Tangles in networks and their application to describing communities

Supervisor: Stephen Marsland

Michal Salter-Duke

Victoria University of Wellington

August 31, 2020

Introduction

Tangles are graph-theoretic construct that describe highly cohesive regions of a graph - giving essentially the n-connected subgraphs.

My project involves the creation of an algorithm for finding tangles and an attempt to find practical applications for them. One possible application is community detection, and I am testing how well they describe ground truth communities in real networks.

I will define tangles and give some examples, describe my algorithm, then describe community detection and present some results.

Why tangles?

- Each tangle corresponds to a different highly connected region.
- I am investigating whether these highly connected regions correspond to the communities of the network - communities also being highly-connected regions.
- I hope that tangles provide a way to give rigorous descriptions of the topological properties of communities.

What are tangles?

- A graph-theoretic construct that gives essentially the n-connected regions of a graph.
- Developed by Robertson and Seymour as part of their work on graph minors.
- Tangles can be found on any abstract separation system, but this talk will focus primarily on tangles in graphs.
- Composed of a collection of separations of the graph of given order, adhering to certain axioms, oriented in such a way that they *point* to a highly connected region.

[16, 8, 3]

Connectivity functions

Each separation has a specific order, given by a connectivity function. For graphs, there are two common connectivity functions:

Vertex connectivity the number of vertices in the intersection of two sets of edges

Edge connectivity the number of edges between two sets of vertices

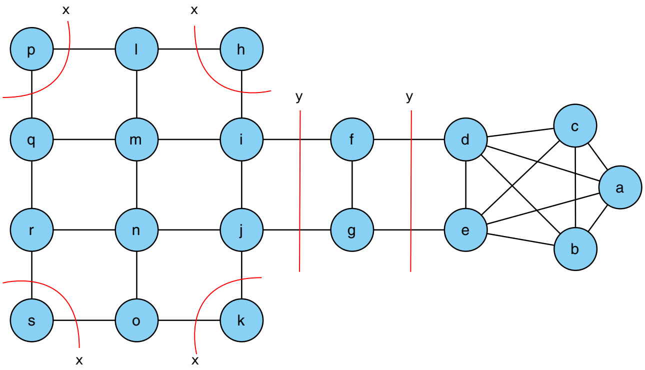
The majority of my work is on edge-connectivity tangles.
I will discuss more general connectivity functions after describing graph tangles.

Separations

- Each separation is a bipartition of some ground set. For vertex connectivity, the ground set is all edges of the graph, and for edge connectivity, the ground set is all the vertices.
- Separations are represented as ordered pairs of "sides" that are oriented in one of two ways: (A, B) or (B, A).
- The union of the sides make up the entire ground set, and the intersection is empty.

Example graph tangle

Figure: An example graph, with order 2 separations shown

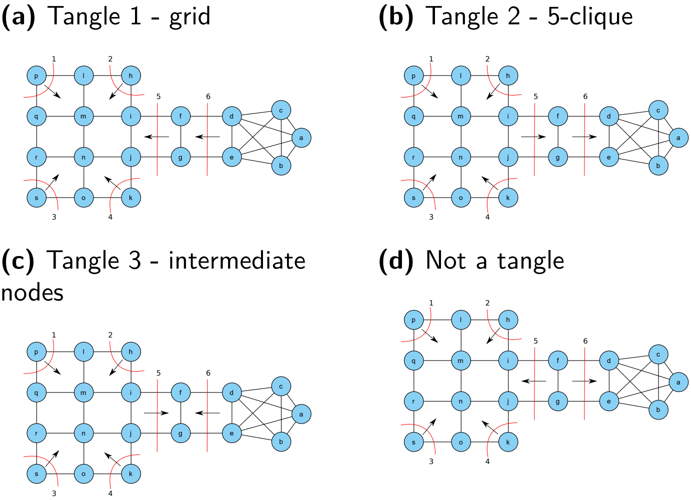


Tangle Axioms

Formally, a tangle $\mathcal{T}$ in graph G of order θ is a set of oriented separations of the ground set of G, all of order $< \theta$, such that the following conditions hold:

- T1 For every separation (A, B) of order $< \theta$, exactly one of (A, B) and (B, A) is in $\mathcal{T}$
- T2 If $(A_1, B_1), (A_2, B_2), (A_3, B_3) \in \mathcal{T}$ then $A_1 \cup A_2 \cup A_3 \neq$ the ground set
- T3 For $(A, B) \in \mathcal{T}, |B| > 1$

Example Tangles



(a) Tangle 1 - grid

(b) Tangle 2 - 5-clique

(c) Tangle 3 - intermediate nodes

(d) Not a tangle

Tangles in other domains

A tangle can be defined on any ground set S with a connectivity function λ that has certain properties. λ must be:

- integer-valued
- symmetric: for $X \subseteq S,\ \lambda(X) = \lambda(S - X)$
- sub-modular: for $A, B \subseteq S$, $\lambda(A \cup B) + \lambda(A \cap B) \leq \lambda(A) + \lambda(B)$

An example - matroids

Matroids are a generalisation of the concept of linear independence. A matroid M is formed on a ground set E, and can be defined in several equivalent ways. One definition has $M = (E, \mathcal{I})$ where $\mathcal{I}$ is the family of subsets of E which are independent.
The exact definition of independence in this description is beyond the scope of this talk, but for concrete examples:

- if E is a set of vectors forming a vector space, then the independence agrees with the standard linear algebra definition of independence.
- if E is the set of edges of a graph, then the independent sets $\mathcal{I}$ are those subsets of edges which do not form cycles.

Tangles in matroids

For $X \subseteq E$, $r_M(X)$ is the rank of X in M and is defined as number of elements in the largest subset of X that is independent in M.
Rank is integer valued and sub-modular but not symmetric, but we can use it do define a connectivity function that is integer valued, sub-modular and symmetric:

$$\lambda_M(X) = r_M(X) + r_M(E - X) + r_M(M) + 1$$

This connectivity function then gives the order of separations of M (formed by bipartitions of E), and these separations can form tangles in the way previously described.
[6]

Computational challenges

- Finding tangles is a computationally expensive operation.
- The number of separations of a graph is potentially huge, particularly for high order, and the combinations of orientations of all the separations is exponential in that parameter.
- This algorithm is complete, but scales very poorly with the size of the graph.
- The aim is to find shortcuts which reduce the search space, or heuristics which are sufficiently accurate for most real scenarios.

Overview of the algorithm

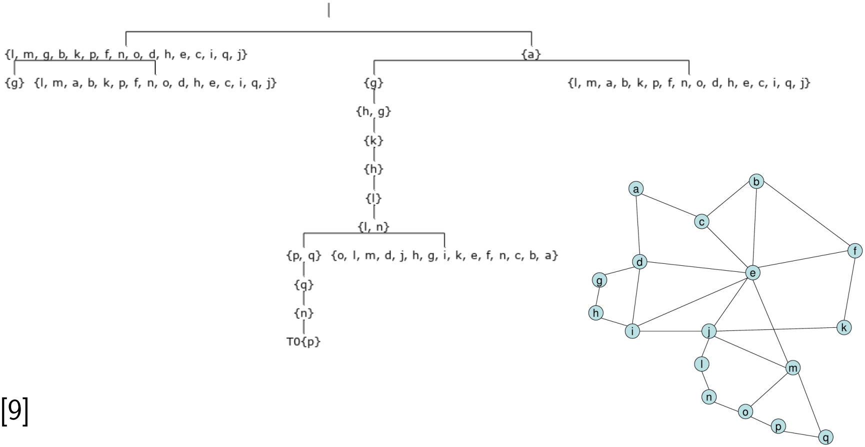
- The algorithm is divided into two parts, with the first, finding the separations, being dependent on the connectivity function, while the second, orienting them into tangles, is independent.
- At present, finding the separations of a given order k involves checking every k set of edges to determine if they separate the graph. If so, the two parts of the graph (A, B) are added to a list of separations.
- Once the list of separations has been built, all the orientations that comply with the tangle axioms are found.

Orienting the separations

- The algorithm creates a tree of tangles of given order. At the start, this tree is empty.
- Then, for each existing tangle in the tree, each separation is tested in turn, to determine whether either orientation complies with the tangle axioms.
- If an orientation of a separation contradicts the axioms, the branch is terminated, and if both orientations of a separation comply with the axioms, the branch bifurcates.

Example tangle tree

(a) Example graph, with the associated tree of separations


{l, m, g, b, k, p, f, n, o, d, h, e, c, i, q, j}
{a}
{g}
{l, m, a, b, k, p, f, n, o, d, h, e, c, i, q, j}
{g}
{l, m, a, b, k, p, f, n, o, d, h, e, c, i, q, j}
{h, g}
{k}
{h}
{l}
{l, n}
{p, q}
{o, l, m, d, j, h, g, i, k, e, f, n, c, b, a}
{q}
{n}
T0{p}
a
b
c
f
d
e
g
h
i
j
k
l
m
n
o
p
q

[9]

Avenues of work

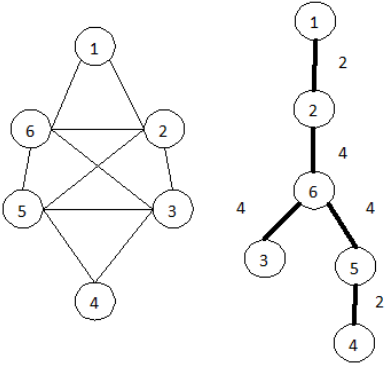
At the moment, the bottleneck in my code is finding all the separations of a graph. Once these are found, orienting them into tangles is relatively quick - certainly for the small graphs I'm currently using. Therefore I'm currently working on methods of finding graph separations.

One promising avenue is utilising a *Gomory-Hu Tree*, which is a weighted tree in which edge weights represent the size of the minimum edge-cuts between every pair of vertices.

[7]

Example Gomory-Hu Tree

(a) Example Gomory-Hu Tree



Fshtea / CC BY-SA (https://creativecommons.org/licenses/by-sa/3.0)

Community Detection

Finding subsets of vertices that are more closely associated with each other than with other parts of the network. Applications:

- Hypothesising gene function based on sharing a community with a gene of known function.
- Targeting disease prevention initiatives to communities at higher risk.
- Identifying websites which share a community with one known to be involved in illegal activities.
- Discovering terrorist cells based on communities found in phone networks.

[12, 5, 11, 17].

Some methods of community detection

- Modularity - based on the concept that there should be more edges between nodes in the same network than expected if edges were placed randomly [14, 13].
- Divisive methods - based on sequentially removing edges based on some criteria, and seeing which parts remain connected. Edge-betweenness is an example criterion [5].

Overlapping communities

The methods just described detect only disjoint communities, whereas for some applications, communities may overlap. Some methods which detect overlapping communities are:

- Line graph methods - a line graph is a new network with a vertex for every original edge, and an edge between vertices if the corresponding original edges are adjacent. A disjoint method is then employed on the line graph. A node of the original graph then inherits the communities of its edges in the line graph. The line graph can be weighted or unweighted [4].
- Clique percolation - based on finding regions composed of small overlapping cliques (completely connected subgraphs) [15].

Parameters for tangle-based community detection

Since a tangle is a collection of oriented separations, potentially representing a community, it is necessary to identify the vertices with communities. We have assigned a vertex to a given tangle community if some proportion (the vertex inclusion threshold, either 0.95 or 1) of the separations are oriented towards that vertex.

A second parameter is the maximum order of tangles to detect. Lower order tangles identify more distinct regions, and higher order tangles can refine these regions. Due to computational issues, the maximum order tested was 6.

Assessing the quality of community detection

Metrics require metadata that reflects the community memberships of each node. In PPIs, we use the Gene Ontology annotations [2, 18]. The quality metrics used are:

Community Quality The average similarity of all pairs of nodes sharing a community divided by the overall average similarity [1]. For PPIs, similarity is Total Ancestry Measure [20], the probability that two proteins share common ancestors in the Gene Ontology.

Normalised Mutual Information (NMI) The mutual information between the GO annotations for each node and the communities it is assigned to [10].

Community Coverage The fraction of nodes that are assigned to at least one non-trivial community (≥ 3 nodes).

Test network A

The algorithm was tested on protein-protein interaction networks (PPIs) representing small parts of the proteome of *Saccharomyces cerevisiae* [19].

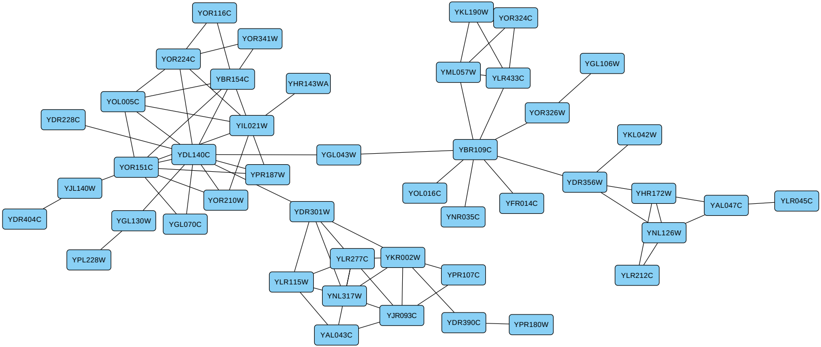


Figure: Schematics of *S. cerevisiae* protein-protein interaction network A.

Test network B

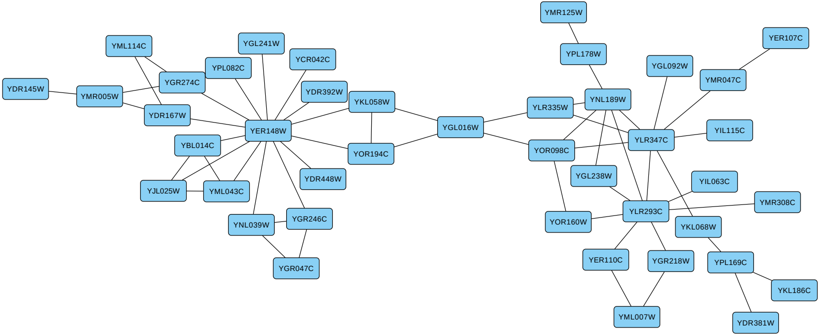


Figure: Schematics of *S. cerevisiae* protein-protein interaction network B.

Sample Results - Network A

Algorithm	# Comms	Similarity	NMI	Coverage
Tangles, o3, t0.95	14	1.4739	0.2256	1
Tangles, o3, t1.0	8	3.1328	0.4276	0.55
Linegraph, EB-W	5	1.5869	0.3415	1.00
Linegraph, Mod-W	6	1.6913	0.2967	1.00
Linegraph, EB-UW	12	1.6143	0.4777	1.00
Linegraph, Mod-UW	5	1.5972	0.2691	1.00
CPM, k = 3	5	3.3709	0.4190	0.49
CPM, k = 4	3	2.3167	0.3745	0.27

Table: Various values of the maximum tangle order and vertex inclusion threshold were tested. Results are only shown for the best performing parameters. The tangle algorithm with order-4 tangles had very similar results, which have therefore been omitted. Clique percolation with clique size 3 gave the best similarity score closely followed by order-3 tangles with vertex inclusion threshold 1.0. These methods assign communities to a much smaller proportion of the vertices than other methods. Of the methods that assign all or most vertices to communities, order-3 tangles with inclusion threshold 0.95 performed relatively poorly.

Sample Results - Network B

Algorithm	# Comms	Similarity	NMI	Coverage
Tangles, o3, t0.95	7	1.2011	0.2592	0.93
Tangles, o3, t1.0	3	0.9596	0.3107	0.57
Tangles, o6, t0.95	20	1.5297	0.2839	0.98
Tangles, o6, t1.0	4	0.9916	0.2953	0.57
Linegraph, EB-W	5	1.5107	0.2934	1.00
Linegraph, Mod-W	7	1.6034	0.3787	1.00
Linegraph, EB-UW	9	1.6454	0.4197	1.00
Linegraph, Mod-UW	5	1.2613	0.2809	1.00
CPM, k = 3	4	2.3158	0.4241	0.38

Table: The NMI was higher for tangles of order 3 than order 6, however similarity was better for order 6, so both are shown. Note that for threshold 1.0, while NMI was better than for several comparison methods, the similarity was particularly low. This appears to be due to a single order 2 tangle which is composed of the entire network excluding leaf nodes, which has lower average similarity than the network itself. Since the network is small with only a few tangles, this low-similarity tangle has a disproportionate effect, particularly at threshold 1 as there are fewer non-trivial tangles. This effect should disappear for larger networks.

Conclusions

The results show promise with results comparable to other methods. This suggests that communities do show some correspondence with tangles, but more work needs to be done. Tangles will never be a viable algorithm for large networks, due to the computational complexity. This work is more focussed on refining the definition of community.
Future work will focus on finding some shortcuts to make the algorithm more efficient, so it can be tested on larger networks, considering different metrics for evaluating the quality, and on considering the theoretical implications of this work for the definition of community.

References I

Ahn, Y.-Y., Bagrow, J. P., and Lehmann, S.
Link communities reveal multiscale complexity in networks.
Nature 466, 7307 (Aug. 2010), 761–764.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., and Sherlock, G.
Gene ontology: tool for the unification of biology. The Gene Ontology Consortium.
Nat. Genet. 25, 1 (May 2000), 25–29.

Diestel, R., Eberenz, P., and Erde, J.
Duality theorems for blocks and tangles in graphs.
SIAM J. Discrete Math. 31, 3 (2017), 1514–1528.

Evans, T. S., and Lambiotte, R.
Line graphs of weighted networks for overlapping communities.
Eur. Phys. J. B 77, 2 (Sept. 2010), 265–272.

Fortunato, S.
Community detection in graphs.
Physics Reports 486, 3-5 (2010), 75–174.

Geelen, J., Gerards, B., and Whittle, G.
Tangles, tree-decompositions and grids in matroids.
J. Combin. Theory Ser. B 99, 4 (2009), 657–667.

Gomory, R. E., and Hu, T. C.
Multi-Terminal network flows.
Journal of the Society for Industrial and Applied Mathematics 9, 4 (1961), 551–570.

References II

Hicks, I. V.
Graphs, branchwidth, and tangles! Oh my!
Networks 45, 2 (Mar. 2005), 55–60.

Hicks, I. V., Koster, A. M. C. A., and Kolotoglu, E.
Branch and tree decomposition techniques for discrete optimization.
Tutorials in Operations Research (2005).

Lancichinetti, A., Fortunato, S., and Kertész, J.
Detecting the overlapping and hierarchical community structure in complex networks.
New J. Phys. 11, 3 (Mar. 2009), 033015.

Mason, O., and Verwoerd, M.
Graph theory and networks in biology.
IET Syst. Biol. 1, 2 (Mar. 2007), 89–119.

Newman, M.
Networks: An Introduction.
Oxford University Press, Oxford, 2010.

Newman, M. E. J.
Modularity and community structure in networks.
Proc. Natl. Acad. Sci. U. S. A. 103, 23 (June 2006), 8577–8582.

Newman, M. E. J., and Girvan, M.
Finding and evaluating community structure in networks.
Phys. Rev. E 69, 2 (Feb. 2004), 026113.

Palla, G., Derényi, I., Farkas, I., and Vicsek, T.
Uncovering the overlapping community structure of complex networks in nature and society.
Nature 435, 7043 (June 2005), 814–818.

References III

Robertson, N., and Seymour, P. D.
Graph minors. X. Obstructions to tree-decomposition.
J. Combin. Theory Ser. B 52, 2 (July 1991), 153–190.

Shakarian, P., Martin, M., Bertetto, J. A., Fischl, B., Hannigan, J., Hernandez, G., Kenney, E., Lademan, J., Paulo, D., and Young, C.
Criminal social network intelligence analysis with the GANG software.
In *Illuminating Dark Networks: The Study of Clandestine Groups and Organizations*. Cambridge University Press, July 2015, pp. 143–156.

The Gene Ontology Consortium.
The gene ontology resource: 20 years and still GOing strong.
Nucleic Acids Res. 47, D1 (Jan. 2019), D330–D338.

Yu, H., Braun, P., Yildirim, M. A., Lemmens, I., Venkatesan, K., Sahalie, J., Hirozane-Kishikawa, T., Gebreab, F., Li, N., Simonis, N., Hao, T., Rual, J.-F., Dricot, A., Vazquez, A., Murray, R. R., Simon, C., Tardivo, L., Tam, S., Svrzikapa, N., Fan, C., de Smet, A.-S., Motyl, A., Hudson, M. E., Park, J., Xin, X., Cusick, M. E., Moore, T., Boone, C., Snyder, M., Roth, F. P., Barabási, A.-L., Tavernier, J., Hill, D. E., and Vidal, M.
High-quality binary protein interaction map of the yeast interactome network.
Science 322, 5898 (Oct. 2008), 104–110.

Yu, H., Jansen, R., Stolovitzky, G., and Gerstein, M.
Total ancestry measure: quantifying the similarity in tree-like classification, with genomic applications.
Bioinformatics 23, 16 (Aug. 2007), 2163–2173.