Atomic Operations for Specifying Graph Visualization Techniques

Charles D. Stolper* Southwestern University
Will Price† Southwestern University
Matt Sanford‡ Southwestern University
Duen Horng Chau§ Georgia Tech
John Stasko¶ Georgia Tech

ABSTRACT

While there are a variety of distinct graph visualization techniques described in the visualization literature and used in practice, many of these techniques have similarities. Graph-Level Operations (GLOs) use these similarities as building blocks for specifying graph visualization techniques. We present two GLO-based models, each consisting of a visual element model and an operation set. GLOv2 enables specifying interactive static-graph visualization techniques with multiple displays and DGLOs enables specifying dynamic-graph visualization techniques.

1 INTRODUCTION

There are a large variety of graph visualization techniques, from commonly known and used techniques to those that exist almost exclusively in research literature. In 2014, Stolper et al. presented the Graph-Level Operations model [11] (GLOv1). They showed that by using the pairwise similarities and differences between a set of seed techniques, one could identify a distinct set of atomic manipulations to a visualization that serves as a declarative model to describe graph visualization techniques.

Following the process used by Stolper et al., we have developed two additional models addressing gaps in the earlier work: GLOv2 and DGLOs. Building on GLOv1, GLOv2 enables specifying techniques that use multiple displays, such as small-multiples views (e.g. Attribute Matrix [8] or GraphDice [2]) or coordinated views (e.g. MatrixExplorer [5] and DOSA [12]). DGLOs is a new GLO model designed around dynamic graph visualization strategies such as animation and small-multiple timelines as well as dynamic graph techniques such as 1.5D Egographs [10] and Gestaltmatrices [3]. Javascript+SVG implementations of both models are in active development and are available on GitHub.

2 GLOv2

In order to determine the viability of the GLO concept, we selected the GLOv2 seed techniques through an extensive literature review. We seeded the review with papers containing graph-related terms in their titles from the IEEE Infovis, IEEE Vis/SciVis, and IEEE VAST conferences using the Visualization Publication Dataset [7]. We then added publications on graph visualization techniques referenced by these works and references of those references. In all, we reviewed 430 graph visualization publications.

Through the course of the review, we coded seven categories of techniques: static-graph visualization techniques, tree visualization techniques, directed acyclic graph visualization techniques, dynamic-graph visualization techniques, graph visualization interactions, graph visualization display options, and graph visualization systems. Graph visualization interactions and graph visualization display options (e.g. edge bundling [6]) are not complete techniques, but rather components that can be integrated into an existing technique. Graph visualization systems are applications that implement one or more graph visualization techniques. In the case where those techniques are novel, we coded the system and the technique separately. For example, the Ploceus system [8] utilizes the novel Attribute Matrix technique.

We chose to limit our potential seeds to the 55 techniques coded as static-graph visualization. We address a selection of the dynamic-graph visualization techniques with DGLOs, described below. Of these, we removed any that first reduce the graph to a tree and then visualize that. We then removed those techniques that relied on three-dimensional rendering in order to focus on two-dimensional techniques. Finally, we removed those that used glyphs to represent attributes in addition to nodes and edges. Our final set of 29 graph visualization techniques can be found in Table 1.

Force-Directed Layout	Edge-Label-Centric
Matrix Plot	Honeycomb
Cluster Circles	GraphDice Segment
Circle Graph	3x3 GraphDice
GeneVis A	GMap
GeneVis B	Attribute Matrix
Arc Diagram	EdgeMap A
Matrix Browser	EdgeMap B
Matrix with Bars	Hive Plot
MatrixExplorer	2x3 Hive Panel
NetLens	ScatterNet
Semantic Substrates	Citevis
PivotGraph	DOSA
MatLink	NodeTrix
List View	

Table 1: The 29 GLOv2 seed techniques derived from a review of 430 graph visualization publications. Citations have been excluded for space.

2.1 Model of Visual Elements

As with GLOv1, each graph-level operation in GLOv2 changes an aspect of a visualization element model, such as the positions of glyphs representing nodes or the shape of glyphs representing edges (rectangles for matrix plots or lines for force-directed diagrams). Due to the larger set of seed techniques, we formalized and extended GLOv1's model of visual elements. *Glyphs* (including node glyphs, edge glyphs, and group glyphs) are shapes that are drawn with visual properties that can be set either from constants or based on properties of the node or edge data. Each node glyph is also in one of three interaction modes: no interaction, highlight neighbors, and highlight in-out-neighbors (where in-neighbors and out-neighbors of a selected node are rendered differently). Each edge glyph has one of six interaction modes: show none, show all, show faded, show incident, show in-out, and show faded and incident (where all edges are shown but at a lower visual salience unless they are incident to a selected node). Glyphs are then grouped into *Generations*. A single node generation contains a node glyph for each node of the graph, and GLOv2 adds analogous edge generations. Rather than modifying individual glyphs, operations act upon generations. Node and edge glyphs can be aggregated into generations of *super-nodes*

*e-mail: stolperc@southwestern.edu
†e-mail:pricew@southwestern.edu
‡e-mail:sanfordm@southwestern.edu
§e-mail:polo@gatech.edu
¶e-mail:stasko@cc.gatech.edu

and *super-edges* based on a shared property. Generations can be *cloned* and then manipulated separately in order to have multiple sets of nodes or edges (for example, the left-side and top labels in a matrix plot). Generations are drawn on *Canvases*, each of which has Cartesian and polar axes. Each canvas has axis labels for the four axes that can be shown or hidden. Finally, canvases are drawn on a single *GLO Display*. Canvases are arranged in a grid along the *x* and *y* axes. Similar to how new node and edge generations can be created by cloning existing node and edge generations, new canvases can be created by partitioning existing canvases along the *x* or *y* axis.

2.2 Operations Set

GLOv2 consists of 72 operations, divided into eight categories: adjusting glyph positioning, adjusting glyph visual properties, cloning generations, aggregating glyphs, displaying axes, partitioning canvases*, drawing translucent convex hulls of generations*, and adjusting glyph interaction modes. (The two starred categories are new to GLOv2.) Operations in the partitioning canvases category allow small-multiples views and multiple coordinated views to be described by a graph-level operations model. Operations in the convex hulls category enable drawing regions of node glyphs, necessary for specifying the GMap [4] technique. While the number of seed techniques represented by GLOv2 has increased from 6 in GLOv1 to 29 in GLOv2, the number of operations only increased by a factor of 2.

3 DGLOs

Both GLOv1 and GLOv2 cover static graph visualization techniques. To understand whether dynamic graph visualization techniques have an analogous set of atomic operations, we applied the same methods to a selection of dynamic graph visualization seed techniques.

We used Beck et al.'s survey of Dynamic Graph Vis [1] to identify seed techniques. We hand-selected techniques that covered a breadth of the design space including animated, small-multiple-based, and dynamic graph-specific techniques. We settled on animated and timeline force-directed plots, animated and timeline matrices, animated GMaps [9], Gestaltmatrices [3], and 1.5D Egographs [10].

As with GLOv1 and GLOv2, transitions between the seed techniques dictated the set of operations. The resulting set of 21 operations can be found in Table 2. The DGLOs visual element model is simpler than GLOv2's and consists of node and edge glyphs, timeline canvases (one per time-step in the data), and a DGLO Canvas. DGLOs' positioning is at a higher level of abstraction than in either GLOv1 or GLOv2, using only force-directed and matrix positioning operations. (The latter of which handles duplicating nodes on the left and top of the display.) In DGLOs, all operations affect all canvases. Thus, DGLOs do not currently support multiple coordinated views, only small-multiples. The operations to enable and disable stepping (rather than starting and stopping animation) lets an implementation handle animation using either continuous animation or UI elements such as buttons or sliders.

drawNodeGlyphs	removeTimesteps
drawEdgeGlyphs	positionNodeGlyphsMatrix
drawRegions	positionNodeGlyphsForceDirected
transformNodeGlyphsToLabels	setSelectedNode
transformNodeGlyphsToCircles	fixSelectedNodePositions
transformEdgeGlyphsToSTLines	positionEdgeGlyphSourceTarget
transformEdgeGlyphsToRects	positionEdgeGlyphsMatrix
transformEdgeGlyphsToGestaltGlyphs	setNodeGlyphAttrs
enableStepping	setEdgeGlyphAttrs
disableStepping	setRegionGlyphAttrs
drawTimesteps	

Table 2: The DGLOs operations set, induced from animated and timeline force-directed plots, animated and timeline matrices, animated GMaps [9], Gestaltmatrices [3], and 1.5D Egographs [10].

4 Implementations and Ongoing Work

We are actively developing Javascript+SVG implementations of GLOv2 and DGLOs. These are available on GitHub at http://github.com/chadstolper/glo and http://github.com/chadstolper/dglos, respectively. Moving forward, we plan to build graph exploration applications atop the implementations. We are also exploring using GLO specifications as a vectorization function for machine learning tasks, such recommending techniques based on properties of the graph data.

Acknowledgments

This work was funded by the National Science Foundation under Grants No. IIS-1320537 and IIS-1563816, the XDATA program sponsored by DARPA and the Air Force Research Laboratory (AFRL), and the SCOPE Program at Southwestern University. The content of the information in this document does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

References

[1] F. Beck, M. Burch, S. Diehl, and D. Weiskopf. The State of the Art in Visualizing Dynamic Graphs. In R. Borgo, R. Maciejewski, and I. Viola, eds., *EuroVis - STARs*. The Eurographics Association, 2014. doi: 10.2312/eurovisstar.20141174

[2] A. Bezerianos, F. Chevalier, P. Dragicevic, N. Elmqvist, and J. Fekete. GraphDice: A System for Exploring Multivariate Social Networks. *Computer Graphics Forum*, 29(3):863–872, June 2010. doi: 10.1111/j.1467-8659.2009.01687.x

[3] U. Brandes and B. Nick. Asymmetric Relations in Longitudinal Social Networks. *IEEE TVCG*, 17(12):2283–2290, Dec. 2011. doi: 10.1109/TVCG.2011.169

[4] E. Gansner, Y. Hu, and S. Kobourov. GMap: Visualizing graphs and clusters as maps. In *Visualization Symposium (PacificVis), 2010 IEEE Pacific*, pp. 201–208, Mar. 2010. doi: 10.1109/PACIFICVIS.2010.5429590

[5] N. Henry and J. Fekete. MatrixExplorer: a Dual-Representation System to Explore Social Networks. *IEEE TVCG*, 12(5):677–684, Sept. 2006. doi: 10.1109/TVCG.2006.160

[6] D. Holten. Hierarchical Edge Bundles: Visualization of Adjacency Relations in Hierarchical Data. *IEEE TVCG*, 12(5):741–748, Sept. 2006. doi: 10.1109/TVCG.2006.147

[7] P. Isenberg, F. Heimerl, S. Koch, T. Isenberg, P. Xu, C. Stolper, M. Sedlmair, J. Chen, T. Mller, and J. Stasko. *Visualization Publication Dataset*. 2015. Published: Dataset: http://vispubdata.org/http://vispubdata.org/\, Published Jun.\ 2015.

[8] Z. Liu, S. Navathe, and J. Stasko. Network-based visual analysis of tabular data. In *IEEE VAST 2011*, pp. 41–50, Oct. 2011. doi: 10.1109/VAST.2011.6102440

[9] D. Mashima, S. Kobourov, and Y. Hu. Visualizing Dynamic Data with Maps. *IEEE TVCG*, 18(9):1424–1437, Sept. 2012. doi: 10.1109/TVCG.2011.288

[10] L. Shi, C. Wang, and Z. Wen. Dynamic network visualization in 1.5d. In *2011 IEEE Pacific Visualization Symposium*, pp. 179–186, Mar. 2011. doi: 10.1109/PACIFICVIS.2011.5742388

[11] C. Stolper, M. Kahng, Z. Lin, F. Foerster, A. Goel, J. Stasko, and D. Chau. GLO-STIX: Graph-Level Operations for Specifying Techniques and Interactive eXploration. *IEEE TVCG*, 20(12):2320–2328, Dec. 2014. doi: 10.1109/TVCG.2014.2346444

[12] S. van den Elzen and J. van Wijk. Multivariate Network Exploration and Presentation: From Detail to Overview via Selections and Aggregations. *IEEE TVCG*, 20(12):2310–2319, Dec. 2014. doi: 10.1109/TVCG.2014.2346441