

Supporting Methods

We model all protein-protein interaction data of an organism using an interaction graph, whose vertices are the organism’s interacting proteins, and whose edges represent pairwise interactions between distinct proteins. A protein subnetwork translates under this representation to a subgraph that approximates a predefined structure. For instance, a linear pathway will correspond to a path in this graph and a protein complex will correspond to a dense subgraph, which we call a cluster.

Estimation of interaction probabilities

Several authors have suggested methods for evaluating the reliabilities of protein interactions [1, 2, 3]. Here, we adapt a method by Bader *et al.* [1] and assign confidence values to protein interactions using a logistic regression model. For a given species, our model represents the probability of a true interaction as a function of three observed random variables on a pair of proteins: (*i*) the number of times an interaction between the proteins was experimentally observed; (*ii*) the Pearson correlation coefficient of expression measurements for the corresponding genes; and (*iii*) the proteins’ small world clustering coefficient [4]. We describe these variables in detail below.

The number of observations was shown by several authors (e.g., ref. [2]) to be predictive of the reliability of an interaction. For yeast, the most well studied organism in this respect, we used the number of references for an interaction as its number of observations. For the other two species, only one large-scale interaction study is available. Hence, we defined the number of observations as the number of times the interaction was observed in the corresponding study.

Let x and y be two m -long vectors of expression levels for two genes. The Pearson correlation coefficient between the two vectors is defined as:

$$\rho = \frac{\frac{1}{m} \sum_{i=1}^m x_i y_i - \bar{x} \bar{y}}{\sigma_x \sigma_y}$$

where \bar{x}, \bar{y} are the sample means and σ_x, σ_y are the standard deviations of x and y , respectively. The correlation coefficient quantifies the similarity of expression between two genes and was shown to be correlated to whether the corresponding proteins interact or not [5, 6]. We used the following expression data sets: yeast expression data over 794 conditions were obtained from Stanford Microarray Database (SMD) [7]; fly expression data over 88 conditions were obtained from ref. [8] and another 170 profiles were retrieved from SMD (the latter spanned only one third of the genome); worm expression data over 553 conditions were obtained from ref. [9].

For proteins v and w , denote the sets of proteins that interact with them by $N(v)$ and $N(w)$, respectively. Let N be the total number of proteins in the network. The small-world clustering coefficient for v and w is:

$$C_{vw} = -\log \sum_{i=|N(v) \cap N(w)|}^{\min\{|N(v)|, |N(w)|\}} \frac{\binom{|N(v)|}{i} \binom{N-|N(v)|}{|N(w)|-i}}{\binom{N}{|N(w)|}}$$

The clustering coefficient was suggested by Goldberg *et al.* [4] to account for similarity in network connections.

According to the logistic distribution, the probability of a true interaction T_{uv} given the three input variables, $X = (X_1, X_2, X_3)$, is:

$$Pr(T_{uv}|X) = \frac{1}{1 + \exp(-\beta_0 - \sum_{i=1}^3 \beta_i X_i)}$$

where β_0, \dots, β_3 are the parameters of the distribution. Given training data, one can optimize the distribution parameters so as to maximize the likelihood of the data. To this end we used the *glmfit* function of MATLAB, where the training data was chosen as follows:

- Positive examples: For yeast, we used the MIPS [10] interaction data, which is an accepted gold standard. For other species, no such gold standard was available. Hence, we considered an interaction to be true if MIPS contained an interaction for putatively orthologous proteins in yeast (BLAST E-value < 10^{-10}).
- Negative examples: we tried two choices of negative training data. The first considers random pairs of proteins; the second, motivated by the abundance of false positives in protein interaction data, considers random observed interactions as true negatives. We performed 5-fold cross-validation experiments to evaluate the two choices. In each iteration of the cross-validation, we hid one fifth of the interaction labels and tested the prediction accuracy with respect to this held-out data. Defining negative interactions as randomly observed interactions yielded better results in the cross-validation experiments, and this definition was used in the sequel. We treated the chosen negative data as noisy indications that the corresponding interactions are false, and assigned those interactions a probability of 0.1397 for being true, where this value was optimized by using cross-validation.

Altogether we collected 1,006 positive examples and 1,006 negative examples for yeast; 92 positive and 92 negative examples for fly; and 24 positive and 50 negative examples for worm. Histograms of the interaction probabilities learned for each species are presented in Fig. 5.

Subnetwork Conservation

Our goal was to identify protein subnetworks that approximate a given structure and are conserved across a group of k species of interest, where in the present study we focused on $k = 2, 3$. A structure is specified as a property on graphs, e.g., being a path or being a clique, and sets our expectations with respect to an interaction subgraph that approximates that structure. For instance, a subgraph that corresponds to a clique subnetwork should involve densely interacting proteins.

Conservation of network structure requires the fulfillment of two conditions: (i) the set of subnetwork interactions within each species should approximate the desired structure; and (ii) there should exist a (many-to-many) correspondence between the sets of proteins exhibiting the structure in the different species, so that groups of k proteins, one from each species, induced by this correspondence, represent k sequence-similar proteins.

To capture these conservation requirements and to allow efficient search for conserved subnetworks we define a network alignment graph. Each node in this graph corresponds to a group of k sequence-similar proteins, one from each species. Each edge in the graph represents a conserved interaction between the proteins that occur in its end nodes. Two proteins are considered to have sufficient sequence similarity if their BLAST E-value is smaller than 10^{-7} (corresponding to a Bonferroni-corrected P value of 0.01), and each is among the 10 best BLAST matches of the other. A group of k distinct proteins, one from each species, comprise a node, if the group cannot be split into two parts with no sequence similarity between them. For $k = 2, 3$, this condition translates to the requirement that every protein in the group has at least one other sequence-similar protein in the group. Two nodes (p_1, \dots, p_k) and (q_1, \dots, q_k) in the graph are connected by an edge if and only if one of the following conditions is met with respect to the protein pairs (p_i, q_i) : (i) one pair of proteins directly interacts and all other pairs include proteins with distance at most two in the corresponding interaction maps; (ii) all protein pairs are of distance exactly two in the corresponding interaction maps; or (iii) at least $\max\{2, k - 1\}$ protein pairs directly interact. Note that it may be the case that for some i , $p_i = q_i$; we then consider the pair (p_i, q_i) to have distance 0.

A subgraph of the network alignment graph corresponds to a conserved subnetwork. For each species S , the set of proteins included in the nodes of the subgraph defines the subnetwork that is induced on S . The node memberships define the sequence-similarity relationships between the sets of proteins of the different species.

A Probabilistic Model of Protein Subnetworks

In order to detect structured subnetworks, we score subgraphs of the alignment graph, which corresponds to collections of conserved subnetworks. Our score is based on a likelihood ratio model for the fit of a single subnetwork to the given structure. The log likelihood ratios are summed over all species to produce the score of the collection. In the following we describe the likelihood ratio model.

Let G be the interaction graph of a given species on a set of proteins P . Note that G is a simple undirected graph. Suppose at first that we have perfect interaction data, i.e., each edge in the interaction graph represents a true interaction and each non-edge represents a true non-interaction. To score the fit of a subgraph to a predefined structure we formulate a log likelihood ratio model that is additive over the edges and non-edges of G , such that high-scoring subgraphs correspond to likely structured subnetworks. Such a model requires specifying a null model and a protein subnetwork model for subgraphs of G . Our models extend those in ref. [11] to account for any target structure; in the discussion below we concentrate on monotone graph properties, that is, graph properties for which: if a graph satisfies it then it continues to satisfy it after adding any set of edges to it.

Let s be a target monotone graph property (e.g., being a clique), let $P' \subseteq P$ be a subset of the proteins, and let H be a labeled graph on P' that satisfies s . We define the two models as follows: the subnetwork model, M_s , corresponding to the target graph H , assumes that every two proteins that are connected in H are also connected in G with some high probability β . In contrast, the null model, M_n , assumes that each edge is present with the probability that one would expect if the edges of G were randomly distributed, preserving the degrees of the vertices. More precisely, we let F^G be the family of all graphs having the same vertex set as G and the same degree sequence (i.e., the sequence of vertex degrees), and define the probability of observing the edge (u, v) to be the fraction of graphs in F^G that include this edge. Note that in this way, edges incident on vertices with higher degrees have higher probability. We estimate these probabilities using a Monte Carlo approach, as described in ref. [11].

Next, we refine the above models to the realistic case in which we are given partial, noisy observations of the true interaction data. In this case our probabilistic model must distinguish between observed interactions and true interactions. For ease of presentation we concentrate on the case that the target structure is a clique (corresponding to a protein complex), but the models generalize to other structures as well. Let us denote by T_{uv} the event that two proteins u, v interact, and by F_{uv} the event that they do not interact. Denote by O_{uv} the (possibly empty) set of available observations on the proteins u and v , that is, the set of experiments in which an interaction between u and v was, or was not, observed. Given a subset U of the vertices, we wish to compute the likelihood of U under a subnetwork

model and under a null model. Denote by O_U the collection of all observations on vertex pairs in U . Under the assumption that all pairwise interactions are independent we have:

$$\begin{aligned} Pr(O_U|M_s) &= \prod_{(u,v) \in U \times U} Pr(O_{uv}|M_c) \\ &= \prod_{(u,v) \in U \times U} [Pr(O_{uv}|T_{uv}, M_c)Pr(T_{uv}|M_c) + Pr(O_{uv}|F_{uv}, M_c)Pr(F_{uv}|M_c)] \\ &= \prod_{(u,v) \in U \times U} [\beta Pr(O_{uv}|T_{uv}) + (1 - \beta)Pr(O_{uv}|F_{uv})] \end{aligned}$$

To compute $Pr(O_U|M_n)$ we must update our null model, which depends on knowing the degree sequence of the (hidden) interaction graph. We overcome this difficulty by approximating the degree of each vertex i by its expected degree, d_i . Our refined null model assumes that G is drawn uniformly at random from the collection of all graphs whose degree sequence is d_1, \dots, d_n . This induces a probability p_{uv} for every vertex pair (u, v) . Thus, we have:

$$Pr(O_U|M_n) = \prod_{(u,v) \in U \times U} [p_{uv}Pr(O_{uv}|T_{uv}) + (1 - p_{uv})Pr(O_{uv}|F_{uv})]$$

Finally, the log likelihood ratio that we assign to a subset of vertices U is

$$L(U) = \log \frac{Pr(O_U|M_c)}{Pr(O_U|M_n)} = \sum_{(u,v) \in U \times U} \log \frac{\beta Pr(O_{uv}|T_{uv}) + (1 - \beta)Pr(O_{uv}|F_{uv})}{p_{uv}Pr(O_{uv}|T_{uv}) + (1 - p_{uv})Pr(O_{uv}|F_{uv})}.$$

Searching for Conserved Subnetworks

Using the above model for comparative interaction data, the problem of identifying conserved protein subnetworks reduces to the problem of identifying high-scoring subgraphs of the network alignment graph. This problem is computationally hard [11]; thus, we present a heuristic strategy for the search problem.

We perform a bottom-up search for high-scoring subgraphs in the alignment graph. The highest-scoring paths with four nodes are identified using an exhaustive search. For dense subgraphs, we start from high-scoring seeds, refine them, and then expand them using local search. Similar approaches based on local search were shown to work well in analyzing high-throughput genomic data [11, 12].

In the first phase of the search we compute a seed around each node v in the alignment graph using two seeding methods. The first method greedily adds p other nodes ($p = 3$), one at a time, such that the added node maximally increases the score of the current seed. Next, we enumerate all subsets of the seed of size at least 3 that contain v . Each such subset serves as a refined seed. The second seeding method computes the highest-scoring path of four nodes that includes v , and these four nodes serve as a refined seed.

In the second phase of the search we apply a local search heuristic on each refined seed. During the local search we iteratively add a node, whose contribution to the score of the current seed is maximum, or remove a node, whose contribution to the current seed is minimum (and negative), as long as this operation increases the overall score of the seed. Throughout the process we preserve the original seed and do not delete nodes from it. For practical considerations, we limit the size of the discovered subgraphs to 15 nodes. For each node in the alignment graph we record up to four highest-scoring subgraphs that were discovered around that node.

As a final stage, we use a greedy algorithm to filter subgraphs with a high degree of overlap. Precisely, we define two subgraphs as highly overlapping if one of the following two conditions is satisfied: (i) their node intersection size over the union size is $> 80\%$; or (ii) for each species separately, the intersection over the union, computed on the subset of proteins from that species that take part in at least one of the two subgraphs, is $> 80\%$. The algorithm iteratively finds the highest scoring subgraph, adds it to the final output list, and removes all other highly overlapping subgraphs.

Statistical Evaluation of Subnetworks

In order to evaluate the statistical significance of the identified subnetworks we compute a P value that is based on the distribution of top scores obtained by applying our method to randomized data. The randomized data are produced by: (i) random shuffling of each of the input interaction graphs, preserving the degrees of the vertices; and (ii) randomizing the sequence-similarity relationships between the different proteins, preserving the number of putative orthologs for each protein. For each randomized dataset, we use our search method to find the highest-scoring subnetworks of a given size. We then estimate the P value of a suggested subnetwork of the same size, as the fraction of random runs which resulted in a subnetwork with a greater score. We retain only subnetworks at a 0.01 significance level. Overview figures of all the identified conserved regions within the three networks are given in Figs. 3 and 6-8. Layouts of representative conserved clusters and paths are given in Figs. 2, 9 and 10.

Scoring Functional Enrichment

Protein paths and clusters were associated with known biological functions using the Gene Ontology annotations (GO; May 2004 version) [13]. Because the GO terms are not independent but connected by an ontology of parent-child relationships, we computed the enrichment of each term conditioned on the enrichment of its parent terms as follows. Define a protein to be below a GO term t , if it is assigned t or any other term that is a descendant of t in

the GO hierarchy. For each path or cluster (specifying a set of proteins) and candidate GO term we recorded the following quantities: (i) The number of proteins in the subnetwork that are below the GO term; (ii) the total number of proteins below the GO term; (iii) the number of proteins in the subnetwork that are below all parents of the GO term; and (iv) the total number of proteins below all parents of the GO term. Given these quantities, we compute a P value of significance using a hypergeometric test. The P value is further Bonferroni corrected for multiple testing. All terms assigned to at least one protein in the set are evaluated.

Prediction of Protein Functions

We used the inferred paths and clusters for predicting novel protein functions. A conserved cluster or path in which many proteins are of the same known function suggests that the remaining proteins in the subnetwork will also have this function. Based on this concept, we predicted new protein functions whenever the following four conditions were satisfied: (i) the set of proteins in a conserved cluster or path (combined across all species) was significantly enriched for a particular GO annotation ($P < 0.01$); (ii) at least five of the proteins in the subnetwork had this significant annotation; (iii) these proteins accounted for at least half of the annotated proteins in the subnetwork overall; and (iv) the annotation was sufficiently specific (at GO level four or higher). For every species, all remaining proteins in the subnetwork were then predicted to have the enriched GO annotation, provided that at least one protein from that species had the enriched annotation.

This process resulted in 4,669 predictions of new GO Biological Process annotations spanning 1,442 distinct proteins in yeast, worm and fly; and 3,221 predictions of new GO Molecular Function annotations covering 1,120 proteins across the three species. We tested the accuracy of our predictions using the technique of cross-validation: we partitioned the set of known protein annotations into 10 parts of equal size. We then iterated over those parts, where at each iteration we hid the annotations that were included in the current part, and used the remaining annotations to predict the held-out annotations. For each protein we predicted at most one function— that with the lowest P value. The prediction was considered correct if the protein had some true annotation that lies on a path in the gene ontology tree from the root to a leaf that visits the predicted annotation. As shown in Tables 3 and 4, depending on the networks and species being compared, 33-63% of our predictions were correct. In particular, our predictions of GO Biological Processes using the three-way clusters and paths achieved success rates of 58% for yeast, 59% for worm and 63% for fly.

We further compared the performance of our function prediction procedure to a simpler prediction process, in which a protein with one or more known functions predicts that its

best sequence match in another species has at least one of those functions. For each pair of species yeast/worm, yeast/fly and worm/fly, we used proteins in the first species to predict the function of their best BLAST matches in the second species. The success rates achieved by this annotation procedure were 36.5%, 40% and 53%, respectively. Even though the annotation using best BLAST matches predicted multiple functions per protein, only one of which had to match a true annotation, the results achieved in the process were comparable to those achieved using the pairwise alignment graphs and inferior to those achieved with the three-way alignment (see Table 3). This comparison demonstrates the superiority of an approach that takes into account the interaction data, and allows the pairing of proteins that are not necessarily each other’s best BLAST matches.

Prediction of Protein Interactions

We also used the alignment graph and the computed subnetworks to predict protein interactions. We experimented with several ways of predicting interactions. The simplest criterion that we tested is to predict an interaction between two proteins whenever there were two nodes in the alignment graph that contained them, such that for at least l of the species, the two respective proteins included in those nodes had distance at most 2 within that species’ interaction graph. We tried both $l = 1$ and $l = 2$ and tested our predictions by using 5-fold cross-validation.

We defined the training interaction data for the cross-validation experiments as follows: we considered the n highest-scoring interactions in each species as positive examples, and the n lowest-scoring interactions as negative examples. To avoid bias toward interactions within dense network regions due to their high clustering coefficient, we recomputed the reliabilities of the protein interactions excluding the clustering coefficient from the model. We removed from the training data interactions that were used for estimating the interaction probabilities; we also removed protein pairs that were not included in the alignment graph being analyzed. At each iteration of the cross-validation experiments we hid one fifth of the interactions (both positives and negatives) and used the remaining data for prediction. Because the yeast and fly networks were considerably richer we used $n = 1,500$ for these two species and $n = 500$ for worm.

We applied this strategy to the three-way alignment graph and to the three pairwise graphs. For yeast, $l = 2$ gave the highest success rates (percents of correct predictions) in the cross-validation; for worm and fly, $l = 1$ yielded the highest success rates. Denote by TP , FP , TN , and FN the numbers of true positives, false positives, true negatives and false negatives, respectively. The sensitivity of the predictions, which is defined as $TP/(TP + FN)$, varied between 19 – 50%; the specificity of the predictions, which is defined as $TN/(TN + FP)$, varied between 78 – 94%. In addition, we also computed the hypergeometric P value for the

results, defined as the probability of choosing at random (without replacement) $(TP + FP)$ balls from an urn with $(TP + FN)$ balls that are labeled positive and $(TN + FP)$ balls that are labeled negative, so that at least TP balls are positive. In all cases our prediction accuracy was highly significant. The results of the cross validation experiments are summarized in Table 5.

Next, we tested the utility of using information on inferred clusters and paths in improving the accuracy of the predictions. By adding the requirement that the two proteins in a predicted interaction are included in an inferred cluster or path, we eliminated virtually all the false positives, although at the price of greatly reducing the percents of true positives. The performance of this inference strategy for the three-way alignment graph is summarized in Table 5.

Based on the high specificity achieved in the cross-validation experiments, we applied our approach to predict novel protein-protein interactions using the more stringent criteria described above. Overall, we predicted 176 interactions for yeast, 1139 for worm and 1294 for fly.

Automatic Layout of Conserved Clusters

We developed a plug-in for cytoscape [14] to automatically lay out collections of conserved clusters for visual inspection. An ideal layout has two properties: (i) within a given cluster, nodes do not overlap; and (ii) nodes that are connected by an edge are located in close proximity. Laying out several conserved clusters imposes an additional constraint, namely, proteins that are similar in sequence should be located in analogous positions in their respective clusters. The first two constraints are well addressed by existing graph layout strategies. One such strategy is Kamada and Kawai’s layout algorithm [15]. In this scheme, each edge is modeled as a spring which exerts a force attracting its endpoint nodes. In addition, all nodes exert a repulsive force to discourage overlap. Given this framework, an ideal layout is one with the lowest possible energy as determined by the forces exerted in the system. In order to satisfy the additional constraint imposed by the conserved clusters, we modify this basic scheme. First, edges are added between all pairs of sequence-similar proteins. Then, the repulsive forces between nodes in distinct clusters are eliminated. After applying the force directed layout, each cluster is overlaid with sequence-similar proteins in similar locations. These individual clusters are then separated to yield side-by-side layouts of conserved clusters.

Two-Hybrid Tests

We tested protein interaction predictions using the two-hybrid protocol described by Uetz *et al.* [16] and Cagney *et al.* [17]. In short, full-length open reading frames of yeast genes were expressed as Gal4 DNA-binding domain fusions (“baits”, vector: pOBD2, strain: PJ69-4alpha) and Gal4 transcriptional activation domain fusions (“preys”, vector: pOAD, strain: PJ69-4a), respectively. These strains have been described by James *et al.* [18]. Haploid yeast strains with bait and prey plasmids were mated; diploids selected on media lacking Leucine and Tryptophan; and two-hybrid positives selected for 10 days on media lacking Leucine, Tryptophan, and Histidine supplemented with 3 mM 3-Amino-Triazole.

References

- [1] Bader, J., Chaudhuri, A., Rothberg, J., & Chant, J. (2004) *Nat. Biotechnol.* **22**, 78–85.
- [2] Deng, M., Sun, F., & Chen, T. (2003) *Proc. PSB* **8**, 140–151.
- [3] von Mering, C., Krause, R., Snel, B., Cornell, M., Oliver, S., Fields, S., & Bork, P. (2002) *Nature* **417**, 399–403.
- [4] Goldberg, D. & Roth, F. (2003) *Proc. Natl. Acad. Sci. USA* **100**, 4372–4376.
- [5] Grigoriev, A. (2001) *Nucleic Acids Res.* **29**, 3513–3519.
- [6] Ge, H., Liu, Z., Church, G., & Vidal, M. (2001) *Nat. Genet.* **29**, 482–486.
- [7] Gollub, J., Ball, C., Binkley, G., Demeter, J., Finkelstein, D., Hebert, J., Hernandez-Boussard, T., Jin, H., Kaloper, M., Matese, J., et al. (2003) *Nucleic Acids Res.* **31**, 94–6.
- [8] Spellman, P. & Rubin, G. (2002) *J. Biol.* **1**, 5.
- [9] Harris, T., Chen, N., Cunningham, F., Tello-Ruiz, M., Antoshechkin, I., Bastiani, C., Bieri, T., Blasiar, D., Bradnam, K., Chan, J., et al. (2004) *Nucleic Acids Res.* **32**, D411–D417.
- [10] Mewes, H., Amid, C., Arnold, R., Frishman, D., Guldener, U., Mannhaupt, G., Munsterkotter, M., Pagel, P., Strack, N., Stumpflen, V., et al. (2004) *Nucleic Acids Res.* **32**, D41–D44.
- [11] Sharan, R., Ideker, T., Kelley, B., Shamir, R., & Karp, R. (2004) *Proc. RECOMB* **8**, 282–289.

- [12] Tanay, A, Sharan, R, Kupiec, M, & Shamir, R. (2004) *Proc. Natl. Acad. Sci. USA* **101**, 2981–2986.
- [13] The Gene Ontology Consortium (2000) *Nat. Genet.* **25**, 25–29.
- [14] Ideker, T., Ozier, O., Schwikowski, B., & Siegel, A. (2002) *Bioinformatics* **18**, **Suppl. 1**, S233–S240.
- [15] Kamada, T. & Kawai, S. (1989) *Information Proc. Lett.* **31**, 7–15.
- [16] Uetz, P., Giot, L., Cagney, G., Mansfield, T., Judson, R., Knight, J., Lockshon, D., Narayan, V., Srinivasan, M., Pochart, P., et al. (2000) *Nature* **403**, 623–627.
- [17] Cagney, G., Uetz, P., & Fields, S. (2000) *Methods Enzymol.* **328**, 3–14.
- [18] James, P., Halladay, J., & Craig, E. (1996) *Genetics* **144**, 1425–1436.