Topic II: Graph Mining

Discrete Topics in Data Mining
Universität des Saarlandes, Saarbrücken
Winter Semester 2012/13

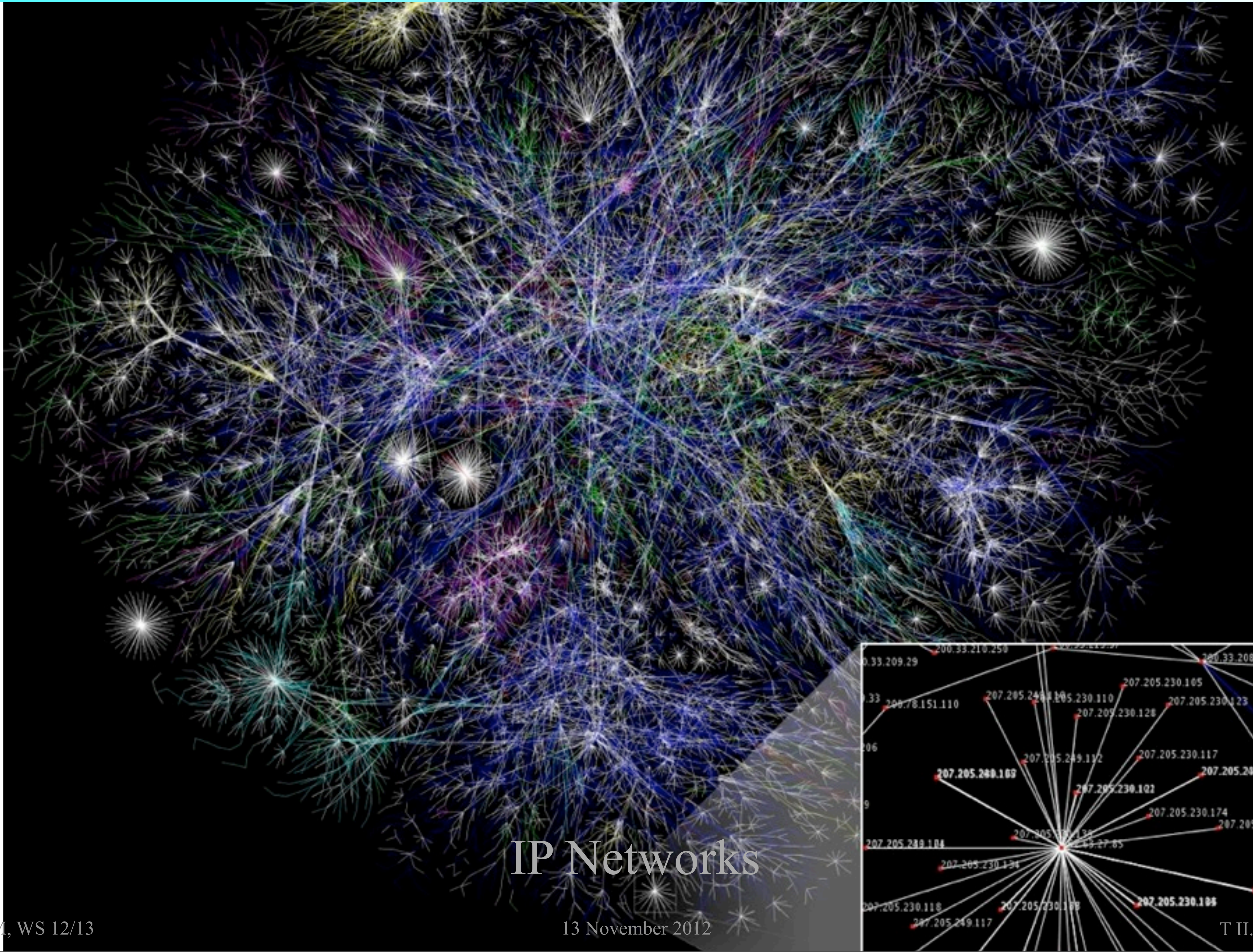
Topic II Intro: Graph Mining

1. Why Graphs?
2. What is Graph Mining
3. Graphs: Definitions
4. Centrality
5. Graph Properties
    5.1. Small World
    5.2. Scale Invariance
    5.3. Clustering Coefficient
6. Random Graph Models

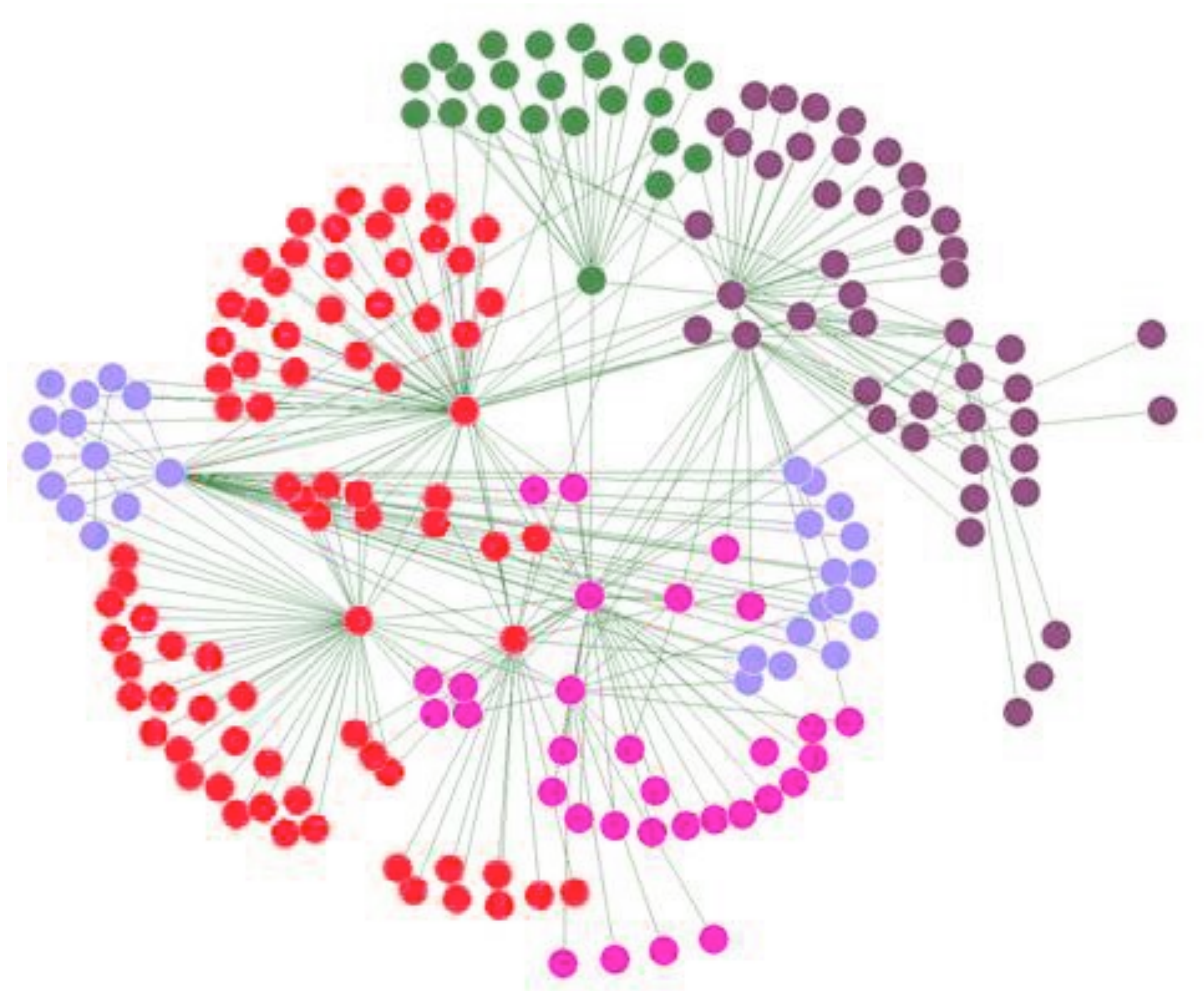
Z&M, Ch. 4

Why Graphs?

Why Graphs?

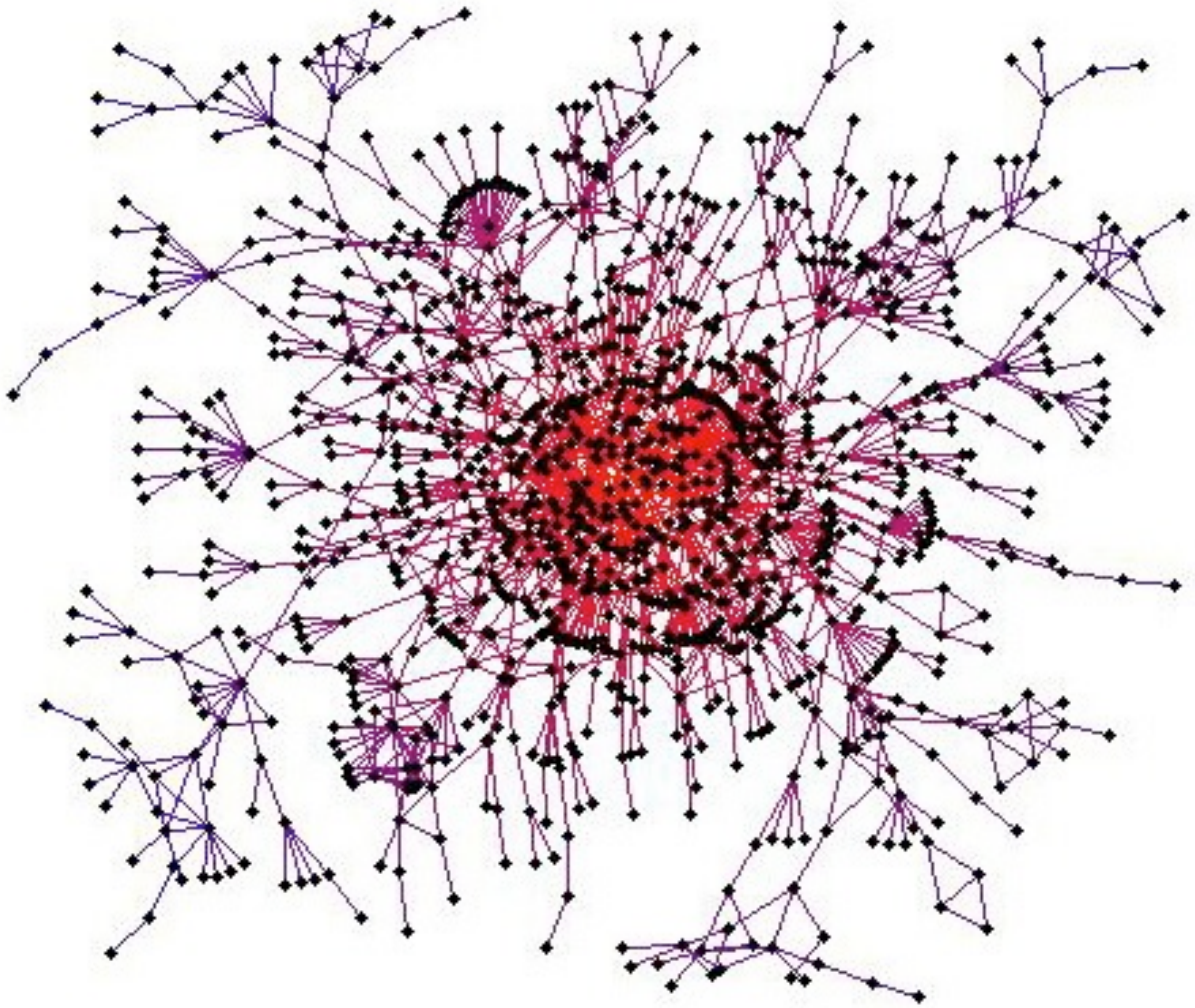

IP Networks
207.205.230.105
207.205.230.128
207.205.230.117
207.205.230.174
200.78.151.110

Why Graphs?



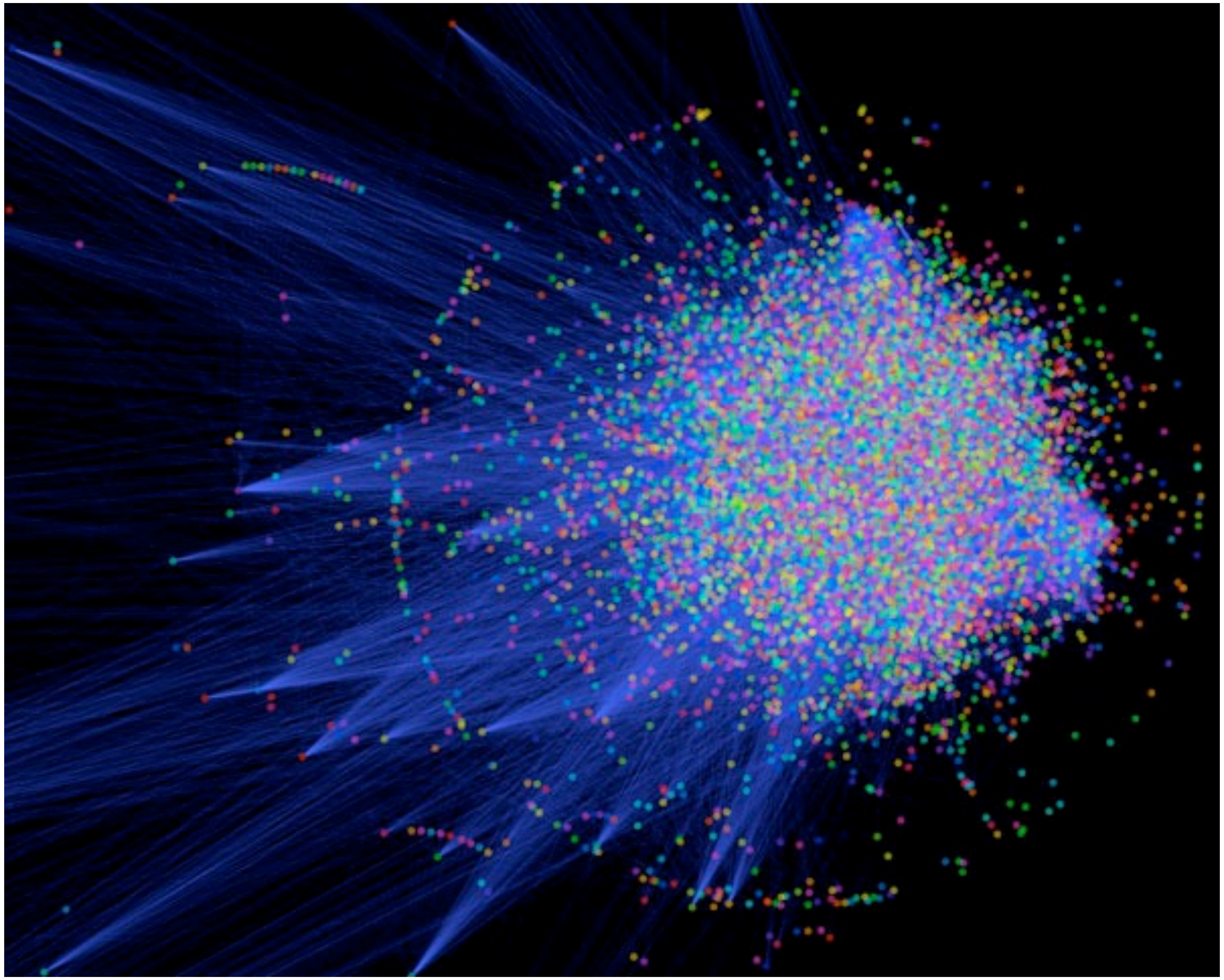
Social Networks

Why Graphs?



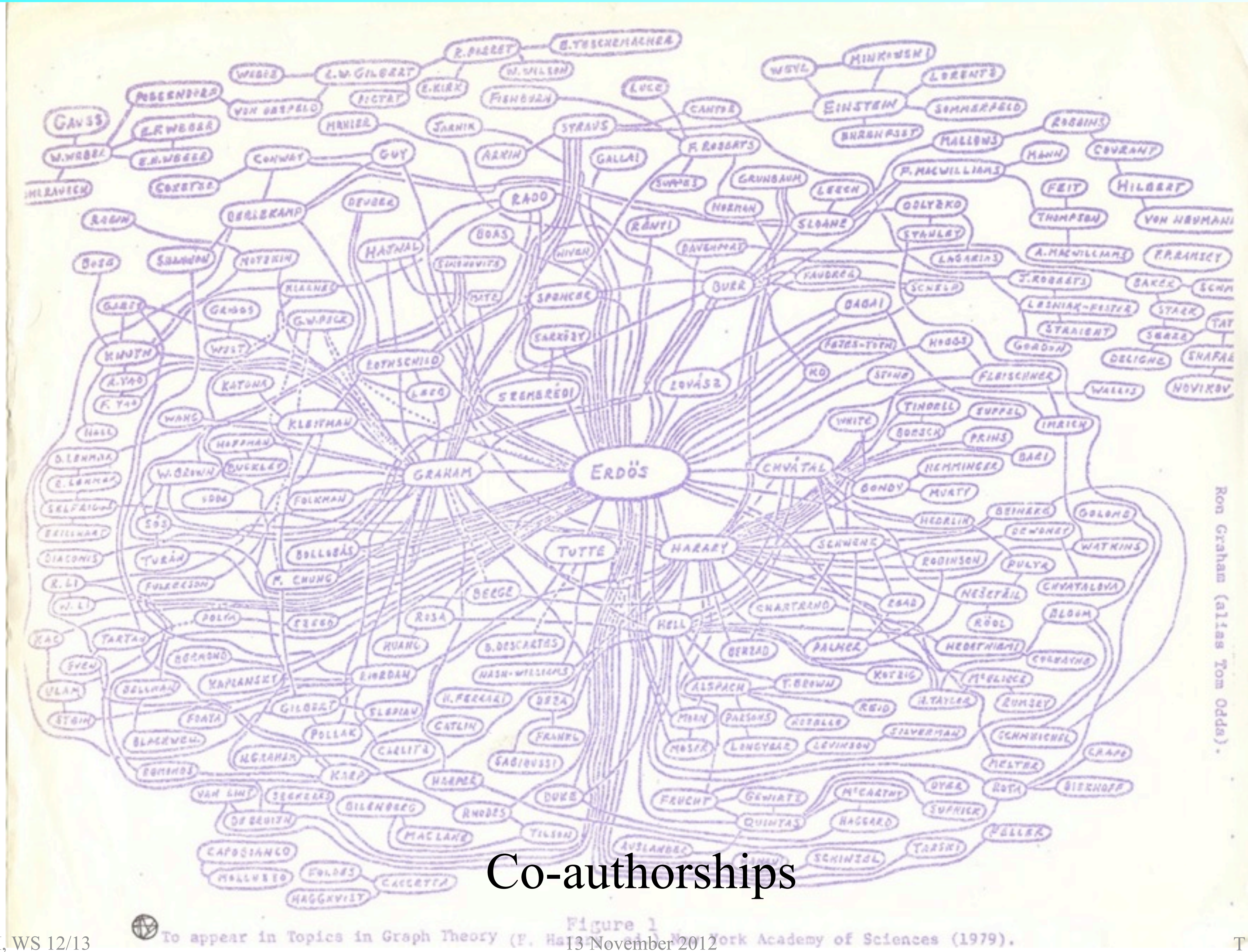
World Wide Web

Why Graphs?



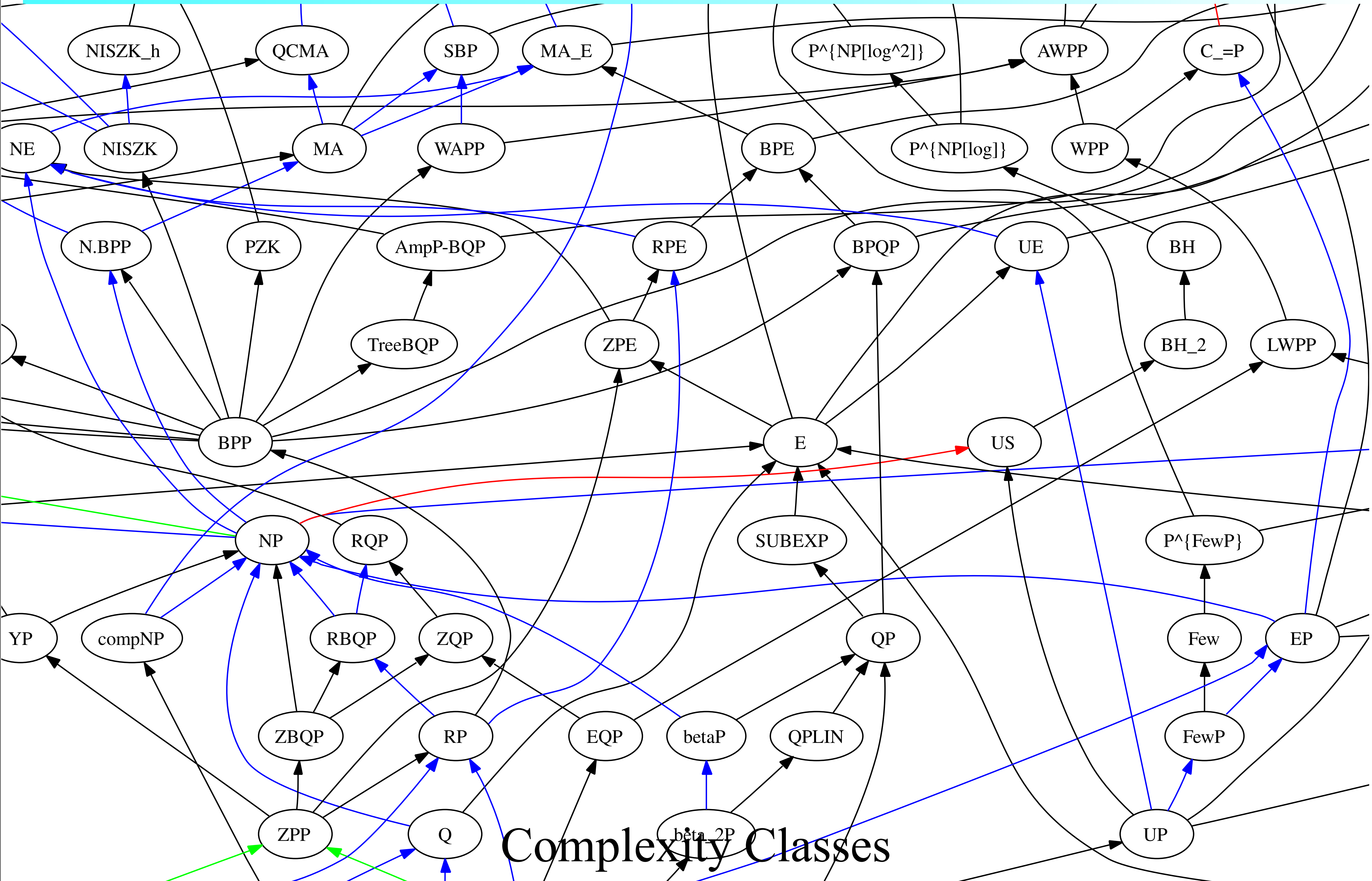
Protein–Protein Interactions

Why Graphs?



Co-authorships

Why Graphs?



Complexity Classes

Why Graphs?

Graphs are Everywhere!

Graphs: Definitions

- An **undirected graph** G is a pair (V, E)
  - $V = \{v_i\}$ is the set of **vertices**
  - $E = \{e_i = \{v_i, v_j\} : v_i, v_j \in V\}$ is the set of **edges**
- In **directed graph** the edges have a direction
  - $E = \{e_i = (v_i, v_j) : v_i, v_j \in V\}$
- And edge from a vertex to itself is *loop*
  - A graph that does not have loops is *simple*
- The **degree** of a vertex v, $d(v)$, is the number of edges attached to it, $d(v) = |\{\{v, u\} \in E : u \in V\}|$
  - In directed graphs vertices have *in-degree* $id(v)$ and *out-degree* $od(v)$

Subgraphs

- A graph $H = (V_H, E_H)$ is a **subgraph** of $G = (V, E)$ if
  - $V_H \subseteq V$
  - $E_H \subseteq E$
  - The edges in E_H are between vertices in V_H
- If $V' \subseteq V$ is a set of vertices, then $G' = (V', E')$ is the **induced subgraph** if
  - For all $v_i, v_j \in V'$ such that $\{v_i, v_j\} \in E$, $\{v_i, v_j\} \in E'$
- Subgraph $K = (V_K, E_K)$ of G is a **clique** if
  - For all $v_i, v_j \in V_K$, $\{vi, vj\} \in E_K$
  - Cliques are also called *complete subgraphs*

Bipartite Graphs

- A graph $G = (V, E)$ is **bipartite** if V can be partitioned into two sets U and W such that
  - $U \cap W = \emptyset$ and $U \cup W = V$ (a *partition*)
  - For all $\{v_i, v_j\} \in E$, $v_i \in U$ and $v_j \in W$
    - No edges within U and no edges within W
- Any subgraph of a bipartite graph is also bipartite
- A **biclique** is a complete bipartite subgraph $K = (U \cup V, E)$
  - For all $u \in U$ and $v \in V$, edge $\{u, v\} \in E$

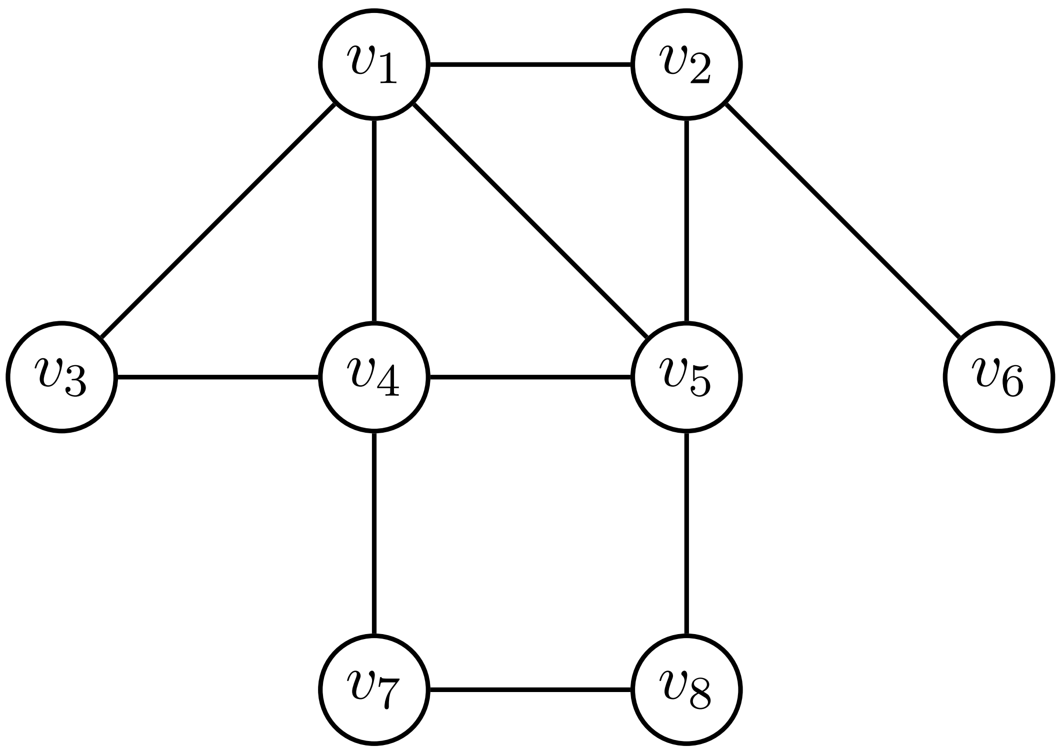
Paths and Distances

- A **walk** in graph G between vertices x and y is an ordered sequence $\langle x = v_0, v_1, v_2, \ldots, v_{t-1}, v_t = \mathrm{y}\rangle$
  - $\{v_{i-1}, v_i\} \in E$ for all $i = 1, \ldots, t$
  - If $x = y$, the walk is *closed*
  - The same vertex can re-appear in the walk many times
- A *trail* is a walk where *edges* are distinct
  - $\{v_{i-1}, v_i\} \neq \{v_{j-1}, v_j\}$ for $i \neq j$
- A *path* is a walk where *vertices* are distinct
  - $v_i \neq v_j$ for $i \neq j$
  - A closed path with $t \geq 3$ is a *cycle*
- The **distance** between x and y, $d(x, y)$ is the length of the *shortest* path between them

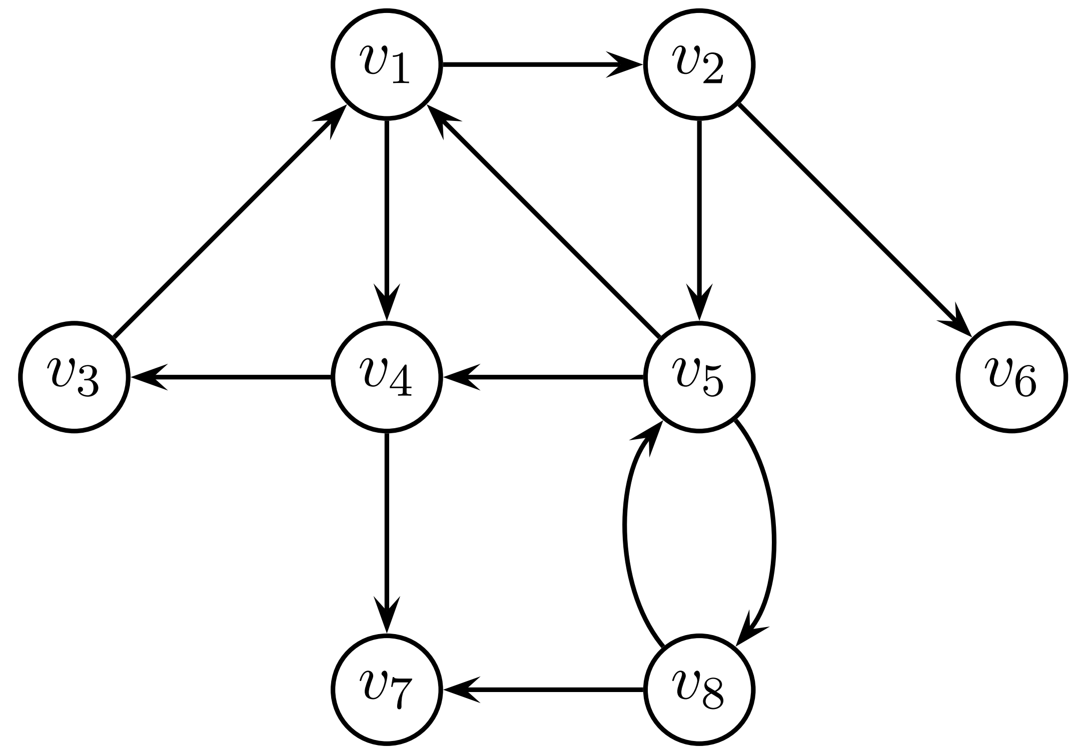
Connectedness

- Two vertices *x* and *y* are **connected** if there is a path between them
  - A graph is connected if all pairs of its vertices are connected
- A **connected component** of a graph is a maximal connected subgraph
- A directed graph is **strongly connected** if there is a directed path between all ordered pairs of its vertices
  - It is **weakly connected** if it is connected only when considered as an undirected graph
- If a graph is not connected, it is **disconnected**

Example



(a)



(b)

Adjacency Matrix

- The **adjacency matrix** of an undirected graph $G = (V, E)$ with $|V| = n$ is the n-by-n symmetric binary matrix $\boldsymbol{A}$ with
  - $a_{ij} = 1$ if and only if $\{v_i, v_j\} \in E$
  - A *weighted* adjacency matrix has the weights of the edges
- For directed graphs, the adjacency matrix is not necessarily symmetric
- The **bi-adjacency matrix** of a bipartite graph $G = (U \cup V, E)$ with $|U| = n$ and $|V| = m$ is the n-by-m binary matrix $\boldsymbol{B}$ with
  - $b_{ij} = 1$ if and only if $\{u_i, v_j\} \in E$

Topological Attributes

- The *weighted degree* of a vertex v_i is $d(v_i) = \sum_j a_{ij}$
- The *average degree* of a graph is the average of the degrees of its vertices, $\Sigma_i \, d(v_i)/n$
  - Degree and average degree can be extended to directed graphs
- The *average path length* of a connected graph is the average of path lengths between all vertices

$$\sum_i \sum_{j>i} d(v_i, v_j) / \binom{n}{2} = \frac{2}{n(n-1)} \sum_i \sum_{j>i} d(v_i, v_j)$$

Eccentricity, Radius & Diameter

- The **eccentricity** of a vertex v_i, $e(v_i)$, is its maximum distance to any other vertex, $\max_j\{d(v_i, v_j)\}$
- The **radius** of a connected graph, $r(G)$, is the minimum eccentricity of any vertex, $\min_i\{e(v_i)\}$
- The **diameter** of a connected graph, $d(G)$, is the maximum eccentricity of any vertex, $\max_i\{e(v_i)\} = \max_{i,j}\{d(v_i, v_j)\}$
  - The *effective diameter* of a graph is smallest number that is larger than the eccentricity of a large fraction of the vertices in the graph
    - "Large fraction" e.g. 90%

Clustering Coefficient

- The **clustering coefficient** of vertex v_i, $C(v_i)$, tells how clique-like the neighbourhood of v_i is
  - Let n_i be the number of neighbours of v_i and m_i the number of edges *between* the neighbours of v_i (v_i excluded)

$$C(v_i) = m_i / \binom{n_i}{2} = \frac{2m_i}{n_i(n_i - 1)}$$

  - Well-defined only for v_i with at least two neighbours
    - For others, let $C(v_i) = 0$
- The clustering coefficient of the graph is the average clustering coefficient of the vertices: $C(G) = n^{-1}\Sigma_i C(v_i)$

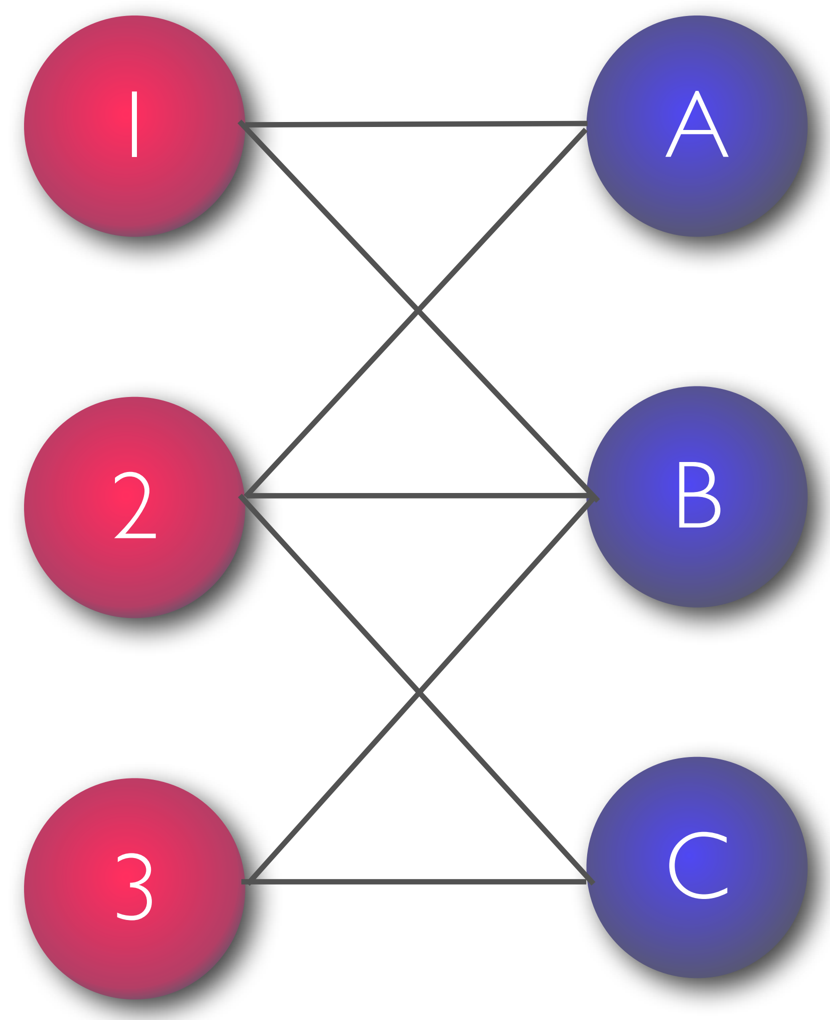
Graph Mining

- Graphs can explain *relations* between objects
- Finding these relations is the task of graph mining
  - The type of the relation depends on the task
- Graph mining is an umbrella term that encompasses many different techniques and problems
  - Frequent subgraph mining
  - Graph clustering
  - Path analysis/building
  - Influence propagation
  - …

Example: Tiling Databases

- Binary matrices define a bipartite graph
- A tile is a biclique of that graph
- Tiling is the task of finding a minimum number of bicliques to cover all edges of a bipartite graph
  - Or to find *k* bicliques to cover most of the edges

	A	B	C
1	1	1	0
2	1	1	1
3	0	1	1



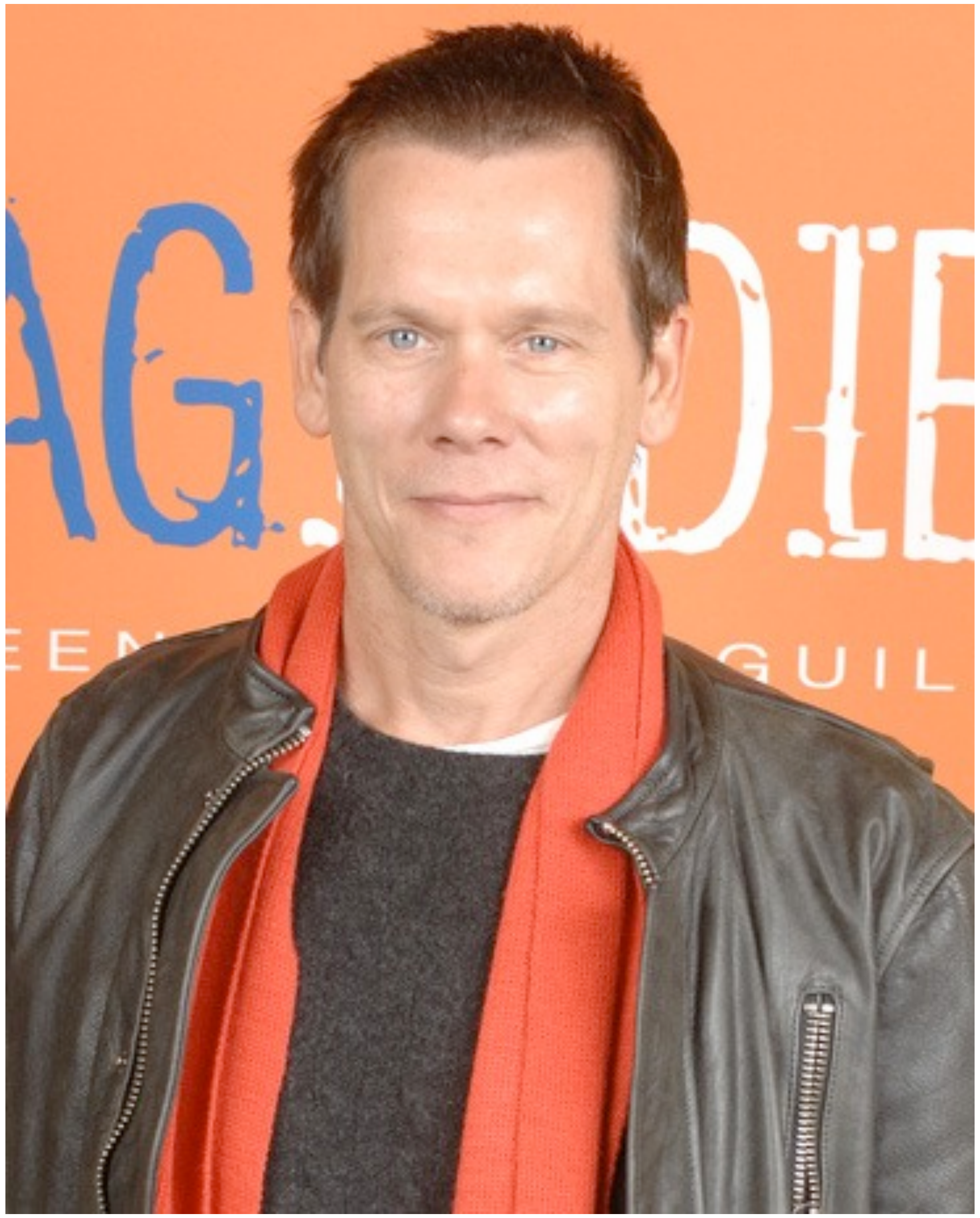
Example: The Characteristics of Erdős Graph

- Co-authorship graph of mathematicians
- 401K authors (vertices), 676K co-authorships (edges)
  - Median degree = 1, mean = 3.36, standard deviation = 6.61
- Large connected component of 268K vertices
  - The radius of the component is 12 and diameter 23
  - Two vertices with eccentricity 12
  - Average distance between two vertices 7.64 (based on a sample)
    - "Eight degrees of separation"
- The clustering coefficient is 0.14

http://www.oakland.edu/enp/

Centrality

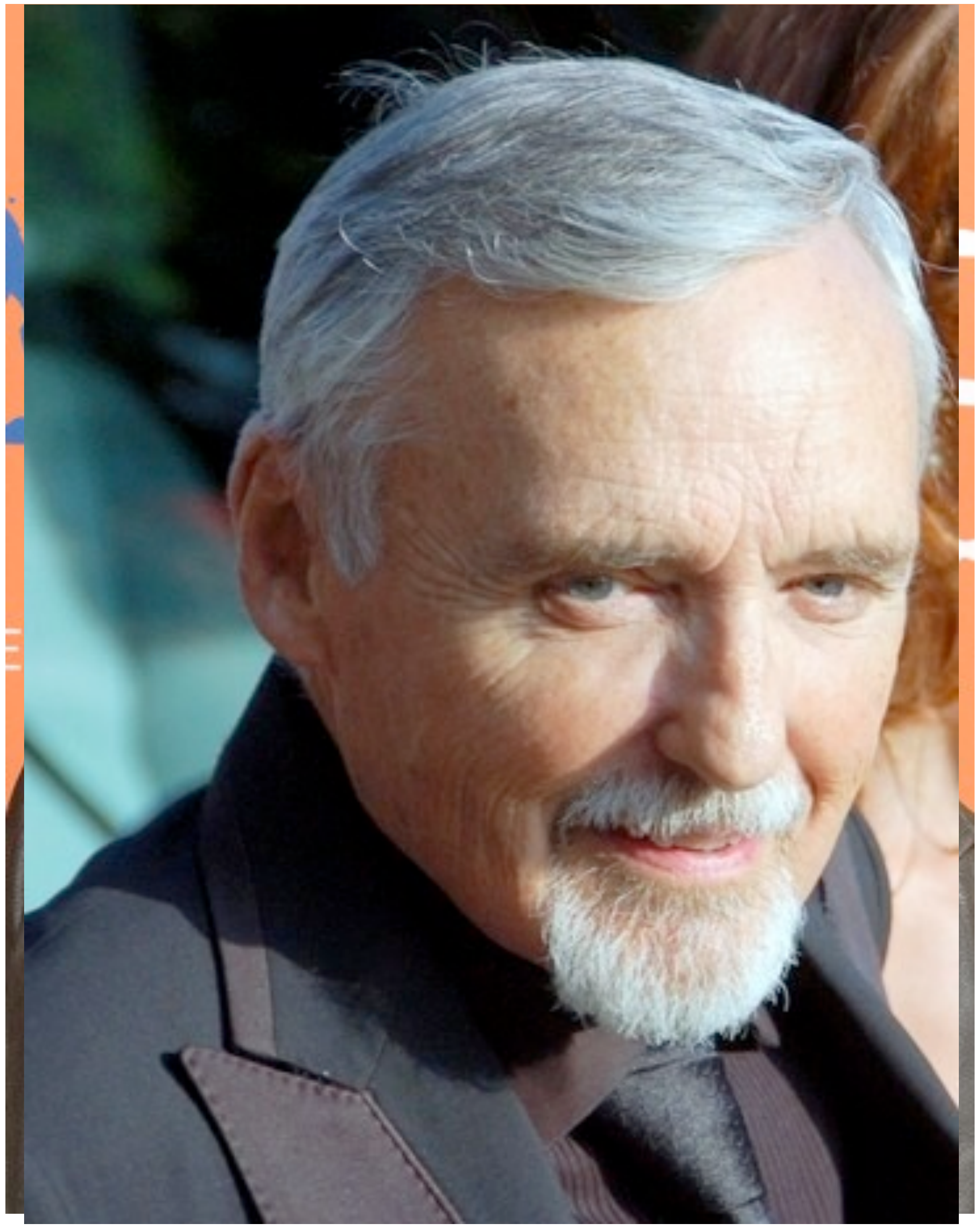
- Six degrees of Kevin Bacon
  - ”Every actor is related to Kevin Bacon by no more than 6 hops”
  - Kevin Bacon has acted with many, that have acted with many others, that have acted with many others…
- That makes Kevin Bacon a *centre* of the co-acting graph
  - Although he’s not the centre: the average distance to him is 2.994 but to Dennis Hopper it is only 2.802



http://oracleofbacon.org

Centrality

- Six degrees of Kevin Bacon
  - ”Every actor is related to Kevin Bacon by no more than 6 hops”
  - Kevin Bacon has acted with many, that have acted with many others, that have acted with many others…
- That makes Kevin Bacon a *centre* of the co-acting graph
  - Although he’s not the centre: the average distance to him is 2.994 but to Dennis Hopper it is only 2.802



http://oracleofbacon.org

Degree and Eccentricity Centrality

- **Centrality** is a function $c: V \to \mathbb{R}$ that induces a total order in V
  - The higher the centrality of a vertex, the more important it is
- In **degree centrality** $c(v_i) = d(v_i)$, the degree of the vertex
- In **eccentricity centrality** the least eccentric vertex is the most central one, $c(v_i) = 1/e(v_i)$
  - The lest eccentric vertex is *central*
  - The most eccentric vertex is *peripheral*

Closeness Centrality

- In **closeness centrality** the vertex with least distance to *all other* vertices is the centre

$$c(v_i) = \left(\sum_j d(v_i, v_j) \right)^{-1}$$

- In eccentricity centrality we aim to minimize the maximum distance
- In closeness centrality we aim to minimize the average distance
  - This is the distance used to measure the centre of Hollywood

Betweenness Centrality

- The **betweenness centrality** measures the number of shortest paths that travel through v_i
  - Measures the “monitoring” role of the vertex
  - “All roads lead to Rome”
- Let η_{jk} be the number of shortest paths between v_j and v_k and let $\eta_{jk}(v_i)$ be the number of those that include v_i
  - Let $\gamma_{jk}(v_i) = \eta_{jk}(v_i)/\eta_{jk}$
  - Betweenness centrality is defined as

$$c(v_i) = \sum_{j \neq i} \sum_{\substack{k \neq i \\ k > j}} \gamma_{jk}$$

Prestige

- In **prestige**, the vertex is more central if it has many incoming edges from other vertices of high prestige
  - $\boldsymbol{A}$ is the adjacency matrix of the directed graph G
  - $\boldsymbol{p}$ is n-dimensional vector giving the prestige of the vertices
  - $\boldsymbol{p} = \boldsymbol{A}^T\boldsymbol{p}$
  - Starting from an initial prestige vector $\boldsymbol{p}_0$, we get $\boldsymbol{p}_k = \boldsymbol{A}^T\boldsymbol{p}_{k-1} = \boldsymbol{A}^T(\boldsymbol{A}^T\boldsymbol{p}_{k-2}) = (\boldsymbol{A}^T)^2\boldsymbol{p}_{k-2} = (\boldsymbol{A}^T)^3\boldsymbol{p}_{k-3} = \ldots = (\boldsymbol{A}^T)^k\boldsymbol{p}_0$
- Vector $\boldsymbol{p}$ converges to the dominant eigenvector of $\boldsymbol{A}^T$
  - Under some assumptions

PageRank

- PageRank uses normalized prestige to rank web pages
- If there is a vertex with no out-going edges, the prestige cannot be computed
  - PageRank evades this problem by adding a small probability of a random jump to another vertex
  - Random Surfer model
- Computing the PageRank is equivalent to computing the stationary distribution of a certain Markov chain
  - Which is again equivalent to computing the dominant eigenvector

Graph Properties

- Several real-world graphs exhibit certain characteristics
  - Studying what these are and explaining why they appear is an important area of network research
- As data miners, we need to understand the consequences of these characteristics
  - Finding a result that can be explained merely by one of these characteristics is not interesting
- We also want to *model* graphs with these characteristics

Small-World Property

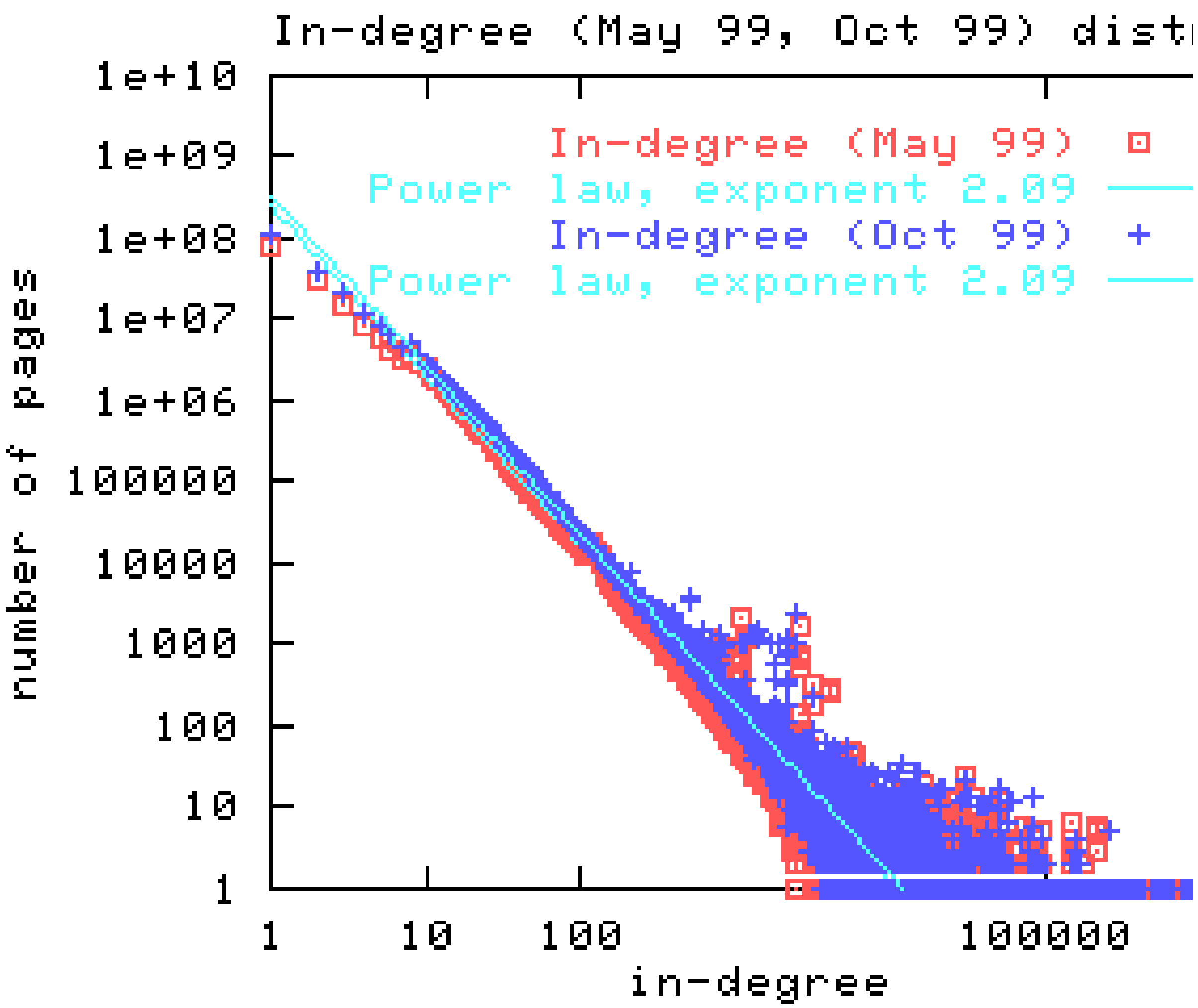
- A graph G is said to exhibit a **small-world property** if its average path length scales logarithmically, $\mu_L \propto \log n$
  - The six degrees of Kevin Bacon is based on this property
  - Also the Erdős number
    - How far a mathematician is from Hungarian combinatorist Paul Erdős
    - A radius of a large, connected mathematical co-authorship network (268K authors) is 12 and diameter 23

Scale-Free Property

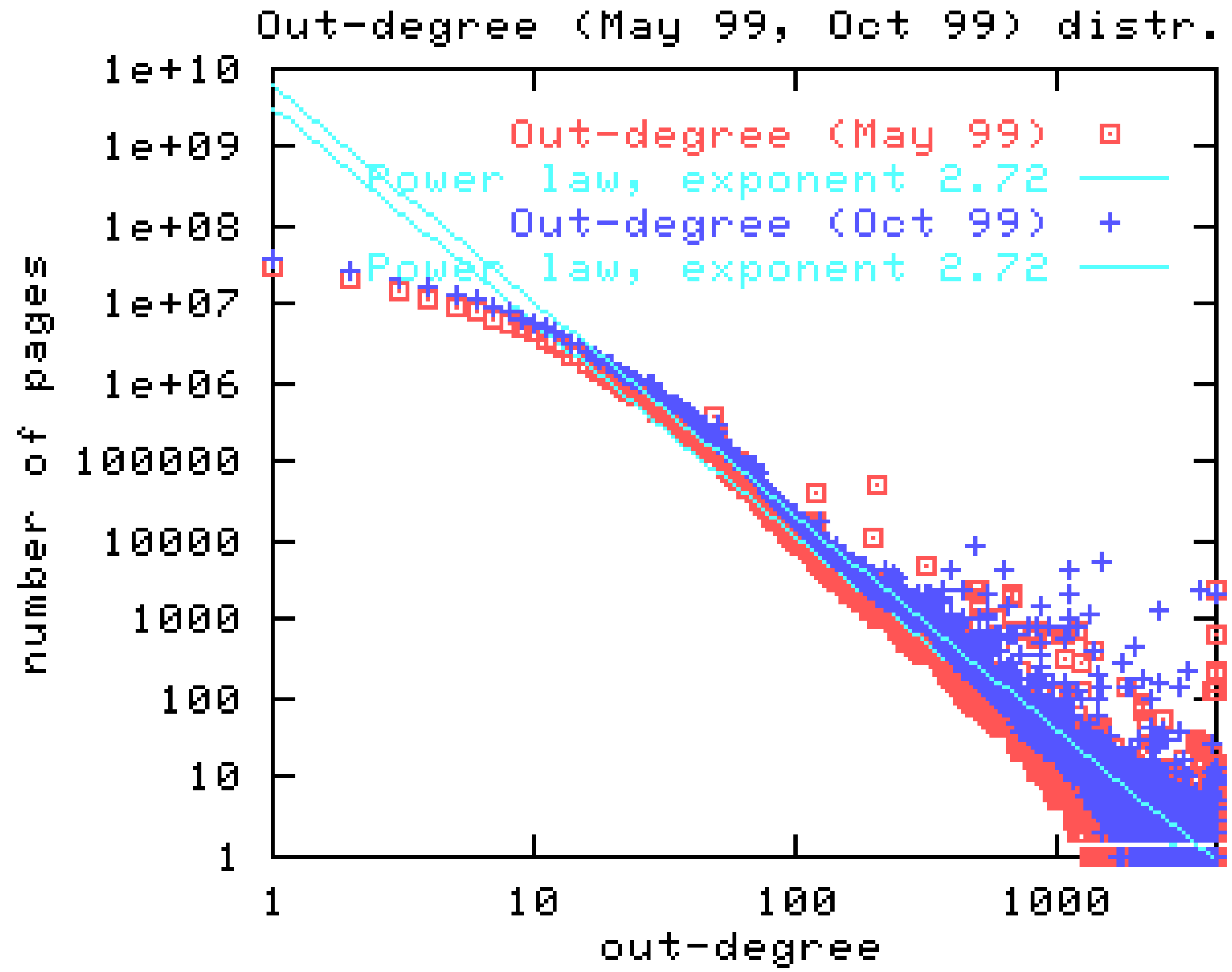
- The **degree distribution** of a graph is the distribution of its vertex degrees
  - How many vertices with degree 1, how many with degree 2, etc.
  - $f(k)$ is the number of edges with degree k
- A graph is said to exhibit **scale-free property** if $f(k) \propto k^{-\gamma}$
  - So-called power-law distribution
  - Majority of vertices have small degrees, few have very high degrees
  - Scale-free: $f(ck) = \alpha(ck)^{-\gamma} = (\alpha c^{-\gamma})k^{-\gamma} \propto k^{-\gamma}$

Example: WWW Links

In-degree



Out-degree



Broder et al. *Graph structure in the web*. WWW'00

$s = 2.09$

$s = 2.72$

Clustering Effect

- A graph exhibits **clustering effect** if the distribution of average clustering coefficient (per degree) follow the power law
  - If $C(k)$ is the average clustering coefficient of all vertices of degree k, then $C(k) \propto k^{-\gamma}$
- The vertices with small degrees are part of highly clustered areas (high clustering coefficient) while "hub vertices" have smaller clustering coefficients

Random Graph Models

- Begin able to generate random graphs that exhibit these properties is very useful
  - They tell us something how such graphs have come to be
  - They let us study what we find in an "average" graph
  - With some graph models, we can also make analytical studies of the properties
    - What to expect

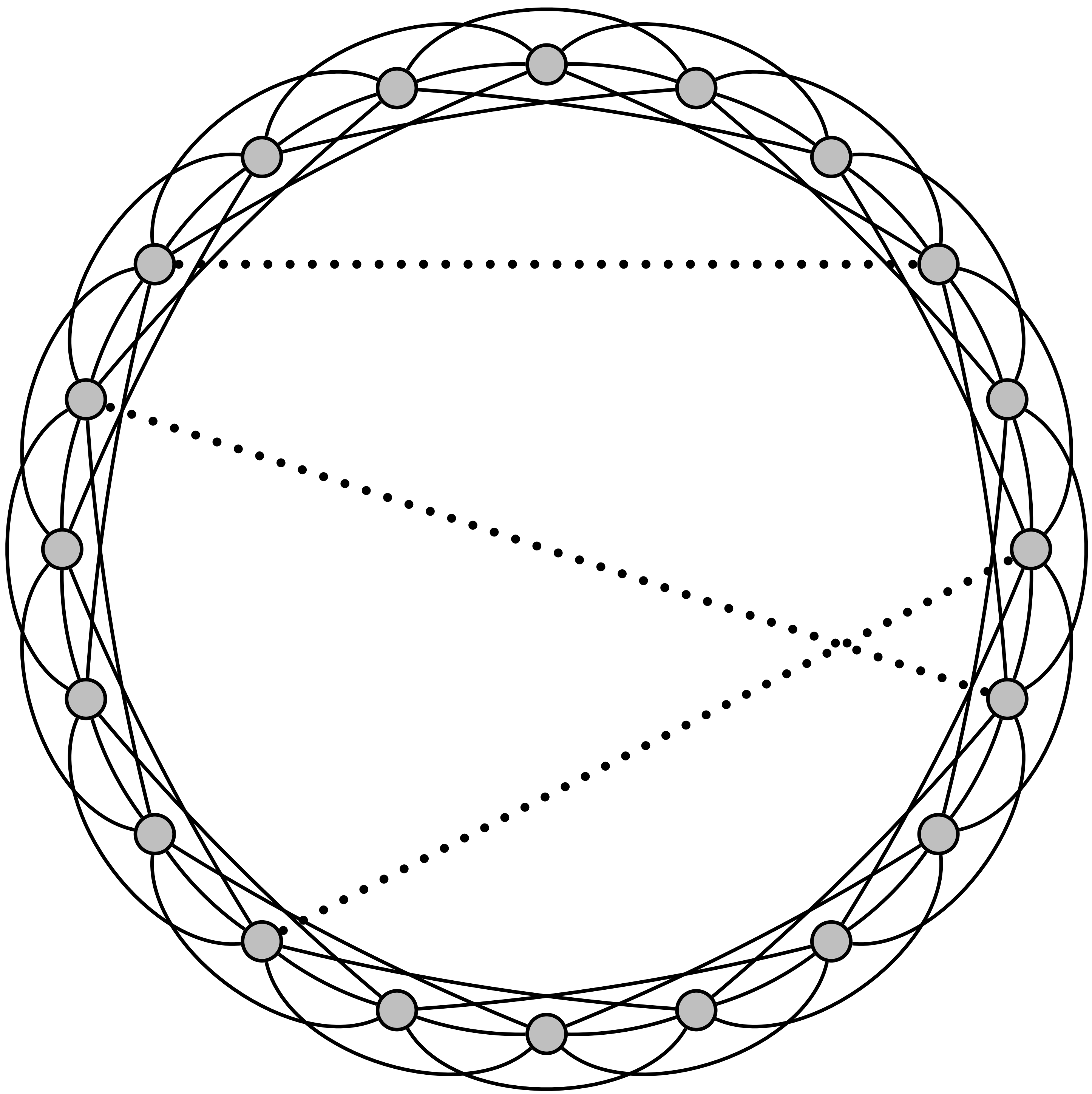
Erdős–Rényi Graphs

- Two parameters: number of vertices n and number of edges m
- Samples uniformly from all such graphs
  - Sample m edges u.a.r. without replacement
- Average degree is $2m/n$
- Degree distribution follows Poisson, not power law
- Clustering coefficient is uniform
- Exhibits small-world property

Watts–Strogatz Graphs

- Aims for high local clustering
- Starts with vertices in a ring, each connected to k neighbours left and right
- Adds random perturbations
  - Edge rewiring: move the end-point of random edges to random vertices
  - Edge shortcuts: add random edges between vertices
- Not scale-free
- High clustering coefficient for small amounts of perturbations
- Small diameter with some amount of perturbations

Example



Barabási–Albert Graphs

- Mimics dynamic evolution of graphs
  - Preferential attachment
- Starts with a regular graph
- At each time step, adds a new vertex u
  - From u, adds q edges to other vertices
  - Vertices are sampled proportional to their degree
    - High degree, high probability to get more edges
- Degree distribution follows power law (with $\gamma = 3$)
- Ultra-small world behaviour
- Very small clustering coefficient