Bioinformatics, 31(16), 2015, 2697–2704
doi: 10.1093/bioinformatics/btv170
Advance Access Publication Date: 24 March 2015
Original Paper

OXFORD

Systems biology

Proper evaluation of alignment-free network comparison methods

Ömer Nebil Yaveroğlu[1], Tijana Milenković[2] and Nataša Pržulj[3,*]

[1]California Institute for Telecommunications and Information Technology (Calit2), University of California, Irvine, CA 92697, USA, [2]Department of Computer Science and Engineering, University of Notre Dame, IN 46556, USA and [3]Department of Computing, Imperial College London, London SW7 2AZ, UK

*To whom correspondence should be addressed.

Associate Editor: Igor Jurisica

Received on February 3, 2015; revised on March 7, 2015; accepted on March 18, 2015

Abstract

Motivation: Network comparison is a computationally intractable problem with important applications in systems biology and other domains. A key challenge is to properly quantify similarity between wiring patterns of two networks in an *alignment-free* fashion. Also, *alignment-based* methods exist that aim to identify an actual node mapping between networks and as such serve a different purpose. Various alignment-free methods that use different *global* network properties (e.g. degree distribution) have been proposed. Methods based on small *local* subgraphs called *graphlets* perform the best in the alignment-free network comparison task, due to high level of topological detail that graphlets can capture. Among different graphlet-based methods, *Graphlet Correlation Distance* (GCD) was shown to be the most accurate for comparing networks. Recently, a new graphlet-based method called *NetDis* was proposed, which was claimed to be superior. We argue against this, as the performance of NetDis was not properly evaluated to position it correctly among the other alignment-free methods.

Results: We evaluate the performance of available alignment-free network comparison methods, including GCD and NetDis. We do this by measuring accuracy of each method (in a systematic precision-recall framework) in terms of how well the method can group (cluster) topologically similar networks. By testing this on both synthetic and real-world networks from different domains, we show that GCD remains the most accurate, noise-tolerant and computationally efficient alignment-free method. That is, we show that NetDis does *not* outperform the other methods, as originally claimed, while it is also computationally more expensive. Furthermore, since NetDis is dependent on the choice of a network null model (unlike the other graphlet-based methods), we show that its performance is highly sensitive to the choice of this parameter. Finally, we find that its performance is *not* independent on network sizes and densities, as originally claimed.

Contact: natasha@imperial.ac.uk

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Networks (or graphs) are widely used for representing different types of relational data in the cell, such as protein–protein (Prasad *et al.*, 2009; Stark *et al.*, 2006), genetic (Tong *et al.*, 2004), metabolic (Okuda *et al.*, 2008) and gene regulatory (Hu *et al.*, 2007; Lee *et al.*, 2002) interactions. The information encoded in the wiring patterns (i.e. topology, or structure) of biological networks complements the information obtained from protein sequence and structure (Pevzner and Shamir, 2011). Because of this, graph-theoretic analyses of biological networks can advance our understanding of fundamental cellular functioning.

When analysing biological networks, one needs to compare them. For example, evolutionary insights can be gained by identifying topological similarities between networks of different species (Singh *et al.*, 2008). The difficulty is that network comparison is computationally intractable (Cook, 1971), so heuristic approaches that produce approximate solutions are the only feasible way to compare networks.

Depending on the purpose of the network comparison, relevant approaches can be split into two major sub-categories: (i) alignment-based network comparison and (ii) alignment-free network comparison. *Alignment-based* methods aim to find a mapping between the nodes of two (or more) networks that preserves many edges and a large subgraph between the networks. These methods are useful for identifying the evolutionary conserved parts of biological networks, and they enable the transfer of functional annotations between aligned network regions across species (Faisal *et al.*, 2014; Ibragimov *et al.*, 2013, 2014; Kelley *et al.*, 2003; Kuchaiev and Pržulj, 2011; Liao *et al.*, 2009; Neyshabur *et al.*, 2013; Saraph and Milenković, 2014) and the identification of structural similarities between proteins (Malod-Dognin and Pržulj, 2014; Zhang and Skolnick, 2005). On the other hand, *alignment-free* network comparison methods aim to quantify the overall topological similarity between networks, irrespective of node mappings between the networks, and without intending to identify any conserved edges or subgraphs. These methods have applications in evaluating the fit of a random network model to a real-world network (Hayes *et al.*, 2013; Pržulj, 2007; Pržulj *et al.*, 2004; Rito *et al.*, 2010), tracking the dynamics of time-series networks (Garlaschelli and Loffredo, 2005; Kossinets and Watts, 2006; Yaveroğlu *et al.*, 2014) or grouping (clustering) of networks based on their topological similarities (Milo *et al.*, 2004). The clustering can be used to reconstruct phylogenetic relationships of species based on similarities of their networks (Ali *et al.*, 2014). Alignment-free network comparison methods are typically computationally less expensive than alignment-based methods, and again, they do not produce a node mapping between the compared networks, but a score that quantifies the overall similarity between the two networks. As such, alignment-free and alignment-based network comparison methods have different purposes. Thus, comparing the approaches across the two groups might be misleading.

Of alignment-free network comparison methods, earlier approaches use network properties such as degree distribution, clustering coefficient, diameter (Estrada, 2011; Newman, 2010) and graph spectra (Thorne and Stumpf, 2012; Wilson and Zhu, 2008) for quantifying the overall similarity between two networks. Currently, the best alignment-free network comparison method is based on graphlets, small subgraphs of large networks (Pržulj *et al.*, 2004), called *Graphlet Correlation Distance* (GCD; Yaveroğlu *et al.*, 2014). GCD was systematically compared both with graphlet-based and non-graphlet-based alignment-free predecessors, and it was shown to be the most accurate in clustering topologically similar networks, the most noise-tolerant and the most computationally efficient. Subsequently, another graphlet-based alignment-free method called *NetDis* was proposed (Ali *et al.*, 2014). Although the suggested methodology of NetDis is interesting (Section 2.1), the claimed superior performance of NetDis over the existing state-of-the-art network comparison methods is questionable. This is because the performance of NetDis was not systematically evaluated, so its claimed superiority might be inaccurate. For example, NetDis was not compared against GCD. Also, its comparison against an *alignment-based* method, based on which almost all conclusions of its superiority were drawn, is inappropriate, as argued above. Further potential fallacies with the NetDis method itself exist, such as its dependence on the choice of a network null model, which was not taken into account in the original NetDis study (Ali *et al.*, 2014). Thus, here we systematically and fairly evaluate the performance of NetDis in comparison to other alignment-free network comparison methods, and address all issues present in the paper by Ali *et al.* (2014).

2 Materials and Methods

2.1 Alignment-free network comparison methods

Alignment-free network comparison involves quantifying the overall topological similarity between two networks. As the exact solution is computationally intractable, approximate solutions have been devised for this purpose. Such approximate solutions are conventionally called *network distance* or *network similarity* measures.

2.2 Network distance measures based on global network properties:

The overall similarity between two networks can be quantified in a simple fashion by comparing the networks' global properties, such as the degree distribution, clustering coefficient or diameter (Newman, 2010). The most sophisticated of these network properties are based on graph spectra (Thorne and Stumpf, 2012; Wilson and Zhu, 2008). Although network comparison methods based on global properties are computationally efficient, they usually capture limited aspects of complex wirings of real-world networks. For this reason, it is no surprise that they perform poorly in grouping topologically similar networks together and separating dissimilar networks (Yaveroğlu *et al.*, 2014). Hence, local network properties have been proposed, which can capture the topology of complex networks in more detail.

2.3 Network distance measures based on local network properties

Graphlets are small, connected, non-isomorphic, induced subgraphs of a network (Pržulj *et al.*, 2004). Each graphlet contains 'symmetrical node groups' known as *automorphism orbits* (Fig. 1; Pržulj, 2007). Graphlets can be used to derive detailed descriptors of network topology at network, node and edge level (Milenković and Pržulj, 2008; Solava *et al.*, 2012; Yaveroğlu *et al.*, 2014). By using graphlets in different ways, four different alignment-free network comparison measures are defined:

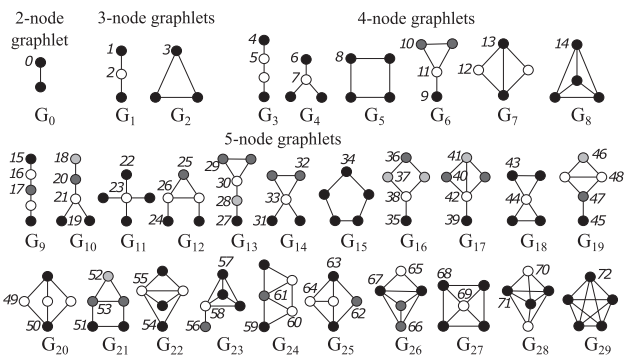

Fig. 1. The thirty 2- to 5-node graphlets and their 73 automorphism orbits (Pržulj, 2007)

1. *Relative graphlet frequency distance (RGFD):* The topology of a network can be described by the number of times that each graphlet appears in the network. RGFD is a non-parametric method that uses the graphlet frequency statistics of 3- to 5-node graphlets to quantify the overall difference between two networks (Pržulj *et al.*, 2004). Given the 29-dimensional graphlet frequency vectors of two networks, RGFD first normalizes these vectors based on the total number of graphlets that appear in the networks, and then, it computes the sum of absolute differences between the normalized graphlet frequencies. The resulting score indicates the topological difference between the two networks.
2. *Graphlet degree distribution agreement (GDDA):* Graphlets are also used to define detailed descriptors of the wiring around a node in a network. Namely, the *i*th *graphlet degree* of a node is the number of graphlets that the node touches at orbit *i*. The *graphlet degree vector (GDV)* of a node is the 73-dimensional vector containing graphlet degrees for the 73 automorphism orbits shown in Figure 1 (Milenković and Pržulj, 2008). Considering GDVs of all nodes in a given network, the degree distribution can be extended into 73 graphlet degree distributions (GDD), where each GDD corresponds to the graphlet degrees of one of the 73 orbits. Given two networks, the non-parametric GDDA method compares these 73 GDDs and quantifies the overall topological similarity between the two networks as an average over all 73 comparisons (Pržulj, 2007). GDDA scores are scaled between 0 and 1, and higher scores indicate better topological similarity.
3. *Graphlet correlation distance (GCD):* Graphlets are Lego-like pieces that assemble with each other at different orientations to build large networks. Exploiting this observation, the complex structure of any network can be summarized into an $n \times n$ *graphlet correlation matrix*, where n is the number of considered graphlet orbits (Yaveroğlu *et al.*, 2014). Each cell of this matrix quantifies the level of dependency between two graphlet orbits in the network. For a given network, the cell values are computed by Spearman's correlation between the corresponding graphlet degrees over all nodes in the network. Then, GCD computes the Euclidean distance between graphlet correlation matrices of two networks (Yaveroğlu *et al.*, 2014). As RGFD and GDDA, GCD is non-parametric and it does not require any network null model for the computation. Different GCD versions exist depending on the orbits that are used for constructing the matrices: (i) *GCD-73* accounts for the complete set of 73 orbits from all 2- to 5-node graphlets and (ii) *GCD-11* accounts for 11 non-redundant orbits of 2- to 4-node graphlets (i.e. orbits 0, 1, 2, 4, 5, 6, 7, 8, 9, 10, 11). In our experiments, we choose to use GCD-11 and GCD-73 rather than other GCD versions because GCD-11 is shown to perform the best in grouping topologically similar networks (Yaveroğlu *et al.*, 2014) and because GCD-73 considers all the orbits of typically used 2- to 5-node graphlets.
4. *NetDis:* This most recent graphlet-based alignment-free network comparison method (Ali *et al.*, 2014) first obtains ego-networks of radius two (i.e. subgraphs induced on the nodes that are in the first and second neighbourhood of a given node) for each node in a given network and computes the number of graphlets in each of the resulting ego-networks. Then, NetDis compares these graphlet counts with the graphlet counts from the same density ego-networks of a 'gold-standard network' (i.e. network null model). It then represents the structure of the given network with a vector containing the sum of the 'centred' graphlet counts of all ego-networks, where the centring is performed by computing the difference between the observed and expected (obtained from the gold-standard network) graphlet counts of the ego-networks. Finally, NetDis computes the distance between two given networks by comparing their vectors of centred graphlet counts. Similar to GCD, NetDis has different versions depending on the size of the graphlets that are used. The current implementation considers 3- or 4-node graphlets, corresponding to *NetDis-3* and *NetDis-4*, respectively. Unlike the three other graphlet-based measures described above, NetDis is parametric, requiring a gold-standard network, which is its major disadvantage, as we show below.

2.4 Method evaluation and comparison

We systematically evaluate the performance of the network distance measures by computing how well they can cluster topologically similar networks generated from the same graph family. We do this by mimicking the established evaluation approach from Yaveroğlu *et al.* (2014). That is, we first generate networks from seven graph families: Erdös–Renyi model (ER; Erdos and Rényi, 1961), ER degree distribution preserving model (ERDD; Newman, 2010), scale-free preferential attachment model (SFBA; Barabási and Albert, 1999), scale-free gene duplication and divergence model (SFGD; Vázquez *et al.*, 2002), geometric random graph model (GEO; Penrose, 2003), geometric model with gene duplication (GEOGD; Pržulj *et al.*, 2010) and stickiness-index based model (STICKY; Pržulj and Higham, 2006). The generated networks contain 1000 and 2000 nodes, and they have edge densities of 0.5% and 1%. We choose these specific values because the selected sizes and densities are in stable regions of the graph families (Hayes *et al.*, 2013). To account for randomness in the network generators, we create 10 networks for each network size, edge density and graph family combination, producing a total of 2 (network size) × 2 (edge density) × 7 (graph families) × 10 (network instances) = 280 networks. For graph families that require a predefined degree distribution, we use networks generated from the preferential attachment (SF) model.

Given the resulting set of 280 networks, we evaluate the network clustering performance of a given network distance measure using a systematic *area under precision—recall curve* (AUPR) framework. That is, a given network pair is in the *True* evaluation set if the two networks are generated from the same graph family and in the *False* set otherwise. For a given distance threshold ϵ, a network pair is considered as a *Positive* sample if the distance between the two networks is $\leq \epsilon$ and as a *Negative* sample otherwise. Then, given a set of network pairs, the Precision-Recall curve is obtained by varying the distance threshold ϵ and computing the precision and recall for each ϵ value:

$$Precision = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}, \quad (1)$$

$$Recall = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}. \quad (2)$$

AUPR summarizes the quality of the classification illustrated with the precision-recall curve into a single value, with the maximum of 1. AUPR can be interpreted as the probability of obtaining a True sample when it is randomly drawn from the Positive sample set at any ϵ threshold. In other words, AUPR represents the average precision of the given network distance measure. Thus, measures achieving higher AUPR scores have better performance, i.e. they more correctly cluster similar networks generated from the same

graph family and separate dissimilar networks generated from different graph families.

To test the effect of network sizes and densities on the performance of a network distance measure, we compute AUPR scores in two different ways: (i) we only consider distances between network pairs that are of the same sizes and densities and (ii) we consider distances between all network pairs, comparing networks of different sizes and densities. While the first approach fairly tests how well a network comparison method distinguishes between different graph families without the bias of the network sizes and densities, the second approach should be taken with more caution, as any observed difference between networks could be due not to the actual differences in network topologies but differences in network sizes and densities. A good network distance measure should be able to easily identify networks generated from the same graph family when the networks with same sizes and densities are considered. An ideal network distance measure should also be able to identify networks generated from the same graph family even if their sizes and densities are different. However, this is a more challenging task for network distance measures. In both scenarios, we expect a good network distance measure to produce high AUPR scores.

To compare noise-tolerance performance of different network distance measures, we repeat the above experiments by rewiring the edges of the 280 synthetic networks at different rewiring rates; namely, we rewire $10\%, 20\%, \ldots, 90\%$ of edges in each of the networks. More specifically, for a network that has $|E|$ edges, a 'k% noisy network' is generated as follows: at each step, three nodes, a, b, c, are chosen randomly with the condition that there is an edge (a, b), but there is no edge (a, c). Edge (a, b) is removed from the network and edge (a, c) is added into the network. This process is repeated $(|E| \times k)/100$ times. Once all of the 280 networks are randomly rewired as described, we compute AUPR scores of the new set of rewired networks. To understand the effect of randomization, we repeat the rewiring and evaluation process 30 times at each rewiring rate k. In all these experiments, a successful network distance measure is expected to produce high average AUPR scores over the 30 random runs at each threshold.

3 Results and Discussion

We systematically compare the performance of all network distance measures (Section 2) to correctly position NetDis among other alignment-free network comparison methods. We answer the following questions that the original NetDis study failed to address. What is the effect of the choice of the gold-standard network (Section 3.1) and of network sizes and densities (Section 3.2) on the performance of NetDis? We compare the accuracy (Section 3.3) and computational running time efficiency (Section 3.4) of NetDis to those of competing methods. Finally, we argue that the biological application of NetDis to phylogeny reconstruction, as designed and carried out in the original study of Ali *et al.* (2014), is scientifically inaccurate (Section 3.5).

3.1 NetDis is highly sensitive to the choice of gold-standard network

NetDis requires a gold-standard network to normalize graphlet counts in the ego-networks of the compared networks (Section 2.1). However, for almost all network comparison tasks, there is no prior information on the structure of the compared networks, and consequently, a well-fitting network null model for these networks is unknown. Note that one of the purposes of network comparison is to produce this information as its *output* rather than using it as *input*. The fact that NetDis assumes a specific gold-standard network as its input and that the same gold-standard network is used for normalizing the graphlet counts of both of the compared networks (which might belong to different network null models and thus require different gold-standard networks) raises serious concerns about the accuracy of NetDis's results, as using different gold-standard networks can lead to very different results (Artzy-Randrup *et al.*, 2004). For this reason, NetDis becomes impractical, as its network clustering performance is highly dependent on the chosen gold-standard network.

To test the effect of the gold-standard network on the results of NetDis, we evaluate its performance by using different gold-standard networks corresponding to different network null models. Namely, we generate gold-standard networks with 5000 nodes and 20 000 edges [as suggested in the original NetDis paper (Ali *et al.*, 2014)] from each of the following seven graph families: ER, ERDD, SFBA, SFGD, GEO, GEOGD and STICKY (Section 2.2). We find that the AUPR scores of NetDis vary for different gold-standard networks with a minimum AUPR difference of 0.25 (Fig. 2A), which means that the network clustering performance of NetDis is highly sensitive to the chosen gold-standard network (Fig. 2). Therefore, the choice on the gold-standard network can have a huge impact on the quality of the network distances obtained by NetDis.

If NetDis was robust to the choice of gold-standard network, it should yield qualitatively the same results for all tests performed using a particular network null model as the gold-standard (Section 2.2). However, this is not the case. In particular, when NetDis uses the same network null model as the gold-standard, the results of its clustering of synthetic networks of the same sizes and densities are not qualitatively the same as the results of clustering of synthetic networks of different sizes and densities. Namely, when clustering synthetic networks of different sizes and densities, NetDis returns the highest AUPR scores when using SFGD network null model as the gold-standard (Fig. 2A and C). This is true for both NetDis-3 and NetDis-4. However, when clustering synthetic networks of the same sizes and densities, NetDis returns the highest AUPR scores when using ERDD or ER network null model as the gold-standard, depending on NetDis version (Fig. 2B and D). Hence, for the same NetDis version, different network null models produce qualitatively the same (the best) results in the two evaluation tests; the evaluation tests differ only in whether the compared input networks are all of the same sizes and densities or not. In other words, the same network null model gives qualitatively different results in the two evaluation tests. This demonstrates that NetDis is highly sensitive to the choice of a gold-standard network.

3.2 NetDis is affected by network sizes and densities

We argue that Ali *et al.* (2014) made an incorrect statement that 'NetDis can correctly separate different random network model types independent of network size and density.' Our results do not support this. When networks of the same sizes and densities are compared, the highest achieved AUPR score is 0.79 with NetDis-3 (Fig. 2B) and 0.9 with NetDis-4 (Fig. 2D). However, when networks of different sizes and densities are also included into the computation, the highest achieved AUPR score is 0.52 for NetDis-3 and 0.59 for NetDis-4. Thus, the performance of NetDis is *not* independent of the network sizes and densities.

3.3 GCD is more accurate than NetDis

Also, Ali *et al.* (2014) did not systematically evaluate the performance of NetDis to position it correctly among other network

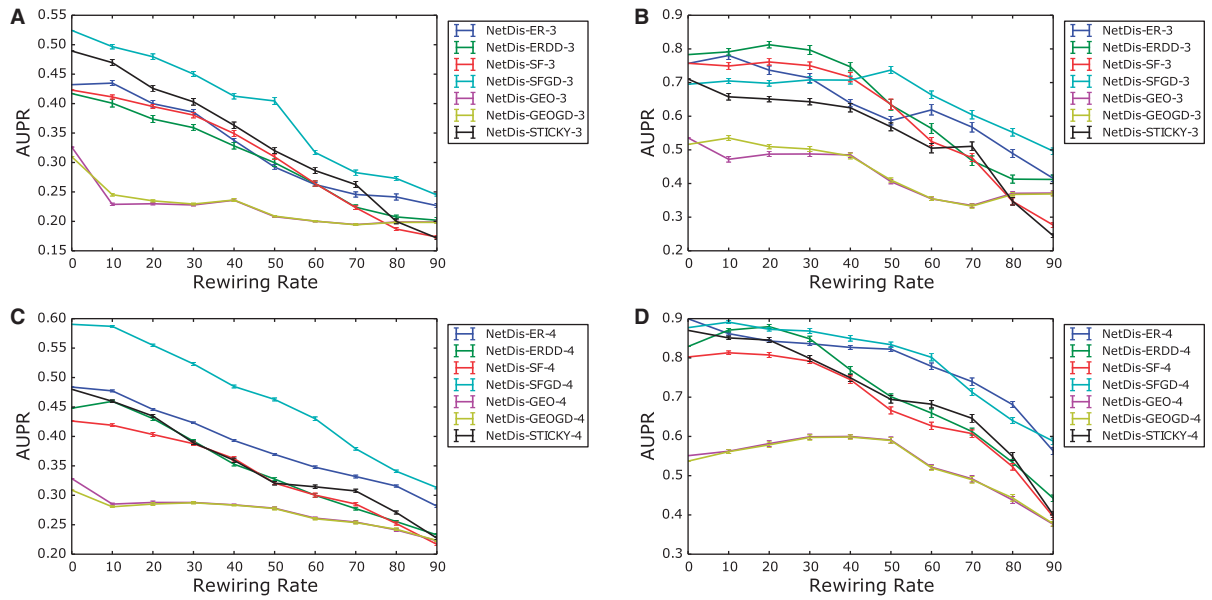

Fig. 2. Performance of different NetDis versions for clustering of networks from different graph families. The plots illustrate the AUPR scores measuring how well NetDis clusters networks generated from the same graph family, for NetDis versions that use different network null models as gold-standard (i.e. ER, ERDD, SF, SFGD, GEO, GEOGD and STICKY) and different graphlet sizes (3-node and 4-node graphlets) to define their network distance measures. 'NetDis-*X*-*Y*' denotes the NetDis version with network null model *X* and graphlet size *Y*. The horizontal axis represents different rewiring rates on the synthetic networks and the vertical axis represents the resulting AUPR scores for these noisy synthetic network sets, when: (**A**) clustering networks of different sizes and densities with NetDis-3 (NetDis with 3-node graphlets), (**B**) clustering only networks of the same sizes and densities with NetDis-3, (**C**) clustering networks of different sizes and densities with NetDis-4 (NetDis with 4-node graphlets) and (**D**) clustering only networks of the same sizes and densities with NetDis-4

distance measures in terms of accuracy. NetDis was only compared to an *alignment-based* network comparison algorithm called MI-GRAAL (Kuchaiev and Pržulj, 2011) and almost all claims about superiority of NetDis were drawn from that comparison. However, the purpose of the *alignment-based* network comparison problem is very different from the purpose of the *alignment-free* methods (Section 1). Thus, it is inaccurate to assess NetDis, an alignment-free approach, in comparison to an alignment-based approach, due to the difference in the goals of the two approach categories. In addition, a number of newer alignment-based approaches have been proposed since MI-GRAAL, including GHOST (Patro and Kingsford, 2012), NETAL (Neyshabur *et al.*, 2013) and MAGNA (Neyshabur *et al.*, 2013). Hence, even if we accept a comparison of alignment-based with non-alignment based methods as a valid evaluation framework (which it is not), NetDis should have also been evaluated against the newer and consequently more accurate alignment-based methods. To *properly* evaluate NetDis, an alignment-free method, one should compare it to the existing state-of-the-art alignment-free methods described in Section 2.1.

GCD has been compared with all of the previous alignment-free methods (Yaveroğlu *et al.*, 2014) and it was shown to be superior when clustering networks generated from the same model. Here, we include NetDis into this existing comparison framework to properly assess its performance against the existing approaches. Because NetDis performs the best with SFGD network null model (Fig. 2), we give NetDis the best-case advantage by using this network null model as the gold-standard. We include RGFD, GDDA, GCD-11 and GCD-73 into this comparison as representatives of other methods based on graphlets (local network properties; Section 2.1). We also include the best alignment-free method that is based on a *global* network property, namely clustering coefficient and exclude the remaining network distance measures that are already evaluated in Yaveroğlu *et al.* (2014) and were shown to perform worse than GCD.

We find that it is GCD (and in particular GCD-11) and *not* NetDis that performs the best in all experimental settings (Fig. 3). GCD-11 is followed by RGFD, GCD-73 and NetDis-4, which have comparable performance with each other. These are then followed by the remaining approaches. As expected, the performance of all methods declines with the increasing levels of noise. However, GCD-11 still performs the best even when 90% of edges in the input networks are rewired. The above results are computed when the best performing network null model (SFGD) is used within NetDis; any of the other network null models would position NetDis even further below the other network distance measures (Fig. 2).

In addition, we perform identical experiments on *real-world* networks from different domains rather than on *synthetic* networks from different random graph models. Hence, we evaluate the performance of the methods on real-world network data. We use real-world networks from 11 different domains (detailed in the Supplementary Material). We find that RGFD, NetDis-SFGD-4 and GCD-73 are the top three methods with the highest AUPR scores, all of comparable performance (Fig. 4). While NetDis-SFGD-4 has the second highest AUPR score, both RGFD and GCD-73 have higher precision values than NetDis-SFGD-4 at all recall values up to ~0.6. This means that, at smaller distance thresholds, both RGFD and GCD-73 identify more true network pairs than NetDis, which is an important property for the early classification task. In addition, the above results are computed when using the best performing network null model within NetDis. When any alternative network null model is used on real-world networks, NetDis never outperforms any of RGFD and GCD-73 (see Supplementary Material).

3.4 GCD remains superior to NetDis in terms of computational efficiency

Typically, global network properties are computationally more efficient to compute than the graphlet-based local properties. However,

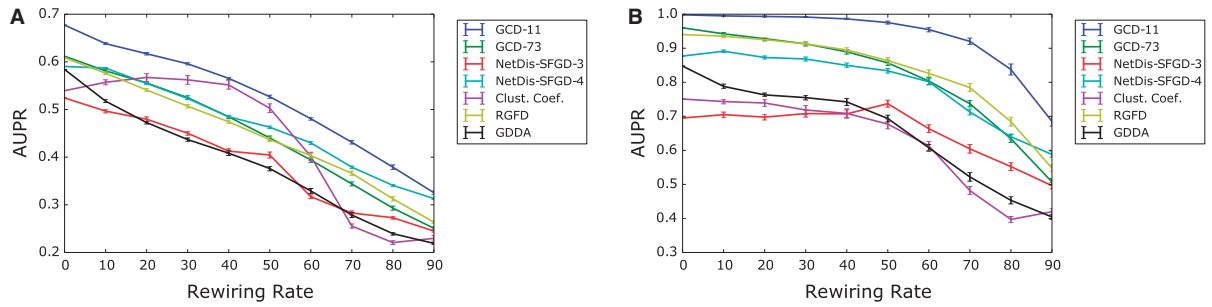

Fig. 3. Performance of different alignment-free network distance measures for clustering networks from different graph families. The plots illustrate the AUPR scores measuring how well different alignment-free network distance measures (i.e. GCD-11, GCD-73, NetDis-SFGD-3, NetDis-SFGD-4, Clustering Coefficient, RGFD and GDDA) cluster networks from the same graph family. The horizontal axis represents different rewiring rates on the networks and the vertical axis represents the resulting AUPR scores for these noisy network sets, when: (**A**) clustering networks of different sizes and densities and (**B**) clustering only networks of the same sizes and densities

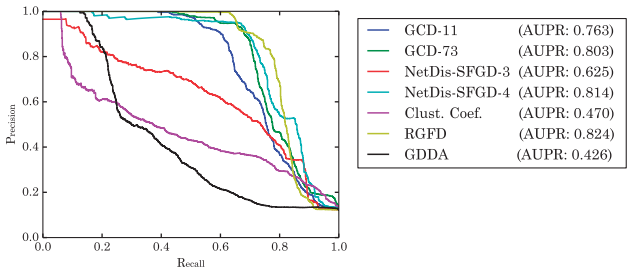

Fig. 4. Performance of different alignment-free network distance measures for clustering of real-world networks. The plot illustrates precision-recall curves of each alignment-free network distance measure (i.e. GCD-11, GCD-73, NetDis-SFGD-3, NetDis-SFGD-4, clustering coefficient, RGFD and GDDA) obtained by clustering real-world networks from different domains. The corresponding Area Under Precision-Recall curve (AUPR) scores of the distance measures are provided in the panel to the right

the graphlet-based methods capture the topology of complex networks in more detail and thus perform better than the best performing global network property, the clustering coefficient (Fig. 3). Due to the high computational complexity of the graphlet counting process, graphlet-based network comparison methods should be designed carefully. Given a network with n nodes, the worst case running time for counting all graphlets and graphlet degrees for 2- to k-node graphlets is $O(n^k)$ and a tighter upper-bound is $O(nd^{k-1})$, where $d \leq n$ is the maximum degree over all nodes in the network (Hočevar and Demšar, 2014). Because most real-world networks are sparse, this computational complexity does not affect the applicability of graphlet-based methods to real-world networks. However, the dependence on the number of nodes in the network should not be ignored; because real-world networks tend to contain thousands to millions of nodes.

Earlier methods that use graphlet properties (i.e. RGFD, GDDA and GCD) are computationally easier to compute after one has already obtained the graphlet counts and GDVs of all nodes in the given networks. Once these counts are computed, the remaining steps of RGFD, GDDA, and GCD computation require low computational times of $O(1)$, $O(n)$, and $O(nln(n))$, respectively (Yaveroğlu *et al.*, 2014). Therefore, their computational bottlenecks lie in the step of producing the graphlet counts, which takes $O(nd^{k-1})$ (see above). Importantly, since GCD-11 requires counting only 2- to 4-node graphlets, its computational complexity is significantly lower than complexities of graphlet-based measures that rely on larger graphlets, corresponding to $O(nd^3)$.

The computational complexity of NetDis is much higher than complexities of the other graphlet-based methods, because NetDis constructs an ego-network of radius two for each of the n nodes and then counts the graphlets within each ego-network separately, where different ego-networks overlap. Due to the overlap, graphlet counting is redundantly done in the same network parts, whereas the other graphlet-based methods simply account for them only once. The complexity of obtaining graphlet counts in all ego-networks is $O(n^2d^{k-1})$. Once these counts are all obtained, the normalization of the subgraph counts and the computation of distances are negligible because their complexities are much lower than the graphlet counting step. Therefore, the computational complexity of NetDis-3 and NetDis-4 is $O(n^2d^2)$ and $O(n^2d^3)$, respectively, which makes these measures more expensive than GCD by an order of $O(n)$.

In addition to the above discussion of *theoretical* running times of different methods, we measure *empirical* running times for NetDis and GCD. Because the implementation of NetDis computes NetDis-3 and NetDis-4 distances in the same run, we represent their running times with a single NetDis measurement. First, we measure the running times by increasing the number of compared networks from 10 to 100 to 1000, where the networks are generated by the preferential attachment model to have 100 nodes and 0.05 edge density. While, as expected, the required running time increases exponentially for all measures with the increase in the number of compared networks, GCD is on average 10 times faster than NetDis (Fig. 5A). Second, we test the effect of the network size on running times. Here, we generate 10 networks from the preferential attachment model of different sizes, containing 100, 500, 1000, 5000 and 10 000 nodes; for each network size, we use the attachment factor of 10, which is the number of nodes that an added node is attached to during network construction. While the running time for GCD is not affected much by the increase in network size, the running time of NetDis increases exponentially with the increase in network size. This is because NetDis needs to compute graphlet counts in each node's ego-network, so it over-counts graphlets (Fig. 5B). Finally, to test the effect of network density on running times, we generate 10 networks from the preferential attachment model, all containing 1000 nodes, but we vary their density by using different attachment factors, namely 1, 5, 10, 20, 50 and 100. Again, while GCD is only slightly affected by network density changes, the running time of NetDis increases exponentially (Fig. 5C). Thus, GCD is computationally more efficient than NetDis, and GCD-11 is the most efficient graphlet-based network distance measure to date.

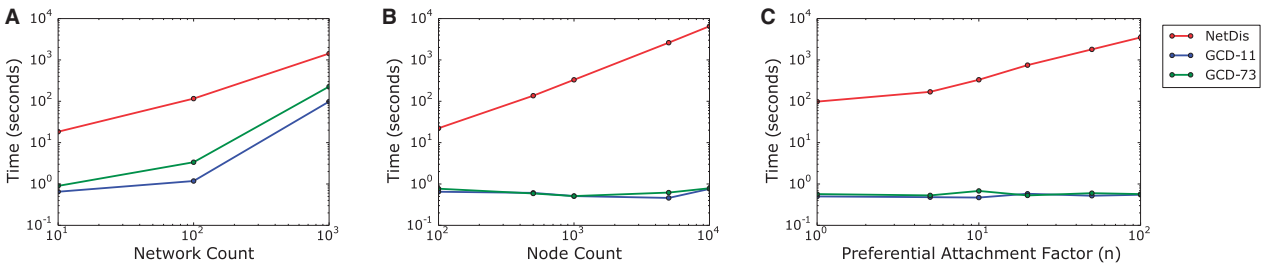

Fig. 5. Performance of GCD and NetDis in terms of computational efficiency. Running times are shown when increasing: (**A**) the number, (**B**) the size (number of nodes) and (**C**) the density of the compared networks. Note that both axes are in logarithmic scale

3.5 Application of NetDis to phylogeny reconstruction is inaccurate

As a potential application of NetDis, Ali *et al.* (2014) compare the protein–protein interaction networks of five species (*Helicobacter pylori*, *Escherichia coli*, fly, human and yeast). Then, they reconstruct the phylogenetic tree of these species by applying average linkage hierarchical clustering on the resulting NetDis distances. We show that this is not a valid evaluation strategy for NetDis (see Supplementary Material for details) because: (i) protein–protein interaction networks are incomplete, which makes the alignment-free comparisons extremely biased, (ii) NetDis produces different hierarchical trees for different parameters (i.e. the number of used graphlets and the gold-standard network) and the reconstructed phylogenetic tree in Ali *et al.* (2014) is a cherry-picked case out of many possible outcomes, (iii) the same phylogenetic tree cannot be reproduced with protein–protein interaction networks that are obtained from different data sources (e.g. BioGRID), (iv) similar phylogenetic trees can be partially reproduced by using very simple network properties such as network density and (v) using five networks only for this purpose might not give enough statistical power to properly evaluate significance of the resulting tree.

4 Conclusion

We systematically, comprehensively and fairly compare available alignment-free network comparison methods, positioning NetDis correctly among other existing approaches. We observe that NetDis, the newest graphlet-based approach for comparing networks, does not perform as well as GCD in clustering networks with similar topologies and it is also computationally more expensive than previous graphlet-based approaches. Furthermore, NetDis is highly sensitive to the choice on the gold-standard network that it uses and this makes it impractical. This is because a well-fitting network null model is hard to determine and differs for most real-world networks and therefore, it is not possible to choose a theoretically well-founded gold-standard network for NetDis comparison of real-world networks, especially because each of the compared networks might require a different gold-standard network. Hence, GCD is still the best performing alignment-free network comparison method to date, which is also highly efficient and not dependent on any network null models. These make GCD a natural choice in alignment-free network comparison.

Funding

This work is supported by the National Science Foundation (NSF) Cyber-Enabled Discovery and Innovation (CDI) [OIA-1028394], European Research Council (ERC) Starting Independent Researcher Grant [278212], ARRS project [J1-5454], the Serbian Ministry of Education and Science Project [III44006], NSF [CCF-1319469] and NSF [CAREER CCF-1452795].

Conflict of interest: none declared.

References

Ali,W. *et al.* (2014) Alignment-free protein interaction network comparison. *Bioinformatics,* **30**, i430–i437.

Artzy-Randrup,Y. *et al.* (2004) Comment on" network motifs: simple building blocks of complex networks" and" superfamilies of evolved and designed networks". *Science,* **305**, 1107–1107.

Barabási,A.L. and Albert,R. (1999) Emergence of scaling in random networks. *Science,* **286**, 509–512.

Cook,S.A. (1971) The complexity of theorem-proving procedures. In: *Proceedings of the Third Annual ACM Symposium on Theory of Computing, ACM*. Shaker Heights, Ohio, USA, pp. 151–158.

Erdos,P. and Rényi,A. (1961) On the evolution of random graphs. *Bull. Inst. Int. Stat.,* **38**, 343–347.

Estrada,E. (2011) *The Structure of Complex Networks: Theory and Applications*. Oxford University Press, Oxford, U.K.

Faisal,F. *et al.* (2014) Global network alignment in the context of aging. *Computational Biology and Bioinformatics, IEEE/ACM Transactions on Computational Biology and Bioinformatics*, Washington DC, USA, pp. 99.

Garlaschelli,D. and Loffredo,M.I. (2005) Structure and evolution of the world trade network. *Phys. A Stat. Mech. Appl.,* **355**, 138–144.

Hayes,W. *et al.* (2013) Graphlet-based measures are suitable for biological network comparison. *Bioinformatics,* **29**, 483–491.

Hočevar,T. and Demšar,J. (2014) A combinatorial approach to graphlet counting. *Bioinformatics,* **30**, 559–565.

Hu,Z. *et al.* (2007) Genetic reconstruction of a functional transcriptional regulatory network. *Nat. Genet.,* **39**, 683–687.

Ibragimov,R. *et al.* (2013) Gedevo: an evolutionary graph edit distance algorithm for biological network alignment. In: *German Conference on Bioinformatics 2013 (GCB'2013)*, Göttingen, Germany, pp. 68–79.

Ibragimov,R. *et al.* (2014) Multiple graph edit distance: simultaneous topological alignment of multiple protein-protein interaction networks with an evolutionary algorithm. *In:* Proceedings of the 2014 Conference on Genetic and Evolutionary Computation (GECCO'2014), Vancouver, BC, Canada, pp. 277–284.

Kelley,B.P. *et al.* (2003) Conserved pathways within bacteria and yeast as revealed by global protein network alignment. *Proc. Natl. Acad. Sci. USA.,* **100**, 11394–11399.

Kossinets,G. and Watts,D.J. (2006) Empirical analysis of an evolving social network. *Science,* **311**, 88–90.

Kuchaiev,O. and Pržulj,N. (2011) Integrative network alignment reveals large regions of global network similarity in yeast and human. *Bioinformatics,* **27**, 1390–1396.

Lee,T.I. *et al.* (2002) Transcriptional regulatory networks in *Saccharomyces cerevisiae*. *Science,* **298**, 799–804.

Liao,C.S. *et al.* (2009) Isorankn: spectral methods for global alignment of multiple protein networks. *Bioinformatics,* **25**, i253–i258.

Malod-Dognin,N. and Pržulj,N. (2014) GR-align: fast and flexible alignment of protein 3D structures using graphlet degree similarity. *Bioinformatics*, **30**, 1259–1265.

Milenković,T. and Pržulj,N. (2008) Uncovering biological network function via graphlet degree signatures. *Cancer Inform.,* **6**, 257–273.

Milo,R. *et al*. (2004) Superfamilies of evolved and designed networks. *Science,* **303**, 1538–1542.

Newman,M. (2010) *Networks: An Introduction.* Oxford University Press, Oxford, U.K.

Neyshabur,B. *et al*. (2013) NETAL: a new graph-based method for global alignment of protein–protein interaction networks. *Bioinformatics,* **29**, 1654–1662.

Okuda,S. *et al*. (2008) KEGG atlas mapping for global analysis of metabolic pathways. *Nucleic Acids Res*., **36**(Suppl. 2), W423–W426.

Patro,R. and Kingsford,C. (2012) Global network alignment using multiscale spectral signatures. *Bioinformatics,* **28**, 3105–3114.

Penrose,M. (2003) *Random Geometric Graphs*. Vo. 5. Oxford University Press, Oxford.

Pevzner,P. and Shamir,R. (2011) *Bioinformatics for Biologists*. Cambridge University Press, Cambridge, U.K.

Prasad,T.K. *et al*. (2009) Human protein reference database 2009 update. *Nucleic Acids Res.,* **37**(Suppl. 1), D767–D772.

Pržulj,N. (2007) Biological network comparison using graphlet degree distribution. *Bioinformatics,* **23**, e177–e183.

Pržulj,N. and Higham,D.J. (2006) Modelling protein–protein interaction networks via a stickiness index. *J. R. Soc. Interface,* **3**, 711–716.

Pržulj,N. *et al*. (2004) Modeling interactome: scale-free or geometric? *Bioinformatics,* **20**, 3508–3515.

Pržulj,N. *et al*. (2010) Geometric evolutionary dynamics of protein interaction networks. In: *Pacific Symposium on Biocomputing,* Vol. 2009, pp. 178–189. World Scientific, The Big Island of Hawaii, Hawaii, USA.

Rito,T. *et al*. (2010) How threshold behaviour affects the use of subgraphs for network comparison. *Bioinformatics,* **26**, i611–i617.

Saraph,V. and Milenković,T. (2014) MAGNA: maximizing accuracy in global network alignment. *Bioinformatics,* **30**, 2931–2940.

Singh,R. *et al*. (2008) Global alignment of multiple protein interaction networks with application to functional orthology detection. *Proc. Natl. Acad. Sci. USA.,* **105**, 12763–12768.

Solava,R. *et al*. (2012) Graphlet-based edge clustering reveals pathogen-interacting proteins. *Bioinformatics,* **18**, i480–i486.

Stark,C. *et al*. (2006) Biogrid: a general repository for interaction datasets. *Nucleic Acids Res.,* **34**(Suppl 1), D535–D539.

Thorne,T. and Stumpf,M.P. (2012) Graph spectral analysis of protein interaction network evolution. *J. R. Soc. Interface*. **9**, 2653–2666.

Tong,A.H.Y. *et al*. (2004) Global mapping of the yeast genetic interaction network. *Science,* **303**, 808–813.

Vázquez,A. *et al*. (2002) Modeling of protein interaction networks. *Complexus,* **1**, 38–44.

Wilson,R.C. and Zhu,P. (2008) A study of graph spectra for comparing graphs and trees. *Pattern Recognit.,* **41**, 2833–2841.

Yaveroğlu,Ö.N. *et al*. (2014) Revealing the hidden language of complex networks. *Sci. Rep.,* **4**, 1–9.

Zhang,Y. and Skolnick,J. (2005) Tm-align: a protein structure alignment algorithm based on the tm-score. *Nucleic Acids Res.,* **33**, 2302–2309.