SPINAL: scalable protein interaction network alignment

Ahmet E. Aladag and Cesim Erten

1Department of Computer Engineering, Bogazici University, Bebek, Istanbul 34342 and 2Department of Computer Engineering, Kadir Has University, Cibali, Istanbul 34083 Turkey

ABSTRACT

Motivation: Given protein–protein interaction (PPI) networks of a pair of species, a pairwise global alignment corresponds to a one-to-one mapping between their proteins. Based on the presumption that such a mapping provides pairs of functionally orthologous proteins accurately, the results of the alignment may then be used in comparative systems biology problems such as function prediction/verification or construction of evolutionary relationships.

Results: We show that the problem is NP-hard even for the case where the pair of networks are simply paths. We next provide a polynomial time heuristic algorithm, SPINAL, which consists of two main phases. In the first coarse-grained alignment phase, we construct all pairwise initial similarity scores based on pairwise local neighborhood matchings. Using the produced similarity scores, the fine-grained alignment phase produces the final one-to-one mapping by iteratively growing a locally improved solution subset. Both phases make use of the construction of neighborhood bipartite graphs and the contributors as a common primitive. We assess the performance of our algorithm on the PPI networks of yeast, fly, human and worm. We show that based on the accuracy measures used in relevant work, our method outperforms the state-of-the-art algorithms. Furthermore, our algorithm does not suffer from scalability issues, as such accurate results are achieved in reasonable running times as compared with the benchmark algorithms.

Availability: Supplementary Document, open source codes, useful scripts, all the experimental data and the results are freely available at http://code.google.com/p/spinal/.

Contact: cesim@khas.edu.tr

Supplementary information: Supplementary data are available at Bioinformatics online.

Received on July 9, 2012; revised on November 16, 2012; accepted on February 7, 2013

1 INTRODUCTION

Several high-throughput techniques including the yeast two-hybrid system (Finley and Brent, 1994), co-immunoprecipitation coupled mass spectrometry ( Aebersold and Mann, 2003) and computational methods such as those based on genome-wide analysis of gene fusion, metabolic reconstruction and gene co-expression (Goh and Cohen, 2002) enable extraction of large-scale protein–protein interaction (PPI) networks of various species. Several problem formulations related to network topologies (Han et al., 2004), module detections (Bader and Hogue, 2002) and evolutionary patterns (Hunter et al., 2002) have been proposed for the analysis of these networks. From a comparative interactomics perspective, network alignment problems constitute yet another important family of problem formulations for the analysis of PPI networks.

In general terms, given two or more PPI networks from different species, where for each network, nodes represent the proteins and the edges represent the interactions between the proteins, the network alignment problem is to align the nodes of the networks or subnetworks within them. Functional orthology is an important application that serves as the main motivation to study the alignment problems as part of a comparative analysis of PPI networks; a successful alignment could provide a basis for deciding the proteins that have similar functions across species. Such information may further be used in predicting functions of proteins with unknown functions or in verifying those with known functions (Dutkowski and Tiuryn, 2007; Singh et al., 2008), in detecting common orthologous pathways between species (Kelley et al., 2003) or in reconstructing the evolutionary dynamics of various species (Kuchaiev and Pržulj, 2011). Before the introduction of network alignment as a model, common methods to detect orthologous groups of proteins have been solely based on measures of evolutionary relationships, usually in the form of sequence similarities. HomoloGene and Inparanoid (Remm et al., 2001) are examples of such approaches. Network alignment algorithms on the other hand incorporate the interaction data as well as the evolutionary relationships represented possibly in the form of sequence data. Based on the assumption that the interactions among functionally orthologous proteins should be conserved across species, such an incorporation is usually achieved by aligning proteins so that both the sequence similarities of aligned proteins and the number of conserved interactions are large.

Two versions of this general alignment framework have been suggested. In local network alignment, the goal is to identify from the input PPI networks, subnetworks that closely match in terms of network topology and/or sequence similarities. Approaches proposed for this version of the problem include PathBLAST (Kelley et al., 2004), NetworkBLAST (Sharan et al., 2005), MaWISh (Koyutürk et al., 2006), Graemlin (Flannick et al., 2006) and the graph match-and-split algorithm of Narayanan and Karp (2007). Typically many overlapping subnetworks from a single PPI network are provided as part of the local alignments; this gives rise to ambiguity, as a protein may be matched with many proteins from a target PPI network. In global network alignment on the other hand, the goal is to align the networks as a whole, providing unambiguous one-to-one mappings between the proteins of different networks.

*To whom correspondence should be addressed.
Starting with IsoRank (Singh et al., 2008), several global network algorithms using more or less similar definitions have been suggested. IsoRank is based on an eigenvalue formulation of local neighborhood alignments. PATH and GA of Zaslavskiy et al. (2009) are based on appropriate relaxations of a cost formulation over the set of doubly stochastic matrices. PISwap uses a greedy heuristic based on iterative swaps of mappings until local optimum (Chindelevitch et al., 2010). MI-GRAAL (Kuchaiev and Pržulj, 2011) and variants (Kuchaiev et al., 2010; Memišević and Pržulj, 2012; Milenković et al., 2010) use greedy heuristics based on cost formulations including one or more of the graphlet degree signatures, degrees, clustering coefficients, eccentricities and the sequence similarities in terms of BLAST E-values. Other related network alignment problems include global many-to-many alignments (Ay et al., 2011; Liao et al., 2009) and queries in interaction networks and pathways (Banks et al., 2008; Dost et al., 2008; Pinter et al., 2005; Shlomi et al., 2006).

A major issue in network alignment is the computational intractability of all the appropriate optimization formulations. It becomes even more apparent with some input PPI networks containing tens of thousands of nodes and interactions. An important feature expected of the global network alignments is then scalability; the running time performances of the suggested methods should not degrade drastically with increasing network sizes. At the same time, accurate alignment scores close to optimum values of appropriate formulations is a natural expectation. However, existing approaches either aggressively optimize for better accuracy at the expense of scalability or vice versa. We propose a novel global network alignment algorithm, SPINAL, which consists of two phases: a coarse-grained alignment score estimations phase and a fine-grained conflict resolution and improvement phase. Both phases make use of the construction of neighborhood bipartite graphs and a set of contributors as a common primitive. Using these concepts within iterative local improvement heuristics constitute the backbone of the algorithm. In terms of scalability, SPINAL runs much faster and provides more accurate results than the compared state-of-the-art methods in almost all of the experimented instances under consideration.

2 METHODS AND ALGORITHMS

2.1 Problem definition

Let \( G_1 = (V_1, E_1) \) and \( G_2 = (V_2, E_2) \) be two PPI networks where \( V_1, V_2 \) denote the sets of nodes corresponding to the proteins and \( E_1, E_2 \) denote the sets of edges corresponding to the interactions between proteins. We define an alignment network \( A_{12} = (V_{12}, E_{12}) \). Each node of \( V_{12} \) is denoted with a pair \( <u_i, v_j> \), where \( u_i \in V_1 \) and \( v_j \in V_2 \). For any pair of nodes \( <u_i, v_j> \in V_{12} \) and \( <u_i', v_j'> \in V_{12} \), it should be the case that \( u_i \neq u_i' \) and \( v_j \neq v_j' \). The edge set of the alignment network is defined so that any conserved interaction gives rise to an edge in the network, that is, for \( <u_i, v_j> \in V_{12} \) and \( <u_i', v_j'> \in V_{12} \), the edge \( (u_i, u_i') \) in \( E_{12} \) if and only if \( (u_i, u_i') \in E_1 \) and \( (v_j, v_j') \in E_2 \).

Although an explicit definition of an alignment network is not given, informally the common goal in most of the previous global PPI network alignment approaches is to provide an alignment so that the edge set \( E_{12} \) is large and each pair of node mappings in the set \( V_{12} \) contains proteins with high sequence similarity (Chindelevitch et al., 2010; Kuchaiev and Pržulj, 2011; Singh et al., 2008; Zaslavskiy et al., 2009). Formally, we define the pairwise global PPI network alignment problem as that of finding the alignment network \( A_{12} = (V_{12}, E_{12}) \) that maximizes the global network alignment score, defined as follows:

\[
GNAS(A_{12}) = \alpha \times |E_{12}| + (1 - \alpha) \times \sum_{v \in V_{12}} seq(u_i, v_j)
\]

The constant \( \alpha \in [0, 1] \) in this equation is a balancing parameter intended to vary the relative importance of the network-topological similarity (conserved interactions) and the sequence similarities reflected in the second term of the sum. Each \( seq(u_i, v_j) \) can be an appropriately defined sequence similarity score based on measures such as BLAST bit-scores or E-values.

2.2 The SPINAL global alignment algorithm

For the special case of \( \alpha = 1 \), the pairwise global PPI network alignment problem becomes a generalized version of the Maximum Common Edge Subgraph (MCES) problem used commonly in the matchings of 2D/3D chemical structures (Raymond and Willett, 2002). The MCES of two undirected graphs \( G_1, G_2 \) is a common subgraph (not necessarily induced) that contains the largest number of edges common to both \( G_1 \) and \( G_2 \). The NP-hardness of the MCES problem (Garey and Johnson, 1979) trivially implies that the defined network alignment problem is also NP-hard. Although useful in certain aspects, such a result does not provide sufficient intuition to grasp the nature of the problem, which involves simultaneous optimization of two possibly conflicting properties. In addition, PPI networks usually exhibit certain topological properties that may affect the computational complexity of an optimization problem defined on them. Nevertheless, we show that the problem with its simultaneous nature is computationally intractable even for two paths. This result holds for all \( \alpha \) values other than 0 and 1. The full proof of the following theorem can be found in the Supplementary Document.

**Theorem 2.1.** The pairwise global PPI network alignment problem is NP-hard for a pair of paths.

The intrinsic computational hardness of the problem gives rise to the design of local heuristic approaches rather than globally optimum solutions. Most of the global network alignment algorithms can be viewed to proceed in two phases. For each pair \( u_i \in V_1, v_j \in V_2 \), an estimate confidence score is sought at an initial coarse-grained phase. The score represents the level of confidence that the match \( (u_i, v_j) \) is in the optimum alignment maximizing the global score defined in Equation (1). This is usually followed by a fine-grained phase that consists of refining an initial global alignment based on the estimate scores attained in the previous phase. Similar in spirit to the previous global PPI network alignment algorithms, SPINAL also proceeds in two phases. However, the definition and the construction method of the confidence scores matrix in the coarse-grained phase, and the refinement method in the fine-grained phase constitute
the novelties of our algorithm. We first introduce the construction of neighborhood bipartite graph and the computation of its maximum weight matching, both of which together constitute the common primitive operation used in both phases. Let $S$ be a function mapping every pair of vertices $u_i \in V_1, v_j \in V_2$ to a real valued weight. Denote the set of neighbors of $u_i$ in $G_1$ with $N(u_i)$ and the set of neighbors of $v_j$ in $G_2$ with $N(v_j)$. The neighborhood bipartite graph of the pair $<u_i, v_j>$ on $S$, denoted with $N^B_G(<u_i, v_j>, S)$ is a complete edge-weighted bipartite graph defined on the partitions $N(u_i)$ and $N(v_j)$. The weight of an edge $(x_i, y_j)$ in $N_B G$ is $S(x_i, y_j)$. Similarly, we define $N^B_G$ of a set of pairs rather than that of a single pair, as the union of the $N^B_G$s of the constituent pairs.

Algorithm 1 SPINAL global alignment algorithm

1: **Input:** $G_1 = (V_1, E_1), G_2 = (V_2, E_2), \text{seq} \alpha$
2: **Output:** Node set $V_{12}$ of the global alignment network $A_{12}$
3: // Coarse-grained
4: for all $u_i \in V_1, v_j \in V_2$ do
5: $P(u_i, v_j) = \alpha \times \text{DegDiff}(u_i, v_j) + (1 - \alpha) \times \text{seq}(u_i, v_j)$
6: end for
7: repeat
8: $P = P$
9: for all $u_i \in V_1, v_j \in V_2$ do
10: construct $N^B_G(<u_i, v_j>, \{P\})$
11: construct contributors set $C$ of $N^B_G$
12: compute $P(u_i, v_j)$ as in Equation (2)
13: end for
14: until enough iterations
15: // Fine-grained
16: $SP = \text{List of } <u_i, v_j> \text{ sorted w.r.t} \ P, \text{ for } u_i \in V_1, v_j \in V_2$
17: repeat
18: // Find new connected component in $A_{12}$
19: pop unaligned $<u_i, v_j>$ from $SP$, insert into $V_{12}$
20: repeat
21: construct $N^B_G(V_{12}, P)$
22: construct contributors set $C$ of $N^B_G$
23: swap improvements for each $N^B_G$ edge not in $C$
24: insert $<x_i, y_j>$ into $V_{12}$, for each $(x_i, y_j) \in C$
25: until no contributors
26: until no unaligned pair in $SP$

2.2.2 Fine-grained conflict resolution and improvement Once the scores matrix $P$ is ready, the next step is to extract a one-to-one mapping of node pairs in a way that the resulting mapping induces a high score in terms of Equation (1). We follow a seed-and-extend approach coupled with local improvements based on iterative swaps. We note that both these techniques are standard heuristics in combinatorial optimization and different versions have also been used in previous alignment algorithms (Altschul et al., 1990; Chindelevitch et al., 2010; Kuchaiev et al., 2010; Kuchaiev and Pržulj, 2011; Shi and Parthasarathy, 2011).

The $N^B_G$ and the contributors’ concepts, which constituted the basis of the coarse-grained phase are the main primitives of this phase as well. The pseudocode is provided in lines 16–26 of Algorithm 1. The basic idea is to find a connected component of the alignment network $A_{12}$ at each iteration of the outer repeat loop. Each component starts with the best available seed. It is the pair $(u_i, v_j)$ with the largest score in $P$, such that neither $u_i$ nor $v_j$ is aligned. The component grows layer by layer in an almost breadth-first manner. At each iteration of the inner repeat loop, a new breadth-first layer of $G_{12}$ is added to the current component of $A_{12}$. For this, we first construct the $N^B_G$ of the set of the aligned pairs in the current component, which is the union of $N^B_G$s of each pair. Assuming the weight of each edge is its estimate confidence score in $P$, a maximum-weight matching of $N^B_G$ provides a set of candidate contributors to be added to the current component of the alignment graph. Because the scores in $P$ are solely estimate scores of confidence, even an optimum maximum-weight matching may have room for improvement as far as the GNAS score of Equation (1) is concerned. Therefore, our final step is to improve the candidate set locally.

The $N^B_G$ and the contributors’ concepts, which constituted the basis of the coarse-grained phase are the main primitives of this phase as well. The pseudocode is provided in lines 16–26 of Algorithm 1. The basic idea is to find a connected component of the alignment network $A_{12}$ at each iteration of the outer repeat loop. Each component starts with the best available seed. It is the pair $(u_i, v_j)$ with the largest score in $P$, such that neither $u_i$ nor $v_j$ is aligned. The component grows layer by layer in an almost breadth-first manner. At each iteration of the inner repeat loop, a new breadth-first layer of $G_{12}$ is added to the current component of $A_{12}$. For this, we first construct the $N^B_G$ of the set of the aligned pairs in the current component, which is the union of $N^B_G$s of each pair. Assuming the weight of each edge is its estimate confidence score in $P$, a maximum-weight matching of $N^B_G$ provides a set of candidate contributors to be added to the current component of the alignment graph. Because the scores in $P$ are solely estimate scores of confidence, even an optimum maximum-weight matching may have room for improvement as far as the GNAS score of Equation (1) is concerned. Therefore, our final step is to improve the candidate set locally.
via possible swaps. Each pair in \(N^G\) but not among the candidates is compared against its overlap set, that is, the set of candidate contributors sharing a node with it. If the contribution of the new pair to the \(G^A\) score is not smaller than that of its overlap set, it is inserted into \(A_{12}\) rather than the overlap set.

In terms of running time requirements, in almost all the tests, >95% of the execution time is spent by the initial coarse-grained phase. We note that in the actual implementation, the contributors set in the first phase is computed via a greedy maximal matching algorithm, whereas for the second phase, an optimum solution is used. Details of the SPINAL algorithm, including implementation details, a discussion of stability and running time analysis, are provided in the Supplementary Document.

3 DISCUSSION OF RESULTS

SPINAL is implemented in C++ using LEDA (Mehlhorn and Naher, 1999). Source code, useful Python scripts for testing and evaluations, all the data and output results are available as part of the Supplementary Material. We experiment on data from four species:Saccharomyces cerevisiae, Drosophila melanogaster, Caenorhabditis elegans and Homo sapiens. All the data are from IsoBase (Park et al., 2011), which is the same as that used in IsoRank and IsoRankN. The PPI network sizes are as follows: 5499 proteins and 31 261 interactions in the \(S\).cerevisiae network, 7518 proteins and 25 635 interactions in the \(D\).melanogaster network, 2805 proteins and 4495 interactions in the \(C\).elegans network and 9633 proteins and 34 327 interactions in the \(H\).sapiens network. Potentially, SPINAL can be compared with other alignment algorithms with a similar problem definition formalized by Equation (1). These are IsoRank, MI-GRAAL and variants, GA, PATH heuristics and the PISwap algorithm. We extensively compare SPINAL with IsoRank and MI-GRAAL. IsoRank is a popular benchmark algorithm in global network alignment. Recently suggested MI-GRAAL, to the best of our knowledge, provided the best alignments in terms of the number of conserved interactions previously. The current implementations of GA and PATH are not amenable for the alignment of networks with sizes similar to those under consideration (Kuchaiev and Pržulj, 2011). For lack of a publicly available implementation of PISwap, only brief comparisons with the published results are made whenever applicable.

3.1 Global network alignment score evaluations

We first measure the extent of accuracies of the algorithms in terms of the maximization objective formulated in Equation (1). The number of conserved interactions, that is, the edge set size of the alignment network, denoted with \(E_{12}\) in the equation is a common performance indicator used in almost all the global network alignment studies (Chindelevitch et al., 2010; Klau, 2009; Kuchaiev et al., 2010; Kuchaiev and Pržulj, 2011; Milenković et al., 2010; Singh et al., 2008; Zaslavskiy et al., 2009). Because the optimization goal is also commonly defined as in Equation (1), we include the score obtained from \(G^A_{12}\) as well as \(|E_{12}|\) in our evaluations of an alignment \(A_{12}\). Table 1 summarizes our findings for the SPINAL, IsoRank and MI-GRAAL algorithms. For each of the six dataset pairs, we include two rows: top row indicates the size of conserved interactions set \(E_{12}\) and the bottom row indicates the score obtained from \(G_{45}(A_{12})\). Each column represents the scores of an alignment output by a specific algorithm under a specific setting of input parameters. Parameter settings for SPINAL and IsoRank consist of varying the \(\alpha\) constant from 0.3 to 0.7 in the increments of 0.1. As for the MI-GRAAL algorithm, three alignment versions are described in the original description (Kuchaiev and Pržulj, 2011). The Alignment3 version refers to an output alignment obtained when signatures, degrees, clustering coefficients and BLAST sequence similarities are all used by the algorithm. It is mentioned that the largest set of conserved interactions are obtained under Alignment3 and that its results are the most stable, in the sense that different runs provide almost the same results (Kuchaiev and Pržulj, 2011). Therefore, we present evaluations of this version for MI-GRAAL. For each row measuring the size of conserved interactions set, the largest score is marked in bold. The number of conserved interactions attained by the SPINAL alignments is impressive. The state-of-the-art algorithm known to achieve the largest conservation scores was MI-GRAAL. Table 1 indicates that in five of the six alignment pairs, SPINAL provides the highest score in terms of \(E_{12}\) sizes. Only for the \(C\).elegans–\(D\).melanogaster pair, MI-GRAAL provides better edge conservation. The \(G_{45}(A_{12})\) scores for the MI-GRAAL alignments are computed under the setting of \(\alpha = 0.7\). For the instances where MI-GRAAL columns are marked with a \(X\), Alignment3 could not be successfully executed until completion. We were able to execute Alignment1 version using signatures for the hs-sc instance. Interaction conservation and the \(G_{45}\) scores of a single run were, respectively, 5277 and 3963.95. Regarding scores of conserved interactions, our final result is on published results of PISwap using the data of Bandyopadhyay et al. (2006). On the same dataset, SPINAL produces an alignment with 3890 conserved interactions for the \(D\).melanogaster–\(S\).cerevisiae pair, whereas the PISwap alignment achieves 398 interactions. Emphasizing the issue of scalability, we provide a sample comparison of execution times. The pair of largest and densest networks for which all three methods provide alignments is \(H\).sapiens–\(S\).cerevisiae. The execution times of SPINAL, IsoRank and MI-GRAAL (The Alignment3 version of MI-GRAAL that uses graphlet degree signatures is used. Nevertheless, Alignment3 version, which could not be executed until completion on this dataset is expected to require an even larger execution time because it uses three additional cost functions.) on this dataset are, respectively, 49, 116 and 305 min. The contrast between SPINAL and MI-GRAAL is especially significant, as previously the latter was known to provide the highest conserved interaction ratios. SPINAL runs almost five times faster than MI-GRAAL and provides almost 10% more conserved interactions. We note that the running time experiments were performed on a 64-bit machine with Intel Core i5 2.27 GHz processors and 4 GB of memory.

3.2 Gene ontology consistency evaluations

A common measure to test the biological quality of alignments is based on gene ontology (GO) consistency of the aligned pairs of proteins. For an alignment \(A_{12}\), we define \(GOC(A_{12})\) as the sum of \(|GO(u_i) \cap GO(v_j)|/|GO(u_i) \cup GO(v_j)|\), over all aligned pairs...
the other hand, it may not be possible to produce alignments that make sense, as a large portion of all pairs contributes little to the PPI network alignment problem formalized by Equation (1), this corresponding algorithm provided in the columns.

Table 1. GNAS evaluations

<table>
<thead>
<tr>
<th>Dataset</th>
<th>SPINAL</th>
<th>IsoRank</th>
<th>MI-GRAAL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\alpha = 0.3$</td>
<td>$\alpha = 0.4$</td>
<td>$\alpha = 0.5$</td>
</tr>
<tr>
<td>ce-dm</td>
<td>2343</td>
<td>2320</td>
<td>2300</td>
</tr>
<tr>
<td>ce-hs</td>
<td>717.99</td>
<td>941.19</td>
<td>1159.93</td>
</tr>
<tr>
<td>ce-sc</td>
<td>2370</td>
<td>2446</td>
<td>2437</td>
</tr>
<tr>
<td>dm-sc</td>
<td>2384</td>
<td>2320</td>
<td>2300</td>
</tr>
<tr>
<td>hs-sc</td>
<td>709.12</td>
<td>962.87</td>
<td>1168.95</td>
</tr>
</tbody>
</table>

| $< u_i, v_j > \in F_{12}$. Here, GO($x$) denotes the set of GO terms annotating a protein $x$. We exclude the annotations to the root terms, Biological Process, Cellular Component and Molecular Function. The GO annotations are retrieved from the GO Consortium (Ashburner et al., 2000).

The results presented in Table 1 are valuable in providing an idea on the extent of conserved interactions achieved by different algorithms. However the same strategy of comparisons based on fixed $\alpha$ values can not be directly used in GOC evaluations of IsoRank and SPINAL, although both algorithms use the same global optimization function. This is mainly due to the variance in total sequence similarity scores achieved by resulting alignments even for the same $\alpha$ instances. Because many GO annotations are based on sequence alignments themselves, such comparisons would produce misleading results. This discrepancy has been observed and handled in different ways in previous studies (Kuchaiev and Przulj, 2011; Zaslavskiy et al., 2009).

We follow both approaches and compute GOC scores accordingly.

The main idea of Zaslavskiy et al. (2009) is to compare the alignments achieved under fixed total sequence similarity scores when possible. The SPINAL algorithm, especially in the fine-grained phase in Algorithm 1, aggressively aims at increasing the size of $E_{12}$ to achieve higher scores for $\text{GNAS}(A_{12})$. For the PPI network alignment problem formalized by Equation (1), this makes sense, as a large portion of all pairs contributes little to the alignment score through their sequence similarity scores. On the other hand, it may not be possible to produce alignments with some specific total sequence similarity values, especially the large ones. Therefore, we introduce another version of our algorithm, SPINAL$_f$, that only makes use of the coarse-grained phase of Algorithm 1 and similar to IsoRank simply applies a maximum weight bipartite matching for the fine-grained phase. This provides an opportunity to evaluate SPINAL and IsoRank better, as the coarse-grained phases of both algorithms are defined to solve exactly the same problem. The results for all six pairs of PPI networks are presented in Table 2. The IsoRank [IsoRank provides two separate alignments. To provide a fair comparison, the GO consistency evaluations of Table 2 are those obtained from the IsoRank$_{k_{125}}$ version, the alignment that is mentioned to provide better GO consistencies (Singh et al., 2008)] results in the table correspond to the alignments under the shown $\alpha$ values ranging from 0.3 to 0.7 in the increments of 0.1. On the other hand, for a fixed $\alpha$, each SPINAL$_f$ result corresponds to the alignment that achieves as close a total sequence similarity score as possible, to that of the IsoRank alignment under $\alpha$. In almost all cases, the difference in the corresponding total sequence similarity scores is <0.1; hence, the gathered alignments are comparable. Among all 30 alignment instances, SPINAL$_f$ provides better results than IsoRank, except for three instances. The differences between the GOC scores become more apparent as the network sizes get larger. Also, in terms of the number of conserved interactions, for all pairwise alignments and $\alpha$ values, SPINAL$_f$ provides much better results than IsoRank. This is significant because it provides a clue that optimizing the number of conserved interactions under fixed total sequence similarities leads to better functional orthology detection, a conjecture assumed to have limited evidence previously (Zaslavskiy et al., 2009). For comparisons with MI-GRAAL, we use the Alignment3 version of the algorithm, as it makes use of sequence information and is favored over the other alignment types to be the basis of function predictions of unannotated proteins (Kuchaiev and Przulj, 2011). Both the SPINAL and the MI-GRAAL algorithms aggressively aim at improving the number of conserved interactions. For a fair comparison, we can actually pick any alignment of SPINAL that provides better conserved interaction scores than those of the MI-GRAAL Alignment3 results from Table 1. We pick $\alpha = 0.7$ instance of SPINAL, even though in many cases even $\alpha = 0.3$ alignments with better chances of large GOC scores produce better conserved interaction ratios. Nevertheless, SPINAL GO consistency scores are much higher than those of MI-GRAAL in
all pairwise alignments. For the *C. elegans–D. melanogaster* pair, the SPINAL alignment produces a GOC score of 79.57, whereas the score of MI-GRAAL alignment is 14.41. For the *C. elegans–H. sapiens, C. elegans–S. cerevisiae* and the *D. melanogaster–S. cerevisiae* pairs, the scores are 43 versus 15.64, 60.03 versus 24.97 and 113.01 versus 50.51, respectively.

Secondly, to account for the effects of sequence similarities in the GO consistency evaluations, we repeated the same experiments following the approach of Kuchaiev and Pržulj (2011). The idea is to consider only the experimental GO annotations, that is, those with evidence codes IPI, IGI, IMP, IDA, IEP, TAS and IC. Because the resulting relative GOC scores are almost the same, we do not provide separate tables. Among all 30 instances corresponding to the ones presented in Table 2, in only five of them IsoRank provides slightly better GOC scores than SPINAL. For the rest, SPINAL provides higher scores and the differences between achieved scores are relatively large for many of them. Finally, comparing SPINAL and MI-GRAAL, we get the same results as in the previous approach. In all instances, SPINAL provides much higher scores than MI-GRAAL.

We note that because GO category organization is hierarchical and there might be specific categories at levels further away from the root of the GO DAG, expecting exact category overlaps can be a strong requirement for GO consistency evaluations. Therefore, similar to the evaluation method suggested in Singh et al. (2008), we repeated the same tests annotating each protein to a standardized set of GO categories (those at distance 5 from the root of GO DAG) and considering the resulting category overlaps. Furthermore, to test the algorithms on different datasets, we created experiments based on synthetic PPI network data of Sahraeian and Yoon (2012) and evaluated the algorithms using this database and the IsoBase database under several additional metrics including mean normalized entropy, coverage, correct nodes and specificity. In general, the results are along the lines of those presented in this section. Details regarding all these extensive evaluations can be found in the Supplementary Document.

### 3.3 Annotation transfers via network alignment

PPI networks of single species have been studied in depth to predict functions of unannotated proteins or to extract biological pathways; see Sharan et al. (2007) for a survey on the topic. Another way to extract such information has been through a detailed analysis of proteins with sequence similarities (Louie et al., 2009). It is natural to assume that alignment networks of pairwise PPIs should provide analog information because they provide a model to integrate both kinds of data. Accordingly, previous network alignment studies suggest protein function predictions via *annotation transfers*, that is, via assigning the annotations of a protein in an aligned pair to the unannotated member of the same pair (Kuchaiev and Pržulj, 2011; Singh et al., 2008). However, a detailed analysis demonstrates that such automated transfers by themselves may not always be sufficient to provide immediate function predictions. Incorporating the global alignment results into the function prediction methods using network analysis techniques provides more reliable predictions (Sharan and Ideker, 2006). Although a methodological treatment of this issue is beyond the scope of this article, we present a more detailed analysis of the *H. sapiens–S. cerevisiae* alignment network to provide a basis for such an integration. We choose to analyze the SPINAL alignment resulting from the settings used in the $\alpha = 0.3$ column of Table 2. Details regarding this alignment network can be found in the Supplementary Document.

Graph-theoretic approaches to identify key regulatory proteins in an organism by analyzing local PPI network structures have been suggested previously (Fox et al., 2011). Following similar reasoning, we extract neighborhood subgraphs induced by a node and its neighbors in the alignment network to identify key pairs of proteins. Each key pair is considered suitable for a possible annotation transfer. For each $< u_i, v_j >$, we compute a
dominating annotation, dom(<u_i, v_j>) and a domination count, \( dc(<u_i, v_j>) \). Let \( S_{u_i, v_j} \) denote the subgraph induced by \( \{ u_i, v_j \}, \{ u_i, v_j^\prime \} \in E[G] \). We count the number of times each GO annotation appears in any node of \( S_{u_i, v_j} \). Note that an annotation appearing in any of the proteins of a node contributes to the count. The largest count is \( dc(<u_i, v_j>) \) and the corresponding annotation is \( dom(<u_i, v_j>) \). We exclude all GO annotations derived from Cellular Component. To extract a list of hubs in decreasing order of importance, we sort the \( dc \) values of all nodes with two exceptions. If \( <u_i^\prime, v_j^\prime> \in S_{u_i, v_j} \) and \( dc(<u_i^\prime, v_j^\prime>) < dc(<u_i, v_j>) \), then \( <u_i^\prime, v_j^\prime> \) is not included in the list. Additionally, if \( dom(<u_i^\prime, v_j^\prime>) = dom(<u_i, v_j>) \) and \( dc(<u_i^\prime, v_j^\prime>) < dc(<u_i, v_j>) \), then \( <u_i^\prime, v_j^\prime> \) is not in the list.

For this analysis, among the top 10 nodes in the list, we consider those with five or more neighbors that contain three or more GO annotation overlaps. Six such nodes are identified. Going from 1 to 6, the matches corresponding to those nodes and their dominating annotations are, respectively, as follows: TBP/YER148W regulation of transcription, DNA-dependent (GO:0006355), RAN/YLR293C transport (GO:0006810), LOC392454/YBR088C DNA binding (GO:0003677), POLR2A/YDL140C transcription, DNA dependent (GO:0006351), TAF7/YPL011C RNA polymerase II transcriptional preinitiation complex assembly (GO:0051123), MCM2/YBL023C DNA replication (GO:0006260). The domination counts are 17, 15, 14, 13, 10 and 10, respectively. It is worth noting that some of the identified hub matches themselves contain considerably large GO annotation overlaps. The TBP/YER148W match has 5, RAN/YLR293C match has 10, POLR2A/YDL140C match has 9 and MCM2/YBL023C match has 14 overlaps. We expect each protein involved in a match to contain an annotation same as or similar to (descending from a not too distant common ancestor in the GO dag) its dominating annotation. We realize an annotation transfer for an unannotated protein in a match, if its mate in the alignment and a considerable number of its neighbors in its own PPI network are annotated with the dominating annotation.

Both proteins in the TBP/YER148W match are annotated exactly with the dominating annotation. Proteins in the RAN/YLR293C match on the other hand are not annotated with the dominating annotation, GO:0006810, although both are annotated with a similar category, GO:0006886 (intracellular protein transport). Considering the LOC392454/YBR088C match, LOC392454 does not contain any annotations, whereas YBR088C contains the dominating annotation of the match, GO:0003677 (DNA binding). The neighborhood of LOC392454 in the \( H. sapiens \) PPI network contains 81 proteins. Among these, 44 of them are unannotated. On the other hand, only 14 are not annotated with DNA binding or related categories. Twelve neighbors have been annotated with exactly DNA binding and 11 have annotations that are similar (nucleic acid binding, chromatin binding, double-stranded DNA binding, damaged DNA binding). This provides a clue that the match LOC392454/YBR088C has been correctly identified as a regulating hub and LOC392454 should also be annotated with GO:0003677 (DNA binding). Regarding the POLR2A/YDL140C match, we verify that YDL140C is annotated with GO:0006351 (transcription, DNA-dependent). Although POLR2A is not annotated with the same category, it has a similar annotation GO:0006355 (regulation of transcription, DNA-dependent). With regard to the TAF7/YPL011C match, YPL011C is annotated with exactly the dominating annotation. Although it is tempting to transfer the dominating annotation to TAF7, which is unannotated, a careful analysis reveals that among the 20 neighbors of TAF7, only one of them contains the annotation GO:0051123. Twelve do not contain related categories, and the rest are unannotated. This is in accordance with the results of Fox et al. (2011), as the TAF7/YPL011C hub is what Fox et al. (2011) call a single-component hub and can not be counted as a regulating hub. Therefore, we do not apply an annotation transfer in this case. Finally, regarding the MCM2/YBL023C match, it is verified that both proteins are annotated with the dominating annotation.

Funding: TUBITAK, 112E137 (in part).

Conflict of Interest: none declared.

REFERENCES


