

Simultaneous Optimization of Both Node and Edge Conservation in Network Alignment via WAVE

Yihan Sun^{1,2,3}, Joseph Crawford¹, Jie Tang² and Tijana Milenković^{1,*}

¹Department of Computer Science and Engineering, Interdisciplinary Center for Network Science and Applications, and ECK Institute for Global Health, University of Notre Dame, Notre Dame, IN, USA.

²Department of Computer Science and Technology, Tsinghua University, Beijing, China.

³Computer Science Department, Carnegie Mellon University, Pittsburgh, Pennsylvania.

*Corresponding Author (E-mail: tmilenko@nd.edu)

Abstract. Network alignment can be used to transfer functional knowledge between conserved regions of different networks. Existing methods use a node cost function (NCF) to compare nodes across networks and an alignment strategy (AS) to find high-scoring alignments with respect to total NCF over all aligned nodes (or node conservation). Then, they evaluate alignments via a measure that is different than node conservation used to guide alignment construction. Typically, one measures edge conservation, but only after alignments are produced. Hence, we recently directly maximized edge conservation while constructing alignments, which improved their quality. Here, we aim to maximize both node and edge conservation during alignment construction to further improve quality. We design a novel measure of edge conservation that (unlike existing measures that treat each conserved edge the same) weighs conserved edges to favor edges with highly NCF-similar end-nodes. As a result, we introduce a novel AS, **Weighted Alignment VotEr** (WAVE), which can optimize any measures of node and edge conservation. Using WAVE on top of well-established NCFs improves alignments compared to existing methods that optimize only node or edge conservation or treat each conserved edge the same. We evaluate WAVE on biological data, but it is applicable in any domain.

1 Introduction

1.1 Motivation

Network alignment aims to find topologically or functionally similar regions between different networks. It has applications in different areas, including computational biology [1–7], ontology matching [8–11], pattern recognition [12, 13], social networks [14, 15], language processing [16], and others [17–21]. Our study focuses mainly on the computational biology domain.

Protein-protein interaction (PPI) networks have been the main focus of network alignment research among all biological networks. PPI network alignment can be used to transfer biological knowledge from the network of a poorly studied species to the network of a well studied species. This is of importance because not all cellular processes can easily be studied via biological experiments. For example, studying aging in human has to rely on across-species transfer of aging-related knowledge from model species [22]. And network alignment can be (and has been) used for this [6, 7]. However, the problem is computationally intractable, as the underlying subgraph isomorphism problem is NP-complete [23]. Thus, network alignment methods are heuristics.

Network alignment can be local or global. Local network alignment aims to align well local network regions [24–32]. As such, it often fails to find large conserved regions between networks. Hence, majority of recent research has focused on global network alignment [33–36, 3, 1, 2, 4, 5, 37–41, 3, 7, 6, 42, 43], which can find large conserved regions between networks. Typically, global network alignment aims to generate one-to-one node mapping between two networks [42] (although exceptions exist that produce many-to-many node mappings or that align more than two networks [3], but such methods are out of the scope of our study).

Of one-to-one global network alignment methods, many consist of two algorithmic components, namely, a node cost function (NCF) and an alignment strategy (AS) [7, 43]. NCF captures pairwise similarities between nodes in different networks, and AS then searches for good alignments based on the NCF information. It has already been recognized that when two methods of this two-component NCF-AS type are compared, to fairly evaluate the methods, one should mix and match the different methods’ NCFs and ASs, because NCF of one method and AS of another method could lead to a new method that is actually superior to the original methods [7, 43].

We base our work on established state-of-the-art NCFs of existing methods. Then, we propose a novel AS, **Weighted Alignment VotEr** (WAVE), which when used on top of the established NCFs leads to a new superior method for global network alignment. And while we evaluate our new method in the computational biology domain, the method is easily applicable in any domain.

1.2 Related Work

We focus on NCFs of two popular existing methods, MI-GRAAL [4] and GHOST [5], and we aim to improve with our new WAVE AS upon these methods’ ASs.

MI-GRAAL improves upon its predecessors, GRAAL [1] and H-GRAAL [2], by using the same NCF but by combining their ASs (see below). MI-GRAAL’s NCF computes topological similarity between extended network neighborhoods of two nodes [44–47]. It does so by relying on the concept of small induced subgraphs called graphlets (e.g., a triangle or a square) [48, 49], which are used to summarize the topology of up to 4-deep network neighborhood of a node into its graphlet degree vector (GDV) [44, 50, 51]. Then, GDV-similarity is used as MI-GRAAL’s NCF, which compares nodes’ GDVs to compute their topological similarity. MI-GRAAL also allows for integration of other node similarity measures into its NCF, such as protein sequence similarity. We showed [7] that MI-GRAAL’s NCF is superior to another, Google PageRank algorithm-based NCF, which is used by IsoRank [33] and IsoRankN [3]. Regarding AS [43], MI-GRAAL combines GRAAL’s greedy seed-and-extend AS with H-GRAAL’s optimal AS that uses the Hungarian algorithm to solve linear assignment problem of maximizing total NCF over all aligned nodes.

GHOST’s NCF is conceptually similar to MI-GRAAL’s, as it also assumes two nodes from different networks to be similar if their neighborhoods are similar. However, the mathematical and implementation details of the two NCFs are different. Namely, GHOST’s NCF takes into account a node’s k -hop neighborhood, (in this study, $k = 4$). Then, its NCF computes topological distance (or equivalently, similarity) between two nodes by comparing the nodes’ “spectral signatures”. We recently fairly compared MI-GRAAL’s GDV-similarity-based NCF with GHOST’s “spectral signature”-based NCF within our above mix-and-match framework, concluding that MI-GRAAL’s NCF is superior or comparable to GHOST’s NCF, depending on data [43]. Hence, since none of the two NCFs was dominant in all cases, we consider both NCFs in our study. Just as MI-GRAAL, GHOST also allows for integration of protein sequence information into its NCF. Regarding AS, GHOST is also a seed-and-extend algorithm, like MI-GRAAL. However, GHOST’s AS considers the quadratic (instead of linear) assignment problem. When we evaluated the two ASs, their performance was data-dependent [43]. Hence, we consider both ASs in our study.

There exist additional more recent network alignment methods [42], both those that also belong to the category of NCF-AS methods, such as NETAL [37], and those that do not, such as MAGNA [6]. These methods became available close to completion of our study, and as such, we were not able to include them into the design of our new method. (Hence, NETAL implements a different NCF compared to NCFs of MI-GRAAL and GHOST, along with a different AS compared to ASs of MI-GRAAL, GHOST, and WAVE.) However, we still consider these methods in our evaluation. Importantly, our goal is to show that when we use under an existing NCF (such as MI-GRAAL’s or GHOST’s) our new WAVE AS, we get alignments of higher quality compared to when using an existing AS (such as MI-GRAAL’s or GHOST’s) on the same NCF. This would be sufficient to illustrate the superiority of WAVE. If in the process we also improve upon the more recent methods, such as those that use a different NCF and especially those that do not belong to the NCF-AS category, that would further demonstrate WAVE’s superiority.

1.3 Our Contributions and Significance

We introduce WAVE, a novel, general, and as we will show well-performing AS, which can be combined with any NCF. WAVE is applicable to any domain. In this study, we evaluate it on biological networks.

Its novelty and significance is as follows. The existing ASs use NCF scores to rapidly identify from possible alignments the high-scoring alignments with respect to the overall NCF (henceforth also referred to as node conservation). But, their alignment accuracy is then evaluated with some other measure that is different than NCF used to construct the alignments [6]. Typically, one measures the amount of conserved (i.e., aligned) edges. Hence, a recent attempt aimed to directly maximize edge conservation during alignment construction [6]. Here, we aim to optimize both node and edge conservation while constructing an alignment, as also recognized by a recent effort [37]. In the process, unlike the existing methods that treat each conserved edge the same, we aim to favor conserved edges with NCF-similar end nodes over those with NCF-dissimilar end nodes. And we design WAVE with these goals in mind.

We combine WAVE with NCF of MI-GRAAL as well as with NCF of GHOST. We denote the resulting network aligners as M-W and G-W, respectively. We compare M-W and G-W against the original MI-GRAAL (henceforth also denoted by M-M) and GHOST (henceforth also denoted by G-G), which use MI-GRAAL’s NCF and AS and GHOST’s NCF and AS, respectively. Further, we compare M-W and G-W with a new method introduced recently [43], which is the combination of GHOST’s NCF and MI-GRAAL’s AS (henceforth also denoted by G-M). This allows us to test the performance of WAVE against the performance of MI-GRAAL’s and GHOST’s ASs, under each of MI-GRAAL’s and GHOST’s NCF. We note that we cannot compare M-W and G-W against the combination of MI-GRAAL’s NCF and GHOST’s AS (i.e., M-G), as the current implementation of GHOST does not allow for plugging MI-GRAAL’s NCF into GHOST’s AS [43]. Finally, we compare M-W and G-W against the very recent NETAL and MAGNA methods.

We evaluate all methods on synthetic and real-world PPI networks, relying on established data and performance measures [1, 2, 4–7]. We find that WAVE AS is overall superior to the existing ASs, especially in terms of topological alignment quality. Also, WAVE overall performs comparably to or better than NETAL and MAGNA, especially on synthetic data. This further validates WAVE, because NETAL implements a newer and thus possibly more efficient NCF compared to NCFs of M-W or G-W, which might give NETAL unfair advantage over WAVE.

2 Methods

2.1 Data

We evaluate WAVE on two popular network sets [1, 2, 4–7]: 1) “synthetic” networks with known node mapping, and 2) real-world networks with unknown node mapping.

The “synthetic” data consists of a high-confidence yeast PPI network [52] with 1,004 nodes and 8,323 PPIs, and of five noisy networks constructed by adding to the high-confidence network a percentage of low-confidence PPIs from the same data set [52]; we vary the percentage from 5% to 25% in increments of 5%. We align the original high-confidence network to each of the five noisy networks, resulting in five network pairs to be aligned. Since we know the correct node correspondence, we can measure to what extent an aligner correctly reconstructs the correspondence.

The real-world set contains binary (yeast two-hybrid, Y2H) PPI networks of four species: *S. cerevisiae* (yeast/Y), with 3,321 nodes and 8,021 edges, *D. melanogaster* (fly/F), with 7,111 nodes and 23,376 edges, *C. elegans* (worm/W), with 2,582 nodes and 4,322 edges, and *H. sapiens* (human/H), with 6,167 nodes and 15,940 edges. We align each pair of the networks, resulting in six pairs. If we aimed to predict new biological knowledge, we would have evaluated our method on additional PPIs, such as those obtained via affinity purification followed by mass spectrometry (AP/MS). However, since our main focus is method evaluation, of all PPIs, we focus on binary Y2H PPIs because: 1) they have been argued to be of higher quality than literature-curated PPIs supported by a single publication [53, 50], and 2) the same Y2H networks have already been used in many existing studies [1, 2, 4–7]. Ultimately, what is important for a fair evaluation is that all methods are tested on the same data, be it Y2H, AP/MS, or other PPIs [7].

When we combine within NCF nodes’ topological similarity scores with their sequence similarity scores (see below), for the latter, we rely on BLAST bit-values from the NCBI database [54]. When we evaluate biological alignment quality with respect to functional enrichment of the aligned nodes (see below), we rely on Gene Ontology (GO) data [55] to evaluate the biological alignment quality. We use same data versions as in our recent work [7, 43].

2.2 Combining Topological and Sequence Information Within NCF

We compute the linear combination of topological node similarity scores s_t and sequence node similarity scores s_s of nodes u and v as: $s(u, v) = \alpha s_t(u, v) + (1 - \alpha) s_s(u, v)$. We vary α from 0.0 to 1.0 in increments of 0.1. We do this for all combinations of MI-GRAAL’s, GHOST’s, and WAVE’s NCFs and ASs. When we compare WAVE to recent NETAL and MAGNA, since current implementations of NETAL and MAGNA do not support inclusion of sequence information, for these methods, we only study topology-based alignments (corresponding to α of 1).

2.3 Evaluation of Alignment Quality

If we align graph $G(V_G, E_G)$ to graph $H(V_H, E_H)$ (where $|V_G| \leq |V_H|$) via an injective function $f : V_G \rightarrow V_H$, let us denote with E'_G this edge set: $E'_G = \{(f(u), f(v)) | u \in V_G, v \in V_G, (u, v) \in E_G\}$. Also, let us denote with E'_H the edge set of the subgraph of H that is induced on nodes from V_H that are images of nodes from V_G . $E'_H = \{(f(u), f(v)) | u \in V_G, v \in V_G, (f(u), f(v)) \in E_H\}$. With these notations in mind, we next define alignment quality measures that we use.

Topological Alignment Quality Measures

Node correctness (NC). Given a known true node mapping (which is typically not available in real-world applications), NC is the percentage of node pairs that are correctly mapped by an alignment. If $f^* : V_G \rightarrow V_H$ is the correct node mapping of G to H and $f : V_G \rightarrow V_H$ is an alignment produced by the aligner, $NC = \frac{|\{u \in V_G : f^*(u) = f(u)\}|}{|V_G|} \times 100\%$ [1].

Edge Correctness (EC). EC represents the percentage of edges from G , the smaller network (in terms of the number of nodes), which are aligned to edges from H , the larger network [1]. Formally, $EC = \frac{|E'_G \cap E'_H|}{|E_G|} \times 100\%$, where the numerator is the number of conserved edges.

Induced conserved structure (ICS). ICS is defined as $ICS = \frac{|E'_G \cap E'_H|}{|E'_H|} \times 100\%$. It was introduced because EC fails to penalize for misaligning edges in the larger network, i.e., E'_H , as EC is defined with respect to edges in E_G only [5].

Hence, ICS accounts for this. However, ICS now fails to penalize for misaligning edges in the smaller network, i.e., E_G , as it is defined with respect to edges in E'_H only. Hence, the following measure, S^3 , was introduced recently to penalize for misaligning edges in both the smaller and the larger network [6].

Symmetric substructure score (S^3). S^3 is defined as $S^3 = \frac{|E'_G \cap E'_H|}{|E_G| + |E'_H| - |E'_G \cap E'_H|} \times 100\%$ [6]. Thus, S^3 keeps advantages of both EC and ICS while addressing their drawbacks. S^3 was already shown to be the superior of the three measures [6]. Thus, we discard EC and ICS measures from further consideration, and instead, we report results for S^3 .

The size of the *largest connected common subgraph (LCCS)*[1]. In addition to counting aligned edges via S^3 measure, it is important that the aligned edges cluster together to form large, dense, and connected subgraphs, rather than being isolated. In this context, a connected common subgraph (CCS) is defined as a connected subgraph (not necessarily induced) that appears in both networks [2]. We measure the size of the largest CCS (LCCS) in terms of the number of nodes as well as edges, as defined in the MAGNA paper [6].

In summary, we focus on NC, S^3 , and LCCS. The larger their values, the better the topological alignment quality.

Biological Alignment Quality Measures To transfer function from well annotated network regions to poorly unannotated ones, which is the main motivation behind network alignment in computational biology, alignment should be of good biological quality, mapping nodes that perform similar function.

Gene Ontology Enrichment (GO). One could measure GO, the percentage of aligned protein pairs in which the two proteins *share* at least one GO term, out of all aligned protein pairs in which both proteins are annotated with at least one GO term [6, 43]. In this case, complete GO annotation data is used, independent of GO evidence code.

Experimental GO (Exp-GO). However, since many GO annotations have been obtained via sequence comparison, and since the aligners use sequence information within their NCF, it is important to test the aligners when considering only GO annotation data with experimental evidence codes. This avoids the circular argument of evaluating alignment quality with respect to the same data that was used to construct the alignments [1, 2, 4, 6, 7, 43]. Thus, we discard GO measure from further consideration, and instead, we report results for Exp-GO.

In summary, we focus Exp-GO. The larger its value, the better the biological alignment quality.

2.4 Our Methodology

Problem Definition Existing network alignment methods aim to maximize either node conservation or edge conservation. Further, they treat each conserved edge the same. Here, we aim to simultaneously maximize both node and edge conservation, while favoring conserved edges whose end nodes are highly similar. Given a measure of node conservation (denoted as Node Alignment Quality, NAQ) and a measure of edge conservation (denoted as Edge Alignment Quality, EAQ), our goal is to optimize the following expression (denoted as Alignment Quality, AQ):

$$AQ(G, H, f) = \beta_n NAQ(G, H, f) + \beta_e EAQ(G, H, f), \quad (1)$$

where β_n and β_e are parameters used to balance between NAQ and NEQ. We note that a previous study [36] proposed a similar objective function; however, in our study, we define a new way to measure EAQ (see below).

As a proof of concept, we use the following measures as NAQ and EAQ (although any other measure can be used instead). We use the sum of NCF scores over all aligned pairs as our NAQ, which we denote as weighted node conservation (WNC). We design a novel measure of edge conservation as our EAQ, as follows. Similar to EC, ICS, and S^3 , this new measure counts the number of conserved edges, but unlike EC, ICS, or S^3 that treat each conserved edge the same, our new measure weighs each conserved edge by the NCF-based similarity of its end nodes, so that aligning an edge with highly similar end nodes is preferred over aligning an edge with dissimilar end nodes. We denote our new EAQ measure as weighted edge conservation (WEC).

Formally, we define WNC and WEC as follows. Given a pairwise node similarity matrix s with respect to the given NCF, we denote similarity between $u \in V_G$ and $v \in V_H$ in this matrix as s_{uv} . Also, we represent the injection $f : V_G \rightarrow V_H$ as a matrix $y_{|V_G| \times |V_H|}$, where $y_{ij} = 1$ if and only if $f(i) = j$ and $y_{ij} = 0$ otherwise. Thus, the matrix satisfies the following three constraints:

$$y_{ij} \in \{0, 1\}, \quad \forall i \in V_G, \forall j \in V_H; \quad \sum_{l=1}^{|V_H|} y_{il} \leq 1, \quad \forall i \in V_G; \quad \sum_{l=1}^{|V_G|} y_{lj} \leq 1, \quad \forall j \in V_H \quad (2)$$

Then:

$$WNC = \sum_{i \in V_G} \sum_{j \in V_H} y_{ij} s_{ij} \quad (3)$$

To formally define WEC, recall the definitions of EC, ICS, and S^3 (Section 2.3). All three measures have the same numerator, which we can now rewrite as:

$$|E'_G \cap E'_H| = \frac{1}{2} \sum_{i \in V_G} \sum_{j \in V_H} \sum_{k \in \mathcal{N}_i} \sum_{l \in \mathcal{N}_j} y_{ij} y_{kl} \quad (4)$$

Here, \mathcal{N}_i denotes the neighborhood of node i , i.e., the set of nodes connected to i . Since each conserved edge will be counted twice, the $\frac{1}{2}$ constant corrects for this.

Now, to leverage the weight of conserved edges by the NCF-based similarity of its end nodes (see above), we define WEC as follows:

$$WEC = \sum_{i \in V_G} \sum_{j \in V_H} \sum_{k \in \mathcal{N}_i} \sum_{l \in \mathcal{N}_j} y_{ij} y_{kl} s_{kl} \quad (5)$$

With WNC as our NAQ and WEC as our EAQ, formally, our problem is to find a matrix y that satisfies Eqn. 2 and maximizes the following objective function:

$$\begin{aligned} AQ(G, H, y) &= \beta_n NAQ + \beta_e EAQ = \beta_n WNC + \beta_e WEC \\ &= \beta_n \sum_{i \in V_G} \sum_{j \in V_H} y_{ij} s_{ij} + \beta_e \sum_{i \in V_G} \sum_{j \in V_H} \sum_{k \in \mathcal{N}_i} \sum_{l \in \mathcal{N}_j} y_{ij} y_{kl} s_{kl} \end{aligned} \quad (6)$$

Optimizing the WNC part in Eqn. 6 is solvable in polynomial time (e.g., by using Hungarian algorithm for maximum bipartite weighted matching). However, optimizing the whole function on general graphs is NP-hard. We propose WAVE to solve this problem, while allowing for trade off between node conservation and edge conservation (as the two might not always agree).

Weighted Alignment VotEr (WAVE) Initially, we evaluate different values of β_n and β_e and thus the effect of these parameters on WAVE’s results. Since (as we will show in Section 3.2) equally favoring WNC and WEC (i.e., setting the two parameters to the same value) in general yields best results, in all subsequent analyses, we set $\beta_n = \beta_e = 1$. Given this parameter value choice, we can rewrite Eqn. 6 as:

$$AQ(G, H, y) = \sum_{(i,j) \in V_G \times V_H} y_{ij} \left(s_{ij} + \sum_{(k,l) \in \mathcal{N}_i \times \mathcal{N}_j} y_{kl} s_{kl} \right) \quad (7)$$

Next, we use set $A = \{(u, v) \mid u \in V_G, v \in V_H, y_{uv} = 1\}$ to denote our alignment, so our objective function has set A as a variable. Then, we use a greedy approach to maximize the objective function, as follows. We start with an empty alignment set A_0 . In each step t , given the current alignment A_{t-1} , we calculate the marginal gain of adding an available node pair (u, v) (in the sense that so far v and u are both unaligned) into A . (For a function $f(S)$ with variable S as a set, the marginal gain of adding an element e into S is defined as $f(S \cup \{e\}) - f(S)$.) That is, we calculate: $AQ(A_{t-1} \cup \{(u, v)\}) - AQ(A_{t-1})$. Then, we align the pair (u^*, v^*) with the highest marginal gain, i.e., $A_t = A_{t-1} \cup \{(u^*, v^*)\}$. To calculate the marginal gain efficiently, we keep the current marginal gain of each node pair and update it in each step. The marginal gain of the node pair (u, v) to AQ is s_{uv} at the beginning (when A is empty, if we align this pair, we can only get s_{uv} in WNC part). In each step, note that if we align two nodes $u \in V_G$ and $v \in V_H$, the side effect is that, in the following steps, when we align another pair of nodes $u' \in \mathcal{N}_u, v' \in \mathcal{N}_v$, both the similarity of (u, v) and (u', v') will be counted once more by the correctly linked edge, namely, the edge $(u, u') \in E_G$ and $(v, v') \in E_H$. Thus, the marginal gain of (u', v') will be $s_{uv} + s_{u'v'}$ more after (u, v) is aligned.

Intuitively, this process is like voting. When a pair of nodes is aligned, this node pair has a chance to vote for their neighbors: when u and v are aligned, all other node pairs in $\mathcal{N}_u \times \mathcal{N}_v$ receive a weighted vote (with weight $s_{uv} + s_{u'v'}$) from (u, v) , and the weight consists of two parts: 1) the “authority” of the voter, i.e., s_{uv} , 2) the “certainty” of the votee, i.e., $s_{u'v'}$.

The weight for the initial votes of each node pair is the original s_{uv} (which forms the WNC part in the objective function). In every round of WAVE, node pair (u^*, v^*) with the highest vote is aligned, and (u^*, v^*) then vote for all the pairs in $\mathcal{N}_{u^*} \times \mathcal{N}_{v^*}$. The current vote that a node pair gets from its aligned neighbors is the marginal gain to objective function of aligning them.

For WAVE’s pseudocode, see Algorithm 1. For its implementation, visit: <http://nd.edu/~cone/WAVE/WAVE.zip>

Algorithm 1 Weighted Alignment VotEr (WAVE) pseudocode

Input: $G = (V_G, E_G)$, $H = (V_H, E_H)$, $s_{uv} ((u, v) \in V_G \times V_H)$
Output: Alignment $f : V_G \rightarrow V_H$

- 1: **for** $(u, v) \in V_G \times V_H$ **do**
- 2: $vote_{u,v} \leftarrow \beta_n \times s_{u,v}$;
- 3: **end for**
- 4: **for** $u \in V_G$ **do**
- 5: $visitedSrc_u \leftarrow false$;
- 6: **end for**
- 7: **for** $v \in V_H$ **do**
- 8: $visitedTar_v \leftarrow false$;
- 9: **end for**
- 10: **for** $round = 1$ to $|V_G|$ **do**
- 11: $(u^*, v^*) \leftarrow \arg \max_{unaligned(u,v)} vote_{u,v}$;
 where $unaligned(u, v)$ means both $visitedSrc_u$ and $visitedTar_v$ are
 false
- 12: $visitedSrc_{u^*} \leftarrow true$;
- 13: $visitedTar_{v^*} \leftarrow true$;
- 14: $f(u^*) \leftarrow v^*$;
- 15: **for** $(u, v) \in \mathcal{N}_{u^*} \times \mathcal{N}_{v^*}$ **do**
- 16: $vote_{u,v} \leftarrow vote_{u,v} + \beta_e \times (s_{u,v} + s_{u^*,v^*})$
- 17: **end for**
- 18: **end for**

Return f

3 Results and Discussion

We denote the five aligners resulting from mixing and matching NCFs of MI-GRAAL and GHOST with ASs of MI-GRAAL, GHOST, and WAVE as M-M, M-W, G-M, G-G, and G-W (Section 1.3).

Recall that a key novelty of WAVE is that while optimizing edge conservation (in addition to node conservation), WAVE weighs each conserved edge to favor aligning edges with highly NCF-similar end nodes. Thus, to evaluate whether weighing conserved edges leads to better alignments, we first compare the performance of the edge-weighted versions of WAVE (i.e., M-W(W) and G-W(W)) and its edge-unweighted versions (i.e., M-W(U) and G-W(U)). As we will show, the edge-weighted versions are superior.

Further, we evaluate the effect of the different parameters (i.e., β_n and β_e) on WAVE’s results. As we will show, assigning the same value to the two parameters, i.e., equally favoring node and edge conservation, is superior to other parameter variations.

Next, using the edge-weighted versions of WAVE with $\beta_n = 1$ and $\beta_e = 1$, we evaluate the five aligners (M-M, M-W, G-M, G-G, and G-W) against each other. Also, we evaluate WAVE (the best of M-W and G-W) against NETAL and MAGNA. By comparing M-M and M-W, we can directly and fairly evaluate ASs of MI-GRAAL and WAVE under MI-GRAAL’s NCF. By comparing G-M, G-G, and G-W, we can directly and fairly evaluate ASs of MI-GRAAL, GHOST, and WAVE under GHOST’s NCF. If WAVE AS produces better alignments compared to the existing methods’ ASs under both of the existing NCFs, this would indicate WAVE’s superiority. If WAVE also produces better alignments compared to NETAL and MAGNA, this would even further demonstrate WAVE’s superiority. However, this is not a strict requirement, as the two new methods either implement both different (newer, and thus possibly superior) NCF than any of M-W and G-W as well as different AS (in case of NETAL), which might give them an unfair advantage, or they work on different principles (in case of MAGNA) and could be thus viewed as complementary to WAVE.

For each combination of network pair, value of α (denoting topological versus sequence information within NCF), and alignment quality measure (Section 2), we do the following. First, to extract the most out of each source of biological information, it would be beneficial to know how much of new biological knowledge can be uncovered solely from topology before integrating it with other sources of biological information, such as protein sequence information [1, 2, 4]. Thus, we first compare the different edge-weighted and edge-unweighted versions of WAVE, the different combinations of β_n and β_e parameter values, and the different NCF-AS methods, on topology-only alignments (corresponding to α of 1 within NCF). Also, since NETAL and MAGNA also produce topology-only alignments, here, we can compare WAVE to these methods. Second, we examine different contributions of topology versus sequence information in NCF (by varying α), and for each method, we choose the best value of α , i.e., the

method’s *best alignment*. We do this when comparing the different edge-weighted and edge-unweighted versions of WAVE, as well as the five NCF-AS methods to each other. On the other hand, we do not do this when comparing the different combinations of β_n and β_e parameter values, due to the large number of evaluation tests required in this analysis. Also, since current implementations of NETAL and MAGNA do not allow for inclusion of sequence information, we cannot directly compare WAVE to these methods when adding sequence information into NCF. However, since in real-life applications one should give the best-case advantage to each method, we do compare best alignments of WAVE with topology-only alignments of NETAL and MAGNA, and we do consider this as comparison of the methods’ best alignments.

For “synthetic” (noisy yeast) networks with known node mapping, we report alignment quality with respect to NC, S³, LCCS, and Exp-GO. For real-world PPI networks of different species with unknown node mapping, we report alignment quality with respect to S³, LCCS, and Exp-GO.

3.1 Comparison of Edge-weighted and Edge-unweighted Versions of WAVE

Here, we compare the edge-weighted and edge-unweighted versions of WAVE. We find that weighing conserved edges in general improves alignment quality (Figs. 1 and 2, as well as Figs. A.1 and A.2 in the Appendix), as follows.

Networks with Known Node Mapping

Topological alignments. Weighing conserved edges improves alignment quality of topology-only alignments under both MI-GRAAL’s and GHOST’s NCFs, since the edge-weighted version of WAVE is comparable or superior to the edge-unweighted version in the majority of cases across all alignments and all alignment quality measures (Fig. 1).

Best alignments. Here, under MI-GRAAL’s NCF, the edge-weighted version of WAVE is comparable or superior to the edge-unweighted version across all alignments and all alignment quality measures (Fig. 2). Under GHOST’s NCF, the edge-weighted version is comparable or superior to the edge-unweighted version in the majority of cases. Thus, the edge-weighted version is even more preferred by best alignments compared to topology-only alignments.

Networks with Unknown Node Mapping

Topological alignments. Here, the edge-weighted version of WAVE is comparable or superior to the edge-unweighted version under MI-GRAAL’s NCF for two out of three alignment quality measures (Fig. A.1 in the Appendix). Under GHOST’s NCF, the edge-weighted version of WAVE is rarely favored in this evaluation test. Nonetheless, the edge-weighted version is still favored over all evaluation tests.

Best alignments. The edge-weighted version is preferred under MI-GRAAL’s NCF for all three alignment quality measures and under GHOST’s NCF for one of the measures, since in these cases the edge-weighted version is comparable or superior to the edge-unweighted version in the majority of cases (Fig. A.2 in the Appendix).

In summary, over both network sets (with known and unknown node mapping), both topology-only and best alignments, and all alignment quality measures, the edge-weighted version of WAVE is overall (though not always) superior to the edge-unweighted version. Over all cases in which we do observe superiority of the edge-weighted version over the edge-unweighted version, the level of superiority ranges from 1.73% to 9.56% (with the average of 4.54%) for topology-only alignments and from 1.65% to 5.51% (with the average of 2.84%) for best alignments (Table 1). Interestingly, superiority of the edge-weighted version of WAVE becomes more pronounced with increase of noise in the data, especially for topology-only alignments (we base this conclusion only on “synthetic” (noisy yeast) networks for which we know the level of noise in the data). Because the edge-weighted version of WAVE is overall the superior one, in the following sections, we use the edge-weighted version.

Table 1. Improvements of an edge-weighted version of WAVE over its edge-unweighted counterpart, over all evaluation tests in which the edge-weighted version is the superior one. The results are shown in terms of the minimum (“Min”), maximum (“Max”) and average (“Avg”) improvement over all such tests, along with the corresponding standard deviation (“Stdev”).

Aligner / Network set	Topology-only alignments				Best alignments			
	Min	Max	Avg	Stdev	Min	Max	Avg	Stdev
M-W / “Synthetic” (noisy yeast) networks	0.03%	11.79%	4.71%	4.24%	0.17%	4.11%	1.41%	1.21%
M-W / Real-world PPI networks of different species	1.69%	6.51%	3.51%	1.73%	1.22%	6.51%	2.60%	1.85%
G-W / “Synthetic” (noisy yeast) networks	0.03%	11.84%	3.22%	4.7%	0.04%	3.11%	0.61%	1.11%
G-W / Real-world PPI networks of different species	5.16%	8.30%	6.73%	2.22%	5.16%	8.30%	6.73%	2.22%
Average across all aligners and network sets	1.73%	9.56%	4.54%	3.24%	1.65%	5.51%	2.84%	1.60%

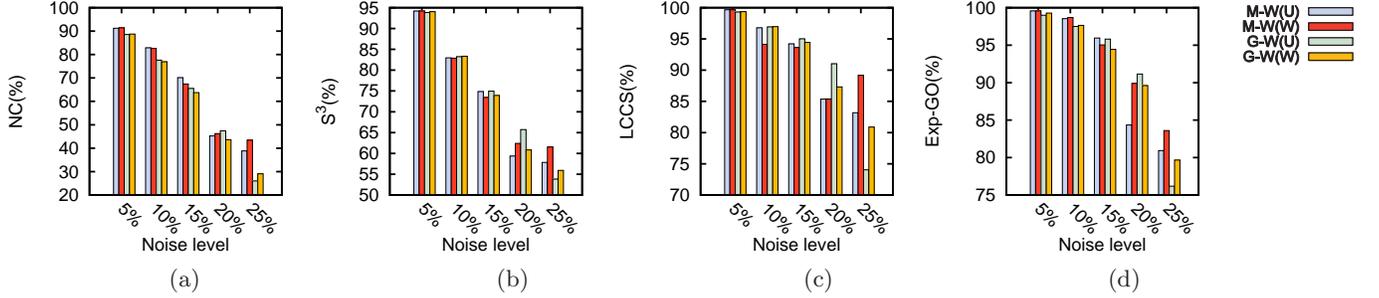


Fig. 1. Comparison of the edge-weighted and edge-unweighted versions of WAVE on topology-only alignments of “synthetic” (noisy yeast) networks with respect to (a) NC, (b) S^3 , (c) LCCS, and (d) Exp-GO. For analogous results for real-world PPI networks of different species, see Fig. A.1 in the Appendix.

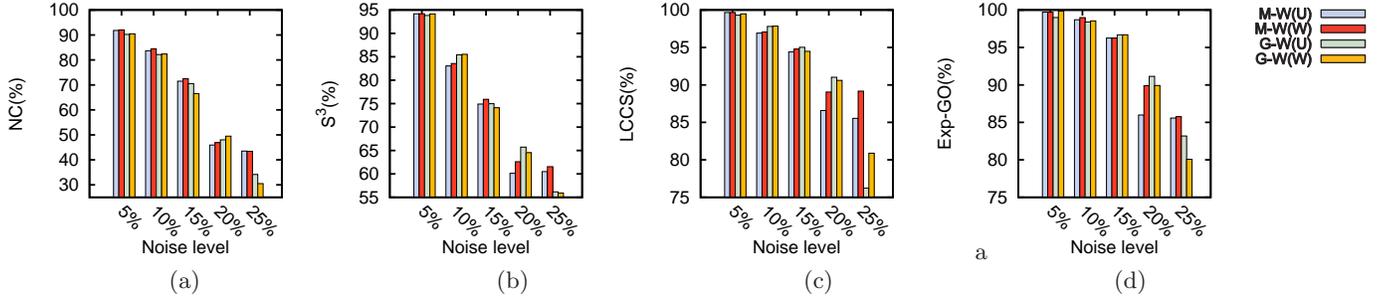


Fig. 2. Comparison of the edge-weighted and edge-unweighted versions of WAVE on best alignments of “synthetic” (noisy yeast) networks with respect to (a) NC, (b) S^3 , (c) LCCS, and (d) Exp-GO. For analogous results for real-world PPI networks of different species, see Fig. A.2 in the Appendix.

3.2 Comparison of different parameter values within WAVE

When we vary values of β_n and β_e parameters, corresponding to levels of node and edge conservation considered within WAVE, we find that assigning the same value to the two parameters overall leads to the best results (Fig. 3). This holds for M-W and G-W combined (Fig. 3(a)) as well as for M-W only (Fig. 3(b)) and G-W only (Fig. 3(c)).

Our above conclusion comes from the following observations. The combination in which only β_n is used (i.e., $\beta_n = 1$ and $\beta_e = 0$) is always inferior to any other combination of the parameter values. That is, considering only node conservation within WAVE (which is what most of the existing methods do) is inferior, and accounting for edge conservation improves results. The remaining combinations of parameter values that use some level of edge conservation (i.e., $\beta_n = 0.75$ and $\beta_e = 0.25$, $\beta_n = 0.5$ and $\beta_e = 0.5$, $\beta_n = 0.25$ and $\beta_e = 0.75$, and $\beta_n = 0$ and $\beta_e = 1$) are overall comparable to each other, with slight superiority of $\beta_n = 0.5$ and $\beta_e = 0.5$, especially for G-W (Fig. 3(c)). Even for all versions of WAVE (Fig. 3(a)), we argue that the combination of $\beta_n = 0.5$ and $\beta_e = 0.5$ is overall superior. Namely, even though the combination of $\beta_n = 0$ and $\beta_e = 1$ is ranked as the first best combination in most of the cases, the combination of $\beta_n = 0.5$ and $\beta_e = 0.5$ is following very closely (Fig. 3(a)). Further, the combination of $\beta_n = 0.5$ and $\beta_e = 0.5$ is ranked as the second or third best in more cases than the combination of $\beta_n = 0$ and $\beta_e = 1$; in other words, the combination of $\beta_n = 0$ and $\beta_e = 1$ is ranked as *the worst* (i.e., fourth) in more cases than the combination of $\beta_n = 0.5$ and $\beta_e = 0.5$ (or any other combination that considers some level of edge conservation). For these reasons, in our study, we have adopted this overall superior combination of $\beta_n = 0.5$ and $\beta_e = 0.5$ (or equivalently $\beta_n = 1$ and $\beta_e = 1$), which equally favors node and edge conservation.

Due to a large number of tests involved into evaluating different combinations of β_n and β_e values, all experiments in this section have been performed only on topology-only alignments of “synthetic” (noisy yeast) networks.

3.3 Comparison of Five NCF-AS Methods

Here, we compare M-M, M-W, G-M, G-G, and G-W, to test whether WAVE AS improves upon ASs of MI-GRAAL and GHOST under the same (MI-GRAAL’s or GHOST’s) NCF.

Networks With Known Node Mapping

Topological alignments. WAVE is always superior to the existing methods (M-W is superior to M-M, and G-W is superior to G-M and G-G), for all noise levels and alignment quality measures, under both MI-GRAAL’s and GHOST’s NCFs (Fig. 4 (a) and 5).

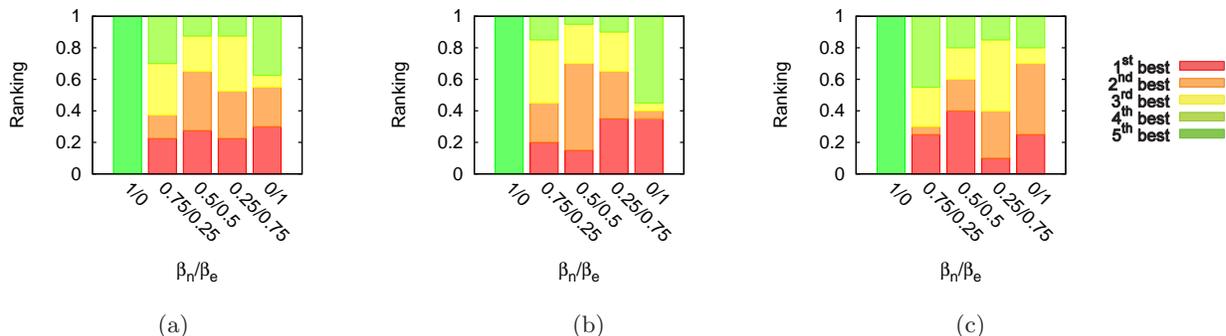


Fig. 3. Overall ranking of each tested β_n/β_e combination over all network pairs in the “synthetic” (noisy yeast) network data set and over all alignment quality measures, for topology-only alignments of: (a) both M-W and G-W combined, (b) M-W only, and (c) G-W only. The ranking is expressed as a percentage of all cases (i.e., all evaluation tests) in which the combination ranks as the k^{th} best method. That is, the more cases in which a given combination achieves a higher ranking, the better the combination. For example, in panel (c), the combination of $\beta_n = 0.5$ and $\beta_e = 0.5$ (or equivalently, $\beta_n = 1$ and $\beta_e = 1$) is superior to all other combinations, since is ranked the highest (i.e., as the 1st best method) in most of the cases.

WAVE in general works better under MI-GRAAL’s NCF than under GHOST’s NCF, as M-W is overall superior to G-W. WAVE (at least one of M-W and G-W) beats both MI-GRAAL and GHOST (all of M-M, G-M, and G-G) in 20/20=100% of all cases (Figs. 4 (a) and 5). These results hold across all noise levels.

Best alignments. Here, we give the best-case advantage to each method by selecting its optimal α parameter value. Under MI-GRAAL’s NCF, WAVE is always superior (M-W is better than M-M), for all noise levels and alignment quality measures (Figs. A.3 (a) and A.4 in the Appendix).

Under GHOST’s NCF, WAVE is always superior to MI-GRAAL’s AS (G-W is better than G-M), and WAVE is superior to GHOST’s AS (G-W is better than G-G) with respect to two of the four measures (edge-based S^3 and LCCS), while GHOST’s AS is superior (G-G is better than G-W) with respect to the other two measures (node-based NC and Exp-GO) (Figs. A.3 (a) and A.4 in the Appendix). Hence, WAVE and GHOST’s AS are comparable overall.

Again, WAVE in general works better under MI-GRAAL’s NCF than under GHOST’s, as M-W is overall superior to G-W. WAVE (at least one of M-W and G-W) beats both MI-GRAAL and GHOST (all of M-M, G-M, and G-G) in 6/10=60% of cases dealing with the two edge-based measures of alignment quality (Figs. A.3 (a) and A.4 in the Appendix). The ranking of the different methods does not change with increase of noise level with respect to NC and Exp-GO, but it does change with respect to S^3 and LCCS for the highest noise levels.

Networks With Unknown Node Mapping

Topological alignments. Under MI-GRAAL’s NCF, WAVE is always superior (M-W is better than M-M) with respect to S^3 , it is almost always superior with respect to LCCS, and it is sometimes superior with respect to Exp-GO (Figs. 4 (b) and 6). Hence, here WAVE seems to be favored by topological alignment quality measures.

Under GHOST’s NCF, WAVE is superior to MI-GRAAL’s AS (G-W is better than G-M) in almost all cases, for each of S^3 , LCCS, and Exp-GO (Figs. 4 (b) and 6). Also, here WAVE is overall superior to GHOST’s AS (G-W is better than G-G) with respect to Exp-GO but not with respect to S^3 or LCCS (Figs. 4 (b) and 6).

WAVE in general works better under MI-GRAAL’s NCF than under GHOST’s NCF, as M-W is overall superior to G-W. WAVE (at least one of M-W and G-W) beats both MI-GRAAL and GHOST (all of M-M, G-M, and G-G) in 14/18=78% of all cases (Figs. 4 (b) and 6).

Best alignments. Under MI-GRAAL’s NCF, WAVE is always superior (M-W is better than M-M) with respect to S^3 , and it is almost always superior with respect to LCCS as well as Exp-GO (Figs. A.3 (b) and A.5 in the Appendix). Hence, here WAVE is even more superior than for topological alignments only.

Under GHOST’s NCF, WAVE is superior to MI-GRAAL’s AS (as G-W is better than G-M) in most cases for each of S^3 and Exp-GO, and in some cases for LCCS. Also, here WAVE is overall superior to GHOST’s AS (G-W is better than G-G) with respect to Exp-GO but not with respect to S^3 or LCCS (Figs. A.3 (b) and A.5 in the Appendix).

Again, WAVE works better under MI-GRAAL’s NCF than under GHOST’s AS, as M-W is superior to G-W. WAVE (at least one of M-W and G-W) beats both MI-GRAAL and GHOST (all of M-M, G-M, and G-G) in 13/18=72% of all cases (Figs. A.3 (b) and A.5 in the Appendix).

The fact that WAVE in general works better under MI-GRAAL’s NCF than under GHOST’s NCF further adds to our recent finding that MI-GRAAL’s NCF is superior to other NCFs [7, 43].

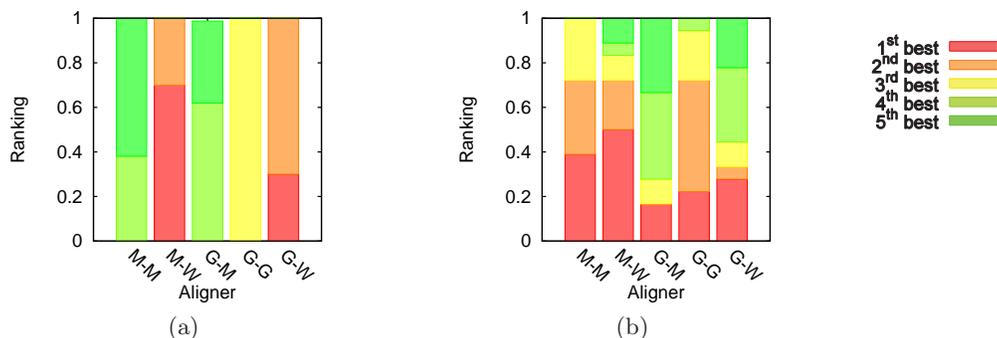


Fig. 4. Representative results for overall ranking of each NCF-AS method over all network pairs in a given data set and over all alignment quality measures. The ranking is expressed as a percentage of all cases (i.e., all evaluation tests) in which the given method ranks as the k^{th} best method. That is, the more cases in which a given method achieves a higher ranking, the better the method. For example, in panel (a), M-W is the highest scoring of all methods shown on x -axis, since it is ranked the highest (i.e., as the 1st best method) in most of the cases. **(a)** Results for the five NCF-AS methods on topology-only alignments of “synthetic” (noisy yeast) networks. For equivalent results for best alignments, see Fig. A.3 (a) in the Appendix. **(b)** Results for the five NCF-AS methods on topology-only alignments of real-world PPI networks of different species. For equivalent results for best alignments, see Fig. A.3 (b) in the Appendix. Details (per network pair and alignment quality measure) for panels (a)-(b) are shown in Figs. 5-6, respectively. Recall that M-M and G-G are MI-GRAAL and GHOST.

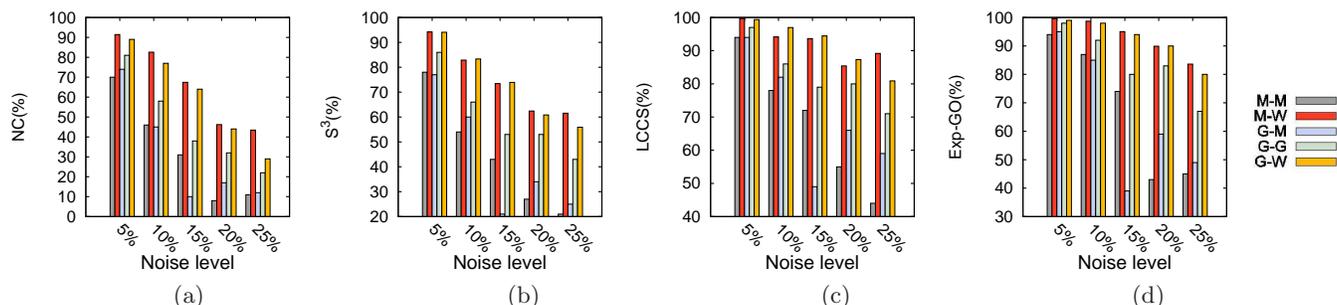


Fig. 5. Comparison of the five NCF-AS methods on topology-only alignments of “synthetic” (noisy yeast) networks with respect to: **(a)** NC, **(b)** S^3 , **(c)** LCCS, and **(d)** Exp-GO. For analogous results for best alignments, see Fig. A.4 in the Appendix.

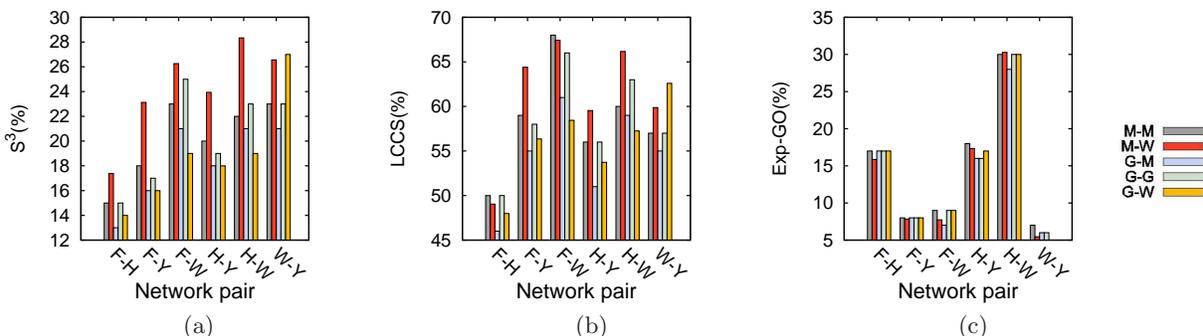


Fig. 6. Comparison of the five NCF-AS methods on topology-only alignments of real-world PPI networks of different species with respect to: **(a)** S^3 , **(b)** LCCS, and **(c)** Exp-GO. For analogous results for best alignments, see Fig. A.5 in the Appendix.

3.4 Comparison of WAVE with Very Recent Methods

Here, we compare WAVE (the best of M-W and G-W) with NETAL and MAGNA. Since the latter two became available at completion of our study, we could not include their novelties (e.g., NETAL’s NCF) into our methodology.

Networks With Known Node Mapping

Topological alignments. WAVE is always superior to both NETAL and MAGNA, for all noise levels and alignment quality measures (Fig. 7 (a) and Fig. A.6 in the Appendix). Only in $2/20 = 10\%$ of all cases, MAGNA is superior:

with respect to S^3 for two largest noise levels. But this is not surprising, as MAGNA optimizes S^3 . Overall, the ranking of the different methods does not change with increase in noise level.

Best alignments. Recall that NETAL and MAGNA do now allow for inclusion of sequence information into the alignment construction process. So, for these two methods, their best alignments are actually their topology-only alignments. For WAVE, on this “synthetic” (noisy yeast) network set, topology-only alignments are the best of all alignments (i.e., inclusion of sequence information decreases alignment quality). Thus, results do not change from topology-only to best alignments when comparing the three methods: WAVE remains superior to NETAL and MAGNA (Figs. 7 (b) and 8). Again, overall, the ranking of the methods does not change with increase in noise level.

Networks With Unknown Node Mapping

Topological alignments. WAVE is always superior to MAGNA, for all noise levels and alignment quality measures (Fig. 7 (c) and Fig. A.7 in the Appendix). Only in one out of 18 cases, MAGNA is superior to WAVE: with respect to S^3 for one of the six network pairs. NETAL is overall superior to the other two methods, especially with respect to topological alignment quality measures (S^3 and LCCS) (Figs. 7 (c) and A.7). This could be because NETAL has both different NCF and AS compared to WAVE, and as such, its superiority might be a consequence not of its ASs but rather of its NCF. So, if its NCF was fed into WAVE AS, this could perhaps result in a superior new method. This possibility of designing a novel superior method simply by mixing NCF of one method and AS of another method has already been confirmed on several occasions [7, 43].

Best alignments. When each method is given the best-case advantage, WAVE remains superior to MAGNA, and moreover, its ranking against NETAL now improves compared to topology-only alignments (Figs. 7 (d) and 9), which confirms real-life relevance of WAVE.

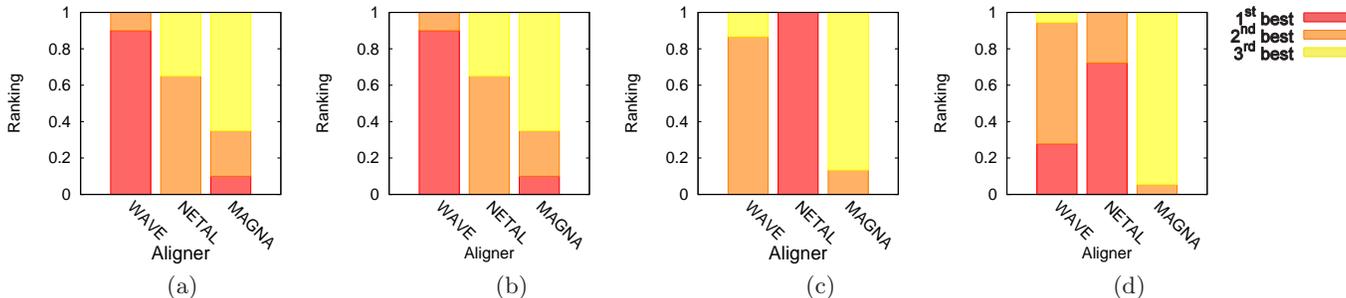


Fig. 7. Overall ranking of WAVE (the best of M-W and G-W) against the recent methods (NETAL and MAGNA) over all network pairs in a given data set and over all alignment quality measures. The ranking is expressed as a percentage of all cases (i.e., all evaluation tests) in which the given method ranks as the k^{th} best method. That is, the more cases in which a given method achieves a higher ranking, the better the method. Results are shown for: (a) topology-only alignments of “synthetic” (noisy yeast) networks, (b) best alignments of “synthetic” (noisy yeast) networks, (c) topology-only alignments of real-world PPI networks of different species, and (d) best alignments of real-world PPI networks of different species. Details (per network pair and alignment quality measure) for panels (a)-(d) are shown in Fig. A.6 in the Appendix, Fig. 8, Fig. A.7 in the Appendix, and Fig. 9, respectively. Recall that M-M and G-G are MI-GRAAL and GHOST.

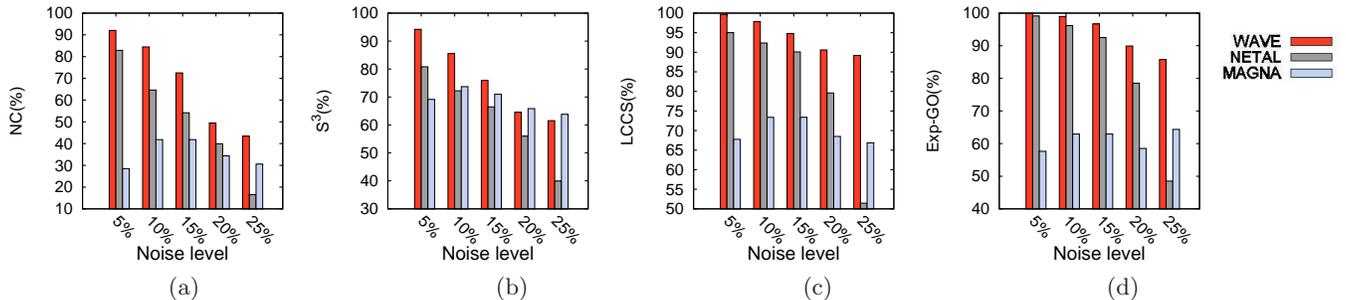


Fig. 8. Comparison of WAVE (the best of M-W and G-W) with very recent network alignment methods on best alignments of “synthetic” (noisy yeast) networks with respect to: (a) NC, (b) S^3 , (c) LCCS, and (d) Exp-GO. For analogous results for topology-only alignments of “synthetic” (noisy yeast) networks, see Fig. A.6 in the Appendix.

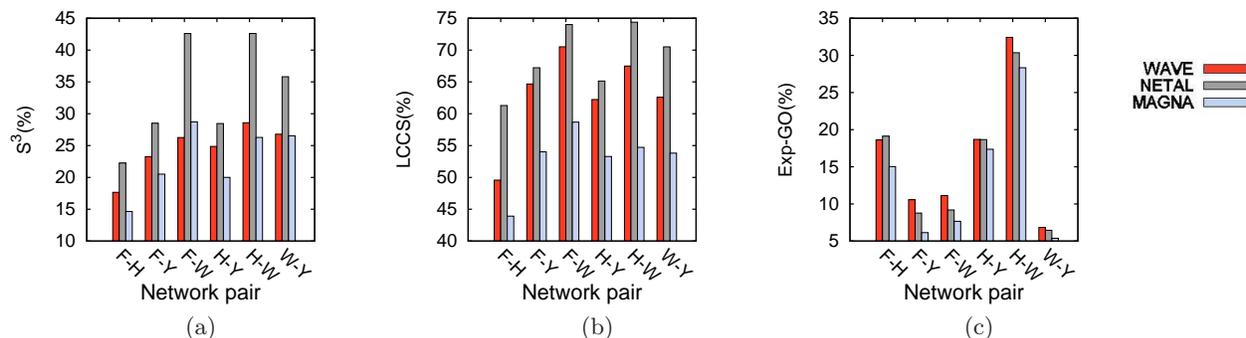


Fig. 9. Comparison of WAVE (the best of M-W and G-W) with very recent network alignment methods on best alignments of real-world PPI networks of different species with respect to: (a) S^3 , (b) LCCS, and (c) Exp-GO. For analogous results for topology-only alignments of real-world PPI networks of different species, see Fig. A.7 in the Appendix.

4 Concluding Remarks

We have presented WAVE, a general network alignment strategy for simultaneously optimizing both node conservation and weighted edge conservation, which can be used with any node cost function or combination of multiple node cost functions. We have demonstrated overall superiority of WAVE against existing state-of-the-art alignment strategies under multiple node cost functions, especially with respect to topological alignment quality. Moreover, we have demonstrated that WAVE is comparable or superior even to very recent approaches that became available only close to completion of our study, especially on the synthetic network data. This only further validates the effectiveness of WAVE.

Since WAVE can be combined with any node cost function, doing so for any recent function might improve its alignment quality. Also, WAVE itself can be modified to optimize any other measure of node and edge conservation, which could further improve its accuracy; the measures that we have used are merely a proof of concept that optimizing both node and weighted edge conservation can lead to better alignments compared to optimizing just node conservation (as e.g., MI-GRAAL and GHOST do) or just unweighted edge conservation (as e.g., MAGNA does).

As more biological network data are becoming available, network alignment will only continue to gain importance in the computational biology domain [56, 42, 57]. For example, network alignment has already redefined the notion of sequence-based orthology to the notion of network-based orthology, as it can identify conserved network (rather than sequence) regions between different species [58]. Then, network alignment can guide the transfer of biological (e.g., aging-related) knowledge from well-studied model species to poorly-studied species such as human. Hence, given WAVE’s superiority as demonstrated in our study, WAVE could further our biological insights. Applying WAVE to an interesting biological question is out of the scope of this current method evaluation study and is subject of our future work. Further, network alignment (and thus WAVE) has implications in many domains. For example, it can be used to de-anonymize online social networks and thus impact privacy [38]. Hence, further theoretical improvements that would lead to better network alignments have a potential to lead to important discoveries in different fields.

Acknowledgements

This work was funded by the National Science Foundation CAREER CCF-1452795 and CCF-1319469 grants.

References

1. Oleksii Kuchaiev, Tijana Milenković, Vesna Memišević, Wayne Hayes, and Nataša Pržulj. Topological network alignment uncovers biological function and phylogeny. *Journal of The Royal Society Interface*, 7(50):1341–1354, 2010.
2. Tijana Milenković, Weng Leong Ng, Wayne Hayes, and Nataša Pržulj. Optimal network alignment with graphlet degree vectors. *Cancer Informatics*, 9:121–137, 2010.
3. Chung-Shou Liao, Kanghao Lu, Michael Baym, Rohit Singh, and Bonnie Berger. IsoRankN: Spectral methods for global alignment of multiple protein networks. *Bioinformatics*, 25(12):i253–258, 2009.
4. Oleksii Kuchaiev and Nataša Pržulj. Integrative network alignment reveals large regions of global network similarity in yeast and human. *Bioinformatics*, 27(10):1390–1396, 2011.

5. Rob Patro and Carl Kingsford. Global network alignment using multiscale spectral signatures. *Bioinformatics*, 28(23):3105–3114, 2012.
6. Vikram Saraph and Tijana Milenković. MAGNA: Maximizing Accuracy in Global Network Alignment. *Bioinformatics*, page btu409, 2014.
7. Tijana Milenković, Han Zhao, and Fazle E. Faisal. Global network alignment in the context of aging. *Computational Biology and Bioinformatics, IEEE/ACM Transactions on Computational Biology and Bioinformatics*, PP(99), 2014.
8. Juanzi Li, Jie Tang, Yi Li, and Qiong Luo. Rimom: A dynamic multistrategy ontology alignment framework. *Knowledge and Data Engineering, IEEE Transactions on*, 21(8):1218–1232, 2009.
9. Fabian M Suchanek, Serge Abiteboul, and Pierre Senellart. Paris: Probabilistic alignment of relations, instances, and schema. *Proceedings of the VLDB Endowment*, 5(3):157–168, 2011.
10. Simon Lacoste-Julien, Konstantina Palla, Alex Davies, Gjergji Kasneci, Thore Graepel, and Zoubin Ghahramani. Sigma: Simple greedy matching for aligning large knowledge bases. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 572–580. ACM, 2013.
11. Sergey Melnik, Hector Garcia-Molina, and Erhard Rahm. Similarity flooding: A versatile graph matching algorithm and its application to schema matching. In *Proc. 18th ICDE Conf.*, 2002.
12. Donatello Conte, Pasquale Foggia, Carlo Sansone, and Mario Vento. Thirty years of graph matching in pattern recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 18(03):265–298, 2004.
13. Mikhail Zaslavskiy, Francis Bach, and J-P Vert. A path following algorithm for the graph matching problem. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(12):2227–2242, 2009.
14. Danai Koutra, Hanghang Tong, and David Lubensky. Big-align: Fast bipartite graph alignment. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, pages 389–398. IEEE, 2013.
15. Yutao Zhang and Jie Tang. Social network integration: Towards constructing the social graph. *arXiv preprint arXiv:1311.2670 [cs.SI]*, 2013.
16. Mohsen Bayati, Margot Gerritsen, David F Gleich, Amin Saberi, and Ying Wang. Algorithms for large, sparse network alignment problems. In *Data Mining, 2009. ICDM'09. Ninth IEEE International Conference on*, pages 705–710. IEEE, 2009.
17. Lorenzo Torresani, Vladimir Kolmogorov, and Carsten Rother. Feature correspondence via graph matching: Models and global optimization. In *Computer Vision–ECCV 2008*, pages 596–609. Springer, 2008.
18. Alexandre Noma and Roberto Marcondes Cesar. Sparse representations for efficient shape matching. In *Graphics, Patterns and Images (SIBGRAPI), 2010 23rd SIBGRAPI Conference on*, pages 186–192. IEEE, 2010.
19. Olivier Duchenne, Francis Bach, In-So Kweon, and Jean Ponce. A tensor-based algorithm for high-order graph matching. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(12):2383–2395, 2011.
20. Lorenzo Torresani, Vladimir Kolmogorov, and Carsten Rother. Feature correspondence via graph matching: Models and global optimization. In *Computer Vision–ECCV 2008*, pages 596–609. Springer, 2008.
21. Aaron Smalter, Jun Huan, and Gerald Lushington. Gpm: A graph pattern matching kernel with diffusion for chemical compound classification. In *Bioinformatics and BioEngineering, 2008. BIBE 2008. 8th IEEE International Conference on*, pages 1–6. IEEE, 2008.
22. Fazle E Faisal and Tijana Milenković. Dynamic networks reveal key players in aging. *Bioinformatics*, 30(12):1721–1729, 2014.
23. Stephen A Cook. The complexity of theorem-proving procedures. In *Proceedings of the 3rd Annual ACM Symposium on Theory of Computing*, pages 151–158, 1971.
24. Brian P. Kelley, Bingbing Yuan, Fran Lewitter, Roded Sharan, Brent R. Stockwell, and Trey Ideker. PathBLAST: a tool for alignment of protein interaction networks. *Nucleic Acids Research*, 32:83–88, 2004.
25. Roded Sharan, Silpa Suthram, Ryan M Kelley, Tanja Kuhn, Scott McCuine, Peter Uetz, Taylor Sittler, Richard M Karp, and Trey Ideker. Conserved patterns of protein interaction in multiple species. *Proceedings of the National Academy of Sciences*, 102(6):1974–1979, 2005.
26. Jason Flannick, Antal Novak, Balaji S Srinivasan, Harley H McAdams, and Serafim Batzoglou. Graemlin general and robust alignment of multiple large interaction networks. *Genome Research*, 16(9):1169–1181, 2006.
27. Mehmet Koyutürk, Yohan Kim, Umut Topkara, Shankar Subramaniam, Wojciech Szpankowski, and Ananth Grama. Pairwise alignment of protein interaction networks. *Journal of Computational Biology*, 13(2), 2006.
28. Johannes Berg and Michael Lässig. Local graph alignment and motif search in biological networks. *Proceedings of the National Academy of Sciences*, 101(41):14689–14694, 2004.
29. Zhi Liang, Meng Xu, Maikun Teng, and Liwen Niu. NetAlign: A web-based tool for comparison of protein interaction networks. *Bioinformatics*, 22(17):2175–2177, 2006.
30. Johannes Berg and Michael Lässig. Cross-species analysis of biological networks by Bayesian alignment. *Proceedings of the National Academy of Sciences*, 103(29):10967–10972, 2006.
31. Marco Mina and Pietro Hiram Guzzi. Improving the robustness of local network alignment: design and extensive assessment of a markov clustering-based approach. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 99(PrePrints):1, 2014.
32. Giovanni Ciriello, Marco Mina, Pietro H. Guzzi, Mario Cannataro, and Concettina Guerra. AlignNemo: A Local Network Alignment Method to Integrate Homology and Topology. *PLOS ONE*, 7(6), 2012.
33. Rohit Singh, Jinbo Xu, and Bonnie Berger. Pairwise global alignment of protein interaction networks by matching neighborhood topology. In *Research in computational molecular biology*, pages 16–31. Springer, 2007.

34. Jason Flannick, Antal Novak, Chuong B. Do, Balaji S. Srinivasan, and Serafim Batzoglou. Automatic parameter learning for multiple network alignment. In *Proceedings of the 12th Annual International Conference on Research in Computational Molecular Biology*, pages 214–231, 2008.
35. Rohit Singh, Jinbo Xu, and Bonnie Berger. Global alignment of multiple protein interaction networks. In *Proceedings of Pacific Symposium on Biocomputing 13*, pages 303–314, 2008.
36. Mikhail Zaslavskiy, Francis Bach, and Jean-Philippe Vert. Global alignment of protein-protein interaction networks by graph matching methods. *Bioinformatics*, 25(12):i259–i267, 2009.
37. Behnam Neyshabur, Ahmadreza Khadem, Somaye Hashemifar, and Seyed Shahriar Arab. NETAL: a new graph-based method for global alignment of protein-protein interaction networks. *Bioinformatics*, 29(13):1654–1662, 2013.
38. Arvind Narayanan, Elaine Shi, and Benjamin IP Rubinstein. Link prediction by de-anonymization: How we won the Kaggle social network challenge. In *Proceedings of the 2011 International Joint Conference on Neural Networks (IJCNN)*, pages 1825–1834. IEEE, 2011.
39. Xin Guo and Alexander J Hartemink. Domain-oriented edge-based alignment of protein interaction networks. *Bioinformatics*, 25(12):i240–i246, 2009.
40. Gunnar W Klau. A new graph-based method for pairwise global network alignment. *BMC Bioinformatics*, 10(Suppl 1):S59, 2009.
41. Mohammed El-Kebir, Jaap Heringa, and Gunnar W Klau. Lagrangian relaxation applied to sparse global network alignment. In *Pattern Recognition in Bioinformatics*, pages 225–236. Springer, 2011.
42. Connor Clark and Jugal Kalita. A comparison of algorithms for the pairwise alignment of biological networks. *Bioinformatics*, page btu307, 2014.
43. Joseph Crawford, Yihan Sun, and Tijana Milenković. Fair evaluation of global network aligners. *Algorithms for Molecular Biology*, 10(19), 2015.
44. Tijana Milenković and Nataša Pržulj. Uncovering biological network function via graphlet degree signatures. *Cancer Informatics*, 6:257–273, 2008.
45. Ryan W Solava, Ryan P Michaels, and T Milenković. Graphlet-based edge clustering reveals pathogen-interacting proteins. *Bioinformatics*, 18(28):i480–i486, 2012.
46. Vesna Memišević, Tijana Milenković, and Nataša Pržulj. Complementarity of network and sequence information in homologous proteins. *Journal of Integrative Bioinformatics*, 7(3):135, 2010.
47. Tijana Milenković, Vesna Memišević, Anand K Ganesan, and Nataša Pržulj. Systems-level cancer gene identification from protein interaction network topology applied to melanogenesis-related interaction networks. *Journal of the Royal Society Interface*, 7(44):423–437, 2010.
48. Nataša Pržulj. Biological network comparison using graphlet degree distribution. *Bioinformatics*, 23:e177–e183, 2007.
49. Tijana Milenković, Jason Lai, and Nataša Pržulj. GraphCrunch: a tool for large network analyses. *BMC Bioinformatics*, 9(70), 2008.
50. Yuriy Hulovatyy, Ryan W Solava, and Tijana Milenkovic. Revealing missing parts of the interactome via link prediction. *PLOS ONE*, 9(3):e90073, 2014.
51. Tijana Milenković, Vesna Memišević, Anthony Bonato, and Nataša Pržulj. Dominating biological networks. *PLOS ONE*, 6(8):e23016, 2011.
52. Sean R Collins, Patrick Kemmeren, Xue-Chu Zhao, Jack F Greenblatt, Forrest Spencer, Frank CP Holstege, Jonathan S Weissman, and Nevan J Krogan. Toward a comprehensive atlas of the physical interactome of *Saccharomyces cerevisiae*. *Molecular Cell Proteomics*, 6(3):439–450, 2007.
53. Kavitha Venkatesan, Jean-Francois Rual, Alexei Vazquez, Ulrich Stelzl, Irma Lemmens, Tomoko Hirozane-Kishikawa, Tong Hao, Martina Zenkner, Xiaofeng Xin, Kwang-Il Goh, et al. An empirical framework for binary interactome mapping. *Nature Methods*, 6(1):83–90, 2009.
54. Stephen F Altschul, Warren Gish, Webb Miller, Eugene W Myers, and David J Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–410, 1990.
55. João Pedro De Magalhães, Arie Budovsky, Gilad Lehmann, Joana Costa, Yang Li, Vadim Fraifeld, and George M Church. The human ageing genomic resources: online databases and tools for biogerontologists. *Ageing cell*, 8(1):65–72, 2009.
56. Roded Sharan and Trey Ideker. Modeling cellular machinery through biological network comparison. *Nature Biotechnology*, 24(4):427–433, 2006.
57. V. Vijayan, V. Saraph, and T. Milenković. Magna++: Maximizing accuracy in global network alignment via both node and edge conservation. *Bioinformatics*, 2015.
58. Fazle E. Faisal, Lei Meng, Joseph Crawford, and Tijana Milenković. The post-genomic era of biological network alignment. *EURASIP Journal on Bioinformatics and Systems Biology*, 2015(1), 2015.

Appendix for:

Simultaneous Optimization of Both Node and Edge Conservation in Network Alignment via WAVE

Yihan Sun^{1,2,3}, Joseph Crawford¹, Jie Tang² and Tijana Milenković^{1,*}

¹Department of Computer Science and Engineering, Interdisciplinary Center for Network Science and Applications, and ECK Institute for Global Health, University of Notre Dame.

²Department of Computer Science and Technology, Tsinghua University.

³Computer Science Department, Carnegie Mellon University.

*Corresponding Author (E-mail: tmilenko@nd.edu)

4.1 Appendix Figures

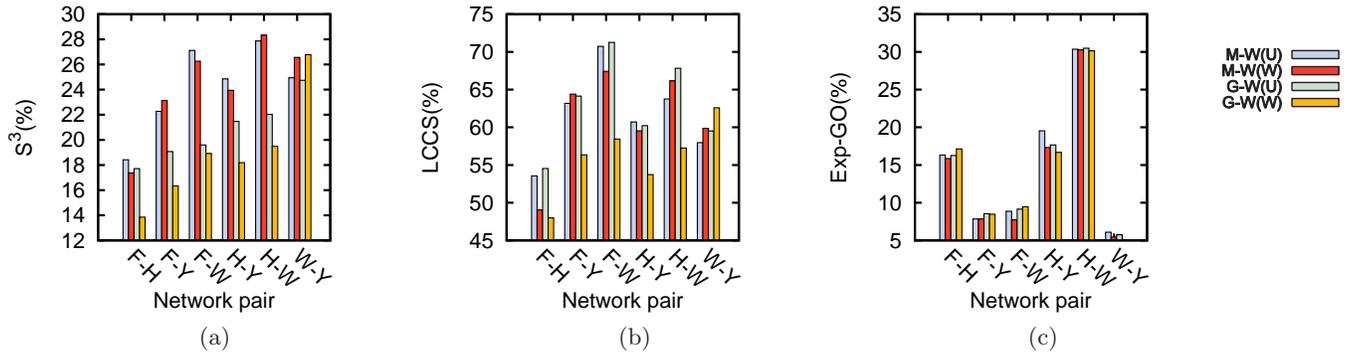


Fig. A.1. Comparison of the edge-weighted and edge-unweighted versions of WAVE on topology-only alignments of real-world PPI networks of different species with respect to (a) S^3 , (b) LCCS, and (c) Exp-GO.

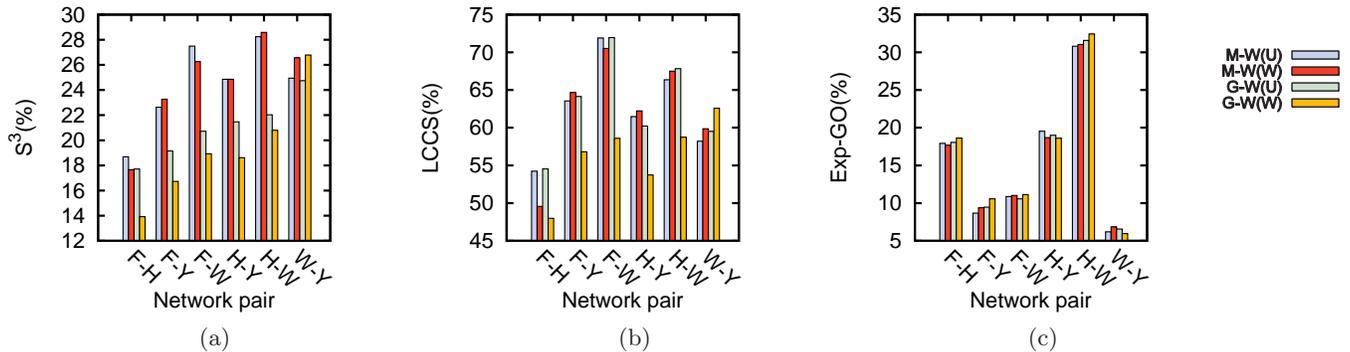


Fig. A.2. Comparison of the edge-weighted and edge-unweighted versions of WAVE on best alignments of real-world PPI networks of different species with respect to (a) S^3 , (b) LCCS, and (c) Exp-GO.

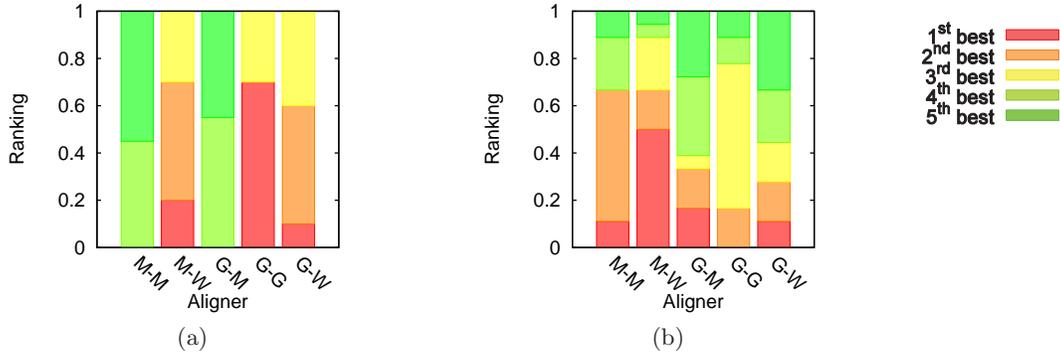


Fig. A.3. Remaining results for overall ranking of each method over all network pairs in a given data set and over all alignment quality measures. The ranking is expressed as a percentage of all cases in which the given method ranks as the k^{th} best method. That is, the more cases in which a given method achieves a higher ranking, the better the method. For example, in panel (b), M-W is the highest scoring of all methods shown on x -axis, since it is ranked the highest (i.e., as the 1st best method) in most of the cases. (a) Results for the five NCF-AS methods on best alignments of “synthetic” (noisy yeast) networks. (b) Results for the five NCF-AS methods on best alignments of real-world PPI networks of different species. Details (per network pair and alignment quality measure) for panels (a)-(b) are shown in Figures A.4-A.5. Recall that M-M and G-G are MI-GRAAL and GHOST.

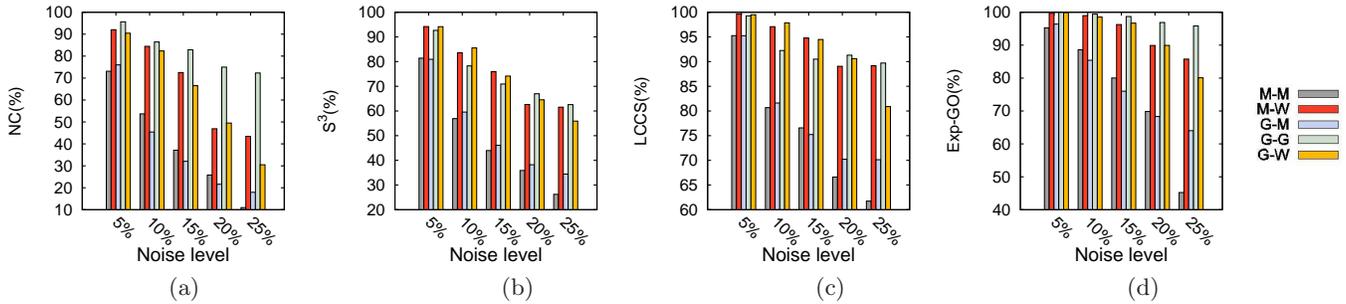


Fig. A.4. Comparison of the five NCF-AS methods on best alignments of “synthetic” (noisy yeast) networks with respect to: (a) NC, (b) S^3 , (c) LCCS, and (d) Exp-GO.

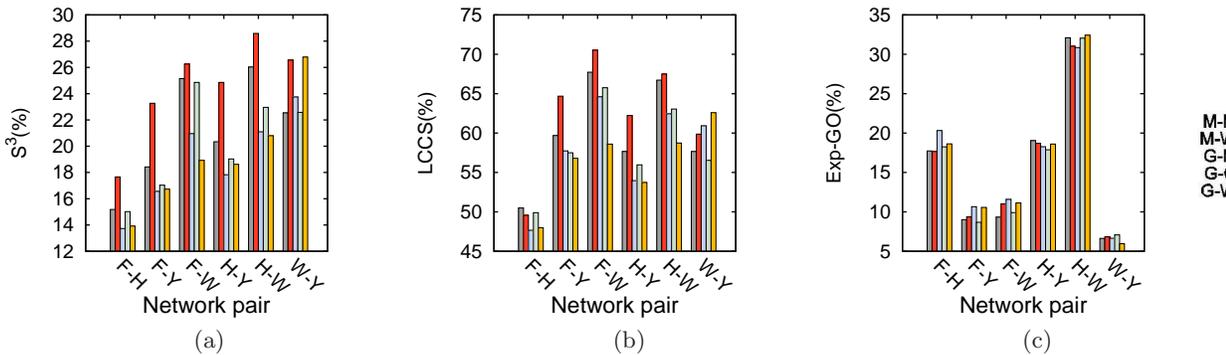


Fig. A.5. Comparison of the five NCF-AS methods on best alignments of real-world PPI networks of different species with respect to: (a) S^3 , (b) LCCS, and (c) Exp-GO.

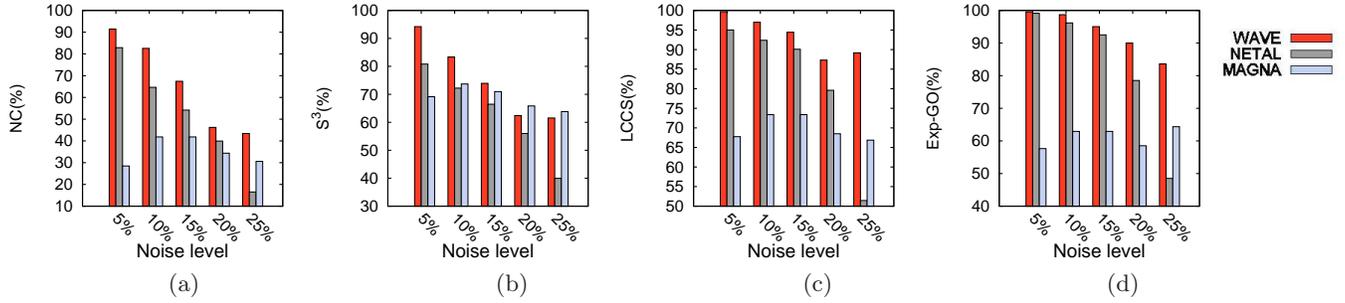


Fig. A.6. Comparison of WAVE (the best of M-W and G-W) with very recent network alignment methods on topology-only alignments of “synthetic” (noisy yeast) networks with respect to: (a) NC, (b) S^3 , (c) LCCS, and (d) Exp-GO.

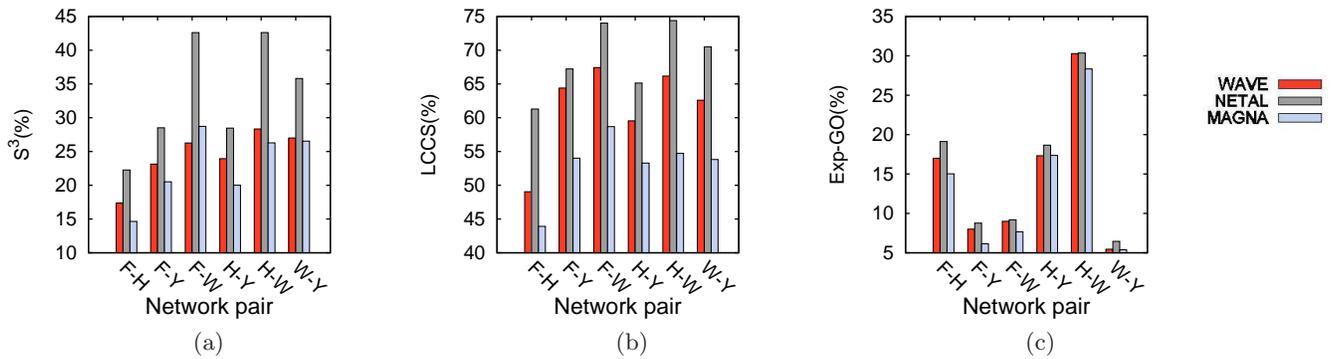


Fig. A.7. Comparison of WAVE (the best of M-W and G-W) with very recent network alignment methods on topology-only alignments of real-world PPI networks of different species with respect to: (a) S^3 , (b) LCCS, and (c) Exp-GO.