Mapping Networks

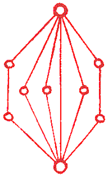
Gonzalo Mateos
Dept. of ECE and Goergen Institute for Data Science
University of Rochester
gmateosb@ece.rochester.edu
http://www.hajim.rochester.edu/ece/sites/gmateos/

January 31, 2023

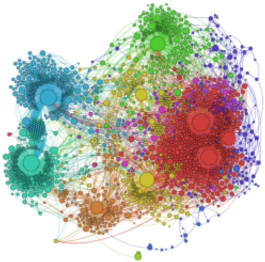
Outline

Network mapping

- Visual imagery key to network analysis as in other quantitative sciences



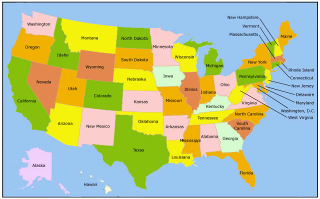
Pattern of social linkages [Moreno'32]

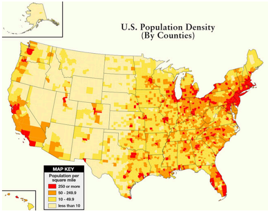


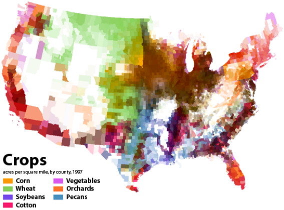
- Hand-drawn, annotated graphs $\Rightarrow$ Computerized, automated diagrams
- Q: What is network mapping?
  - The production of a network-based visualization of a complex system
  - Analogy: Geography and the production of cartographic maps

What is "the" network?

- Often not a single network graph representation of a given system







Ex: Which of these maps best depicts the USA?

Visualization challenges

- Suppose a graph representation $G(V, E)$ of a complex system is given

Network graph visualization

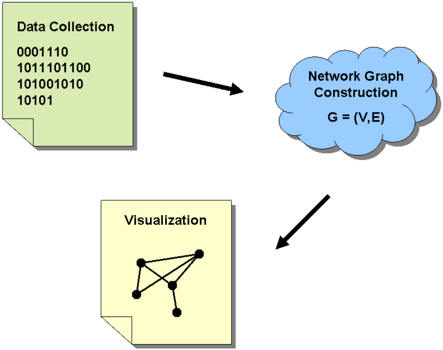
A visualization of G is a mapping $\phi : (V, E) \mapsto \mathbb{R}^2$ (or $\mathbb{R}^3$)

- Several nontrivial graph visualization challenges
  - Lack of inherent geometry in G, just two sets V and E
  - Plenty of degrees of freedom and flexibility in specifying ϕ
  - Convey patterns in high-dimensional data. Summarization and scale
  - A diverse range of information that may be communicated, or lost
- Arguably, graph visualization is a quite young, active area of research

  $\Rightarrow$ Mathematics, algorithms, aesthetics, the human visual system

Stages in network mapping

- Three key stages in the production of network maps



S1: Collection of relational data from the system of interest

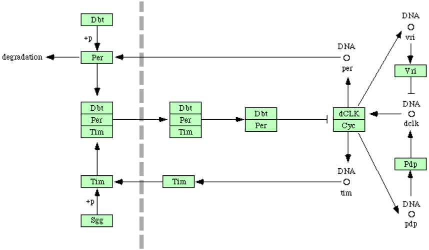
S2: Construction of the network graph representation

S3: Rendering of the representation as a visual image

Collecting relational network data

Measuring elements and interactions

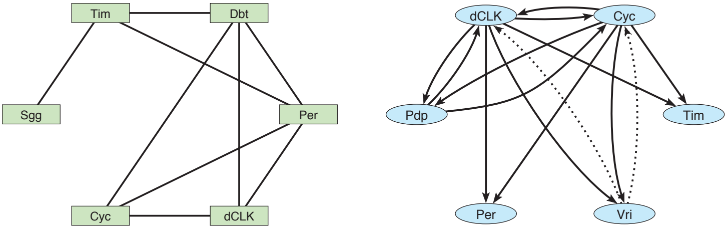
- Start with measurements of system ‘elements’ and ‘interactions’



Drosophila’s circadian rhythm

- Choose what is meant by elements and interactions
  Ex: Proteins and their affinity to bind, or genes and their regulation
- Decide what measurements to take for each
  Ex: Protein affinity experiments, or DNA micro-array experiments
- Choices influence the network graphs that may be constructed

- Related notions of system elements can yield markedly different graphs



Protein interaction network

Gene regulation network

- Ex: Protein *Per* interacts with four other proteins; while Gene coding for *Per* regulates none of the other genes directly
- Each one provides a partial view of the underlying biological system
  - ⇒ Choices a fortiori affect analyses performed and conclusions drawn

Further choices

- There may be different scales at which elements could be labeled

  Ex: Users, routers, autonomous systems (ASs) in Internet studies?

  Ex: Authors, papers, journals, disciplines in citation studies?

- Measures of interaction can take many forms (binary, counts, real)

  Ex: Friendship networks in social network analysis

  - Interview and ask about friendship with other actors (binary)
  - Measure frequency of relations e.g., SMS (counts)

- Questions directly measure the interaction. SMS do indirectly
- Not only what we choose (or are capable of) to measure is important

  $\Rightarrow$ Also is, potentially, what remains unmeasured in the system

Enumerated, partial and sampled data

- Assuming full-accessibility to network data may be overly optimistic
- **Enumerated data:** Collected exhaustively from the full population

  Ex: Social network studies in small groups (clubs, high-schools, . . .)

  Ex: Exhaustive scientific publication databases for citation analyses
- **Partial data:** Full enumeration of only a subset of the population

  Ex: Geographical sub-network or AS of an Internet Service Provider
- **Sampled data:** Selected from the population via a random scheme

  $\Rightarrow$ Sampling is often the rule rather than the exception (More later)

  Ex: Random probing of source-destination pairs in the Internet

  Ex: Social network studies about illegal drug usage, or prostitution

Constructing network graph representations

From measurements to a graph

- Basic goal is specification of $G(V, E)$ from measurements
- The representation may include additional information
  - Edge weights: $\{w_e\}_{e \in E}$ indicating the strength of association
  - Vertex vectors: $\{\mathbf{x}_v\}_{v \in V}$ describing element attributes or labels
- Attribute variables may be discrete or continuous in nature

  Ex: Gender, infection status, population serviced by an airport
- This information we seek to effectively convey in a network map

Specification of vertices and edges

- Measurements may be direct declarations of edge/non-edge status
- Most commonly, edges dictated after processing measurements
  - Comparison of vertex similarity metric to a threshold
  - Frequently ad hoc, sometimes formal methods (topology inference)
- Q: How to address the "ball-of-yarn" phenomenon in visualizations?

- Effective use of scale, node aggregation and thinning of edges
  - Rooted sub-trees or DAGs may be trimmed, hiding inner structure
  - Split dense graph into separate subgraphs based on labels, clustering
- Ex: Associate genes or proteins with their biological functions

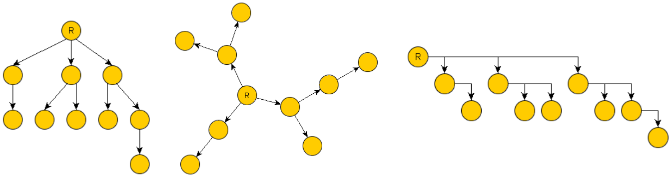
Visualizing network graphs

Elements of graph visualization

- Goal: embed a combinatorial object $G(V, E)$ into 2-D (3-D) space
  $\Rightarrow$ Use symbols (e.g., circles) for vertices, smooth curves for edges
- Uncountably many options, inherently ill-posed
- Q: Does it adequately communicate the relational information in G?
  $\Rightarrow$ Guide drawing process by adding specifications and requirements
- **Drawing conventions:** hard requirements a drawing must satisfy
  Ex: Edges as straight lines, no edges intersect, downward trees, ...
- **Aesthetics:** soft requirements, satisfied if possible
  Ex: Minimize edge crossings, total area, edge bends, ...
- **Constraints:** requirements that pertain to subgraphs $H \subset G$
  Ex: Placement of a specific vertex or cluster, direction of a path, ...

Drawing graphs with special structure

- Structures that receive most attention: planar graphs and trees
- Two common, linear complexity methods for planar graphs
  - Use orthogonal paths for edges (e.g., canonical in integrated circuits)
  - Use k-sided convex polygons for each cycle of length k
- While also planar, structure of trees justifies additional methods

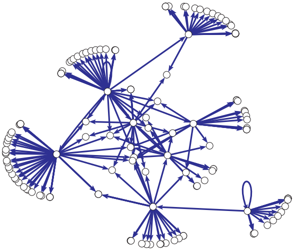


- Often a hierarchical structure is to be communicated

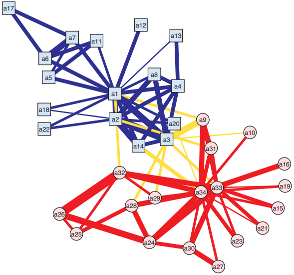
  Ex: Organizational charts, genealogies, information cascades, ...

Drawing using analogies to physical systems

- In the absence of structure, exploit analogies to physical systems
  - Convey relations via “likes $\leftrightarrow$ attraction” and “dislikes $\leftrightarrow$ repulsion”
- Spring-embedder methods view vertices as masses, edges as springs
  - Perturb and let forces converge, particle system reaches equilibrium



Spring embedder



Energy placement

- Energy-placement methods define energy function of vertex positions
  - Minimize system energy to place vertices, reach most relaxed state

Energy placement via multidimensional scaling

- Multidimensional scaling (MDS) commonly used for visualization
- Given pairwise vertex dissimilarities $\{\delta_{ij}\}$ (e.g., geodesic distances)

  $\Rightarrow$ Goal: Find $\{\mathbf{x}_i \in \mathbb{R}^2\}_{i=1}^{N_v}$ so that $\|\mathbf{x}_i - \mathbf{x}_j\|_2 \approx \delta_{ij}$
- Approach: MDS stress (energy function) minimization

$$\underset{\{\mathbf{x}_1,\ldots,\mathbf{x}_{N_v}\}}{\arg\min} \quad \frac{1}{2}\sum_{i=1}^{N_v}\sum_{j=1}^{N_v}\left(\delta_{ij} - \|\mathbf{x}_i - \mathbf{x}_j\|_2\right)^2$$

  $\Rightarrow$ Nonconvex cost. Typically "solved" via gradient descent
- May include structural constraints e.g., vertex centralities
- B. Baingana and G. B. Giannakis, "Centrality-constrained graph embedding," in *ICASSP*, 2013.

Graph layout: Science and art

- Graph visualization software use a handful of standard methods

  Ex: Circular, radial, analogies to physical systems, ...

- Many graph layout packages, some general and some area specific

  Ex: Gephi, Pajek, Graphviz, LaNet-vi, ...

  $\Rightarrow$ I have listed a few under resources in the class website

- Best ones allow for user interaction to manipulate further

  $\Rightarrow$ Graph drawing involves not only science but also some art

- Few computer-generated drawings cannot be improved "by hand"

Case study

Mapping “Science”

- The human enterprise of Science and Technology, i.e., “Science”
- Understand patterns and associations in its growth and development
  - $\Rightarrow$ Goal of the field known as scientometrics
  - $\Rightarrow$ Interests government agencies, industry, sciences themselves

Ex: Network representation and visualization of “Science”?

- K. W. Boyack, R. Klavans, and K. Börner, “Mapping the backbone of science,” *Scientometrics,* vol. 64, no. 3, pp. 351-371, 2005.
- Go over measurement, network graph construction and visualization

Measurement

- **System:** Science as summarized through the archival literature
- **Elements:** authors, articles, journals, communities
- **Interactions:** inter-citation frequencies among journals over time

$$C_{ij} = \text{Number of times journal } i \text{ cites } j \text{ in e.g., one year}$$

- Q: Partial sampling impact?

  $\Rightarrow$ Conference proceedings in Computer Science

- Data from the Institute of Scientific Information (ISI) databases
  - 1.058M articles from 7,349 journals for the year 2000
  - 23.08M total citations, 16.24M among the database journals
  - Computed matrix of inter-citations C_{ij} very sparse (98.6% zeros)

Network graph construction

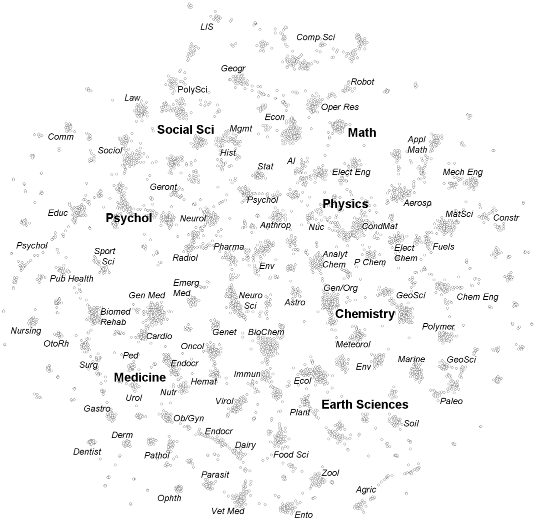
- $G(V, E)$ can be defined directly from the inter-citation matrix
  - $\Rightarrow$ Vertices correspond to the 7,121 citing or cited journals
  - $\Rightarrow$ Edge (i, j) joins journals i and j if $C_{ij} + C_{ji} > 0$
- Validation: found journal clusters not matching human expectation
- Use the Jaccard inter-citation frequency measure to define edges

$$JAC_{ij} = JAC_{ji} = \frac{C_{ij} + C_{ji}}{\sum_{k \neq j} C_{ik} + \sum_{k \neq i} C_{jk}}$$

- Trim weaker edges such that degrees are upper-bounded by 15

Preliminary visualization

- Software package used: VxOrd (Sandia Labs)
- Spring-embedder algorithm
  - Linear complexity $O(N_v)$
  - Edge-cutting criteria
- Journals tend to cluster
  - Densely inter-connected
  - Few ties among clusters
- Manually assigned labels
  - Clusters ⇒ ISI categories
- No edges for readability



Further post-processing

- Goal is to obtain a map at the level of scientific disciplines

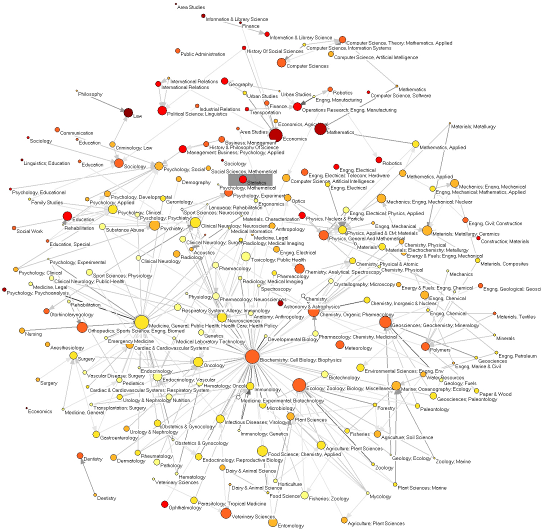
1) Each discipline cluster replaced with a single vertex

- Vertex size $\propto$ number of journals in the cluster
- Vertex color $\propto$ relative frequency of self-citation within discipline
  - Darker vertices suggest more independent disciplines

2) Placed arcs joining pairs of vertices (disciplines)

- Draw arc (i, j) if 7.5% or more of all citations from i were to j
  - Darker edges represent higher percentages
- VxOrd places highly-connected vertices closer to the center

The backbone of Science

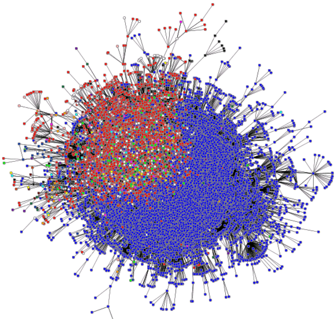


- Backbone of Science: final map at the level of scientific disciplines

Large-scale network visualization

Large-scale network visualization

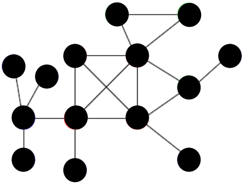
- Many interesting networks are large and complex
  - $\Rightarrow$ Difficult to visualize
  - $\Rightarrow$ Computationally intensive
  - $\Rightarrow$ Structure hindered
- Ex: The blogosphere with > 1M nodes



- Idea: Use the k-core decomposition for hierarchical visualization

The k-core decomposition

- Consider a given graph $G(V, E)$

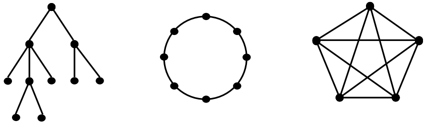


- **Def:** An induced subgraph $G'(V', E')$ of G is a k-core if $d_v(G') \geq k$ for all $v \in V'$, and G' is maximal
- Degrees are in the induced subgraph G', not in G
- Hierarchy: larger "coreness" $\Rightarrow$ larger degrees and centrality
- Algorithm: recursively prune all vertices of degree less than k

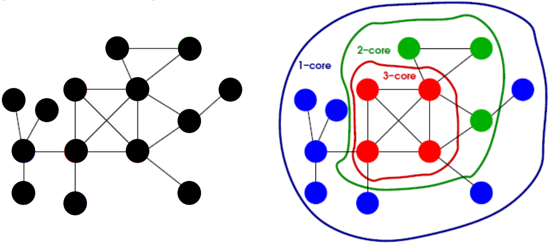
  $\Rightarrow$ Complexity $O(N_v + N_e)$, very efficient for sparse graphs

Example: k-core decompositions

- Ex: Trees are 1-cores, cycles are 2-cores, K_n is a $(n-1)$-core



- Ex: A graph with multiple cores



$\Rightarrow$ A k-core is always included within the $(k-1)$-core

$\Rightarrow$ While some vertices have $d_v(G) = 4$, the 4-core is empty

Preliminary definitions

- Vertex i has coreness $c_i = c$ if $i \in c$-core, but $i \notin (c+1)$-core
- A shell C_c comprises all vertices with coreness c
  - $\Rightarrow$ The maximum value of c such that $C_c \neq \emptyset$ is c_{max}
  - $\Rightarrow$ The k-core is a disjoint union of shells

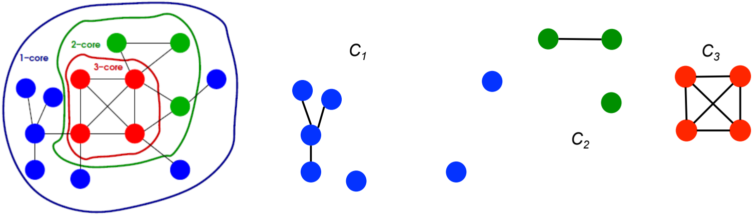
$$k\text{-core} = \bigcup_{k \leq c \leq c_{\text{max}}} C_c$$

- Each connected set of vertices having coreness c is a cluster Q^c
  - $\Rightarrow$ The maximum number of clusters in a shell C_c is q^c_{max}
  - $\Rightarrow$ Each shell is a disjoint union of clusters

$$C_c = \bigcup_{1 \leq m \leq q^c_{\text{max}}} Q^c_m$$

Example

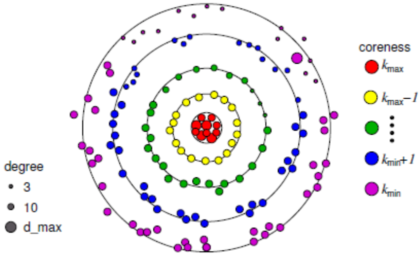
- Blue vertices have coreness $c = 1$, green have $c = 2$, red have $c = 3$
  $\Rightarrow$ Here $c_{\max} = 3$ and shells $\{C_c\}_{c=1}^{3}$ are shown in the right



- All three k-cores are connected, while shells C_1 and C_2 are not
  $\Rightarrow$ Shell C_1 has $q^1_{\max} = 4$ clusters, $q^2_{\max} = 2$ and $q^3_{\max} = 1$

Visualization using the k-core decomposition

- Given $G(V, E)$ determine the polar coords. $\rho_i \angle \varphi_i$ of each $i \in V$



- Key features of the visualization algorithm. For vertex i:
  - Radius ρ_i depends on c_i, and coreness of neighbors $V_{c_j \geq c_i}(i)$
  - Angle φ_i depends on cluster number q_i within shell C_{c_i}
  - Color depends on coreness c_i (e.g., 1 is violet, $c_{\max}$ is red)
  - Diameter is $\propto \log d_i$

Algorithm

- The k-core decomposition of $G(V, E)$ is an input to the algorithm

  $\Rightarrow$ Each vertex $i \in V$ has attributes $[c_i, q_i]^T$, such that $i \in Q_{q_i}^{c_i}$

- Radius ρ_i of vertex i is given by

$$\rho_i = (1-\epsilon)(c_{\max} - c_i) + \frac{\epsilon}{|V_{c_j \geq c_i}(i)|} \sum_{j \in V_{c_j \geq c_i}(i)} (c_{\max} - c_j)$$

  $\Rightarrow$ Parameter $\epsilon \in (0, 1)$ controls potential ring overlap

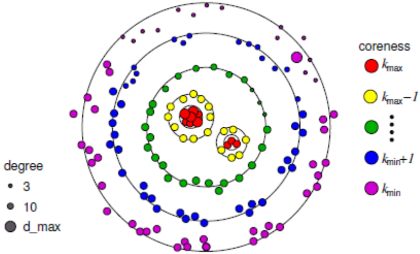
- Angle φ_i is random, with Normal distribution

$$\varphi_i \sim \mathcal{N}\left(\pi \frac{|Q_{q_i}^{c_i}|}{|C_{c_i}|} + \sum_{1 \leq m < q_i} 2\pi \frac{|Q_m^{c_i}|}{|C_{c_i}|}, \pi \frac{|Q_{q_i}^{c_i}|}{|C_{c_i}|}\right)$$

  $\Rightarrow$ Angular sector $[0, 2\pi]$ is partitioned among the $q_{\max}^{c_i}$ clusters

Fragmented k-cores

- In general, one may obtain disconnected (fragmented) k-cores



- The general algorithm can reveal such structure. For details, see:
- J. I. Alvarez-Hamelin et al, "Large scale networks fingerprinting and visualization using the k-core decomposition," in *NeurIPS*, 2005

Case study

Mapping the Internet

- A single, comprehensive map of the Internet is lacking. Reasons:
  - Dynamic and self-organized nature
  - Proprietary and security constraints among service providers
  - Sheer size
- What is "the" Internet?
  - The physical infrastructure
  - Logical paths of information flow over that infrastructure
  - The content underlying that information
  - Usage patterns of those disseminating, consuming that content
  - Traffic created by such usage

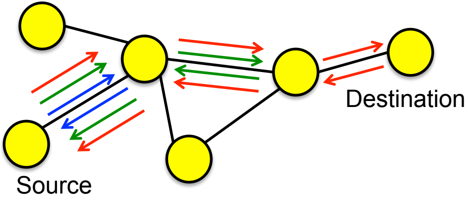
Ex: Hierarchical visualization of the Internet's logical structure?

- Go over measurement, network graph construction and visualization

Measurement

- **System:** logical Internet, paths over which packets are routed
- **Elements:** used routers, aggregations e.g., autonomous systems (AS)
- **Interactions:** router connections, effective connections between ASs

  ⇒ Large-scale measurement via probing, e.g., `traceroute`

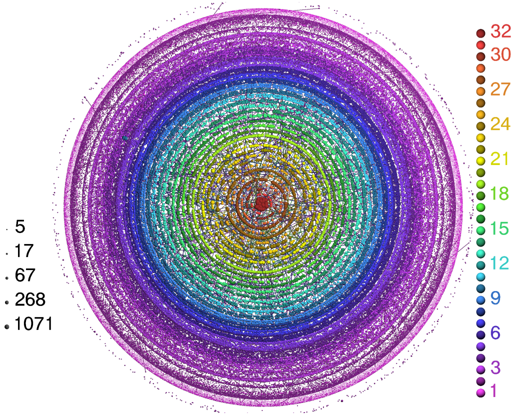


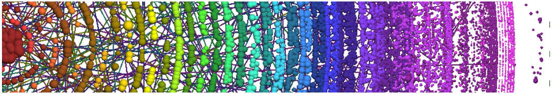
- Data by the Cooperative Assoc. for Internet Data Analysis (CAIDA)
  - Use the Skitter topology project. 20 worldwide measurement centers
  - Sends 800k `traceroute`-like probes to suitably spread destinations
  - Measurements taken from April 21 to May 3, 2003

Network graph construction

- $G(V, E)$ can be inferred from sequences of `traceroute` probes
  - Use paths from a source to construct trees (or DAGs)
  - Merge collections of trees from multiple sources to form G
- Vertices correspond to the 192,244 discovered routers
- The 609,066 edges join routers along the discovered paths
- Caveat on a few practical difficulties
  - **Asymmetric routing:** Studies realistically produce directed paths
  - **Time sensitivity:** Merge paths that changed (disappeared) over time
  - **Multiple interfaces:** Router may be discovered via multiple "aliases"
  - **Security policies:** Firewalls "hide" the topology behind them

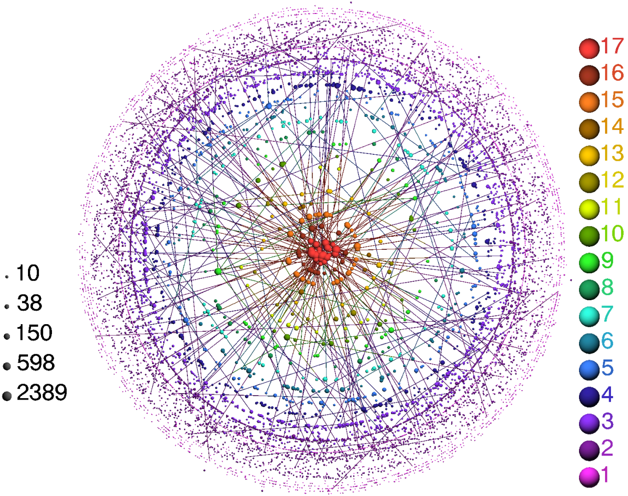
The router-level Internet





- Hierarchical structure of the Internet using *k*-core decomposition

The AS-level Internet



- Data from the University of Oregon Route Views Project

Glossary

- Network mapping
- Graph summarization
- Elements and interactions
- Scale
- Measurements of relation
- Enumerated and sampled data
- Vertex similarity
- "Ball-of-yarn" phenomenon
- Graph embedding
- Drawing conventions
- Aesthetics
- Spring-embedder methods
- Energy-placement methods
- Scientometrics
- Jaccard inter-citation frequency
- k-core decomposition
- Vertex coreness
- k-shell and k-core
- Physical and logical Internet
- `traceroute` probing