Connections

- Connections throughout our lives and the world
  - Circle of friends
  - Delta’s flight plans
  - ...
- Model connected set as a *Graph*
What is a Graph?

- Vertices (nodes) connected by
- Edges (links)

![Adjacency matrix]

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Adjacency list

1: 2
2: 1, 3
3: 2

Graph Terminology

- Graphs can have cycles
- Graph edges can be directed or undirected
- The degree of a vertex is the number of edges connected to it
  - In-degree and out-degree for directed graphs
- Graph edges can have values (weights) on them (nominal, ordinal or quantitative)
**Trees are Different**

- Subcase of general graph
- No cycles
- Typically directed edges
- Special designated root vertex

**Graph Uses**

- In information visualization, any number of data sets can be modeled as a graph
  - US telephone system
  - World Wide Web
  - Distribution network for on-line retailer
  - Call graph of a large software system
  - Semantic map in an AI algorithm
  - Set of connected friends
- Graph/network visualization is one of the oldest and most studied areas of InfoVis
Graph Visualization Challenges

- Graph layout and positioning
  - Make a concrete rendering of abstract graph

- Navigation/Interaction
  - How to support user changing focus and moving around the graph

- Scale
  - Above two issues not too bad for small graphs, but large ones are much tougher

Layout Examples

- Homework assignment
- Let’s judge!
Results

• What led to particular layouts being liked more?

• Discuss

Graph Drawing

Entire research community’s focus
Vertex Issues

- Shape
- Color
- Size
- Location
- Label

Edge Issues

- Color
- Size
- Label
- Form
  - Polyline, straight line, orthogonal, grid, curved, planar, upward/downward, ...
Aesthetic Considerations

- Develop a set of metrics to quantitatively rate the “goodness” of a graph layout
- What metrics would you use?

Aesthetic Considerations

- **Crossings** -- minimize towards planar
- **Total Edge Length** -- minimize towards proper scale
- **Area** -- minimize towards efficiency
- **Maximum Edge Length** -- minimize longest edge
- **Uniform Edge Lengths** -- minimize variances
- **Total Bends** -- minimize orthogonal towards straight-line
Which Matters?

- Various studies examined which of the aesthetic factors matter most and/or what kinds of layout/vis techniques look best
  - Purchase, Graph Drawing ‘97
  - Ware et al, *Info Vis* 1(2)
  - Ghoniem et al, *Info Vis* 4(2)
  - van Ham & Rogowitz, *TVCG* ‘08
  - ...

- Results mixed: Edge crossings do seem important

Shneiderman’s NetViz Nirvana

1) Every node is visible
2) For every node you can count its degree
3) For every link you can follow it from source to destination
4) Clusters and outliers are identifiable
Classic Problem

- With enough vertices and enough edges, you get...
- A hairball! (ball-of-string)

But What about User Tasks?

- So what do people want to do with or learn from network visualizations?
  - Recurring theme of this class: Too often this is neglected
Graph Vis Task Taxonomy

- Start with Amar et al ‘05 low-level tasks (retrieve value, find extreme, sort, etc.)
- Then add four types of other tasks (next pages)

Graph Vis Task Taxonomy

1. Topology-based tasks
   - Adjacency
     Find the set of nodes adjacent to a node
   - Accessibility
     Find the set of nodes accessible to a node
   - Common connection
     Given nodes, find the set of nodes connected to all
   - Connectivity
     Find shortest path
     Identify clusters
     Identify connected components
Graph Vis Task Taxonomy

2. Attribute-based tasks
   - On the nodes
     Find the nodes having a specific attribute value
   - On the edges
     Given a node, find the nodes connected only by certain kinds of edges

3. Browsing tasks
   - Follow path
     Follow a given path
   - Revisit
     Return to a previously visited node

4. Overview task
   - Compound exploratory task
     Estimate size of a network
     Find patterns
Layout Heuristics

- Layout algorithms can be
  - polyline edges
  - planar
    - No edge crossings
  - orthogonal
    - horizontal and vertical lines/polylines
  - grid-based
    - vertices, crossings, edge bends have integer coords
  - curved lines
  - hierarchies
  - circular
  - ...

Types of Layout Algorithms

From:
P. Mutzel, et al
Graph Drawing '97
Common Layout Techniques

- Hierarchical
- Force-directed
- Circular
- Geographic-based
- Clustered
- Attribute-based
- Matrix

We will discuss many of these further in the slides to come.

Scale Challenge

- May run out of space for vertices and edges (turns into “ball of string”)
- Can really slow down algorithm

- Sometimes use clustering to help
  - Extract highly connected sets of vertices
  - Collapse some vertices together
Navigation/Interaction Challenge

- How do we allow a user to query, visit, or move around a graph?
- Changing focus may entail a different rendering

Graph Drawing Uses

- Many domains and data sets can benefit significantly from nice graph drawings
- Let’s look at some examples...
Human Diseases

Mapping the Human ‘Diseaseome’

Researchers created a map linking different diseases, represented by circles, to the genes they have in common, represented by squares. Related Article: Redefining Disease, Gene by Gene

Note the two extra variables per vertex

Music Artists

http://www.liveplasma.com/
US Budget

Social Analysis

- Facilitate understanding of complex socio-economic patterns
- Social Science visualization gallery (Lothar Krempel):
- Next slides: Krempel & Plumper’s study of World Trade between OECD countries, 1981 and 1992
Social Network Visualization

- Social Network Analysis
  - http://www.insna.org

Hot topic again
Why?
Terrorists
Facebook
People connections

Steroids in MLB

http://www.slate.com/id/2180392/
Geo Applications

- Many problems and data sets have some geographic correspondence
Follow the Money

Where does a dollar bill go?

3 Subway Diagrams

- Geographic landmarks largely suppressed on maps, except water (rivers in London & Paris) and asphalt (highways in Atlanta)
  - Rather fitting, no?
- These are more graphs than maps!

But Is It InfoVis?

- I generally don’t consider a pure graph layout (drawing) algorithm to be InfoVis
  - Nothing wrong with that, just an issue of focus
- For InfoVis, I like to see some kind of interaction or a system or an application...
  - Still, understanding the layout algorithms is very important for infovis
  - Let’s look at a few...
Circular Layout

Ultra-simple
May not look so great

Space vertices out around circle
Draw lines (edges) to connect vertices

Arc Diagram Layout


Wattenberg
InfoVis '02
**Tree Layout**

- Run a breadth-first search from a vertex
  - This imposes a spanning tree on the graph
- Draw the spanning tree

- Simple and fast, but obviously doesn’t represent the whole graph

---

**Hierarchical Layout**

Often called Sugiyama layout

Try to impose hierarchy on graph
- Reverse edges if needed to remove cycles
- Introduce dummy nodes
- Put nodes into layers or levels
- Order l->r to minimize crossings


*Figure: A graph showing a layered layout, created with the Sugiyama heuristic, with the layers shown. The bends in the edges correspond to dummy nodes.*
**Force-directed Layout**

- Example of constraint-based layout technique
- Impose constraints (objectives) on layout
  - Shorten edges
  - Minimize crossings
  - ...
- Define through equations
- Create optimization algorithm that attempts to best satisfy those equations

**Force-directed Layout**

- Spring model (common)
  - Edges – Springs (gravity attraction)
  - Vertices – Charged particles (repulsion)
- Equations for forces
- Iteratively recalculate to update positions of vertices
- Seeking local minimum of energy
  - Sum of forces on each node is zero
Force-directed Example

Figure 2: A graph drawing through a number of iterations of a force directed algorithm.


In Action

http://vis.stanford.edu/protovis/ex/force.html
Variant

- Spring layout
  - Simple force-directed spring embedder

---

Variant

- Fruchterman-Reingold Algorithm
  - Add global temperature
  - If hot, nodes move farther each step
  - If cool, smaller movements
  - Generally cools over time
Variant

- Kamada-Kawai algorithm
  - Examines derivatives of force equations
  - Brought to zero for minimum energy

Other Applications

- Email
- How would you visualize all email traffic in CoC between pairs of people?
- Solutions???
Possible Solutions

- Put everyone on circle, lines between
  - Color or thicken line to indicate magnitude

- Use spring/tension model
  - People who send a lot to each other are drawn close together
  - Shows clusters of communications

Mucho Examples

http://www.visualcomplexity.com
Graph Drawing Support

- Libraries
  - JUNG (Java Universal Network/Graph Framework)
  - Graphviz (formerly dot?)
- Systems
  - Gephi
  - TouchGraph

http://jung.sourceforge.net/
**Graphviz**

http://www.graphviz.org

**Gephi**

http://gephi.org
TouchGraph

Graph Drawing Resources

• **Book**

• **Tutorial (talk slides)**

• **Web links**
  - http://graphdrawing.org
Upcoming

- **Graphs and Networks 2**
  - Reading
    Perer & Shneiderman ‘06

- **Visual Analytics**
  - Readings
    Keim et al ‘08
    Stasko, Görg & Liu ‘08