

# TabbyGraph: Interactive Graph Transformation

Fabricio Layedra<sup>1,2\*</sup> Tiffany Wun<sup>2,3†</sup> Gonzalo Gabriel Méndez<sup>1,3‡</sup> Bongshin Lee<sup>4§</sup> Sheelagh Carpendale<sup>2,3¶</sup>

<sup>1</sup>Escuela Superior Politécnica del Litoral <sup>2</sup>Simon Fraser University <sup>3</sup>University of Calgary <sup>4</sup>Microsoft Research

## ABSTRACT

TabbyGraph supports the interactive construction of tabular-based visualizations (e.g., bar charts) from relational data visualization layouts (e.g., node-link diagrams), with the aim of combining graph and chart analysis. TabbyGraph enables agency in manipulating and defining visual elements, while automatically generating those visual elements once defined (e.g., positioning nodes along data attributes). We demonstrate the potential of using TabbyGraph to create scatterplots and bar charts from node-link diagram layouts.

## 1 INTRODUCTION

The overarching goal of this work is to better support multivariate data analysis within the visualization view without requiring prior transformation of the underlying data, using construction techniques which preserves personal agency and transparency in the transformation process. With TabbyGraph, we aim to provide the ability to alter relational-based, force-directed node-link diagrams into charts and plots that are more typical for tabular data. Each type of these layouts contains different advantages based on the tasks at hand. Node-link diagrams show the overall structure—such as hierarchies and relationships—between entities, as well as paths and cycles. Charts and plots can provide easy access to the numerically comparable aspects of the data. Typically, transformation of the underlying data is needed to reveal countable and numerical aspects from relational layouts.

Template editors and shelf-configuration—two of the six main visualization construction approaches [3]—provide a top-down approach to visualization construction; visually mapping all data points before allowing for customization and refinement [7]. While top-down approaches support rapid design of visualizations, the largely automated process obscures the design process. This has been shown to limit peoples’ sense of agency in their interactions with the software, as well as their ability to comprehend the data [2]. Agency refers to one’s sense of control over the system, and has been studied in the context of visualization and interaction [6]. On the other hand, constructing with bottom-up approaches has been shown to promote greater accessibility for non-experts [5].

Hadlak et al. [4] surveyed and categorized different visualization techniques for graph layouts (including multivariate graphs) based on the various data aspects portrayed. GraphDice [1] supports the transformation of graph-based relational layouts into tabular plot layouts using an interface showing all possible combinations of data attributes. TabbyGraph differs by facilitating transformation of the layout within the visualization view using interactive construction and manipulation of data entities with a bottom-up approach.

By allowing manipulation of data entities through interactive construction of tabular layouts, TabbyGraph offers possibilities of

combining graph and chart analysis for flexibly altering layouts to suit the analysis tasks at hand, while maintaining agency and transparency in the construction process.

## 2 TABBYGRAPH

TabbyGraph allows people to interactively construct scatterplots and bar charts from a multivariate graph layout, where vertices have associated attributes  $a_1 \dots a_k$ , and  $k \geq 2$ . This construction takes place within a canvas, where the elements of the node-link diagram can be manipulated directly via drag and drop. By default, TabbyGraph shows a force-directed node-link diagram. To build visualizations that are more typical for tabular data starting from the node-link diagram, TabbyGraph allows graph rearrangement by: (1) working with data attributes; and (2) working with data-driven tools.

### 2.1 Graph Transformation

**Working with Data Attributes.** Within TabbyGraph, data attributes from a graph’s entities and edges are made available for manipulation in the interface as draggable elements. The data attributes can be placed on the left of the canvas to form a vertical axis, or on the bottom of the canvas to create a horizontal axis. These axes display their range, and data type. For example, quantitative attributes show the attribute’s minimum and maximum values to specify the range, with spaced intermediate values. Nominal attributes uniformly distribute the attribute’s values.

**Working with Data-Driven Tools.** Data entities can be manipulated by using data-driven tools. TabbyGraph has two types of data-driven tools: Physics-related tools and Action-related tools.

**Physics-related tools** emulate physics rules such as gravity and attraction forces. *Gravity* is activated by dragging the gravity icon, an apple, onto the canvas. The current notion of “down” is adjustable by specifying a direction vector on the icon. *Attraction* is activated by associating it to an axis’ attribute value. Dropping the force tool on the attribute value will attract the data entities with that value to the corresponding axis tick.

**Action-related tools**, such as position and walls, offer a particular power on the canvas. *Position* is used on placed axes by modifying the location of a data entity to align to its corresponding attribute value on the position-applied axis. *Walls* are represented in the system as lines, and used to contain moving graph nodes, such as in the creation of a bar chart.

### 2.2 Scenarios

We use a fictional dataset in this section, which describes a network of paper authors. Each author entity contains values about the total number of papers and citations, as well as the country of affiliation.

**Scatterplot.** First, the data attribute that will be form the y-axis (i.e., Citations) is chosen and dragged onto the canvas. Then the action tool *Position* is applied to adjust the nodes’ y-position values. Similarly an attribute is chosen for the x-axis (i.e., Papers) and *position* is applied again. Figure 1 shows the formation of a scatterplot, with the number of publications on the x-axis and the number of citations on the y-axis.

**Bar Chart.** To create a tallied bar chart with vertical bars, the *Country* attribute is chosen and placed on the canvas to create an x-axis (Figure 2-1). *Position* is used to align the nodes to the axis, but the nodes’ y-position is not affected. At this stage, if gravity

\*e-mail: fblayedr@fiec.espol.edu.ec

†e-mail: tiffany\_wun@sfu.ca

‡e-mail: gmendez@espol.edu.ec

§e-mail: bongshin@microsoft.com

¶e-mail: sheelagh@sfu.ca

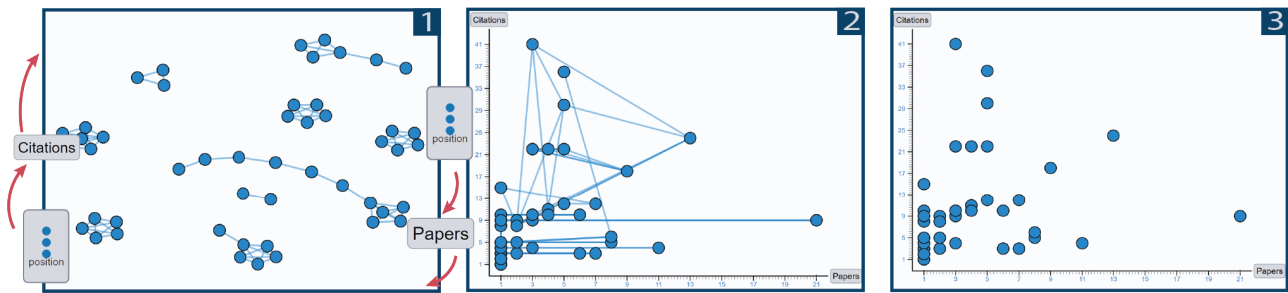


Figure 1: Creating a scatterplot of the number of papers published versus their citation count in TabbyGraph. 1) The default typical force-directed node-link diagram for selected nodes. Dragging the *Papers* data attribute to the bottom right side generates an x-axis, and dragging *Citation* to the top-left generates a y-axis. Using the *Position* force aligns data entities to their respective axes to create 2). 3) Hiding links shows the scatterplot.

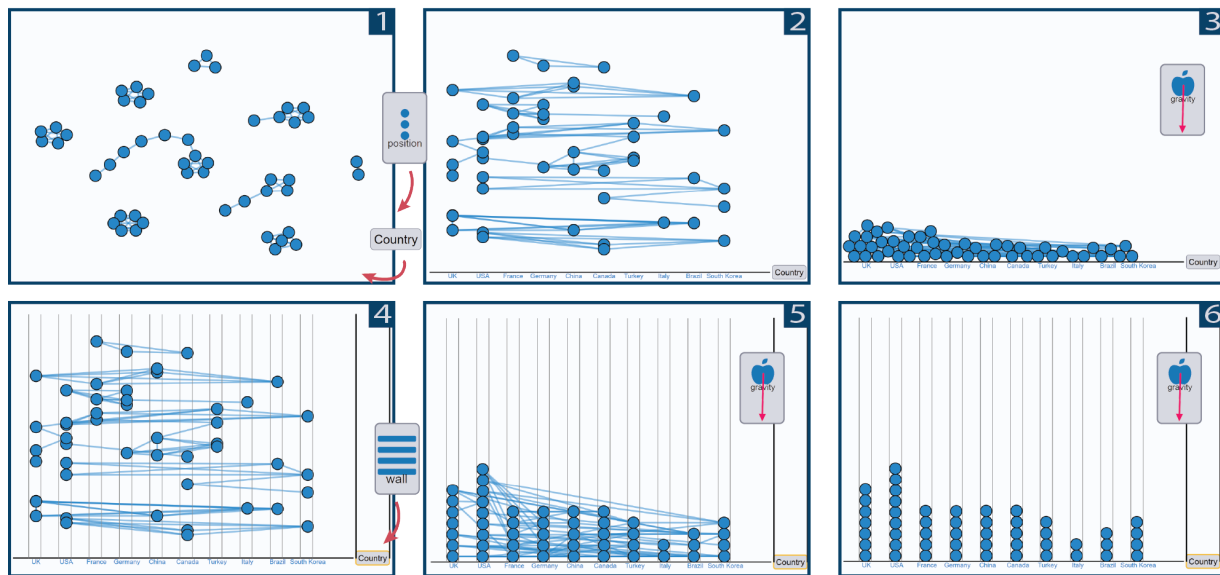


Figure 2: Creating a bar chart of authors by country of institution in TabbyGraph. 1) The *Country* attribute is used to create an x-axis from the starting node-link diagram, and *Position* force aligns the data entities to the x-axis shown in 2). 3) Applying *Gravity* alone makes nodes collapse on each other to form a heap. 4) Applying a *Wall* tool to the *Country* axis in 2) creates containers for the data entities. 5) When *Gravity* forces are applied, the nodes fall neatly within their data attribute value. 6) Hiding the edges then shows the tally bar chart.

is applied, the nodes simply fall down and slide over each other, creating an unorganized heap (Figure 2-3). To organize the nodes so they fall neatly within their data attribute, the *Wall* action tool should first be applied to the X-axis (Figure 2-4), and then *Gravity* (Figure 2-5), creating a tallied bar chart (Figure 2-6).

### 3 CONCLUSION

TabbyGraph introduces an interactive approach for creating scatterplots and bar charts from force-directed node-link diagrams, using the existing data entities and attributes of the graph. To realize this approach, we designed and developed interactions based on surfacing the data entity attributes in the interface. These interactions include variations on physics-related and action-related tools as building blocks for user creations. Our future work includes extending TabbyGraph through the addition of more physics-based and action-based tools. This will allow for the interactive creation of various visualization types based on the manipulation of graph elements and data attributes. We also would like to investigate how people use TabbyGraph to create different types of visualizations.

### REFERENCES

- [1] A. Bezerianos, F. Chevalier, P. Dragicevic, N. Elmqvist, and J.-D. Fekete. GraphDice: A System for Exploring Multivariate Social Networks. In *Computer Graphics Forum*, vol. 29, pp. 863–872. Wiley Online Library, 2010.
- [2] D. Coyle, J. Moore, P. O. Kristensson, P. Fletcher, and A. Blackwell. I did that! Measuring Users' Experience of Agency in their own Actions. In *Proc. CHI 2012*, pp. 2025–2034. ACM, 2012.
- [3] L. Grammel, C. Bennett, M. Tory, and M.-A. Storey. A Survey of Visualization Construction User Interfaces. *EuroVis-Short Papers*, pp. 19–23, 2013.
- [4] S. Hadlak, H. Schumann, and H.-J. Schulz. A Survey of Multi-faceted Graph Visualization. In *EuroVis - STARS*, 2015.
- [5] S. Huron, Y. Jansen, and S. Carpendale. Constructing Visual Representations: Investigating the Use of Tangible Tokens. *IEEE TVCG*, 20(12):2102–2111, 2014.
- [6] P. Koytek, C. Perin, J. Vermeulen, E. André, and S. Carpendale. MyBrush: Brushing and Linking with Personal Agency. *IEEE TVCG*, 24(1):605–615, 2017.
- [7] G. G. Méndez, U. Hinrichs, and M. A. Nacenta. Bottom-up vs. Top-down: Trade-offs in Efficiency, Understanding, Freedom and Creativity with InfoVis Tools. In *Proc. CHI 2017*, pp. 841–852. ACM, 2017.