Tree and Network Analysis and Visualization

Dr. Katy Börner
Cyberinfrastructure for Network Science Center, Director
Information Visualization Laboratory, Director
School of Library and Information Science
Indiana University, Bloomington, IN
http://info.slis.indiana.edu/~katy

With special thanks to Kevin W. Boyack, Micah Linnemeier,
Russell J. Duhon, Patrick Phillips, Joseph Biberstine, Chintan Tank
Nianli Ma, Hanning Guo, Mark A. Price, Angela M. Zoss, and
Scott Weingart

Guest Lecture in S604/S764 Information Networks by Staša Milojević
November 14, 2011

12 Tutorials in 12 Days at NIH—Overview

1. Science of Science Research
2. Information Visualization
3. CISHell Powered Tools: Network Workbench and Science of Science Tool

4. Temporal Analysis—Burst Detection
5. Geospatial Analysis and Mapping
6. Topical Analysis & Mapping

7. Tree Analysis and Visualization
8. Network Analysis
9. Large Network Analysis

10. Using the Scholarly Database at IU
11. VIVO National Researcher Networking
12. Future Developments
# Tree Analysis and Visualization

- General Overview
- Designing Effective Tree Visualizations
- Notions and Notations
- Sci2-Reading and Extracting Trees
- Sci2-Visualizing Trees
- Outlook

## Sample Trees and Visualization Goals & Objectives

<table>
<thead>
<tr>
<th>Sample Trees</th>
<th>Goals &amp; Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hierarchies</strong></td>
<td>Representing hierarchical data</td>
</tr>
<tr>
<td>➢ File systems and web sites</td>
<td>➢ Structural information</td>
</tr>
<tr>
<td>➢ Organization charts</td>
<td>➢ Content information</td>
</tr>
<tr>
<td>➢ Categorical classifications</td>
<td></td>
</tr>
<tr>
<td>➢ Similarity and clustering</td>
<td></td>
</tr>
<tr>
<td><strong>Branching Processes</strong></td>
<td>Objectives</td>
</tr>
<tr>
<td>➢ Genealogy and lineages</td>
<td>➢ Efficient Space Utilization</td>
</tr>
<tr>
<td>➢ Phylogenetic trees</td>
<td>➢ Interactivity</td>
</tr>
<tr>
<td><strong>Decision Processes</strong></td>
<td>➢ Comprehension</td>
</tr>
<tr>
<td>➢ Indices or search trees</td>
<td>➢ Esthetics</td>
</tr>
<tr>
<td>➢ Decision trees</td>
<td></td>
</tr>
</tbody>
</table>

Pat Hanrahan, Stanford U
Radial Tree – How does it work?
See also http://iv.slis.indiana.edu/sw/radialtree.html

- All nodes lie in concentric circles that are focused in the center of the screen.
- Nodes are evenly distributed.
- Branches of the tree do not overlap.


Radial Tree – Pseudo Algorithm

Circle Placement
Maximum size of the circle corresponds to minimum screen width or height. Distance between levels \( d := \text{radius of max circle size} / \text{number of levels in the graph} \).

Node Placement

Level 0
The root node is placed at the center.

Level 1
All nodes are children of the root node and can be placed over all the 360° of the circle - divide 2\( \pi \) by the number of nodes at level 1 to get angle space between the nodes on the circle.
Levels 2 and greater
Use information on number of parents, their location, and their space for children to place all level x nodes.
Loop through the list of parents and then loop through all the children for that parent and calculate the child's location relative to the parent's, adding in the offset of the limit angle.
After calculating the location, if there are any directories at the level, we must calculate the bisector and tangent limits for those directories.

We then iterate through all the nodes at level 1 and calculate the position of the node Bisector Limits
Tangent and bisector limits for directories

Between any two directories, a bisector limit is calculated to ensure that children do not overlap the children of an adjacent directory.
Hyperbolic Tree – How does it work?
See also http://sw.slis.indiana.edu/sw/hyptree.html

Phylogenetic Tree

Hyperbolic Geometry

Inspired by Escher’s Circle Limit IV (Heaven and Hell), 1960.

- Focus+context technique for visualizing large hierarchies
- Continuous redirection of the focus possible.

The hyperbolic plane is a non-Euclidean geometry in which parallel lines diverge away from each other. This leads to the convenient property that the circumference of a circle on the hyperbolic plane grows exponentially with its radius, which means that exponentially more space is available with increasing distance.

Hyperbolic Tree Layout

2 Steps:

Recursively lay out each node based on local information.

- A node is allocated a wedge of the hyperbolic plane, angling out from itself, to put its descendants in.
- It places all its children along an arc in that wedge, at an equal distance from itself, and far enough out so that the children are some minimum distance apart from each other.
- Each of the children then gets a sub-wedge for its descendants. (Because of the divergence of parallel lines in hyperbolic geometry, each child will typically get a wedge that spans about as big an angle as does its parent’s wedge.)

Map hyperbolic plane onto the unit disk

Poincaré model is a canonical way of mapping the hyperbolic plane to the unit disk. It keeps one vicinity in the hyperbolic plane in focus at the center of the disk while the rest of the hyperbolic plane fades off in a perspective-like fashion toward the edge of the disk.

Poincaré model preserves the shapes of fan-outs at nodes and does a better job of using the screen real-estate.

Change of Focus – Animated Transitions

Node & Edge Information

Treemap – How does it work?

See also http://sw.slis.indiana.edu/sw/treemap.html

**Treemaps – Layout**

*Ben Shneiderman, Tree Visualization with Tree-Maps: 2-d Space-Filling Approach*

**Treemap – Pseudo Code**

**Input**
Tree root & a rectangular area defined by upper left and lower right coordinates P1(x1, y1), Q1(x2, y2).

**Recursive Algorithm**

active_node := root_node;
partitioning_direction := horizontal; // nodes are partitioned vertically at even levels and horizontally at odd levels

Treemap(active_node) {
    determine number n of outgoing edges from the active_node;
    if (n<1)
        end;
    if (n>1) {
        divide the region [x1, x2] in partitioning_direction were the size of the n partitions correspond to their fraction (Size(child[i])/Size(active)) of the total number of bytes in the active_node;
        change partitioning_direction;
        for (1<=i<=n) do
            Treemap(child[i]);
    }
Treemap – Properties

Strengths

➢ Utilizes 100% of display space
➢ Shows nesting of hierarchical levels.
➢ Represents node attributes (e.g., size and age) by area size and color
➢ Scalable to data sets of a million items.

Weaknesses

➢ Size comparison is difficult
➢ Labeling is a problem.
➢ Cluttered display
➢ Difficult to discern boundaries
➢ Shows only leaf content information

Treemap – Algorithm Improvements

Sorted treemap
Marc Smith

Cushion treemap
[#07] Tree Analysis and Visualization

- General Overview
- Designing Effective Tree Visualizations
- Notions and Notations
- Sci2-Reading and Extracting Trees
- Sci2-Visualizing Trees
- Outlook
- Exercise: Identify Promising Tree Analyses of NIH Data
Tree Nodes and Edges

The **root node** of a tree is the node with no parents. A **leaf node** has no children.

**In-degree** of a node is the number of edges arriving at that node. **Out-degree** of a node is the number of edges leaving that node.

Sample tree of **size** 11 (=number of nodes) and **height** 4 (=number of levels).

---

[07] Tree Analysis and Visualization

- General Overview
- Designing Effective Tree Visualizations
- Notions and Notations
- Sci2-Reading and Extracting Trees
- Sci2-Visualizing Trees
- Outlook
- Exercise: Identify Promising Tree Analyses of NIH Data
Read and Visualize Trees with Sci2 Tool


Sample Tree: Read Directory Hierarchy

Use ‘File > Read Directory Hierarchy’ with parameters

To view file in different formats right click ‘Directory Tree - Prefuse (Beta) Graph’ in Data Manager and select View.
Select a data format.
Sample Tree: View Directory Hierarchy

File Formats: GraphML (Prefuse)

See documentation at https://nwb.slis.indiana.edu/community/?n=DataFormats.HomePage

```xml
<xml version="1.0" encoding="UTF-8" >
  <graphml xmlns="http://graphml.graphdrawing.org/xmlns" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://graphml.graphdrawing.org/xmlns http://graphml.graphdrawing.org/xmlns/1.0/graphml.xsd">
    <key id="label" for="node" attr.name="label" attr.type="string">
      <default />
    </key>
    <key id="label" for="edge" attr.name="label" attr.type="string">
      <default />
    </key>
    <graph edgedefault="undirected">
      <nodes>
        <node id="n0">
          <data key="label">sci2-with-scimaps</data>
        </node>
        <node id="n1">
          <data key="label">eclipseproduct</data>
        </node>
        <node id="n2">
          <data key="label">sci2.exe</data>
        </node>
        <node id="n3">
          <data key="label">sci2.ini</data>
        </node>
        <node id="n4">
          <data key="label">configuration</data>
        </node>
        <node id="n5">
          <data key="label">config.ini</data>
        </node>
      </nodes>
      <edges>
        <edge source="n0" target="n1" label=""/>
        <edge source="n0" target="n2" label=""/>
        <edge source="n0" target="n3" label=""/>
        <edge source="n0" target="n4" label=""/>
        <edge source="n0" target="n5" label=""/>
        <edge source="n1" target="n2" label=""/>
        <edge source="n1" target="n3" label=""/>
        <edge source="n1" target="n4" label=""/>
        <edge source="n1" target="n5" label=""/>
        <edge source="n2" target="n3" label=""/>
        <edge source="n2" target="n4" label=""/>
        <edge source="n2" target="n5" label=""/>
        <edge source="n3" target="n4" label=""/>
        <edge source="n3" target="n5" label=""/>
        <edge source="n4" target="n5" label=""/>
      </edges>
    </graph>
  </graphml>
```

Sample Tree: View Directory Hierarchy

File Formats: NWB

See documentation at https://nwb.slis.indiana.edu/community/?n=DataFormats.HomePage

```
*Nodes
id:int label:string
1  "sci2-with-scimaps"
2  "eclipseproduct"
3  "sci2.exe"
4  "sci2.ini"
5  "configuration"
6  "config.ini"
7  "default_menu.xml"
...

*UndirectedEdges
source:int target:int label:string
1  2  ""
1  3  ""
1  4  ""
1  5  ""
1  6  ""
1  7  ""
1  8  ""
1  9  ""
1  10  ""
```
Sample Tree: View Directory Hierarchy

File Formats: Pajek .net
See documentation at https://nwb.slis.indiana.edu/community/?n=DataFormats.HomePage

Sample Tree: View Directory Hierarchy

File Formats: Pajek .mat
See documentation at https://nwb.slis.indiana.edu/community/?n=DataFormats.HomePage
Sample Tree: View Directory Hierarchy

File Formats: TreeML (Prefuse)
See documentation at https://nwb.slis.indiana.edu/community/?n=DataFormats.HomePage

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!--  preface TreeML Writer | Sat Jul 17 12:05:02 EDT 2010  -->
<tree>
  <!-- declarations -->
  <attribute name="label" type="String"/>
</declarations>
  <branch>
    <attribute name="label" value="scl2-with-sclmaps"/>
    <leaf>
      <attribute name="label" value="eclipseproduct"/>
    </leaf>
    <leaf>
      <attribute name="label" value="scl2-choe"/>
    </leaf>
    <leaf>
      <attribute name="label" value="scl2.ini"/>
    </leaf>
  </branch>
```
Sample Tree: View Directory Hierarchy

File Formats: XGMML (Prefuse)
See documentation at https://nwb.slis.indiana.edu/community/?n=DataFormats.HomePage

```xml
<graph directed="0" label="Network" xmlns="http://www.cs.rpi.edu/XGMML">

<node id="1" label="edu.ui.scipolicy.database.isi.extract.network.cocitation.journal.core_0.0.1.jar" />
<node id="2" label="org.cisshell.templates.jythonrunner_1.0.0" />
<node id="3" label="feature.xml" />
<node id="4" label="META-INF" />
<node id="5" label="IsiCoCitation.properties" />
<node id="6" label="edu.ui.nwb.converter.nwbapajeknet_1.0.0.jar" />
<node id="7" label="fastep-graphicsio-pdf-2.0.jar" />
<node id="8" label="Welcome.properties" />
<node id="9" label="org.cisshell.reference.gui.persistence_1.0.0.jar" />

<edge source="2" target="244" label="" />
<edge source="2" target="479" label="" />
<edge source="4" target="335" label="" />
<edge source="25" target="360" label="" />
<edge source="26" target="362" label="" />
<edge source="34" target="371" label="" />
<edge source="35" target="177" label="" />
<edge source="30" target="372" label="" />
<edge source="30" target="366" label="" />
```

[#07] Tree Analysis and Visualization

- General Overview
- Designing Effective Tree Visualizations
- Notions and Notations
- Sci2-Reading and Extracting Trees
- Sci2-Visualizing Trees
- Outlook
- Exercise: Identify Promising Tree Analyses of NIH Data
**Sample Tree Visualizations**

**Indented Lists** and **Tree View** showing nesting of, e.g., directory hierarchies. Visualize 'Directory Tree - Prefuse (Beta) Graph' using

- “Visualization > Networks > Tree View (prefuse beta)”

Press right mouse button and use mouse wheel/touch pad to zoom in and out.
Click on directory to expand/collapse.
Use search field to find specific files.

---

**Sample Tree Visualizations**

**Radial Tree** and **Balloon Tree** showing the structure of, e.g., directory hierarchies. Visualize 'Directory Tree - Prefuse (Beta) Graph' using

- “Visualization > Networks > Radial Tree/Graph (prefuse alpha)”
- “Visualization > Networks > Balloon Graph (prefuse alpha)” (not in Sci2 Tool, Alpha 3)
Sample Tree Visualization

**Tree Map** showing the structure of, e.g., directory hierarchies. Visualize ‘Directory Tree - Prefuse (Beta) Graph’ using
- ‘Visualization > Networks > Tree Map (prefuse beta)’

Sample Tree Visualization

**Flow Maps** showing migration patterns

*Soon available in Sci2 Tool.*
[#07] Tree Analysis and Visualization

- General Overview
- Designing Effective Tree Visualizations
- Notions and Notations
- Sci2-Reading and Extracting Trees
- Sci2-Visualizing Trees
- Outlook
- Exercise: Identify Promising Tree Analyses of NIH Data

Outlook

Planned extensions of Sci2 Tool:

- (Flowmap) tree network overlays for geo maps and science maps.
- Bimodal network visualizations.
- Scalable visualizations of large hierarchies.

Research Collaborations by the Chinese Academy of Sciences
By Węczia (Bonnie) Huang, Russell J. Duhon, Elisha F. Hardy, Katy Börner, Indiana University, USA
[08] Network Analysis and Visualization

- General Overview
- Designing Effective Network Visualizations
- Notions and Notations
- Sci2-Reading and Extracting Networks
- Sci2-Analysing Networks
- Sci2-Visualizing Networks
- Outlook

Sample Networks

- Communication networks
  - Internet, telephone network, wireless network.
- Network applications
  - The World Wide Web, Email interactions
- Transportation network/ Road maps
- Relationships between objects in a data base
  - Function/module dependency graphs
  - Knowledge bases

Network Properties

- Directed vs. undirected
- Weighted vs. unweighted
- Additional node and edge attributes
- One vs. multiple node & edge types
- Network type (random, small world, scale free, hierarchical networks)
Reducing the number of edges via pathfinder network scaling.

Co-word space of the top 50 highly frequent and bursty words used in the top 10% most highly cited PNAS publications in 1982-2001.

(Mane & Börner, 2004)

Historiograph of DNA Development
(Garfield, Sher, & Torpie, 1964)

\[\text{Figure 6.3 Historiograph of DNA development.}\]

- Direct or strongly implied citation
- Indirect citation
Force Directed Layout – How does it work?

The algorithm simulates a system of forces defined on an input graph and outputs a locally minimum energy configuration. Nodes resemble mass points repelling each other and the edges simulate springs with attracting forces. The algorithm tries to minimize the energy of this physical system of mass particles.

Required are
- A force model
- Technique for finding locally minimum energy configurations.

P. Eades, "A heuristic for graph drawing"
*Congressus Numerantium, 42, 149-160, 1984.*

---

**Force Directed Layout cont.**

**Force Models**

<table>
<thead>
<tr>
<th>Force Model</th>
<th>Formula</th>
<th>Example of usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring Force</td>
<td>$F = k(1-a)$</td>
<td>Assigning different $k$ and $a$ to different edges to separate nodes by different distances.</td>
</tr>
<tr>
<td></td>
<td>$k$: stiffness of spring</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$a$: natural length of spring</td>
<td></td>
</tr>
<tr>
<td>Gravity Force</td>
<td>$F = g t^2$</td>
<td>Apply gravity force between node pairs to prevent node overlapping.</td>
</tr>
<tr>
<td></td>
<td>$g$: associated with mass of node, usually equals 1.</td>
<td></td>
</tr>
<tr>
<td>Electrical and Magnetic Force</td>
<td>$F = eE$</td>
<td>Changes nodes distribution along a direction.</td>
</tr>
<tr>
<td></td>
<td>$F = qB$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$E$: electric field strength</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$B$: magnetic field strength</td>
<td></td>
</tr>
</tbody>
</table>

A simple algorithm to find the equilibrium configuration is to trace the move of each node according to Newton’s 2nd law. This takes time $O(n^3)$, which makes it unsuitable for large data sets. Rob Forbes (1987) proposed two methods that were able to accelerate convergence of a FDP problem 3-4 times. One stabilizes the derivative of the repulsion force and the other uses information on node movement and instability characteristics to make a predictive extrapolation.
Most existing algorithms extend Eades’ algorithm (1984) by providing methods for the intelligent initial placement of nodes, clustering the data to perform an initial coarse layout followed by successively more detailed placement, and grid-based systems for dividing up the dataset.

GEM (Graph EMbedder) attempts to recognize and forestall non-productive rotation and oscillation in the motion of nodes in the graph as it cools, see Frick, A., A. Ludwig and H. Mehldau (1994). *A fast adaptive layout algorithm for undirected graphs*, Graph Drawing, Springer-Verlag: 388-403.


Today, the algorithm developed by Kamada and Kawai (Kamada and Kawai 1989) and Fruchterman and Reingold (Fruchterman and Reingold 1991) are most commonly used, partially because they are available in Pajek.


Notions and Notations

1. Introduction ................................................................. 2
2. Notions and Notations .................................................... 4
  2.1 Graphs and Subgraphs ............................................. 5
  2.2 Graph Connectivity .................................................. 5
3. Network Sampling ....................................................... 5
4. Network Measurements ................................................ 5
  4.1 Node and Edge Properties ........................................ 5
  4.2 Local Structure ...................................................... 5
  4.3 Statistical Properties .............................................. 5
  4.4 Network Types ...................................................... 5
  4.5 Discussion and Exemplification ................................... 5
5. Network Modeling ...................................................... 5
  5.1 Modeling Static Networks ......................................... 5
  5.2 Modeling Evolving Networks ..................................... 5
  5.3 Discussion .......................................................... 34
  5.4 Model Validation ..................................................... 34
6. Modeling Dynamics on Networks .................................. 34
7. Network Visualization ................................................ 34
  7.1 Visualization Design Basics ..................................... 41
  7.2 Matrix Visualization .............................................. 44
  7.3 Tree Layout ........................................................ 45
  7.4 Graph Layout ....................................................... 46
  7.5 Visualization of Dynamics ........................................ 48
  7.6 Interaction and Distortion Techniques ...................... 50
8. Discussion and Outlook ................................................ 50

Notions and Notations

Figure 2: Adjacency matrix and graph presentations of different undirected and directed graphs.


Notions and Notations

2.2.1 Node Degree

In undirected graphs, the degree $k$ of a node is the number of edges connected to it. In directed graphs, the degree of a node is defined by the sum of its in-degree and its out-degree, $k_i = k_{in,i} + k_{out,i}$, where the in-degree $k_{in,i}$ is defined as the number of edges pointing to $i$, and its out-degree $k_{out,i}$ is defined as the number of edges departing from $i$. In terms of the adjacency matrix, we can write

$$k_{in,i} = \sum_j A_{ij}, \quad k_{out,i} = \sum_j A_{ji}$$

(1)

For an undirected graph, with a symmetric adjacency matrix, $k_{in,i} = k_{out,i} \equiv k_i$ holds. For example, node 1 in Figure 2a has a degree of three. Node 1 in Figure 2e has an in-degree of two and an out-degree of one.

2.2.2 Nearest Neighbors

The nearest neighbors of a node $i$ are the nodes to which it is connected directly by an edge, so the number of nearest neighbors of the node is equal to the node degree. For example, node 1 in Figure 2a has nodes 0, 2, and 3 as nearest neighbors.

2.2.3 Path

A path $P_{i,j}$ that connects the nodes $i_0$ and $i_n$ in a graph $G = (V,E)$ is defined as an ordered collection of $n+1$ nodes $V_p = \{i_0,i_1,...,i_n\}$ and $n$ edges $E_p = \{(i_{\alpha-1},i_{\alpha}),(i_{\alpha},i_{\alpha+1}),...,(i_{n-1},i_n)\}$, such that $i_0 \in V$ and $(i_{\alpha-1},i_{\alpha}) \in E$, for all $\alpha$. The length of the path $P_{i,j}$ is $n$. For example, the path in Figure 2f that interconnects nodes 0, 1, and 2 has a length of two.

**Notions and Notations**

Betweenness centrality is a measure that aims to describe a node’s position in a network in terms of the flow it is able to control. As an example, consider two highly connected subgraphs that share one node but no other nodes or edges. Here, the shared node controls the flow of information, for example, rumors in a social network. Any path from any node in one subgraph to any node in the other subgraph leads through the shared node. The shared node has a rather high betweenness centrality. Mathematically, the betweenness centrality is defined as the number of shortest paths between pairs of nodes that pass through a given node (Freeman, 1977). More precisely, let $L_{ij}$ be the total number of shortest paths from $i$ to $j$ and $L_{h,j}$ be the number of those shortest paths that pass through the node $h$. The betweenness $b_i$ of node $i$ is then defined as $b_i = \sum L_{h,j} / L_{h,j}$, where the sum runs over all $h,j$ pairs with $j \neq h$. An efficient algorithm to compute betweenness centrality was reported by Brandes (2001). The betweenness centrality is often used in transportation networks to provide an estimate of the traffic handled by different nodes, assuming that the frequency of use can be approximated by the number of shortest paths passing through a given node. It is important to stress that while the betweenness centrality is a local attribute of any given node, it is calculated by looking at all paths among all nodes in the network and therefore it is a measure of the node centrality with respect to the global topology of the network.

Network Extraction - Examples

Sample paper network (left) and four different network types derived from it (right). From ISI files, about 30 different networks can be extracted.

Local citation counts (within this dataset) are given in black and global citation counts (ISI times cited) are given in green above each paper.

Extract Networks with Sci2 Tool – Database


See also Tutorial #3
Extract Networks with Sci2 Tool – Text Files


See also Tutorial #3

Fake NIH Dataset of Awards and Resulting Publications

Ten existing awards and a fake set of resulting publications.

<table>
<thead>
<tr>
<th>Award ID</th>
<th>Publication ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>C06CA859690</td>
<td>9405464,903032</td>
</tr>
<tr>
<td>C06CA859697</td>
<td>2057532,8980722,2042765,30165186,20019401,10567228</td>
</tr>
<tr>
<td>C06RA011192</td>
<td>18913728,18362150,19480321</td>
</tr>
<tr>
<td>C06RA012176</td>
<td>9714740,19480321</td>
</tr>
<tr>
<td>C06RR012408</td>
<td>15346791,11943438,12006525,12004528</td>
</tr>
<tr>
<td>C06RR012511</td>
<td>19866515,18487208,19214230</td>
</tr>
<tr>
<td>C06RR012512</td>
<td>18991362,1712941,1836192,18621368,18959716,17604144,17360927,17134906,19156177</td>
</tr>
<tr>
<td>C06RR012537</td>
<td>18207487,17318410,17961162,19480321</td>
</tr>
<tr>
<td>C06RR013501</td>
<td>16138041</td>
</tr>
<tr>
<td>C06RR014469</td>
<td>17621623</td>
</tr>
</tbody>
</table>

Load resulting using ‘File > Load > Fake-NIH-Awards+Publications.csv’ as csv file format.

Extract author bipartite grant to publications network using ‘Data Preparation > Text Files > Extract Directed Network’ using parameters:
Fake NIH Dataset cont.

**Network Analysis Toolkit (NAT)**

This graph claims to be directed.
Nodes: 43
Isolated nodes: 0
Edges: 35
No self loops were discovered.
No parallel edges were discovered.
Did not detect any edge attributes
This network does not seem to be a valued network.

Average total degree: 1.6279
Average in degree: 0.814
Average out degree: 0.814
This graph is not weakly connected.
There are 8 weakly connected components. (0 isolates)
The largest connected component consists of 10 nodes.

Density (disregarding weights): 0.0194

**GUESS**
GEM Layout, Bin pack

Fake NIH Dataset cont.

**In Sci2**
Node Indegree was selected.

...........
Node Outdegree was selected.

**GUESS**
GEM Layout, Bin pack
Color using Graph Modifier
In Sci2
Weak Component Clustering,
Input Parameters:
Number of top clusters: 10
8 clusters found, generating graphs for the top 8 clusters.

Visualize giant component in GUESS

---

#08 Network Analysis and Visualization
- General Overview
- Designing Effective Network Visualizations
- Notions and Notations
- Sci2-Reading and Extracting Networks
- Sci2-Analyzing Networks
- Sci2-Visualizing Networks
- Outlook
- Exercise: Identify Promising Network Analyses of NIH Data
Discover Landmark Nodes based on
- Connectivity (degree or BC values)
- Frequency of access
(Source: Mukherjea & Hara, 1997; Hearst p. 38 formulas)

Identify Major (and Weak) Links

Identify the Backbone

Show Clusters

See also Ketan Mane’s Qualifying Paper
http://ella.slis.indiana.edu/~katy/teaching/ketan-quals-slides.ppt

Figure 2: Approaches to deal with large networks

Pajek Tutorial

[#08] Network Analysis and Visualization
- General Overview
- Designing Effective Network Visualizations
- Notions and Notations
- Sci2-Reading and Extracting Networks
- Sci2-Analysing Networks
- Sci2-Visualizing Networks
- Outlook
- Exercise: Identify Promising Network Analyses of NIH Data
Network Visualization

General Visualization Objectives
- Representing structural information & content information
- Efficient space utilization
- Easy comprehension
- Aesthetics
- Support of interactive exploration

Challenges in Visualizing Large Networks
- Positioning nodes without overlap
- De-cluttering links
- Labeling
- Navigation/interaction

General Network Representations

Matrices

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0</th>
<th>6</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>10.5</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0.015</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>150.5</td>
<td>0</td>
<td>-280</td>
<td>13.32</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>12</td>
</tr>
</tbody>
</table>

Structure Plots

Equivalenced representation of US power network

Lists of nodes & links

*Vertices 3
1 "Doc1" 0.00000000 c Green bc Brown
2 "Doc2" 0.00000000 c Green bc Brown
3 "Doc3" 0.00000000 c Green bc Brown
*Arcs
1 2 3 c Green
2 3 5 c Black
*Edges
1 3 4 c Green

Network layouts of nodes and links
Aesthetic Criteria for Network Visualization

- Symmetric.
- Evenly distributed nodes.
- Uniform edge lengths.
- Minimized edge crossings.
- Orthogonal drawings.
- Minimize area / bends / slopes / angles

Optimization criteria may be relaxed to speed up layout process.

(Source: Fruchterman & R. alg p. 76, see Table & discussion Hearst, p 88)

Small Networks

- Up to 100 nodes
- All nodes and edges and most of their attributes can be shown.

**General mappings for**

**nodes**
- # -> (area) size
- Intensity (secondary value) -> color
- Type -> shape

**edges**
- # -> thickness
- Intensity, age, etc. -> color
- Type -> style

Medium Size Networks

- Up to 10,000 nodes
- Most nodes can be shown but not all their labels.
- Frequently, the number of edges and attributes need to be reduced.

**Major design strategies:**

Show only important nodes, edges, labels, attributes
Order nodes spatially
Reduce number of displayed nodes
Visualize Networks with Sci2 Tool


NSF Medical+Health Funding:
Bimodal Network of NSF Organization to Program(s)

Extract Directed Network was selected.
Source Column: NSF Organization
Text Delimiter: |
Target Column: Program(s)

Nodes: 167
Isolated nodes: 0
Edges: 177
No parallel edges were discovered.
Did not detect any edge attributes
Density (disregarding weights): 0.00638
Load into NWB, open file to count records, compute total award amount.

Run ‘Scientometrics > Extract Directed Network’ using parameters:

- Select “Extracted Network ..” and run ‘Analysis > Network Analysis Toolkit (NAT)’
- Remove unconnected nodes via ‘Preprocessing > Delete Isolates’.
- Run ‘Analysis > Unweighted & Directed Network > Node Indegree / Node Outdegree’.
- ‘Visualization > GUESS’, layout with GEM, Bin Pack
- Use Graph Modifier to color/size network.

NIH CTSA Grants:
Co-Project Term Descriptions Occurrence Network

Load... was selected.
Loaded: ...

Extract Co-Occurrence Network was selected.
Input Parameters:
Text Delimiter: ...
Column Name: Project term descriptions

Network Analysis Toolkit (NAT) was selected.
Nodes: 5723
Isolated nodes: 3
Edges: 353218
NIH CTSA Publications: Co-Mesh Terms Occurrence Network

Load... was selected.
Loaded: \NIH-data\NIH-CTSA-Publications.csv

Extract Co-Occurrence Network was selected.
Input Parameters:
Text Delimiter: ;
Column Name: Mesh Terms

Network Analysis Toolkit (NAT) was selected.
Nodes: 10218
Edges: 163934

---

[#09] Large Network Analysis and Visualization

- General Overview
- Designing Effective Network Visualizations
- Sci2-Reading Networks
- Sci2-Analysing Large Networks
- Sci2-Visualizing Large Networks and Distributions
- Outlook
Large Networks

- More than 10,000 nodes.
- Neither all nodes nor all edges can be shown at once. Sometimes, there are more nodes than pixels.

Examples of large networks

- Communication networks:
  - Internet, telephone network, wireless network.
- Network applications:
  - The World Wide Web, Email interactions
- Transportation network/road maps
- Relationships between objects in a data base:
  - Function/module dependency graphs
  - Knowledge bases

http://loadrunner.uits.iu.edu/weathermaps/abilene/

Amsterdam RealTime project, WIRED Magazine, Issue 11.03 - March 2003
Direct Manipulation

Modify focusing parameters while continuously provide visual feedback and update display (fast computer response).

- Conditioning: filter, set background variables and display foreground parameters
- Identification: highlight, color, shape code
- Parameter control: line thickness, length, color legend, time slider, and animation control
- Navigation: Bird’s Eye view, zoom, and pan
- Information requests: Mouse over or click on a node to retrieve more details or collapse/expand a subnetwork


VxInsight Tool

VxInsight is a general purpose knowledge visualization software package developed at Sandia National Laboratories.

It enables researchers, analysts, and decision-makers to accelerate their understanding of large databases.

Davidson, G.S., Hendrickson, B., Johnson, I
### Other Tools

<table>
<thead>
<tr>
<th>Tool</th>
<th>Year</th>
<th>Domain</th>
<th>Description</th>
<th>Open Source</th>
<th>Operating System</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>BibExcel</td>
<td>2000</td>
<td>Scientom.</td>
<td>Transforms bibliographic data into forms usable in Excel, Pajek, NetDraw, and other programs.</td>
<td>No</td>
<td>Windows</td>
<td>(Persson, 2008)</td>
</tr>
<tr>
<td>Boost Graph Library</td>
<td>2000</td>
<td>CS</td>
<td>Extremely useful and flexible C++ library for extremely large networks.</td>
<td>Yes</td>
<td>All Major</td>
<td>(Siek et al., 2002)</td>
</tr>
<tr>
<td>Visone</td>
<td>2001</td>
<td>SocSci</td>
<td>Social network analysis tool for research and teaching, with a focus on innovative and advanced visual methods.</td>
<td>No</td>
<td>All Major</td>
<td>(Brandes &amp; Wagner, 2008)</td>
</tr>
<tr>
<td>Cytoscape</td>
<td>2002</td>
<td>Bio</td>
<td>Network visualization and analysis tool focusing on biological networks, with particularly nice visualizations.</td>
<td>Yes</td>
<td>All Major</td>
<td>(Cytoscape-Consortium, 2008)</td>
</tr>
</tbody>
</table>


### Other Tools cont.

<table>
<thead>
<tr>
<th>Tool</th>
<th>Year</th>
<th>Domain</th>
<th>Description</th>
<th>Open Source</th>
<th>Operating System</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>GeoVISTA</td>
<td>2002</td>
<td>Geo</td>
<td>GIS software that can be used to lay out networks on geospatial substrates.</td>
<td>Yes</td>
<td>All Major</td>
<td>(Takatsuka &amp; Gehegan, 2002)</td>
</tr>
<tr>
<td>iGraph</td>
<td>2003</td>
<td>CS</td>
<td>A library for classic and cutting edge network analysis usable with many programming languages.</td>
<td>Yes</td>
<td>All Major</td>
<td>(Csárdi &amp; Nepusz, 2006)</td>
</tr>
<tr>
<td>Tulip</td>
<td>2003</td>
<td>CS</td>
<td>Graph visualization software for networks over 1,000,000 elements.</td>
<td>Yes</td>
<td>All Major</td>
<td>(Auber, 2003)</td>
</tr>
<tr>
<td>CiteSpace</td>
<td>2004</td>
<td>Scientom.</td>
<td>A tool to analyze and visualize scientific literature, particularly co-citation structures.</td>
<td>Yes</td>
<td>All Major</td>
<td>(Chen, 2006)</td>
</tr>
<tr>
<td>GraphViz</td>
<td>2004</td>
<td>Networks</td>
<td>Flexible graph visualization software.</td>
<td>Yes</td>
<td>All Major</td>
<td>(AT&amp;T Research Group, 2008)</td>
</tr>
<tr>
<td>R</td>
<td>2004</td>
<td>Statistics</td>
<td>A statistical computing language with many libraries for sophisticated network analysis.</td>
<td>Yes</td>
<td>All Major</td>
<td>(Bakka &amp; Gentleman, 1990)</td>
</tr>
<tr>
<td>Prefuse</td>
<td>2005</td>
<td>Visualiz.</td>
<td>A general visualization framework with many capabilities to support network visualization and analysis.</td>
<td>Yes</td>
<td>All Major</td>
<td>(Heer et al., 2005)</td>
</tr>
<tr>
<td>NWB Tool</td>
<td>2006</td>
<td>Bio, IS, SocSci, Scientom</td>
<td>Network analysis &amp; visualization tool conducive to new algorithms supportive of many data formats.</td>
<td>Yes</td>
<td>All Major</td>
<td>(Huang, 2007)</td>
</tr>
</tbody>
</table>

# Large Network Analysis and Visualization

- General Overview
- Designing Effective Network Visualizations
- Sci2-Reading Networks
- Sci2-Analyzing Large Networks
- Sci2-Visualizing Large Networks and Distributions
- Outlook
- Exercise: Identify Promising Large Network Analyses of NIH Data

## Network Analysis and Visualization – General Workflow

**Original Data**

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A</td>
<td>C</td>
</tr>
<tr>
<td>2</td>
<td>B</td>
<td>D</td>
</tr>
<tr>
<td>3</td>
<td>A</td>
<td>C</td>
</tr>
<tr>
<td>4</td>
<td>B</td>
<td>D</td>
</tr>
</tbody>
</table>

**Extract Network**

Extract Bipartite Network was selected.

**Input Parameters**:
- First column: Source Node
- Text Delimiter: ;
- Second column: Target Nodes

**Calculate Node Attributes**

```
Nodes:
id int label string bipartitetype string indegree int outdegree int
1 "A" "Source Node" 0 3
2 "B" "Target Node" 3 0
3 "B" "Target Node" 3 0
4 "C" "Target Node" 2 0
5 "D" "Target Node" 2 0
6 "D" "Target Node" 2 0
7 "D" "Target Node" 1 0
8 "D" "Target Node" 0 1
```

**Visualization/Layout**

Diagram showing network visualization.
**Large Network Analysis & Visualization – General Workflow**

**Original Data**

Millions of records, in 100s of columns.
SAS and Excel might not be able to handle these files.
Files are shared between DB and tools as delimited text files (.csv).

**Derived Statistics**

Degree distributions
Number of components and their sizes
Extract giant component, subnetworks for further analysis

**Extract Network**

It might take several hours to extract a network on a laptop or even on a parallel cluster.

**Visualizations**

It is typically not possible to layout the network.
DrL scales to 10 million nodes.

---

**DrL Large Network Layout**

*See Section 4.9.4.2 in Sci2 Tutorial,*


DrL is a force-directed graph layout toolbox for real-world large-scale graphs up to 2 million nodes. It includes:

- Standard force-directed layout of graphs using algorithm based on the popular VxOrd routine (used in the VxInsight program).
- Parallel version of force-directed layout algorithm.
- Recursive multilevel version for obtaining better layouts of very large graphs.
- Ability to add new vertices to a previously drawn graph.

The version of DrL included in Sci2 only does the standard force-directed layout (no recursive or parallel computation).

How to use: DrL expects the edges to be *weighted* and *undirected* where the non-zero weight denotes how similar the two nodes are (higher is more similar). Parameters are as follows:

- The **edge cutting parameter** expresses how much automatic edge cutting should be done. 0 means as little as possible, 1 as much as possible. Around .8 is a good value to use.
- The **weight attribute parameter** lets you choose which edge attribute in the network corresponds to the similarity weight. The X and Y parameters let you choose the attribute names to be used in the returned network which corresponds to the X and Y coordinates computed by the layout algorithm for the nodes.

DrL is commonly used to layout large networks, e.g., those derived in co-citation and co-word analyses. In the Sci2 Tool, the results can be viewed in either GUESS or ‘Visualization > Specified (prefuse alpha)’.

See also [https://nwb.slis.indiana.edu/community/?n=VisualizeData.Drl](https://nwb.slis.indiana.edu/community/?n=VisualizeData.Drl)
Load isi formatted file

As csv, file looks like:

<table>
<thead>
<tr>
<th>A</th>
<th>Authors</th>
<th>Authors (Full Names)</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The systematic study of Colizza, VJ;Barat, AI;Barthelemy, M;Vespignani, A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Uncovering the hidden</td>
<td>Colizza, VJ;Flammini, AI;Serrano, MA;Vespignani, A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Computer viruses can</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Mapping the Internet</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>LECTURE NOTES IN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Visualize each time slide separately:
Relevant Sci2 Manual entry

Slice Table by Time

"Slice Into" allows the user to slice the table by days, weeks, months, quarters, years, decades, and centuries. There are two additional parameters for time slicing: cumulative and align with calendar. The former produces tables containing all data from the beginning to the end of each table's time interval, which can be seen in the Data Manager and below:

The latter option aligns the output tables according to calendar intervals:

Choosing "Years" under "Slice Into" creates multiple tables beginning from January 1st of the first year. If "Months" is chosen, it will start from the first day of the earliest month in the chosen time interval.
Visualize Each Network, Keep Node Positions

1. To see the evolution of Vespignani’s co-authorship network over time, check ‘cumulative’.

2. Extract co-authorship networks one at a time for each sliced time table using 'Data Preparation > Extract Co-Author Network', making sure to select "ISI" from the pop-up window during the extraction.

3. To view each of the Co-Authorship Networks over time using the same graph layout, begin by clicking on longest slice network (the 'Extracted Co-Authorship Network' under 'slice from beginning of 1990 to end of 2006 (101 records)') in the data manager. Visualize it in GUESS using 'Visualization > Networks > GUESS'.

4. From here, run 'Layout > GEM' followed by 'Layout > Bin Pack'. Run 'Script > Run Script …' and select 'yoursci2directory/scripts/GUESS/co-author-nw.py'.

5. In order to save the x, y coordinates of each node and to apply them to the other time slices in GUESS, select 'File > Export Node Positions' and save the result as 'yoursci2directory/NodePositions.csv'. Load the remaining three networks in GUESS using the steps described above and for each network visualization, run 'File > Import Node Positions' and open 'yoursci2directory/NodePositions.csv'.

6. To match the resulting networks stylistically with the original visualization, run 'Script > Run Script …' and select 'yoursci2directory/scripts/GUESS/co-author-nw.py', followed by 'Layout > Bin Pack', for each.

http://sci2.wiki.cns.iu.edu/5.1.2+Time+Slicing+of+Co-Authorship+Networks+(ISI+Data)

---

Relevant CIShell plugin

Slice Table by Time

http://cishell.wiki.cns.iu.edu/Slice+Table+by+Time
Visualize Each Network, Keep Node Positions

http://sci2.wiki.cns.iu.edu/5.1.2+Time+Slicing+of+Co-Authorship+Networks+(ISI+Data)

http://cns.iu.edu
CNS Facebook: http://www.facebook.com/cnscenter
Mapping Science Exhibit Facebook: http://www.facebook.com/mappingscience