GLOBAL ALIGNMENT OF MULTIPLE PROTEIN INTERACTION NETWORKS

ROHIT SINGH[a] JINBO XU[b] BONNIE BERGER[a*]

[a] *Computer Science and Artificial Intelligence Laboratory*
Massachusetts Institute of Technology
[b] *Toyota Technological Institute, Chicago*
E-mail: {*rsingh@mit.edu, j3xu@tti-c.org, bab@mit.edu*}

We describe an algorithm for global alignment of multiple protein-protein interaction (PPI) networks, the goal being to maximize the overall match across the input networks. The intuition behind our algorithm is that a protein in one PPI network is a good match for a protein in another network if the former's neighbors are good matches for the latter's neighbors. We encode this intuition by constructing an eigenvalue problem for every pair of input networks and then using k-partite matching to extract the final global alignment across all the species. We compute the first known global alignment of PPI networks from five species: yeast, fly, worm, mouse and human. The global alignment immediately suggests functional orthologs across these species; we believe these are the first set of functional orthologs that cover all the five species. We show that these functional orthologs compare favorably with current sequence-only orthology prediction approaches, including better prediction of orthologs for some human disease-related proteins.
Supplementary Information: `http://groups.csail.mit.edu/cb/mna`

1. Introduction

Over the past few years, the use of high-throughput experimental techniques[15,12] for discovering protein-protein interactions (PPIs) has led to a tremendous increase in the corpus of available PPI data in various species. A useful representation of this data is as a network: each node in such a network corresponds to a protein and an edge between two nodes indicates that the corresponding proteins interact. Analysis of such PPI networks has yielded some deep biological insights[9]. In this paper, we explore methods for comparing PPI networks across species. Such comparative analysis has proven to be a valuable tool. It has led, for example, to the identification of conserved functional components across various species, complementing traditional sequence-only phylogenetic analysis. It also helps in identifying errors in experimental PPI data and in transferring annotation across species.

[*] Corresponding author. Also in the MIT Dept. of Mathematics.

We also explore the use of such comparative analysis in improving orthology predictions across species. Identifying cross-species gene correspondences (orthologs) is a problem of fundamental biological importance— it is crucial for transferring insights and information across species.

Contributions

One of the main contributions of this paper is the first algorithm for global alignment of multiple protein interaction networks. We perform a global alignment of PPI networks from five species: yeast, fly, worm, mouse, and human. We pursue the following intuition: a node in a PPI network is a good match for a node in another network if its neighbors are good matches for the neighbors of the other node. To formalize the intuition, we construct a set of eigenvalue problems in an approach similar to Google's PageRank[18] algorithm and then use k-partite matching to compute the final alignment.

The multiple network alignment directly leads to the first comprehensive estimates of functional orthologs that incorporate both sequence and PPI data and cover all the five species mentioned previously. These estimates are more comprehensive than the two most commonly used orthology sets: Homologene[5] and Inparanoid[16]. Our list covers more genes than Homologene. Unlike Inparanoid, which considers pairs of species at a time, our method analyzes data from all input species simultaneously.

We also introduce a novel approach, *functional coherence*, for evaluating orthology predictions. Currently such predictions are evaluated by manually analyzing selected sets of orthologs. In contrast, our automated approach measures the functional similarity within each set of orthologous proteins and computes an aggregate score. Using it, we demonstrate that our algorithm makes predictions with slightly better overall quality than Homologene and Inparanoid. Also, further analysis indicates that some of the improved predictions from our method include disease-related proteins.

Related Work

PPI Network Alignment: The protein network alignment problem can be formulated either as a global or a local network alignment problem. Much of the previous work[3,11,9] in the field has focused on the problem of local network alignment (see Sec. 2). In contrast, we focus on the global alignment problem. Recently, we have proposed the first algorithm for *pairwise* global alignment of PPI networks. The multiple network alignment algorithm we present in this paper is, we believe, the first algorithm for global alignment of multiple protein networks. While this paper builds upon some of the methods presented in our previous work, there are also many significant differences between the two problems and the corresponding algorithms (see Sec. 2).

Functional Ortholog Prediction: Currently, orthology prediction is usually done by using sequence-similarity information between various genes to estimate sets of genes that have descended from a common ancestor. A key challenge here is to distinguish between orthologs and paralogs, the latter being genes that are created by duplication after the two species have diverged. We briefly describe here two commonly used orthology prediction methods: Inparanoid and Homologene (see Chen *et al.*[6] for more). Inparanoid[16] computes orthologs between pairs of species by making explicit assumptions about the relative sequence similarity scores between orthologs and paralogs. One of its drawbacks is that it is limited to pairwise orthology estimates, i.e., it cannot analyze data from multiple species simultaneously. Homologene[5] is an approach that can simultaneously compute orthologs across multiple species by using sequence-similarity scores to construct a tree of proteins and, based upon certain heuristics, grouping them into clusters of orthologous genes.

Recently, efforts have been made to integrate PPI data into the orthology prediction process, to identify sets of proteins that perform the same function. Bandyopadhyay *et al.* [2] have described the use of local network alignment results in identifying *functional orthologs* between yeast and fly. In previous work, we have described a two-way global alignment algorithm which directly suggests functional orthologs between yeast and fly; these predictions compare favorably with Bandyopadhyay *et al.*'s. This paper is the first, we believe, to present functional orthologs across *multiple* species. By integrating data from multiple species simultaneously, we should be able to improve upon predictions made from pairs of species.

2. Problem Formulation

The input to our algorithm consists of two or more protein interaction networks (one per species). Each input network can be represented as an undirected graph $G = (V, E)$ where V is the set of nodes and E is the set of edges. Furthermore, a confidence measure $w(e)$ $(0 < w(e) \leq 1)$ may be associated with each edge e in E. Additionally, the input may also consist of pairwise node similarity measures between nodes from the different networks. In this paper, these similarity measures are BLAST Bit-values, but other scores (e.g., synteny-based scores[10]) can also be used. Given these inputs, our goal is to find the best overall match (i.e, optimal global alignment) between the input networks. This will directly lead to a list of functional orthologs.

Local vs. Global Network Alignment: Network alignment problems vary in the scope of the input (two vs. multiple networks), and the kind

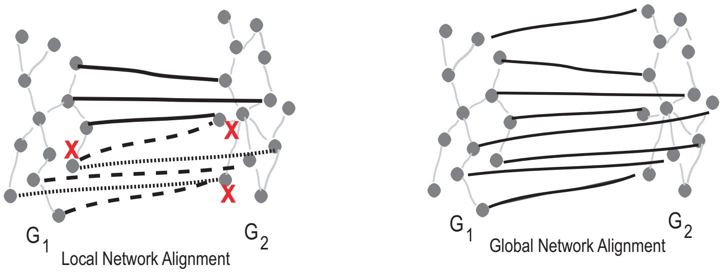


Figure 1: **Cartoon comparing global and local network alignments**: The local network alignment between G_1 and G_2 specifies three different alignments; the mappings for each are marked by a different kind of line (solid, dashed, dotted). Each alignment describes a small common subgraph. Local alignments need not be consistent in their mapping– the points marked with 'X' each have ambiguous/inconsistent mappings under different alignments. In global network alignment, the maximum common subgraph is desired. In both cases, there are 'gap' nodes for which no mappings could be predicted (here, the nodes with no incident black edges are such nodes).

of node-mapping desired. In general, the goal in all such problems is to identify one or more mappings between the nodes of the input networks and, for each mapping, the corresponding set of conserved edges. A mapping may be partial, i.e., it need not be defined for all the nodes in the networks. Each mapping implies a common subgraph between the input networks: when protein a_1 from network G_1 is mapped to proteins a_2 from G_2 and a_3 from G_3, then a_1, a_2, and a_3 refer to the same node in the common subgraph; the edges in the common subgraph correspond to the conserved edges. A key difference between our approach and many previous network alignment approaches is in the kind of mapping desired.

Much of the previous work[3,11,9] has focused on local network alignment (LNA), i.e., on finding local regions of isomorphism (i.e., same graph structure) between the input networks. Each such region implies a mapping independently of others. Many independent, high-scoring local alignments are usually possible between two input networks; in fact, the corresponding local alignments need not even be mutually consistent (i.e., a protein might be mapped differently under each alignment; see Fig. 1).

In contrast, we focus on the global network alignment (GNA) problem. The aim in GNA is to find the best overall alignment between the input networks. The mapping in a GNA should cover all the input nodes: each node in an input network is either matched to one or more nodes in other network(s) or explicitly marked as a gap node (i.e., with no match in another network). In contrast, a LNA algorithm outputs multiple, independent mappings, each corresponding to a local region of similarity. Furthermore, these partial mappings may be mutually inconsistent. The mapping corresponding to a GNA is also required to be transitive: if a_1 in G_1 is mapped to a_2 in G_2 and a_2 is mapped to nodes a_3, a'_3 in G_3, then a_1 should also be mapped to a_3, a'_3. Our goal in GNA then is to find a comprehensive mapping between the nodes of the input networks such that the size of the single corresponding common subgraph is maximized. Our previous work[17] contains a more detailed comparison of the LNA and GNA problems.

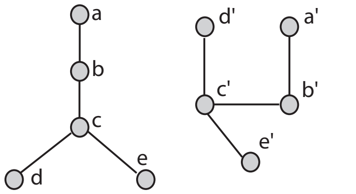
A key difference between the multiple-network GNA (the focus of this paper) and pairwise GNA (the focus of our previous work[17]) is in the scope of the mapping desired. In the latter, we required that a node may be mapped to at most one node in the other network, the motivation being to find the *best* match for a node. In contrast, for the multiple networks case we allow for a node to map to multiple nodes in another network. This is necessary because gene duplication, mutation, and deletion events might make it impossible to find a valid one-to-one, transitive mapping between proteins across an arbitrary collection of species.

3. Algorithm

To describe a global alignment between input networks, we need to specify a node mapping between the input networks and the corresponding common subgraph. We focus on the computing the node mapping, since the subgraph can be easily computed once the former is known.

Our algorithm works in two stages. First, given k input networks, we create a k-partite graph $\mathcal{H}$. Each of its k parts contains nodes from one of the input networks. Edges are only allowed between nodes from different parts. The presence of an edge e_{ij} implies that node i (from G_1) can potentially be mapped to j (from G_2); the edge-weight R_{ij} indicates the strength of the potential match. In the second stage, we perform k-partite matching on $\mathcal{H}$ to group nodes into clusters. All nodes in a cluster are then mapped to each other in the corresponding GNA.

First Stage (Creating the k-partite graph): We start with the k input PPI networks and sequence similarity scores between the nodes. For every pair of input networks, we compute a score for every possible pairing between the nodes of the two networks. Let R_{ij} ($R_{ij} \geq 0$) be the score for the protein pair (i, j) where i is from network G_1 and j is from network G_2. Intuitively, R_{ij} should capture how good a match i and j are: higher R_{ij} implies a better match. In the second stage, we will use these scores to guide our algorithm towards the optimal k-partite matching of $\mathcal{H}$.



$$\begin{array}{c|ccccc} R & a' & b' & c' & d' & e' \\ \hline a & 0.0312 & & 0.0937 & & \\ b & & 0.1250 & & 0.0625 & 0.0625 \\ c & 0.0937 & & 0.2812 & & \\ d & & 0.0625 & & 0.0312 & 0.0312 \\ e & & 0.0625 & & 0.0312 & 0.0312 \end{array}$$

$$R_{aa'} = \frac{1}{4} R_{bb'} \qquad R_{bb'} = \frac{1}{3} R_{ac'} + \frac{1}{3} R_{a'c} + R_{aa'} + \frac{1}{9} R_{cc'} \qquad R_{dd'} = \frac{1}{9} R_{cc'}$$

$$R_{cc'} = \frac{1}{4} R_{bb'} + \frac{1}{2} R_{be'} + \frac{1}{2} R_{bd'} + \frac{1}{2} R_{eb'} + \frac{1}{2} R_{db'} + R_{ee'} + R_{ed'} + R_{de'} + R_{dd'}$$

Figure 2: **Intuition behind the First Stage of the algorithm**: Here we show, for a pair of small, isomorphic graphs how the vector of pairwise scores (R) is computed (see Eqn. 1). Only a partial set of constraints is shown here. Here we show the vector of scores R reshaped as a table, for ease of viewing (empty cells indicate a value of zero). Observe that high values of R (e.g., $R_{cc'}$ or $R_{bb'}$) correctly indicate that the respective pairings represent good matches.

To compute R (the vector of all R_{ij}s for G_1 and G_2) we construct an eigenvalue problem. First consider the case when no sequence similarity scores are available (i.e., R_{ij} depends only on G_1 and G_2's topologies). We require that R_{ij}s satisfy the following system of constraints (for all i, j):

$$R_{ij} = \sum_{u \in N(i)} \sum_{v \in N(j)} \frac{1}{|N(u)||N(v)|} R_{uv} \quad i \in V_1,\ j \in V_2 \tag{1}$$

where $N(a)$ is the set of neighbors of node a, $|N(a)|$ is the size of this set, and V_1 and V_2 are the sets of nodes in G_1 and G_2, respectively. These constraints can be re-written as an eigenvalue equation:

$$\begin{gathered} R = AR \\ A[i,j][u,v] = \frac{1}{|N(u)||N(v)|} \end{gathered} \tag{2}$$

where A is a $|V_1||V_2| \times |V_1||V_2|$ matrix and $A[i,j][u,v]$ refers to the entry at the row (i,j) and column (u,v) (the row and column are doubly-indexed). The value of R we are interested in is the principal eigenvector of A. Typically, A is a very large matrix (about $10^8 \times 10^8$ for fly-vs.-yeast GNA). However, A is a stochastic matrix[14] and both A and R are very sparse, so R can be efficiently computed by iterative techniques, like the *power method* [14].

The intuition behind these equations is that they require that the score R_{ij} for any match (i,j) be equal to the total support provided to it by each of the $|N(i)||N(j)|$ possible matches between the neighbors of i and j. In return, each match (u,v) must distribute back its score R_{uv} equally among the $|N(u)||N(v)|$ possible matches between its neighbors (see Fig. 2 for an example). We note that these equations also capture non-local influences on the score R_{ij}: it depends on the score of neighbors of i and j and the latter, in turn, depend on the neighbors of the neighbors and so on. Also, these equations can be extended to the weighted-graph case very naturally[17].

It is straightforward to incorporate sequence similarity information, e.g. BLAST scores, into this model. Let B_{ij} denote the score between i and j; for instance, B_{ij} can be the Bit-Score of the BLAST alignment between sequences i and j. Let B be the vector of B_{ij}s. We first compute E, the normalized version of B: $E = B/|B|$. The eigenvalue equation is then modified to (this equation can also be solved by iterative techniques):

$$R = \alpha A R + (1-\alpha)E \quad \text{where} \quad 0 \leq \alpha \leq 1 \tag{3}$$

Changing α lets us control the weight of the network data (relative to sequence data) in this computation. For example, $\alpha = 0$ implies no network data will be used, while $\alpha = 1$ indicates only network data will be used.

Second Stage (K-partite matching): We construct the k-partite graph $\mathcal{H}$ as follows: for any pair of nodes i and j from different PPI networks, we add an edge e_{ij} to $\mathcal{H}$ if $R_{ij} > 0$, and set the edge-weight to R_{ij}. We now find a k-partite matching of $\mathcal{H}$ (recall that each part corresponds to nodes from one PPI network). The matching must be transitive, i.e, if i is matched to j and j is matched to l, then i must be matched to l. Furthermore, we aim to match nodes connected by high-scoring edges. More precisely, our goal is to find the maximum-weight k-partite matching of $\mathcal{H}$ where each set of matched nodes may contain upto r nodes from each of the k parts. Here, r is a user-defined parameter ($r \geq 1$). Allowing a one-to-many mapping lets us express that, for example, a particular fly protein has no corresponding yeast protein but two corresponding human proteins. In our previous work on two-way network alignment, this flexibility was not present.

The standard k-partite matching problem formulation requires that a node can match at most one node in each of the other $k-1$ parts. Our formulation thus generalizes this problem (the standard version corresponds to $r = 1$). However, the classical problem is already known to be NP-Hard[13], so our formulation is NP-Hard as well. Thus, it is unlikely that an exact solution for it can be found efficiently. Here, we present an approach that computes the matching by identifying a seed match and extending it:

- While the k-partite graph $\mathcal{H}$ has any edges left:
  (1) Select the edge (i, j) with the highest score (let i be from G_1 and j from G_2). Initialize a new match-set with i and j as its initial members.

(2) In every other species $(G_3, \ldots, G_k)$, if a node l exists such that (A) R_{il} and R_{jl} are the highest scores between l and any node in G_1 and G_2, respectively and, (B) the scores $R_{ik} \geq \beta_1 R_{ij}$ and $R_{jk} \geq \beta_1 R_{ij}$, add it to the set. These set of nodes form the primary match-set; it has at most one node from each species.

(3) Add upto $r - 1$ nodes from different parts of the graph to the primary match-set. Suppose u (from G_x) is in the primary match-set. Then, a node v (from G_x) is added to the set if $R_{vw} \geq \beta_2 R_{uw}$ for each node w $(w \neq u)$ in the primary set.

(4) Remove from $\mathcal{H}$ all the nodes in this match-set and their edges.

Here, the parameters r, β_1, β_2 are user-defined $(0 < \beta_2, \beta_1 < 1)$; we chose their values such that the functional coherence (see Sec. 4.1) of the resulting sets of matched nodes was maximized.

Given a mapping between the nodes of the input networks, the corresponding common subgraph in the GNA can be identified relatively easily. For example, if a_1 is aligned to a_2, and b_1 is aligned to b_2, the output subgraph should contain an edge between the corresponding nodes if and only if both the input networks contain supporting edges.

4. Results

Datasets: We constructed PPI networks for five species: *S. cerevisiae*, *D. melanogaster*, *C. elegans*, *M. musculus*, and *H. sapiens.* These networks were constructed by combining data retrieved from the DIP[8], BioGRID[4] and HPRD[7] databases. The relative coverage of the PPI data varied heavily; the number of edges per species were: 36387 (human), 31899 (yeast), 25831 (fly), 4573 (worm), and 255 (mouse). Sequence data for the various proteins was retrieved from Ensembl and the BLAST Bit-values were used as the score of sequence similarity between input proteins. Even in species with relatively high PPI coverage (e.g., yeast), there were many proteins that did not occur in the PPI network. To ensure that these proteins were included in the functional ortholog lists, we added singleton (disconnected) nodes corresponding to each such protein in the respective PPI networks, thus using only sequence data.

Global Alignment of Yeast, Fly, Worm, Human and Mouse networks: When performing the alignment, we chose the following parameter settings: $\alpha = 0.6, r = 5, \beta_1 = 0.1, \beta_2 = 0.1$. These settings correspond to the node mapping with the best functional coherence (see Sec. 4.1).

We analyzed the common subgraph implied by the multiple alignment. The common subgraph has 1663 edges that are supported by edges in at least two PPI networks and 157 edges that are supported by atleast three networks. There are very few edges with support from four or more species; however, this is not surprising since the worm and mouse networks are very small. The size of the common subgraph is relatively small (only about 5% of human PPI network). One reason for the small overlap between the PPI networks, we believe, is that the current PPI data is both incomplete and noisy. As the quality and quantity of data improves, this overlap should increase further. Even with this incomplete data, we believe that the currently computed (partial) set of node-pairings is robust. In previous work[17], we have performed experiments which suggest that the eigenvalue formulation is robust to errors in PPI data, especially when sequence data is provided.

A naive approach to multiple network alignment would use current sequence-based orthology predictions to perform the mapping; however, by incorporating both sequence and network data, our algorithm performs much better. The common subgraph implied by Homologene's sequence-only mapping contains only 509 edges with support in two or more species and 40 edges with support in three or more species. Thus, the addition of network topology in computing the mappings increases the size of the common subgraph by over three-fold (from 509 to 1663). A direct comparison can not be performed against Inparanoid orthology lists because the Inparanoid's pairwise orthology lists can not be used for multiple network alignment. Instead, we evaluated the total number of conserved edges implied by Inparanoid in 10 $(= \binom{5}{2})$ pairwise network alignments. Even though this final number, 1172, likely over-counts some conserved edges, it is significantly less than the number of conserved edges implied by our algorithm.

The common subgraph in the global alignment consists of multiple components, many of which are significantly larger than those from local alignment methods. Also, unlike the latter, these subgraphs correspond to a variety of topologies: linear, complex-like, tree-shaped, etc. Some of them are also enriched in proteins involved in a specific function (see Supp. Info. for details).

4.1. Functional Coherence: Evaluating Orthology Predictions

We propose a method for scoring the quality of an ortholog list (i.e., a list which specifies sets of orthologous proteins across two or more species). The method is motivated by the lack of automated, direct measures of quality of orthology lists. Currently, the most common strategy for comparing two orthology lists is to identify pairs of proteins which are grouped differently under the two lists and perform a manual, case-by-case analysis of some pairs. Because of the manual approach, a comprehensive evaluation can be time-consuming. Recently, Chen *et al.*[6] have described a computational approach where they compare many ortholog lists to identify the list(s) with the best overall agreement with the remaining ones. However, this approach does not measure if the orthology predictions are biologically plausible.

We aim to find a direct, automated measure of ortholog quality by using functional information. The intuition behind our method is simple: given an ortholog list, we select the sets of orthologs that have many proteins with known function. For each set, we collect all the Gene Ontology[1] (GO) terms corresponding to proteins in it. We evaluate if the set is functionally coherent, i.e., if the GO terms describe similar functions. Finally, an aggregate score (across all sets) is computed. Higher scores imply higher coherence, indicating that the ortholog list groups proteins with known function together. In Supp. Info., we describe the algorithm more precisely.

In Supp. Info., we describe some experiments which demonstrate that the functional coherence scoring scheme does capture the desired biological intuition. This scoring scheme allows us to measure how similar the functions of proteins mapped to the same ortholog set are. One potential problem with this approach is that there might not be enough proteins for which GO terms are available to compute such scores. However, for both Homologene and our functional ortholog predictions, there are over 1500 sets of orthologous proteins such that functional information is available for at least 80% of the proteins in the set. We believe that this degree of coverage is sufficient to generate statistically reliable estimates of functional coherence. In Supp. Info., we also describe in greater detail these sets of orthologs: their sizes, group-wise coherence score etc.

Functional Orthologs from Multiple Network Alignment: In this paper, we present the first known set of functional orthologs (FO) across five species: yeast, fly, worm, mouse and human. The FO mapping is simply the node mapping computed by our algorithm (see Supp. Info. for the list of FOs). Of the 86932 proteins from the five species, 59539 (68.5%) of the proteins in our list were matched to atleast one protein in another species (i.e., had at least one FO). In contrast, Homologene has lower coverage, predicting at least one ortholog for only 33434 (38.5%) proteins. Also, the functional coherence of our predicted functional orthologs is comparable with that of Homologene and Inparanoid predictions. The functional coherence scores are: 0.220 (our predictions), 0.223 (Homologene) , and 0.206 (mean score across Inparanoid's pairwise ortholog sets). Homologene's slightly better score may partly be due to its use of data from many species (more than 5). Rather than relying excessively on sequence-score based heuristics, our method uses functional information (from PPI networks) to predict FOs— these scores suggest that our approach is a simpler and better way of capturing functional similarities between proteins. At the same time, our predicted FOs do not deviate drastically from sequence-only predictions: 66% of protein-pairs grouped together by Inparanoid are also grouped together by our approach.

Our predicted FOs have certain limitations. Our approach relies on PPI data to identify functionally related proteins. For many proteins, however, no PPI data is available. In such cases, the algorithm's ability to identify functionally-related sets of proteins may suffer. However, the expected increase in the availability of PPI data should help overcome this limitation.

Case-study: Functional Orthologs of two Human Disease-related Proteins: A key application of this work is in a more comprehensive prediction in of orthologs of human disease-related genes in model organisms. An accurate understanding of which genes in, say, fly are relevant in human diseases would be of significant value in directing scientific work. The human gene DHRC7 has been linked to the Smith-Lemli-Opitz syndrome. Homologene predicts only a mouse homolog for this gene. Our algorithm predicted B0250.9 (from worm), dLBR (from fly) and YNL280C (from yeast) as orthologs. Each of these proteins has been observed to perform a function similar to that of the human gene (sterol reductase). Similarly, B3GN3 is a human gene observed to be differentially expressed in colon cancer. Homologene fails to find a fly homolog of this gene; our algorithm predicts the fly gene *brn* as its homolog. This prediction is supported by the fact that both the proteins are galactosyltransferases.

Another application of the proposed algorithm is to predict a comprehensive human PPI network by combining PPI data from other species. Analysis of the connections of disease-related proteins in this large network may offer improved insights about the disease mechanisms and possible drug targets.

5. Conclusion

In this paper, we focus on the global network alignment problem and present an algorithm for computing the global alignment of multiple protein interaction networks. The algorithm is simple, yet powerful— it provides users the ability to control the relative weights of the sequence and network data in the alignment. Using the algorithm we compute the first known global alignment of PPI networks from five species: yeast, fly, worm, mouse and human. The results provide valuable insights into the conserved functional components across the various species. They also enable us to predict functional orthologs between these five species; the quality of these functional orthologs compares favorably with current sequence-only functional orthologs. Our algorithm also has some parallels with Google's PageRank algorithm, specifically in the construction of eigenvalue problem(s) (see Supp. Info.).

In future work, we intend to more deeply explore the differences and similarities between our predicted functional orthologs and currently used ortholog lists. We also intend to improve the algorithm by exploring better algorithms for k-partite matching. Finally, we plan to explore the application of this algorithm to other biological and non-biological network data.

References

1. *http://www.geneontology.org.*
2. S. Bandyopadhyay, R. Sharan, and T. Ideker. Systematic identification of functional orthologs. *Genome Research*, 16(3):428–435, 2006.
3. B. P. Kelley *et al.* PathBLAST: a tool for alignment of protein interaction networks. *Nucleic Acids Research*, 32:W83–8, 2004.
4. C. Stark *et al.* BioGRID: a general repository for interaction datasets. *Nucleic Acids Research*, 34:D535–9, 2006.
5. D. L. Wheeler *et al.* Database resources of the National Center for Biotechnology Information. *Nucleic acids research*, 35:D5–12, 2007.
6. F. Chen *et al.* Assessing performance of orthology detection strategies applied to eukaryotic genomes. *PLoS ONE*, 2:e383, 2007.
7. G. R. Mishra *et al.* Human Protein Reference Database– 2006 update. *Nucleic Acids Research*, 34:D411–4, 2006.
8. I. Xenarios *et al.* DIP, the Database of Interacting Proteins. *Nucleic Acids Research*, 30(1):303–305, 2002.
9. J. Flannick *et al.* Graemlin: general and robust alignment of multiple large interaction networks. *Genome Research*, 16(9):1169–1181, 2006.
10. M. Kellis *et al.* Methods in comparative genomics. *Journal of Computational Biology*, 11(2-3):319–355, 2004.
11. M. Koyuturk *et al.* Pairwise alignment of protein interaction networks. *Journal of Computational Biology*, 13(2):182–199, 2006.
12. P. Uetz *et al.* From ORFeomes to protein interaction maps in viruses. *Genome Research*, 14(10B):2029–2033, 2004.
13. M. R. Garey and D. S. Johnson. Computers and Intractability. *Freeman*, 1979.
14. G. H. Golub and C. Van Loan. Matrix Computations. *J.H.U. Press*, 2006.
15. T. Ito, T. Chiba, and M. Yoshida. Exploring the protein interactome. *Trends in Biotechnology*, 19(10):S23–7, 2001.
16. M. Remm, C. E. Storm, and E. L. Sonnhammer. Automatic clustering of orthologs and in-paralogs. *J of Mol Bio*, 314(5):1041–1052, 2001.
17. R. Singh, J. Xu, and B. Berger. Pairwise Alignment of Protein Interaction Networks. *Proc. of Conf on Res in Comp Mol Bio, RECOMB*, 2007.
18. L. Page *et al.* PageRank Citation System: Bringing Order to the Web. *Tech Report. Stanford Dig Lib Proj*, 1998.