Edge Thresholding Working Group Report

Tim Dwyer[1] and Patrick Healy[2]

[1] Clayton School of Information Technology Monash University
Wellington Rd, Clayton
Victoria 3800
Australia
tgdwyer@gmail.com

[2] Computer Science Department,
University of Limerick,
Ireland
patrick.healy@ul.ie

Abstract When working with very large networks it is typical for scientists to present a "thinned out" version of the network in order to avoid the clutter of the entire network. For example in the hypothetical case of illustrating trading patterns between groups of nations it might be appropriate to limit the inclusion of inter-nation edges to all those that are significant in terms of their weight but do not, say, associate with a country outside the grouping. Arising from a discussion during one of the introductory sessions we became interested in a problem relating to the discovery of "key events" in a network, in terms of an ordered addition of edges to the network.

1 Background

During one of the introductory presentations at Dagstuhl seminar 08191 Stephen Borgatti described how it was typical for people in the social network community to explore visually a network with weighted edges by not displaying those edges with weight below some threshold. This activity was useful both in understanding the network initially and subsequently in finding the right picture of the network to illustrate a point. Two of the most frequently used features appeared to be *connectedness* and *reachability*.

This process of reducing a network so that only the "interesting" part of the network remains is not new. For example, Batagelj and Zaversnik [2,1] introduce the notion of the *line-cut* of a network. The line-cut of a network $N = (V, L, w)$ with vertex set V, directed edge (line) set L and edge weighting $w : L \to \mathbb{R}$, at a threshold t, is a subnetwork $N(t) = (V[L'], L', w))$ determined by

$$L' = \{e \in L : w(e) \geq t\}.$$

These cuts decompose the network into a set of components the authors call *line islands*. A hierarchy of line islands exist with respect to the value (water level) of t. The authors present an $O(|L| \log |V|)$-time algorithm for computing "maximal regular islands of limited size" [2].

Dagstuhl Seminar Proceedings 08191
Graph Drawing with Applications to Bioinformatics and Social Sciences
http://drops.dagstuhl.de/opus/volltexte/2008/1557

2 Threshold Finding

Our research direction was a) to consider how a user might be assisted in the discovery of interesting thresholds, and b) how the process might automated. A graphical user interface was envisaged comprising a layout window and a slider bar that controlled the display threshold, t. In this way the user could adjust the degree of filtering / eliding of edges, discovering when interesting events would take place. An enhancement of this would be to mark points on the spectrum that correspond to significant features arising in the network with the "sticky" or "snap-to" property that is common in graphical user interfaces.[1]

2.1 Variable vs. Fixed Embeddings

It became apparent that separate issues arose depending on whether the embedding of the network was fixed, or not. Fixed positions of vertices (or slowly changing positions between successive redisplays) is known to be an important factor in assisting the user maintain understanding of an evolving network (*cf.* "mental map" in the literature). If a fixed embedding is required then we might be concerned about when the first edge crossing takes place, or issues of connectivity for spatially defined regions.

On the other hand if the embedding is allowed change then in addition to it being possible to optimize the layout for the visible portion of the network a much richer set of features can be investigated. As we have remarked earlier connectivity and reachability appear to be important features. So, for example, marking where on the spectrum the entire network becomes connected (in the undirected graph sense) or when a pair of vertices from a nominated set become reachable may be useful.

Other features that we considered to be of potential use and worth noting on the slider perhaps were when the graph became:

- disconnected / biconnected
- planar / genus-k (of interest to practitioners?)
- cliques or cycles broken
- clusters (by some definition) become disconnected
- maximum degree is limited to some value
- fractions of total edge count

In the arena of networks with weighted edges *stress majorization* is the method of choice. One of its requirements is a matrix of distances between all nodes, where the length of an edge is the inverse of its weight. In this case we must compute *all-pairs shortest paths* (APSP) dynamically since edges enter and leave the network. We are presently investigating recent algorithms by Demetrescu and Italiano [3]. The authors present an $O(n^2 log^3 n)$-amortized time algorithm for updates and constant-time queries. As we hop from one feature point

[1] One example of this behaviour is when moving a window around the screen: once the window is moved to within ϵ of the screen border it immediately jumps to be aligned with the border.

to another on the spectrum we envisage multiple edge additions or removals; we are investigating whether their algorithms might accommodate a series of k additions or removals in less time than k single updates. Also, their model accommodates edge-weight adjustment, whereas at this stage we see the need for simply adding, or removing, an edge. We intend to consider if an improvement to their algorithm is possible in this restricted setting.

3 Extensions

What we have described above is, we consider, the core system that we wish to continue our work on. However, there are some additional aspects that we have identified that may merit study also. We briefly discuss these now.

Node Thresholds Analogous to edge thresholds we can consider thresholds on nodes, too. Batagelj [1] has considered this previously.

Edge Opacity Instead of strict visibility / invisibility, one might want to control the opacity of below-threshold edges. This is in recognition to the difference between visualisation for exploration versus visualisation for communication: for exploration one may wish to have degrees of visibility whereas for communication or presentation purposes it would appear to be preferable to limit to two states an edge's visibility.

Crossing Cost In the fixed-embedding scenario one may wish to explore the "crossing cost" of the layout. This could be computed as the sum of weights between crossing edges.

Blending Sliders In the case of multi-attribute data one could introduce sliders for each dimension; in addition one could introduce "meta-sliders" that control the blending of attributes in to a single weight.

Use Cases Further consideration should be given to how the tool could be applied to different domains. For example, in a presentation of the Gap Minder tool by Hans Rosling[2] a set of scatter plots, animated to show change over time, is used to explore the relationships between various economic and social statistics across countries. The presentation gives a very compelling view of the interplay between aspects of economic growth and stability such as income and life expectancy and Rosling draws attention to the similarities and differences between first and third world nations. We believe that a network based visualisation of these relationships would help to give a deeper understanding of more complex systems of relationships (for example presenting countries as nodes and edges indicating statistical correlations between them) and would help in both the exploration and presentation of such data.

[2] `http://www.gapminder.org`

References

1. Vladimir Batagelj. (Nonstatistical) Analysis of large networks. In *COSIN Final Meeting*. Tarragona, Spain, March 2005. http://vlado.fmf.uni-lj.si/pub/networks/doc/mix/Cosin05VB.pdf.
2. Vladimir Batagelj and Matjaž Zaveršnik. Islands. In *COSIN Meeting*. University of Karlsruhe, November 2004. http://vlado.fmf.uni-lj.si/pub/networks/doc/mix/IslandsUK.pdf.
3. Camil Demetrescu and Giuseppe F. Italiano. Algorithmic techniques for maintaining shortest routes in dynamic networks. *Elec. Notes in Theor. Comp. Sci.*, 171:3–15, 2007.