Graphs and Networks 2

CS 7450 - Information Visualization
October 23, 2013
John Stasko

Review

- Last time we looked at graph layout aesthetics and algorithms, as well as some example applications

- Today we look at more recent InfoVis projects
**Interaction**

- One of the key ways we move beyond graph layout to graph visualization (InfoVis) is interaction with the graph.

**MoireGraph**

- Uses radial layout not terribly unlike hyperbolic tree, but no hyperbolic geometry.
- Impose levels on graph by doing min span tree from some node.
- Put root at center, nodes at subsequent levels further out radially, with decreasing space for each.
- Interaction is key.
Sample Views

Layout Principles

Decreasing exponential space for outer layers

Spreading the “children” nodes
Figure 3: Changing the focus strength. As the focus strength increases, the rest of the graph is pushed to the periphery (left image). Conversely, as the focus strength decreases, more room is allocated to the focus’ children (right image).

Figure 4: Level highlighting. By highlighting a level in a MoreGraph, the space allocated to the level is increased to provide a more detailed look at the level’s visual content.
Navigation and interaction...

![Image](image.png)

Figure 5: Animated Navigation. Selecting a node in a MoireGraph changes the focus. The angular coordinates of a node and the node’s size are interpolated during the animation.

Video

Focus of Graph

- Particular node may be focus, often placed in center for circular layout
- How does one build an interactive system that allows changes in focus?
  - Use animation
  - But intuition about changes not always right
Focus Change Animation

Straight linear interpolation of focus changes not as appealing as changes along polar coordinates

Yee, Fisher, Dhamija, Hearst
InfoVis '01

TreePlus

- Don’t draw entire graph
- Have a focus vertex, then incrementally expand and show connections (min span tree) from there
- Interaction:
  - Single-click: show connections via highlight
  - Double-click: new focus vertex
  - Smooth animated change in focus
- “Plant a seed and watch it grow”  

Lee et al  
TVCG’06
TreePlus

Jigsaw’s Graph View

Don’t draw everything, but allow the viewer to interactively explore (expand & compress) the graph.

Stasko, Görg & Liu
Information Visualization ’08
Recent Trends in GraphViz

- Attributes of nodes influence geometric positioning
  - Not just some arbitrary layout
- Utilize graph statistical analysis too
- Largely driven by interest in social network analysis

PivotGraph

- Cluster on common node attributes
  - Put all A’s together, all B’s together, ...
- “Roll up” nodes
  - Draw edge from A to B depending on how many edges from some A to some B
- Position nodes into a grid based on attributes
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.
Semantic Substrates

• Group nodes into regions
  According to an attribute
  Categorical, ordinal, or binned numerical

• In each region:
  Position nodes according to some other
  attribute(s)

• Give users control of link visibility

Shneiderman & Aris
TVCG (InfoVis) '06
Vizster

- Visualize social networking sites like friendster, myspace, facebook
- Implementation
  - Crawled 1.5 million members (Winter 2003)
  - Written in Java using the prefuse toolkit
    (http://prefuse.sourceforge.net)
- Oppose Shneiderman’s mantra. Instead:
  “Start with what you know, then grow.”

Heer & boyd
InfoVis '05
Visualization

Combining Features

Colors: Gender

Halo: Search for “student”

Highlight: Friends of selection

Blobs: Communities
SocialAction

- Combines graph structural analysis (ranking) with interactive visual exploration
- Multiple coordinated views
  - Lists by ranking for analysis data
  - Basic force-directed layout for graph vis

Perer & Shneiderman
TVCG (InfoVis) '06
Social Network Attributes

- **Bary center** – total shortest path of a node to all other nodes
- **Betweenness centrality** – how often a node appears on the shortest path between all other nodes
- **Closeness centrality** – how close a node is compared to all other nodes
- **Cut-points** – the subgraph becomes disconnected if the node is removed
- **Degree** – number of connections for node
- **HITs** – “hubs and authorities” measure
- **Power centrality** – how linked a node is to rest of network

Attribute Ranking

- Run these measures on all nodes and rank them
- Sort the rankings and show in lists and scatterplots
- Allow user to filter based on rankings
- Can aggregate rankings for cohesive subgroups of nodes
Graph Visualization

- Standard node-link
- Node positions remain constant across different metric views to promote comprehension
- Links can have types
- Coherent subgroups can be aggregated (like in Vizster)
  - Uses Newman’s community identification algo

Users begin with an overview of the entire social network. On the left side, overview statistics that describe the overall structure are presented. On the right, the network is visualized using a force directed algorithm.
The gatekeepers are found using a statistical algorithm. Users filter out the unimportant nodes using a dynamic slider which simplifies the visualization while maintaining the node positions and structure of the network.

Labels are always given priority so users can understand what the data represents. When user selects a node, neighbors are highlighted and details appear on the left. In order to protect sensitive information, node labels have been anonymized except for those individuals publicly identified in the Zacarias Moussaoui trial.
Senate Voting Patterns

http://www.cs.umd.edu/hcil/socialaction/

Implementation

- Jung
  - Network data structures and algorithms
- Prefuse
  - Graph drawing
- Piccolo
  - Scatterplot and Matrix views
Comments

• One of my favorite recent InfoVis papers
• Not too innovative on the vis technique side, but wonderful application and synthesis of useful capabilities
• Actually, a very nice visual analytics example
• Good subsequent paper on case studies evaluation of it (on our later Eval day)

PNNL’s Graph Vis Work

• Graph Signatures
• Goal is to characterize the different styles of nodes in graph based on their local connectivity patterns

Wong et al
TVCG ’06
Compute Signature

1. Run BFS from each node
2. Count how many nodes are 1, 2, 3,... steps away
   That is node's signature
   (3-d is recommended)
3. DO MDS to project into 2D scatterplot
4. Run k-means to detect different clusters. (9 is recommended)

Example 1

1. A hierarchy graph or a tree, b) A graph with two loops.
2. A scatterplot generated by projecting the 2-degree node signatures that represent the local topology of graph nodes in Figure 1 onto a 2D space.
3. Results of K-mean on the scatterplot shown in Figure 4 where k = a) 2, b) 3, and c) 4.
Example 2

Figure 6: A force-directed layout of GD068.

Figure 7: A 2D scatterplot generated by classical MDS using the signature vectors extracted from GD068.

Figure 8: Nodes that share the same color belong to the same cluster identified in Figure 7. Eight signatures (represented as bar graphs) are selected to highlight the general topology of the seven clusters. Notice the previously missing organizer (node 9 in red) hidden among a sea of followers in Figure 6.
Really Big Graphs

- May be difficult to keep all in memory
- Often visualized as “hairballs”
- Smart visualizations do structural clustering, so you see a high-level overview of topology

ASK-GraphView

Uses clustering algorithms to construct a hierarchy

Abello, van Ham & Krishnan
TVCG (InfoVis) ’06
**TopoLayout**

- Topological features are detected recursively inside a graph
- Their subgraphs are collapsed into single nodes, forming a hierarchy
- Each feature drawn with an algorithm tuned for its topology

**PNNL Big Graph Work**

- Goal is to show more nodes in a coherent fashion and provide real-time interactions
- Uses multi-level graph drawing
- Preprocesses graph by generating hierarchy of increasingly coarse layouts
Alternate Big Graph Approach

- Show some of the details, rather than high level structure
- Allow users to focus on particular nodes
- Adapt DOI algorithm from trees to graphs
- Rely heavily on interaction

- Different paradigm: “Search, show context, expand on demand”

van Ham & Perer
TVCG (InfoVis) ’09
Graphs as Maps

- Represent a large graph as a map
- Maintain inherent structure and relationships between nodes
- Follow standard cartographic representations

Gansner, Hu & Kobourov
IEEE CG&A (PacificVis) '10
Both Representations

Node-link

Map

Music Graph/Map
Drawing Graphs Better

• Can we do clever “tricks” to make dense graphs more readable?

Hierarchical Edge Bundles

• Bundle edges that go from/to similar nodes together
  – Like wires in a house
• Uses B-spline curves for edges
• Reduces the clutter from many edges

Holten

TVCG (InfoVis) ’06
Example

Fig. 11. A software system and its associated call graph (caller = green, callee = red). (a) and (b) show the system without bundling using a radial and an equidistant treemap layout (node labels disabled), respectively. (b) and (b) mainly show hot spots; the actual connectivity information is more difficult to discern due to visual clutter.

Fig. 12. Radial layout construction. (a) A radial tree layout is used for the inner circle and subsequently mirrored to the outside. (b) the inner layout is hidden and its structure is used to guide the adjacency edges. An icon plot based on the mirrored layout is used to show the hierarchy.

Example

Fig. 13. A software system and its associated call graph (caller = green, callee = red). (a) and (b) show the system with bundling strength $b = 5.5$ using a balloon layout (node labels disabled) and a radial layout, respectively. Bundling reduces visual clutter, making it easier to perceive the actual connections than when compared to the non-bundled versions (figures 13a and 13b). Bundled visualizations also show relations between sparsely connected systems more clearly (encircled regions); these are almost completely obscured in the non-bundled versions. The encircled regions highlight critical parts of the system for (a), (b), and figure 15.
Matrix Representations

- There has been renewed interest in matrix representations of graphs recently.
- I think the regularity, symmetry, and structure of a matrix are a win – people understand them well, but they don’t scale up really well.

MatrixExplorer

- Provides matrix view in combination with node-link and various operations for gaining different perspectives.
Node Reordering

Extremely important operation with matrix representations

Fig. 6. Initial order (left) and TSP order (right). Colors represent clusters found by the user. Clusters are different in the two representations. Users found more clusters with TSP order. Headers red indicators (right) represents the distance between adjacent rows/columns.

NodeTrix

Hybrid of matrix and node-link

Henry & Fekete
TVCG (InfoVis) '07
Simplifying Input

• Make it easier to input graphs and then explore them

http://nodexl.codeplex.com/
Characteristics

- Plug-in for MS Excel
- Includes many network layout and network analysis metrics
- Data import:
  - List out vertices and edges in Excel columns
  - Native importers for email, Twitter, YouTube, etc.

Non-Network Data?

- But what if you don’t have vertex-edge data to begin?
  - May just have tabular data from spreadsheet or database
- Still may want to explore data modeled as a graph
  - Consider DB of NSF grants (PIs, institution, PM, amount, ...)
  - Look for clusters, patterns, connections, ...
**Ploceus**

- Framework and system for modeling and visualizing tabular data as network

- Allow user to model data as graph interactively through direct manipulation
  - What are vertices, edges, edge weights, ...

- Visualizes graph on-the-fly (different layouts and network metrics)

- Advanced ops (project, aggregate, slice-n-dice) can be specified interactively too
Graph Visualization Resource

- Very nice overview & survey
  - Herman et al, *IEEE TVCG* ’00
  - but a little dated now

HW

- OK to remove one?
- Spread those points onto other ones (one per)
Upcoming

• Hierarchies and Trees 1
  – Reading
    Card & Nation '02

• Hierarchies and Trees 2
  – Reading
    Johnson & Shneiderman '91