Systems biology

Temporal network alignment via GoT-WAVE

David Aparício^{1,*}, Pedro Ribeiro¹, Tijana Milenković² and Fernando Silva¹

¹CRACS & INESC-TEC, Departamento de Ciência de Computadores, Faculdade de Ciências, Universidade do Porto, 4169-007 Porto, Portugal and ²Department 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 46556, USA

*To whom correspondence should be addressed. Associate editor: Bonnie Berger Received and revised on January 20, 2019; editorial decision on February 8, 2019; accepted on February 12, 2019

Abstract

Motivation: Network alignment (NA) finds conserved regions between two networks. NA methods optimize node conservation (NC) and edge conservation. Dynamic graphlet degree vectors are a state-of-the-art dynamic NC measure, used within the fastest and most accurate NA method for *temporal* networks: DynaWAVE. Here, we use graphlet-orbit transitions (GoTs), a different graphlet-based measure of temporal node similarity, as a new dynamic NC measure within DynaWAVE, resulting in GoT-WAVE.

Results: On synthetic networks, GoT-WAVE improves DynaWAVE's accuracy by 30% and speed by 64%. On real networks, when optimizing only dynamic NC, the methods are complementary. Furthermore, only GoT-WAVE supports *directed* edges. Hence, GoT-WAVE is a promising new temporal NA algorithm, which efficiently optimizes dynamic NC. We provide a user-friendly user interface and source code for GoT-WAVE.

Availability and implementation: http://www.dcc.fc.up.pt/got-wave/

Contact: daparicio@dcc.fc.up.pt

Supplementary information: Supplementary data are available at Bioinformatics online.

1 Introduction

Network alignment (NA) aims to find conserved regions between networks. NA can be used to transfer knowledge from well- to poorly-studied systems between their conserved regions, e.g. identify topologically similar regions of molecular networks of different species. Pioneering NA methods, such as IsoRank (Singh *et al.*, 2007), align *static* networks, but temporal NA (TNA) is gaining importance as temporal PINs emerge, and other temporal network data becomes prevalent. DynaWAVE is the fastest and most accurate TNA method currently available (Vijayan and Milenković, 2017); it optimizes both *dynamic* node conservation (NC) and edge conservation (EC). As its dynamic NC measure, DynaWAVE uses dynamic graphlet degree vectors (DGDVs) (Hulovatyy *et al.*, 2015).

We recently developed graphlet-orbit transitions (GoTs), a different temporal graphlet measure of node similarity (Aparício *et al.*, 2018). Here, we use GoTs as a new dynamic NC measure within DynaWAVE, since this could lead to a better TNA method (Crawford *et al.*, 2015). We refer to our GoT-modified method as GoT-WAVE. We find that, on synthetic networks, GoT-WAVE is more accurate than DynaWAVE by 30% and faster by 64%. On real networks, using only dynamic NC, GoT-WAVE is more accurate than DynaWAVE on the denser networks and less on the sparser ones. We observe the opposite in terms of their running times. Thus, the two methods are complementary. When combining dynamic NC and EC, DynaWAVE's performance is more enhanced than GoT-WAVE's. However, GoT-WAVE is the only TNA method that supports edge direction.

2 Materials and methods

NA consists of (i) an objective function, typically node conservation (NC) combined with edge conservation (EC), and (ii) an optimization strategy that aims to maximize the objective function. Global NA produces an injection, aiming to maximize NC or EC between aligned node pairs. TNA, extending static NA, aims to optimize dynamic NC or EC. WAVE, DynaWAVE and GoT-WAVE (discussed below) all use the same optimization strategy and maximize objective function $\alpha EC + (1 - \alpha)NC$, where α is a parameter balancing between the two conservation types. What the three methods differ is their scope (i.e. static or dynamic) and their EC and NC measures.

WAVE is a static NA method where EC is the weighted EC (WEC) (Sun *et al.*, 2015) and NC is static graphlet degree vector (GDV) (Pržulj, 2007). WEC is high if many edges are aligned to each other *and* the nodes of the aligned edges are similar with respect to NC. GDVs (Supplementary Figs S1 and S2) have been widely used as topological properties (features) to measure NC in NA (Milenković *et al.*, 2010). WAVE is a seed-and-extend optimization strategy: it first aligns two highly similar seed nodes (with respect to WEC and GDVs), then, the seed's network neighbors that are similar are aligned, the seed's neighbor's neighbors that are similar are aligned.

DynaWAVE is a TNA method where EC is the dynamic WEC (DWEC) and NC is dynamic GDV (DGDV) (Hulovatyy *et al.*, 2015). DynaWAVE is a seed-and-extend method like WAVE. DWEC is a temporal analog of WEC that generalizes an aligned edge to an aligned event (temporal edge) (Vijayan and Milenković, 2017). DGDVs are an extension of GDVs for temporal networks, which yields a measure of dynamic NC.

GoT-WAVE is our proposed TNA method where EC is DWEC and NC is GoTs (Aparício *et al.*, 2018). We perform exact subgraph counting efficiently using g-tries and obtain the GoTs of each node, and use the node's GoTs as its features (see Supplementary Section S1 and Supplementary Fig. S3). The feature vectors over all nodes in a network form a *#nodes* × *#transitions* matrix. When two networks are aligned, we join their respective matrices. Due to high dimensionality and sparsity of the joined matrix we use PCA keeping 99% of its variance. Then, we compute the similarity between all node pairs from different networks as the cosine similarity between the nodes' PCA-reduced features. GoT-WAVE uses these node similarities as the dynamic NC part of the objective function, which is then optimized by WAVE.

Parameters. We use all DGDVs with up to four nodes and six events, as suggested by Hulovatyy *et al.* (2015), and all undirected GoTs with up to four nodes, unless explicitly stated otherwise. Regarding balancing of dynamic NC and EC, we use: $\alpha = 0$, to directly compare the two NC measures, or $\alpha = \frac{1}{2}$, since this α value seems to work the best for DynaWAVE (Vijayan and Milenković, 2017).

3 Results and discussion

We measure the gains in terms of accuracy, running time, node correctness and objective function, as explained in Supplementary Section S2.

We compare GoT-WAVE and DynaWAVE on synthetic networks from different graph models (Supplementary Section S3). A good TNA method should identify networks from the same model as more topologically alike than networks from different models (i.e. yield higher objective function scores). We align all pairs of synthetic networks using GoT-WAVE and DynaWAVE and compute their AUROCs (details in Supplementary Section S4). GoT-WAVE's AUROC is higher than DynaWAVE's by $\approx 30\%$ for both $\alpha = 0$ and $\alpha = \frac{1}{2}$ [Supplementary Table S2(a)]. Extracting GoTs is overall 64%



Fig. 1. GoT-WAVE against DynaWAVE on undirected networks in terms of how well their alignments' objective scores match the objective scores of ideal alignments. *G_O* is the relative gain of GoT-WAVE over DynaWAVE



Fig. 2. GoT-WAVE against DynaWAVE on directed networks in terms of (a) node correctness and (b) how well their alignments' objective scores match the scores of ideal alignments

faster than extracting DGDVs [Supplementary Table S2(b)] while alignment times are similar (Supplementary Table S3).

We analyze eight real networks (Supplementary Table S4). Six are undirected: three biological (zebra, yeast and aging) and three social (arxiv, gallery and school). Due to the lack of directed biological temporal networks, we use two from other fields (emails and tennis). We evaluate TNA on a real network by inserting artificial noise, i.e. rewire a percentage of network's temporal edges (events), and align the original network to the noisy versions. Then, since the aligned networks have the same nodes, we measure the percentage of nodes that are correctly aligned. We use three randomization schemes: *undirected*, *directed* and *pure directed* (details in Supplementary Section S5).

In terms of objective score (Fig. 1 and Supplementary Fig. S4), when $\alpha = 0$: (i) for gallery and zebra, both methods closely match their ideal alignments over all noise levels; (ii) for yeast and aging, GoT-WAVE closely matches its ideal alignment over all noise levels while DynaWAVE drifts from its ideal alignment for high noise levels; and (iii) for arxiv and school, GoT-WAVE closely matches its ideal alignment for high noise levels while DynaWAVE is far from it. Thus, in terms of the total gain Go, GoT-WAVE improves upon DynaWAVE for all six networks. When $\alpha = \frac{1}{2}$, GoT-WAVE matches its ideal alignments more closely for two of the six networks (gallery and aging). In terms of node correctness (Supplementary Fig. S5), for $\alpha = 0$, each method is the best for three of the six networks. For $\alpha = \frac{1}{2}$, DynaWAVE's node correctness improves more substantially than GoT-WAVE's, and is superior for most (though not all) of the networks. We show an example of why that might be in Supplementary Figure S6. In terms of running time [Supplementary Table S6(a)], extracting GoTs is faster than DGDVs for sparse networks (zebra, aging and school) and slower for dense networks (aging, arxiv and gallery) since denser networks induce more GoTs than DGDVs [Supplementary Table S6(b)]. Alignment times are similar (Supplementary Table S7).

On the directed randomization scheme, we find that 3-node directed GoTs are the best for both directed networks (Supplementary Table S8). Thus, we use 3-node directed GoTs (we still use DGDVs with four nodes and six events). In terms of node correctness, for $\alpha = 0$, we observe that GoT-WAVE is better than DynaWAVE for **tennis** over all noise levels, and comparable for **emails** [Fig. 2(a)]. For $\alpha = \frac{1}{2}$, DynaWAVE's node correctness is higher for both networks over most noise levels [Supplementary Fig. S7(b)]. In terms of objective score, for $\alpha = 0$, GoT-WAVE more closely matches its ideal alignments than DynaWAVE for **emails**, and the two are comparable for **tennis** [Fig. 2(b)]. For **tennis**, GoT-WAVE mismatches its ideal alignment at higher noise levels, while DynaWAVE mismatches its ideal alignment over all noise levels. For $\alpha = \frac{1}{2}$, DynaWAVE's performance is again better for both networks [Supplementary Fig. S7(d)]. In terms of running time, results are qualitatively similar to those for undirected networks (Supplementary Table S9). On the pure directed randomization scheme, DynaWAVE cannot differentiate between the networks, as expected, while GoT-WAVE can, since GoTs accounts for edge direction (Supplementary Fig. S8).

4 Conclusion

We propose GoT-WAVE, a new method for TNA. Our results suggest that GoTs are an efficient measure of dynamic NC. While DynaWAVE benefits more from also optimizing dynamic EC, only GoT-WAVE supports directed edges. Future work on better incorporating dynamic EC into GoT-WAVE may yield further improvements. As more real temporal data becomes available, TNA will continue to gain importance.

Funding

This work was supported by FCT (Portuguese Foundation for Science and Technology) [UID/EEA50014/2013] and by the United States Air Force Office of Scientific Research (AFOSR) [YIP FA9550-16-1-0147].

Conflict of Interest: none declared.

References

- Aparício, D. *et al.* (2018) GoT: a fingerprint for temporal network comparison. *PLoS One*, **13**, e0205497.
- Crawford,J. et al. (2015) Fair evaluation of global network aligners. Algorithms Mol. Biol., 10, 19.
- Hulovatyy, Y. *et al.* (2015) Exploring the structure and function of temporal networks with dynamic graphlets. *Bioinformatics*, **31**, i171–i180.
- Milenković, T. *et al.* (2010) Optimal network alignment with graphlet degree vectors. *Cancer Inf.*, 9, 121.
- Pržulj,N. (2007) Biological network comparison using graphlet degree distribution. *Bioinformatics*, 23, e177–e183.
- Singh,R. et al. (2007) Pairwise global alignment of protein interaction networks by matching neighborhood topology. In: Annual International Conference on Research in Computational Molecular Biology. Springer, Berlin, Heidelberg, pp. 16–31.
- Sun,Y. et al. (2015) Simultaneous optimization of both node and edge conservation in network alignment via WAVE. In: Wabi. Springer, pp. 16–39.
- Vijayan, V. and Milenković, T. (2017) Aligning dynamic networks with dynawave. *Bioinformatics*, 34, 1795–1798.