Graph Theory and Complex Networks: An Introduction

Maarten van Steen

VU Amsterdam, Dept. Computer Science
Room R4.20, steen@cs.vu.nl

Chapter 07: Random networks

Version: May 27, 2014

vrije Universiteit amsterdam

Introduction

Observation

Many real-world networks can be modeled as a random graph in which an edge $\langle u, v \rangle$ appears with probability p.

Spatial systems: Railway networks, airline networks, computer networks, have the property that the closer x and y are, the higher the probability that they are linked.

Food webs: Who eats whom? Turns out that techniques from random networks are useful for getting insight in their structure.

Collaboration networks: Who cites whom? Again, techniques from random networks allows us to understand what is going on.

Introduction

Observation

Many real-world networks can be modeled as a random graph in which an edge $\langle u, v \rangle$ appears with probability p.

Spatial systems: Railway networks, airline networks, computer networks, have the property that the closer x and y are, the higher the probability that they are linked.

Food webs: Who eats whom? Turns out that techniques from random networks are useful for getting insight in their structure.

Collaboration networks: Who cites whom? Again, techniques from random networks allows us to understand what is going on.

Introduction

Observation

Many real-world networks can be modeled as a random graph in which an edge $\langle u, v \rangle$ appears with probability p.

Spatial systems: Railway networks, airline networks, computer networks, have the property that the closer x and y are, the higher the probability that they are linked.

Food webs: Who eats whom? Turns out that techniques from random networks are useful for getting insight in their structure.

Collaboration networks: Who cites whom? Again, techniques from random networks allows us to understand what is going on.

Introduction

Observation

Many real-world networks can be modeled as a random graph in which an edge $\langle u, v \rangle$ appears with probability p.

Spatial systems: Railway networks, airline networks, computer networks, have the property that the closer x and y are, the higher the probability that they are linked.

Food webs: Who eats whom? Turns out that techniques from random networks are useful for getting insight in their structure.

Collaboration networks: Who cites whom? Again, techniques from random networks allows us to understand what is going on.

Erdös-Rényi graphs

Erdös-Rényi model

An undirected graph $ER(n, p)$ with n vertices. Edge $\langle u, v \rangle$ ($u \neq v$) exists with probability p.

Note

There is also an alternative definition, which we'll skip.

ER-graphs

Notation

$\mathbb{P}[\delta(u) = k]$ is probability that degree of u is equal to k.

- There are maximally $n-1$ other vertices that can be adjacent to u.
- We can choose k other vertices, out of $n-1$, to join with u
  $\Rightarrow \binom{n-1}{k} = \frac{(n-1)!}{(n-1-k)!\cdot k!}$ possibilities.
- Probability of having exactly one specific set of k neighbors is:

$$p^k(1-p)^{n-1-k}$$

Conclusion

$$\mathbb{P}[\delta(u) = k] = \binom{n-1}{k} p^k (1-p)^{n-1-k}$$

ER-graphs

Notation

$\mathbb{P}[\delta(u) = k]$ is probability that degree of u is equal to k.

- There are maximally $n-1$ other vertices that can be adjacent to u.
- We can choose k other vertices, out of $n-1$, to join with u $\Rightarrow \binom{n-1}{k} = \frac{(n-1)!}{(n-1-k)!\cdot k!}$ possibilities.
- Probability of having exactly one specific set of k neighbors is:

$$p^k(1-p)^{n-1-k}$$

Conclusion

$$\mathbb{P}[\delta(u) = k] = \binom{n-1}{k} p^k (1-p)^{n-1-k}$$

ER-graphs

Notation

$\mathbb{P}[\delta(u) = k]$ is probability that degree of u is equal to k.

- There are maximally $n-1$ other vertices that can be adjacent to u.
- We can choose k other vertices, out of $n-1$, to join with u $\Rightarrow \binom{n-1}{k} = \frac{(n-1)!}{(n-1-k)! \cdot k!}$ possibilities.
- Probability of having exactly one specific set of k neighbors is:

$$p^k(1-p)^{n-1-k}$$

Conclusion

$$\mathbb{P}[\delta(u) = k] = \binom{n-1}{k} p^k(1-p)^{n-1-k}$$

ER-graphs

Notation

$\mathbb{P}[\delta(u) = k]$ is probability that degree of u is equal to k.

- There are maximally $n-1$ other vertices that can be adjacent to u.
- We can choose k other vertices, out of $n-1$, to join with u $\Rightarrow \binom{n-1}{k} = \frac{(n-1)!}{(n-1-k)! \cdot k!}$ possibilities.
- Probability of having exactly one specific set of k neighbors is:

$$p^k(1-p)^{n-1-k}$$

Conclusion

$$\mathbb{P}[\delta(u) = k] = \binom{n-1}{k} p^k(1-p)^{n-1-k}$$

ER-graphs

Notation

$\mathbb{P}[\delta(u) = k]$ is probability that degree of u is equal to k.

- There are maximally $n-1$ other vertices that can be adjacent to u.
- We can choose k other vertices, out of $n-1$, to join with u $\Rightarrow \binom{n-1}{k} = \frac{(n-1)!}{(n-1-k)! \cdot k!}$ possibilities.
- Probability of having exactly one specific set of k neighbors is:

$$p^k(1-p)^{n-1-k}$$

Conclusion

$$\mathbb{P}[\delta(u) = k] = \binom{n-1}{k} p^k (1-p)^{n-1-k}$$

ER-graphs: average vertex degree (the simple way)

Observations

- We know that $\sum_{v \in V(G)} \delta(v) = 2 \cdot |E(G)|$
- We also know that between each two vertices, there exists an edge with probability p.
- There are at most $\binom{n}{2}$ edges
- **Conclusion**: we can expect a total of $p \cdot \binom{n}{2}$ edges.

Conclusion

$$\overline{\delta}(v) = \frac{1}{n} \sum \delta(v) = \frac{1}{n} \cdot 2 \cdot p \binom{n}{2} = \frac{2 \cdot p \cdot n \cdot (n-1)}{n \cdot 2} = p \cdot (n-1)$$

Even simpler

Each vertex can have maximally $n-1$ incident edges $\Rightarrow$ we can expect it to have $p(n-1)$ edges.

ER-graphs: average vertex degree (the simple way)

Observations

- We know that $\sum_{v \in V(G)} \delta(v) = 2 \cdot |E(G)|$
- We also know that between each two vertices, there exists an edge with probability p.
- There are at most $\binom{n}{2}$ edges
- **Conclusion**: we can expect a total of $p \cdot \binom{n}{2}$ edges.

Conclusion

$$\overline{\delta}(v) = \frac{1}{n}\sum \delta(v) = \frac{1}{n} \cdot 2 \cdot p\binom{n}{2} = \frac{2 \cdot p \cdot n \cdot (n-1)}{n \cdot 2} = p \cdot (n-1)$$

Even simpler

Each vertex can have maximally $n-1$ incident edges $\Rightarrow$ we can expect it to have $p(n-1)$ edges.

ER-graphs: average vertex degree (the simple way)

Observations

- We know that $\sum_{v \in V(G)} \delta(v) = 2 \cdot |E(G)|$
- We also know that between each two vertices, there exists an edge with probability p.
- There are at most $\binom{n}{2}$ edges
- **Conclusion**: we can expect a total of $p \cdot \binom{n}{2}$ edges.

Conclusion

$$\overline{\delta}(v) = \frac{1}{n} \sum \delta(v) = \frac{1}{n} \cdot 2 \cdot p \binom{n}{2} = \frac{2 \cdot p \cdot n \cdot (n-1)}{n \cdot 2} = p \cdot (n-1)$$

Even simpler

Each vertex can have maximally $n-1$ incident edges $\Rightarrow$ we can expect it to have $p(n-1)$ edges.

ER-graphs: average vertex degree (the simple way)

Observations

- We know that $\sum_{v \in V(G)} \delta(v) = 2 \cdot |E(G)|$
- We also know that between each two vertices, there exists an edge with probability p.
- There are at most $\binom{n}{2}$ edges
- **Conclusion**: we can expect a total of $p \cdot \binom{n}{2}$ edges.

Conclusion

$$\overline{\delta}(v) = \frac{1}{n} \sum \delta(v) = \frac{1}{n} \cdot 2 \cdot p \binom{n}{2} = \frac{2 \cdot p \cdot n \cdot (n-1)}{n \cdot 2} = p \cdot (n-1)$$

Even simpler

Each vertex can have maximally $n-1$ incident edges $\Rightarrow$ we can expect it to have $p(n-1)$ edges.

ER-graphs: average vertex degree (the hard way)

Observation

All vertices have the same probability of having degree k, meaning that we can treat the degree distribution as a stochastic variable δ. We now know that δ follows a binomial distribution.

Recall

Computing the average (or expected value) of a stochastic variable x, is computing:

$$\overline{x} \stackrel{\text{def}}{=} \mathbb{E}[x] \stackrel{\text{def}}{=} \sum_k k \cdot \mathbb{P}[x = k]$$

ER-graphs: average vertex degree (the hard way)

Observation

All vertices have the same probability of having degree k, meaning that we can treat the degree distribution as a stochastic variable δ. We now know that δ follows a binomial distribution.

Recall

Computing the average (or expected value) of a stochastic variable x, is computing:

$$\overline{x} \stackrel{\text{def}}{=} \mathbb{E}[x] \stackrel{\text{def}}{=} \sum_k k \cdot \mathbb{P}[x = k]$$

ER-graphs: average vertex degree (the hard way)

$$\sum_{k=1}^{n-1} k \cdot \mathbb{P}[\delta = k] =$$

$$=$$

$$=$$

$$=$$

$$=$$

$$=$$

ER-graphs: average vertex degree (the hard way)

$$\begin{aligned}
\sum_{k=1}^{n-1} k \cdot \mathbb{P}[\delta = k] &= \sum_{k=1}^{n-1} \binom{n-1}{k}\, k\, p^k\, (1-p)^{n-1-k} \\
&= \\
&= \\
&= \\
&= \\
&=
\end{aligned}$$

ER-graphs: average vertex degree (the hard way)

$$\begin{aligned}
\sum_{k=1}^{n-1} k \cdot \mathbb{P}[\delta = k] &= \sum_{k=1}^{n-1} \binom{n-1}{k}\, k\, p^k\, (1-p)^{n-1-k} \\
&= \sum_{k=1}^{n-1} \binom{n-1}{k}\, k\, p^k\, (1-p)^{n-1-k} \\
&= \\
&= \\
&= \\
&=
\end{aligned}$$

ER-graphs: average vertex degree (the hard way)

$$
\begin{aligned}
\sum_{k=1}^{n-1} k \cdot \mathbb{P}[\delta = k] &= \sum_{k=1}^{n-1} \binom{n-1}{k} \, k \, p^k \, (1-p)^{n-1-k} \\
&= \sum_{k=1}^{n-1} \binom{n-1}{k} \, k \, p^k \, (1-p)^{n-1-k} \\
&= \sum_{k=1}^{n-1} \frac{(n-1)!}{k!(n-1-k)!} \, k \, p^k \, (1-p)^{n-1-k} \\
&= \\
&= \\
&=
\end{aligned}
$$

ER-graphs: average vertex degree (the hard way)

$$
\begin{aligned}
\sum_{k=1}^{n-1} k \cdot \mathbb{P}[\delta = k] &= \sum_{k=1}^{n-1} \binom{n-1}{k} \, k \, p^k \, (1-p)^{n-1-k} \\
&= \sum_{k=1}^{n-1} \binom{n-1}{k} \, k \, p^k \, (1-p)^{n-1-k} \\
&= \sum_{k=1}^{n-1} \frac{(n-1)!}{k!(n-1-k)!} \, k \, p^k \, (1-p)^{n-1-k} \\
&= \sum_{k=1}^{n-1} \frac{(n-1)(n-2)!}{k(k-1)!(n-1-k)!} \, k \, p \cdot p^{k-1} \, (1-p)^{n-1-k} \\
&= \\
&=
\end{aligned}
$$

ER-graphs: average vertex degree (the hard way)

$$
\begin{aligned}
\sum_{k=1}^{n-1} k \cdot \mathbb{P}[\delta = k] &= \sum_{k=1}^{n-1} \binom{n-1}{k} \, k \, p^k \, (1-p)^{n-1-k} \\
&= \sum_{k=1}^{n-1} \binom{n-1}{k} \, k \, p^k \, (1-p)^{n-1-k} \\
&= \sum_{k=1}^{n-1} \frac{(n-1)!}{k!(n-1-k)!} \, k \, p^k \, (1-p)^{n-1-k} \\
&= \sum_{k=1}^{n-1} \frac{(n-1)(n-2)!}{k(k-1)!(n-1-k)!} \, k \, p \cdot p^{k-1} \, (1-p)^{n-1-k} \\
&= \sum_{k=1}^{n-1} \frac{(n-1)(n-2)!}{k(k-1)!(n-1-k)!} \, k \, p \cdot p^{k-1} \, (1-p)^{n-1-k} \\
&=
\end{aligned}
$$

ER-graphs: average vertex degree (the hard way)

$$
\begin{aligned}
\sum_{k=1}^{n-1} k \cdot \mathbb{P}[\delta = k] &= \sum_{k=1}^{n-1} \binom{n-1}{k} k\, p^k\, (1-p)^{n-1-k} \\
&= \sum_{k=1}^{n-1} \binom{n-1}{k} k\, p^k\, (1-p)^{n-1-k} \\
&= \sum_{k=1}^{n-1} \frac{(n-1)!}{k!(n-1-k)!} k\, p^k\, (1-p)^{n-1-k} \\
&= \sum_{k=1}^{n-1} \frac{(n-1)(n-2)!}{k(k-1)!(n-1-k)!} k\, p \cdot p^{k-1}\, (1-p)^{n-1-k} \\
&= \sum_{k=1}^{n-1} \frac{(n-1)(n-2)!}{k(k-1)!(n-1-k)!} k\, p \cdot p^{k-1}\, (1-p)^{n-1-k} \\
&= p\,(n-1) \sum_{k=1}^{n-1} \frac{(n-2)!}{(k-1)!(n-1-k)!}\, p^{k-1}\, (1-p)^{n-1-k}
\end{aligned}
$$

ER-graphs: average vertex degree (the hard way)

$$\begin{aligned}
\sum_{k=1}^{n-1} k \cdot \mathbb{P}[\delta = k] &= p\,(n-1) \sum_{k=1}^{n-1} \frac{(n-2)!}{(k-1)!(n-1-k)!}\, p^{k-1}\,(1-p)^{n-1-k} \\
&= \\
&= \\
&= \\
&= \\
&=
\end{aligned}$$

ER-graphs: average vertex degree (the hard way)

$$
\begin{aligned}
\sum_{k=1}^{n-1} k \cdot \mathbb{P}[\delta = k] &= p\,(n-1) \sum_{k=1}^{n-1} \frac{(n-2)!}{(k-1)!(n-1-k)!}\, p^{k-1}\,(1-p)^{n-1-k} \\
\{\text{Take } l \equiv k-1\} &= p\,(n-1) \sum_{l=0}^{n-2} \frac{(n-2)!}{l!(n-1-(l+1))!}\, p^{l}\,(1-p)^{n-1-(l+1)} \\
&= \\
&= \\
&= \\
&=
\end{aligned}
$$

ER-graphs: average vertex degree (the hard way)

$$
\begin{aligned}
\sum_{k=1}^{n-1} k \cdot \mathbb{P}[\delta = k] &= p\,(n-1) \sum_{k=1}^{n-1} \frac{(n-2)!}{(k-1)!(n-1-k)!}\, p^{k-1}\,(1-p)^{n-1-k} \\
\{\text{Take } l \equiv k-1\} &= p\,(n-1) \sum_{l=0}^{n-2} \frac{(n-2)!}{l!(n-1-(l+1))!}\, p^{l}\,(1-p)^{n-1-(l+1)} \\
&= p\,(n-1) \sum_{l=0}^{n-2} \frac{(n-2)!}{l!(n-2-l)!}\, p^{l}\,(1-p)^{n-2-l} \\
&= \\
&= \\
&=
\end{aligned}
$$

ER-graphs: average vertex degree (the hard way)

$$
\begin{aligned}
\sum_{k=1}^{n-1} k \cdot \mathbb{P}[\delta = k] &= p\,(n-1) \sum_{k=1}^{n-1} \frac{(n-2)!}{(k-1)!(n-1-k)!}\, p^{k-1}\,(1-p)^{n-1-k} \\
\{\text{Take } l \equiv k-1\} &= p\,(n-1) \sum_{l=0}^{n-2} \frac{(n-2)!}{l!(n-1-(l+1))!}\, p^{l}\,(1-p)^{n-1-(l+1)} \\
&= p\,(n-1) \sum_{l=0}^{n-2} \frac{(n-2)!}{l!(n-2-l)!}\, p^{l}\,(1-p)^{n-2-l} \\
&= p\,(n-1) \sum_{l=0}^{n-2} \binom{n-2}{l}\, p^{l}\,(1-p)^{n-2-l} \\
&= \\
&=
\end{aligned}
$$

ER-graphs: average vertex degree (the hard way)

$$
\begin{aligned}
\sum_{k=1}^{n-1} k \cdot \mathbb{P}[\delta = k] &= p\,(n-1) \sum_{k=1}^{n-1} \frac{(n-2)!}{(k-1)!(n-1-k)!}\, p^{k-1}\,(1-p)^{n-1-k} \\
\{\text{Take } l \equiv k-1\} &= p\,(n-1) \sum_{l=0}^{n-2} \frac{(n-2)!}{l!(n-1-(l+1))!}\, p^{l}\,(1-p)^{n-1-(l+1)} \\
&= p\,(n-1) \sum_{l=0}^{n-2} \frac{(n-2)!}{l!(n-2-l)!}\, p^{l}\,(1-p)^{n-2-l} \\
&= p\,(n-1) \sum_{l=0}^{n-2} \binom{n-2}{l}\, p^{l}\,(1-p)^{n-2-l} \\
\{\text{Take } m \equiv n-2\} &= p\,(n-1) \sum_{l=0}^{m} \binom{m}{l}\, p^{l}\,(1-p)^{m-l} \\
&=
\end{aligned}
$$

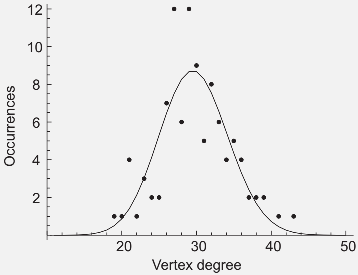
ER-graphs: average vertex degree (the hard way)

$$
\begin{aligned}
\sum_{k=1}^{n-1} k \cdot \mathbb{P}[\delta = k] &= p\,(n-1) \sum_{k=1}^{n-1} \frac{(n-2)!}{(k-1)!(n-1-k)!}\, p^{k-1}\,(1-p)^{n-1-k} \\
\{\text{Take } l \equiv k-1\} &= p\,(n-1) \sum_{l=0}^{n-2} \frac{(n-2)!}{l!(n-1-(l+1))!}\, p^{l}\,(1-p)^{n-1-(l+1)} \\
&= p\,(n-1) \sum_{l=0}^{n-2} \frac{(n-2)!}{l!(n-2-l)!}\, p^{l}\,(1-p)^{n-2-l} \\
&= p\,(n-1) \sum_{l=0}^{n-2} \binom{n-2}{l}\, p^{l}\,(1-p)^{n-2-l} \\
\{\text{Take } m \equiv n-2\} &= p\,(n-1) \sum_{l=0}^{m} \binom{m}{l}\, p^{l}\,(1-p)^{m-l} \\
&= p\,(n-1) \cdot 1
\end{aligned}
$$

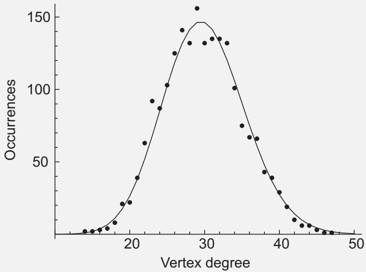
Examples of ER-graphs

Important

$ER(n, p)$ represents a group of Erdös-Rényi graphs: most $ER(n, p)$ graphs are not isomorphic!



$G \in ER(100, 0.3)$



$G^* \in ER(2000, 0.015)$

Examples of ER-graphs

Some observations

- $G \in ER(100, 0.3) \Rightarrow$
  - $\overline{\delta} = 0.3 \times 99 = 29.7$
  - Expected $|E(G)| =$ $\frac{1}{2} \cdot \sum \delta(v) = np(n-1)/2 = \frac{1}{2} \times 100 \times 0.3 \times 99 = 1485.$
  - In our example: 490 edges.
- $G^* \in ER(2000, 0.015) \Rightarrow$
  - $\overline{\delta} = 0.015 \times 1999 = 29.985$
  - Expected $|E(G)| =$ $\frac{1}{2} \sum \delta(v) = np(n-1)/2 = \frac{1}{2} \times 2000 \times 0.015 \times 1999 = 29,985.$
  - In our example: 29,708 edges.
- The larger the graph, the more probable its degree distribution will follow the expected one (**Note**: not easy to show!)

Examples of ER-graphs

Some observations

- $G \in ER(100, 0.3) \Rightarrow$
  - $\overline{\delta} = 0.3 \times 99 = 29.7$
  - Expected $|E(G)|$ = $\frac{1}{2} \cdot \sum \delta(v) = np(n-1)/2 = \frac{1}{2} \times 100 \times 0.3 \times 99 = 1485.$
  - In our example: 490 edges.
- $G^* \in ER(2000, 0.015) \Rightarrow$
  - $\overline{\delta} = 0.015 \times 1999 = 29.985$
  - Expected $|E(G)|$ = $\frac{1}{2} \sum \delta(v) = np(n-1)/2 = \frac{1}{2} \times 2000 \times 0.015 \times 1999 = 29,985.$
  - In our example: 29,708 edges.
- The larger the graph, the more probable its degree distribution will follow the expected one (**Note**: not easy to show!)

Examples of ER-graphs

Some observations

- $G \in ER(100, 0.3) \Rightarrow$
  - $\overline{\delta} = 0.3 \times 99 = 29.7$
  - Expected $|E(G)| =$ $\frac{1}{2} \cdot \sum \delta(v) = np(n-1)/2 = \frac{1}{2} \times 100 \times 0.3 \times 99 = 1485.$
  - In our example: 490 edges.
- $G^* \in ER(2000, 0.015) \Rightarrow$
  - $\overline{\delta} = 0.015 \times 1999 = 29.985$
  - Expected $|E(G)| =$ $\frac{1}{2} \sum \delta(v) = np(n-1)/2 = \frac{1}{2} \times 2000 \times 0.015 \times 1999 = 29,985.$
  - In our example: 29,708 edges.
- The larger the graph, the more probable its degree distribution will follow the expected one (**Note**: not easy to show!)

Examples of ER-graphs

Some observations

- $G \in ER(100, 0.3) \Rightarrow$
  - $\overline{\delta} = 0.3 \times 99 = 29.7$
  - Expected $|E(G)|$ = $\frac{1}{2} \cdot \sum \delta(v) = np(n-1)/2 = \frac{1}{2} \times 100 \times 0.3 \times 99 = 1485.$
  - In our example: 490 edges.
- $G^* \in ER(2000, 0.015) \Rightarrow$
  - $\overline{\delta} = 0.015 \times 1999 = 29.985$
  - Expected $|E(G)|$ = $\frac{1}{2} \sum \delta(v) = np(n-1)/2 = \frac{1}{2} \times 2000 \times 0.015 \times 1999 = 29{,}985.$
  - In our example: 29,708 edges.
- The larger the graph, the more probable its degree distribution will follow the expected one (**Note**: not easy to show!)

ER-graphs: average path length

Observation

For any large $H \in ER(n,p)$ it can be shown that the average path length $\overline{d}(H)$ is equal to:

$$\overline{d}(H) = \frac{\ln(n) - \gamma}{\ln(pn)} + 0.5$$

with γ the Euler constant (≈ 0.5772).

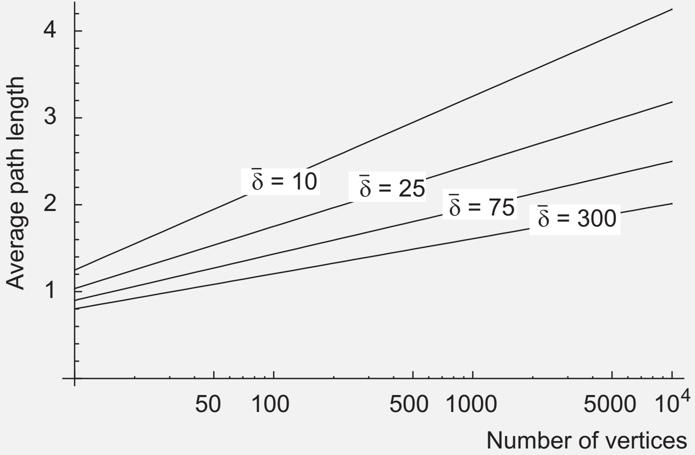
Observation

With $\overline{\delta} = p(n-1)$, we have

$$\overline{d}(H) \approx \frac{\ln(n) - \gamma}{\ln(\overline{\delta})} + 0.5$$

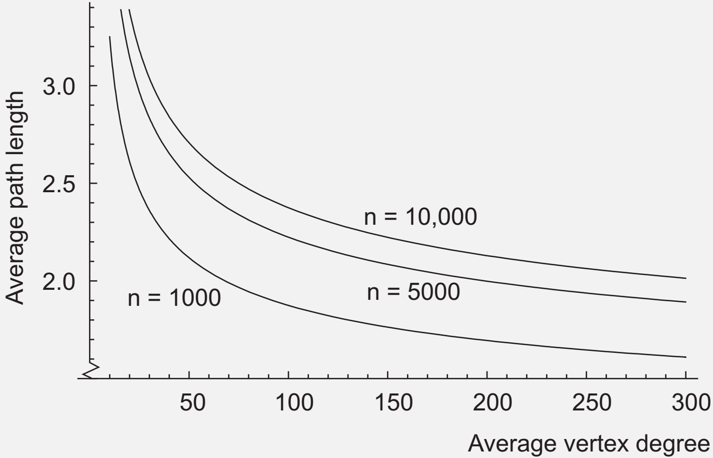
ER-graphs: average path length

Example: Keep average vertex degree fixed, but change size of graphs:



ER-graphs: average path length

Example: Keep size fixed, but change average vertex degree:



ER-graphs: clustering coefficient

Reasoning

- Clustering coefficient: fraction of edges between neighbors and maximum possible edges.
- Expected number of edges between k neighbors: $\binom{k}{2}p$
- Maximum number of edges between k neighbors: $\binom{k}{2}$
- Expected clustering coefficient for every vertex: p

ER-graphs: clustering coefficient

Reasoning

- Clustering coefficient: fraction of edges between neighbors and maximum possible edges.
- Expected number of edges between k neighbors: $\binom{k}{2}p$
- Maximum number of edges between k neighbors: $\binom{k}{2}$
- Expected clustering coefficient for every vertex: p

ER-graphs: clustering coefficient

Reasoning

- Clustering coefficient: fraction of edges between neighbors and maximum possible edges.
- Expected number of edges between k neighbors: $\binom{k}{2}p$
- Maximum number of edges between k neighbors: $\binom{k}{2}$
- Expected clustering coefficient for every vertex: p

ER-graphs: clustering coefficient

Reasoning

- Clustering coefficient: fraction of edges between neighbors and maximum possible edges.
- Expected number of edges between k neighbors: $\binom{k}{2}p$
- Maximum number of edges between k neighbors: $\binom{k}{2}$
- Expected clustering coefficient for every vertex: p

ER-graphs: connectivity

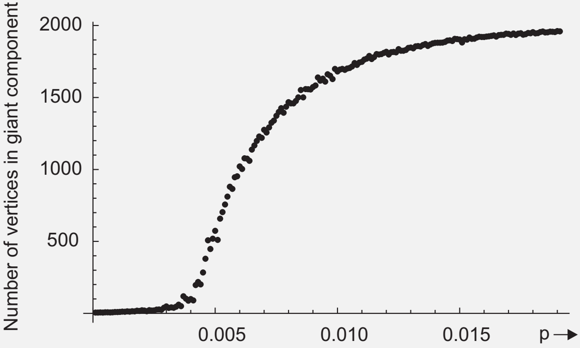
Giant component

Observation: When increasing p, most vertices are contained in the same component.

ER-graphs: connectivity

Giant component

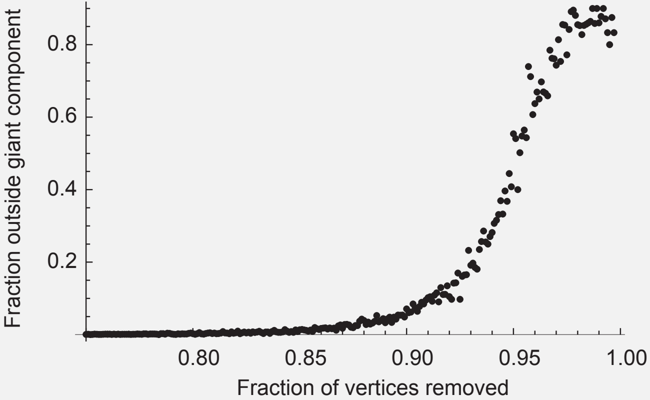
Observation: When increasing *p*, most vertices are contained in the same component.



ER-graphs: connectivity

Robustness

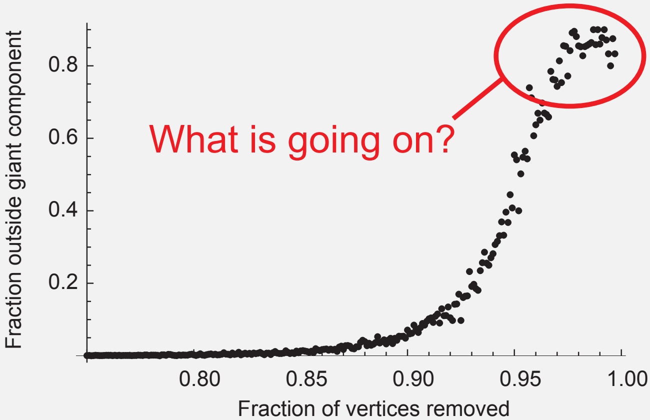
Experiment: How many vertices do we need to remove to partition an ER-graph? Let $G \in ER(2000, 0.015)$.



ER-graphs: connectivity

Robustness

Experiment: How many vertices do we need to remove to partition an ER-graph? Let $G \in ER(2000, 0.015)$.



Small worlds: Six degrees of separation

Stanley Milgram

- Pick two people at random
- Try to measure their distance: *A* knows *B* knows *C* ...
- Experiment: Let Alice try to get a letter to Zach, whom she does not know.
- Strategy by Alice: choose Bob who she thinks has a better chance of reaching Zach.
- Result: On average 5.5 hops before letter reaches target.

Small-world networks

General observation

Many real-world networks show a small average shortest path length.

Observation

ER-graphs have a small average shortest path length, but not the high clustering coefficient that we observe in real-world networks.

Question

Can we construct **more realistic models** of real-world networks?

Small-world networks

General observation

Many real-world networks show a small average shortest path length.

Observation

ER-graphs have a small average shortest path length, but not the high clustering coefficient that we observe in real-world networks.

Question

Can we construct **more realistic models** of real-world networks?

Small-world networks

General observation

Many real-world networks show a small average shortest path length.

Observation

ER-graphs have a small average shortest path length, but not the high clustering coefficient that we observe in real-world networks.

Question

Can we construct **more realistic models** of real-world networks?

Watts-Strogatz graphs

Algorithm (Watts-Strogatz)

$V = \{v_1, v_2, \ldots, v_n\}$. *Let* k *be even. Choose* $n \gg k \gg \ln(n) \gg 1$.

1. *Order the n vertices into a ring*
2. *Connect each vertex to its first* $k/2$ *right-hand (counterclockwise) neighbors, and to its* $k/2$ *left-hand (clockwise) neighbors.*
3. *With probability p, replace edge* $\langle u, v\rangle$ *with an edge* $\langle u, w\rangle$ *where* $w \neq u$ *is randomly chosen, but such that* $\langle u, w\rangle \notin E(G)$.
4. ***Notation**:* $WS(n, k, p)$ *graph*

Watts-Strogatz graphs

Algorithm (Watts-Strogatz)

$V = \{v_1, v_2, \ldots, v_n\}$. *Let* k *be even. Choose* $n \gg k \gg \ln(n) \gg 1$.

1. *Order the* n *vertices into a ring*
2. *Connect each vertex to its first* $k/2$ *right-hand (counterclockwise) neighbors, and to its* $k/2$ *left-hand (clockwise) neighbors.*
3. *With probability* p*, replace edge* $\langle u, v \rangle$ *with an edge* $\langle u, w \rangle$ *where* $w \neq u$ *is randomly chosen, but such that* $\langle u, w \rangle \notin E(G)$.
4. ***Notation****:* $WS(n, k, p)$ *graph*

Watts-Strogatz graphs

Algorithm (Watts-Strogatz)

$V = \{v_1, v_2, \ldots, v_n\}$. *Let* k *be even. Choose* $n \gg k \gg \ln(n) \gg 1$.

1. *Order the* n *vertices into a ring*
2. *Connect each vertex to its first* $k/2$ *right-hand (counterclockwise) neighbors, and to its* $k/2$ *left-hand (clockwise) neighbors.*
3. *With probability* p*, replace edge* $\langle u, v \rangle$ *with an edge* $\langle u, w \rangle$ *where* $w \neq u$ *is randomly chosen, but such that* $\langle u, w \rangle \notin E(G)$.
4. ***Notation***: $WS(n, k, p)$ *graph*

Watts-Strogatz graphs

Algorithm (Watts-Strogatz)

$V = \{v_1, v_2, \ldots, v_n\}$. *Let* k *be even. Choose* $n \gg k \gg \ln(n) \gg 1$.

1. *Order the* n *vertices into a ring*
2. *Connect each vertex to its first* $k/2$ *right-hand (counterclockwise) neighbors, and to its* $k/2$ *left-hand (clockwise) neighbors.*
3. *With probability* p*, replace edge* $\langle u, v\rangle$ *with an edge* $\langle u, w\rangle$ *where* $w \neq u$ *is randomly chosen, but such that* $\langle u, w\rangle \notin E(G)$.
4. ***Notation**: $WS(n, k, p)$ graph*

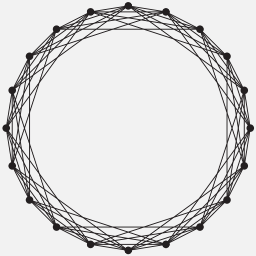
Watts-Strogatz graphs

Algorithm (Watts-Strogatz)

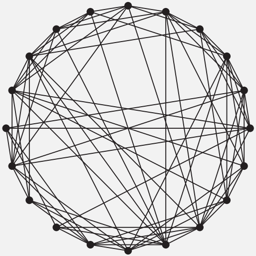
$V = \{v_1, v_2, \ldots, v_n\}$. *Let* k *be even. Choose* $n \gg k \gg \ln(n) \gg 1$.

1. *Order the* n *vertices into a ring*
2. *Connect each vertex to its first* $k/2$ *right-hand (counterclockwise) neighbors, and to its* $k/2$ *left-hand (clockwise) neighbors.*
3. *With probability* p*, replace edge* $\langle u, v\rangle$ *with an edge* $\langle u, w\rangle$ *where* $w \neq u$ *is randomly chosen, but such that* $\langle u, w\rangle \notin E(G)$.
4. ***Notation***: $WS(n, k, p)$ *graph*

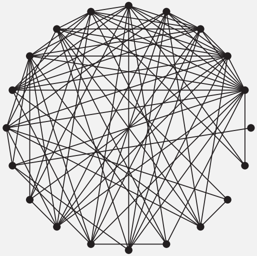
Watts-Strogatz graphs



$p = 0.0$



$p = 0.20$



$p = 0.90$

Note

$n = 20$; $k = 8$; $\ln(n) \approx 3$. Conditions are not really met.

Watts-Strogatz graphs

Observation

For many vertices in a WS-graph, $d(u, v)$ will be small:

- Each vertex has k nearby neighbors.
- There will be direct links to other “groups” of vertices.
- **weak links**: the long links in a WS-graph that cross the ring.

Watts-Strogatz graphs

Observation

For many vertices in a WS-graph, $d(u,v)$ will be small:

- Each vertex has k nearby neighbors.
- There will be direct links to other "groups" of vertices.
- **weak links**: the long links in a WS-graph that cross the ring.

Watts-Strogatz graphs

Observation

For many vertices in a WS-graph, $d(u, v)$ will be small:

- Each vertex has k nearby neighbors.
- There will be direct links to other “groups” of vertices.
- **weak links**: the long links in a WS-graph that cross the ring.

Watts-Strogatz graphs

Observation

For many vertices in a WS-graph, $d(u, v)$ will be small:

- Each vertex has k nearby neighbors.
- There will be direct links to other “groups” of vertices.
- **weak links**: the long links in a WS-graph that cross the ring.

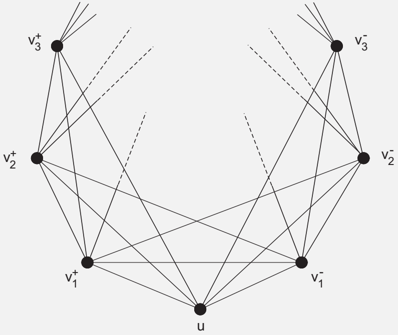
WS-graphs: clustering coefficient

Theorem

For any G from $WS(n,k,0)$, $CC(G) = \frac{3}{4}\frac{k-2}{k-1}$.

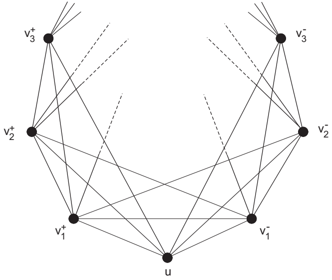
Proof

Choose arbitrary $u \in V(G)$. Let $H = G[N(u)]$. Note that $G[\{u\} \cup N(u)]$ is equal to:



WS-graphs: clustering coefficient

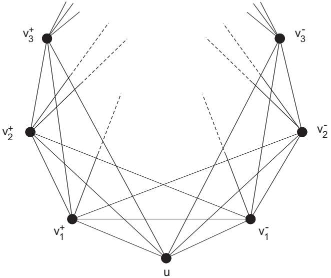
Proof (cntd)



- $\delta(v_1^-)$: The "farthest" right-hand neighbor of v_1^- is $v_{k/2}^-$
- Conclusion: v_1^- has $\frac{k}{2}-1$ right-hand neighbors in H.
- v_2^- has $\frac{k}{2}-2$ right-hand neighbors in H.
- In general: v_i^- has $\frac{k}{2}-i$ right-hand neighbors in H.

WS-graphs: clustering coefficient

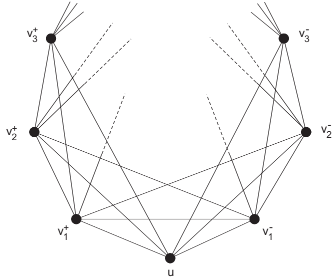
Proof (cntd)



- $\delta(v_1^-)$: The "farthest" right-hand neighbor of v_1^- is $v_{k/2}^-$
- Conclusion: v_1^- has $\frac{k}{2} - 1$ right-hand neighbors in H.
- v_2^- has $\frac{k}{2} - 2$ right-hand neighbors in H.
- In general: v_i^- has $\frac{k}{2} - i$ right-hand neighbors in H.

WS-graphs: clustering coefficient

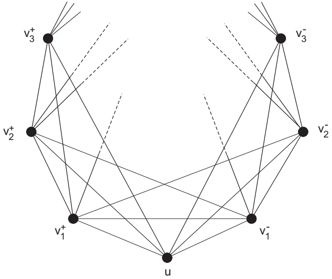
Proof (cntd)



- $\delta(v_1^-)$: The "farthest" right-hand neighbor of v_1^- is $v_{k/2}^-$
- Conclusion: v_1^- has $\frac{k}{2} - 1$ right-hand neighbors in H.
- v_2^- has $\frac{k}{2} - 2$ right-hand neighbors in H.
- In general: v_i^- has $\frac{k}{2} - i$ right-hand neighbors in H.

WS-graphs: clustering coefficient

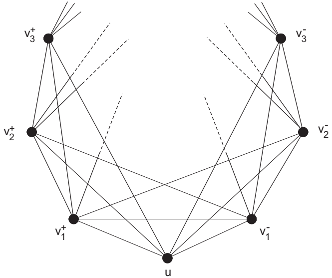
Proof (cntd)



- $\delta(v_1^-)$: The "farthest" right-hand neighbor of v_1^- is $v_{k/2}^-$
- Conclusion: v_1^- has $\frac{k}{2}-1$ right-hand neighbors in H.
- v_2^- has $\frac{k}{2}-2$ right-hand neighbors in H.
- In general: v_i^- has $\frac{k}{2}-i$ right-hand neighbors in H.

WS-graphs: clustering coefficient

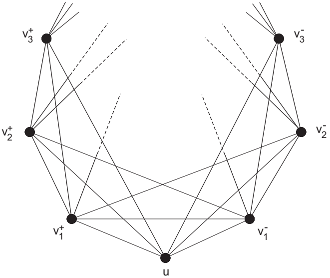
Proof (cntd)



- $\delta(v_1^-)$: The “farthest” right-hand neighbor of v_1^- is $v_{k/2}^-$
- Conclusion: v_1^- has $\frac{k}{2}-1$ right-hand neighbors in H.
- v_2^- has $\frac{k}{2}-2$ right-hand neighbors in H.
- In general: v_i^- has $\frac{k}{2}-i$ right-hand neighbors in H.

WS-graphs: clustering coefficient

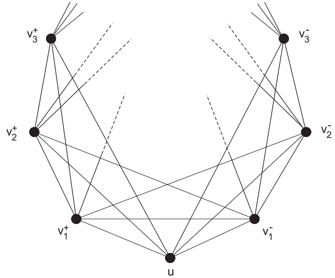
Proof (cntd)



- v_i^- is missing only u as left-hand neighbor in $H \Rightarrow v_i^-$ has $\frac{k}{2} - 1$ left-hand neighbors.
- $\delta(v_i^-) = \left(\frac{k}{2} - 1\right) + \left(\frac{k}{2} - i\right) = k - i - 1$ [Same for $\delta(v_i^+)$]

WS-graphs: clustering coefficient

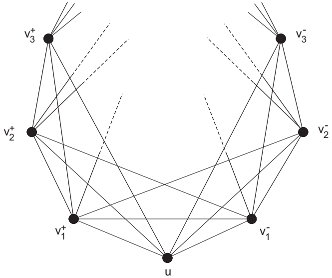
Proof (cntd)



- v_i^- is missing only u as left-hand neighbor in $H \Rightarrow v_i^-$ has $\frac{k}{2} - 1$ left-hand neighbors.
- $\delta(v_i^-) = \left(\frac{k}{2} - 1\right) + \left(\frac{k}{2} - i\right) = k - i - 1$ [Same for $\delta(v_i^+)$]

WS-graphs: clustering coefficient

Proof (cntd)



- v_i^- is missing only u as left-hand neighbor in $H \Rightarrow v_i^-$ has $\frac{k}{2} - 1$ left-hand neighbors.
- $\delta(v_i^-) = \left(\frac{k}{2} - 1\right) + \left(\frac{k}{2} - i\right) = k - i - 1$ [Same for $\delta(v_i^+)$]

WS-graphs: clustering coefficient

Proof (cntd)

- $|E(H)| = \frac{1}{2} \sum\limits_{v \in V(H)} \delta(v) =$

  $\frac{1}{2} \sum\limits_{i=1}^{k/2} \left(\delta(v_i^-) + \delta(v_i^+) \right) = \frac{1}{2} \cdot 2 \sum\limits_{i=1}^{k/2} \delta(v_i^-) = \sum\limits_{i=1}^{k/2} (k - i - 1)$
- $\sum_{i=1}^{m} i = \frac{1}{2} m(m+1) \Rightarrow |E(H)| = \frac{3}{8} k(k-2)$
- $|V(H)| = k \Rightarrow$

$$cc(u) = \frac{|E(H)|}{\binom{k}{2}} = \frac{\frac{3}{8}k(k-2)}{\frac{1}{2}k(k-1)} = \frac{3(k-2)}{4(k-1)}$$

WS-graphs: clustering coefficient

Proof (cntd)

- $|E(H)| = \frac{1}{2} \sum_{v \in V(H)} \delta(v) =$
  $\frac{1}{2} \sum_{i=1}^{k/2} \left(\delta(v_i^-) + \delta(v_i^+) \right) = \frac{1}{2} \cdot 2 \sum_{i=1}^{k/2} \delta(v_i^-) = \sum_{i=1}^{k/2} (k - i - 1)$
- $\sum_{i=1}^{m} i = \frac{1}{2} m(m+1) \Rightarrow |E(H)| = \frac{3}{8} k(k-2)$
- $|V(H)| = k \Rightarrow$

$$cc(u) = \frac{|E(H)|}{\binom{k}{2}} = \frac{\frac{3}{8}k(k-2)}{\frac{1}{2}k(k-1)} = \frac{3(k-2)}{4(k-1)}$$

WS-graphs: clustering coefficient

Proof (cntd)

- $|E(H)| = \frac{1}{2} \sum_{v \in V(H)} \delta(v) =$

  $\frac{1}{2} \sum_{i=1}^{k/2} \left(\delta(v_i^-) + \delta(v_i^+) \right) = \frac{1}{2} \cdot 2 \sum_{i=1}^{k/2} \delta(v_i^-) = \sum_{i=1}^{k/2} (k - i - 1)$
- $\sum_{i=1}^{m} i = \frac{1}{2} m(m+1) \Rightarrow |E(H)| = \frac{3}{8} k(k-2)$
- $|V(H)| = k \Rightarrow$

$$cc(u) = \frac{|E(H)|}{\binom{k}{2}} = \frac{\frac{3}{8}k(k-2)}{\frac{1}{2}k(k-1)} = \frac{3(k-2)}{4(k-1)}$$

WS-graphs: clustering coefficient

Proof (cntd)

- $|E(H)| = \frac{1}{2}\sum_{v \in V(H)} \delta(v) =$

  $\frac{1}{2}\sum_{i=1}^{k/2}\left(\delta(v_i^-)+\delta(v_i^+)\right) = \frac{1}{2}\cdot 2\sum_{i=1}^{k/2}\delta(v_i^-) = \sum_{i=1}^{k/2}(k-i-1)$
- $\sum_{i=1}^{m} i = \frac{1}{2}m(m+1) \Rightarrow |E(H)| = \frac{3}{8}k(k-2)$
- $|V(H)| = k \Rightarrow$

$$cc(u) = \frac{|E(H)|}{\binom{k}{2}} = \frac{\frac{3}{8}k(k-2)}{\frac{1}{2}k(k-1)} = \frac{3(k-2)}{4(k-1)}$$

WS-graphs: average shortest path length

Theorem

$\forall G \in WS(n,k,0)$ the average shortest-path length $\overline{d}(u)$ from vertex u to any other vertex is approximated by

$$\overline{d}(u) \approx \frac{(n-1)(n+k-1)}{2kn}$$

WS-graphs: average shortest path length

Proof

- Let $L(u,1) =$ left-hand vertices $\{v_1^+, v_2^+, \ldots, v_{k/2}^+\}$
- Let $L(u,2) =$ left-hand vertices $\{v_{k/2+1}^+, \ldots, v_k^+\}$.
- Let $L(u,m) =$ left-hand vertices $\{v_{(m-1)k/2+1}^+, \ldots, v_{mk/2}^+\}$.
- Note: $\forall v \in L(u,m)$: v is connected to a vertex from $L(u,m-1)$.

Note

$L(u,m) =$ left-hand neighbors connected to u through a (shortest) path of length m. Define analogously $R(u,m)$.

WS-graphs: average shortest path length

Proof

- Let $L(u,1) =$ left-hand vertices $\{v_1^+, v_2^+, \ldots, v_{k/2}^+\}$
- Let $L(u,2) =$ left-hand vertices $\{v_{k/2+1}^+, \ldots, v_k^+\}$.
- Let $L(u,m) =$ left-hand vertices $\{v_{(m-1)k/2+1}^+, \ldots, v_{mk/2}^+\}$.
- Note: $\forall v \in L(u,m)$: v is connected to a vertex from $L(u,m-1)$.

Note

$L(u,m) =$ left-hand neighbors connected to u through a (shortest) path of length m. Define analogously $R(u,m)$.

WS-graphs: average shortest path length

Proof

- Let $L(u,1)$ = left-hand vertices $\{v_1^+, v_2^+, \ldots, v_{k/2}^+\}$
- Let $L(u,2)$ = left-hand vertices $\{v_{k/2+1}^+, \ldots, v_k^+\}$.
- Let $L(u,m)$ = left-hand vertices $\{v_{(m-1)k/2+1}^+, \ldots, v_{mk/2}^+\}$.
- Note: $\forall v \in L(u,m)$: v is connected to a vertex from $L(u,m-1)$.

Note

$L(u,m)$ = left-hand neighbors connected to u through a (shortest) path of length m. Define analogously $R(u,m)$.

WS-graphs: average shortest path length

Proof

- Let $L(u,1) =$ left-hand vertices $\{v_1^+, v_2^+, \ldots, v_{k/2}^+\}$
- Let $L(u,2) =$ left-hand vertices $\{v_{k/2+1}^+, \ldots, v_k^+\}$.
- Let $L(u,m) =$ left-hand vertices $\{v_{(m-1)k/2+1}^+, \ldots, v_{mk/2}^+\}$.
- Note: $\forall v \in L(u,m)$: v is connected to a vertex from $L(u,m-1)$.

Note

$L(u,m) =$ left-hand neighbors connected to u through a (shortest) path of length m. Define analogously $R(u,m)$.

WS-graphs: average shortest path length

Proof

- Let $L(u,1) =$ left-hand vertices $\{v_1^+, v_2^+, \ldots, v_{k/2}^+\}$
- Let $L(u,2) =$ left-hand vertices $\{v_{k/2+1}^+, \ldots, v_k^+\}$.
- Let $L(u,m) =$ left-hand vertices $\{v_{(m-1)k/2+1}^+, \ldots, v_{mk/2}^+\}$.
- Note: $\forall v \in L(u,m)$: v is connected to a vertex from $L(u,m-1)$.

Note

$L(u,m) =$ left-hand neighbors connected to u through a (shortest) path of length m. Define analogously $R(u,m)$.

WS-graphs: average shortest path length

Proof

- Let $L(u,1) =$ left-hand vertices $\{v_1^+, v_2^+, \ldots, v_{k/2}^+\}$
- Let $L(u,2) =$ left-hand vertices $\{v_{k/2+1}^+, \ldots, v_k^+\}$.
- Let $L(u,m) =$ left-hand vertices $\{v_{(m-1)k/2+1}^+, \ldots, v_{mk/2}^+\}$.
- Note: $\forall v \in L(u,m)$: v is connected to a vertex from $L(u,m-1)$.

Note

$L(u,m) =$ left-hand neighbors connected to u through a (shortest) path of length m. Define analogously $R(u,m)$.

WS-graphs: average shortest path length

Proof (cntd)

- Index p of the farthest vertex v_p^+ contained in any $L(u,m)$ will be less than approximately $(n-1)/2$.
- All $L(u,m)$ have equal size $\Rightarrow m \cdot k/2 \leq (n-1)/2 \Rightarrow m \leq \frac{(n-1)/2}{k/2}$.

$$\overline{d}(u) \approx 2\frac{1\cdot|L(u,1)|+2\cdot|L(u,2)|+\ldots\frac{n-1}{k}\cdot|L(u,m)|}{n}$$

which leads to

$$\overline{d}(u) \approx \frac{k}{n}\sum_{i=1}^{(n-1)/k} i = \frac{k}{2n}\left(\frac{n-1}{k}\right)\left(\frac{n-1}{k}+1\right) = \frac{(n-1)(n+k-1)}{2kn}$$

WS-graphs: average shortest path length

Proof (cntd)

- Index p of the farthest vertex v_p^+ contained in any $L(u,m)$ will be less than approximately $(n-1)/2$.
- All $L(u,m)$ have equal size $\Rightarrow m \cdot k/2 \leq (n-1)/2 \Rightarrow m \leq \frac{(n-1)/2}{k/2}$.

$$\overline{d}(u) \approx 2\frac{1 \cdot |L(u,1)| + 2 \cdot |L(u,2)| + \ldots \frac{n-1}{k} \cdot |L(u,m)|}{n}$$

which leads to

$$\overline{d}(u) \approx \frac{k}{n} \sum_{i=1}^{(n-1)/k} i = \frac{k}{2n}\left(\frac{n-1}{k}\right)\left(\frac{n-1}{k}+1\right) = \frac{(n-1)(n+k-1)}{2kn}$$

WS-graphs: average shortest path length

Proof (cntd)

- Index p of the farthest vertex v_p^+ contained in any $L(u,m)$ will be less than approximately $(n-1)/2$.
- All $L(u,m)$ have equal size $\Rightarrow m \cdot k/2 \leq (n-1)/2 \Rightarrow m \leq \frac{(n-1)/2}{k/2}$.

$$\overline{d}(u) \approx 2\frac{1\cdot|L(u,1)|+2\cdot|L(u,2)|+\ldots\frac{n-1}{k}\cdot|L(u,m)|}{n}$$

which leads to

$$\overline{d}(u) \approx \frac{k}{n}\sum_{i=1}^{(n-1)/k} i = \frac{k}{2n}\left(\frac{n-1}{k}\right)\left(\frac{n-1}{k}+1\right) = \frac{(n-1)(n+k-1)}{2kn}$$

WS-graphs: average shortest path length

Proof (cntd)

- Index p of the farthest vertex v_p^+ contained in any $L(u,m)$ will be less than approximately $(n-1)/2$.
- All $L(u,m)$ have equal size $\Rightarrow m \cdot k/2 \leq (n-1)/2 \Rightarrow m \leq \frac{(n-1)/2}{k/2}$.

$$\overline{d}(u) \approx 2\frac{1\cdot|L(u,1)|+2\cdot|L(u,2)|+\ldots\frac{n-1}{k}\cdot|L(u,m)|}{n}$$

which leads to

$$\overline{d}(u) \approx \frac{k}{n}\sum_{i=1}^{(n-1)/k} i = \frac{k}{2n}\left(\frac{n-1}{k}\right)\left(\frac{n-1}{k}+1\right) = \frac{(n-1)(n+k-1)}{2kn}$$

WS-graphs: average shortest path length

Proof (cntd)

- Index p of the farthest vertex v_p^+ contained in any $L(u,m)$ will be less than approximately $(n-1)/2$.
- All $L(u,m)$ have equal size $\Rightarrow m \cdot k/2 \leq (n-1)/2 \Rightarrow m \leq \frac{(n-1)/2}{k/2}$.

$$\overline{d}(u) \approx 2\frac{1 \cdot |L(u,1)| + 2 \cdot |L(u,2)| + \ldots \frac{n-1}{k} \cdot |L(u,m)|}{n}$$

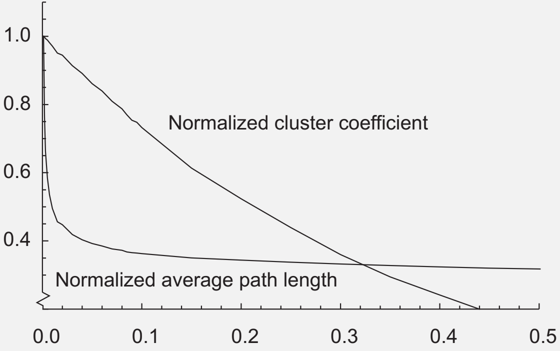
which leads to

$$\overline{d}(u) \approx \frac{k}{n} \sum_{i=1}^{(n-1)/k} i = \frac{k}{2n}\left(\frac{n-1}{k}\right)\left(\frac{n-1}{k}+1\right) = \frac{(n-1)(n+k-1)}{2kn}$$

WS-graphs: comparison to real-world networks

Observation

$WS(n, k, 0)$ graphs have long shortest paths, yet high clustering coefficient. However, increasing p shows that average path length drops rapidly.



Normalized: divide by $CC(G_0)$ and $\overline{d}(G_0)$ with $G_0 \in WS(n, k, 0)$

Scale-free networks

Important observation

In many real-world networks we see very few high-degree nodes, and that the number of high-degree nodes decreases exponentially: Web link structure, Internet topology, collaboration networks, etc.

Characterization

In a scale-free network, $\mathbb{P}[\delta(u) = k] \propto k^{-\alpha}$

Definition

A function f is scale-free iff $f(bx) = C(b) \cdot f(x)$ where $C(b)$ is a constant dependent only on b

Scale-free networks

Important observation

In many real-world networks we see very few high-degree nodes, and that the number of high-degree nodes decreases exponentially: Web link structure, Internet topology, collaboration networks, etc.

Characterization

In a scale-free network, $\mathbb{P}[\delta(u) = k] \propto k^{-\alpha}$

Definition

A function f is scale-free iff $f(bx) = C(b) \cdot f(x)$ where $C(b)$ is a constant dependent only on b

Scale-free networks

Important observation

In many real-world networks we see very few high-degree nodes, and that the number of high-degree nodes decreases exponentially: Web link structure, Internet topology, collaboration networks, etc.

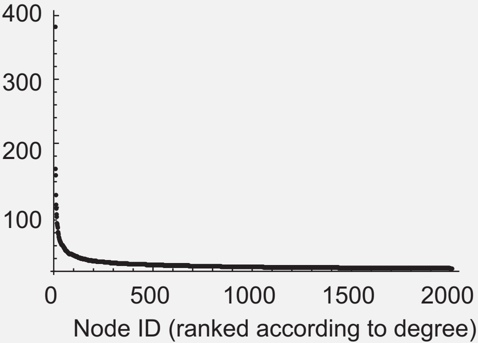
Characterization

In a scale-free network, $\mathbb{P}[\delta(u) = k] \propto k^{-\alpha}$

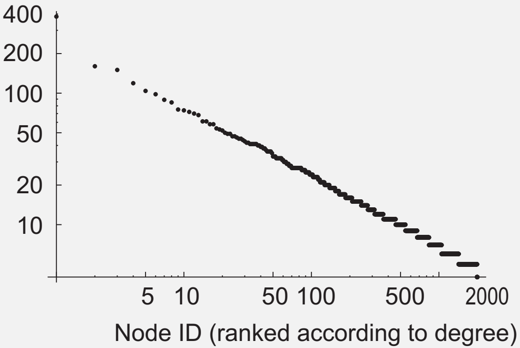
Definition

A function f is scale-free iff $f(bx) = C(b) \cdot f(x)$ where $C(b)$ is a constant dependent only on b

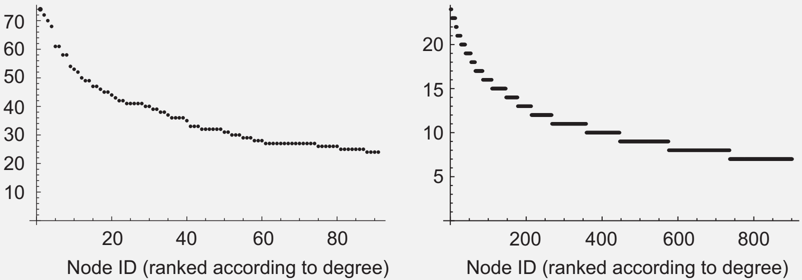
Example scale-free network



Example scale-free network



What's in a name: scale-free



Constructing SF networks

Observation

Where ER and WS graphs can be constructed from a given set of vertices, scale-free networks result from a **growth process** combined with **preferential attachment**.

Barabási-Albert networks

Algorithm (Barabási-Albert)

$G_0 \in ER(n_0, p)$ with $V_0 = V(G_0)$. At each step $s > 0$:

1. *Add a new vertex v_s : $V_s \leftarrow V_{s-1} \cup \{v_s\}$.*
2. *Add $m \leq n_0$ edges incident to v_s and a vertex u from V_{s-1} (and u not chosen before in current step). Choose u with probability*

$$\mathbb{P}[select\ u] = \frac{\delta(u)}{\sum_{w \in V_{s-1}} \delta(w)}$$

   Note: choose u proportional to its current degree.

3. *Stop when n vertices have been added, otherwise repeat the previous two steps.*

Result: a Barabási-Albert graph, $BA(n, n_0, m)$.

Barabási-Albert networks

Algorithm (Barabási-Albert)

$G_0 \in ER(n_0, p)$ with $V_0 = V(G_0)$. At each step $s > 0$:

1. *Add a new vertex v_s : $V_s \leftarrow V_{s-1} \cup \{v_s\}$.*
2. *Add $m \leq n_0$ edges incident to v_s and a vertex u from V_{s-1} (and u not chosen before in current step). Choose u with probability*

   $$\mathbb{P}[select\ u] = \frac{\delta(u)}{\sum_{w \in V_{s-1}} \delta(w)}$$

   Note: choose u proportional to its current degree.
3. *Stop when n vertices have been added, otherwise repeat the previous two steps.*

Result: a Barabási-Albert graph, $BA(n, n_0, m)$.

Barabási-Albert networks

Algorithm (Barabási-Albert)

$G_0 \in ER(n_0, p)$ with $V_0 = V(G_0)$. At each step $s > 0$:

1. *Add a new vertex v_s : $V_s \leftarrow V_{s-1} \cup \{v_s\}$.*
2. *Add $m \leq n_0$ edges incident to v_s and a vertex u from V_{s-1} (and u not chosen before in current step). Choose u with probability*

$$\mathbb{P}[select\ u] = \frac{\delta(u)}{\sum_{w \in V_{s-1}} \delta(w)}$$

   Note: choose u proportional to its current degree.

3. *Stop when n vertices have been added, otherwise repeat the previous two steps.*

Result: a Barabási-Albert graph, $BA(n, n_0, m)$.

Barabási-Albert networks

Algorithm (Barabási-Albert)

$G_0 \in ER(n_0, p)$ with $V_0 = V(G_0)$. At each step $s > 0$:

1. *Add a new vertex v_s : $V_s \leftarrow V_{s-1} \cup \{v_s\}$.*
2. *Add $m \leq n_0$ edges incident to v_s and a vertex u from V_{s-1} (and u not chosen before in current step). Choose u with probability*

$$\mathbb{P}[select\ u] = \frac{\delta(u)}{\sum_{w \in V_{s-1}} \delta(w)}$$

   Note: choose u proportional to its current degree.

3. *Stop when n vertices have been added, otherwise repeat the previous two steps.*

Result: a Barabási-Albert graph, $BA(n, n_0, m)$.

Barabási-Albert networks

Algorithm (Barabási-Albert)

$G_0 \in ER(n_0, p)$ with $V_0 = V(G_0)$. At each step $s > 0$:

1. *Add a new vertex v_s : $V_s \leftarrow V_{s-1} \cup \{v_s\}$.*
2. *Add $m \leq n_0$ edges incident to v_s and a vertex u from V_{s-1} (and u not chosen before in current step). Choose u with probability*

$$\mathbb{P}[select\ u] = \frac{\delta(u)}{\sum_{w \in V_{s-1}} \delta(w)}$$

   Note: choose u proportional to its current degree.

3. *Stop when n vertices have been added, otherwise repeat the previous two steps.*

Result: a Barabási-Albert graph, $BA(n, n_0, m)$.

BA-graphs: degree distribution

Theorem

For any $BA(n, n_0, m)$ *graph* G *and* $u \in V(G)$*:*

$$\mathbb{P}[\delta(u) = k] = \frac{2m(m+1)}{k(k+1)(k+2)} \propto \frac{1}{k^3}$$

Generalized BA-graphs

Algorithm

G_0 has n_0 vertices V_0 and no edges. At each step $s > 0$:

1. *Add a new vertex v_s to V_{s-1}.*
2. *Add $m \leq n_0$ edges incident to v_s and different vertices u from V_{s-1} (u not chosen before during current step). Choose u with probability proportional to its current degree $\delta(u)$.*
3. *For some constant $c \geq 0$ add another $c \times m$ edges between vertices from V_{s-1}; probability adding edge between u and w is proportional to the product $\delta(u) \cdot \delta(w)$ (and $\langle u, w \rangle$ does not yet exist).*
4. *Stop when n vertices have been added.*

Generalized BA-graphs

Algorithm

G_0 has n_0 vertices V_0 and no edges. At each step $s > 0$:

1. *Add a new vertex v_s to V_{s-1}.*
2. *Add $m \leq n_0$ edges incident to v_s and different vertices u from V_{s-1} (u not chosen before during current step). Choose u with probability proportional to its current degree $\delta(u)$.*
3. *For some constant $c \geq 0$ add another $c \times m$ edges between vertices from V_{s-1}; probability adding edge between u and w is proportional to the product $\delta(u) \cdot \delta(w)$ (and $\langle u, w \rangle$ does not yet exist).*
4. *Stop when n vertices have been added.*

Generalized BA-graphs

Algorithm

G_0 has n_0 vertices V_0 and no edges. At each step $s > 0$:

1. *Add a new vertex v_s to V_{s-1}.*
2. *Add $m \leq n_0$ edges incident to v_s and different vertices u from V_{s-1} (u not chosen before during current step). Choose u with probability proportional to its current degree $\delta(u)$.*
3. *For some constant $c \geq 0$ add another $c \times m$ edges between vertices from V_{s-1}; probability adding edge between u and w is proportional to the product $\delta(u) \cdot \delta(w)$ (and $\langle u, w \rangle$ does not yet exist).*
4. *Stop when n vertices have been added.*

Generalized BA-graphs

Algorithm

G_0 has n_0 vertices V_0 and no edges. At each step $s > 0$:

1. *Add a new vertex v_s to V_{s-1}.*
2. *Add $m \leq n_0$ edges incident to v_s and different vertices u from V_{s-1} (u not chosen before during current step). Choose u with probability proportional to its current degree $\delta(u)$.*
3. *For some constant $c \geq 0$ add another $c \times m$ edges between vertices from V_{s-1}; probability adding edge between u and w is proportional to the product $\delta(u) \cdot \delta(w)$ (and $\langle u, w \rangle$ does not yet exist).*
4. *Stop when n vertices have been added.*

Generalized BA-graphs

Algorithm

G_0 has n_0 vertices V_0 and no edges. At each step $s > 0$:

1. *Add a new vertex v_s to V_{s-1}.*
2. *Add $m \leq n_0$ edges incident to v_s and different vertices u from V_{s-1} (u not chosen before during current step). Choose u with probability proportional to its current degree $\delta(u)$.*
3. *For some constant $c \geq 0$ add another $c \times m$ edges between vertices from V_{s-1}; probability adding edge between u and w is proportional to the product $\delta(u) \cdot \delta(w)$ (and $\langle u, w \rangle$ does not yet exist).*
4. *Stop when n vertices have been added.*

Generalized BA-graphs: degree distribution

Theorem

For any generalized BA(n, n_0, m) graph G and $u \in V(G)$:

$$\mathbb{P}[\delta(u) = k] \propto k^{-(2+\frac{1}{1+2c})}$$

Observation

- For $c = 0$, we have a BA-graph;
- $\lim_{c\to\infty} \mathbb{P}[\delta(u) = k] \propto \frac{1}{k^2}$

Generalized BA-graphs: degree distribution

Theorem

For any generalized $BA(n, n_0, m)$ graph G and $u \in V(G)$:

$$\mathbb{P}[\delta(u) = k] \propto k^{-(2+\frac{1}{1+2c})}$$

Observation

- For $c = 0$, we have a BA-graph;
- $\lim_{c\to\infty} \mathbb{P}[\delta(u) = k] \propto \frac{1}{k^2}$

BA-graphs: clustering coefficient

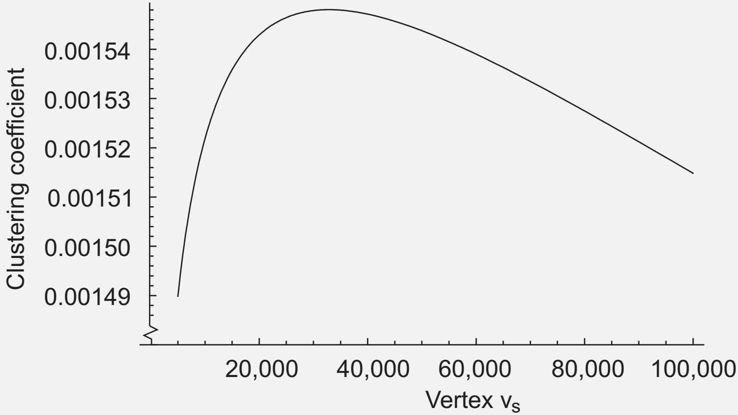
BA-graphs after t steps

Consider clustering coefficient of vertex v_s after t steps in the construction of a $BA(t, n_0, m)$ graph. **Note:** v_s was added at step $s \leq t$.

$$cc(v_s) = \frac{m-1}{8(\sqrt{t}+\sqrt{s}/m)^2}\left(\ln^2(t) + \frac{4m}{(m-1)^2}\ln^2(s)\right)$$

BA-graphs: clustering coefficient

Note: Fix *m* and *t* and vary *s*:



Comparing clustering coefficients

Issue: Construct an ER graph with same number of vertices and average vertex degree:

$$\begin{aligned}
\overline{\delta}(G) &= \mathbb{E}[\delta] = \sum_{k=m}^{\infty} k \cdot \mathbb{P}[\delta(u) = k] \\
&= \sum_{k=m}^{\infty} k \cdot \frac{2m(m+1)}{k(k+1)(k+2)} \\
&= 2m(m+1) \sum_{k=m}^{\infty} \frac{k}{k(k+1)(k+2)} \\
&= 2m(m+1) \cdot \frac{1}{m+1} = 2m
\end{aligned}$$

ER-graph: $\overline{\delta}(G) = p(n-1) \Rightarrow$ choose $p = \frac{2m}{n-1}$

Example

$BA(100,000,0,8)$-graph has $cc(v) \approx 0.0015$; $ER(100,000,p)$-graph has $cc(v) \approx 0.00016$

Comparing clustering coefficients

Issue: Construct an ER graph with same number of vertices and average vertex degree:

$$\begin{aligned}
\overline{\delta}(G) &= \mathbb{E}[\delta] = \sum_{k=m}^{\infty} k \cdot \mathbb{P}[\delta(u) = k] \\
&= \sum_{k=m}^{\infty} k \cdot \frac{2m(m+1)}{k(k+1)(k+2)} \\
&= 2m(m+1) \sum_{k=m}^{\infty} \frac{k}{k(k+1)(k+2)} \\
&= 2m(m+1) \cdot \frac{1}{m+1} = 2m
\end{aligned}$$

ER-graph: $\overline{\delta}(G) = p(n-1) \Rightarrow$ choose $p = \frac{2m}{n-1}$

Example

$BA(100,000,0,8)$-graph has $cc(v) \approx 0.0015$; $ER(100,000,p)$-graph has $cc(v) \approx 0.00016$

Comparing clustering coefficients

Issue: Construct an ER graph with same number of vertices and average vertex degree:

$$\begin{aligned}
\overline{\delta}(G) &= \mathbb{E}[\delta] = \sum_{k=m}^{\infty} k \cdot \mathbb{P}[\delta(u) = k] \\
&= \sum_{k=m}^{\infty} k \cdot \frac{2m(m+1)}{k(k+1)(k+2)} \\
&= 2m(m+1) \sum_{k=m}^{\infty} \frac{k}{k(k+1)(k+2)} \\
&= 2m(m+1) \cdot \frac{1}{m+1} = 2m
\end{aligned}$$

ER-graph: $\overline{\delta}(G) = p(n-1) \Rightarrow$ choose $p = \frac{2m}{n-1}$

Example

$BA(100,000,0,8)$-graph has $cc(v) \approx 0.0015$; $ER(100,000,p)$-graph has $cc(v) \approx 0.00016$

Comparing clustering coefficients

Issue: Construct an ER graph with same number of vertices and average vertex degree:

$$\begin{aligned}
\overline{\delta}(G) &= \mathbb{E}[\delta] = \sum_{k=m}^{\infty} k \cdot \mathbb{P}[\delta(u) = k] \\
&= \sum_{k=m}^{\infty} k \cdot \frac{2m(m+1)}{k(k+1)(k+2)} \\
&= 2m(m+1) \sum_{k=m}^{\infty} \frac{k}{k(k+1)(k+2)} \\
&= 2m(m+1) \cdot \frac{1}{m+1} = 2m
\end{aligned}$$

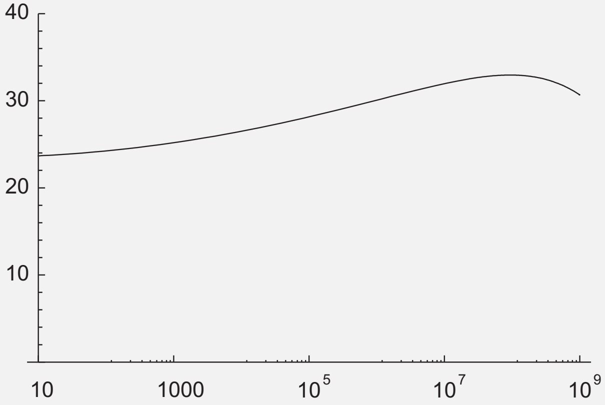
ER-graph: $\overline{\delta}(G) = p(n-1) \Rightarrow$ choose $p = \frac{2m}{n-1}$

Example

$BA(100,000,0,8)$-graph has $cc(v) \approx 0.0015$; $ER(100,000,p)$-graph has $cc(v) \approx 0.00016$

Comparing clustering coefficients

Further comparison: Ratio of $cc(v_s)$ between $BA(N \leq 1\,000\,000\,000, 0, 8)$-graph to an $ER(N,p)$-graph

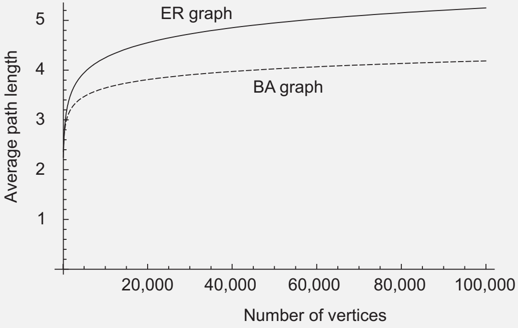


Average path lengths

Observation

$$\overline{d}(BA) = \frac{\ln(n) - \ln(m/2) - 1 - \gamma}{\ln(\ln(n)) + \ln(m/2)} + 1.5$$

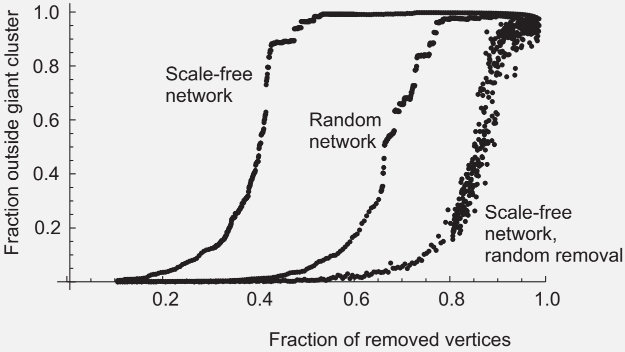
with $\gamma \approx 0.5772$ the Euler constant. For $\overline{\delta}(v) = 10$:



Scale-free graphs and robustness

Observation

Scale-free networks have **hubs** making them vulnerable to **targeted attacks**.



Barabási-Albert with tunable clustering

Algorithm

Consider a small graph G_0 with n_0 vertices V_0 and no edges. At each step $s > 0$:

1. *Add a new vertex v_s to V_{s-1}.*
2. *Select u from V_{s-1} not adjacent to v_s, with probability proportional to $\delta(u)$. Add edge $\langle v_s, u\rangle$.*
   - (a) *If $m - 1$ edges have been added, continue with Step 3.*
   - (b) *With probability q: select a vertex w adjacent to u, but not to v_s. If no such vertex exists, continue with Step c. Otherwise, add edge $\langle v_s, w\rangle$ and continue with Step a.*
   - (c) *Select vertex u' from V_{s-1} not adjacent to v_s with probability proportional to $\delta(u')$. Add edge $\langle v_s, u'\rangle$ and set $u \leftarrow u'$. Continue with Step a.*
3. *If n vertices have been added stop, else go to Step 1.*

Barabási-Albert with tunable clustering

Algorithm

Consider a small graph G_0 with n_0 vertices V_0 and no edges. At each step $s > 0$:

1. *Add a new vertex v_s to V_{s-1}.*
2. *Select u from V_{s-1} not adjacent to v_s, with probability proportional to $\delta(u)$. Add edge $\langle v_s, u\rangle$.*
   - (a) *If $m-1$ edges have been added, continue with Step 3.*
   - (b) *With probability q: select a vertex w adjacent to u, but not to v_s. If no such vertex exists, continue with Step c. Otherwise, add edge $\langle v_s, w\rangle$ and continue with Step a.*
   - (c) *Select vertex u' from V_{s-1} not adjacent to v_s with probability proportional to $\delta(u')$. Add edge $\langle v_s, u'\rangle$ and set $u \leftarrow u'$. Continue with Step a.*
3. *If n vertices have been added stop, else go to Step 1.*

Barabási-Albert with tunable clustering

Algorithm

Consider a small graph G_0 with n_0 vertices V_0 and no edges. At each step $s > 0$:

1. *Add a new vertex v_s to V_{s-1}.*
2. *Select u from V_{s-1} not adjacent to v_s, with probability proportional to $\delta(u)$. Add edge $\langle v_s, u \rangle$.*
   - (a) *If $m-1$ edges have been added, continue with Step 3.*
   - (b) *With probability q: select a vertex w adjacent to u, but not to v_s. If no such vertex exists, continue with Step c. Otherwise, add edge $\langle v_s, w \rangle$ and continue with Step a.*
   - (c) *Select vertex u' from V_{s-1} not adjacent to v_s with probability proportional to $\delta(u')$. Add edge $\langle v_s, u' \rangle$ and set $u \leftarrow u'$. Continue with Step a.*
3. *If n vertices have been added stop, else go to Step 1.*

Barabási-Albert with tunable clustering

Algorithm

Consider a small graph G_0 with n_0 vertices V_0 and no edges. At each step $s > 0$:

1. *Add a new vertex v_s to V_{s-1}.*
2. *Select u from V_{s-1} not adjacent to v_s, with probability proportional to $\delta(u)$. Add edge $\langle v_s, u \rangle$.*
   (a) *If $m-1$ edges have been added, continue with Step 3.*
   (b) *With probability q: select a vertex w adjacent to u, but not to v_s. If no such vertex exists, continue with Step c. Otherwise, add edge $\langle v_s, w \rangle$ and continue with Step a.*
   (c) *Select vertex u' from V_{s-1} not adjacent to v_s with probability proportional to $\delta(u')$. Add edge $\langle v_s, u' \rangle$ and set $u \leftarrow u'$. Continue with Step a.*
3. *If n vertices have been added stop, else go to Step 1.*

Barabási-Albert with tunable clustering

Algorithm

Consider a small graph G_0 with n_0 vertices V_0 and no edges. At each step $s > 0$:

1. *Add a new vertex v_s to V_{s-1}.*
2. *Select u from V_{s-1} not adjacent to v_s, with probability proportional to $\delta(u)$. Add edge $\langle v_s, u\rangle$.*
   - (a) *If $m-1$ edges have been added, continue with Step 3.*
   - (b) *With probability q: select a vertex w adjacent to u, but not to v_s. If no such vertex exists, continue with Step c. Otherwise, add edge $\langle v_s, w\rangle$ and continue with Step a.*
   - (c) *Select vertex u' from V_{s-1} not adjacent to v_s with probability proportional to $\delta(u')$. Add edge $\langle v_s, u'\rangle$ and set $u \leftarrow u'$. Continue with Step a.*
3. *If n vertices have been added stop, else go to Step 1.*

Barabási-Albert with tunable clustering

Algorithm

Consider a small graph G_0 with n_0 vertices V_0 and no edges. At each step $s > 0$:

1. *Add a new vertex v_s to V_{s-1}.*
2. *Select u from V_{s-1} not adjacent to v_s, with probability proportional to $\delta(u)$. Add edge $\langle v_s, u \rangle$.*
   (a) *If $m-1$ edges have been added, continue with Step 3.*
   (b) *With probability q: select a vertex w adjacent to u, but not to v_s. If no such vertex exists, continue with Step c. Otherwise, add edge $\langle v_s, w \rangle$ and continue with Step a.*
   (c) *Select vertex u' from V_{s-1} not adjacent to v_s with probability proportional to $\delta(u')$. Add edge $\langle v_s, u' \rangle$ and set $u \leftarrow u'$. Continue with Step a.*
3. *If n vertices have been added stop, else go to Step 1.*

Barabási-Albert with tunable clustering

Algorithm

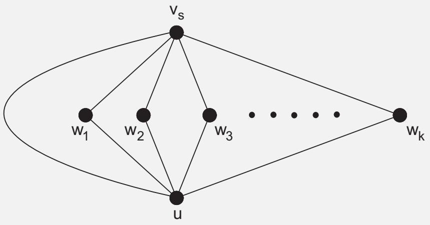
Consider a small graph G_0 with n_0 vertices V_0 and no edges. At each step $s > 0$:

1. *Add a new vertex v_s to V_{s-1}.*
2. *Select u from V_{s-1} not adjacent to v_s, with probability proportional to $\delta(u)$. Add edge $\langle v_s, u\rangle$.*
   - (a) *If $m-1$ edges have been added, continue with Step 3.*
   - (b) *With probability q: select a vertex w adjacent to u, but not to v_s. If no such vertex exists, continue with Step c. Otherwise, add edge $\langle v_s, w\rangle$ and continue with Step a.*
   - (c) *Select vertex u' from V_{s-1} not adjacent to v_s with probability proportional to $\delta(u')$. Add edge $\langle v_s, u'\rangle$ and set $u \leftarrow u'$. Continue with Step a.*
3. *If n vertices have been added stop, else go to Step 1.*

Barabási-Albert with tunable clustering

Special case: $q = 1$

If we add edges $\langle v_s, w \rangle$ with probability 1, we obtain a previously constructed subgraph.



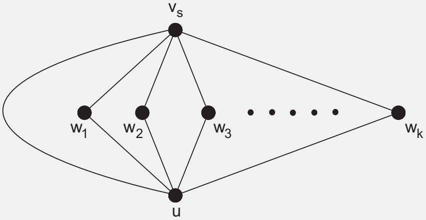
Recall

$$cc(x) = \begin{cases} 1 & \text{if } x = w_i \\ \frac{2}{k+1} & \text{if } x = u, v_s \end{cases}$$

Barabási-Albert with tunable clustering

Special case: $q = 1$

If we add edges $\langle v_s, w \rangle$ with probability 1, we obtain a previously constructed subgraph.



Recall

$$cc(x) = \begin{cases} 1 & \text{if } x = w_i \\ \frac{2}{k+1} & \text{if } x = u, v_s \end{cases}$$