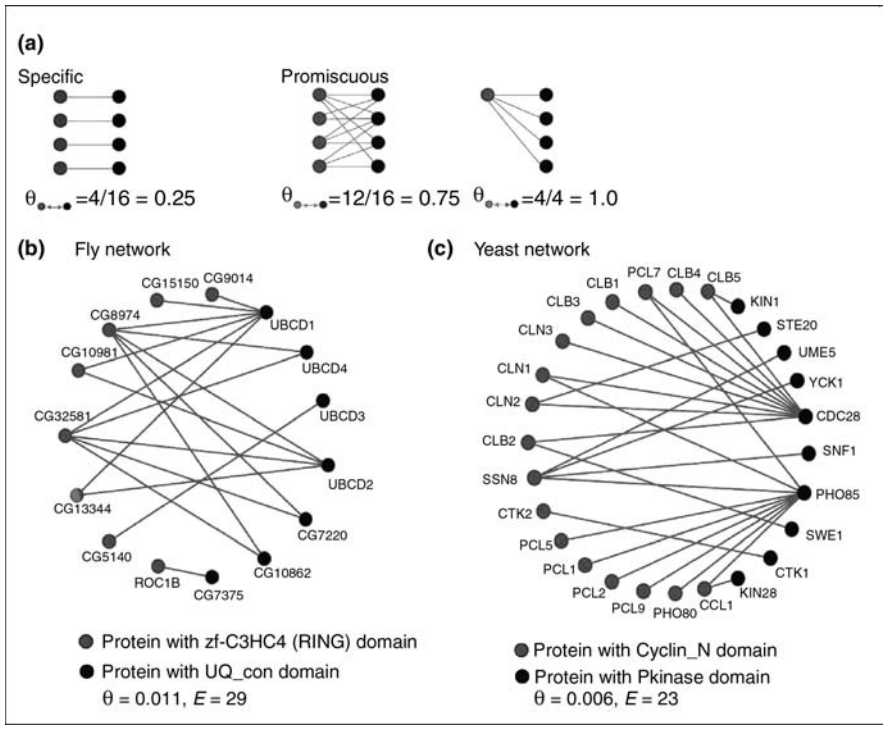


FIGURE 21.10 (**a**) Promiscuous and specific interactions; (**b–c**) Examples of two domain–domain interactions scored highly by the *E*-value method (score E) but missed by the EM method (score θ). Image reprinted from [52] with permission.

21.3.2.4 Lowest p-value method The lowest p-value method, proposed by Nye et al. [44], is an alternate statistical approach to predict domain–domain interactions. The idea behind this approach is to test, for every domain pair $D_{ij} \in P_{mn}$, the null hypothesis $\mathcal{H}_{ij}$ that the interaction between proteins P_m and P_n is independent of the presence of domain pair D_{ij}. They also consider a global null hypothesis $\mathcal{H}_\infty$ that the interaction between proteins P_m and P_n is entirely unrelated to the domain architectures of proteins. There are two specific assumptions made by this method, which were not made by other network-based approaches. First, every protein interaction is assumed to be mediated by exactly one domain–domain interaction. Second, each occurrence of a domain in a protein sequence is counted separately.

To test the hypothesis $\mathcal{H}_{ij}$, for each domain pair D_{ij}, consider the following two-by-two matrix X_{ij}:

	D_{ij}	Domain Pairs Other Than D_{ij}
Interacting domain pairs	$X_{ij}(1, 1)$	$X_{ij}(1, 2)$
Noninteracting domain pairs	$X_{ij}(2, 1)$	$X_{ij}(2, 2)$

In particular, $X_{ij}(1, 1)$ denotes the number of times domain pair D_{ij} is in physical interaction, and $X_{ij}(1, 2)$ denotes the number of times domain pairs other than D_{ij} interact. The method for estimating the values of table X_{ij} is given later in this subsection. Given the matrix X_{ij}, the log-odds score s_{ij} for domain D_{ij} is defined as

$$s_{ij} = \log \frac{X_{ij}(1, 1)/X_{ij}(2, 1)}{X_{ij}(1, 2)/X_{ij}(2, 2)} \tag{21.9}$$

The score s_{ij} is then converted into a p-value measuring the probability that hypothesis $\mathcal{H}_{ij}$ is true. This is done by estimating how likely a score at least this high can be obtained by chance (under hypothesis $\mathcal{H}_\infty$). To compute the p-value, the domain composition within the proteins is randomized. During the randomization procedure, the degree of each node in the protein–protein interaction network remains the same. The details of the randomization procedure exceeds the scope of this chapter and for the complete description we refer the reader to [44].

Finally, we show how to estimate the values in table X_{ij}. Value $X_{ij}(1, 1)$ is computed as the expected number of times domain pair D_{ij} mediates a protein–protein interaction under the null hypothesis $\mathcal{H}_\infty$ given the experimental data $\mathcal{O}$:

$$E(D_{ij}) = \sum_{P_{mn}} \mathcal{P}r(P_{mn} = 1|\mathcal{O})\mathcal{P}r(D_{ij} = 1|P_{mn} = 1), \tag{21.10}$$

where $\mathcal{P}r(P_{mn} = 1|\mathcal{O})$ is computed from the approximations of false positive and false negative rates in a way similar to that described in the previous subsection. The computation of $\mathcal{P}r(D_{ij} = 1|P_{mn} = 1)$ takes into account multiple occurrences of the same domain in a protein chain. Namely, let N_{ij}^{mn} be the number of possible interactions between domains D_i and D_j in protein pair P_{nm}. Then

$$\mathcal{P}r(D_{ij} = 1|P_{mn} = 1) = \frac{N_{ij}^{mn}}{\sum_{D_{kt}} N_{kt}^{mn}}, \tag{21.11}$$

and the value N_{ij} is, in this case, computed as

$$N_{ij} = \sum_{P_{kt}} N_{ij}^{kt}.$$

Consequently, the values of the table are estimated as follows:

$$X_{ij}(1, 1) = E(D_{ij})$$
$$X_{ij}(2, 1) = N_{ij} - E(D_{ij})$$

$$X_{ij}(1,2) = \sum_{D_{kt} \neq D_{ij}} E(D_{kt})$$

$$X_{ij}(2,2) = \sum_{D_{kt} \neq D_{ij}} (N_{kt} - E(D_{kt})).$$

Nye et al. [44] evaluated their method using a general approach introduced by them, which is described in Section 21.3.1. Namely, they predict that within the set of domain pairs belonging to a given interacting protein pair, the domain pair with the lowest p-value is likely to form a contact. To confirm this, they used protein complexes in the PQS database [27] (a database of quaternary states for structures contained in the Brookhaven Protein Data Bank (PDB) that were determined by X-ray crystallography) restricted to protein pairs that are meaningful in this context (e.g., at least one protein must be multidomain, both proteins contain only domain present in the yeast protein–protein interaction network used in their study, etc.). The results of this test for the lowest p-value method compared to random selection (random) and the AM and EM methods (discussed before) are presented in Fig. 21.11. It is striking from this comparison that the improvement these methods achieve over a random selection is small, although the improvement increases with the number of possible domain pair contacts.

21.3.2.5 Most Parsimonious Explanation (PE) Recently, Guimaraes et al. [26] introduced a new domain interaction prediction method called the most parsimonious explanation [26]. Their method relies on the hypothesis that interactions between proteins evolved in a parsimonious way and that the set of correct domain–domain interactions is well approximated by the minimal set of domain interactions necessary to justify a given protein–protein interaction network. The EM problem is formulated as a linear programming optimization problem, where each potential domain–domain contact is a variable that can receive a value ranging between 0 and 1 (called the *LP-score*), and each edge of the protein–protein interaction network corresponds to one linear constraint. That is, for each (unordered) domain pair D_{ij} that belongs to some interacting protein pair, there is a variable x_{ij}. The values of x_{ij} are computed using the linear program (LP):

$$\text{minimize} \sum_{D_{ij}} x_{ij} \tag{21.12}$$

$$\text{subject to} \sum_{D_{ij} \in P_{mn}} x_{ij} \geq 1, \text{ where } P_{mn} \in \mathcal{I}.$$

To account for the noise in the experimental data, a set of linear programs is constructed in a probabilistic fashion, where the probability of including an LP constraint in Equation 21.12 equals the probability with which the corresponding protein–protein interaction is assumed to be correct. The LP-score for a domain pair D_{ij} is then averaged over all LP programs. An additional randomization experiment is used

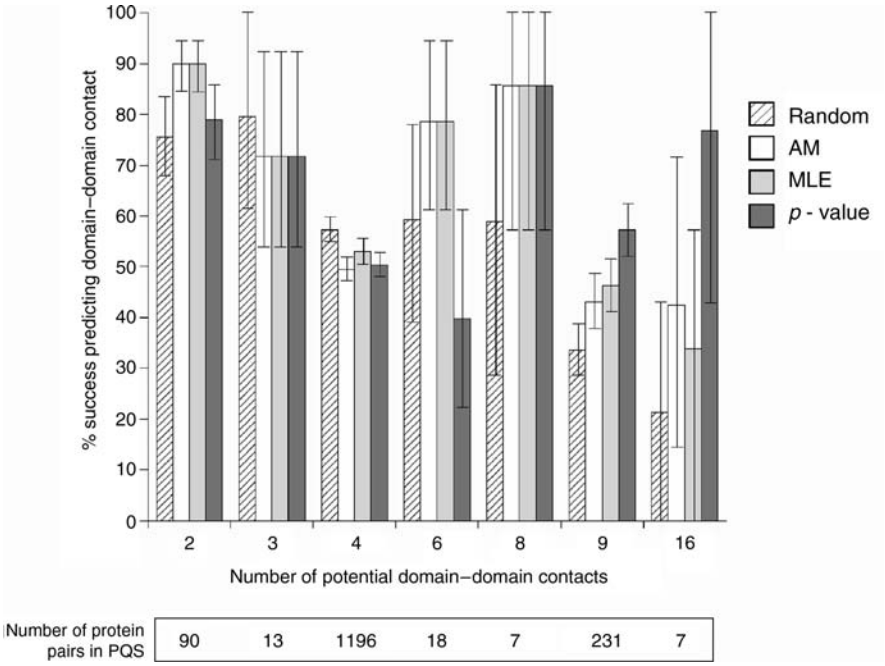


FIGURE 21.11 Domain–domain contact prediction results. The results are broken down according to the potential number of domain–domain contacts between protein pairs in the PQS database, and the number of protein pairs within each such category is shown at the bottom of the figure. The proportion of protein pairs for which four different prediction methods correctly predict a domain–domain contact is shown in the main graph. It is often observed in the PQS that several different domain pairs are in contact within each interacting protein pair. Any potential contact picked at random therefore has some probability of being confirmed as a contact in the PQS, and this baseline success rate is shown by the hatched bars. The error bars for the nonrandom methods correspond to a 90% confidence interval based on a binomial distribution assumption. Image reprinted from [44] with permission.

to compute p-values and prevent overprediction of interactions between frequently occurring domain pairs. Guimaraes at al. [26] demonstrated that the PE method outperforms the EM and DPEA methods.

GLOSSARY

Coevolution Coordinated evolution. It is generally agreed that proteins that interact with each other or have similar function undergo coordinated evolution.

Gene fusion A pair of genes in one genome is fused together into a single gene in another genome.

HMMer HMMer is a freely distributable implementation of profile HMM (hidden Markov model) software for protein sequence analysis. It uses profile HMMs to do sensitive database searching using statistical descriptions of a sequence family's consensus.

iPfam iPfam is a resource that describes domain–domain interactions that are observed in PDB crystal structures.

Ortholog Two genes from two different species are said to be orthologs if they evolved directly from a single gene in the last common ancestor.

PDB The protein data bank (PDB) is a central repository for 3D structural data of proteins and nucleic acids. The data, typically obtained by X-ray crystallography or NMR spectroscopy, are submitted by biologists and biochemists from around the world, released into the public domain, and can be accessed for free.

Pfam Pfam is a large collection of multiple sequence alignments and hidden Markov models covering many common protein domains and families.

Phylogenetic profile A phylogenetic profile for a protein is a vector of 1s and 0s representing the presence or absence of that protein in a reference set organisms.

Distance matrix A matrix containing the evolutionary distances of organisms or proteins in a family.

ACKNOWLEDGMENTS

This work was funded by the intramural research program of the National Library of Medicine, National Institutes of Health.

REFERENCES

1. HMMer. http://hmmer.wustl.edu
2. RPS-BLAST. http://www.ncbi.nlm.nih.gov/Structure/cdd/wrpsb.cgi
3. Altschuh D, Lesk AM, Bloomer AC, Klug A. Correlation of coordinated amino acid substitutions with function in viruses related to tobacco mosaic virus. *J Mol Biol* 1987;193(4):683–707.
4. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. *J Mol Biol* 1990;215(3):403–410.
5. Apic G, Gough J, Teichmann SA. Domain combinations in archaeal, eubacterial and eukaryotic proteomes. *J Mol Biol* 2001;310(2):311–325.
6. Atwell S, Ultsch M, De Vos AM, Wells JA. Structural plasticity in a remodeled protein–protein interface. *Science* 1997;278(5340):1125–1128.
7. Berger JM, Gamblin SJ, Harrison SC, Wang JC. Structure and mechanism of DNA topoisomerase II. *Nature* 1996;379(6562):225–232.
8. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, Bourne PE. The Protein Data Bank. *Nucl Acid Res* 2000;28(1):235–242.

9. Butland G et al. Interaction network containing conserved and essential protein complexes in escherichia coli. *Nature* 2005;433(7025):531–537. **[Q1]**
10. Chothia C, Gough J, Vogel C, Teichmann SA. Evolution of the protein repertoire. *Science* 2003;300(5626):1701–1703.
11. Dandekar T, Snel B, Huynen M, Bork P. Conservation of gene order: a fingerprint of proteins that physically interact. *Trends Biochem Sci* 1998;23(9):324–328.
12. Date SV, Marcotte EM. Discovery of uncharacterized cellular systems by genome-wide analysis of functional linkages. *Nat Biotechnol* 2003;21(9):1055–1062.
13. Deng M, Mehta S, Sun F, Chen T. Inferring domain–domain interactions from protein–protein interactions. *Genome Res* 2002;12(10):1540–1548.
14. Edgar RC. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucl Acid Res* 2004;32(5):1792–1797.
15. Enright AJ, Iliopoulos I, Kyrpides NC, Ouzounis CA. Protein interaction maps for complete genomes based on gene fusion events. *Nature* 1999;402(6757):86–90.
16. Finn RD, Marshall M, Bateman A. iPfam: visualization of protein–protein interactions in PDB at domain and amino acid resolutions. *Bioinformatics* 2005;21(3): 410–412.
17. Finn RD et al. Pfam: clans, web tools and services. *Nucleic Acids Res* 2006,34(Database issue):D247–D251. **[Q1]**
18. Gaasterland T, Ragan MA. Microbial genescapes: phyletic and functional patterns of ORF distribution among prokaryotes. *Microb Comp Genomics* 1998;3(4):199–217.
19. Gavin AC et al. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature* 2002;415(6868):141–147. **[Q1]**
20. Gertz J, Elfond G, Shustrova A, Weisinger M, Pellegrini M, Cokus S, Rothschild B. Inferring protein interactions from phylogenetic distance matrices. *Bioinformatics* 2003;19(16):2039–2045.
21. Giot L et al. A protein interaction map of drosophila melanogaster. *Science* 2003;302(5651):1727–1736. **[Q1]**
22. Glazko GV, Mushegian AR. Detection of evolutionarily stable fragments of cellular pathways by hierarchical clustering of phyletic patterns. *Genome Biol* 2004;5(5):R32.
23. Gobel U, Sander C, Schneider R, Valencia A. Correlated mutations and residue contacts in proteins. *Proteins* 1994;18(4):309–317.
24. Goh CS, Bogan AA, Joachimiak M, Walther D, Cohen FE. Co-evolution of proteins with their interaction partners. *J Mol Biol* 2000;299(2):283–293.
25. Goh CS, Cohen FE. Co-evolutionary analysis reveals insights into protein–protein interactions. *J Mol Biol* 2002;324(1):177–192.
26. Guimaraes K, Jothi R, Zotenko E, Przytycka TM. Predicting domain–domain interactions using a parsimony approach. *Genome Biol* 2006;7(11):R104.
27. Henrick K, Thornton JM. PQS: a protein quarternary structure file server. *Trends Biochem Sci* 1998;23(9):358–361.
28. Ho Y. et al. Systematic identification of protein complexes in saccharomyces cerevisiae by mass spectrometry. *Nature* 2002;415(6868):180–183. **[Q1]**
29. Ito T, Chiba T, Ozawa R, Yoshida M, Hattori M, Sakaki Y. A comprehensive two-hybrid analysis to explore the yeast protein interactome. *Proc Natl Acad Sci USA* 2001;98(8):4569–4574.

30. Jespers L, Lijnen HR, Vanwetswinkel S, Van Hoef B, Brepoels K, Collen D, De Maeyer M. Guiding a docking mode by phage display: selection of correlated mutations at the staphylokinase-plasmin interface. *J Mol Biol* 1999;290(2):471–479.
31. Jothi R, Cherukuri PF, Tasneem A, Przytycka TM. Co-evolutionary analysis of domains in interacting proteins reveals insights into domain–domain interactions mediating protein–protein interactions. *J Mol Biol* 2006. **[Q2]**
32. Jothi R, Kann MG, Przytycka TM. Predicting protein–protein interaction by searching evolutionary tree automorphism space. *Bioinformatics* 2005;21(Suppl 1):i241–i250.
33. Jothi R, Przytycka TM, Aravind L. Discovering functional linkages and cellular pathways using phylogenetic profile comparisons: a comprehensive assessment. Unpublished Manuscript, 2007.
34. Kann MG, Jothi R, Cherukuri PF, Przytycka TM. Predicting protein domain interactions from co-evolution of conserved regions. *Proteins* 2007. Forthcoming.
35. Krogan NJ. et al. Global landscape of protein complexes in the yeast saccharomyces cerevisiae. *Nature* 2006;440(7084):637–643. **[Q1]**
36. Li S. et al. A map of the interactome network of the metazoan c. elegans. *Science* 2004;303(5657):540–543. **[Q1]**
37. Marcotte EM, Pellegrini M, Ng HL, Rice DW, Yeates TO, Eisenberg D. Detecting protein function and protein–protein interactions from genome sequences. *Science* 1999;285(5428):751–753.
38. Metropolis N, Rosenbluth AW, Teller A, Teller EJ. Simulated annealing. *J Chem Phys* 1955;21:1087–1092.
39. Mirkin BG, Fenner TI, Galperin MY, Koonin EV. Algorithms for computing parsimonious evolutionary scenarios for genome evolution, the last universal common ancestor and dominance of horizontal gene transfer in the evolution of prokaryotes. *BMC Evol Biol* 2003;3:2.
40. Moyle WR, Campbell RK, Myers RV, Bernard MP, Han Y, Wang X. Co-evolution of ligand-receptor pairs. *Nature* 1994;368(6468):251–255.
41. Neher E. How frequent are correlated changes in families of protein sequences? *Proc Natl Acad Sci USA* 1994;91(1):98–102.
42. Ng SK, Zhang Z, Tan SH. Integrative approach for computationally inferring protein domain interactions. *Bioinformatics* 2003;19(8):923–929.
43. Notredame C, Higgins DG, Heringa J. T-Coffee: A novel method for fast and accurate multiple sequence alignment. *J Mol Biol* 2000;302(1):205–217.
44. Nye TM, Berzuini C, Gilks WR, Babu MM, Teichmann SA. Statistical analysis of domains in interacting protein pairs. *Bioinformatics* 2005;21(7):993–1001.
45. Overbeek R, Fonstein M, D'Souza M, Pusch GD, Maltsev N. Use of contiguity on the chromosome to predict functional coupling. *In Silico Biol* 1999;1(2):93–108.
46. Pazos F, Helmer-Citterich M, Ausiello G, Valencia A. Correlated mutations contain information about protein–protein interaction. *J Mol Biol* 1997;271(4):511–523.
47. Pazos F, Ranea JA, Juan D, Sternberg MJ. Assessing protein co-evolution in the context of the tree of life assists in the prediction of the interactome. *J Mol Biol* 2005;352(4):1002–1015.
48. Pazos F, Valencia A. Similarity of phylogenetic trees as indicator of protein–protein interaction. *Protein Eng* 2001;14(9):609–614.

49. Pazos F, Valencia A. In silico two-hybrid system for the selection of physically interacting protein pairs. *Proteins* 2002;47(2):219–227.
50. Pellegrini M, Marcotte EM, Thompson MJ, Eisenberg D, Yeates TO. Assigning protein functions by comparative genome analysis: protein phylogenetic profiles. *Proc Natl Acad Sci USA* 1999;96(8):4285–4288.
51. Ramani AK, Marcotte EM. Exploiting the co-evolution of interacting proteins to discover interaction specificity. *J Mol Biol* 2003;327(1):273–284.
52. Riley R, Lee C, Sabatti C, Eisenberg D. Inferring protein domain interactions from databases of interacting proteins. *Genome Biol* 2005;6(10):R89.
53. Saitou N, Nei M. The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Mol Biol Evol* 1987;4(4):406–425.
54. Sato T, Yamanishi Y, Kanehisa M, Toh H. The inference of protein–protein interactions by co-evolutionary analysis is improved by excluding the information about the phylogenetic relationships. *Bioinformatics* 2005;21(17):3482–3489.
55. Shindyalov IN, Kolchanov NA, Sander C. Can three-dimensional contacts in protein structures be predicted by analysis of correlated mutations? *Protein Eng* 1994;7(3): 349–358.
56. Sprinzak E, Margalit H. Correlated sequence-signatures as markers of protein–protein interaction. *J Mol Biol* 2001;311(4):681–692.
57. Thompson JD, Higgins DG, Gibson TJ. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucl Acid Res* 1994;22(22):4673–4680.
58. Uetz P. et al. A comprehensive analysis of protein–protein interactions in saccharomyces cerevisiae. *Nature* 2000;403(6770):623–627. **[Q1]**
59. Valencia A, Pazos F. Computational methods for the prediction of protein interactions. *Curr Opin Struct Biol* 2002;12(3):368–373.

Author Query

Q1 : Please provide the complete author list in Ref. [9,17,19,21,28,35,36,58]
Q2 : Please provide the volume and page number?
Q3 : Please check the change made