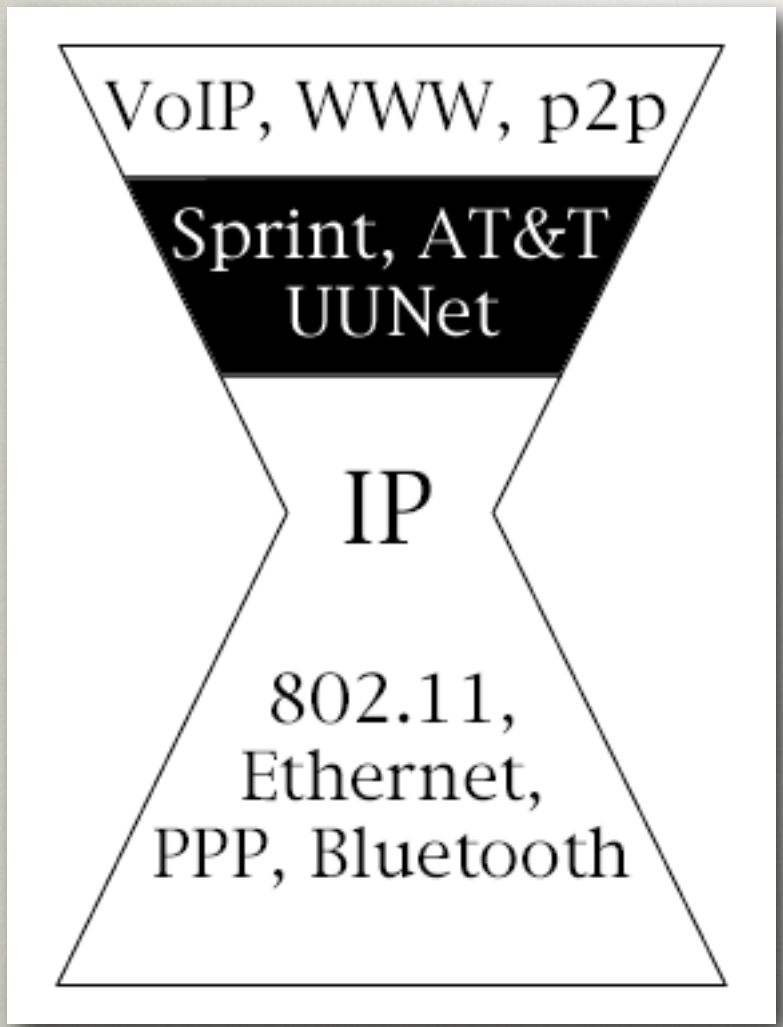
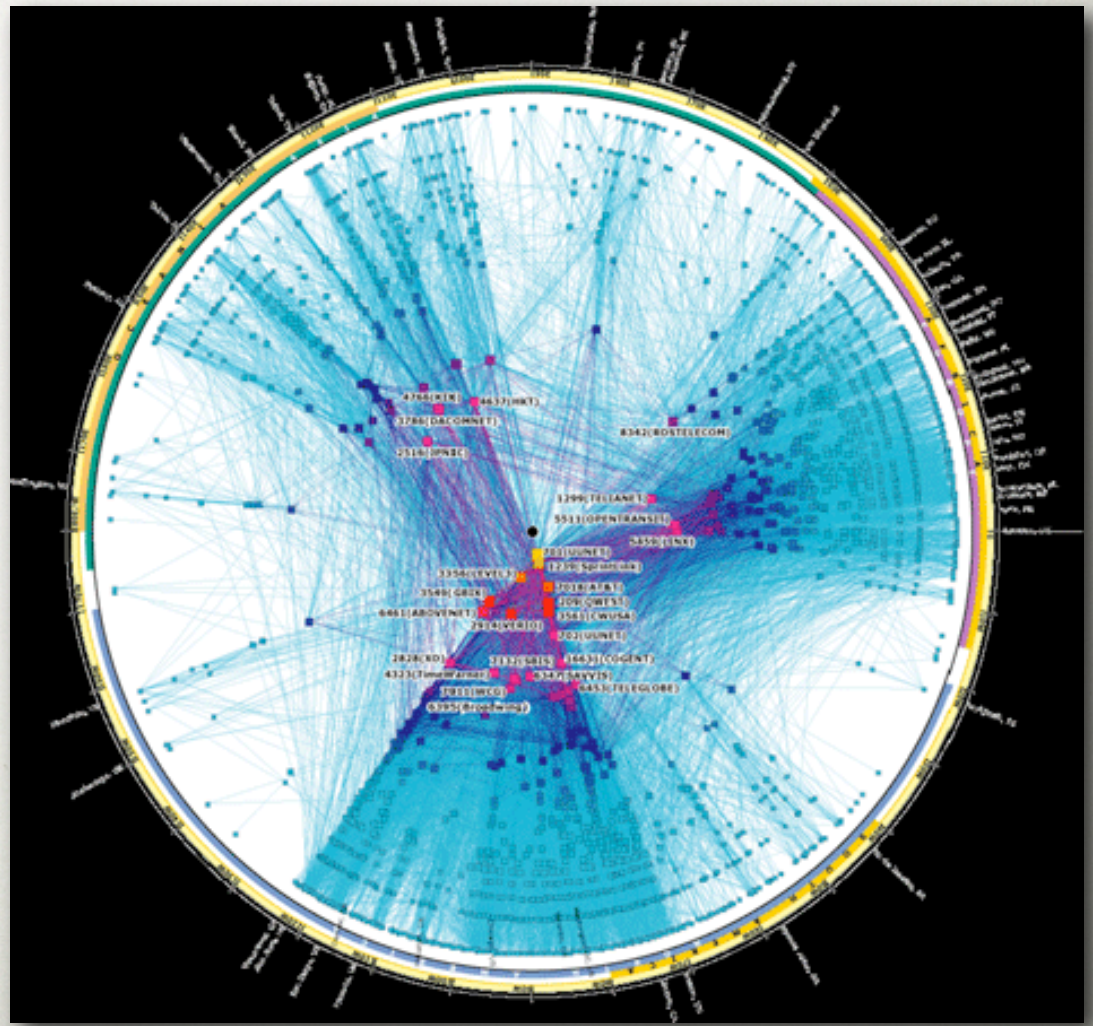
The Bias of Traceroute : Accuracy & Scaling in Internet Mapping

Aaron Clauset
UNM Computer Science at
CAIDA WIT
11 May 2006

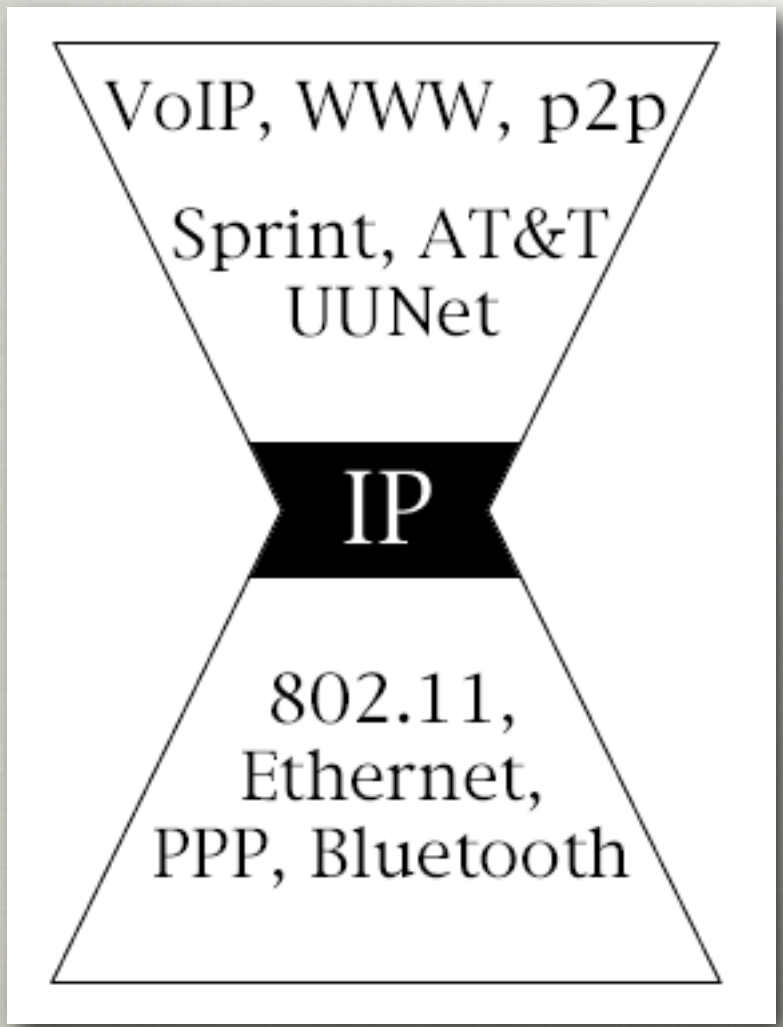
with Cristopher Moore
David Kempe, and
Dimitris Achlioptas

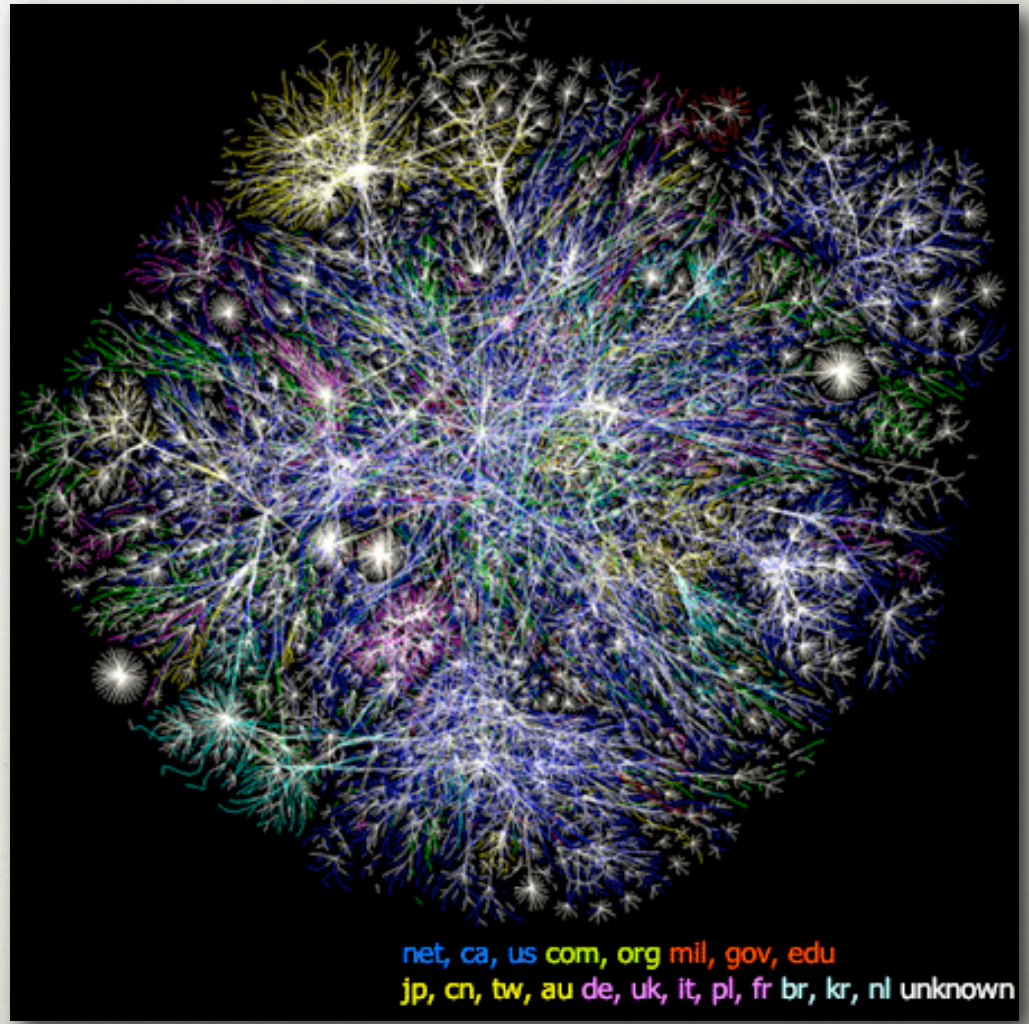
Internet Maps (1)



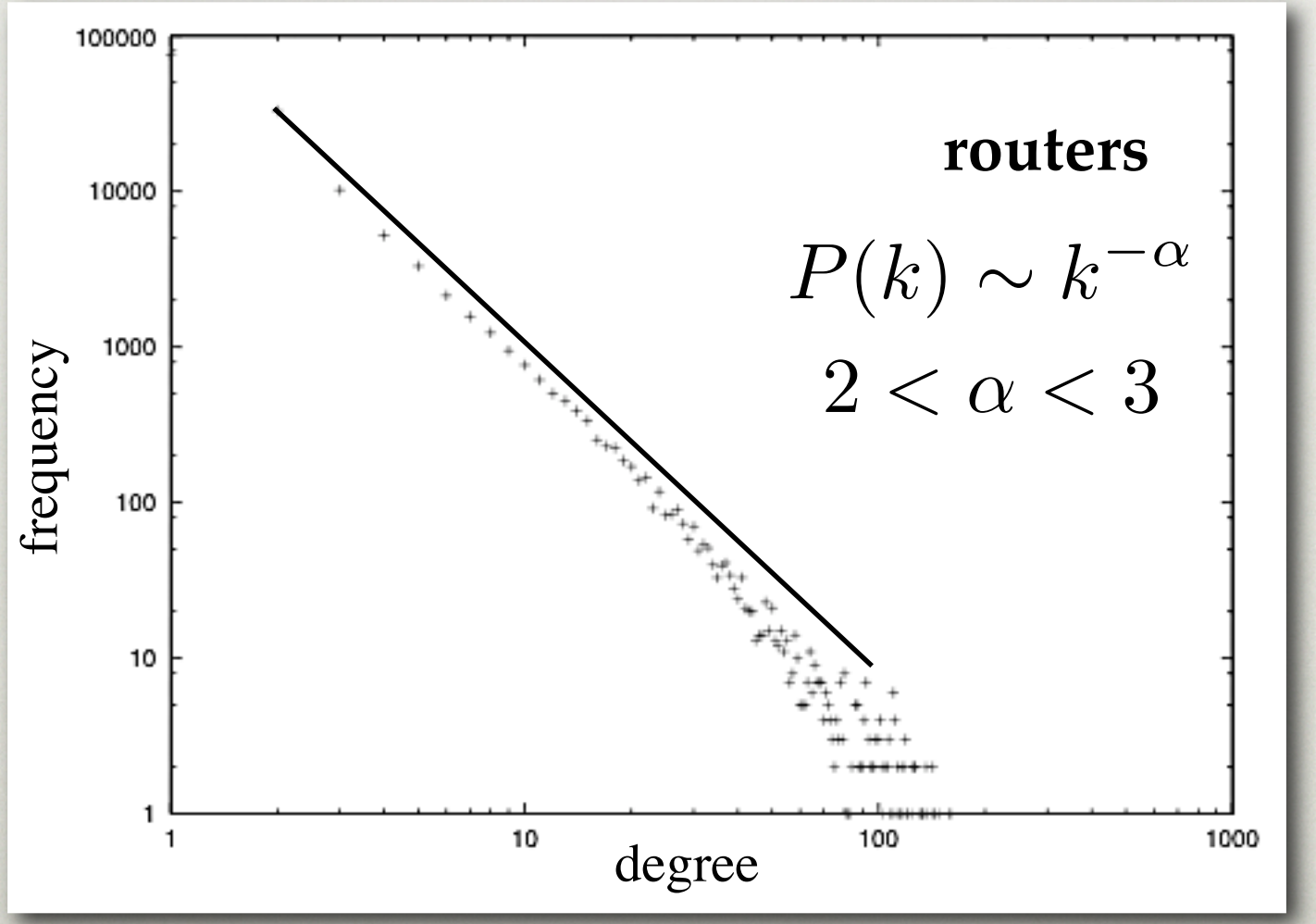


Internet Maps (2)





Internet Maps (3)

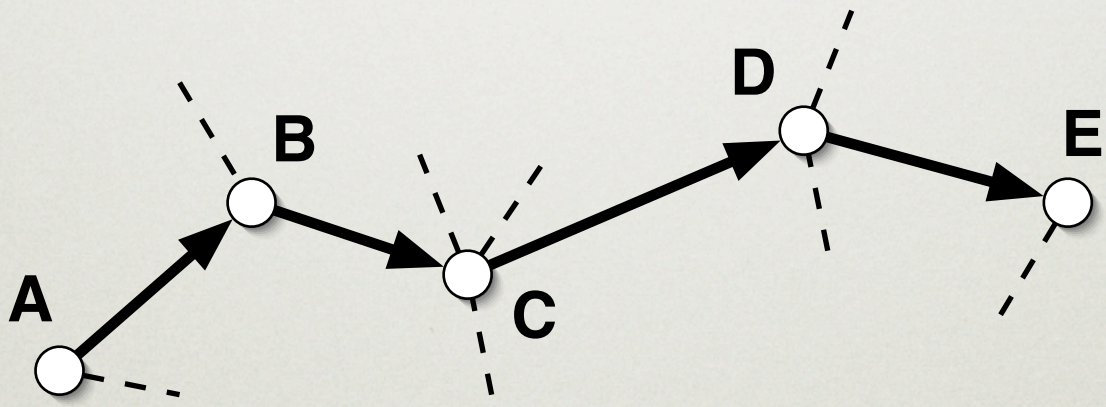


originally shown by [FFF99]

The Problem

Topology must be inferred

- router links not directly observable
- links queried indirectly by paths



- [FFF99] data based on single-source traceroute

Inherent Bias

Lakhina et al., INFOCOM 2003

- Traceroute sampling apparently *biased*
- Distant edges less likely to be observed, and nearby edges are over-represented
- Power-law degree distributions can appear when none exist...
- ... even in Erdös-Rényi random graphs!
- Our first result **confirms this analytically.**

Random Graphs

E-R random graphs

$$G(n, p = c/n)$$

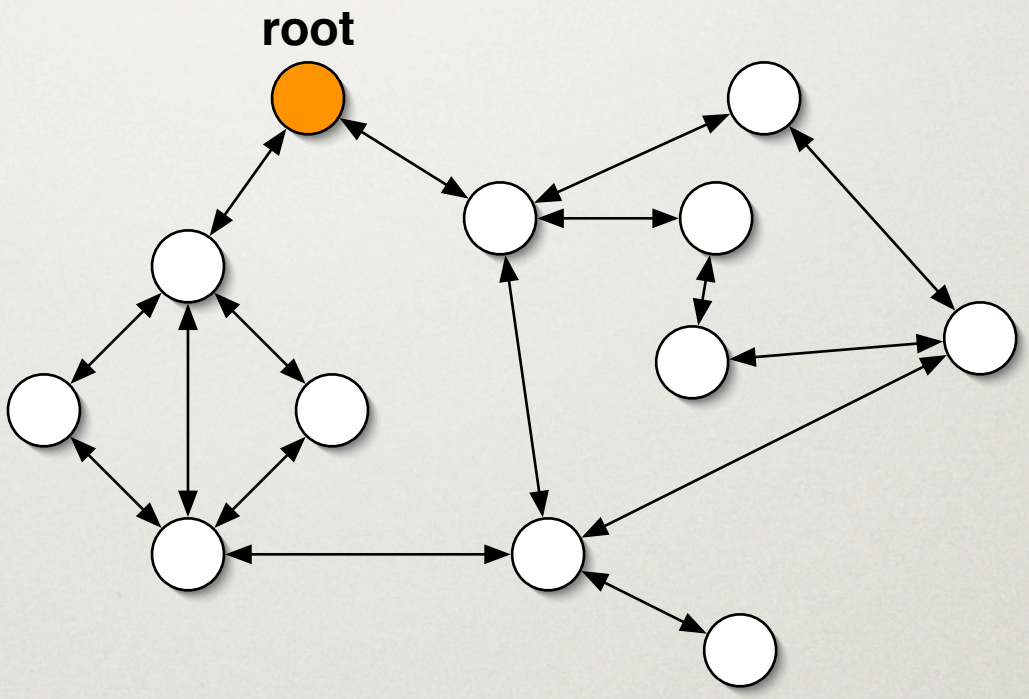
- n vertices; each pair connected with probability p
- Degree distribution $P(k)$ is Poisson with mean c
- For $c > 1$, $\Theta(n)$ vertices form a giant component.

Some Perspective

- Random graphs are totally unrealistic models of anything **real**
- But, random structures are good **null-models**
- If you think some property **P** is *interesting* but a totally random graph also exhibits **P**, then maybe not so interesting...
- Many power laws have trivial explanations, e.g., exponential sampling and certain multiplicative random processes

Back to Traceroute

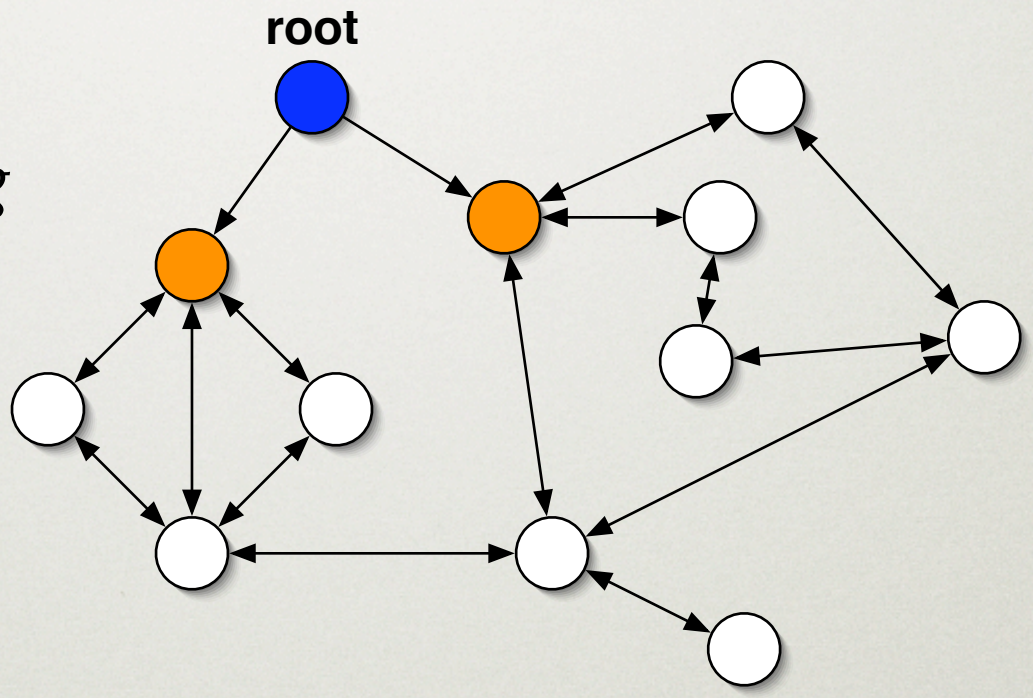
1. choose root



underlying network topology

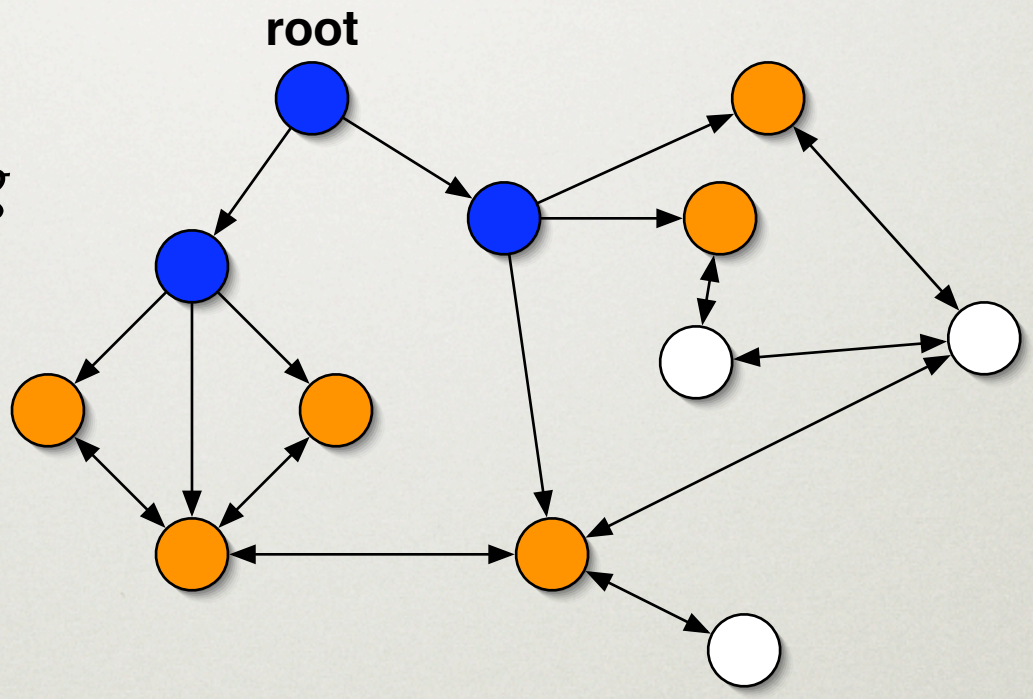
Building a Spanning Tree

1. choose root
2. add edges to
   unknown neighbors
   they become *pending*



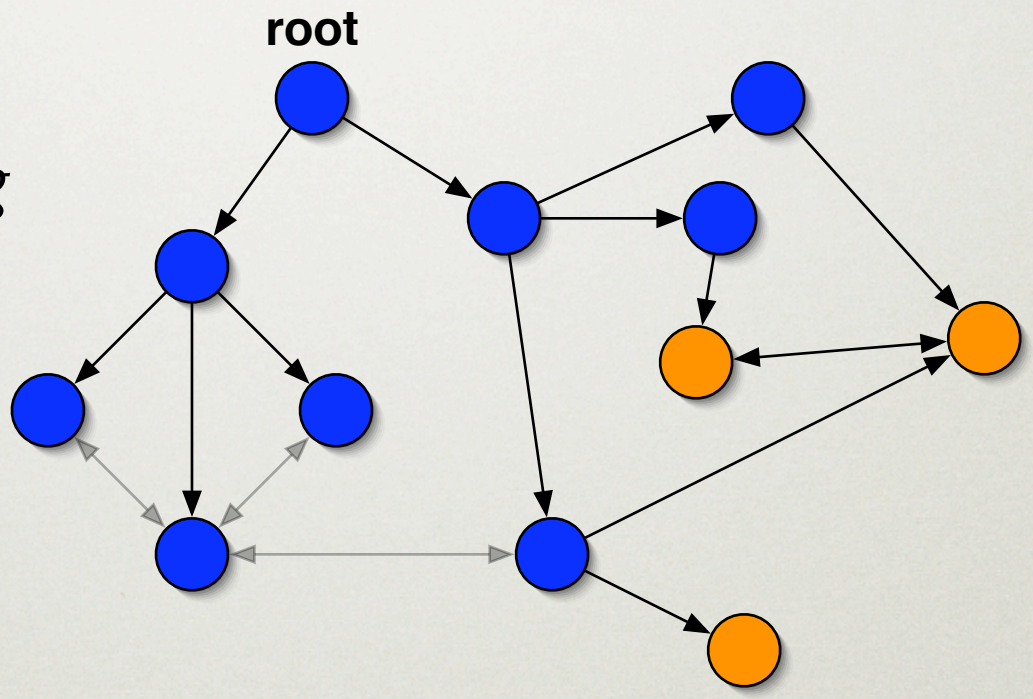
Building a Spanning Tree

1. choose root
2. add edges to
   unknown neighbors
   they become *pending*
3. repeat 2 for each
   pending node;
   it becomes *reached*



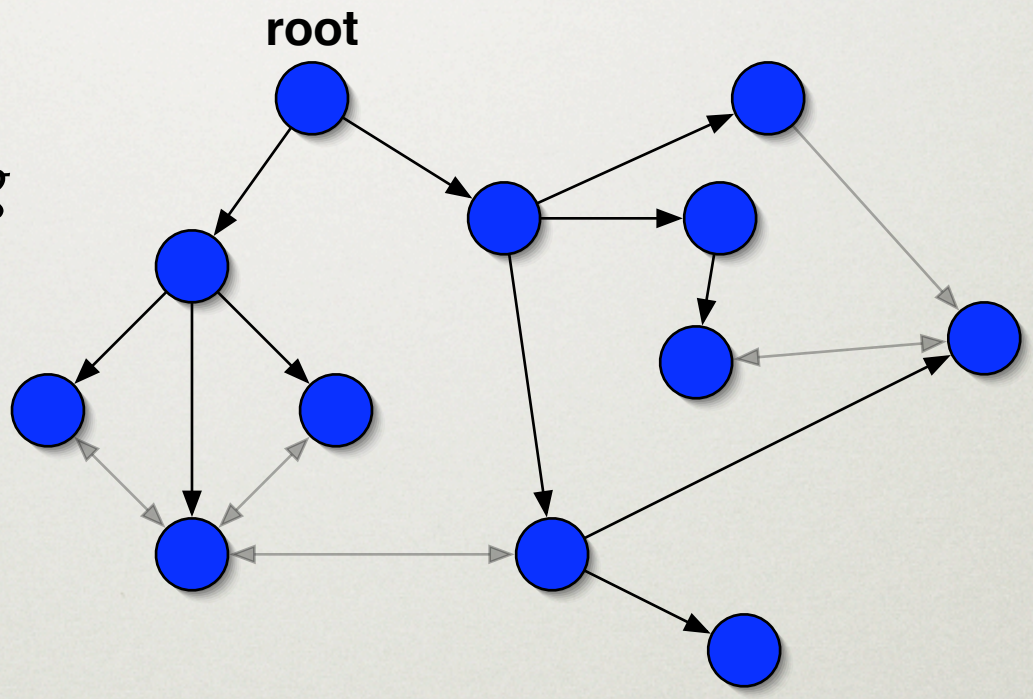
Building a Spanning Tree

1. choose root
2. add edges to
   unknown neighbors
   they become *pending*
3. repeat 2 for each
   pending node;
   it becomes *reached*



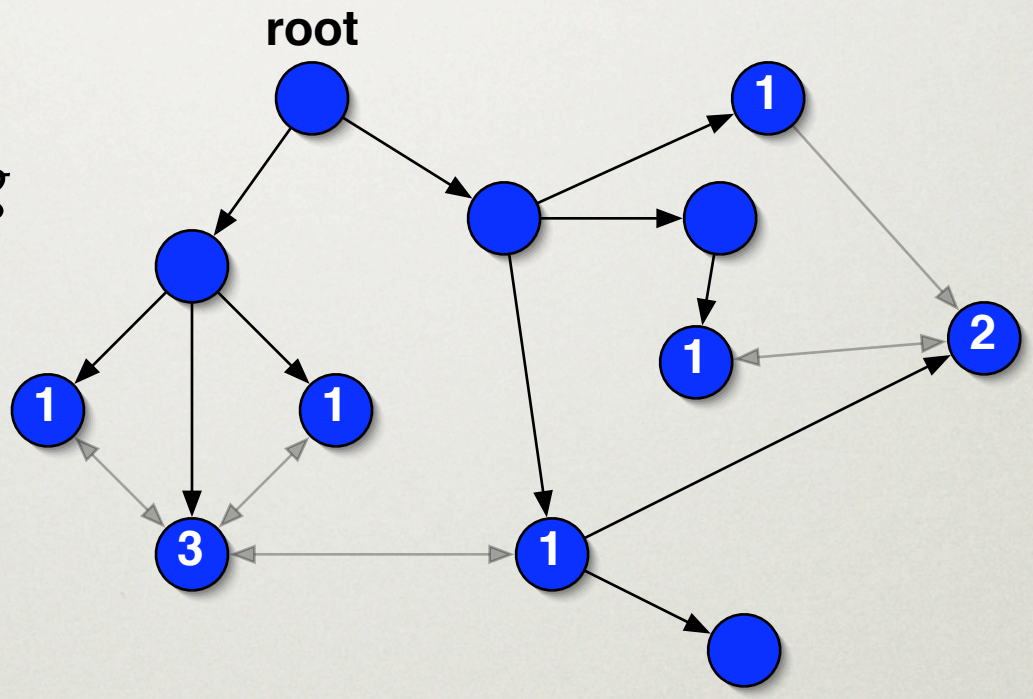
Building a Spanning Tree

1. choose root
2. add edges to
   unknown neighbors
   they become *pending*
3. repeat 2 for each
   pending node;
   it becomes *reached*



Building a Spanning Tree

1. choose root
2. add edges to
   unknown neighbors
   they become *pending*
3. repeat 2 for each
   pending node;
   it becomes *reached*



sampled network topology (**with errors**)

Building a Spanning Tree

label all vertices UNKNOWN, label one PENDING

while there are PENDING vertices

choose* a PENDING vertex v

label v REACHED

for every UNKNOWN neighbor u of v

label u PENDING

add (u, v) to TREE

* e.g., depth-, breadth- or random-first

Analysis

- Let $S(T)$ be number of PENDING vertices at step T
- Let $U(T)$ be number of UNKNOWN vertices at step T
- The expected differences at each step are

$$\mathrm{E}[\ U(T+1) - U(T)\ ] = -pU(T)$$

$$\mathrm{E}[\ S(T+1) - S(T)\ ] = -1 + pU(T)$$

Analysis

- We rescale to $t = T/n, u = U/n, s = S/n$ to obtain

$$\frac{\mathrm{d}u}{\mathrm{d}t} = -cu \qquad \frac{\mathrm{d}s}{\mathrm{d}t} = cu - 1$$

with solutions

$$u(t) = \mathrm{e}^{-ct} \qquad s(t) = 1 - t - \mathrm{e}^{-ct}$$

- Wormald's theorem makes this rigorous; we can predict $S(T)$ and $U(T)$ w.h.p. to within $o(n)$.

Analysis

- At each time t, degree of a PENDING vertex is a Poisson distribution

$$\text{Poisson}(m, k) = \frac{e^{-m} m^k}{k!}$$

with mean $m = cu(t)$

- So, integrate this up to time t_f, at which point every vertex in giant component is REACHED

Analysis

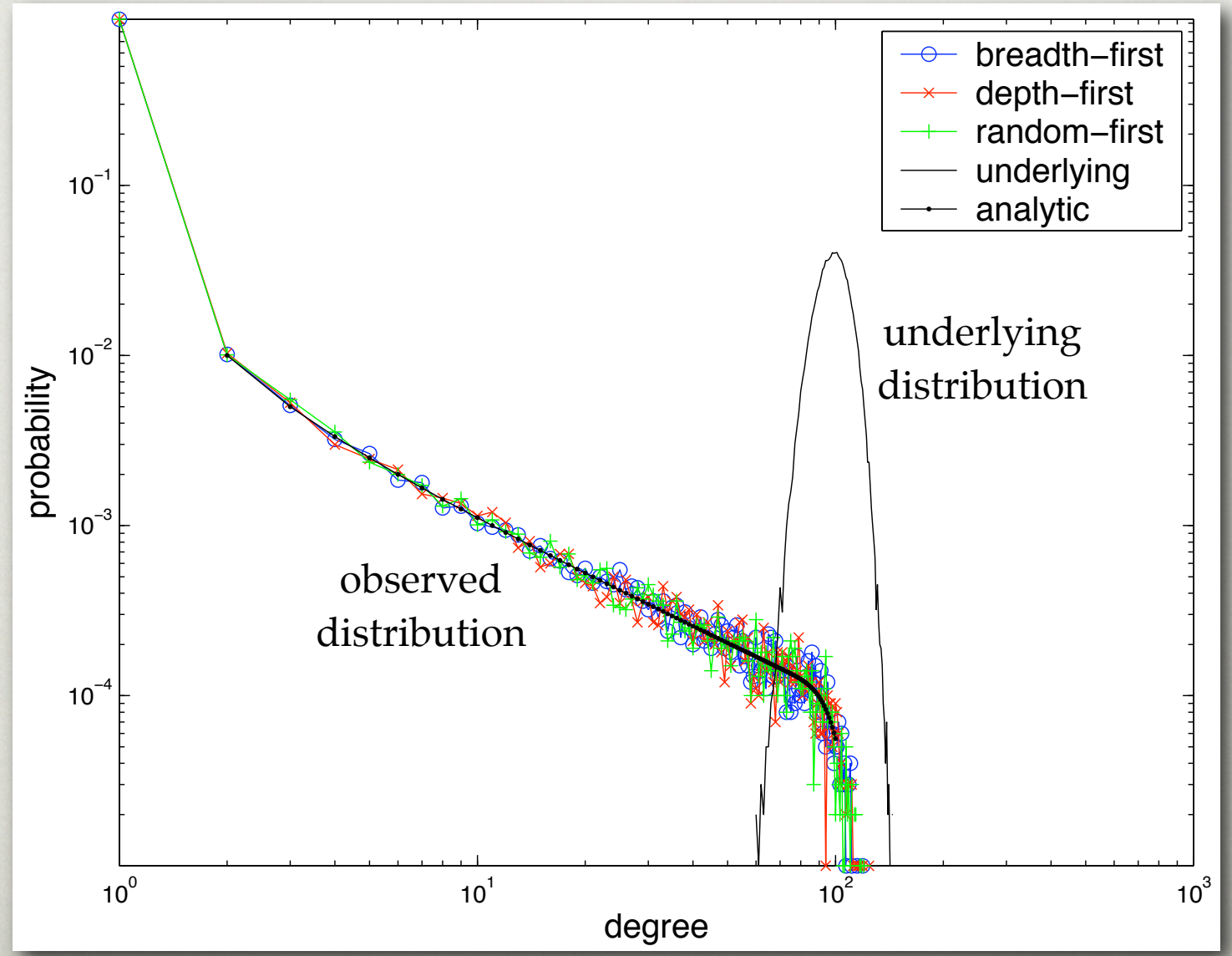
- Thus, averaging over all vertices in TREE

$$P(k+1) = \frac{1}{t_f} \int_0^{t_f} \mathrm{d}t \, \mathrm{Poisson}(cu(t), k)$$

which is a power law with exponent $\alpha = -1$

$$P(k+1) = (1 - o(1)) \frac{\Gamma(k)}{ck!} \sim \frac{1}{ck}$$

Theory & Experiment



C. and Moore, PRL 2005

More Generally

- Generalized process for random graphs with arbitrary underlying degree distributions, e.g.,
  - d-regular $\stackrel{\text{obs}}{\longrightarrow}$ power law!
  - Poisson $\stackrel{\text{obs}}{\longrightarrow}$ power law
  - power law $\stackrel{\text{obs}}{\longrightarrow}$ different power law
- Mathematical crank to get observed distribution

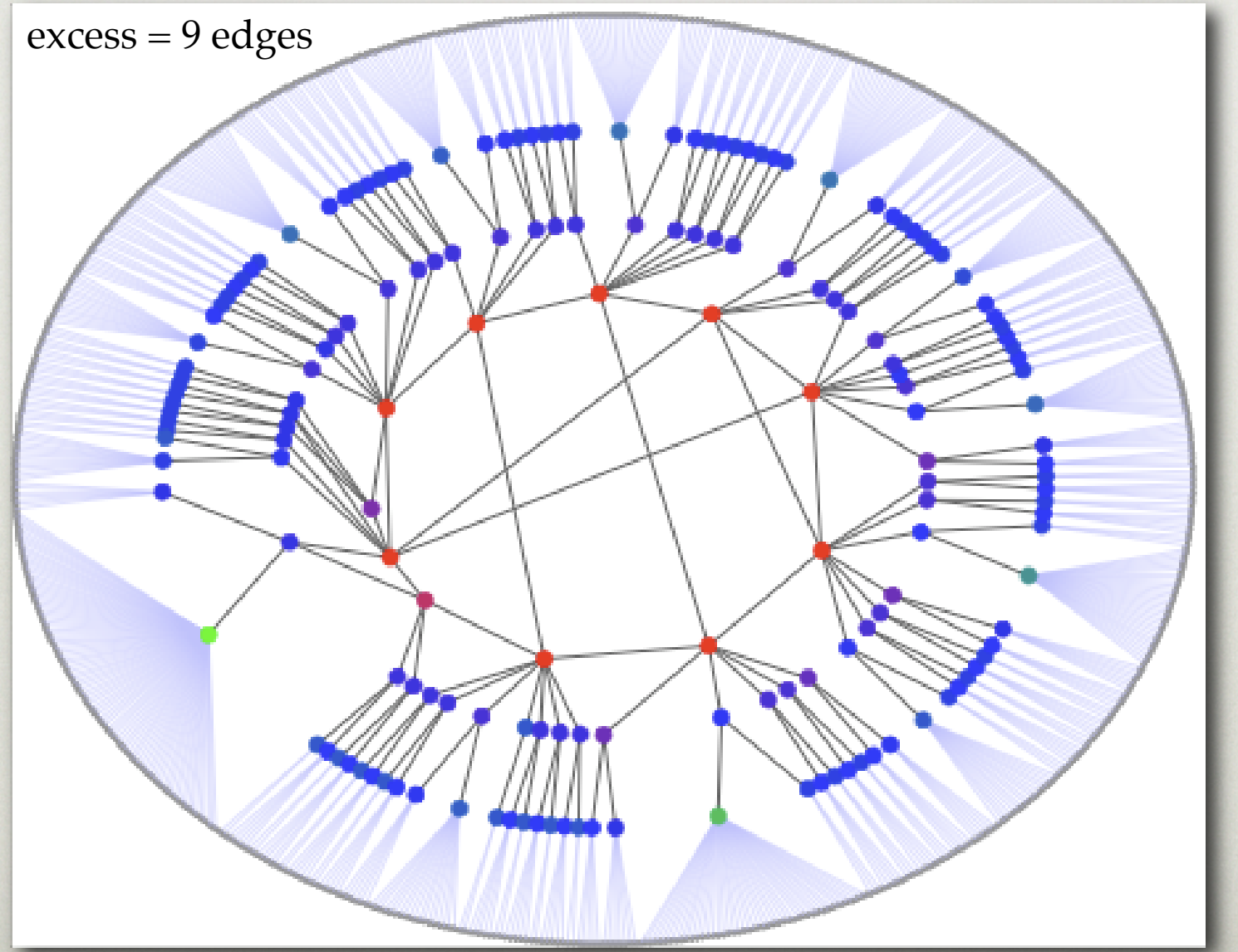
$$g^{\text{obs}}(z) = z \int_0^1 \mathrm{d}t \, g'\left[t - \frac{(1-z)}{g'(1)} \, g'\left(\frac{g'(t)}{g'(1)}\right)\right]$$

C., Moore, Kempe and Achlioptas, STOC 2005

It Gets Worse

- Suppose underlying graph has a power law
- What is *observed* exponent?
- Severity of bias depends on number of **redundant edges** (the ones missed by traceroute)

Low Edge-Redundancy

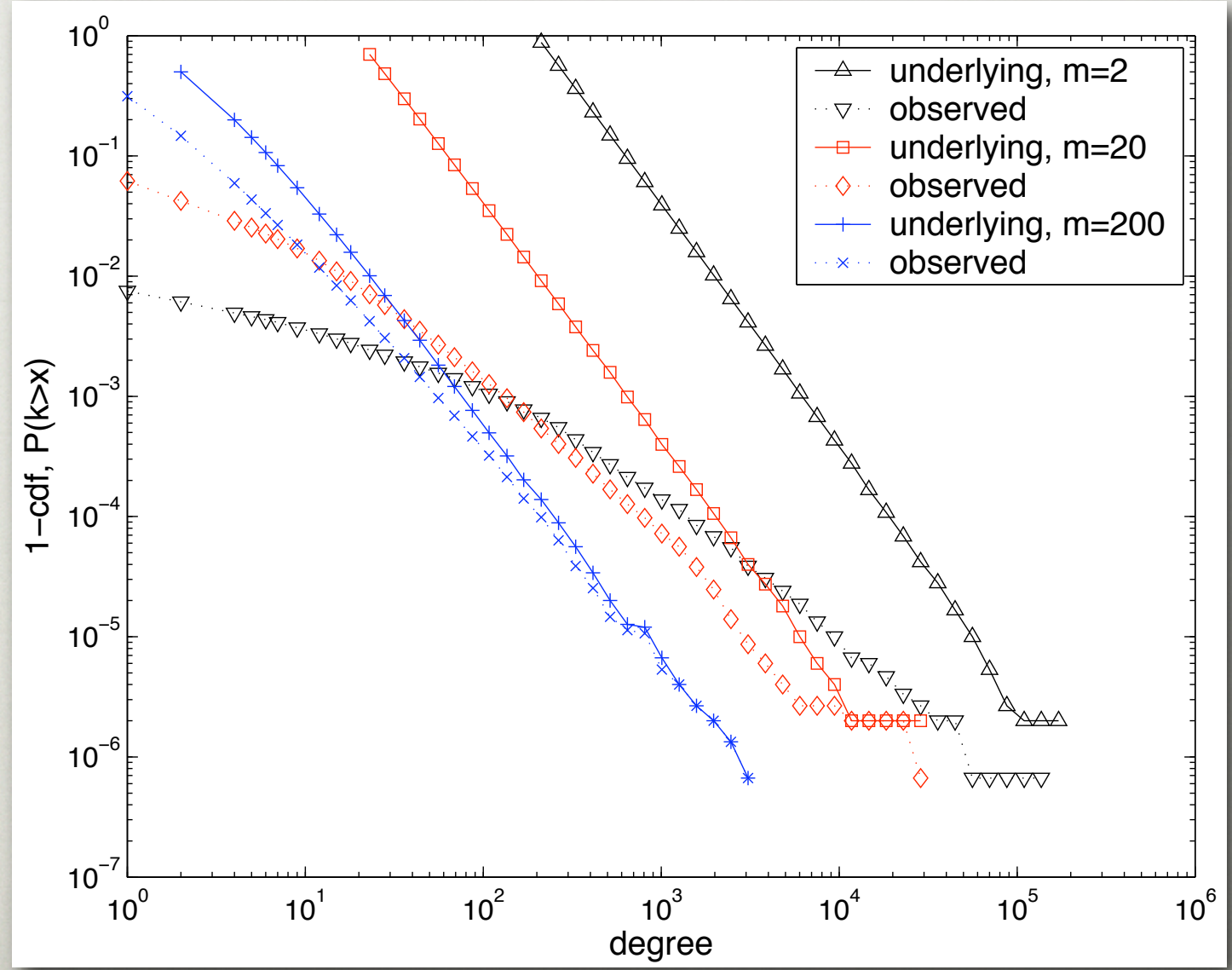


courtesy of D. Alderson

It Gets Worse

- Suppose underlying graph has a power law
- What is *observed* exponent α_{obs}?
- Severity of bias depends on number of **redundant edges** (the ones missed by traceroute)
  - You pick any underlying exponent α,
  - You pick observed exponent $\alpha_{\text{obs}} < \alpha$
  - I can pick mean degree $\langle k \rangle$ so that traceroute gives you α_{obs} from α

It Gets Worse

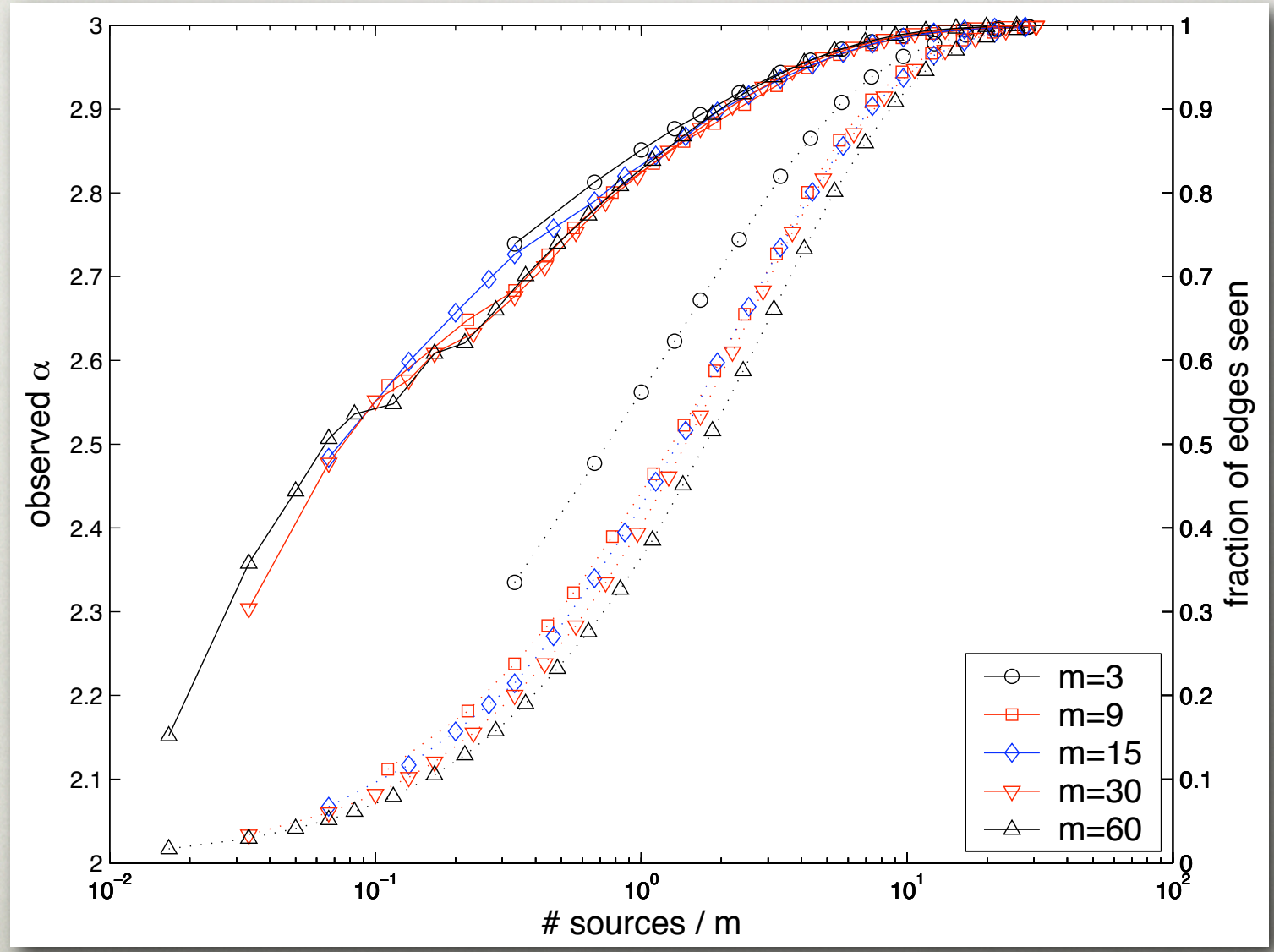


C. and Moore, PRL 2005

And Worse Still...

- Okay, so just use more sources!
- Bias remains severe **until we see almost all edges**
- The marginal value of an additional source is low, but positive!

And Worse Still...



C. and Moore, PRL 2005

Challenges

- **Topological** inferences from traceroute **must** account for its bias (or be heavily caveated)
- But, how to do this? Not yet clear... some ideas:
  - Estimate real marginal value of new sources, for many sources (DIMES?)
  - Estimate mean degree $\langle k \rangle$
  - Use convergence rates to extrapolate
  - Use model of hierarchical organization to estimate the missing edges

Fin