Hierarchies, Graphs, and Networks (oh my) part two
C. Andrews
Space filling trees

Connections

Containment
What are we trying to discover?

It is all about structure and relationships

- descendants
- ancestors
- child
- sibling
- parent
- grandparent

densesparse

height
depth
Conventional tree visualization

http://www.informatik.uni-koeln.de/ls_juenger/research/vbctool/
Radial layout

http://bl.ocks.org/mbostock/4339607
Hyperbolic browser

The tree is laid out in hyperbolic space and mapped back to the unit circle.

Can show 10x the number of nodes of a standard 2D approach (1000 vs. 100)

Lamping et al. “A Focus + Context Technique Based on Hyperbolic Geometry for Visualizing large Hierarchies”
Treemaps

http://www.cs.umd.edu/hcil/VisuMillion/
Even more approaches...

- Slice-and-dice
- Cluster
- Squarified
- Pivot-by-middle
- Pivot-by-size
- Strip

borrowed from J. Stasko
Radial space filling
Networks

Tree

Network
Graph representations

**vertex and edge tables**

<table>
<thead>
<tr>
<th>edge</th>
<th>source</th>
<th>sink</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A</td>
<td>B</td>
</tr>
<tr>
<td>2</td>
<td>B</td>
<td>C</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>node</th>
<th>attr1</th>
<th>attr2</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>B</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>C</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

**adjacency lists**

A: B
B: A, C
C: B

**adjacency matrix**

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

**Node - link diagram**
Shneiderman’s NetViz Nirvana

Every node is visible

For every node, you can count its degree

For every link you can follow it from source to destination

Clusters and outliers are identifiable
Aesthetic considerations

**Minimize**
- edge crossings
- area
- line bends
- line slopes
- total edge length
- max edge length
- edge length variance

**Maximize**
- smallest angle between edges
- symmetry
Graph visualization

What do nodes look like?

What do links look like?

How do we lay out the graph?
Edge styles

Straight

Orthogonal

Curved
Common layout styles

Hierarchical
Force directed
Circular
Geographic
Clustered
Matrix
Attribute based

http://www.thenetworkthinkers.com/2013/03/big-data.html
Hierarchical graph layout
Sugiyama layout

http://www.roguewave.com/
**Force-directed layout**

**Constraint-based layout**
Force directed layout

http://bl.ocks.org/mbostock/4600693
Circular layout

http://www.perceptualedge.com/blog/?p=680
Circular graph + hierarchical edge bundling
Chord diagram

http://www.facegroup.com/tag/influencers
Source: Uber Blog.

http://bost.ocks.org/mike/uberdata/
Clustered graph layout

Balzer and Deussen, “Level of Detail Visualization of Clustered Graph Layouts”


http://www.personal.psu.edu/lug129/blogs/serendipity/graph-drawing/
MatrixExplorer

MatrixExplorer is a network visualization system that uses two representations: node-link diagrams and matrices. Its design comes from a list of requirements formalized after several interviews and a participatory design session conducted with social science researchers. Although matrices are commonly used in social networks analysis, very few systems support the matrix-based representations to visualize and analyze networks.

MatrixExplorer provides several novel features to support the exploration of social networks with a matrix-based representation, in addition to the standard interactive filtering and clustering functions. It provides tools to reorder (layout) matrices, to annotate and compare findings across different layouts and find consensus among several clusterings. MatrixExplorer also supports Node-link diagram views which are familiar to most users and remain a convenient way to publish or communicate exploration results.

Matrix and node-link representations are kept synchronized at all stages of the exploration process.

Index Terms—social networks visualization, node-link diagrams, matrix-based representations, exploratory process, matrix ordering, interactive clustering, consensus.

Fig. 1. MatrixExplorer showing two synchronized representations of the same network: matrix on the left and node-link on the right.

INTRODUCTION

Information visualization has been used to support social network analysis since the 1930s. Social scientists use visual representations both to explore datasets and to communicate their results. Some information visualization systems focus on exploration, taking advantage of features of the human perceptual system to discern visual patterns in the data. Others help researchers draw social networks, usually in the form of node-link diagrams to represent trees and graphs. Although adjacency matrices have played an important role in social networks analysis since the 1940s [16], few social scientists use their visual representations to communicate their findings.

This article presents MatrixExplorer (Figure 1), which offers both node-link and matrix representations to help sociologists and historians explore and communicate social networks. The node-link diagrams provide intuitive representations for relatively small networks, and, when adequately visualized, remain a powerful means of communication. MatrixExplorer also provides tools for reorganizing, clustering and filtering graphs using a matrix representation. These matrices are always readable, even for large and dense graphs, and thus support exploration throughout the analysis process. MatrixExplorer offers several novel features to help explore complex social networks, using the most suitable representation at any time.

This paper is organized as follows: we first present related work and describe the requirements for a visual exploration system that we defined together with social sciences researchers. We then describe MatrixExplorer and detail its major features for matrix-based representations. We conclude with discussion and future work.

Nathalie Henry is with INRIA Futurs/LRI and University of Sydney, E-Mail: Nathalie.henry@lri.fr.

Jean-Daniel Fekete is INRIA Futurs/LRI, E-Mail: jean-daniel.fekete@inria.fr.

Manuscript received 31 March 2006; accepted 1 August 2006; posted online 6 November 2006.

For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

Henry and Fekete, “MatrixExplorer: a Dual-Representation System to Explore Social Networks”
MatrixExplorer

MatrixExplorer is based on two representations: matrix-based and node-link. Participants agreed that aggregating networks, and insisting in being able to get back to the full data when concerned by the loss of information when dealing with aggregated networks, is a major concern.

R11 - Outliers

R12 - Consensus

R13 - Aggregation

Many participants asked for tools to identify or detect outliers, and the visual representations.

E XPLORER

In social networks analysis, cluster detection or relationships and attributes for actors. Thus, participants asked that the system remind them that filtered data still exists. Alternating the data structure. Therefore, participants asked for tools to synchronize the visualizations in one click.

MatrixExplorer proposes a quick overview of the user workspace. The different visualizations related to a dataset are shown below (overview).

Fig. 3. Visual patterns in MatrixExplorer and Node-link representations of a co-authoring network.

Algorithm: In social networks analysis, cluster detection or relationships and attributes for actors. Thus, participants asked that the system remind them that filtered data still exists. Alternating the data structure. Therefore, participants asked for tools to synchronize the visualizations in one click.

Multiple visualizations are synchronized by selection and filtering, on all visualizations and their overviews. Selection improves the transition from one representation or the other and visualizing the effect of a selection, or representations, accomplishing tasks more easily with one data filtered in one visualization will disappear from all others.

With our system, users explore their networks using both presentations, switching smoothly from one to the other. Node-link and matrix visualizations are synchronized in order to let the user work with both representations, switching smoothly from one to the other.

4.1 Coupling node-link diagrams and matrices

MatrixExplorer is a first attempt to fulfill social sciences researchers' requirements for an exploratory system that allows the user to choose between node-link diagrams and matrices or to choose a third representation. A large panel of automatic methods exists to cluster networks. Users expressed the need to handle several clusterings for a network and to lay out the network size and help present the results. However, they were tired of dealing with a large number of windows and can show/hide components, number of vertex attributes, number of edges, number of connected components, minimum and maximum degree, and in/out degree if the graph is directed. We defined this list with social-science researchers, who specifically pointed out the lack of edge attributes and their labels, the number of vertex attributes that were visible in node-link diagrams, and the correspondence of visual patterns in matrix and node-link diagrams.

With MatrixExplorer, the user first automatically orders the clusters or communities and attributed colors to identify them. He then identifies the global structure of the network. In addition, visualizations can be synchronized by any visual variable. This overview includes for each dataset: general information on the dataset (dataset general information) and analytical information (analytical information) as well as a visual overview of the user workspace.

4.2.1 Datasets and workspace overview

Information appearing for each graph and a visual overview of the workspace overview in order to better visualize how to explore a social network can generate many windows and can show/hide components, number of vertex attributes, number of edges, number of connected components, minimum and maximum degree, and in/out degree if the graph is directed. We defined this list with social-science researchers, who specifically pointed out the lack of edge attributes and their labels, the number of vertex attributes that were visible in node-link diagrams, and the correspondence of visual patterns in matrix and node-link diagrams.
Case Study: VAST 2008 Challenge
Case Study: VAST 2008 Challenge
Longabaugh, “Combing the hairball with BioFabric: a new approach for visualization of large networks”
This section describes the PivotGraph software that allows users to interactively explore roll-ups and selections of multivariate graphs. PivotGraph is a desktop application written in Java. See Figure 5 for a screenshot. (Note that in this and several other screenshots some text was changed in order to mask confidential data.) The PivotGraph interface has three components. A traditional menu bar lets users handle files and change various viewing parameters. At the left is a panel with three parts: two drop-down menus to determine roll-up dimensions for the $x$- and $y$-axes; a legend; and a set of drop-down menus, one for each dimension, that allow the user to specify selection parameters. Finally, the bulk of the screen is devoted to the graph visualization itself.

Visualization

Although the basic idea of the scatterplot representation described in the previous section is simple, it turns out there are a number of subtle challenges that need to be addressed for the visualization to remain legible.

Before diving into the details, it may be helpful to describe the data shown in Figure 5, which is a good example of the visualization at work. The screenshot shows an anonymized view of a real social network within a corporation. Nodes in the graph represent people, and edges represent communication. The graph is rolled up by gender ($x$-axis) and office location ($y$-axis). Several patterns can be seen in the visualization. There is a large amount of cross-gender communication in Location B, for example, but very little elsewhere. Men in Location B seem to be especially central, with women in locations C, D, and E communicating more with them than with men in their own locations. The node sizes provide an indication of how many men and women are at each location, and it is easy to see that in the graph one location (A) has only men.

Layout

Each node is represented by a circle whose $x$- and $y$-coordinates are determined by the current roll-up dimensions. (If there is only one roll-up dimension, then the dots are laid out on a line, as in figure 6.) The area of each circle is proportional to the size variable of the node. In an early version of the program, the sequence of dimension values on the axes was determined simply by alphabetical ordering. A second version of the program rearranged the order of the values to create a more meaningful use of space. To do this for a given dimension, the roll-up of the graph onto that dimension is created, after which a...
PivotGraph

Node and Link Diagram

PivotGraph Roll-up

Wattenberg, “Visual Exploration of Multivariate Graphs”
data. In each case, the users had employed other sophisticated tools for previous analyses. Since the goal of PivotGraph is to help users find new patterns and spot undiscovered features of graph data, one test of its value was to watch experts analyze familiar data to see if they would find previously unknown facts and patterns.

For each of the three pilots, each analyst was given a copy of the program which automatically loaded their data. The author then engaged in a semi-structured interview in which the program was first described, and then the analyst was asked to spend some time using it while they explained what they were looking for and what patterns they were finding. In each pilot a subject found patterns which they had not been aware of before. All users reported a positive response to the tool; while this is potentially due to a novelty effect or desire to please the interviewer, in several cases users took actions that backed up their reports. Specifics are below, although names have been changed and details have been removed for reasons of privacy and confidentiality.

**Pilot 1**

The initial motivation for the creation of PivotGraph came from a project that involved analysis of a transition matrix consisting of 521 states (nodes) and 2,671 transition probabilities (weighted edges). Each state had four associated categorical attributes.

The group studying this data had worked with it for several months and used custom-built database reporting and charting tools for viewing and analysis. In conversations with the author they expressed interest in having a new way of visualizing the data. The first version of PivotGraph was created in response to this interest. Three of the people on the project ran PivotGraph: Allen, a computer scientist; Deanna, a software developer, and Bob, a senior executive. Each was interviewed separately.

Bob, the executive, began trying different x- and y-axis combinations as soon as he was shown the tool. He pointed to several cross-variable trends, including one he termed "weird and interesting." Bob said he had not previously spotted any of these patterns, despite having "stared" at the data before, and said he definitely wanted to use the program again. During the session Bob continued to change the view configuration and made several positive comments on the animated transitions.

Deanna, the developer, said that she saw "totally different" patterns in the data than she had seen before. As the keeper of the database that contained the transition information, she had spent a significant amount of time handling the data. Like Bob, she made many cross-variable comparisons. She too said she would like to use the program again.

When Allen, the computer scientist, began using PivotGraph, he quickly pointed to a series of anomalous transitions. In one roll-up (Figure 9) a certain feature was so implausible that it seemed likely to be caused by a bug in the PivotGraph code. After careful checking, however, it was confirmed that the anomaly really was present in the data—a finding which led to subsequent action on the part of the group. Thus the software had uncovered a significant fact about the data.

**Figure 9.** Anomalous transitions: In this data set, each of the arcs spanning more than two columns represents a surprising data point.

**Pilot 2**

Liz, a social network analyst and researcher, had recently analyzed data on communication patterns in a community of 146 people within a large company. Each person in the community was classified on five dimensions, such as corporate division and geographic location. The network had directed edges, with an edge from person A to person B meaning that person A had reported communicating recently with person B. Liz used both UCINet/NetDraw [6] and Microsoft Excel to study the data. An anonymized screenshot of one view of her data in PivotGraph is shown in Figure 10.

**Figure 10.** Communication network of people in a large company. X-axis is division, y-axis is office geography. The division in the leftmost column has far more cross-location communication than the others.

As with other users, when using PivotGraph she pointed out one and two-variable comparisons as she used the software. Inspecting a roll-up on the company division variable, she said, "There is more communication between Division X and Division Y than between Division X and Division Z."