Measuring Networks, and Random Graph Models

CS224W: Analysis of Networks
Jure Leskovec, Stanford University
http://cs224w.stanford.edu

Network Properties: How to Measure a Network?

Plan: Key Network Properties

Degree distribution: $P(k)$

Path length: h

Clustering coefficient: C

Connected components: s

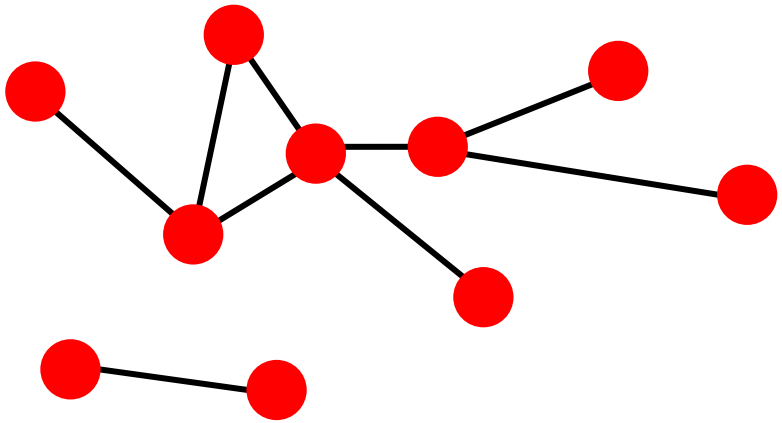
(1) Degree Distribution

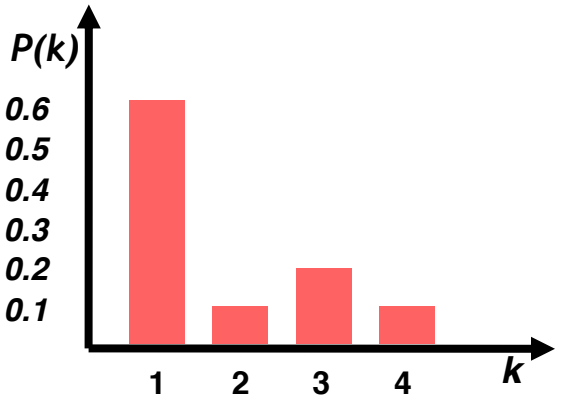
- **Degree distribution $P(k)$:** Probability that a randomly chosen node has degree k

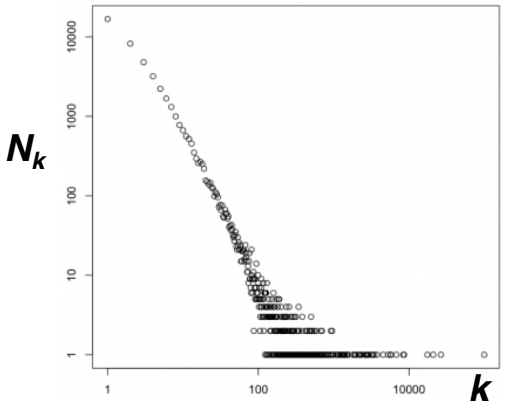
  N_k = # nodes with degree k

- Normalized histogram:

  $P(k) = N_k / N$ → **plot**





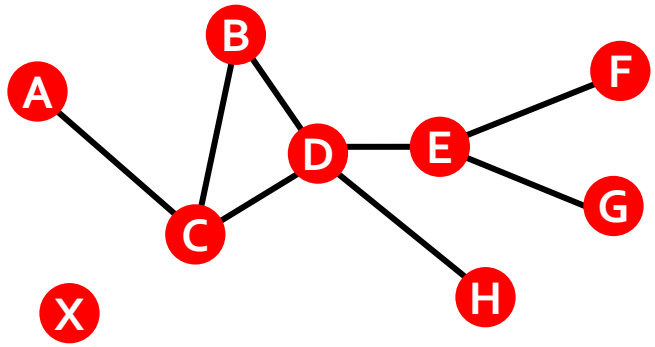


(2) Paths in a Graph

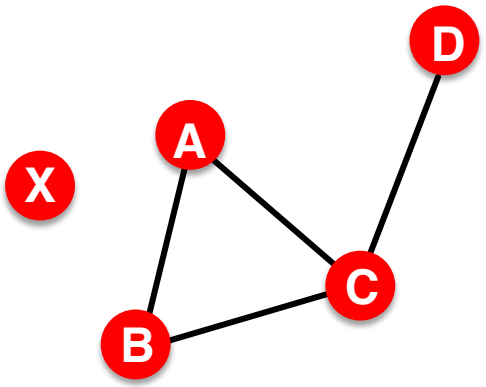
- A ***path*** is a sequence of nodes in which each node is linked to the next one

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

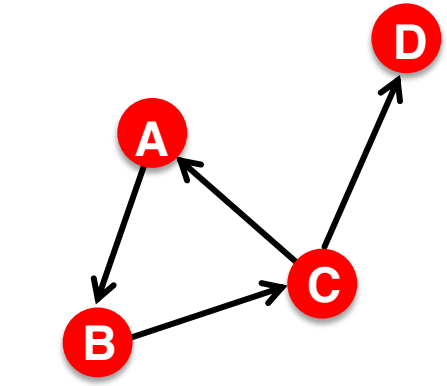
- Path can intersect itself and pass through the same edge multiple times
  - E.g.: ACBDCDEG
  - In a directed graph a path can only follow the direction of the "arrow"



Distance in a Graph



$h_{B,D} = 2$
$h_{A,X} = \infty$



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

- **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 not connected, the distance is usually defined as infinite
- In **directed graphs** paths need to follow the direction of the arrows
  - Consequence: Distance is **not symmetric**: $h_{B,C} \neq h_{C,B}$

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

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

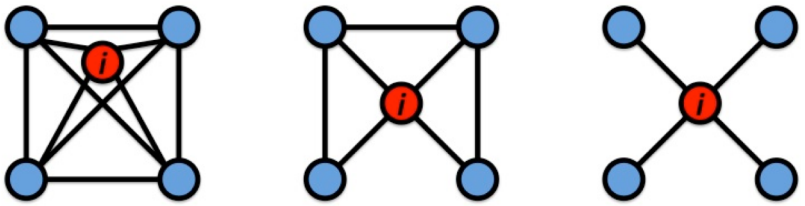
where h_{ij} is the distance from node i to node j
E_{max} is max number of edges (total number of node pairs) $= n(n-1)/2$

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

(3) Clustering Coefficient

- **Clustering coefficient:**
  - What portion of ***i***'s neighbors are connected?
  - Node ***i*** with degree k_i
  - $C_i \in [0,1]$
  - $C_i = \dfrac{2e_i}{k_i(k_i-1)}$

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



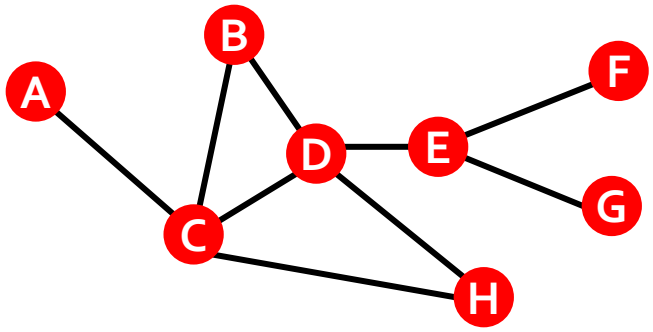
$C_i = 1$ $C_i = 1/2$ $C_i = 0$

- **Average clustering coefficient:** $C = \dfrac{1}{N}\sum_{i}^{N} C_i$

Clustering Coefficient

- **Clustering coefficient:**
  - What portion of ***i***'s neighbors are connected?
  - Node i with degree $\boldsymbol{k_i}$
  - $C_i = \dfrac{2e_i}{k_i(k_i-1)}$

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



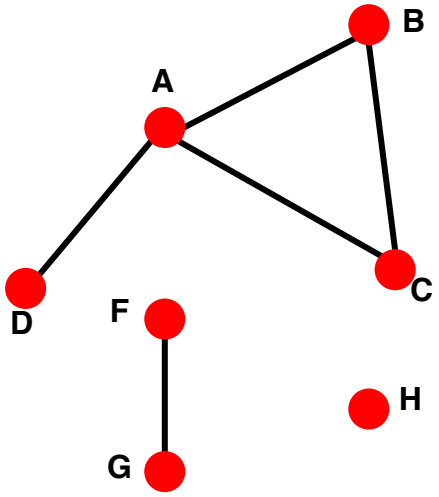
$k_B=2,\ e_B=1,\ C_B=2/2 = 1$

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

Avg. clustering: C=0.33

(4) Connectivity

- **Size of the largest connected component**
  - Largest set where any two vertices can be joined by a path
- **Largest component = Giant component**



How to find connected components:

- Start from random node and perform Breadth First Search (BFS)
- Label the nodes BFS visited
- If all nodes are visited, the network is connected
- Otherwise find an unvisited node and repeat BFS

Summary: Key Network Properties

Degree distribution: $P(k)$

Path length: h

Clustering coefficient: C

Connected components: s

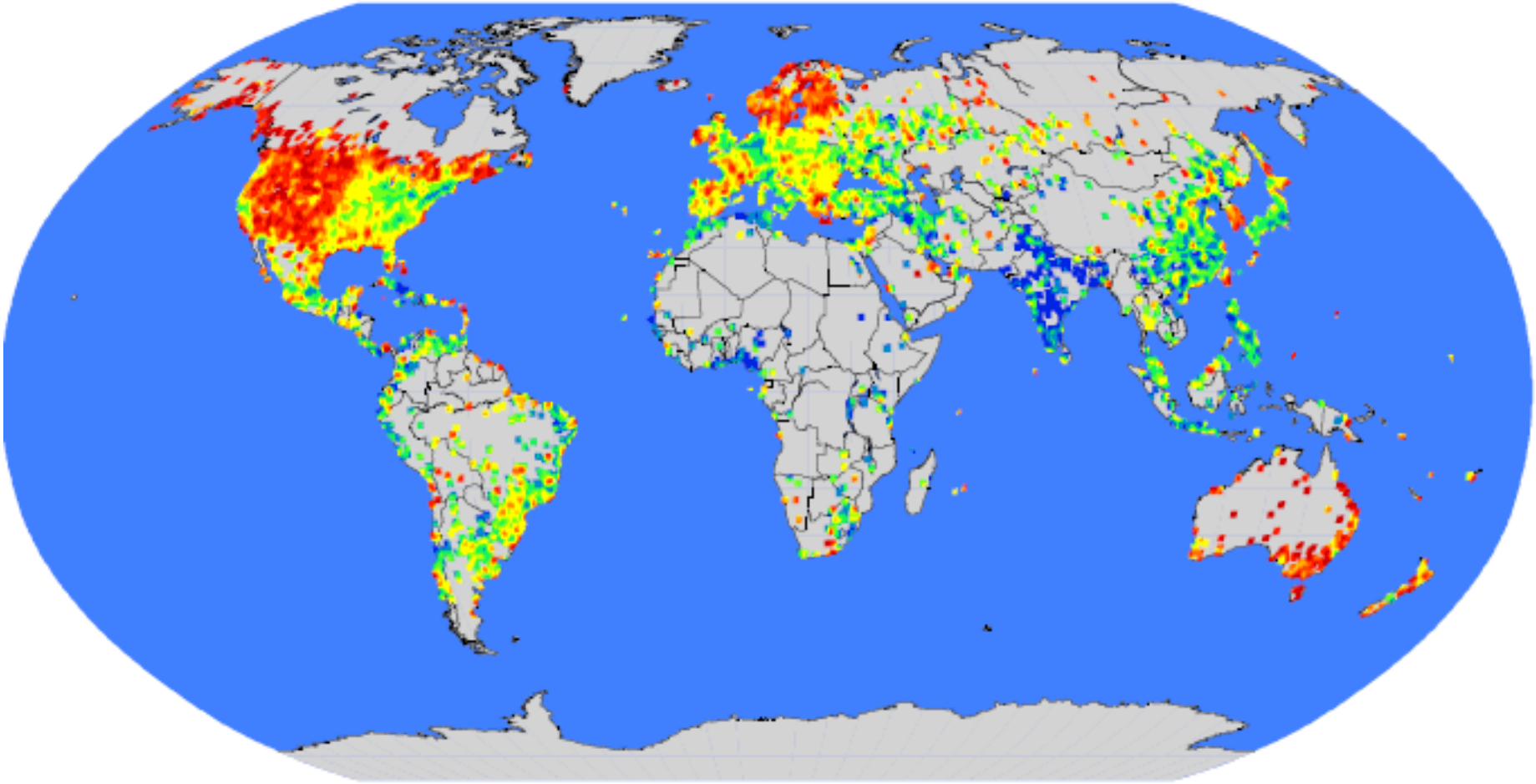
Let's measure P(k), h and C on a real-world network!

MSN Messenger

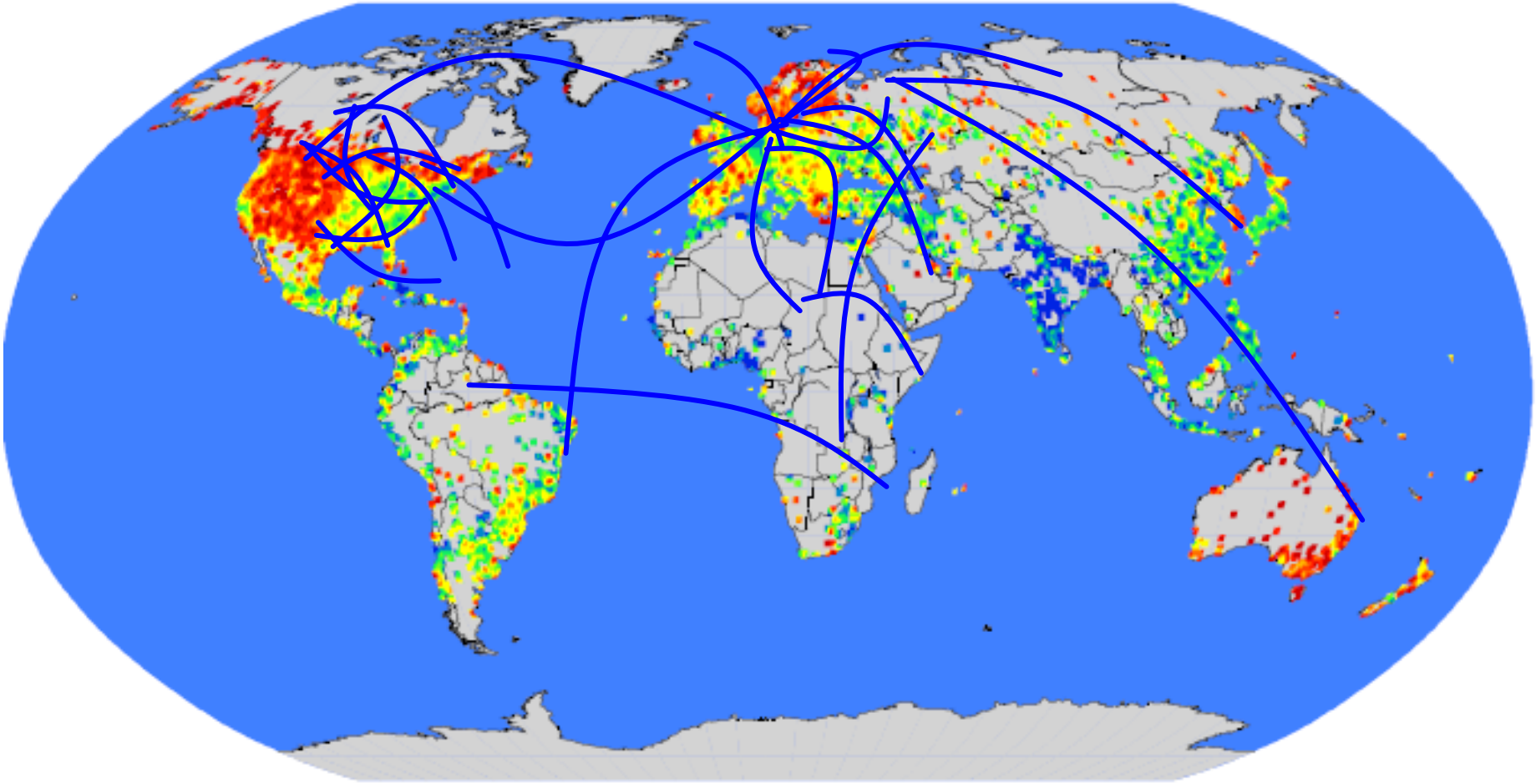
MSN Messenger.

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

Communication: Geography

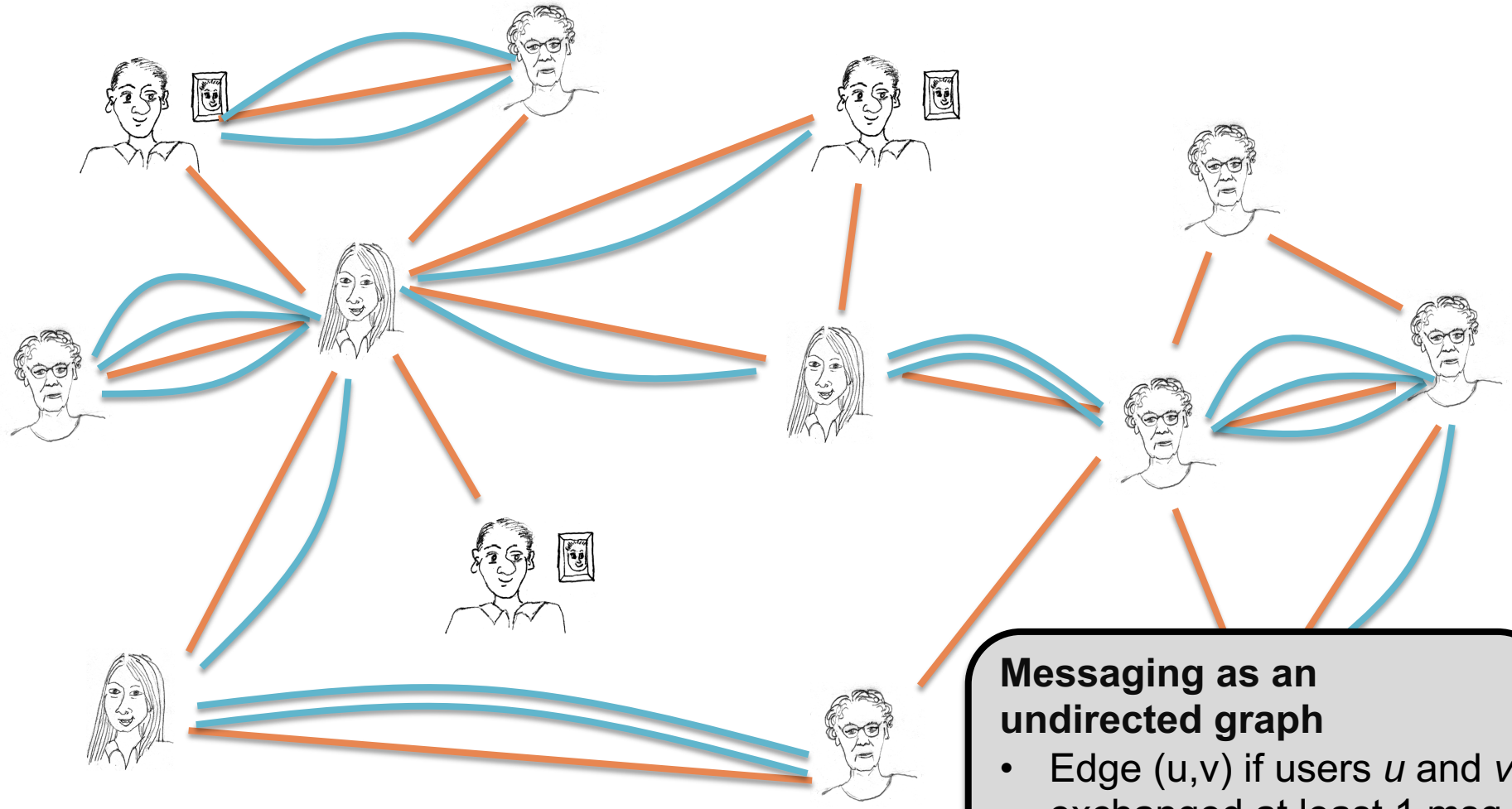


Communication Network



Network: 180M people, 1.3B edges

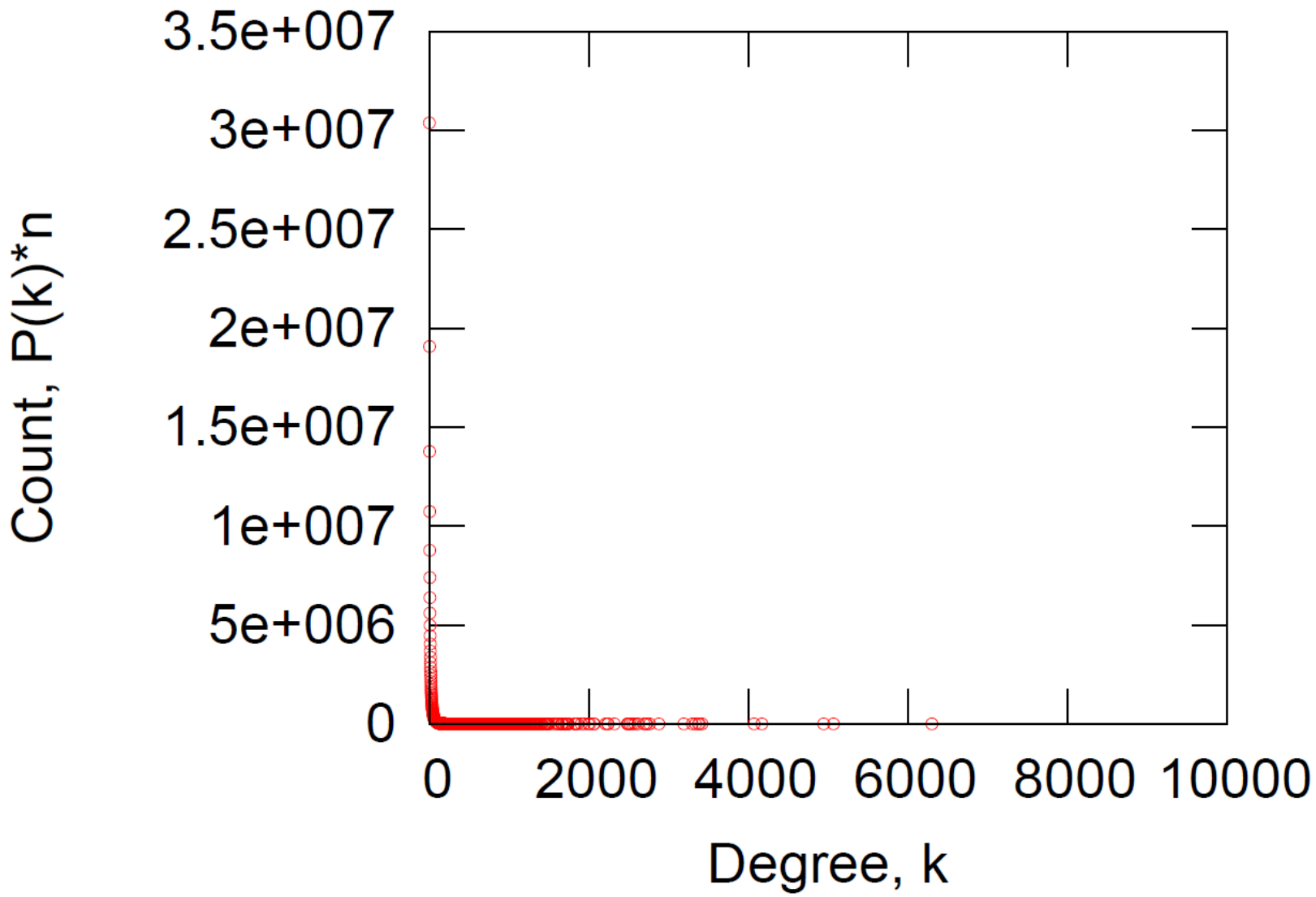
Messaging as a Multigraph



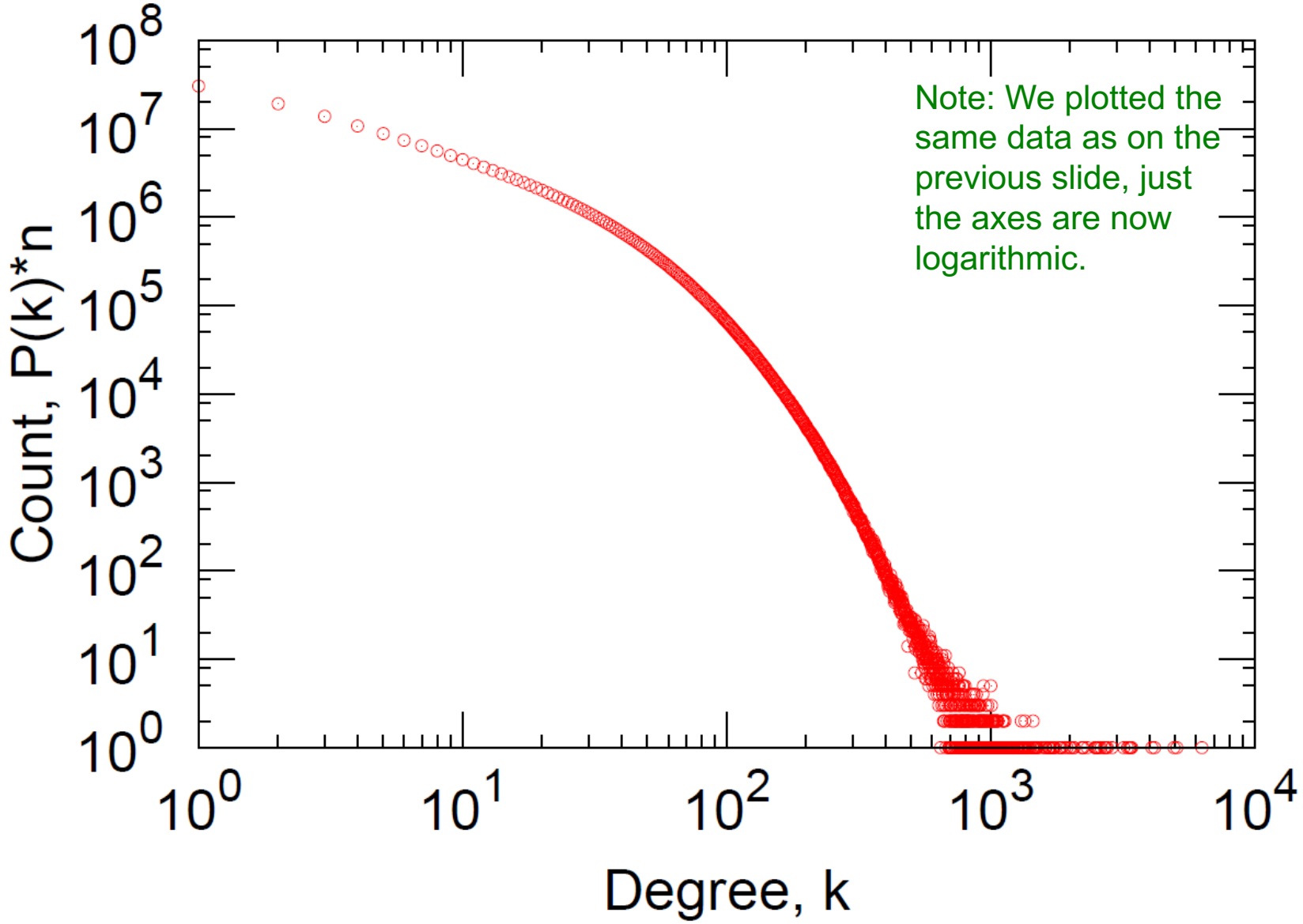
Messaging as an undirected graph

- Edge (u,v) if users *u* and *v* exchanged at least 1 msg
- N=180 million people
- E=1.3 billion edges

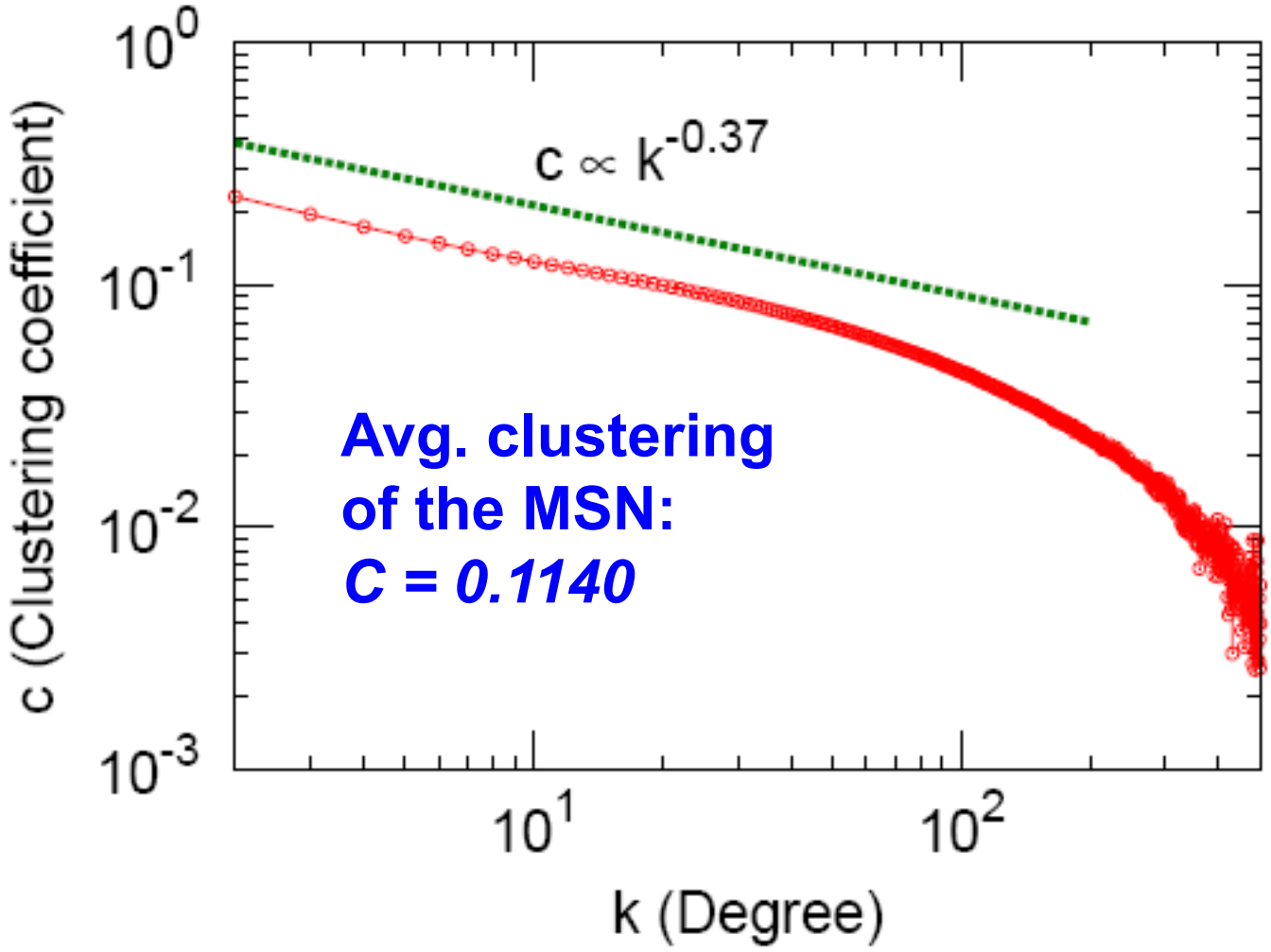
MSN: (1) Degree Distribution



MSN: Log-Log Degree Distribution

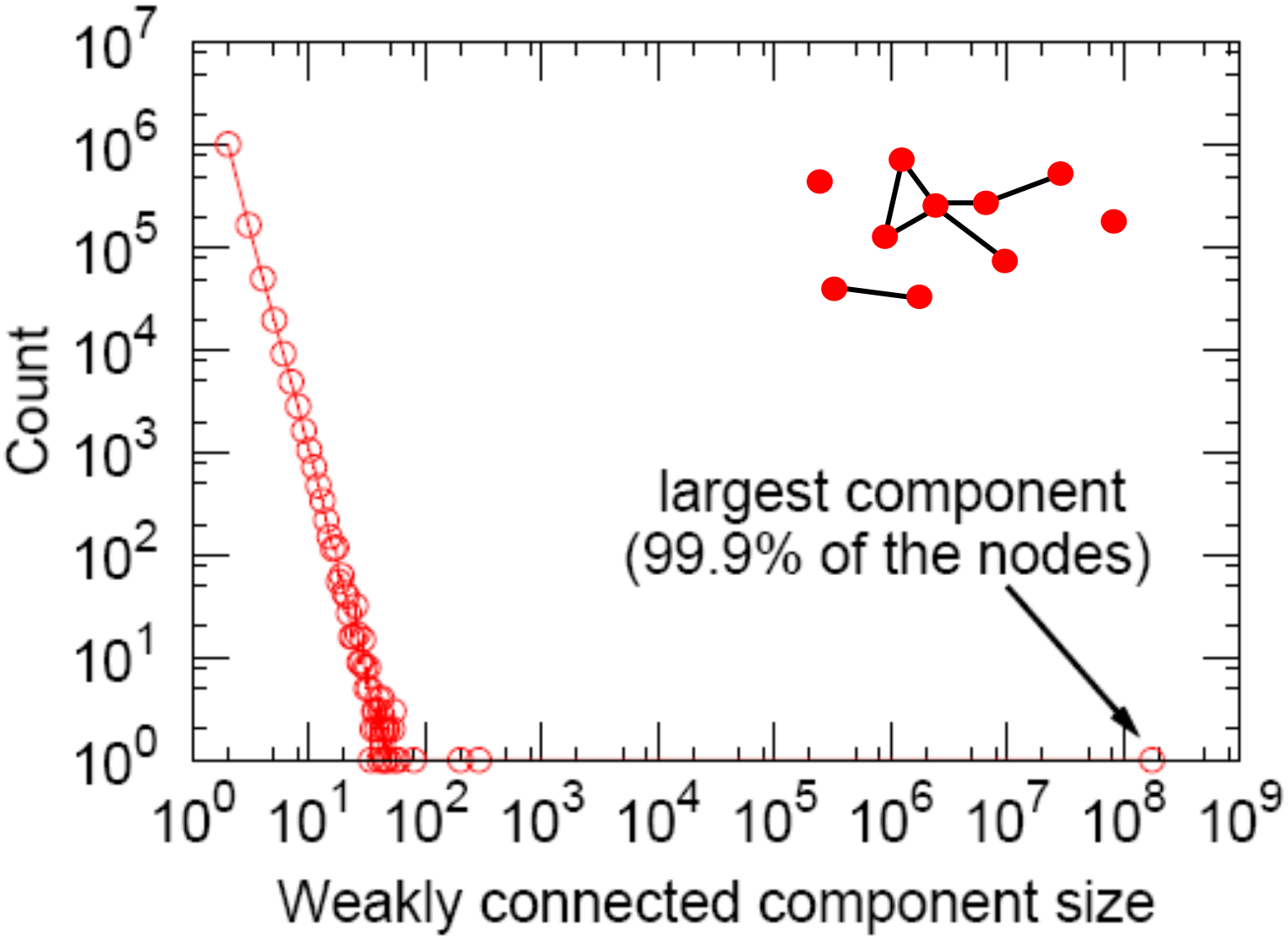


MSN: (2) Clustering

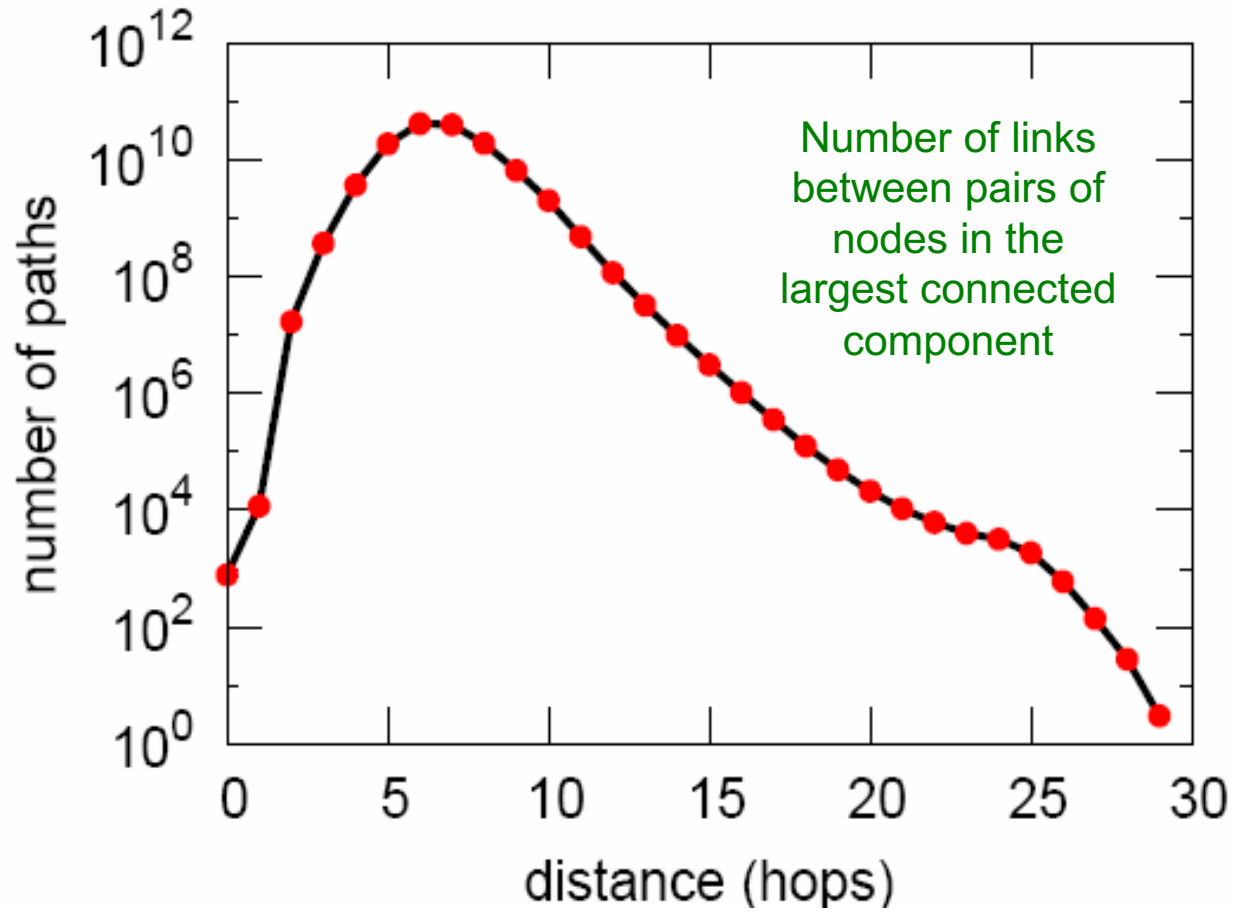


C_k: average C_i of nodes i of degree k: $C_k = \frac{1}{N_k} \sum_{i:k_i=k} C_i$

MSN: (3) Connected Components



MSN: (4) Diameter of WCC



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

nodes as we do BFS out of a random node

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

MSN: Key Network Properties

Degree distribution: *Heavily skewed*
avg. degree= 14.4

Path length: *6.6*

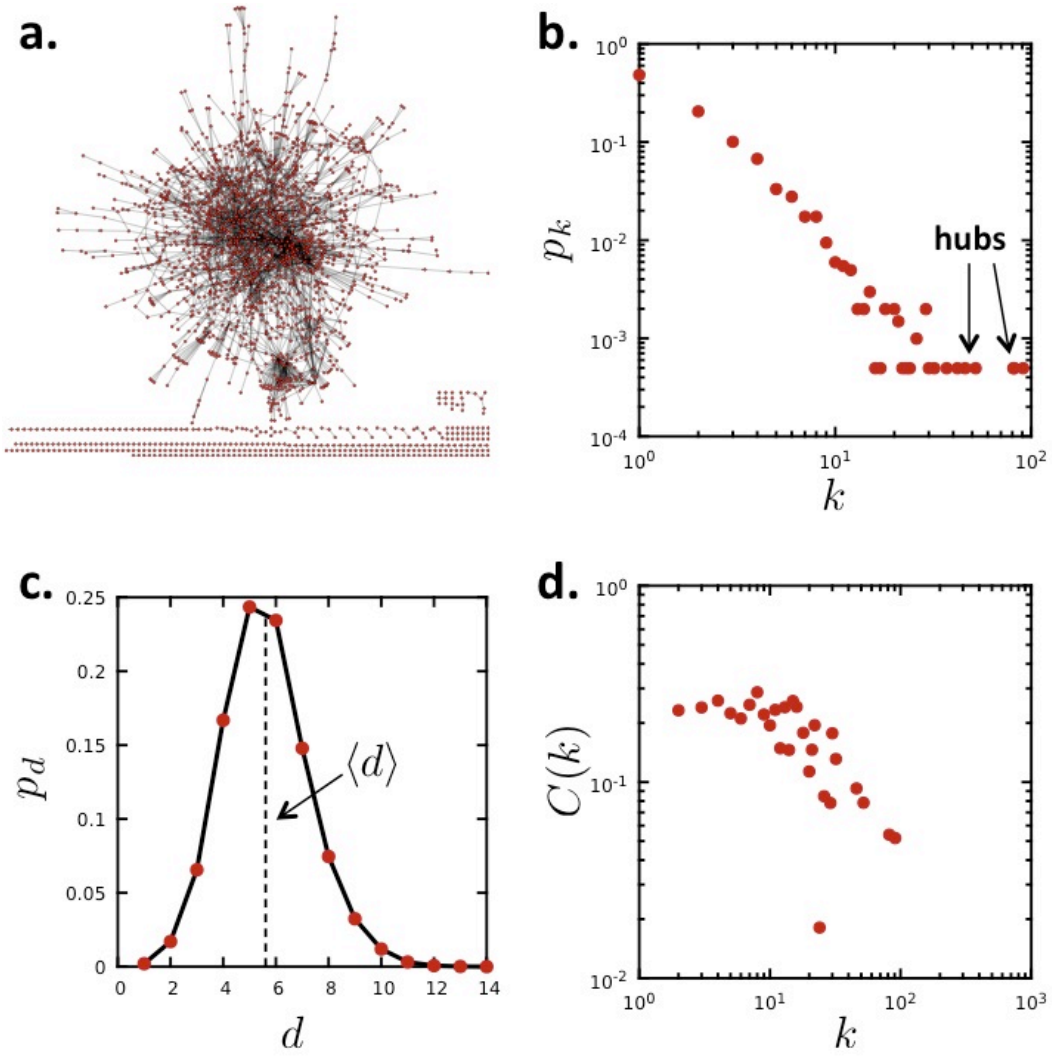
Clustering coefficient: *0.11*

Connectivity: *giant component*

Are these values "expected"?
Are they "surprising"?

To answer this we need a null-model!

Another example: PPI Network



a. Undirected network

N=2,018 proteins as nodes

E=2,930 binding interactions as links.

b. Degree distribution:

Skewed. Average degree <k>=2.90

c. Diameter:

Avg. path length = 5.8

d. Clustering:

Avg. clustering = 0.12

Connectivity: 185 components

the largest component 1,647 nodes (81% of nodes)

Erdös-Renyi Random Graph Model

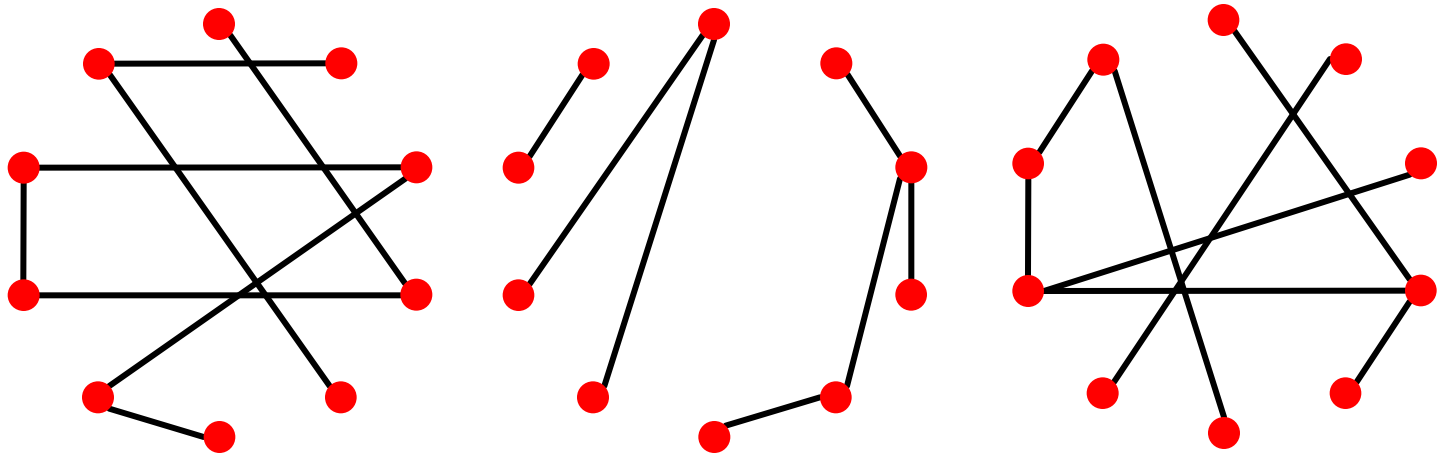
Simplest Model of Graphs

- **Erdös-Renyi Random Graphs** [Erdös-Renyi, '60]
- **Two variants:**
  - $G_{n,p}$: undirected graph on n nodes and each edge (u,v) appears i.i.d. with probability p
  - $G_{n,m}$: undirected graph with n nodes, and m uniformly at random picked edges

What kind of networks do such models produce?

Random Graph Model

- ***n*** **and** ***p*** **do not uniquely determine the graph!**
  - The graph is a result of a random process
- We can have many different realizations given the same ***n*** and ***p***



n = 10
p= 1/6

Properties of G_{np}

Degree distribution: $P(k)$

Path length: h

Clustering coefficient: C

What are the values of these properties for G_{np}?

Degree Distribution

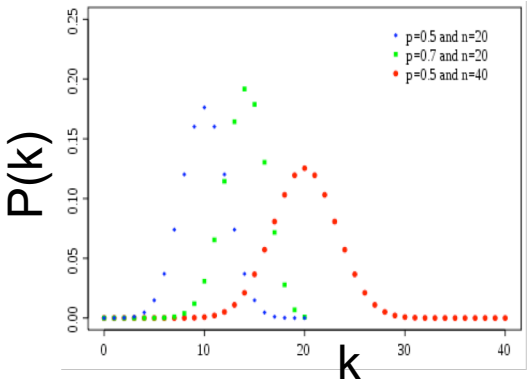
- **Fact:** Degree distribution of G_{np} is binomial.
- Let $P(k)$ denote the fraction of nodes with degree k:

$$P(k) = \binom{n-1}{k} p^k (1-p)^{n-1-k}$$

Select k nodes out of $n-1$

Probability of having k edges

Probability of missing the rest of the $n-1-k$ edges



Mean, variance of a binomial distribution

$$\bar{k} = p(n-1)$$

$$\sigma^2 = p(1-p)(n-1)$$

$$\frac{\sigma}{\bar{k}} = \left[\frac{1-p}{p}\frac{1}{(n-1)}\right]^{1/2} \approx \frac{1}{(n-1)^{1/2}}$$

By the law of large numbers, as the network size increases, the distribution becomes increasingly narrow—we are increasingly confident that the degree of a node is in the vicinity of k.

Clustering Coefficient of G_{np}

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

  Where e_i is the number of edges between i's neighbors

- Edges in G_{np} appear i.i.d. with prob. p

- **So, expected E[e_i] is:** $= p\frac{k_i(k_i-1)}{2}$

  Each pair is connected with prob. p

  Number of distinct pairs of neighbors of node i of degree k_i

- **Then *E[C]*:** $= \frac{p \cdot k_i(k_i-1)}{k_i(k_i-1)} = p = \frac{\bar{k}}{n-1} \approx \frac{\bar{k}}{n}$

Clustering coefficient of a random graph is small.
If we generate bigger and bigger graphs with fixed avg. degree k (that is we set $p = k \cdot 1/n$), then *C* decreases with the graph size *n*.

Network Properties of G_{np}

Degree distribution: $P(k) = \binom{n-1}{k} p^k (1-p)^{n-1-k}$

Clustering coefficient: $C = p = \bar{k}/n$

Path length: ***next!***

Connectivity:

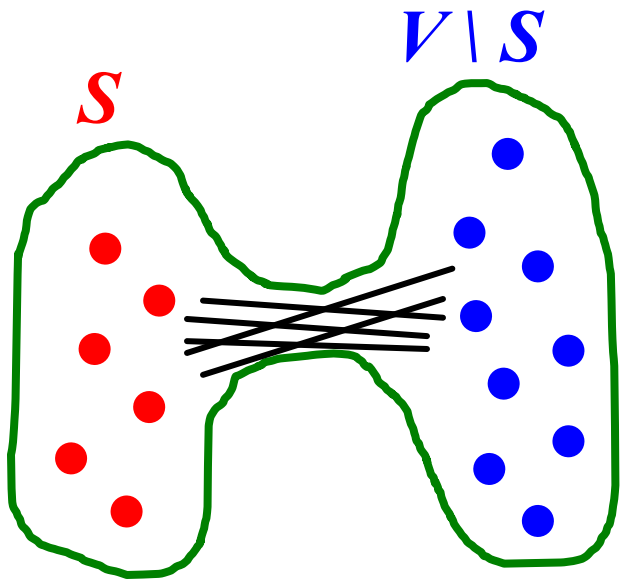
Def: Expansion

- Graph $G(V, E)$ has **expansion α**: if $\forall S \subseteq V$:

  $$\#\text{ of edges leaving } S \geq \alpha \cdot \min(|S|, |V \backslash S|)$$

- **Or equivalently:**

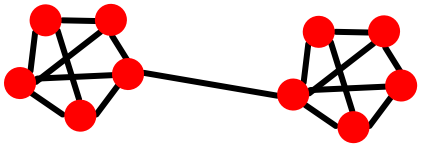
$$\alpha = \min_{S \subseteq V} \frac{\#edges\ leaving\ S}{\min(|S|, |V \setminus S|)}$$



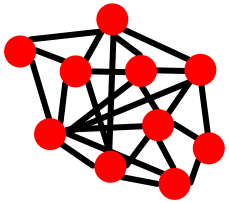
Expansion: Measures Robustness

$$\alpha = \min_{S \subseteq V} \frac{\#\,edges\ leaving\ S}{\min(|S|, |V \setminus S|)}$$

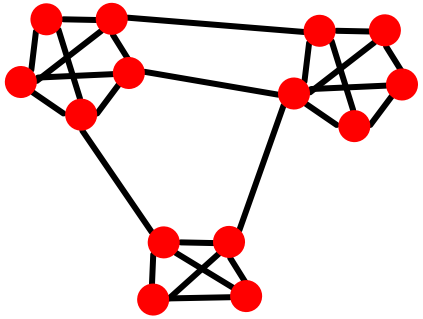
- Expansion is **measure of robustness:**
  - To disconnect l nodes, we need to cut $\geq \alpha \cdot l$ edges
- Low expansion:



- High expansion:

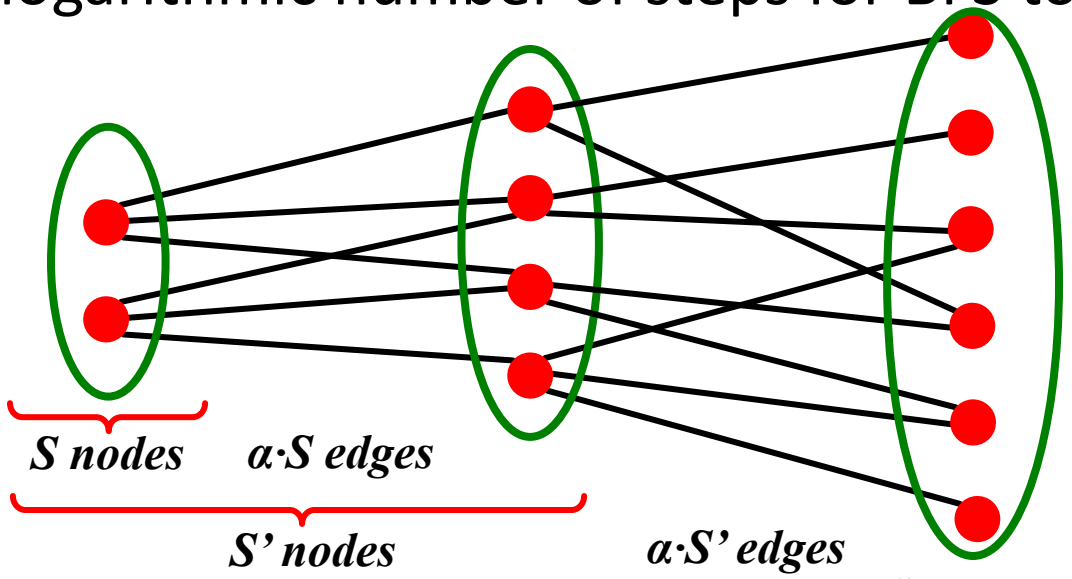


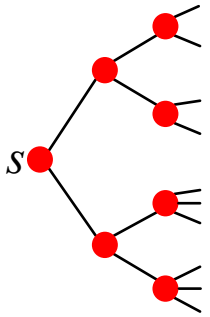
- Social networks:
  - "Communities"



Expansion: Random Graphs

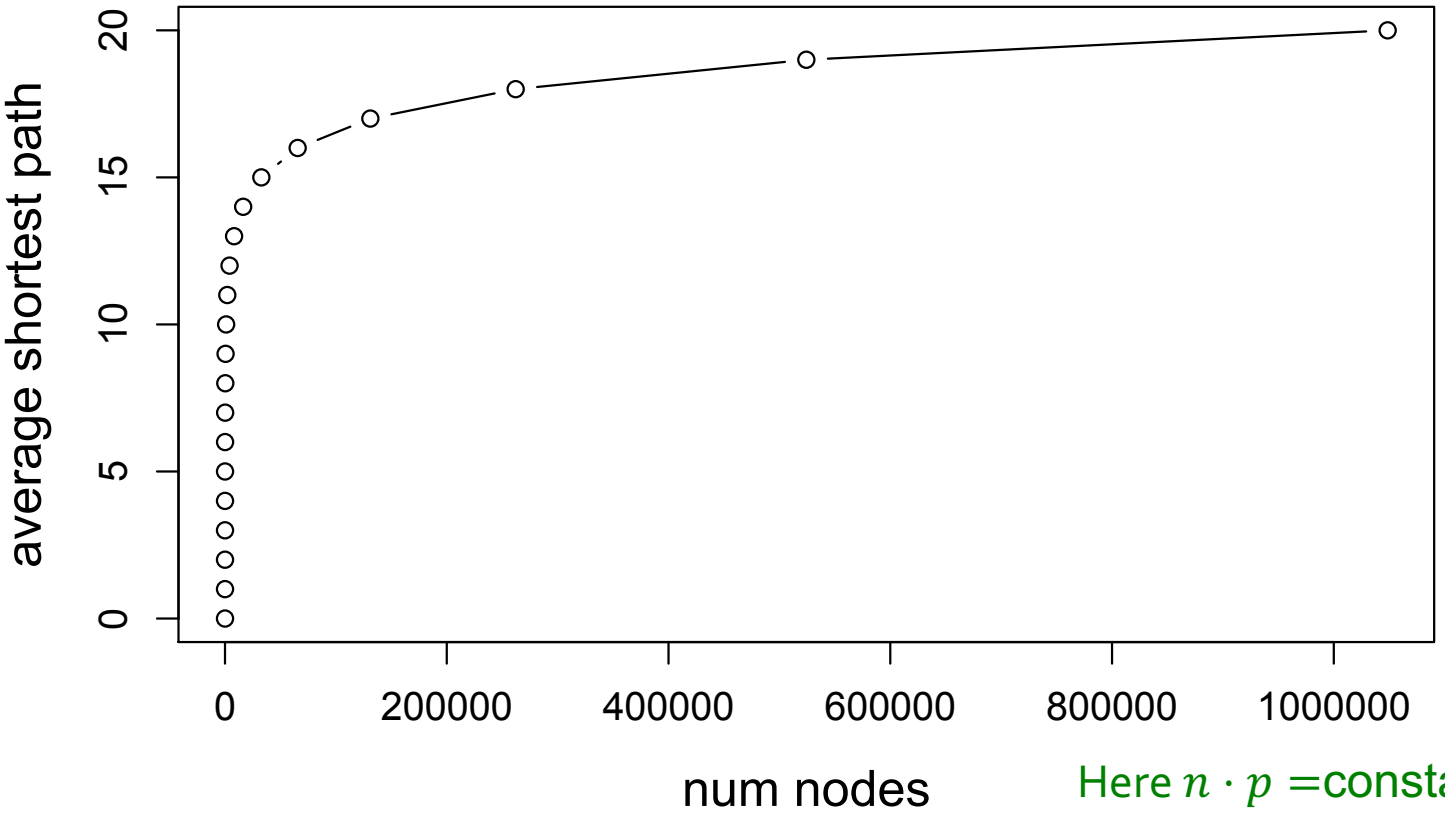
- **Fact:** In a graph on n nodes with expansion α for all pairs of nodes there is a path of length $O((\log n)/\alpha)$.
- **Random graph G_{np}:**
  For $\log n > np > c$, diam(G_{np}) = $O(\log n / \log (np))$
  - Random graphs have good expansion so it takes a logarithmic number of steps for BFS to visit all nodes





Erdös-Renyi avg. shortest path

Erdös-Renyi Random Graph can grow very large but nodes will be just a few hops apart



Here $n \cdot p$ =constant
That is, avg deg k is const

Network Properties of G_{np}

Degree distribution: $P(k) = \binom{n-1}{k} p^k (1-p)^{n-1-k}$

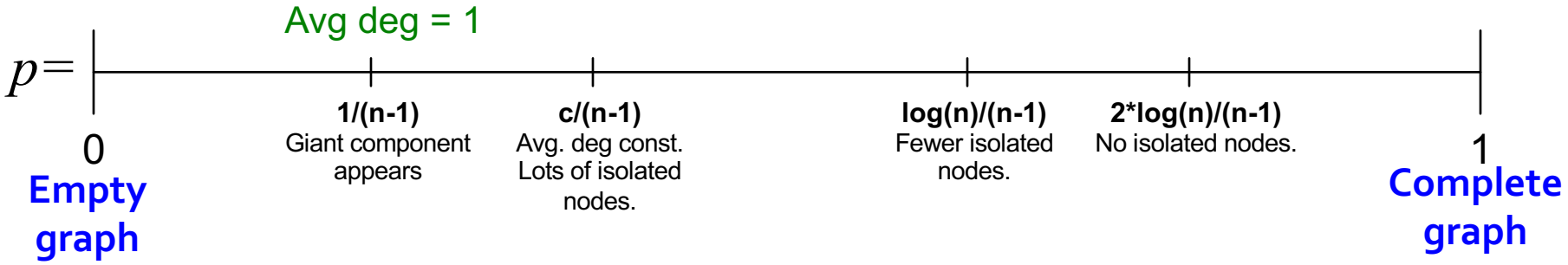
Path length: $O(\log n)$

Clustering coefficient: $C = p = \bar{k} / n$

Connected components: ***next!***

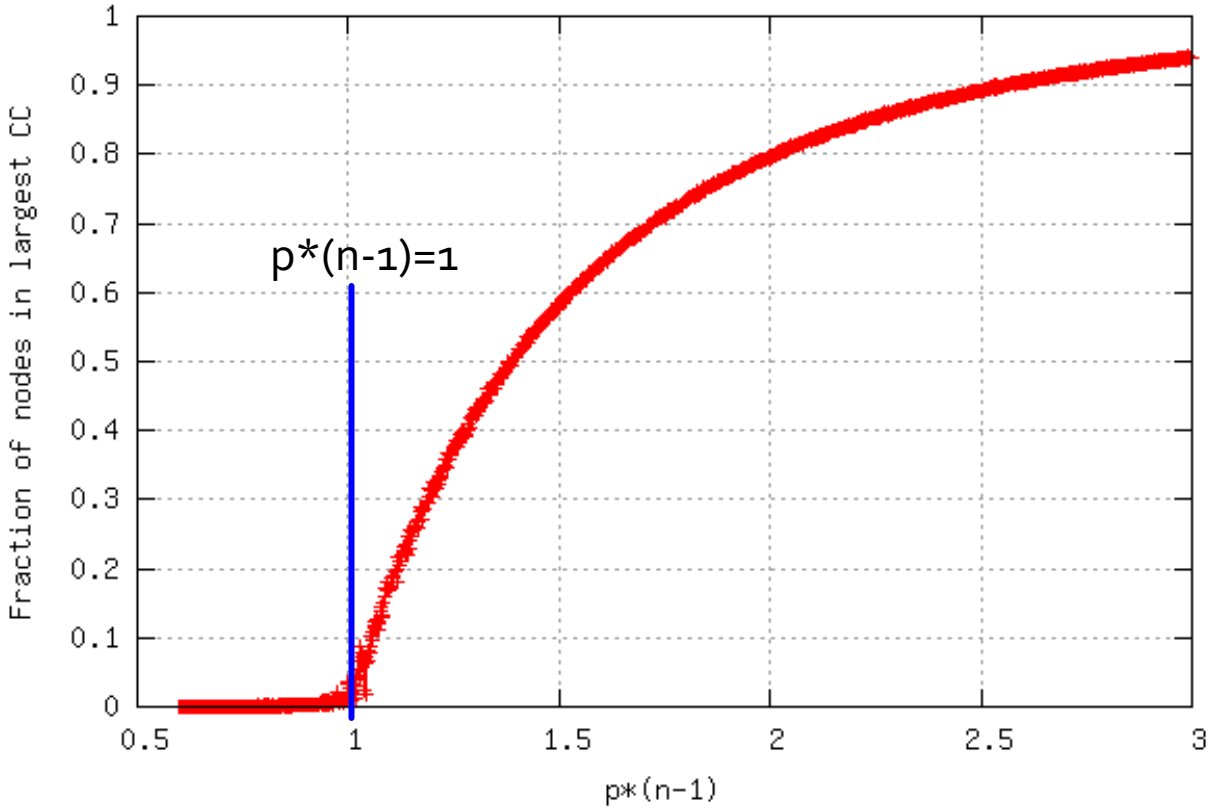
"Evolution" of a Random Graph

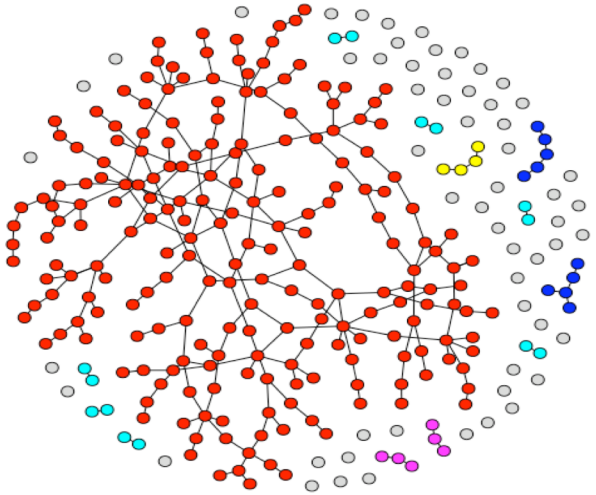
- **Graph structure of G_{np} as p changes:**



- **Emergence of a giant component:**
  avg. degree $k=2E/n$ or $p=k/(n-1)$
  - $k=1-\varepsilon$: all components are of size $\Omega(\log n)$
  - $k=1+\varepsilon$: 1 component of size $\Omega(n)$, others have size $\Omega(\log n)$
    - Each node has at least one edge in expectation

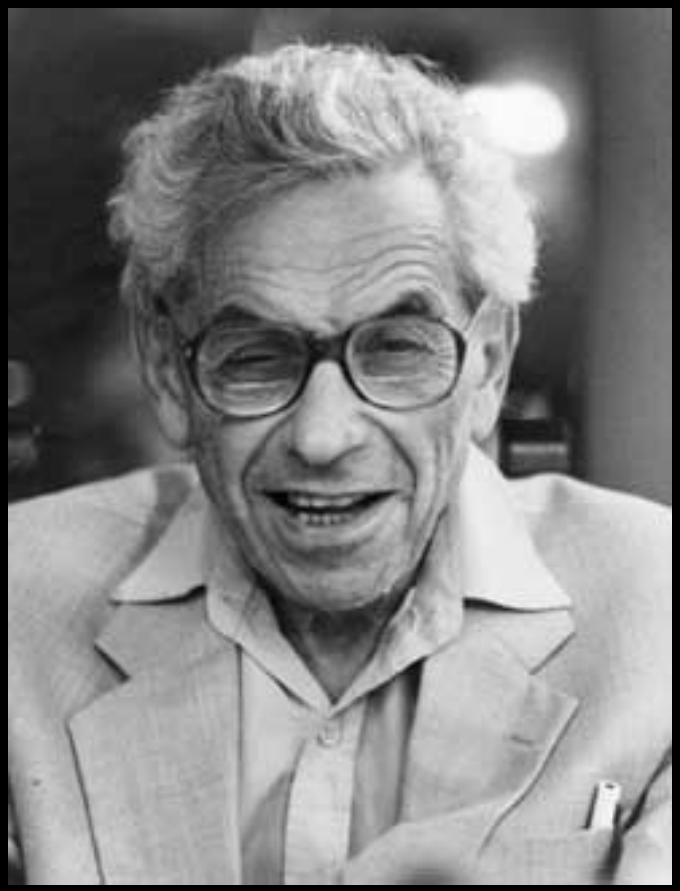
G_{np} Simulation Experiment





Fraction of nodes in the largest component

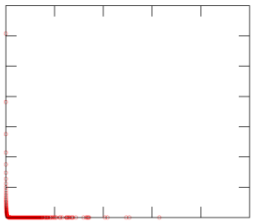
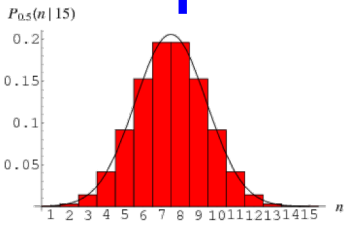
- G_{np}, n=100,000, $k=p(n-1)$ = 0.5 … 3



Paul Erdös

G_{np} is so cool!
Let's compare it to real networks.

Back to MSN vs. G_{np}

	MSN	G_{np} n=180M	
Degree distribution:			☒
Avg. path length:	**6.6**	$O(\log n)$ *h ≈ 8.2*	☑
Avg. clustering coef.:	*0.11*	$\bar{k}/n$ *C ≈ 8·10⁻⁸*	☒
Largest Conn. Comp.:	*99%*	GCC exists when $\bar{k}>1$. $\bar{k} \approx 14$.	☑

Real Networks vs. G_{np}

- **Are real networks like random graphs?**
  - Giant connected component: ☺
  - Average path length: ☺
  - Clustering Coefficient: ☹
  - Degree Distribution: ☹
- **Problems with the random networks model:**
  - Degree distribution differs from that of real networks
  - Giant component in most real network does NOT emerge through a phase transition
  - No local structure – clustering coefficient is too low
- **Most important: Are real networks random?**
  - The answer is simply: **NO!**

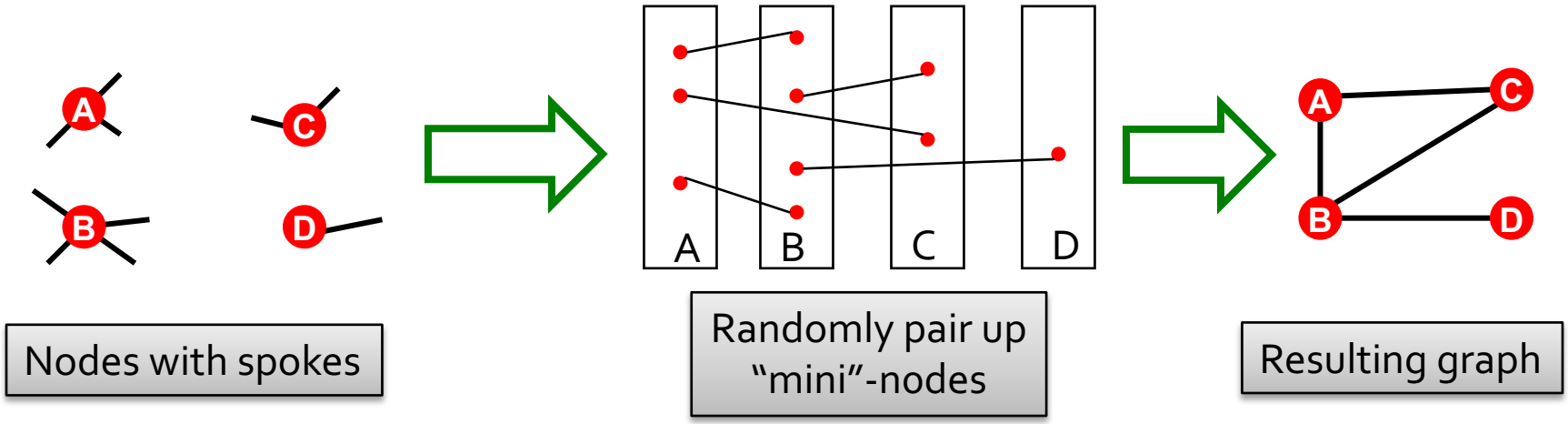
Real Networks vs. G_{np}

- **If G_{np} is wrong, why did we spend time on it?**
  - It is the reference model for the rest of the class
  - It will help us calculate many quantities, that can then be compared to the real data
  - It will help us understand to what degree is a particular property the result of some random process

So, while G_{np} is WRONG, it will turn out to be extremly USEFUL!

Intermezzo: Configuration Model

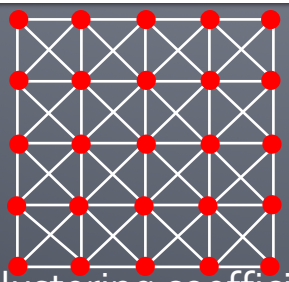
- **Goal:** Generate a random graph with a given degree sequence $k_1, k_2, \ldots k_N$
- **Configuration model:**



- **Useful as a "null" model of networks:**
  - We can compare the real network G and a "random" G' which has the same degree sequence as G

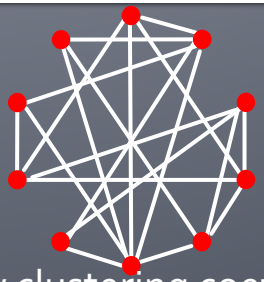
The Small-World Model

Can we have high clustering while also having short paths?



High clustering coefficient,
High diameter

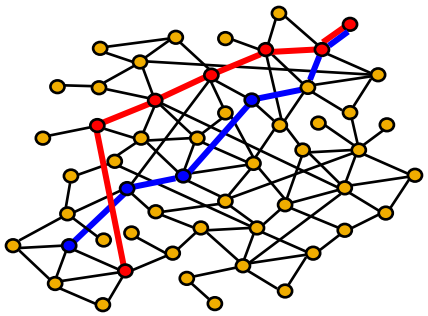
Vs.



Low clustering coefficient
Low diameter

The Small-World Experiment

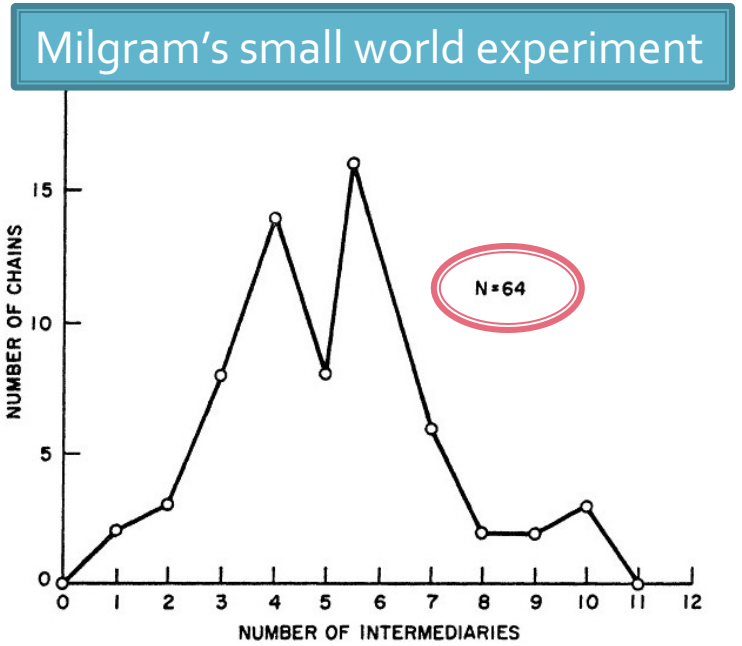
- **What is the typical shortest path length between any two people?**
  - **Experiment on the global friendship network**
    - Can't measure, need to probe explicitly
- **Small-world experiment** [Milgram '67]
  - Picked 300 people in Omaha, Nebraska and Wichita, Kansas
  - Ask them to get a letter to a stock-broker in Boston by passing it through friends
- **How many steps did it take?**



The Small-World Experiment

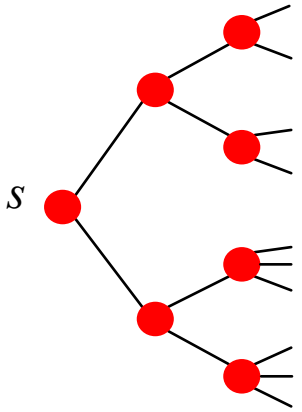
[Milgram, '67]

- **64 chains completed:**
  (i.e., 64 letters reached the target)
  - It took 6.2 steps on the average, thus
    "6 degrees of separation"
- **Further observations:**
  - People who owned stock had shorter paths to the stockbroker than random people: 5.4 vs. 6.7
  - People from the Boston area have even closer paths: 4.4

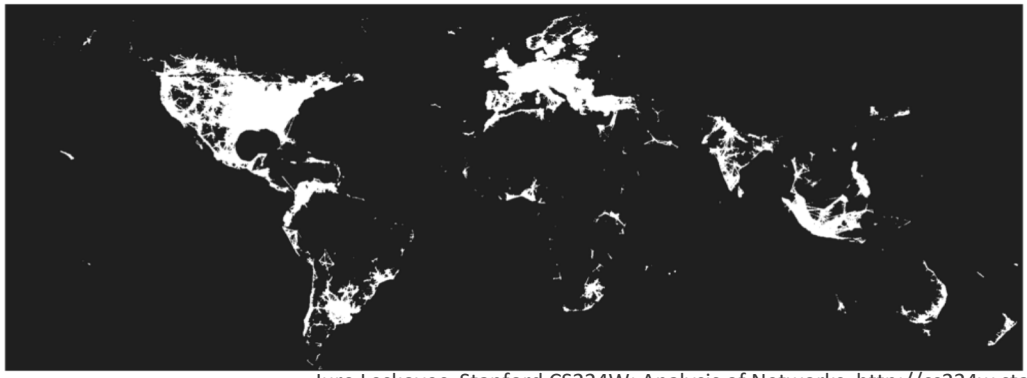


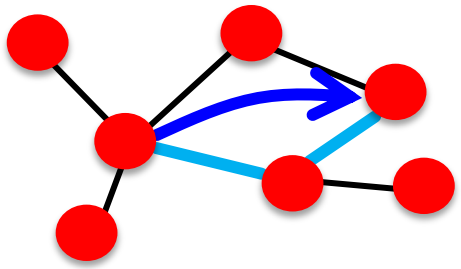
6-Degrees: Should We Be Surprised?

- **Assume each human is connected to 100 other people**
  Then:
  - Step 1: reach 100 people
  - Step 2: reach 100*100 = 10,000 people
  - Step 3: reach 100*100*100 = 1,000,000 people
  - Step 4: reach 100*100*100*100 = 100M people
  - **In 5 steps we can reach 10 billion people**



- **What's wrong here? We ignore clustering!**
  - **Not all edges point to new people**
    - **92% of FB friendships happen through a friend-of-a-friend**





Clustering Implies Edge Locality

- **MSN network has 7 orders of magnitude larger clustering than the corresponding G_{np}!**
- **Other examples:**

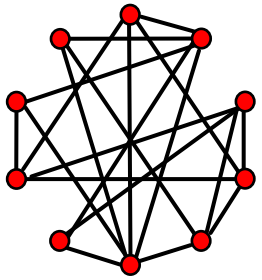
Actor Collaborations (IMDB): N = 225,226 nodes, avg. degree $\overline{k}$ = 61
Electrical power grid: N = 4,941 nodes, $\overline{k}$ = 2.67
Network of neurons: N = 282 nodes, $\overline{k}$ = 14

Network	h_{actual}	h_{random}	C_{actual}	C_{random}
Film actors	3.65	2.99	0.79	0.00027
Power Grid	18.70	12.40	0.080	0.005
C. elegans	2.65	2.25	0.28	0.05

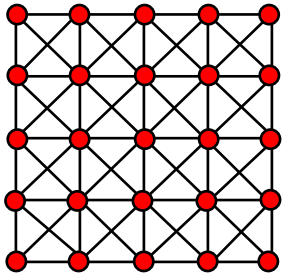
h ... Average shortest path length
C ... Average clustering coefficient
"actual" … real network
"random" … random graph with same avg. degree

The "Controversy"

- **Consequence of expansion:**
  - **Short paths: $O(\log n)$**
    - This is the smallest diameter we can get if we have a constant degree.
  - But clustering is low!
- **But networks have "local" structure:**
  - **Triadic closure:**
    Friend of a friend is my friend
  - High clustering but diameter is also high
- **How can we have both?**



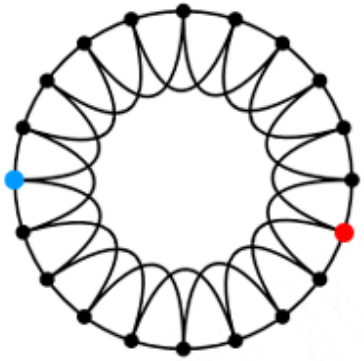
Low diameter
Low clustering coefficient



High clustering coefficient
High diameter

Small-World: How?

- **Could a network with high clustering also be a small world ($\log n$ dimeter)?**
  - How can we at the same time have **high clustering** and **small diameter?**



High clustering
High diameter

Low clustering
Low diameter

  - Clustering implies edge "locality"
  - Randomness enables "shortcuts"

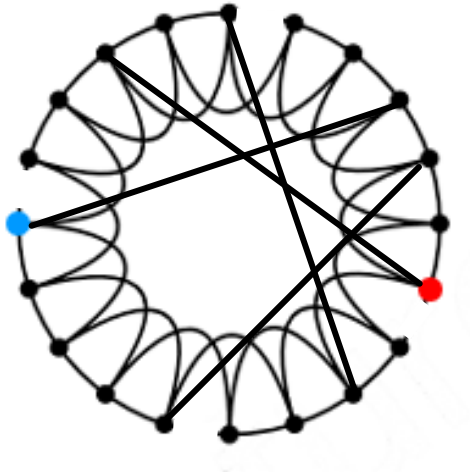
[Watts-Strogatz, '98]

Solution: The Small-World Model

Small-World Model [Watts-Strogatz '98]

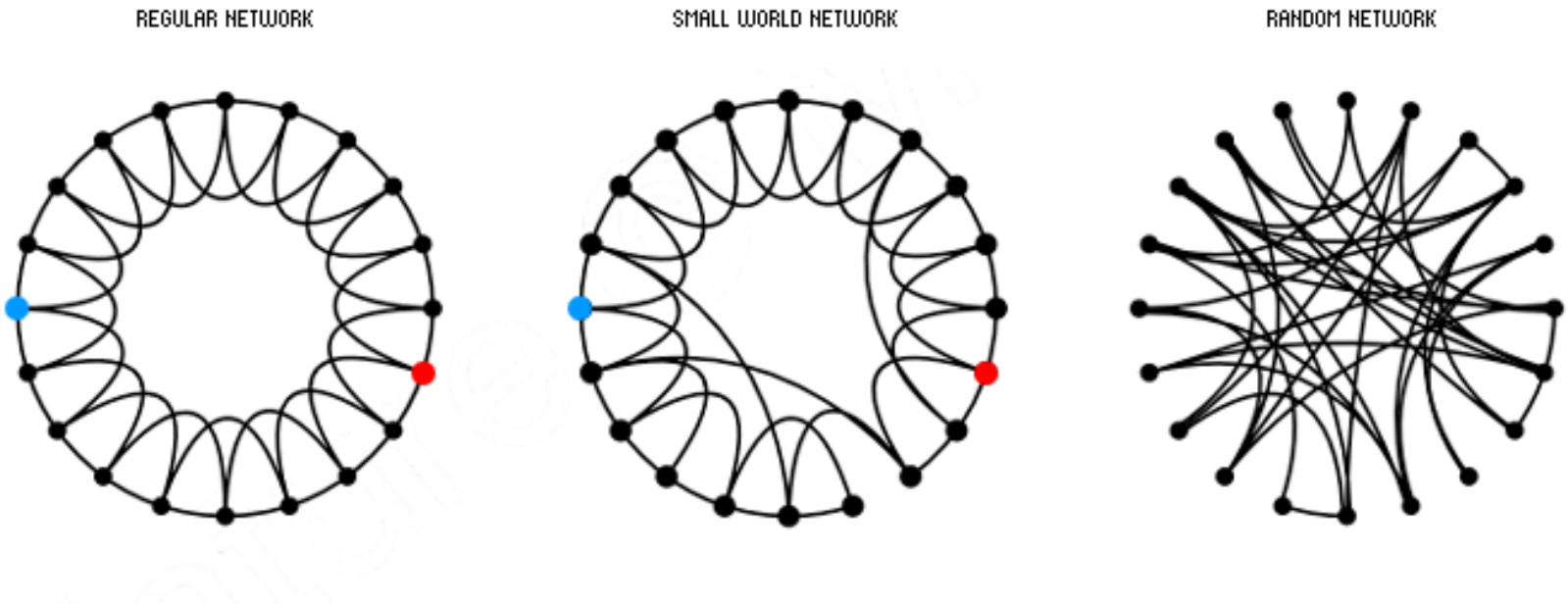
Two components to the model:

- **(1) Start with a low-dimensional regular lattice**
  - (In our case we are using a ring as a lattice)
  - Has high clustering coefficient
- **Now introduce randomness ("shortcuts")**
- **(2) Rewire:**
  - Add/remove edges to create shortcuts to join remote parts of the lattice
  - For each edge with prob. p move the other end to a random node



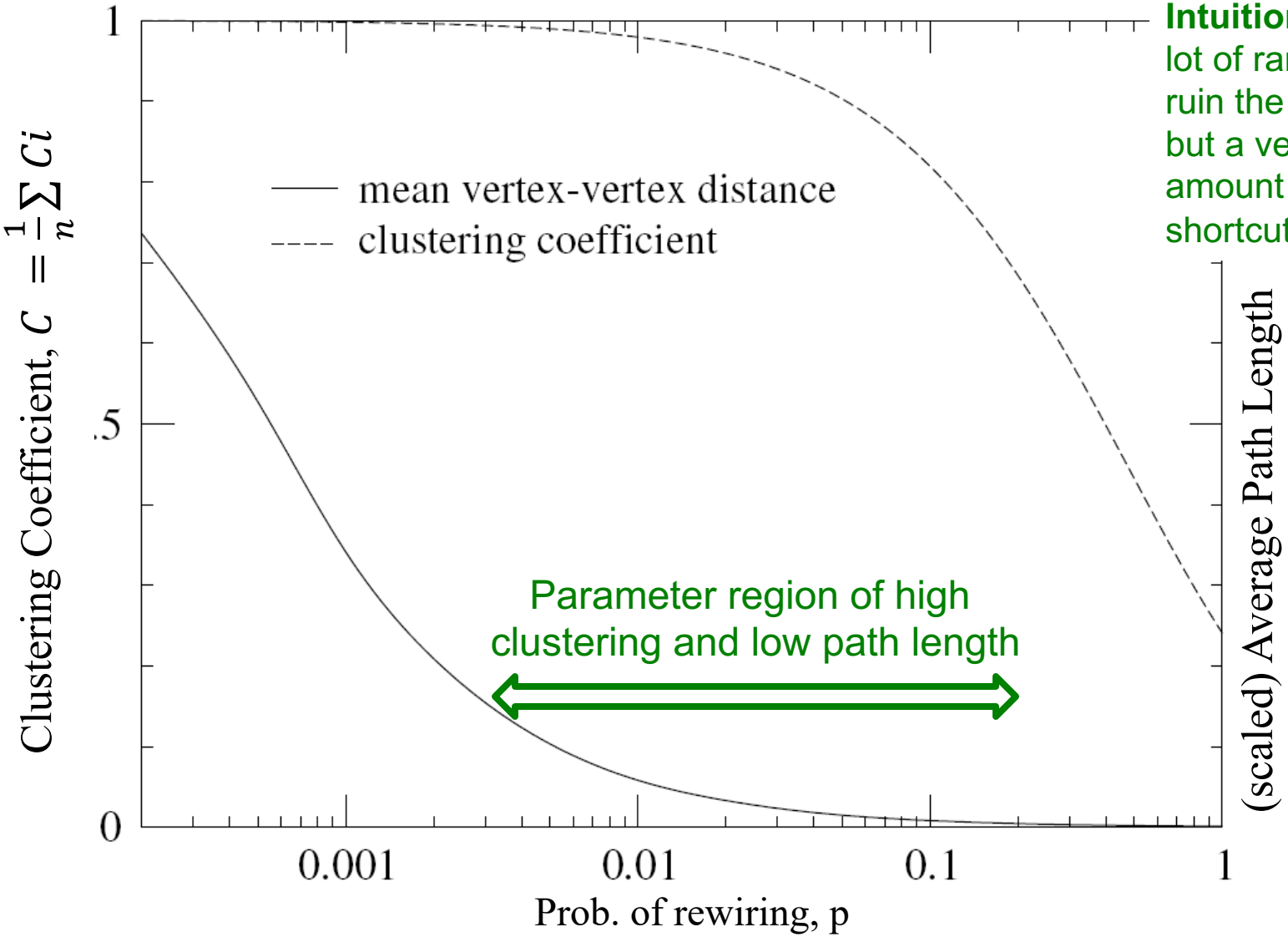
The Small-World Model

[Watts-Strogatz, '98]



Rewiring allows us to "interpolate" between a regular lattice and a random graph

The Small-World Model



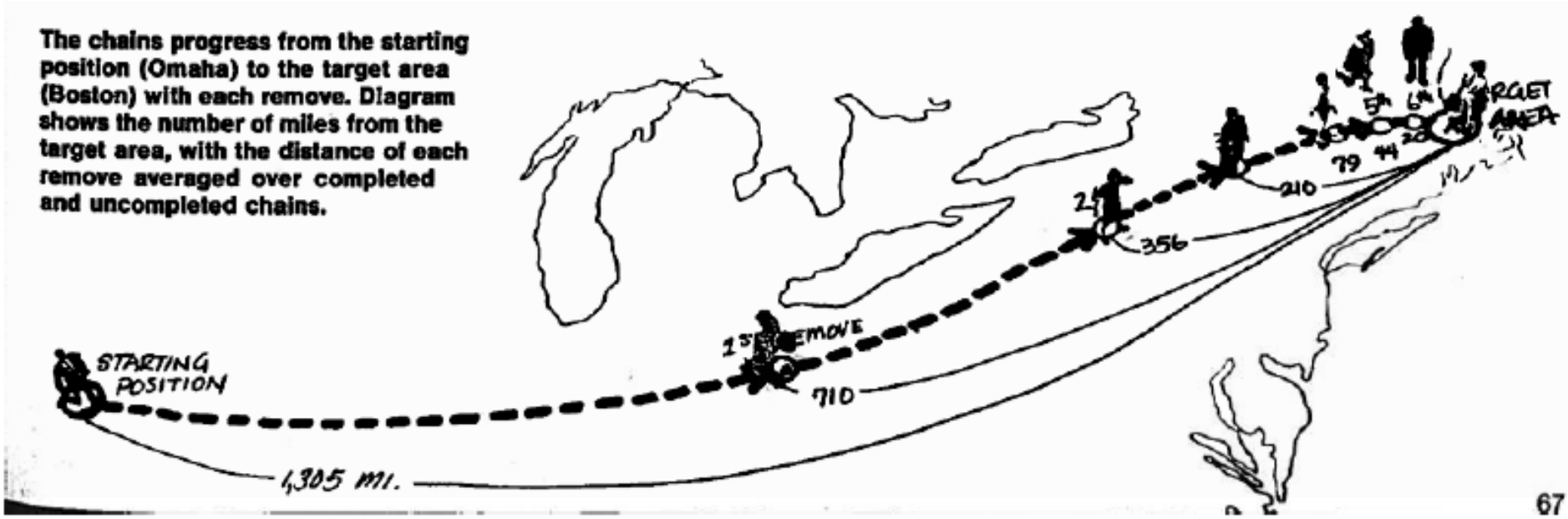
Intuition: It takes a lot of randomness to ruin the clustering, but a very small amount to create shortcuts.

Small-World: Summary

- **Could a network with high clustering be at the same time a small world?**
  - Yes! You don't need more than a few random links
- **The Watts Strogatz Model:**
  - Provides insight on the interplay between clustering and the small-world
  - Captures the structure of many realistic networks
  - Accounts for the high clustering of real networks
  - Does not lead to the correct degree distribution

How to Navigate a Network?

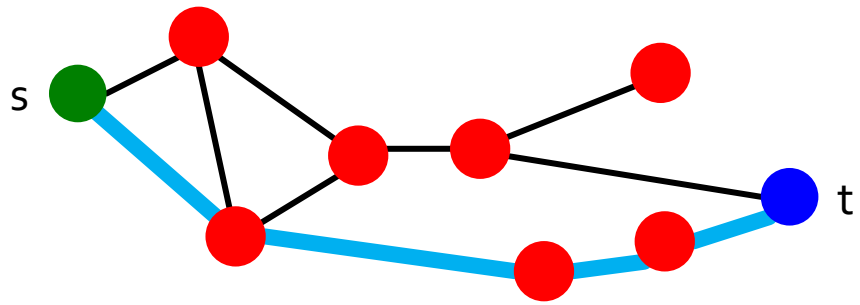
- **What mechanisms do people use to navigate networks and find the target?**



Decentralized Search

The setting:

- s only knows locations of its friends and location of the target t
- s does not know links of anyone else but itself
- **Geographic Navigation:** s "navigates" to a node geographically closest to t
- **Search time T:** Number of steps to reach t



Overview of the Results

Searchable

Search time T:

$$O((\log n)^{\beta})$$

Kleinberg's model

$$O((\log n)^{2})$$

Not searchable

Search time T:

$$O(n^{\alpha})$$

Watts-Strogatz

$$O(n^{\frac{2}{3}})$$

Erdős–Rényi

$$O(n)$$

Note: We know these graphs have diameter *O(log n)*.
So in Kleinberg's model search time is polynomial in *log n*, while in Watts-Strogatz it is exponential (in *log n*).

Navigable Small-World Graph?

- Watts-Strogatz graphs are **not searchable**
- **How do we make a searchable small-world graph?**
- **Intuition:**
  - Our long range links are not random
  - **They follow geography!**

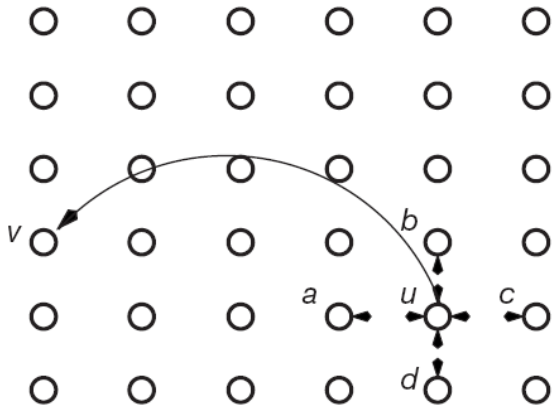
Saul Steinberg, "View of the World from 9th Avenue"

Variation of the Model

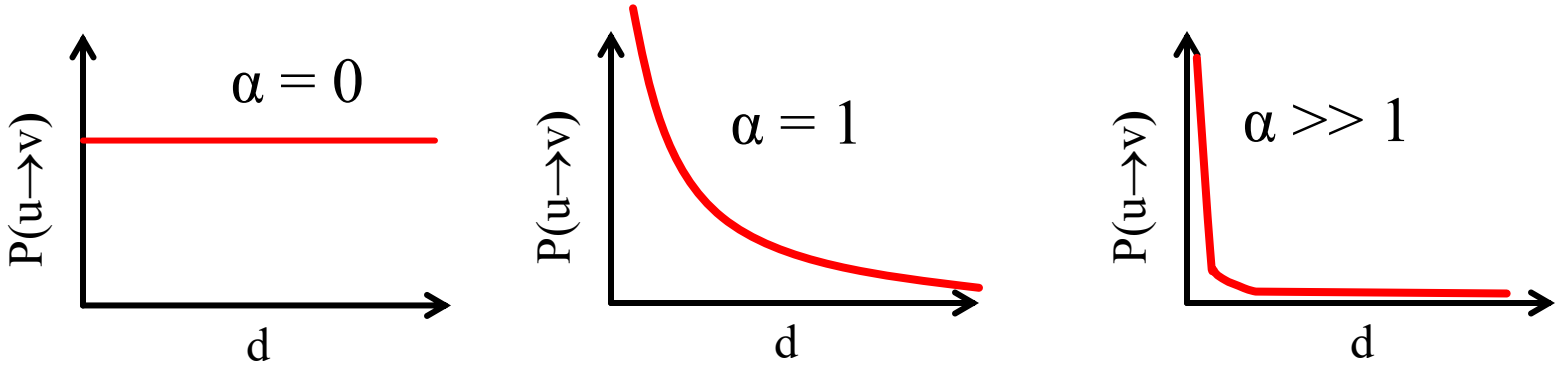
- **Model** [Kleinberg, Nature '01]
  - **Nodes still on a grid**
  - Node has one long range link
  - Prob. of long link to node v:

$$P(u \to v) \sim d(u,v)^{-\alpha}$$

    - $d(u,v)$ … grid distance between u and v
    - α … parameter ≥ 0

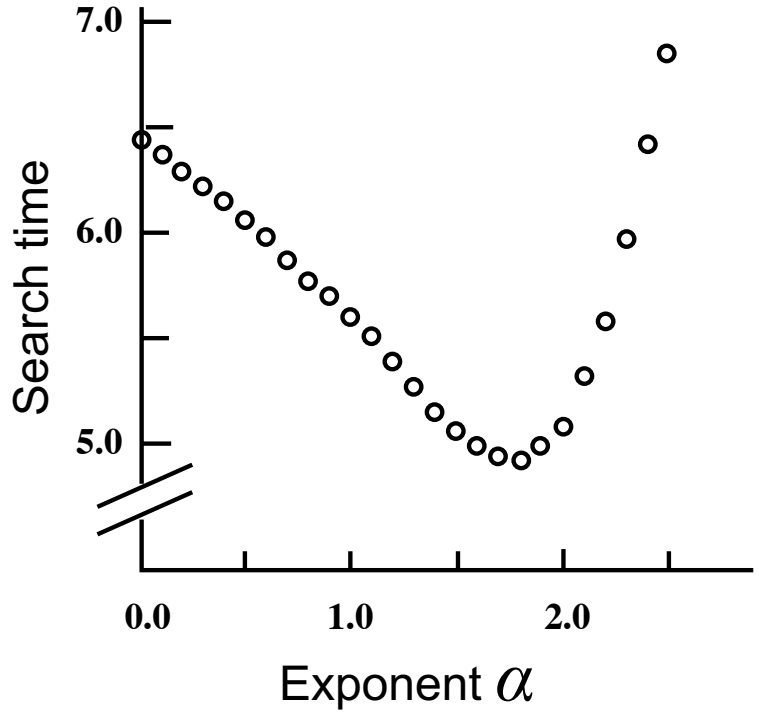


$$P(u \to v) = \frac{d(u,v)^{-\alpha}}{\sum_{w \neq u} d(u,w)^{-\alpha}}$$

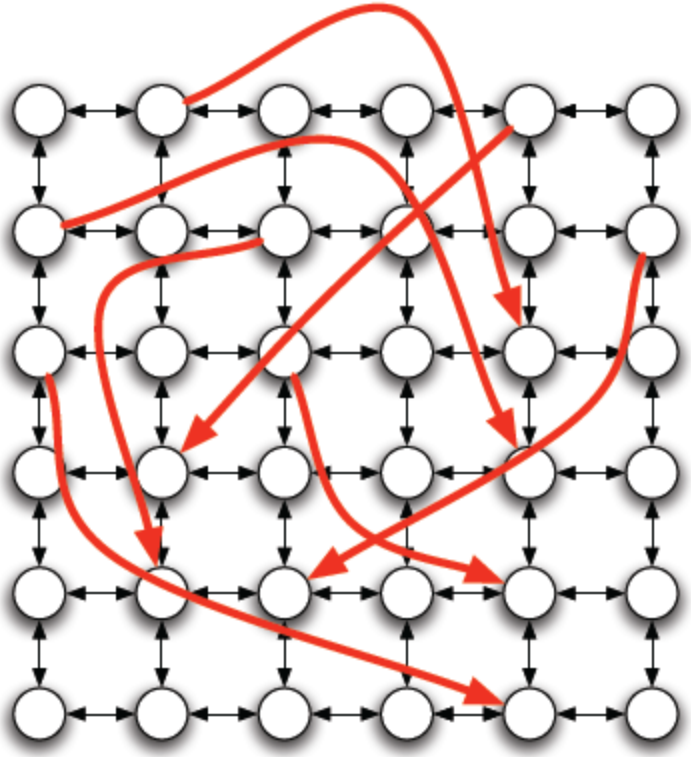


Kleinberg's Model: Search Time

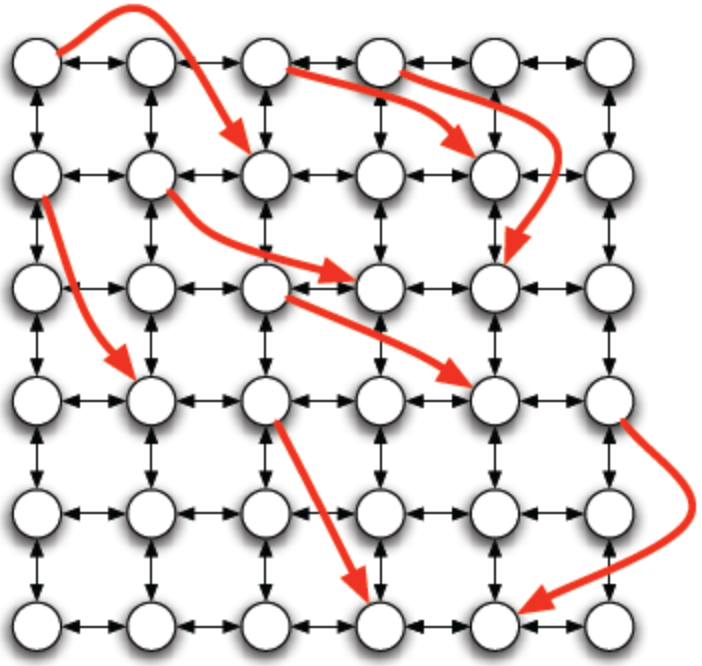
- **We know:**
  - $\alpha = 0$ (i.e., Watts-Strogatz): We need $O(\sqrt{n})$ steps
  - $\alpha = 1$: We need $O(log(n)^2)$ steps



Intuition: Why Search Takes Long



Small α: too many long links



Big α: too many short links