Data Visualisation
CSE2002

Networks and Trees

- The most common visual encoding idiom for tree and network data is with nodelink diagrams
- nodes are drawn as point marks
- the links connecting them are drawn as line marks
- This idiom uses connection marks to indicate the relationships between items.

- Applications:
- Tournaments
- Organization Charts
- Genealogy
- Biological Interactions
- Social Networks etc

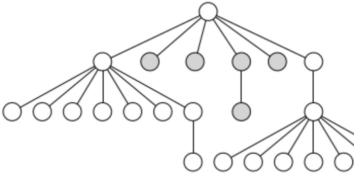
- Spatial layout:
- A primary concern of tree/graph drawing is the spatial arrangement of nodes and edges.
- Graph Structure:
- Connectivity, path-following
- Topological distance
- Clustering / grouping
- Ordering (e.g., hierarchy level)

Networks and Trees

- Networks are also very commonly represented as node - link diagrams, using connection.
- Nodes that are directly connected by a single link are perceived as having the tightest grouping
- nodes with a long path of multiple hops between them are less closely grouped.
- The number of hops within a path is a network-oriented way to measure distances.
- Node - link diagrams are useful to understanding the network topology:
- finding all possible paths from one node to another
- finding the shortest path between two nodes
- finding all the adjacent nodes one hop away from a target node
- finding nodes that act as a bridge between two components of the network that would otherwise be disconnected.

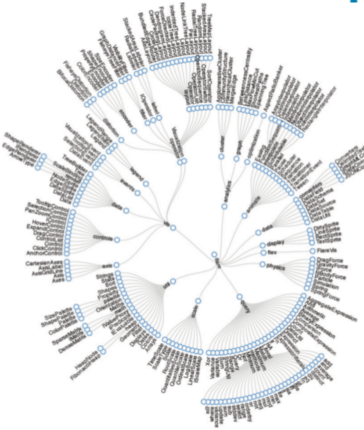
Node Link Diagrams

- Tiny tree of 24 nodes with triangular vertical node-link layout
- root at top and leaves at bottom
- it uses vertical spatial position channel to show the depth in the tree
- The horizontal spatial position of a node does not directly encode any attributes.
- the layout algorithm's calculations to ensure maximum possible information density while guaranteeing that there are no edge crossings or node overlaps



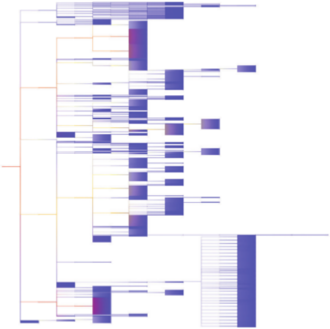
Node Link Diagrams

- small tree of a few hundred nodes with spline radial layout
- It also uses same algorithm for density without overlap
- The visual encoding is radial rather than rectilinear
- The depth of the tree is encoded as distance away from the center of the circle
- The Links of the graph are drawn as smoothly curving **splines**



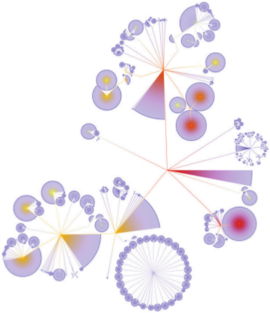
Node Link Diagrams

- A larger tree of 5161 nodes as a rectangular horizontal node-link diagram
- Root at left and leaves stretching out to right.
- The edges are colored with a purple to orange continuous colormap
- The spatial layout is fundamentally the same as the triangular layout
- zoomed-out position, the edges within a subtree form a single perceptual block where the spacing in between them cannot be seen.



Node Link Diagrams

- BubbleTree is a radial rather than rectilinear approach (same dataset)
- subtrees are laid out in full circles rather than partial circular arcs
- Spatial position does encode information about tree depth
- Relative distances to the center of the parent rather than as absolute distances in screen space



- Different Network layouts in Node-link Diagrams
  - Sugiyama-Tagawa-Toda algorithm
  - Force - Direct Layout Algorithm
  - Constraint Based Layout Algorithm

- Force - Direct layout
  - most widely used layout in networks
  - many variants are available
  - the network elements are positioned according to a simulation of physical forces where nodes push away from each other
  - The links act like springs that draw Force-directed placement their endpoint nodes closer to each other.
  - Very easy to implement.
  - Many Force-direct placement algorithms starts by placing the nodes randomly
  - Iteratively refine the positions of the nodes by pushing and pulling of simulated forces to gradually improve the layout.
  - It do not directly use spatial position to encode attribute values.
  - Few algorithms are designed to minimize the edge crossings and node overlaps
  - Analyzing the visual encoding created by force-directed placement is somewhat subtle

- Force - Direct layout
  - Analyzing the visual encoding created by force-directed placement is somewhat subtle.
  - Spatial position does not directly encode any attributes of either nodes or links;
  - Tightly interconnected group of nodes $\rightarrow$ Interpretation in two ways $\rightarrow$ proximity is sometimes meaningful but sometimes arbitrary;
  - Weakness: Non-deterministic $\rightarrow$ The problem with non-deterministic visual encodings is that spatial memory cannot be exploited across different runs of the algorithm.
  - Moreover, the randomness can lead to different proximity relationships each time.
  - Randomness is particularly tricky with dynamic layout.

- Force - Direct layout
  - A major weakness of force-directed placement is scalability, both in terms of the visual complexity of the layout and the time required to compute it.
  - **hairball** problem → of visual clutter with even hundred of nodes → task of path following or understanding overall structural relationships become very difficult → impossible with thousand of nodes
  - The nodes never settle down to a final location in simplest force-direct algorithm → they continue to bounce around if the user does not explicitly intervene to halt the layout process
  - More sophisticated algorithms automatically stop by determining that the layout has reached a good balance between the forces.
  - Computation is struck in local minima
  - occlusion

- Force - Direct layout

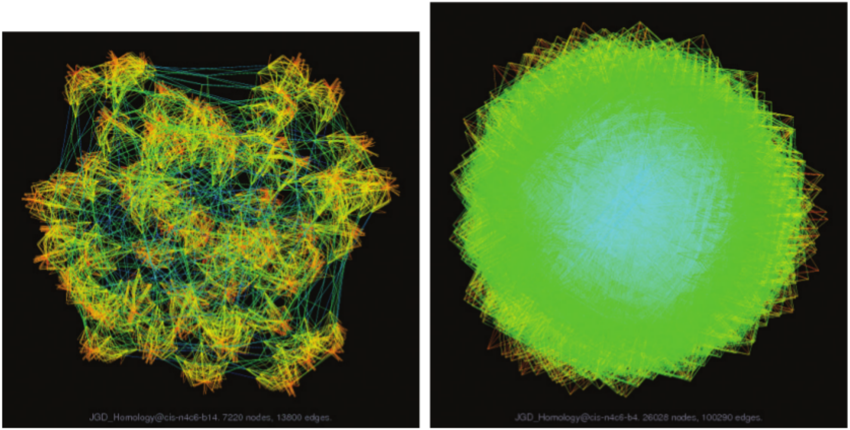
Idiom	Force-Directed Placement
What: Data	Network.
How: Encode	Point marks for nodes, connection marks for links.
Why: Tasks	Explore topology, locate paths.
Scale	Nodes: dozens/hundreds. Links: hundreds. Node/link density: $L < 4N$

- Force - Direct layout
  - Multilevel network Layout
    - Recent approaches to scalable network $\rightarrow$ multilevel network
    - The original network is augmented with a derived cluster hierarchy to form a compound network
    - The cluster hierarchy is computed by coarsening the original network into successively simpler networks
    - These approaches are better at avoiding the local minimum problem

Idiom	Multilevel Force-Directed Placement (sfdp)
What: Data	Network.
What: Derived	Cluster hierarchy atop original network.
What: Encode	Point marks for nodes, connection marks for links.
Why: Tasks	Explore topology, locate paths and clusters.
Scale	Nodes: 1000–10,000. Links: 1000–10,000. Node/link density: $L < 4N$.

- Force - Direct layout
  - Multilevel network Layout
  - The figure in the next slide shows a network of 7220 and 13,800 edges
  - It used the multilevel scalable force-directed placement (sfdp) algorithm
  - In this the edges are colored by length
  - Significant cluster structure is indeed visible in the layout
  - the dense clusters with short orange and yellow edges can be distinguished from the long blue and green edges between them.
  - Another network of 26028 nodes and 100290 edges with sfdp algorithm $\rightarrow$ hairball problem

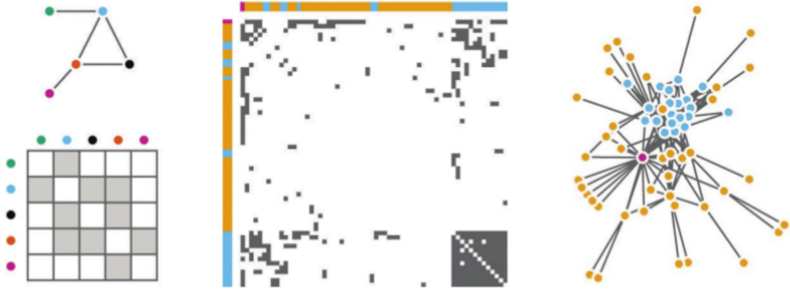
- Multilevel network Layout



(a) (b)

- Matrix Views → Adjacency Matrix Views
  - Network data can also be encoded with a matrix view by deriving a table from the original network data.
  - where all of the nodes in the network are laid out along the vertical and horizontal edges of a square region and links between two nodes are indicated by coloring an area mark in the cell in the matrix that is the intersection between their row and column
  - That is, the network is transformed into the derived dataset of a table with two key attributes that are separate full lists of every node in the network, and one value attribute for each cell records whether a link exists between the nodes that index the cell.

- Adjacency Matrix Views



(a) (b) (c)

- Matrix Views → Adjacency Matrix Views
  - size coding matrix cells is limited by the number of available pixels per cell → only a few levels would be distinguishable between the largest and the smallest cell size
  - Network matrix views can also show weighted networks → each link has an associated quantitative value attribute, by encoding with an ordered channel such as luminance or size.
  - Only half of the matrix is enough for symmetric undirected networks
  - Full Matrix view is required for directed networks

Idiom	**Adjacency Matrix View**
What: Data	Network.
What: Derived	Table: network nodes as keys, link status between two nodes as values.
How: Encode	Area marks in 2D matrix alignment.
Scale	Nodes: 1000. Links: one milllion.

- Connection Vs Matrix
  - supporting many of the abstract tasks that pertain to small network data.
  - It gives the solutions for many tasks which are rely on topological structure of the network
  - It also identifies the solutions to the tasks like general overview or finding similar substructures.
  - Limitation: network size and link density, occlusion
  - The link density of a network is the number of links compared with the number of nodes.
  - The upper limit for nodelink diagram effectiveness is a link density of around three or four.
  - Even for the networks with link density below four, it can have unreadable hairball due to increase in network size.
  - Filtering, aggregation, and navigation are design choices that can ameliorate the clutter problem $\rightarrow$ but they do impose cognitive load on the user who must then remember the

- Connection Vs Matrix
  - A major strength of matrix views is perceptual scalability for both large and dense networks.
  - Matrix views completely eliminate the occlusion of node-link views
  - a single-level matrix view can handle up to one million edges and an aggregated multilevel matrix view might handle up to ten billion edges.
  - Another strength of matrix views is their predictability, stability and support for reordering.
  - Matrix views use less screen space where as node-link views takes more screen space based on dataset characteristics.
  - matrix views are stable; adding a new item will only cause a small visual change.
  - it supports the tasks related to estimating the number of nodes in a graph, fast searching through fast node lookup, finding an item label in ordered list, etc

- Connection Vs Matrix
  - major weakness of matrix views is unfamiliarity $\rightarrow$ training is needed to interpret matrix views.
  - crucial weakness of matrix views is their lack of support for investigating topological structure
  - nodelink views are best for small networks
  - matrix views are best for large networks