Identification of functional modules from conserved ancestral protein–protein interactions
Janusz Dutkowski* and Jerzy Tiuryn
Institute of Informatics, Warsaw University, Poland

ABSTRACT

Motivation: The increasing availability of large-scale protein–protein interaction (PPI) data has fuelled the efforts to elucidate the building blocks and organization of cellular machinery. Previous studies have shown cross-species comparison to be an effective approach in uncovering functional modules in protein networks. This has in turn driven the research for new network alignment methods with a more solid grounding in network evolution models and better scalability, to allow multiple network comparison.

Results: We develop a new framework for protein network alignment, based on reconstruction of an ancestral PPI network. The reconstruction algorithm is built upon a proposed model of protein network evolution, which takes into account phylogenetic history of the proteins and the evolution of their interactions. The application of our methodology to the PPI networks of yeast, worm and fly reveals that the most probable conserved ancestral interactions are often related to known protein complexes. By projecting the conserved ancestral interactions back onto the input networks we are able to identify the corresponding conserved protein modules in the considered species.

In contrast to most of the previous methods, our algorithm is able to compare many networks simultaneously. The performed experiments demonstrate the ability of our method to uncover many functional modules with high specificity.

Availability: Information for obtaining software and supplementary results are available at http://bioputer.mimuw.edu.pl/papers/cappi.

Contact: januszd@mimuw.edu.pl

1 INTRODUCTION

Large-scale protein–protein interaction (PPI) networks are now available for many model organisms. The arising challenge is to analyze this data to reveal the basic components and organization of cell machinery. One of the key approaches to the analysis of large PPI networks is through network alignment, or global comparison of two or more (multiple alignment) networks to identify similar regions (Sharan and Ideker, 2006). As in the case of biological sequence data, the comparative approach demonstrates high potential. The basic motivation behind it is the principle of evolutionary conservation of functional units of the cell. Just like homologous gene and protein sequences are retained in many species, the common patterns of protein interaction realizing essential functions are expectedly retained as well. Cross-species comparison of PPI networks may enable functional annotation of the proteins, prediction and verification of protein interactions and finally identification of essential cellular units.

Research in the area of network alignment has begun recently and is constantly progressing towards more advanced and general methods. One of the first studies of this type was performed by Matthews et al. (2001) who identified homologous pairs of interactions (one from each of the two PPI networks). In a more recent formulation of network alignment Kelley et al. (2003) construct an alignment graph, in which the nodes represent pairs of homologous proteins (one from each of the two compared networks) and edges represent conserved interactions. The alignment graph is used to search for conserved pathways (simple paths). This approach is extended in Sharan et al. (2005b) to enable the search for conserved complexes (dense subgraphs of the alignment graph). Similar procedure, but with different scoring of the edges is presented in Koyuturk et al. (2006). In Sharan et al. (2005a) the authors modify their previous procedure to align three (rather than two) networks simultaneously.

One of the drawbacks of the alignment graph is that it includes a node for every pair (or triplet) of similar proteins (one from each input network). The commonly used similarity functions (e.g. BLAST E-value threshold) generally impose a many-to-many correspondence between proteins, which causes the size of the alignment graph to grow exponentially with the number of aligned networks. Flannick et al. (2006) present a different approach to network alignment, which addresses this problem by greedily assigning the aligned proteins to non-overlapping homology classes and progressively aligning multiple input networks. The algorithm also allows searching for a larger class of conserved topologies defined by the user.

Another important issue is the evolutionary relevance of the identified alignments. As suggested in a recent review by Sharan and Ideker (2006), network alignment procedures should be more strongly rooted in an evolutionary model of how networks evolve. This is not quite achieved by the previous methods. Although Koyuturk et al. (2006) apply evolutionary-based scoring scheme and Flannick et al. (2006) reconstruct the most parsimonious history of protein evolution, these methods do not attempt to probabilistically model the evolution of protein interactions. This problem is addressed in two recent works by Berg and Lassig (2006) and Hirsh and Sharan (2006). The former work builds network alignments using a score function which incorporates link dynamics represented as a Markov process, while the latter assigns probabilities to interaction loss and emergence in a single evolutionary step leading from an ancestral module (complex) to the observed interaction patterns.

© 2007 The Author(s)
This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/2.0/uk/) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Here we present a different approach based explicitly on the phylogenetic history of proteins and a stochastic evolutionary model of interaction emergence, loss and conservation during each of the events, which have impact on the network.

We develop a novel approach to aligning PPI networks. At the heart of our procedure lies reconstruction of the conserved ancestral PPI (CAPPI) network. First, we reconstruct the hypothetical sequence of evolutionary events (duplications, deletions and speciations), by which the proteins of the input PPI networks evolved from their counterparts in the common ancestral network. In the next step, we determine the posterior probabilities of interaction between proteins at each stage of evolution up to and including the common ancestor network. The probability of protein interaction is calculated under a proposed stochastic model of network evolution. The topology of the ancestral network (and each network at every stage of evolution) is determined by the most probable interactions. Finally, we identify modules in the CAPPI network and project them back onto the input networks to determine the alignment. Compared to previous approaches the proposed framework provides more insight into the protein network evolution. Its parameters are clearly tied to the evolutionary events shaping the network. The algorithm also scales well with the number of species, allowing for simultaneous comparison of many networks. We apply the CAPPI procedure to align the PPI networks of *Saccharomyces cerevisiae*, *Drosophila melanogaster* and *Caenorhabditis elegans*, three of the largest publicly available PPI networks to date. Within the aligned network modules we detect a large number of known yeast complexes with high specificity, favorably comparing to the previous study on the same data.

The article is organized as follows: Section 2 presents a brief overview of our methodology. Section 3 provides the details of the CAPPI reconstruction procedure, presents the model of network evolution and describes the identification of conserved network modules. Section 4 presents the performed experiments and their outcome, as well as comparison to previously proposed methods. A brief summary of the work is given in Section 5.

2 APPROACH

The aim of network alignment is the identification of similar (conserved) network regions in two or more networks. Here we present a general framework for aligning protein interaction networks. We approach the problem by reconstructing the hypothetical ancestral PPI network based on the protein sequences and interactions found in the input networks. Nodes of the ancestral network represent ancestral proteins, descendants of which have been conserved in the observed input networks. Edges of the ancestral network are assigned weights, which denote the probability of interaction between the respective ancestral proteins. These probabilities are calculated as the posterior probabilities given the interaction data in the input networks and the evolutionary model, by which the input networks are assumed to have evolved from the common ancestral network. The regions of the ancestral network with high interaction probability can therefore be regarded as a merged representation of the common proteins and interactions that have been conserved in the input networks.

3 METHODS

We now present the details of our methods. The subsequent steps of the CAPPI procedure are outlined in Figure 1.

3.1 Reconstructing the phylogenetic history

We assume that we are given a number of PPI networks, each coming from a different species. The first step towards aligning the networks is determining the homology relationship between all proteins. We choose to split the proteins into non-overlapping families of (putatively) homologous proteins [a similar approach was presented in Flannick et al. (2006)]. The proteins in each cluster are believed to have descended from a common ancestral protein. In contrast to the approach taken by Flannick et al. (2006), where proteins are greedily assigned to families during the iterative alignment process (so as to maximize the alignment score), we choose to determine the homology relationship for all proteins directly by previously established methods for identification of protein families. Our grouping, therefore, has a greater chance of representing true homologous protein families. As one possibility, we could split the proteins according to the clusters of the COG database (Tatusov et al., 2003). However, to allow the application to all species and protein sequences we decide on a more general approach. Specifically, we determine the homologous clusters using TRIBE-MCL algorithm by Enright et al. (2002) with BLAST E-values as pairwise distances between proteins.

In the next step, for each group of homologous proteins we reconstruct its evolutionary history by means of phylogenetic analysis. We perform multiple alignment of the protein sequences using the CLUSTALW method (Higgins et al., 1994). Next, we calculate the distance matrix using the PROTDIST procedure and construct phylogenetic trees using the Neighbor Joining algorithm implemented in the Phylip package (Felsenstein, 2005).

For consistency with the evolutionary history, it is necessary to reconcile the gene tree of each protein class with the species tree of the aligned organisms (Page and Charleston, 1997). The reconciliation algorithm minimizing the duplication and loss score function is implemented in NOTUNG (Durand et al., 2006) and a more efficient reconciliation algorithm has been recently shown by Görecki and Tiuryn (2006).

Given the sequence of evolutionary events extracted from the reconciled trees we proceed to determine the posterior probability of interactions between proteins at each stage of evolution, given the interactions observed in the input networks. The reconstructed evolutionary history of each protein family serves as the backbone for the reconstruction of protein interactions. The formal description of the network reconstruction procedure, the proposed model of network evolution and the details of calculating the posterior probability of ancestral interactions are presented in the next section.

3.2 Reconstructing the ancestral network

Let $S_0$ be the set of species, for which we are given the observed PPI networks. We assume that we are given a phylogenetic tree of the considered species, in which leaves are labeled with the species from $S_0$ and inner nodes are labeled with unknown (hidden) predecessor species. The tree is determined by the indexed set of nodes $S = \{s_1, s_2, \ldots, s_n\}$ of all the species in the tree (observed and hidden) and three functions $LS, RS, F: \{1, \ldots, n\} \rightarrow \{null, 1, \ldots, n\}$. We will assume that the common ancestor (the root of the tree) is $s_1$. $LS(i)$ and $RS(i)$ return the index of the left and right son of species $s_i$ respectively (or null if the right (left) son does not exist) and $F(i)$ returns
Functional modules from conserved ancestral protein interactions

1) Input PPI networks

2) Cluster proteins
   with MCL using
   BLAST E-values
   as pairwise distances

3) Build reconciled gene trees

4) Compute the probability of each ancestral interaction
given protein history, observed interaction
   data and model of network evolution

Fig. 1. Overview of the CAPPI methodology. We start with the input PPI networks (color-coded fragments of three networks are shown at the top). We determine non-overlapping homologous protein groups via clustering. Next, we build gene trees for each protein group and reconcile the trees with a common species tree (reconciled gene trees for three protein groups with outgoing arrows are shown). Finally, we determine the probability of each ancestral protein interaction (dotted lines) based on the interactions observed in the input networks (mapped to the leaves of the trees) and on the sequence of evolutionary events (duplications, speciations and deletions) extracted from the reconciled gene trees.

the index of the father node (or null if \(i = 1\)). We also assume that the set of proteins in all observed species has been split into non-overlapping protein families (equivalence classes) and for each family we are given a gene tree, which is reconciled with the aforementioned species tree. From the reconciled gene trees we are able to extract the set of duplication events taking place in species \(s_i\), that is after the speciation event which established \(s_i\) and before the speciation event when \(s_i\) evolved into two distinct species. For the purpose of this discussion we will assume that we can impose an ordering on duplication events in each species\(^1\) (in practice this is hard to achieve, since we do not know the order of events which occur within different protein families; but we have found by simulation, that changing this specific order, has negligible effect on the final result). We will denote by \(G_{i,j} = (V_{i,j}, E_{i,j})\) the undirected graph representing the protein network of \(s_i\) after the \(j\)-th duplication event occurring in this species. The ancestral graph is denoted by \(G_{1,0}\). Graph \(G_{1,0}\) has exactly one protein from each protein family—the protein placed at the root of the appropriate gene tree. The sequence of duplication events in species \(s_i\) is given by \(D_i = (d_i^{(0)}, \ldots, d_i^{(m_i)})\), where \(d_i^{(l)} = (n_{p}, n_{a}, n_{b})\) denotes duplication of protein \(n_p \in V_{i,j-1}\) into two proteins \(n_a, n_b \in V_{i,j}\). The problem we consider is to determine the probability of interaction between nodes in the graph \(G_{1,0}\) and all its descendants, based on the observed graphs of the species in \(S_0\) and assuming the sequences of evolutionary events, by which each of the observed networks evolved from \(G_{1,0}\).

3.3 Duplication and divergence model of protein network evolution

The following model of network evolution motivated by the general duplication model (Sole et al., 2002; Pastor-Satorras et al., 2003) is used to determine the probability of observing graph \(G_{i,j}\) under the assumption of the sequence of speciations and duplications, by which \(G_{i,j}\) evolved from the ancestral network \(G_{1,0}\).

The model has four parameters: \(p_{d}, p_{s}, p_{d}, \delta\), and \(\delta\),

- We start with the ancestral graph \(G_{1,0}\),\(^2\) and perform a defined sequence of duplications and speciations.
- In case of duplication \(d_i^{(l)} = (n_p, n_a, n_b)\) graph \(G_{i,j}\) is constructed on the basis of \(G_{i,j-1}\) in the following way:
  - D1. All vertices besides \(n_p\) and edges which are not incident to \(n_p\) are copied from \(V_{i,j-1}\) to \(V_{i,j}\).
  - D2. Vertices \(n_a\) and \(n_b\) are added to \(V_{i,j}\).
  - D3. For each edge \(e_{i,j-1} \in E_{i,j-1}\) we add to \(E_{i,j}\) edges \(n_a n_x\) and \(n_b n_x\) independently, each with probability \(p_d\).

\(^2\)For simplicity, we will assume that the initial graph \(G_{1,0}\) does not contain self-loops. However, they are easily incorporated and necessary to model the predecessors of interactions between duplicate proteins.
D. For each vertex $x_i \in V_{S,j}$ such that $x_i,x_j \notin E_{S,j-1}$ we add to $E_{S,j}$ edges $nx_{nj}$ and $nx_{nj}$ independently, each with probability $\beta_{nx}[V_{S,j}]$.

- In case of speciation of species $s_j$ graph $G_{S,j,0}$ is constructed on the basis of $G_{S,m}$ in the following way:
  
  1. All vertices are copied from $V_{S,m} \to V_{S,j,0}$.
  2. Each edge $nx_{nj} \in E_{S,m}$ is added to $E_{S,j,0}$ independently with probability $p_{x}$.
  3. Each edge $nx_{nj} \notin E_{S,m}$ is added to $E_{S,j,0}$ independently with probability $\beta_{nx}[E_{S,j,0}]$.

Steps D3 and D4 associated with duplication are referred to as local or correlated divergence, because they only affect the edges of the newly added vertices. The steps following speciation can be referred to as global divergence, as they can affect any edge in the network. Wagner (2001) points out that duplicate gene products usually diverge quickly and lose common interactions. One reason for this is that there is greater tolerance for mutations of the newly duplicated proteins, because of functional and structural redundancy of the duplicates. This would suggest that $p_{x}$ the probability of edge conservation following duplication should generally be much lower than $p_{e}$ (the probability of edge conservation after speciation). Also, as suggested by Berg et al. (2004), the overall rate of interaction attachment and loss should be equal making the model parameters interrelated.

Under the proposed model the probability of interaction between proteins in network $G_{S,j}$ is determined by the interactions in the ancestral network $G_{S,0}$ and the assumed sequence of speciations and duplications, which led to the formation of $G_{S,j}$. In the following, we use the above model to infer the posterior probabilities of ancestral interactions given the observed input networks.

### 3.4 The most probable ancestral topology

For each graph $G_{S,j}$ and each pair of vertices $n_s, n_t \in V_{S,j}$ we will denote by $X_{n_sn_t}^{G_{S,j}}$ the binary random variable equal 1 when there exists an edge $n_sn_t \in E_{S,j}$, and 0 otherwise.

Assuming the duplication and divergence model described earlier the probability $P\left(X_{n_sn_t}^{G_{S,j}} = 1\right)$ depends on the existence or lack of an edge between the protein pair being the direct predecessor of the pair $(n_s, n_t)$. Let us consider the last evolutionary event which could affect the pair $(n_s, n_t)$. Three cases are possible:

1. **Vertex $n_s \in V_{S,j}$ was created by duplication $d_{l}^{(i)}$ from vertex $n_t \in V_{S,k-1}$, where $k \leq j$.**
   - Then we have $P\left(X_{n_sn_t}^{G_{S,j}} = 1\right) = \text{prob}$. (4) This is the parallel duplication of a functional module. To identify the modules of the ancestral network $G_{S,j}$, we assume a specific topology of functional modules. The candidate network regions were scored for fulfilling the desired structure. In contrast to the previous methods, we do not impose a predefined topology when searching for conserved modules. Instead, we specifically identify network regions, which are descendents of putative ancestral modules.
   - **Vertex $n_t \in V_{S,j}$ duplicated from vertex $n_t \in V_{S,k-1}$, where $k \leq j$. (symmetrical to i.)**
   - **Vertices $n_s, n_t \in V_{S,j}$ emerged by means of speciation from vertices $n_s, n_t \in V_{S,k,0}$.** We then have $P\left(X_{n_sn_t}^{G_{S,j}} = 1\right) = \text{prob}$.

The above dependencies can be represented using a Bayesian network (BN) model (Neapolitan, 2003). We will start the construction of the BN from the instantiated random variables, which represent the edges or non-edges in the observed graphs of the species in $S_d$. By considering the last duplication or speciation event we recursively determine the direct predecessor of each possible edge and assign the conditional probabilities as described above, until we reach the corresponding possible edge (random variable) in the ancestral graph.

Each random variable corresponding to a possible edge $n_sn_t$ depends on exactly one random variable denoting the edge (or non-edge) in the direct predecessor graph. Therefore, the considered BN will be in the form of a set of trees (forest), where each tree models the joint distribution of the random variables corresponding to interactions, which are descendents of one of the interactions in the ancestral graph.

Now we can formulate the problem of finding the ancestral topology as a Bayesian network inference problem. Precisely, we would like to determine for each $i$ and $j$ the posterior probability

$$P(V_{n_i,n_j}^{G_{S,j}} = 1|E)$$

of interaction between a protein pair $(n_i, n_j)$ in species $j$ after $j$-th duplication, given the set of instantiated variables $E$, which are the interactions and non-interactions between nodes in the networks of observed species. We assume that the prior probabilities of interactions between proteins in the ancestral network $G_{S,0}$ are given and we have the conditional probabilities linking each child node in the BN to its father. The problem we solve here is the classical problem of inference in Bayesian networks. The tree structure of our BN model enables an efficient solution using a simple version of the message-passing algorithm due to Pearl (Neapolitan, 2003).

In the above discussion, we have considered duplications and speciations, but left out (for simplicity) the protein deletion events present in the reconciled trees. In fact, deletions do not complicate the model, but only prune the set of possible interactions that have to be considered. Before we move on, let us also point out that in the above framework one can easily incorporate the probability of not observing an existing interaction or observing a false interaction. It would only require the inclusion of additional random variables at the leaves of our Bayesian tree models. We will not explore this issue further here, leaving it as a possible direction for future work.

### 3.5 Identifying conserved ancestral modules

Our ultimate goal is to determine the conserved functional modules in observed networks. Previously proposed network alignment approaches assumed a specific topology of functional modules. The candidate network regions were scored for fulfilling the desired structure. In contrast to the previous methods, we do not impose a predefined topology when searching for conserved modules. Instead, we specifically identify network regions, which are descendents of putative ancestral modules.

The ancestral network $G_{S,0}$ reconstructed according to the procedure described in previous sections is a complete graph in which each edge $n_sn_t$ is assigned a weight corresponding to the probability of interaction between the adjacent proteins $n_s$ and $n_t$. The subsets of vertices connected by highly weighted edges are likely to constitute a functional module. To identify the modules of the ancestral network we set an edge threshold value $t$ and eliminate from the graph $G_{S,0}$ all edges with weights below that threshold. As shown in the next section, the threshold value can be determined by observing gradual decomposition of the largest component. At low values of $t$ the nodes of the ancestral network form one giant component. As we raise the value of the threshold $t$, the giant component decomposes and many components with heavy edges are revealed. The cut-off value can further be refined by determining the level, at which the probability of interaction between two vertices in the ancestral graph is statistically significant compared to the background model. To estimate the level of edge significance we repeatedly run our algorithm on randomized versions of the input data maintaining the original homology relationship of the proteins and their phylogenetic history and permuting the protein interaction data. The original PPi
are randomized by redistributing the edges of the input networks, while maintaining their node degree sequences [similar technique was applied in Kelley et al. (2003); Koyuturk et al. (2006) and Sharan et al. (2005b)]. For each set of randomized networks we reconstruct the ancestral PPI network and count the number of edges with weights exceeding a given threshold. Next, we compute the false discovery rate (FDR) for multiple hypothesis testing. The FDR q-value for a given edge weight \( w \) is estimated as:

\[
q(w) = \frac{\sum_{i=1}^{N} C(w_i) / N}{C_{\text{real}}(w)},
\]

where \( C_{\text{real}}(w) \) is the number of edges with weights equal to or higher than \( w \) in the previously reconstructed ancestral network and \( C(w_i) \) \( (i = 1, \ldots, N) \) is the number of such edges in the \( i \)-th randomized network. We determine the threshold value \( t \) at which the edge weights are significant (e.g. \( q \)-value < 0.05). We then decompose the graph \( G_{1,0} \) by deleting the edges of \( G_{1,0} \) with weights lower than \( t \) and removing all nodes without any interactions. The remaining connected components of the network constitute the ancestral network modules.

With the putative ancestral modules at hand, we proceed to identify the respective descendant modules in each of the considered species. To this end we project the ancestral modules onto the input networks by mapping the nodes of the ancestral network (ancestral proteins) to their descendants (proteins) in the input networks and identify the conserved descendant interactions.

4 RESULTS AND DISCUSSION

We applied our method to search for conserved functional modules in the networks of yeast (\( S. \) cerevisiae), worm (\( C. \) elegans) and fly (\( D. \) melanogaster), three of the largest PPI networks available in public domain. In the following, we present our results and comparison to the previously proposed methods. All algorithms are used with their default parameters, except where noted.

4.1 Ancestral network reconstruction

We downloaded the PPI and protein sequence data from the database of interacting proteins (DIP) (Salwinski et al., 2004) (April 2006 download). In order to identify protein families, we performed MCL clustering (I = 1.2) of all the sequences available in the DIP database (including sequences from species other that the three species of interest) taking BLAST E-values as the pairwise distances. We found that better clustering results are achieved when a larger number of species are included. This is intuitively correct: as homologous proteins should be more similar to each other than to other proteins, larger and more diverse protein universe serves as a better background for homology identification. The clustering identified 6971 non-overlapping protein groups, 460 of which included protein representatives from all three species (yeast, fly and worm). We decided to limit further analysis to only these 460 clusters. Note that our protein groups thus have support in at least three species, which was rooted and reconciled with the species tree of yeast, fly and worm using the procedures implemented in NOTUNG. For each pair of protein families we calculated the posterior probability of interaction between the respective ancestral proteins according to the model described earlier. The choice of model parameters is discussed subsequently. In an analogous way, we computed the probability of each ancestral self loop (representing the interactions of proteins with themselves) based on the observed interactions within one protein family. The resulting ancestral network consists of 460 nodes and 460 x 459/2 + 460 = 106030 edges weighted by the probability of interaction of adjacent nodes.

4.1.1 Estimating parameters of edge dynamics

Here we discuss the choice of parameters \( p_s, \delta_s, p_d, \delta_d \) and the prior probability of interaction of any two given ancestral proteins. The model of network evolution presented in Section 3 is motivated by and related to the proteome growth model (Sole et al., 2002), often referred to as the general duplication and divergence model. The general duplication model included only the parameters of link conservation or emergence following the duplication event (corresponding to \( p_d \) and \( \delta_d \) in our model). Sole et al. (2002) denoted the probability of edge deletion following duplication event by \( \delta \) and set its value to 0.53, which was influenced by estimations of rates of link addition and deletion made by Wagner (2001). The general duplication model assumes, however, that after a duplication event one of the duplicates remains unchanged. Our model is symmetrical in the sense that both of the duplicates are subject to deletion/emergence of edges. Our first estimate of the probability of edge conservation \( p_d \) is therefore 0.7, which (assuming independence) gives almost the same joint probability of edge conservation in both duplicates as in Sole et al. (2002). The probability that a new edge is introduced was estimated by Sole et al. (2002) to be 0.06 divided by the number of nodes in the network. For the purpose of the experiments we used a constant value \( \delta_d = 0.01 \) (without normalizing it further by the size of the network at each stage, as assumed in our theoretical model). The probability of edge conservation after the speciation event was set to \( p_s = 0.95 \). This parameter is related to the overall divergence of the network over time and should be more conservative than the probability \( p_d \) related to the fast divergence of interactions of newly duplicated proteins. The probability of edge emergence after speciation was set to \( \delta_s = 0.01 \). As in the case of \( \delta_d \) we did not normalize it by the number of edges in our experiments. We set the prior probability of each possible ancestral interaction at 0.01, which is motivated by the observation that PPI networks are generally sparse with a small average node degree. As shown in the following section network reconstructed with these parameters yielded satisfactory results in terms of identified modules. We further studied the effect of individual parameters (additional results are available on our web page) on the reconstructed alignment and decided for a more conservative setting, raising the probability of edge conservation and lowering the probability of edge emergence. Precisely we set \( p_d = 0.85, p_s = 0.99, \delta_d = \delta_s = 0.001 \). This allowed us to
identify more modules, which were well conserved across the considered networks.

4.2 Decomposition of the giant component
As stated above, the reconstructed ancestral network is a complete weighted graph with 460 nodes. However, the weights of the edges (probabilities of interaction) vary considerably depending on the evidence supporting the given interaction in the input networks and the evolutionary history. As we gradually eliminate the edges with the lowest weights the initially connected graph decomposes into a large number of small components, suggesting the existence of network modules. This is consistent for a wide range of parameters we have tested. The exact transition point and speed of decomposition varies with the choice of model parameters (especially the choice of prior probability of interaction), however, the general phenomenon was always observed. The decomposition of the ancestral network constructed with the model parameters discussed earlier is presented in Figure 2.

![Fig. 2. Decomposition of the largest component (1a and 2a) and change in the number of modules and identified MIPS complexes (1b and 2b) with increasing edge threshold. The results for the less conservative parameters are presented in panels 1a and 1b and for the more conservative parameters (preferred) in panels 2a and 2b. In case of both settings, it is evident that as the giant component decomposes, many pure modules and in turn MIPS complex categories are identified. The nodes of the ancestral network not involved in any interactions are eliminated from the graph, which explains why the fraction of nodes in the largest component (plot 1a) increases suddenly at high threshold values. The optimal threshold values (in terms of identified pure modules) are indicated on plots 1a and 2a with blue circles.](image)

4.3 Edge weight threshold selection
The connected components of the ancestral graph identified at a sufficiently high edge threshold level determine the ancestral modules of conserved interactions. To determine the significance level we calculated the FDR edge weight $q$-values using randomized networks. We then deleted all edges with weights below threshold 0.48 ($q$-value of 0.049). We also eliminated from the graph all nodes without any interactions.

4.4 Conserved modules and quality assessment
The ancestral network decomposed by eliminating edges below the threshold value of 0.48 contains 40 modules (visualized in Figure 3). Overall, 75 nodes representing ancestors of conserved protein families are present in the identified modules. By projecting the ancestral nodes onto their present-day descendants we obtain an alignment consisting of 40 respective network regions (modules) in the three input networks. We have found that a large part of the detected conserved modules match well to the known yeast protein complexes collected in the MIPS database (Mewes et al., 2006). To formally evaluate the quality of this finding we compute the purity score for each identified module as proposed by Sharan et al. (2005b). The purity score of a module with respect to a given MIPS category is defined as the number of module’s proteins annotated to that category divided by the number of
all annotated proteins contained in that module. The module is defined to be pure if it contains at least three annotated proteins and at least half of these share the same annotation (purity $\geq 0.5$). The module is impure if it contains at least three annotated proteins and its purity with respect to any considered MIPS category is less than 0.5. All other modules are treated as not sufficiently annotated (unknown). Following previous studies (Sharan et al., 2005a, b) we only consider the annotations at MIPS level 3 and exclude annotations based on high-throughput experiments (category 550).

Overall, out of the 40 identified modules 14 are pure and 1 is impure. The 14 pure modules match to 16 MIPS categories summarized in Table 1. The largest ancestral module is found to be pure with respect to the MIPS category 360.10.20 (proteasome). The identified module projected onto the three PPI networks of the considered species is presented in Figure 4. The only one impure module we have found is composed of the ancestral nodes 109 and 54. Closer examination of this module yields interesting results. The module contains 15 annotated proteins assigned to five level 3 MIPS categories: 260.20.10 (4 proteins), 260.20.30 (4 proteins), 260.20.20 (4 proteins), 260.30.10 (2 proteins), 260.20.99 (1 protein). The module is thus pure with respect to level 2 category 260.20 (intracellular transport) and the interactions between the members of the respective level 3 categories are perhaps not unexpected.

The effect of edge threshold selection (discussed in the previous section) on the identified modules is visualized in Figure 2. The results for the chosen parameter values are presented in the bottom right panel. We observe that immediately after the decomposition of the giant component the number of pure modules increases to 14 and the number of impure modules drops to 1. Pure modules matching known protein complexes are connected to other pure modules by lighter edges, which disappear as the threshold is raised. It can also be seen that by choosing a sufficiently high threshold (0.96) we can eliminate the impure module from our solution and still identify seven pure modules. For comparison, we also plot the results for less restrictive parameters, which are perhaps better motivated by previous studies. Similar observations can be made with respect to these results and in fact, in this case, even with small threshold values we do not find any impure modules.

**4.5 Comparison to existing methods**

Our approach implemented in the CAPPI framework achieves two main goals. First, it is rooted in an evolutionary network growth model based on edge dynamics and phylogenetic information about the history of the network constituents. Second, it is able to simultaneously align multiple networks. Of the available network alignment methods only NetworkBLAST (Sharan et al., 2005a, b) and Graemlin (Flannick et al., 2006) have demonstrated the ability to align multiple (>2) networks and none of the methods apply evolutionary history reconstruction or probabilistic modeling of network growth to the extent presented here. A preliminary implementation of Graemlin is publicly available, however, the algorithm has many network dependent parameters, which have only been estimated for the SRINI networks used by the authors. The estimation of these parameters for other networks is not supported by the current implementation. NetworkBLAST was previously used to align the DIP networks of yeast, worm and fly (Sharan et al., 2005a) and the results provided by the authors enabled a straightforward preliminary comparison of the methods. Below we compare the results of the two methods using the MIPS complex database as a reference set of true functional modules.

NetworkBLAST experiments were performed on an earlier version of the DIP database (February 2004), which contained fewer interactions and protein sequences. In order to allow a fair comparison we have repeated our experiments on the same version of the database. The results are summarized in Table 2. Overall, the NetworkBLAST returned a much larger number of aligned modules. Also, a larger fraction of the returned modules were pure. However, the identified modules overlap considerably (up to 80% overlap was allowed by the authors) and among the 80 pure modules only 5 different MIPS categories are matched. In contrast, the alignments returned by CAPPI are non-overlapping. We provide CAPPI results for two edge weight thresholds. With a less restrictive edge threshold (corresponding to $q$-value $<0.05$) our method

![Fig. 3. The ancestral network modules identified with edge threshold value of 0.48. The pure modules (colored yellow) matching known protein complexes are described in Table 1. The projection of the largest module (in the top left corner) onto the input PPI networks is presented in Figure 4.](image-url)
returned 39 modules among which 10 were pure. The 10 pure modules matched 13 MIPS complex categories altogether. The only one impure module returned was the same as the impure module identified in case of 2006 database version described earlier.

Raising the edge threshold to a more significant level (corresponding to $q$-value < 0.03) resulted in a smaller number of pure modules and less identified MIPS categories. However, with this setting no impure modules were returned by CAPPI. At the same time CAPPI still identified two more MIPS categories than NetworkBLAST. We note that in case of both methods a considerable number of identified modules contain less than three annotated proteins (thus, they are not classified as being pure or impure). These modules may perhaps represent yet unknown protein complexes. Note that we intentionally do not formally define a sensitivity measure here, because the number of conserved complexes present in the DIP database is unknown. Instead we rely on the absolute number of matched MIPS complexes to compare the ability of the methods to identify true complexes. The assessment of the specificity of both methods is based on the number of impure modules, divided by the number of all identified modules. With the more restrictive edge threshold CAPPI achieves a perfect specificity score.

Table 1. MIPS categories matched by pure modules identified by CAPPI

| Module | Annotated proteins | Purity | MIPS category | Description |
|--------|--------------------|--------|---------------|-------------|
| 193 - 266 - 134 - 219 - 84 | 7 | 1 | 360.10.20 | Proteasome |
| 28 | 14 | 0.83 | 410.40.30 | Replication |
| 311 - 174 | 5 | 1 | 500.40.10 | Translation |
| 331 - 280 | 3 | 1 | 500.20.10 | Translation |
| 176 - 439 | 4 | 1 | 510.190.110 | Transcription |
| 176 - 439 | 4 | 1 | 510.190.40 | Transcription |
| 41 | 3 | 0.67 | 510.190.130 | Transcription |
| 199 - 256 - 261 | 4 | 0.5 | 510.70.20 | Transcription |
| 199 - 256 - 261 | 4 | 0.5 | 510.190.10 | Transcription |
| 153 - 125 | 4 | 1 | 260.50.20 | Intracellular transport |
| 117 | 3 | 1 | 260.30.20 | Intracellular transport |
| 91 - 49 | 4 | 1 | 440.30.10 | RNA processing |
| 338 - 29 - 359 | 5 | 0.6 | 440.30.10 | RNA processing |
| 143 - 226 | 3 | 1 | 140.20.20 | Cytoskeleton |
| 106 | 3 | 0.67 | 510.180.20 | DNA repair |
| 199 - 256 - 261 | 4 | 0.5 | 230.20.20 | Histone acetyltransferase |

Pure conserved modules identified by projecting each ancestral module onto the input PPI network and assigning the annotations from the MIPS database to the respective yeast proteins. The identifier of the ancestral module (see also Figure 3), the number of annotated yeast proteins, and purity with respect to matching MIPS category is shown. Note that some proteins (members of modules 176-439 and 199-256-261) are annotated to more than one MIPS category.

Fig. 4. The conserved modules identified by projecting ancestral module 193-266-134-219-84 (see also Figure 3) onto the networks of yeast, fly and worm. Seven of the nine yeast proteins are assigned to the MIPS category 360.10.20 (Proteasome). The families corresponding to ancestral nodes 193, 266, 134, 219 and 84 are colored red, yellow, blue, purple and green, respectively. All three species contain representatives of each protein family, but only interacting proteins are visualized to present truly conserved PPI regions. The overall small coverage of *C. elegans* protein interactions in DIP may explain the apparent weak conservation of this module in case of this species.
For completeness we also provide the results for the DIP 2006 version discussed in the previous section. We observe that at the lower edge threshold (corresponding to \(q\)-value < 0.05) CAPPI identified more pure modules and MIPS categories on the more complete 2006 DIP database than on the 2004 edition. The results obtained with the more restrictive threshold remained unchanged with the introduction of new proteins and interactions, indicating that the group of alignments with the best support in the data is the same for both database versions.

The above experiments show that CAPPI is able to identify a relatively high number of known functional modules, while at the same time avoiding false-positive (impure) modules. More extensive tests are still needed to assess the performance of CAPPI and compare it to other methods under various conditions and with different reference databases.

## 5 CONCLUSION

We have presented a new approach to PPI network alignment. The method is based on probabilistic modeling of link dynamics guided by reconstructed evolutionary history of the proteins. The parameters of our model are clearly tied to the evolutionary events shaping the PPI networks. We have shown that the proposed framework can successfully identify conserved functional modules in multiple networks. Other applications of the proposed methodology are possible. The basic novel principle behind our approach, the reconstruction of ancestral PPI network, can be applied not only to align different PPI networks, but also to predict unknown protein interactions and functions, and to integrate many sources of PPI data. It can also be useful in evaluating hypothesis about PPI network evolution and comparing different evolutionary models. We have implemented our method using established algorithms for protein family detection and phylogeny reconstruction. However, the basic components of our framework can easily be replaced by better counterparts if such should become available. Our future studies will be directed towards formal analysis of the proposed model of network evolution, as well as the development of methods of comparing the identified conserved network modules with respect to the phylogenetic history.

### ACKNOWLEDGEMENTS

We would like to thank Bartek Wilczyński, Bogusław Kluge and Norbert Dojer for helpful discussions and Trey Ideker for providing the NetworkBLAST results. This work was supported by the Polish Ministry of Science grants No 3 T11F 021 28 and PBZ-MNiI-2/1/2005.

Conflict of Interest: none declared.

### REFERENCES

Berg,J. and Lassig,M. (2006) Cross-species analysis of biological networks by Bayesian alignment. PNAS, 103, 10967.
Berg,J. et al. (2004) Structure and evolution of protein interaction networks: a statistical model for link dynamics and gene duplications. BMC Evolution. Biol., 4, 51.
Durand,D. et al. (2006) A hybrid micro-macroevolutionary approach to gene tree reconstruction. J. Comput. Biol., 13, 320–335.
Enright,A.J. et al. (2002) An efficient algorithm for large-scale detection of protein families. Nucleic Acids Res., 30, 1575–1584.
Felsenstein,J. (2005) PHYLP (Phylogeny Inference Package) version 3.6. Department of Genome Sciences, University of Washington, Seattle, US. Distributed by the author.
Flannick,J. et al. (2006) Graemlin: general and robust alignment of multiple large interaction networks. Genome Res., 16, 1169–1181.
Görecki,P. and Tiuryn,J. (2006) Inferring phylogeny from whole genomes. Bioinformatics, 23, e116 – e122.
Higgins,D. et al. (1994) CLUSTALW: improving the sensitivity of progressive-multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. Nucleic Acids Res., 22, 4673–4680.
Hirsh,E. and Sharon,R. (2006) Identification of conserved protein complexes based on a model of protein network evolution. Bioinformatics, 23, e170–e176.
Kelley,B.P. et al. (2003) Conserved pathways within bacteria and yeast as revealed by global protein network alignment. Proc. Natl Acad. Sci. USA, 100, 11394–11399.
Koyuturk,M. et al. (2006) Pairwise alignment of protein interaction networks. J. Comput. Biol., 13, 182–199.
Matthews,L.R. et al. (2001) Identification of potential interaction networks using sequence-based searches for conserved protein-protein interactions or ‘interologs’. Genome Res., 11, 2120–2126.
Mewes,H.W. et al. (2006) MIPS: analysis and annotation of proteins from whole genomes in 2005. Nucleic Acids Res., 34, D109–D112.
Neapolitan,R.E. (2003) Learning Bayesian Networks. NJ: Upper Saddle River, Prentice Hall.
Page,R. and Charleston,M. (1997) From gene to organisinal phylogeny: reconciled trees and the gene treespecies tree problem. Mol Phylogenet Evol., 7, 231–240.
Pastor-Satorras,R. et al. (2003) Evolving protein interaction networks through gene duplication. J. Theor. Biol., 222, 199–210.
Salwinski,L. et al. (2004) The database of interacting proteins: 2004 update. Nucleic Acids Res., 32, D449.
Sharan, R. and Ideker, T. (2006) Modeling cellular machinery through biological network comparison. Nat. Biotechnol., 24, 427–433.
Sharan, R. et al. (2005a) Conserved patterns of protein interaction in multiple species. Proc. Natl Acad. Sci. USA, 102, 1974–1979.
Sharan, R. et al. (2005b) Identification of protein complexes by comparative analysis of yeast and bacterial protein interaction data. J. Comput. Biol., 12, 835–846.

Sole, R.V. et al. (2002) A model of large-scale proteome evolution. Adv. Compl. Syst., 5, 43.
Tatusov, R.L. et al. (2003) The COG database: an updated version includes eukaryotes. BMC Bioinformatics, 4.
Wagner, A. (2001) The yeast protein interaction network evolves rapidly and contains few redundant duplicate genes. Mol. Biol. Evol., 18, 1283–1292.