



EECS6414:

Data Analytics & Visualization

Network Properties: Characterizing/ Measuring Networks

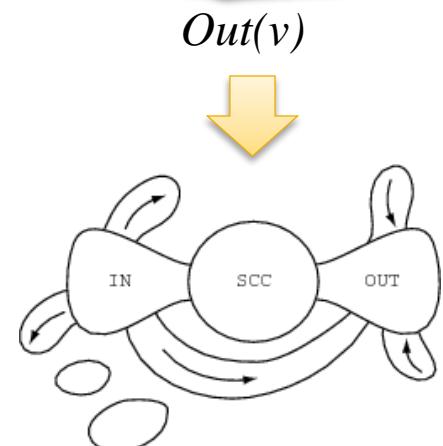
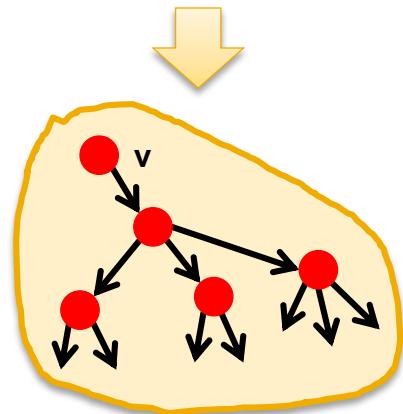
Agenda

- Characterizing/Measuring Networks
 - Network Properties
- Case Study: A Real World Network (MSN)

Structure of Networks

- For example, last time we talked about Observations and Models for the Web graph:

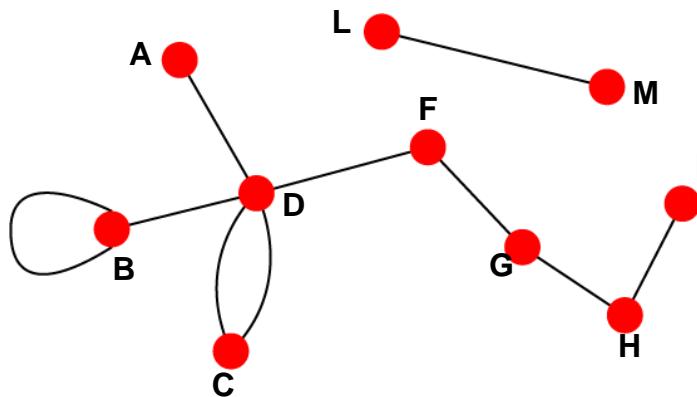
- 1) We took a real system: **the Web**
- 2) We represented it as a **directed graph**
- 3) We used the language of graph theory
 - Strongly Connected Components**
- 4) We designed a **computational experiment**:
 - Find In- and Out-components of a given node v
- 5) **We learned something about the structure of the Web: BOWTIE!**



Undirected vs. Directed Networks

Undirected graphs

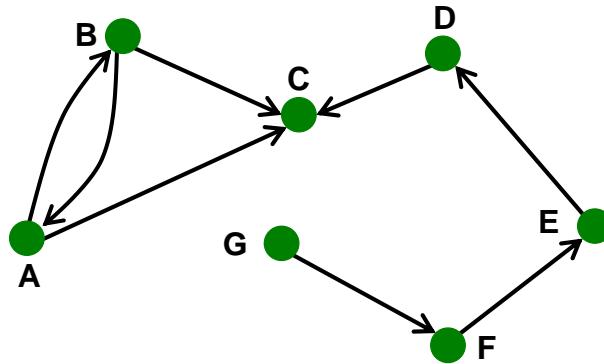
- Links: undirected (symmetrical, reciprocal relations)



- Undirected links:
 - Collaborations
 - Friendship on Facebook

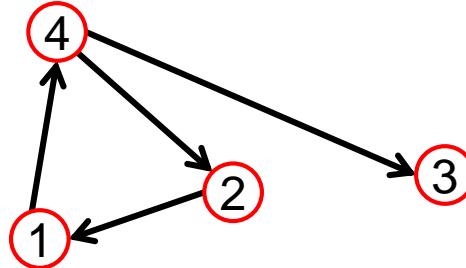
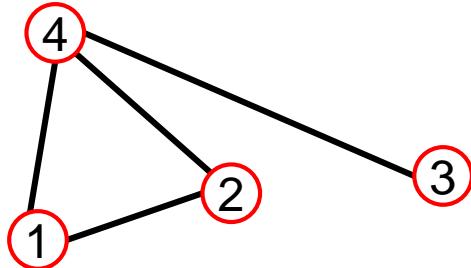
Directed graphs

- Links: directed (asymmetrical relations)



- Directed links:
 - Phone calls
 - Following on Twitter

Adjacency Matrix



$A_{ij} = 1$ if there is a link from node i to node j

$A_{ij} = 0$ otherwise

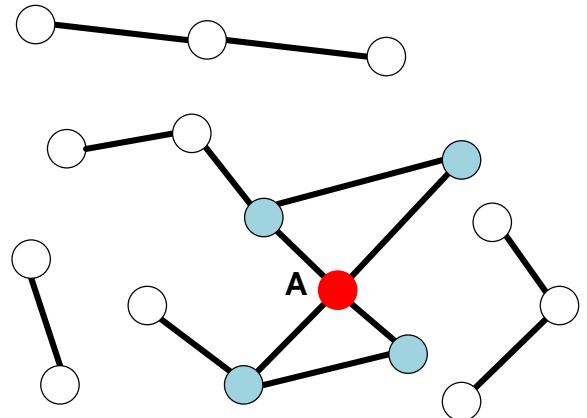
$$A = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

$$A = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

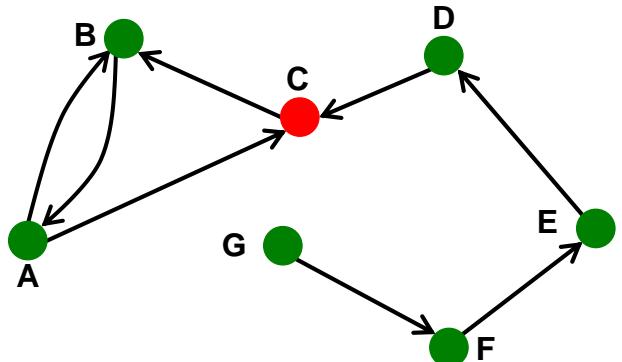
Note that for a directed graph (right) the matrix is not symmetric.

Node Degrees

Undirected



Directed



Source: Node with $k^{in} = 0$

Sink: Node with $k^{out} = 0$

Node degree, k_i : the number of edges adjacent to node i

$$k_A = 4$$

Avg. degree: $\bar{k} = \langle k \rangle = \frac{1}{N} \sum_{i=1}^N k_i = \frac{2E}{N}$

In directed networks we define an **in-degree** and **out-degree**. The (total) degree of a node is the sum of in- and out-degrees.

$$k_C^{in} = 2 \quad k_C^{out} = 1 \quad k_C = 3$$

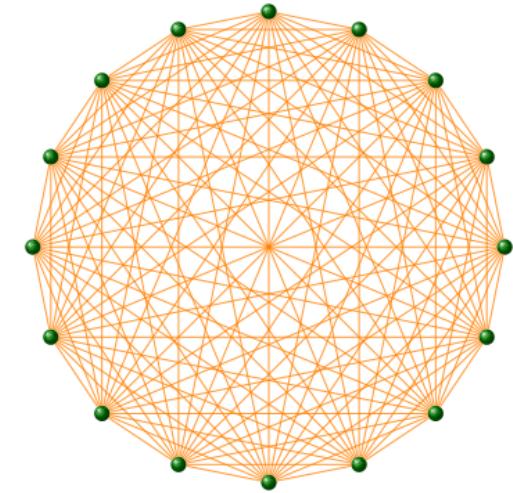
$$\bar{k} = \frac{E}{N}$$

$$\bar{k}^{in} = \bar{k}^{out}$$

Complete Graph

The **maximum number of edges** in an undirected graph on N nodes is

$$E_{\max} = \binom{N}{2} = \frac{N(N-1)}{2}$$



An undirected graph with the number of edges $E = E_{\max}$ is called a **complete graph**, and its average degree is $N-1$

Networks are Sparse Graphs

Most real-world networks are **sparse**

$$E \ll E_{\max} \text{ (or } \bar{k} \ll N-1)$$

WWW (Stanford-Berkeley):	$N=319,717$	$\langle k \rangle=9.65$
Social networks (LinkedIn):	$N=6,946,668$	$\langle k \rangle=8.87$
Communication (MSN IM):	$N=242,720,596$	$\langle k \rangle=11.1$
Coauthorships (DBLP):	$N=317,080$	$\langle k \rangle=6.62$
Internet (AS-Skitter):	$N=1,719,037$	$\langle k \rangle=14.91$
Roads (California):	$N=1,957,027$	$\langle k \rangle=2.82$
Proteins (S. Cerevisiae):	$N=1,870$	$\langle k \rangle=2.39$

(Source: Leskovec et al., *Internet Mathematics*, 2009)

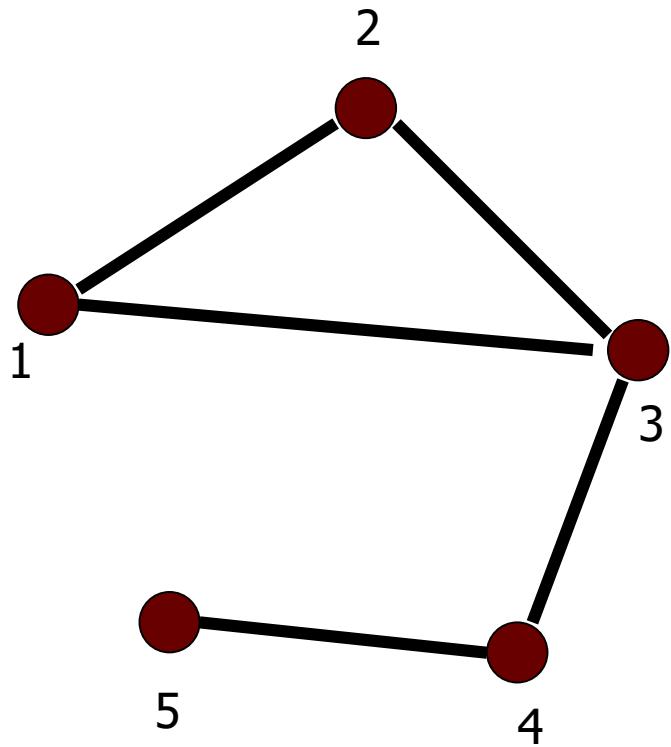
Consequence: Adjacency matrix is filled with zeros!

(Density of the matrix (E/N^2): WWW= 1.51×10^{-5} , MSN IM = 2.27×10^{-8})

Graph Representation

- Adjacency Matrix
 - symmetric matrix for undirected graphs

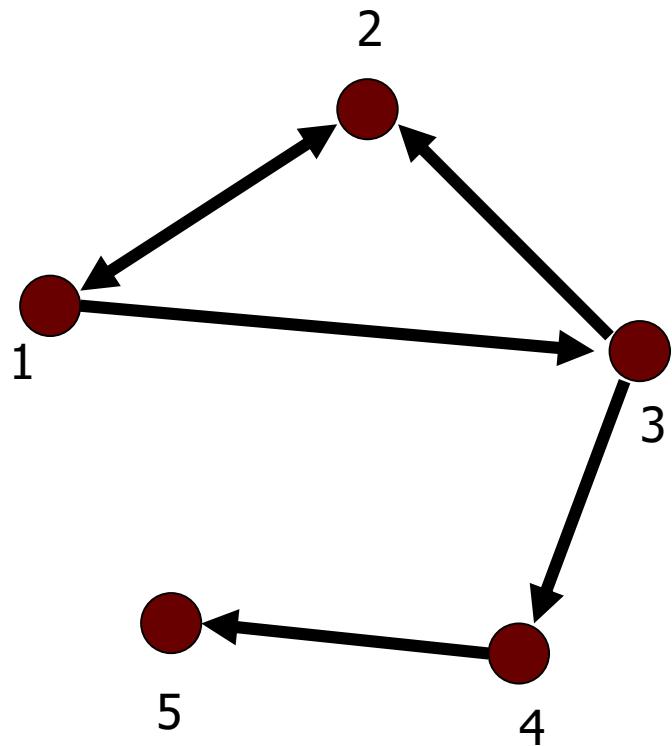
$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$



Graph Representation

- Adjacency Matrix
 - unsymmetric matrix for undirected graphs

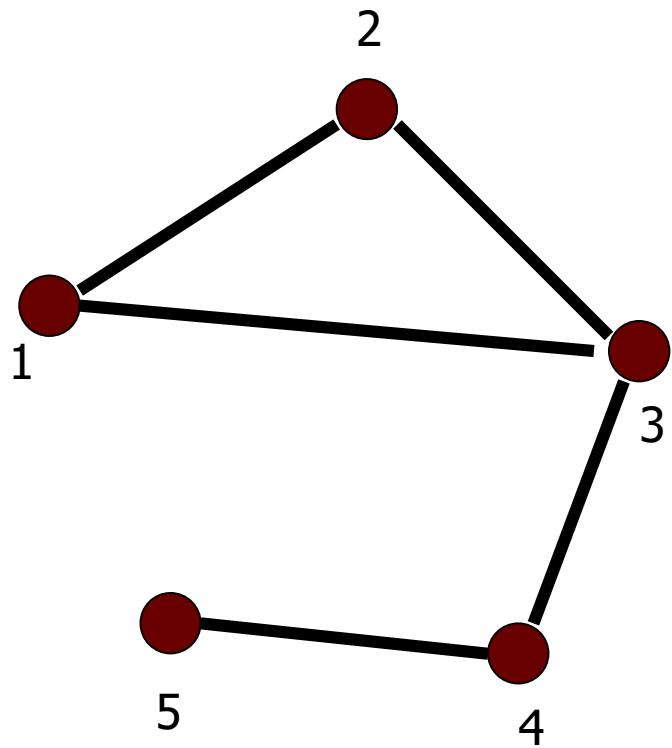
$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$



Graph Representation

- Adjacency List
 - For each node keep a list with neighboring nodes

1: [2, 3]
2: [1, 3]
3: [1, 2, 4]
4: [3, 5]
5: [4]



Graph Representation

- Adjacency List
 - For each node keep a list of the nodes it points to

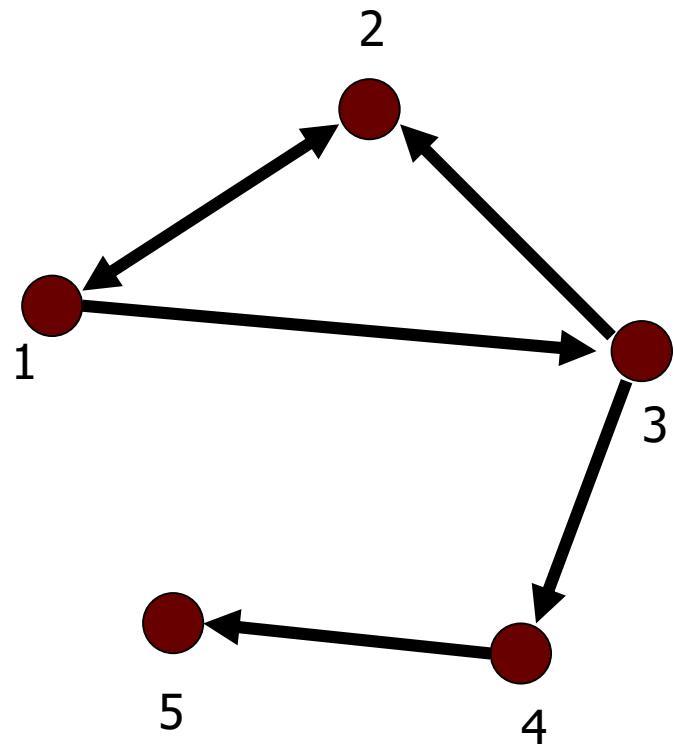
1: [2, 3]

2: [1]

3: [2, 4]

4: [5]

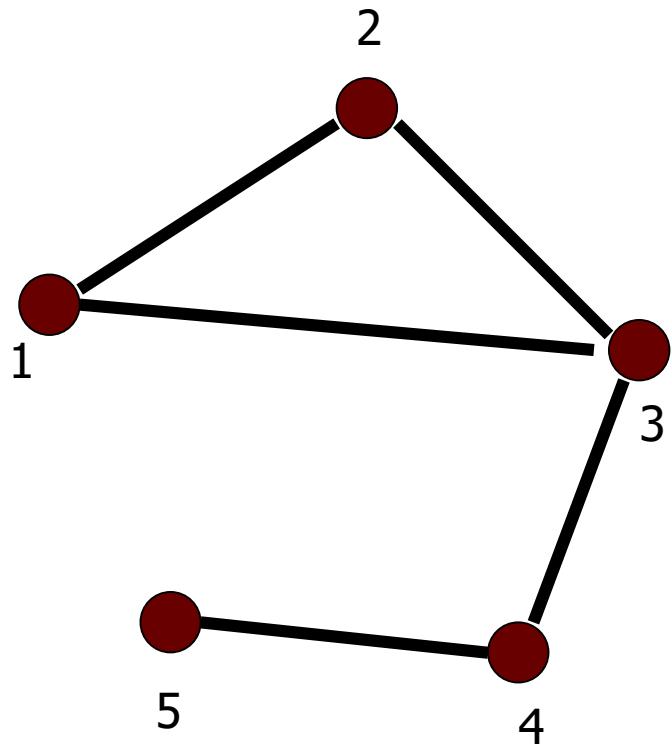
5: [null]



Graph Representation

- List of edges
 - Keep a list of all the edges in the graph

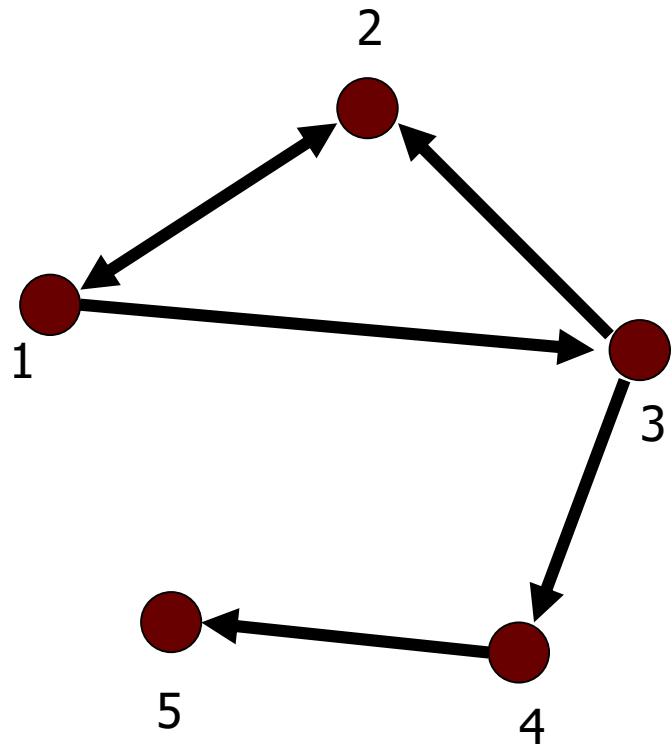
(1,2)
(2,3)
(1,3)
(3,4)
(4,5)



Graph Representation

- List of edges
 - Keep a list of all the directed edges in the graph

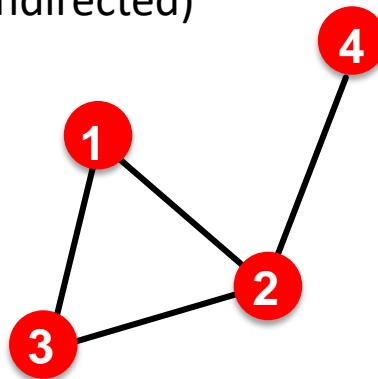
(1,2)
(2,1)
(1,3)
(3,2)
(3,4)
(4,5)



More Types of Graphs:

■ Unweighted

(undirected)



$$A_{ij} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

$$A_{ii} = 0$$

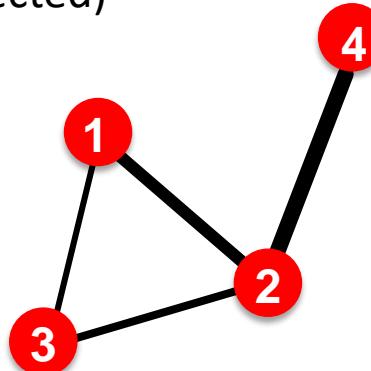
$$A_{ij} = A_{ji}$$

$$E = \frac{1}{2} \sum_{i,j=1}^N A_{ij} \quad \bar{k} = \frac{2E}{N}$$

Examples: Friendship, Hyperlink

■ Weighted

(undirected)



$$A_{ij} = \begin{pmatrix} 0 & 2 & 0.5 & 0 \\ 2 & 0 & 1 & 4 \\ 0.5 & 1 & 0 & 0 \\ 0 & 4 & 0 & 0 \end{pmatrix}$$

$$A_{ii} = 0$$

$$A_{ij} = A_{ji}$$

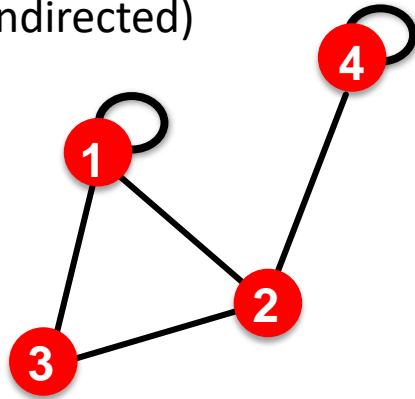
$$E = \frac{1}{2} \sum_{i,j=1}^N \text{nonzero}(A_{ij}) \quad \bar{k} = \frac{2E}{N}$$

Examples: Collaboration, Internet, Roads

More Types of Graphs:

■ Self-edges (self-loops)

(undirected)



$$A_{ij} = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}$$

$$A_{ii} \neq 0$$

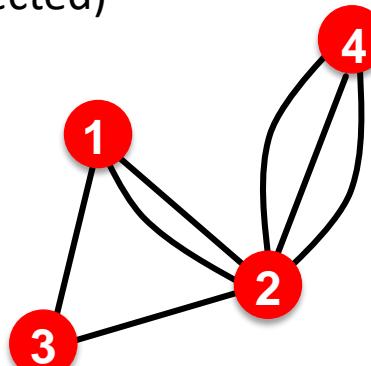
$$A_{ij} = A_{ji}$$

$$E = \frac{1}{2} \sum_{i,j=1, i \neq j}^N A_{ij} + \sum_{i=1}^N A_{ii}$$

Examples: Proteins, Hyperlinks

■ Multigraph

(undirected)



$$A_{ij} = \begin{pmatrix} 0 & 2 & 1 & 0 \\ 2 & 0 & 1 & 3 \\ 1 & 1 & 0 & 0 \\ 0 & 3 & 0 & 0 \end{pmatrix}$$

$$A_{ii} = 0$$

$$A_{ij} = A_{ji}$$

$$E = \frac{1}{2} \sum_{i,j=1}^N \text{nonzero}(A_{ij})$$

$$\bar{k} = \frac{2E}{N}$$

Examples: Communication, Collaboration

Network Representations

WWW >> directed multigraph with self-edges

Facebook friendships >> undirected, unweighted

Citation networks >> unweighted, directed, acyclic

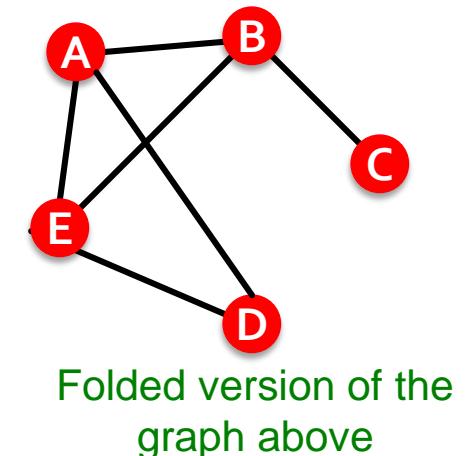
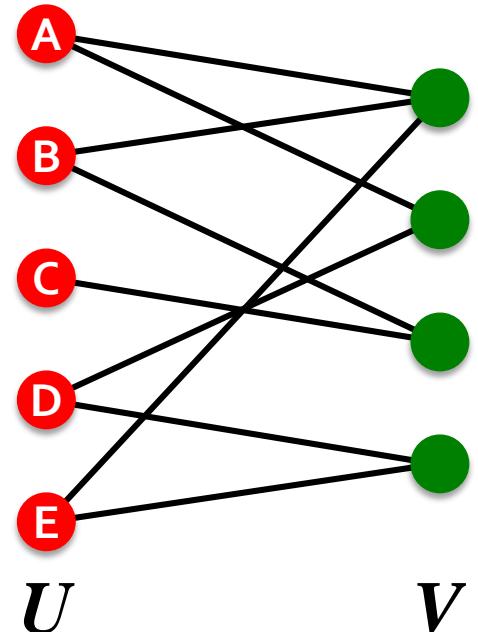
Collaboration networks >> undirected multigraph or weighted graph

Mobile phone calls >> directed, (weighted?) multigraph

Protein Interactions >> undirected, unweighted with self-interactions

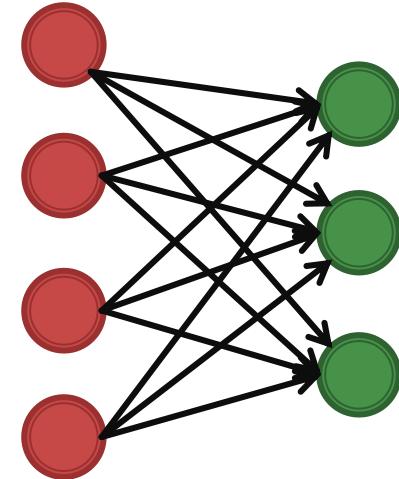
Bipartite Graph

- **Bipartite graph** is a graph whose nodes can be divided into two disjoint sets U and V such that every link connects a node in U to one in V ; that is, U and V are **independent sets**
- **Examples:**
 - Authors-to-papers (they authored)
 - Actors-to-Movies (they appeared in)
 - Users-to-Movies (they rated)
- **“Folded” networks:**
 - Author collaboration networks
 - Movie co-rating networks



Web Cores

- **Cores**: Small complete bipartite graphs (of size 3×3 , 4×3 , 4×4)
 - Similar to the triangles in undirected graphs
- Found more frequently than expected on the Web graph
- Correspond to communities of enthusiasts (e.g., fans of Japanese rock bands)

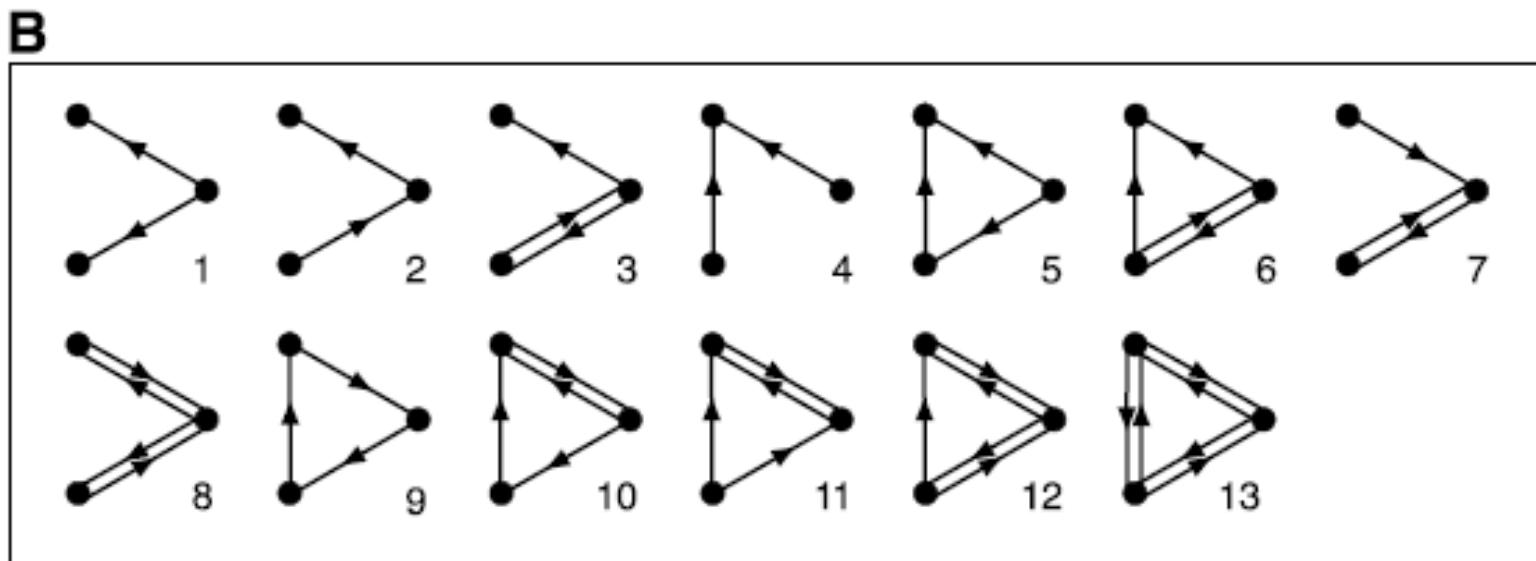


Motifs

- Most networks have the same characteristics with respect to **global measurements**
 - can we say something about the **local structure** of the networks?
- **Motifs:** Find small subgraphs that are **over-represented** in the network

Example

- Motifs of size 3 in a directed graph

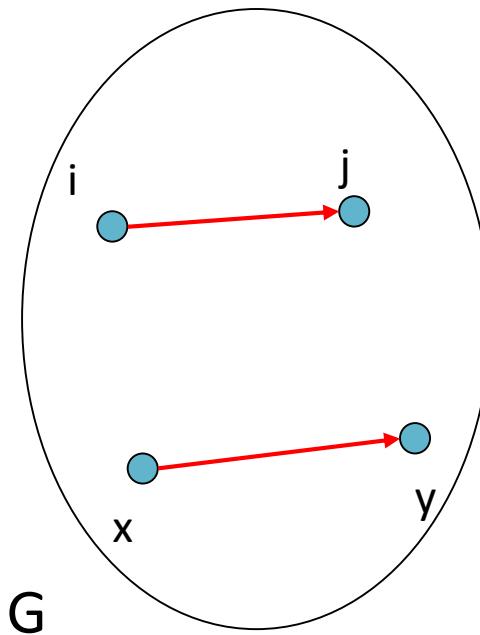


Finding Interesting Motifs

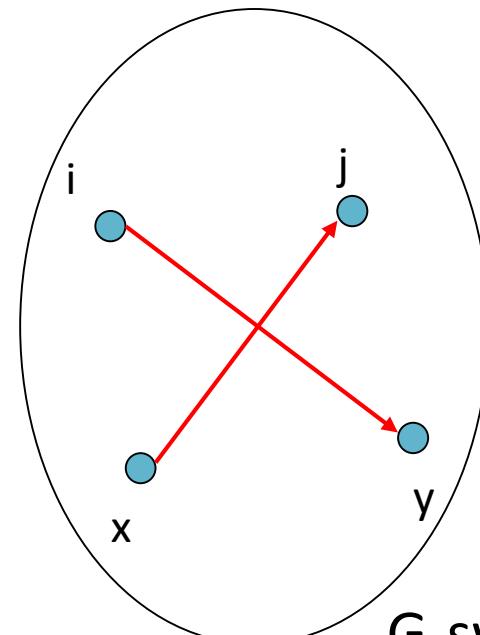
- Sample a part of the graph of size **S**
- Count the **frequency of the motifs** of interest
- Compare against the **frequency of the motif in a random graph** with the same number of nodes **and** the same degree distribution

Generating a Random Graph

- Find edges (i,j) and (x,y) such that edges (i,y) and (x,j) do not exist, and swap them
 - repeat for a large enough number of times



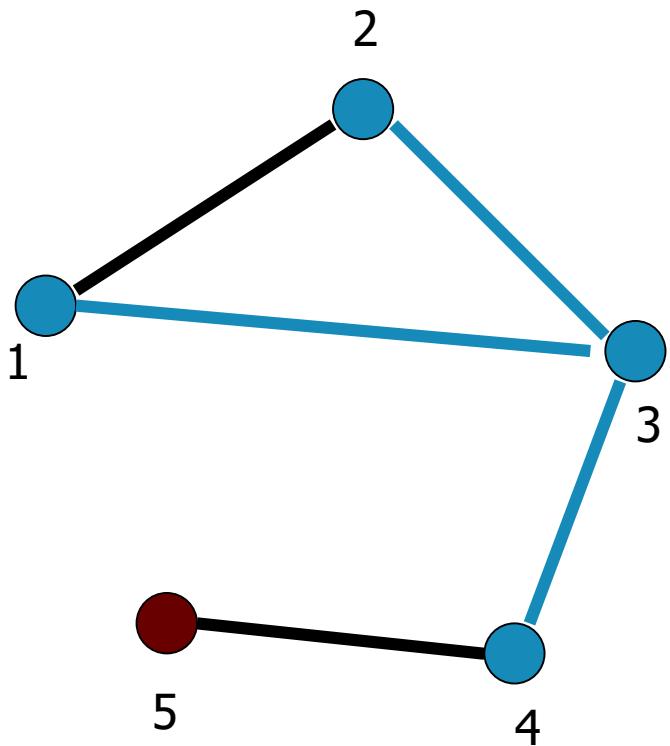
degrees of i,j,x,y
are preserved



G-swapped

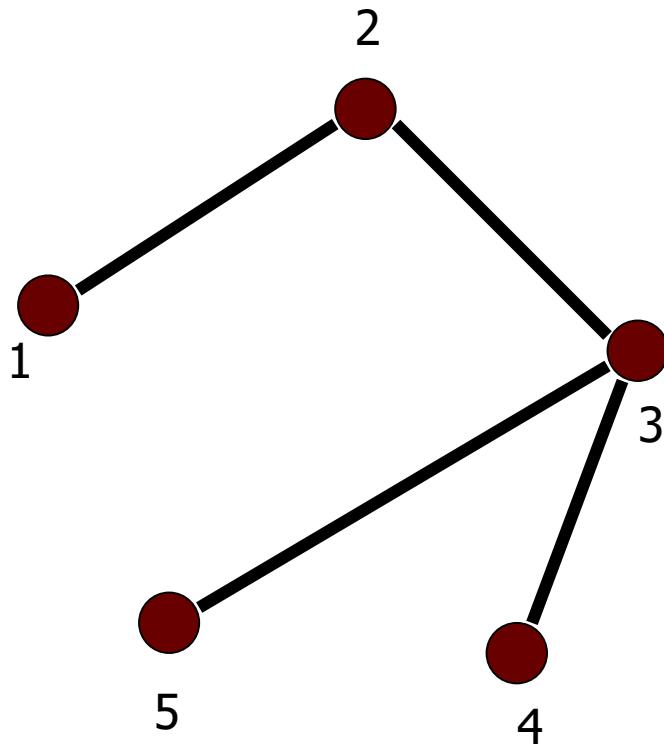
Subgraphs

- **Subgraph:** Given $V' \subseteq V$, and $E' \subseteq E$, the graph $G'=(V',E')$ is a subgraph of G .
- **Induced subgraph:** Given $V' \subseteq V$, let $E' \subseteq E$ is the set of all edges between the nodes in V' . The graph $G'=(V',E')$, is an induced subgraph of G



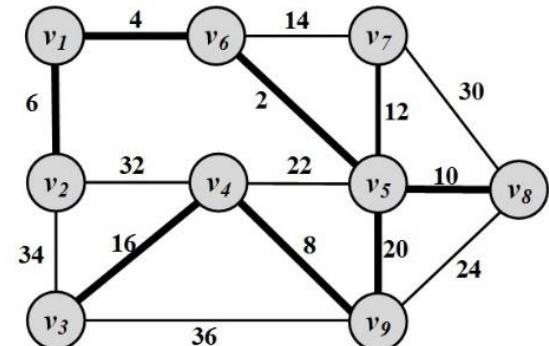
Trees

- Connected Undirected graphs without cycles

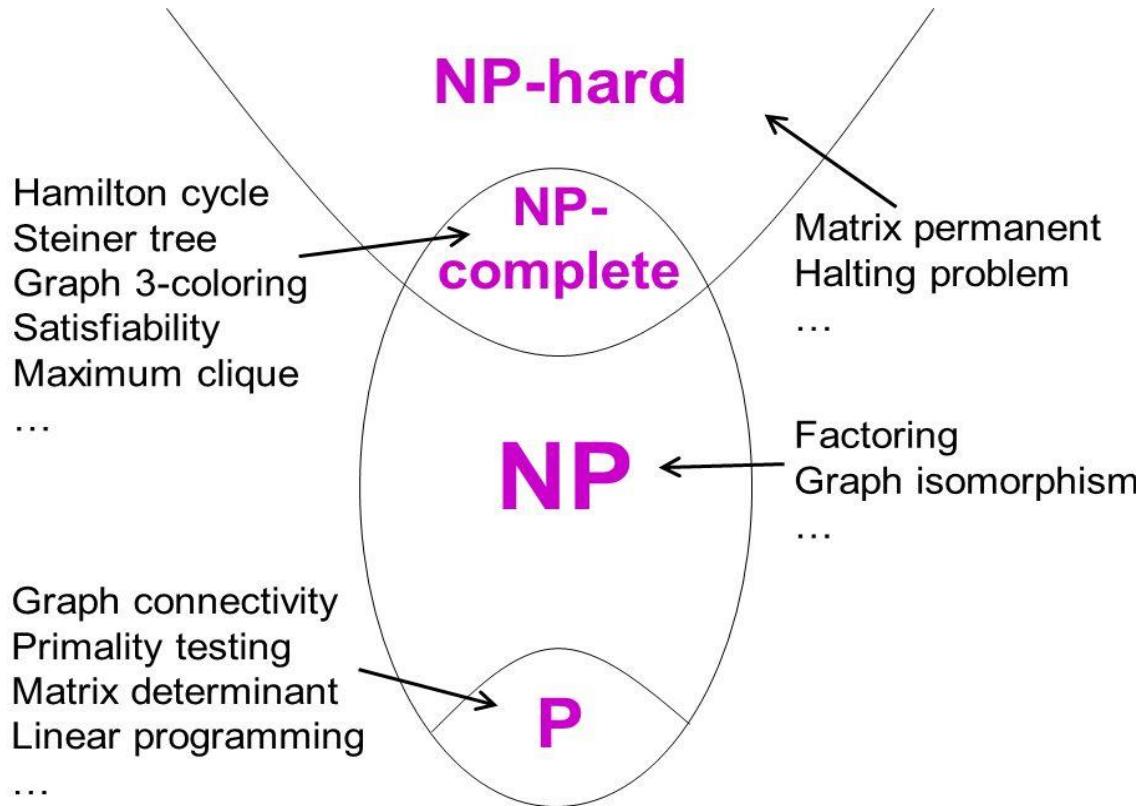


Spanning Tree

- For any connected graph, the **spanning tree** is a subgraph and a tree that includes all the nodes of the graph
- There may exist **multiple spanning trees** for a graph
- The **weigh of a spanning tree** (among multiple spanning trees) of a graph is the summation of the edge weights in that spanning tree
- Minimum Spanning Tree (MST):** The spanning tree with the minimum weight



Classes of Complexity



P: Solvable in polynomial time

NP: Verified in polynomial time, but no known solution in polynomial time

NP-hard: At least as difficult as the hardest NP problems

NP-complete: The hardest of NP problems

More Network Properties...

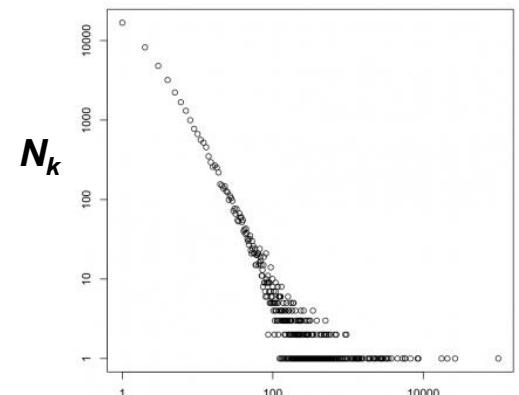
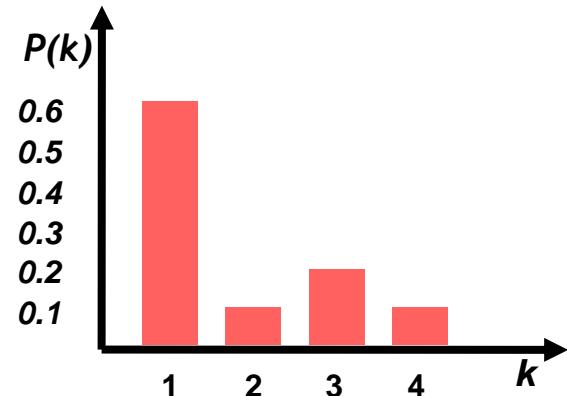
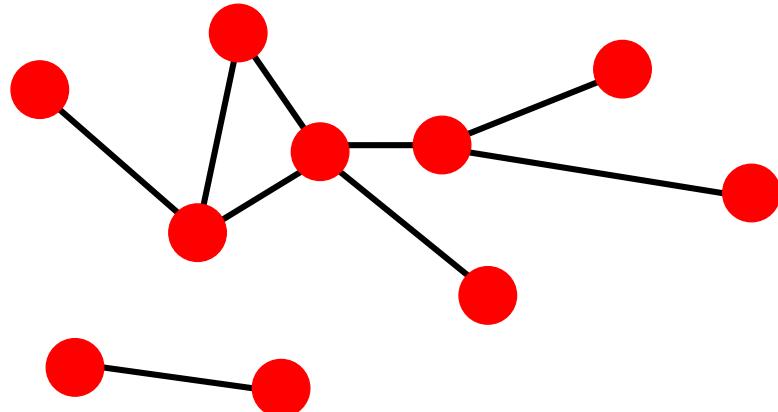
Degree Distribution

- **Degree distribution $P(k)$:** Probability that a randomly chosen node has degree k

$$N_k = \# \text{ nodes with degree } k$$

- Normalized histogram:

$$P(k) = N_k / N \quad \rightarrow \quad \text{plot}$$



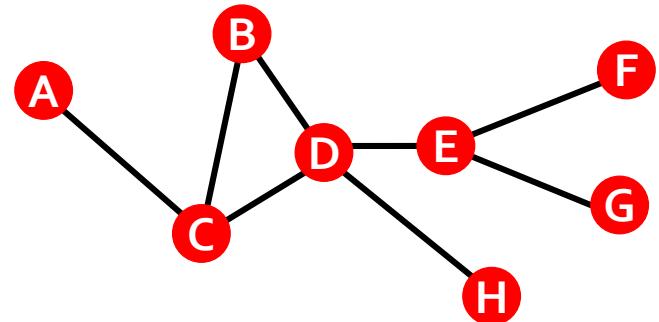
Paths in a Graph

- A **path** is a sequence of nodes in which each node is linked to the next one

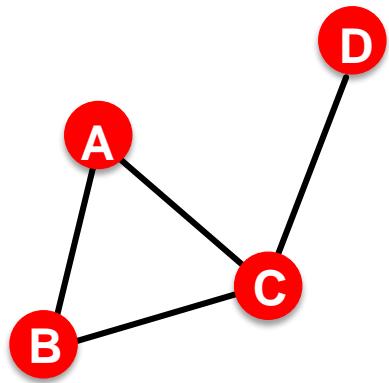
$$P_n = \{i_0, i_1, i_2, \dots, i_n\} \quad P_n = \{(i_0, i_1), (i_1, i_2), (i_2, i_3), \dots, (i_{n-1}, i_n)\}$$

- Path can intersect itself and pass through the same edge multiple times

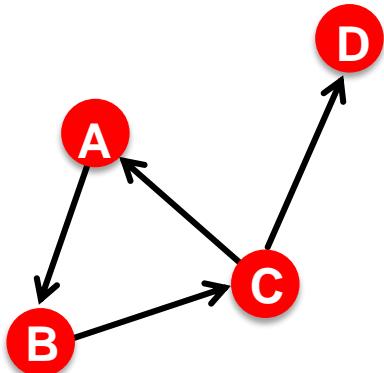
- E.g.: ACBDCDEG
- In a directed graph a path can only follow the direction of the “arrow”



Distance in a Graph



$$h_{B,D} = 2$$



$$h_{B,C} = 1, h_{C,B} = 2$$

- **Distance (shortest path, geodesic)** between a pair of nodes is defined as the number of edges along the shortest path connecting the nodes

- *If the two nodes are disconnected, the distance is usually defined as infinite

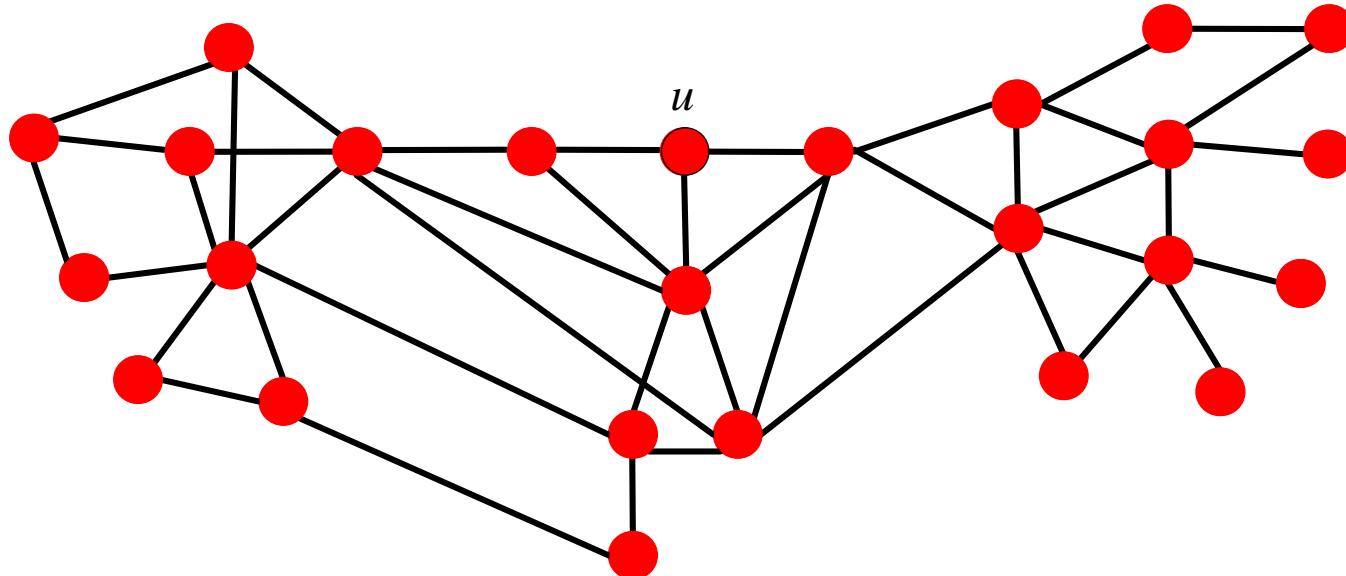
- In **directed graphs** paths need to follow the direction of the arrows

- Consequence: Distance is **not symmetric**: $h_{A,C} \neq h_{C,A}$

Finding Shortest Paths

■ Breadth First Search:

- Start with node u , mark it to be at distance $h_u(u)=0$, add u to the queue
- While the queue not empty:
 - Take node v off the queue, put its unmarked neighbors w into the queue and mark $h_u(w)=h_u(v)+1$



Shortest Paths on Weighted Graphs

- Shortest paths on **weighted** graphs are harder to construct
 - There are several well known algorithms for finding **single-source**, or **all-pairs** shortest paths
- **Single-source Shortest Path (SSSP)**
 - Dijkstra's algorithm (non-negative weights)
 - Bellman-Ford algorithm (allows negative weights)
- **All-pairs Shortest Paths (APSP)**
 - Floyd-Warshall algorithm (allows negative weights)
 - Johnson's algorithm (allows negative weights)

Network Diameter

- **Diameter:** the maximum (shortest path) distance between any pair of nodes in a graph
- **Average path length** for a connected graph (component) or a strongly connected (component of a) directed graph

$$\bar{h} = \frac{1}{2E_{\max}} \sum_{i, j \neq i} h_{ij}$$

where h_{ij} is the distance from node i to node j

- Many times we compute the average only over the connected pairs of nodes (that is, we ignore “infinite” length paths)

Clustering Coefficient

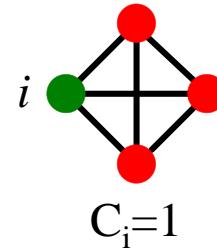
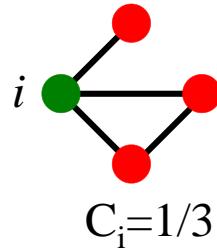
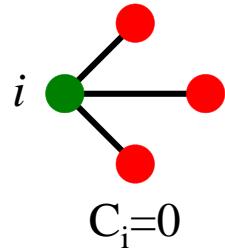
■ Clustering coefficient:

- What portion of i 's neighbors are connected?

- Node i with degree k_i

- $C_i \in [0,1]$

- $C_i = \frac{2e_i}{k_i(k_i - 1)}$ where e_i is the number of edges between the neighbors of node i



■ Average clustering coefficient:

$$C = \frac{1}{N} \sum_i^N C_i$$

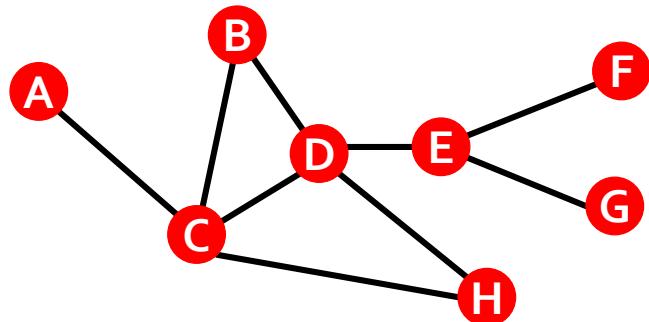
Clustering Coefficient: Example

■ Clustering coefficient:

- What portion of i 's neighbors are connected?
- Node i with degree k_i

- $C_i = \frac{2e_i}{k_i(k_i - 1)}$

where e_i is the number of edges between the neighbors of node i



$$k_B=2, \ e_B=1, \ C_B=2/2 = 1$$

$$k_D=4, \ e_D=2, \ C_D=4/12 = 1/3$$

...

Key Network Properties

Degree distribution: $P(k)$

Path length: h

Clustering coefficient: C

**Let's measure $P(k)$, h and C on
a real-world network!**

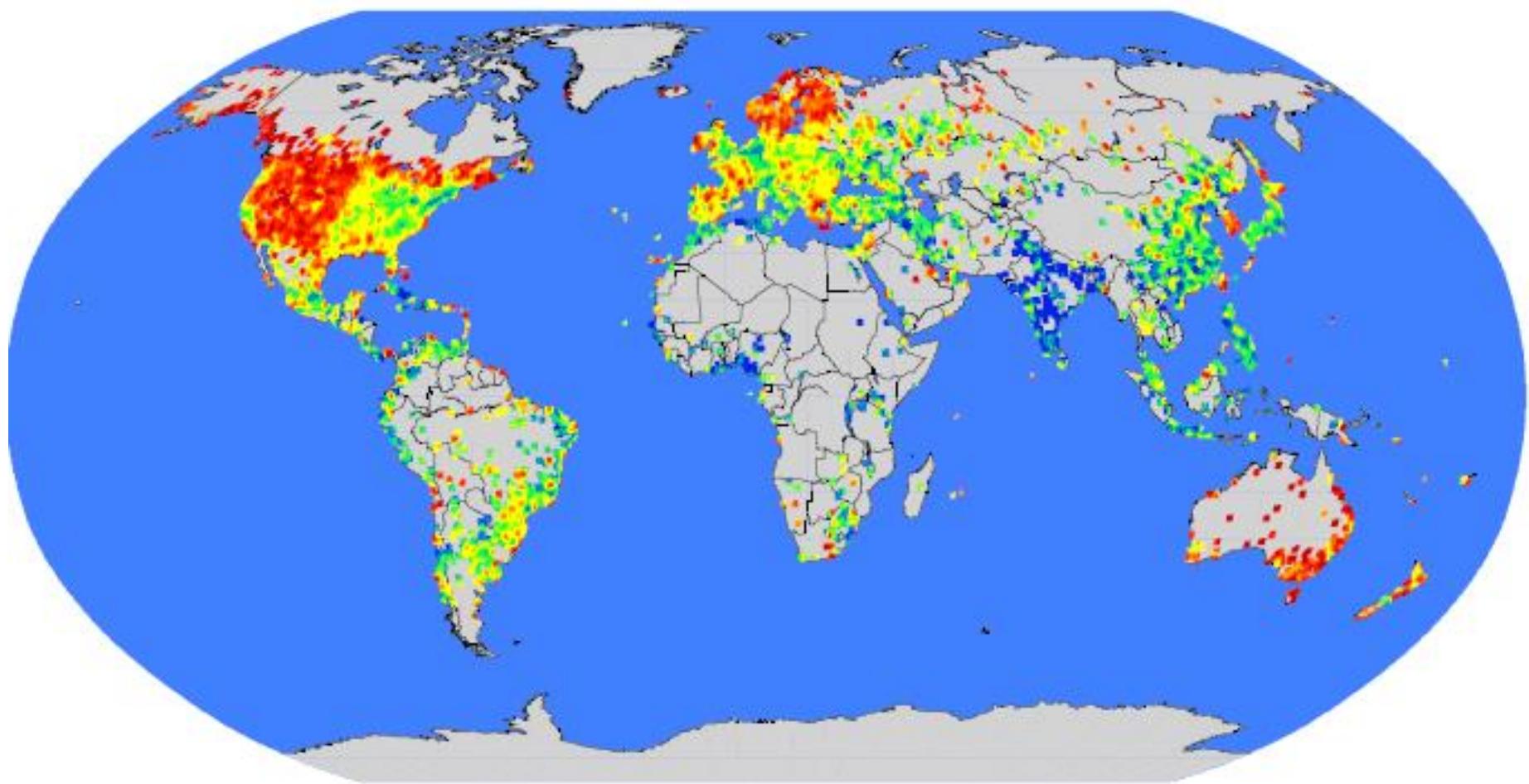
The MSN Messenger



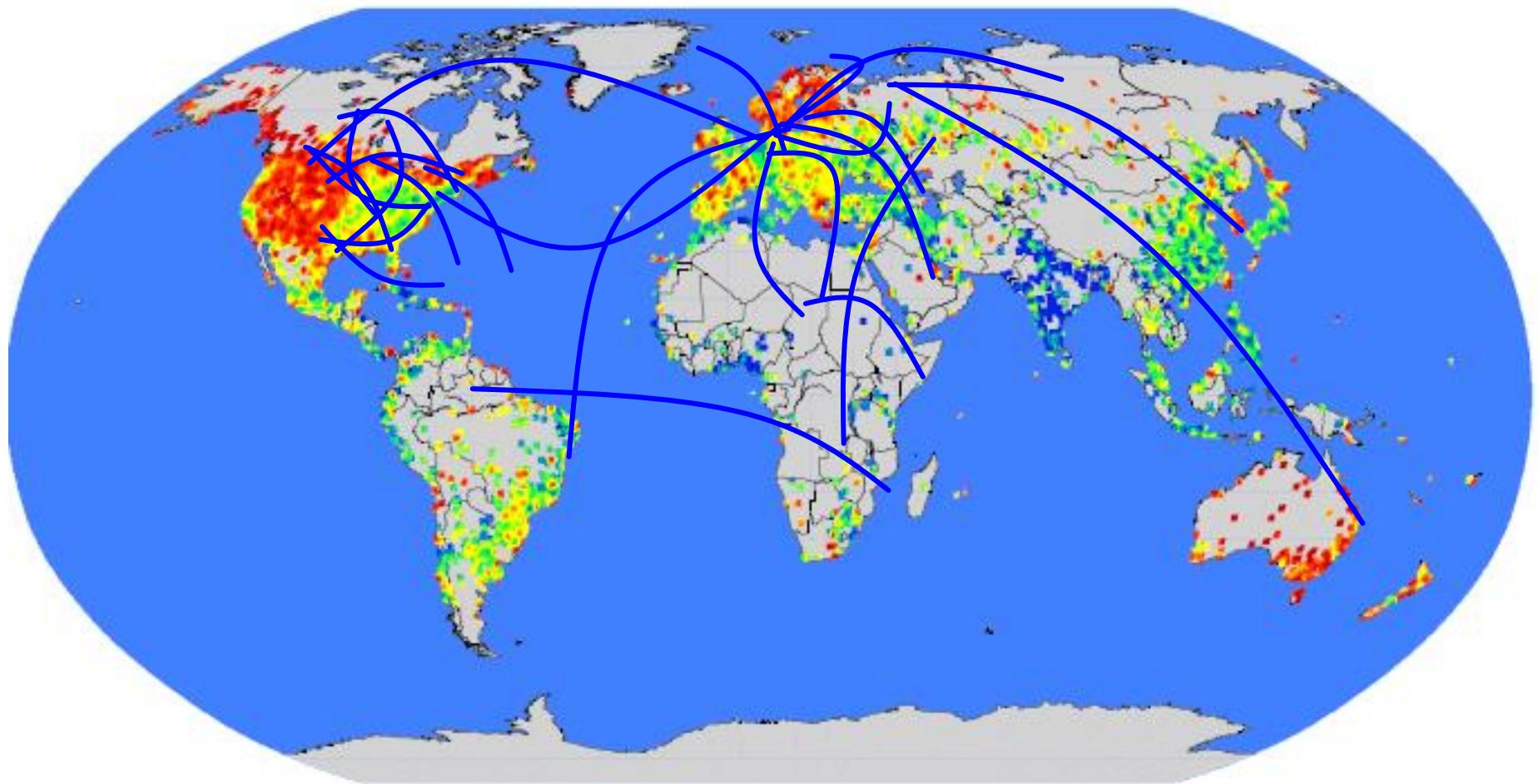
■ MSN Messenger activity in June 2006:

- 245 million users logged in
- 180 million users engaged in conversations
- More than 30 billion conversations
- More than 255 billion exchanged messages

Communication: Geography

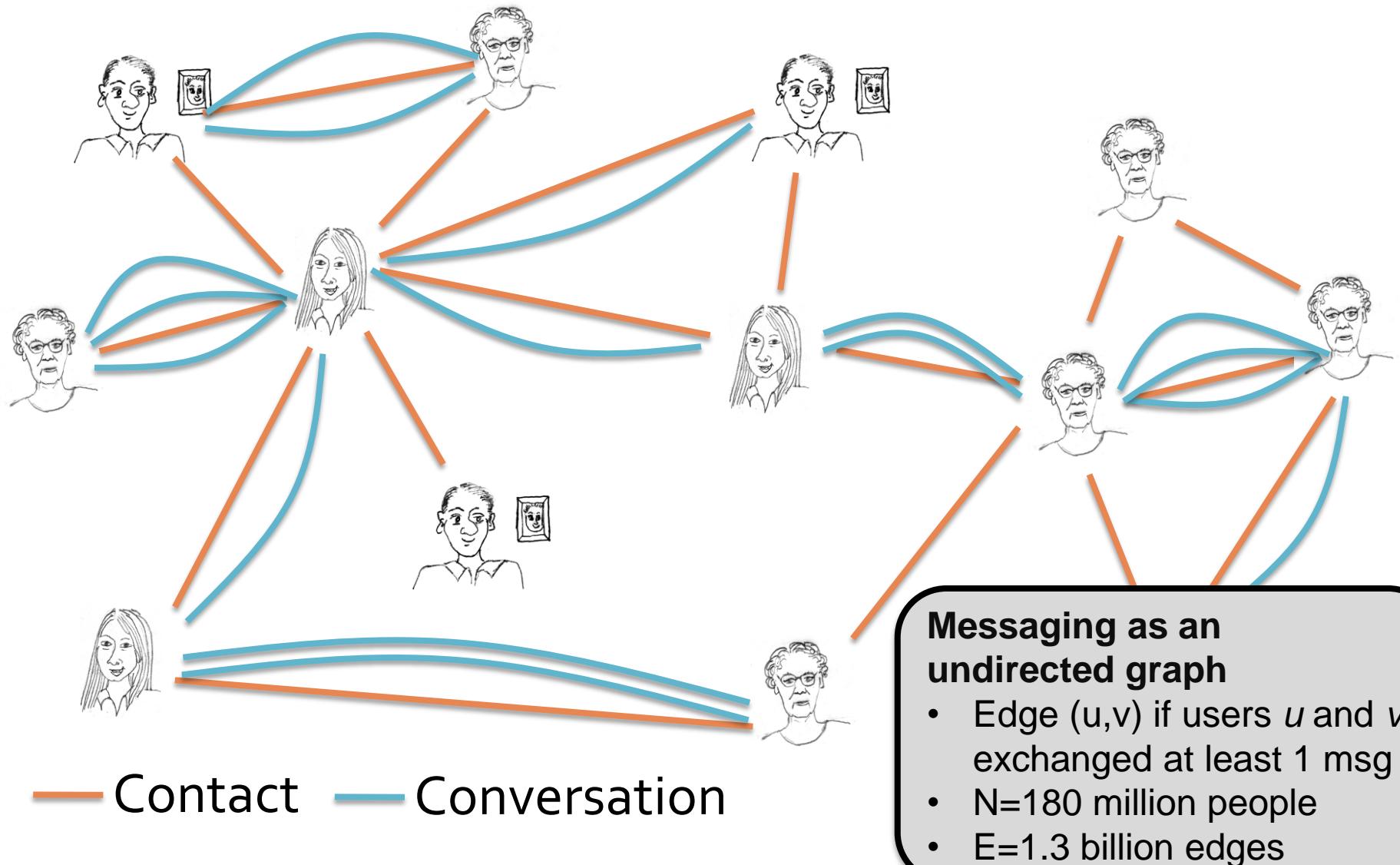


Communication network

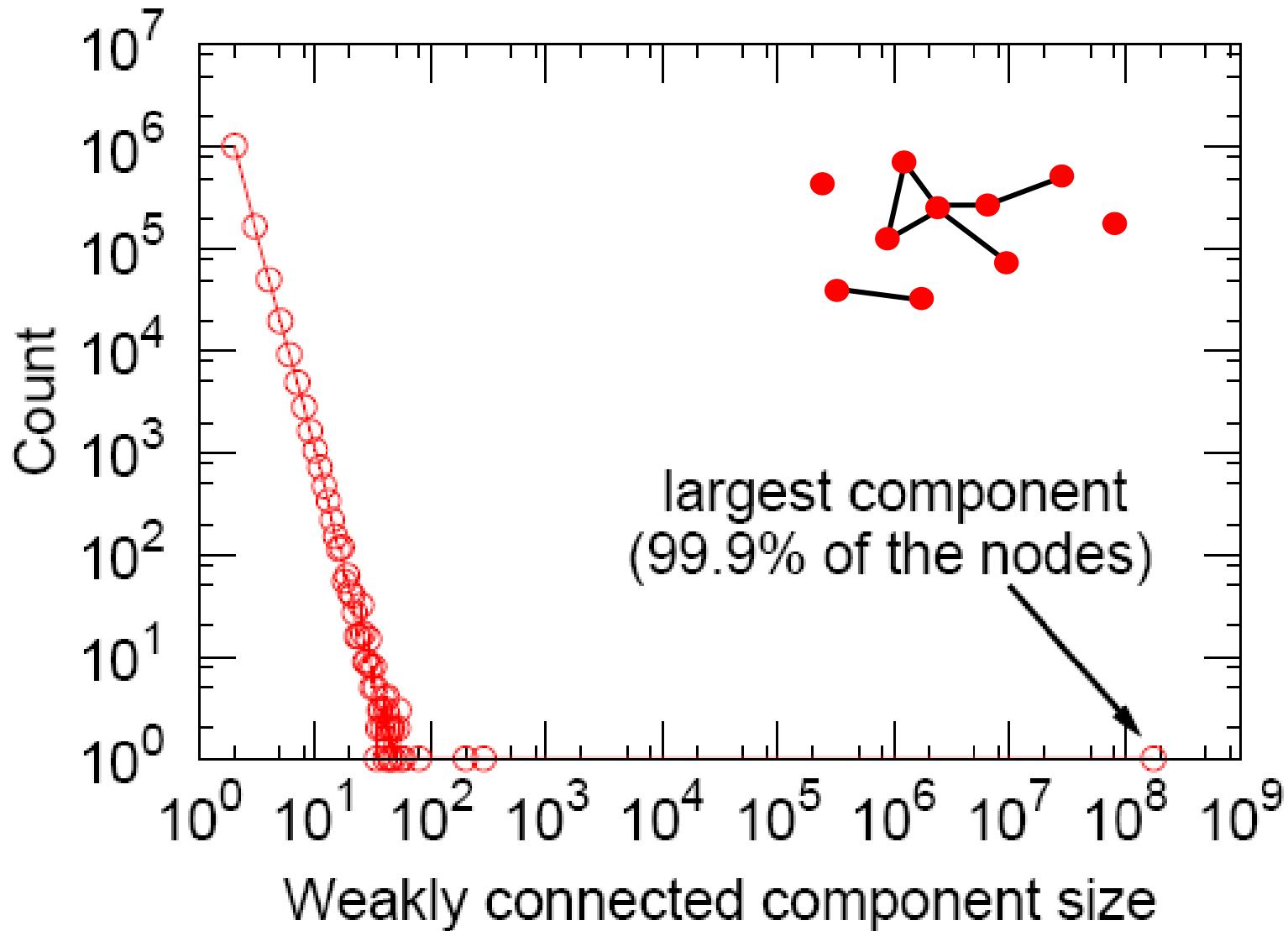


Network: 180M people, 1.3B edges

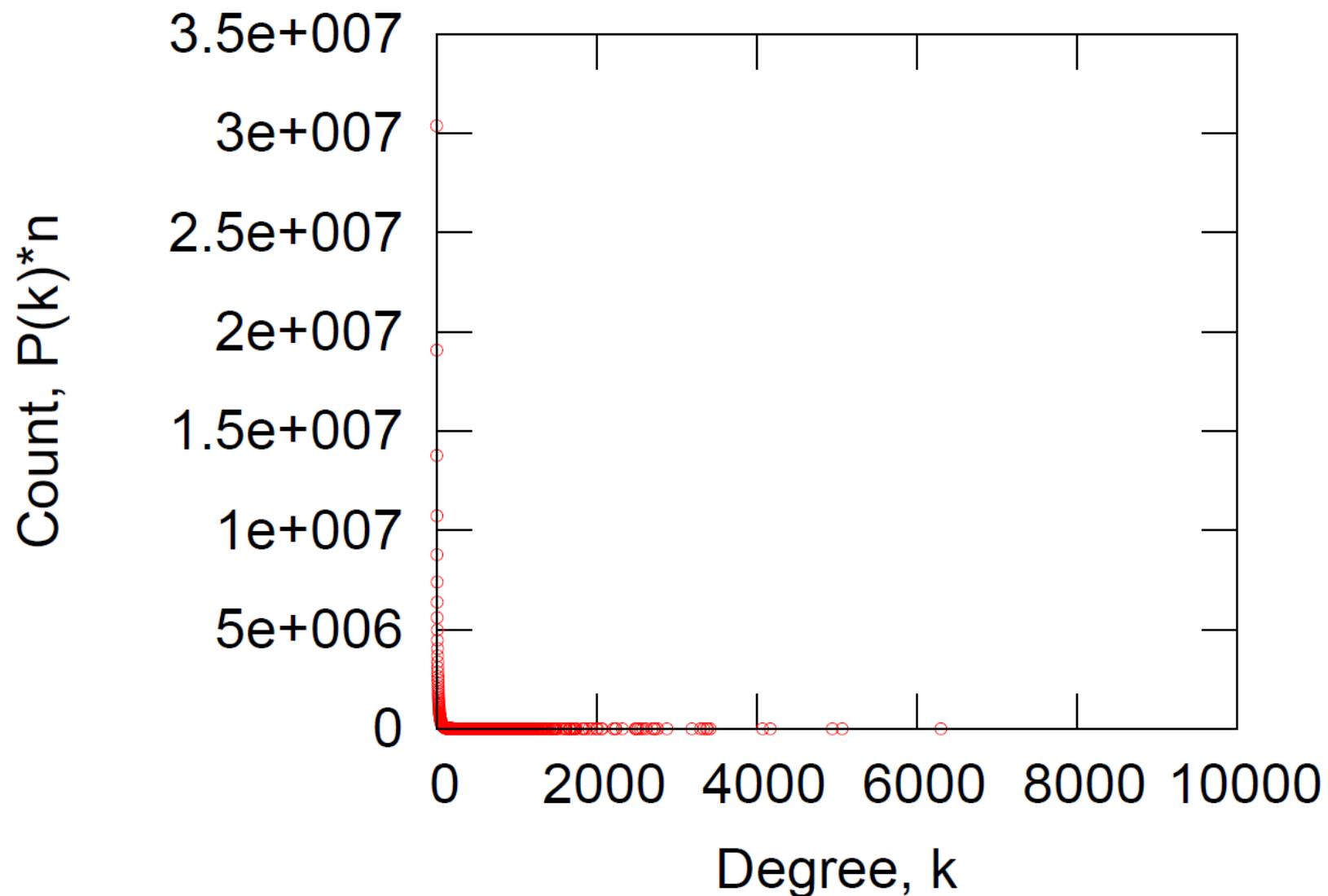
Messaging as a Multigraph



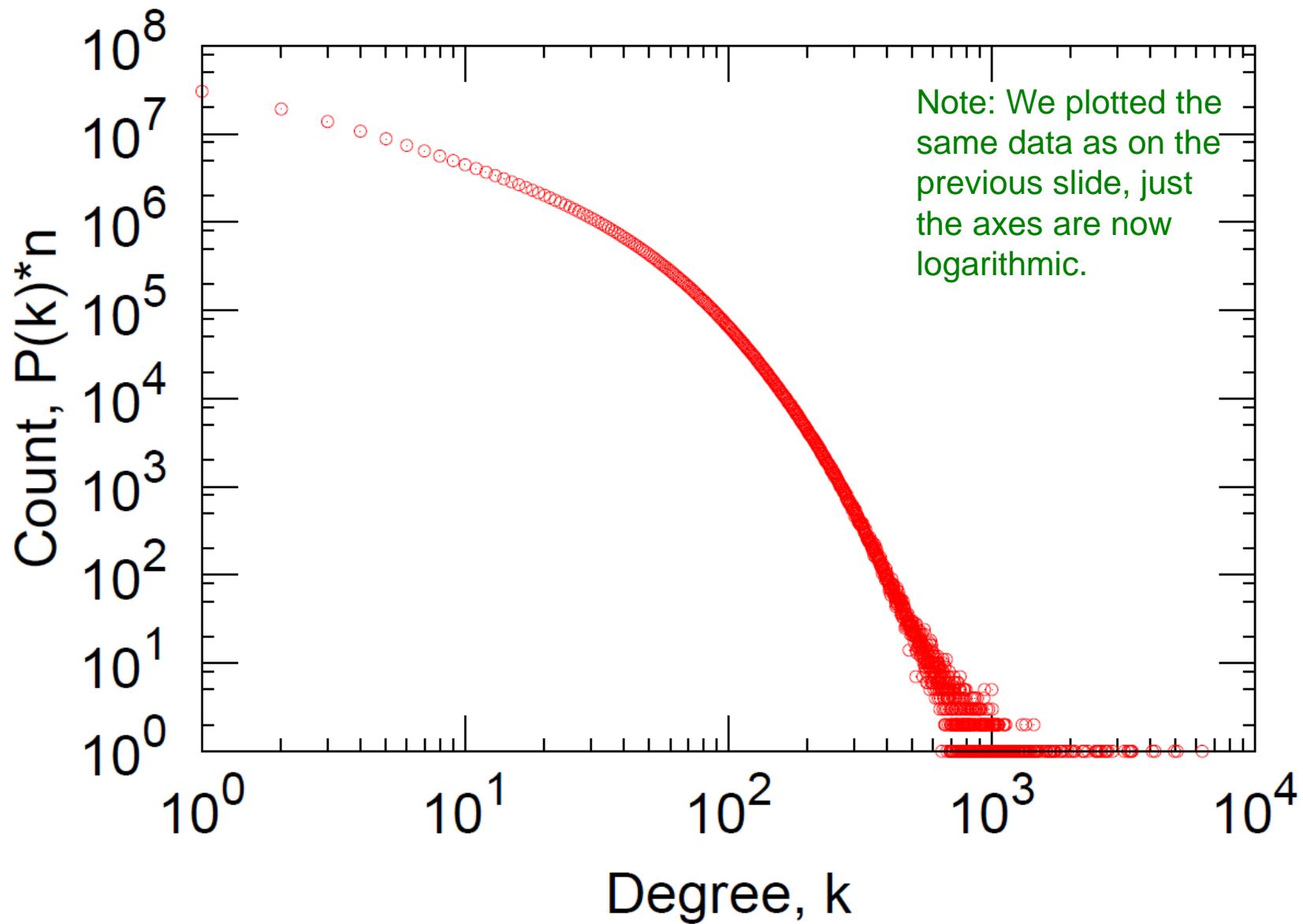
MSN Network: Connectivity



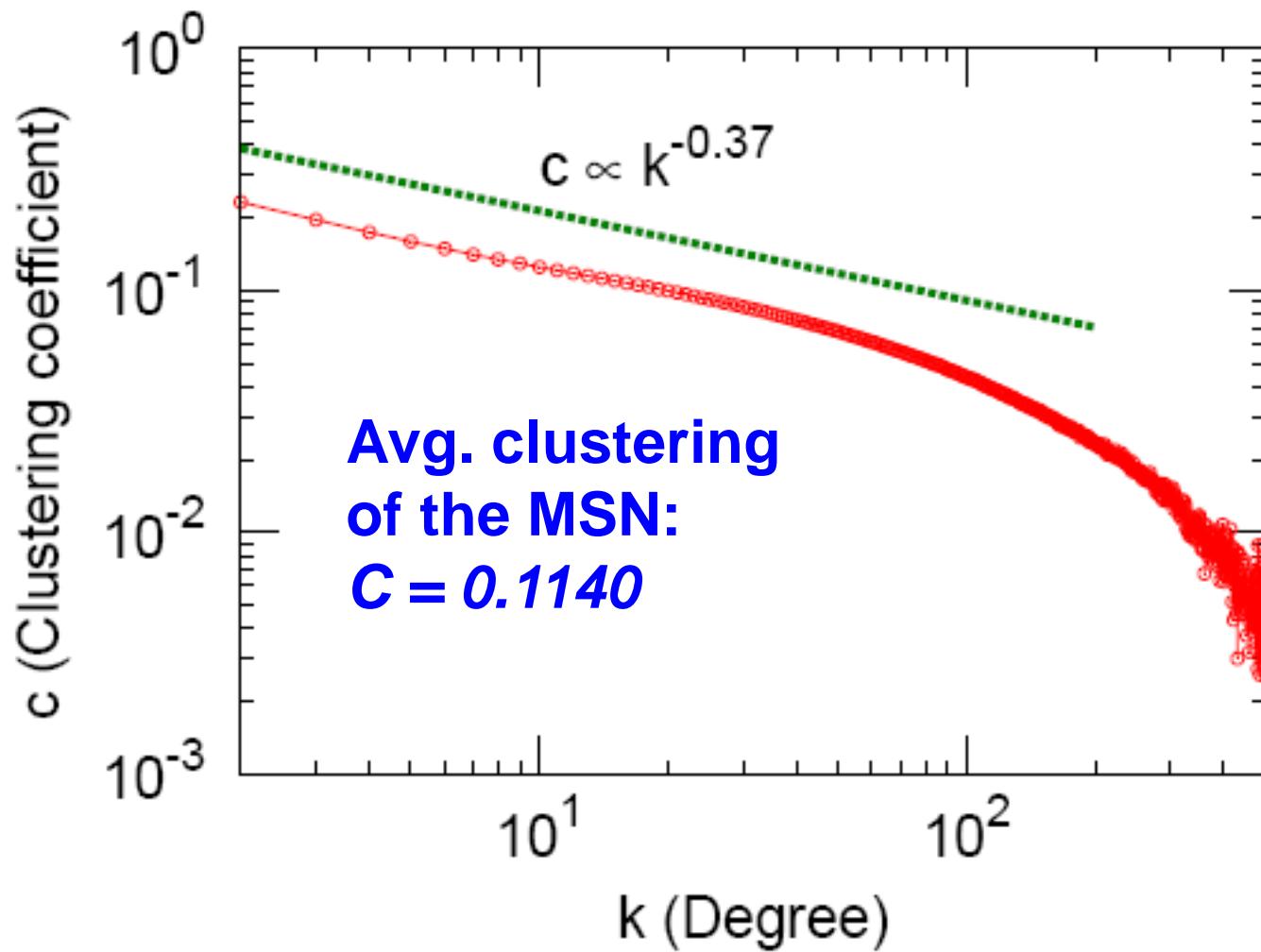
MSN: Degree Distribution



MSN: Log-Log Degree Distribution

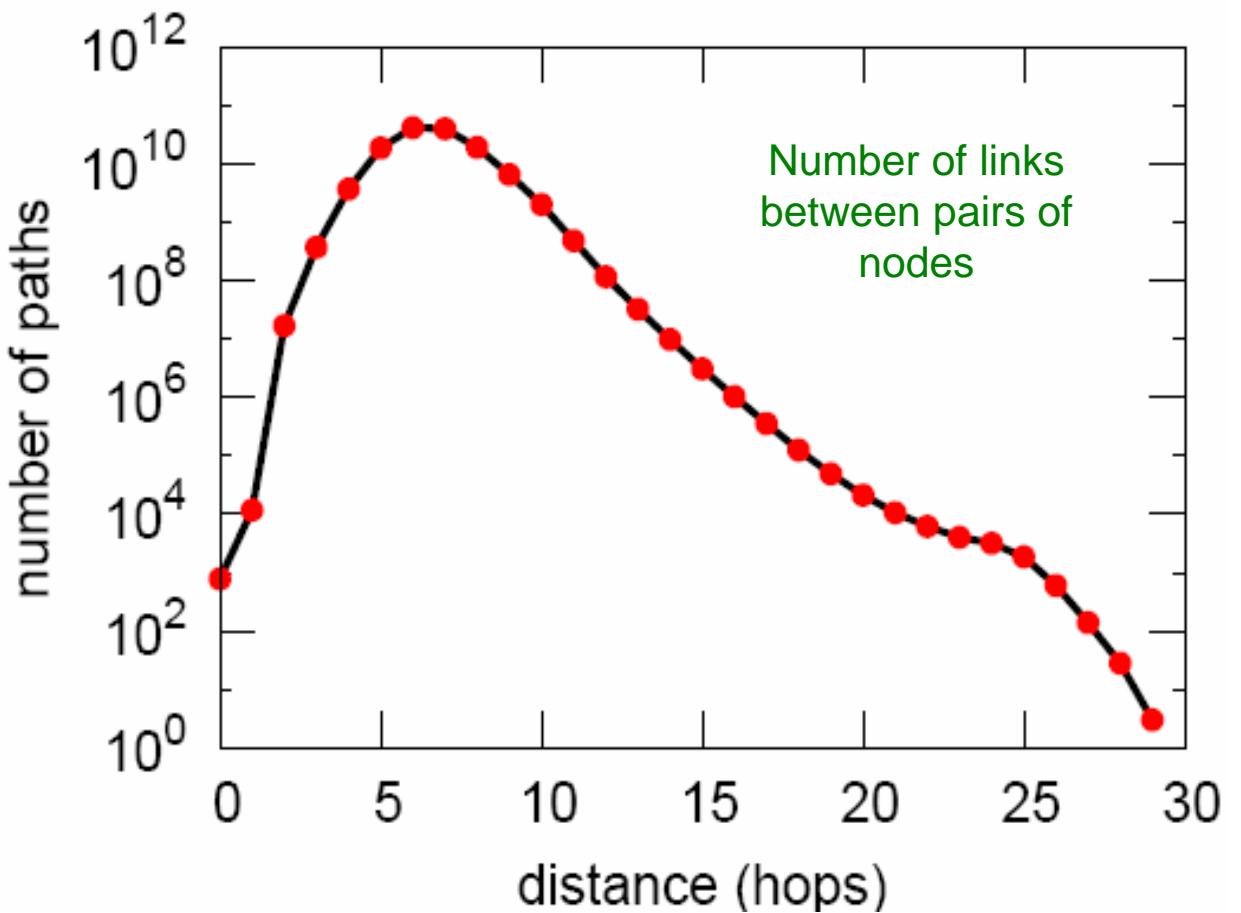


MSN: Clustering



C_k : average C_i of nodes i of degree k : $C_k = \frac{1}{N_k} \sum_{i:k_i=k} C_i$

MSN: Diameter



Number of links
between pairs of
nodes

Avg. path length **6.6**
90% of the nodes can be reached in < 8 hops

Steps	#Nodes
0	1
1	10
2	78
3	3,96
4	8,648
5	3,299,252
6	28,395,849
7	79,059,497
8	52,995,778
9	10,321,008
10	1,955,007
11	518,410
12	149,945
13	44,616
14	13,740
15	4,476
16	1,542
17	536
18	167
19	71
20	29
21	16
22	10
23	3
24	2
25	3

nodes as we do BFS out of a random node

MSN: Key Network Properties

Degree distribution:

Heavily skewed
avg. degree = 14.4

Path length:

6.6

Clustering coefficient:

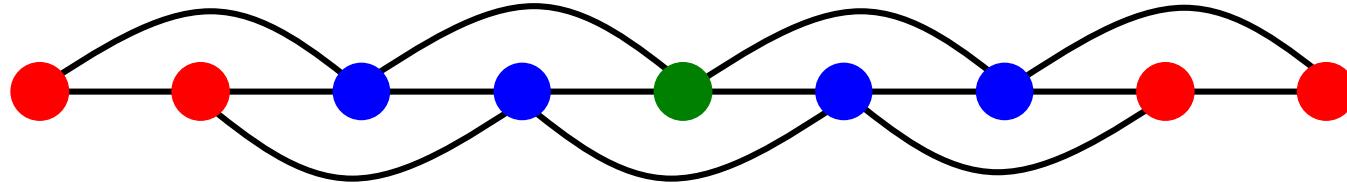
0.11

Are these values “expected”?

Are they “surprising”?

To answer this we need a null-model!

Is MSN Network like a “chain”?



- $P(k) = \delta(k-4)$ $k_i = 4$ for all nodes
- $C = \frac{1}{N} \left(\frac{1}{2} (N - 4) + 2 + 2 \frac{2}{3} \right) = \frac{1}{2}$ as $N \rightarrow \infty$
- Path length: $h_{max} = \frac{N-1}{2} = O(N)$
- Avg. shortest path-length: $\bar{h} < \frac{2}{N(N-1)} \frac{N-1}{2} \frac{N(N-1)}{2} = O(N)$
- So, we have: **Constant degree,**
Constant avg. clustering coeff.
Linear avg. path-length

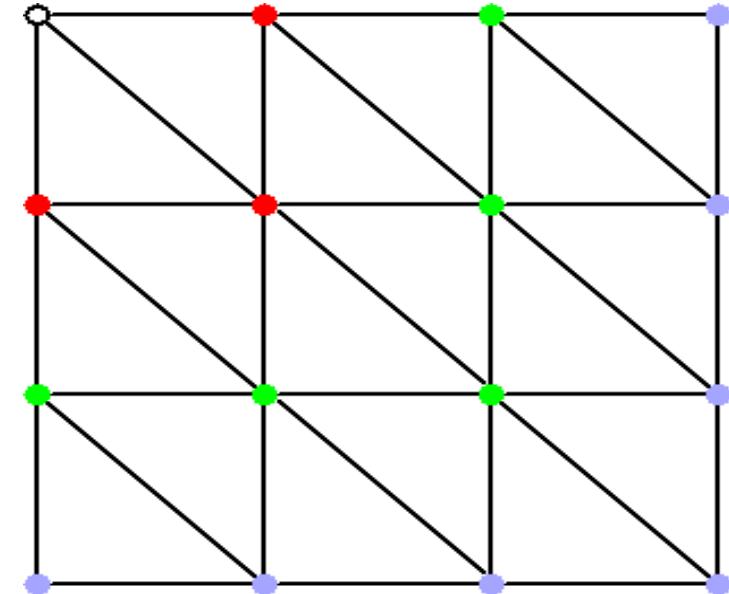
Note about calculations:
We are interested in
quantities as graphs get
 $(N \rightarrow \infty)$

Is MSN Network like a “grid”?

- $P(k) = \delta(k-6)$
 - $k=6$ for each inside node
- $C = 6/15$ for inside nodes
- **Path length:**

$$h_{\max} = O(\sqrt{N})$$

- **In general, for lattices:**
 - Average path-length is $\bar{h} \approx N^{1/D}$ (D... lattice dimensionality)
 - Constant degree, constant clustering coefficient



What did we learn so far?

MSN Network is
neither a chain
nor a grid