Visualize Large-scale Networks with Structural Equivalence

Lei Shi*
Institute of Software
Chinese Academy of Sciences

Qi Liao†
Department of Computer Science
Central Michigan University

Xiaohua Sun ‡
College of Design & Innovation
Tongji University

Yarui Chen §
Department of Computer Science & Technology
Tsinghua University

Chuang Lin ¶
Department of Computer Science & Technology
Tsinghua University

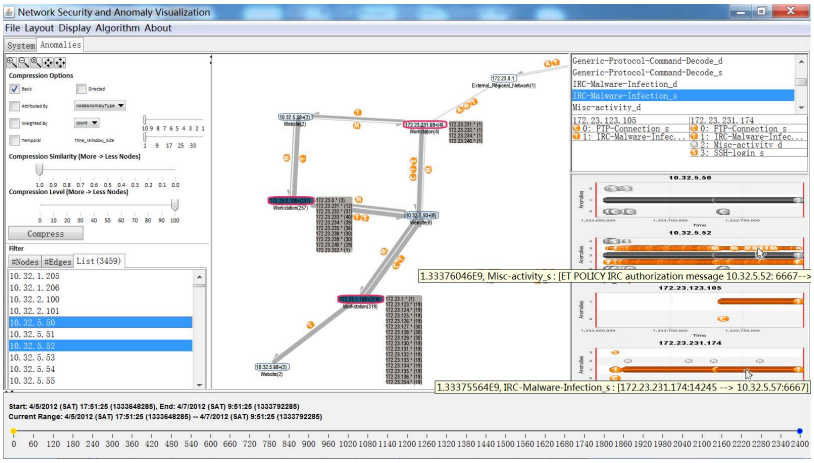


Figure 1: An overview of the Structural Equivalence (SE) based graph anomaly visualization tool. The left panel is the SE grouping options and node/edge selection and filtering; the middle panel is the compress graph view; and the right panel is the timeline plot with network anomaly icons based on the selection of nodes in the graph.

Abstract

As we move into the big data era, the magnitude of inter-connected systems has grown significantly. However, understanding and visualizing such large-scale networks become challenging due to two major reasons. First, rendering extremely large networks with over millions of nodes/edges is infeasibly slow and requires tremendous computing resources. Second, even if it is technically feasible, humans are usually unable to understand the patterns and insights from viewing a smaller graph with only a hundred nodes due to human cognition limitation. Our research targets at reducing the visual complexity of large networks through reducing the graph size while seeking a balance between the information loss and readability. As a supplement to community-based approaches, we apply both strong and relative structural equivalence (SE) to group similar nodes. We have developed interactive visual analytic tools based on SE, and the preliminary results show they are effective in analyzing large graphs.

1 Introduction

The original ARPANET connecting just a few key laboratories in the 1970s has expanded to over four billion interconnected computers. The future Internet will consist of not just computers but also objects (such as smart phones and sensors) or things, known as Internet of Things (IoT). In addition, other types of networks such as social networks and biological networks have also evolved rapidly. For example, one popular social network has exceeded one billion users.

Understanding and visualizing the above large networks and their connection patterns is vital in many research domains, but is very challenging. As computer scientists, we must find efficient ways of visualizing these large-scale networks. Therefore, we ask the question "Is there a way we can reduce the graph size to help human understanding while keeping the key network topological structure intact"? One possible way is to group nodes by clustering or community detection [2]. Though communities can be useful in analyzing networks, the community-level view hides the important context and critical topological details (e.g., edges among the nodes) within the community. Another challenge for community detection is how to group *heterogeneous* nodes, i.e., nodes of different types, in a more meaningful way. We note that many networks are indeed heterogeneous as long as there are at least one semantic attribute on the nodes, e.g., the author-paper-conference network in academic collaborations, the host-user-application network in computer communications, etc.

To that end, we study an alternative node and edge grouping strategy based on the concept of structural equivalence [1,3]. Rather than detecting *proximity*-based communities from a network, structural equivalence (SE) classifies the network nodes into several categories by positions taken in the network, or similar network structures. We implement this idea with a Strong Structural Equivalence (SSE) grouping algorithm completed in linear time for large network traffic graphs. To further control the granularity of visualization, we also develop a fuzzy version called Relative Structure Equivalence (RSE) grouping method according to the similarity of neighbor sets through an interactive control. Nodes with the exactly the same or similar neighbors are rendered as one larger mega-

*e-mail: shil@ios.ac.cn
†e-mail: liao1q@cmich.edu
‡e-mail: xsun@tongji.edu.cn
§e-mail: chenyarui@tsinghua.org.cn
¶e-mail: chlin@tsinghua.edu.cn

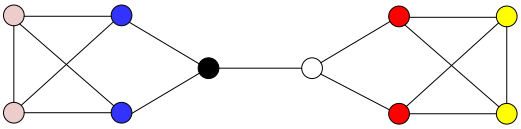


Figure 2: Illustration of Strong Structural Equivalence (SSE) based network abstraction.

node and a network with mega-nodes will regenerated for a compressed version of the original graph for visualization and analysis. An overview of the developed visualization tool based on structural equivalence is shown in Figure 1.

2 Structural Equivalence

Let $G = (V, E)$ be a directed and weighted heterogeneous network. $V = \{v_1, ..., v_n\}$ and $E = \{e_1, ..., e_m\}$ denote the node and link set. The adjacency matrix W can encode bidirectional connections for each node, where $w_{ij} > 0$ indicates a link from v_i to v_j, with w_{ij} denoting the link weight. For each node v_i, $R_i^+ = \{w_{i1}, ..., w_{in}\}$ denotes the outbound vector, $R_i^- = \{w_{1i}, ..., w_{ni}\}$ denotes the inbound vector, both representing its connection pattern. Similarly, $N^+(v_i) = \{j | w_{ij} > 0\}$ and $N^-(v_i) = \{j | w_{ji} > 0\}$ indicate the outbound and inbound neighborhood set. Let $P = \{P_1, ..., P_t\}$ be a partition or grouping over the network G into t sub-group of nodes. $P(v_i)$ indicates the partition index of node v_i.

The Strong Structural Equivalence (SSE) [1] requires the network node to have exactly identical neighborhood set. For any node v_i and v_j in network G, SSE partition network that satisfies:

$$P(v_i) = P(v_j) \Leftrightarrow P_0(v_i) = P_0(v_j) \;\; and \\ N^+(v_i) = N^+(v_j) \;\; and \;\; N^-(v_i) = N^-(v_j) \qquad (1)$$

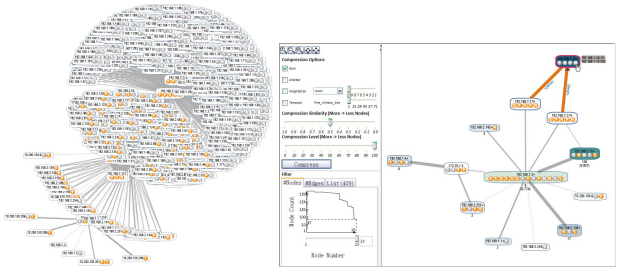
The SSE-based grouping (SSEG) is a deterministic algorithm in that for the same original graph, it always produces the same compressed graph. We implement the SSEG algorithm, which completes in linear time for large network graphs.

In the real scenario of interactive visualization, users may want the flexibility of controlling the compression rate for tradeoff of the visual complexity and precision. We also develop a fuzzy version SE, i.e., Relative Structural Equivalence (RSE). RSE relaxes the requirements of SSE so that we may group nodes with not exactly the same but similar neighbor set. The compression rate can be increased with bounded compensation on accuracy. The key is to define the pairwise similarity score between graph nodes. Here we adopt the standard Jaccard Coefficients between two sample sets A and B for the similarity measure, i.e., $J(A,B) = \frac{|A \bigcap B|}{|A \bigcup B|}$.

Level-of-detail (LOD) control allows users to access more details beyond the compressed graph. By interacting with the graph through clicking on nodes, users may explore the different groupings and switch among SSE, RSE and the original graph. The major gain is to maintain the mental map of users to a certain kind of graph topology. LOD is achieved after the SEG algorithm by re-splitting the aggregated mega-node into smaller mega-nodes of the same size.

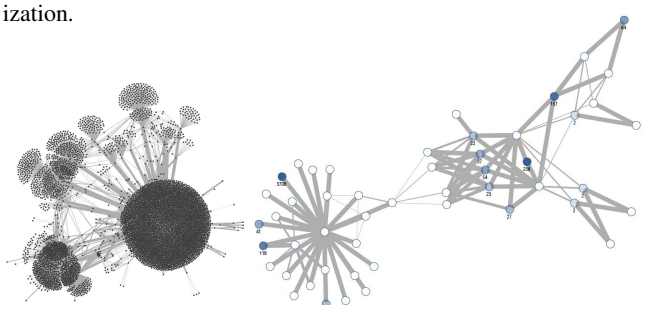
3 Preliminary Results

We evaluate the structural equivalence based visualization on multiple datasets. Figure 3 shows the effect of visualization on the original, uncompressed graph (3(a)) and transformed graph (3(b)). The data is from VAST 2011 Mini Challenge-II dataset which includes a computer network architecture of a shipping company - All Freight Corporation (AFC) and all the necessary traffic data for the tool, including three days of Netflow-like firewall log. We also apply it to VAST 2012 Mini Challenge-II dataset from a financial company's network (Bank of Money) that consists of approximately



(a) Original (b) Compressed

Figure 3: Corporate network traffic overviews from the VAST Challenge 2011 dataset. User interface for SSE-based graph visualization. Left: SSEG controllers. Right: main panel for traffic visualization.



(a) Original (b) Filter to 50 nodes (compressed graph)

Figure 4: Data center flow graph visualization in the original tool, with the node filter and after integrating the SSEG method.

5000 machines. We achieve similar performance in terms of compression rate. Smaller network means reduced visual complexity for the user. The visualization too can easily detect network attacks and anomalies from network traffic data. For example, in Figure 1, it is easy to observe that the IRC traffic exchanged with the websites overwhelms in the whole inspected period. The IRC traffic from the workstations are programmed, with sequentially enumerated source ports. It verifies the hypothesis that these hosts have been compromised as botnet clients. We also experiment the tool on the traffic flow graph among data centers. The traces are collected from a large corporate in the Netflow format, containing statistics of the flows, i.e., timestamp, flow sequence, src/dst IPs and ports, duration, packets, flags, etc. Figure 4(a) shows the original flow graph with 6509 nodes and 18347 edges. A smaller graph containing only 50 nodes is displayed through the node filtering over the compressed graph (4(b)), preserving most of the important topological structure of the original graph.

4 Conclusion

Analyzing large-scale networks is challenging but may have great potential in many research domains. We applied the Structural Equivalence based grouping method to reduce the visual complexity of large network graphs in an effort to seek a balance of amount of details and ease of understanding. Strong and relative structural equivalence methods can effectively reduce the scale of many real-world graphs such as network traffic graphs while still preserving critical topological features of the original graph.

References

[1] F. Lorrain and H. C. White. Structural equivalence of individuals in social networks. *The Journal of Mathematical Sociology*, 1(1):49–80, 1971.

[2] M. E. J. Newman. Fast algorithm for detecting community structure in networks. *Physical Review E*, 69(6):066133, Jun 2004.

[3] D. R. White and K. P. Reitz. Graph and semigroup homomorphisms on networks of relations. *Social Networks*, 5(2):193–234, 1983.