MPRI 2.18.1 (2019/20): Distributed algorithms for networks, 2nd part

Lecture 4: Rumor Spreading in Realistic Networks (2)

Benjamin Doerr, LIX

Outline:
Homeworks
Basics
Random geometric graphs = wireless sensor network
Preferential attachment graphs = social networks

Contents

- Part 1: Basics:
  - a formal definition of the rumor spreading time
  - switching between expectation and “with high probability”
- Part 2: Rumor spreading in random geometric graphs:
  - model for radio networks, connectivity threshold, giant component threshold
  - rumor spreading takes time approx. diameter. Argument: similar to grids
- Part 3: Rumor spreading preferential attachment graphs:
  - model for real networks, low diameter, power-law degree distribution
  - rumor spreading: push takes long, push-pull takes logarithmic time, push-pull without double-calls takes approx. diameter

Homework 3.1

- Consider the following rumor spreading process in the hypercube. The rumor starts in $s = (0, \dots, 0)$ and our only target is to get the rumor to the opposite node $t = (1, \dots, 1)$.
- We run the classic randomized rumor spreading protocol with the following modification: In each round, each informed node selects a random neighbor. If this neighbor is closer to t than itself, then the node actually calls this neighbor. If the selected neighbor is further away from t, then the node does nothing in this round.
- How long does this process need to get t informed?

- Answer: $\Theta(\log n)$ rounds.
- Proof: The proof in the lecture only used movements of the rumor towards the target, so this proof is valid also in our restricted setting.

Homework 3.2

- Consider the classic randomized rumor spreading process in the hypercube started at vertex $s = (0, \dots, 0)$ as regarded in the lecture. Together with the insight from the previous problem, we have shown that with probability at least $1 - n^{-\Omega(1)}$ the rumor traverses a *shortest path* from s to $t = (1, \dots, 1)$ in $O(d)$ time.

- Discuss whether the same property is true for all target vertices t.

- Answer: Not true.
- If v is a neighbor of s, then there is only one shortest path from s to v, namely the edge $\{s, v\}$.
- The time to traverse this edge is geometrically distributed with success probability $1/d$. Hence $\Pr[rumor\ traverses\ \{s,v\}\ in\ time \le kd] = 1 - (1 - 1/d)^{kd} \approx 1 - \exp(-k)$

Basics 1: Definition T_v

- Why now a formal definition, why now discussing other basics?
  - because you might find it helpful in the exam
  - because up to now we have seen enough results where such a definition would have eased the language
    - inverse chronological order of the classic maths lecture/book

- Definition [rumor spreading times T_v]: Consider a round-based rumor spreading process in a graph $G = (V, E)$.
  - For $v \in V$, let T_v denote the number of rounds after which a rumor starting in v for the first time has reached all nodes of G.

- Notes:
  - T_v is a random variable taking values in $\{0,1,2, \ldots, \infty\}$
  - We shall only regard processes with $\Pr[T_v = \infty] = 0$

Towards Defining T_G

- Definition: Let X and Y be random variables. We say that Y stochastically dominates X, written $X \preccurlyeq Y$, if for all $\lambda \in \mathbb{R}$ we have
$$\Pr[Y \leq \lambda] \leq \Pr[X \leq \lambda].$$
    - Note: a very strong sense of "Y is bigger than X"

- **Lemma 4.1:** There is a unique random variable T such that
    1) for all $v \in V$, $T_v \preccurlyeq T$
    2) for any T' satisfying 1) we have $T \preccurlyeq T'$

- Proof: *Go from the probability mass functions to the cumulative distributions, take the minimum, and convert it to a mass function*
    - $F(t) \coloneqq \min\{\Pr[T_v \leq t] \mid v \in V\}$
    - Define T by $\Pr[T = 0] = F(0)$, $\Pr[T = t] = F(t) - F(t-1)$ for $t \geq 1$
    - This T satisfies 1). If some T' satisfies 1), then $F'(t) \coloneqq \Pr[T' \leq t]$ is at most $\Pr[T_v \leq t]$ for all $v \in \mathrm{V}$ by 1), hence $F' \leq F$, and thus $T \preccurlyeq T'$.

Rumor Spreading Time T_G of G

- Definition: The unique T from Lemma 4.1 is called the rumor spreading time of G and denoted by T_G

- Trivialities (which are the reasons for this definition):
  - $\Pr[T_G \leq t] \geq p$ is equivalent to saying that regardless of where the rumor starts, after t rounds with probability at least p all vertices are informed.
  - $\Pr[T_G \geq t] \geq p$ is equivalent to saying that there is a vertex $v \in V$ such that the rumor spreading process started in v with probability at least p has not informed all vertices earlier than after t rounds.

- Note: $E[T_G] \geq \max\{E[T_v] | v \in V\}$
  - I do not know if we have equality

Basics 2: Probability vs. Expectation

- So far, we have shown results for either $E[T_G]$ or we have shown that T_G is at most or at least some value with some (often high) probability.
  - we have done whatever was more convenient ☺

- Lemma [Prob. → E]: If $\Pr[T_G \leq t] \geq p$, then $\max\{E[T_v] | v \in V\} \leq t/p$
  - [provided the rumor spreading process is memory-less (things in the current round only depend on who is informed at the beginning of the round) and extra informed nodes cannot harm.

- Proof: Restart argument
  - Let v be the node initially informed. With prob. p, all nodes are informed after t rounds. If not, pretend that only v is informed and do another set of t rounds. …
  - The expected number of trials (sets of t rounds) is p^{-1}, consequently, $E[T_G] \leq t/p$.

Basics 2: Probability vs. Expectation

- Lemma [E → Prob.]: For all $\lambda \in \mathbb{N}$, $\Pr[T_G \leq 2\lambda \max\{E[T_v] | v \in V\}] \geq 1 - 2^{-\lambda}$
  - [provided the rumor spreading process is memory-less (things in the current round only depend on who is informed at the beginning of the round) and extra informed nodes cannot harm.

- Proof: Markov's inequality and restart argument
  - Let v be the node initially informed.
  - With prob. at least 0.5, all nodes are informed after $2E[T_v]$ rounds (Markov's inequality).
  - The probability that λ such trials fail, is at most $2^{-\lambda}$. Hence $\Pr[T_v \leq 2\lambda E[T_v]] \geq 1 - 2^{-\lambda}$.

Part 2: Rumor Spreading in Wireless Sensor Networks

- *Wireless sensor network*: spatially distributed system of *sensor nodes* used for collecting data in a large area (often difficult to access)

- Sensor node
  - radio transmitter and receiver: communication with near-by nodes
  - sensors: collect data
  - microcontroller
  - battery: crucial for the life-time of the system

- Typical characteristics: Simple and cheap, so you don’t care about setting up a clever network, but you distribute the nodes in a simple fashion (randomly)
  - multi-hop network: communication is via intermediate nodes, so that the communication range can be kept small (saves energy)

Mathematical Model

- *Random geometric graph* (RGG) $G(n, r)$:
  - Nodes: $v_1, \dots, v_n \in [0,1]^2$ be chosen uniformly at random
  - Edges: $\{v_i, v_j\}$ is an edge if and only if $d(v_i, v_j) \leq r$

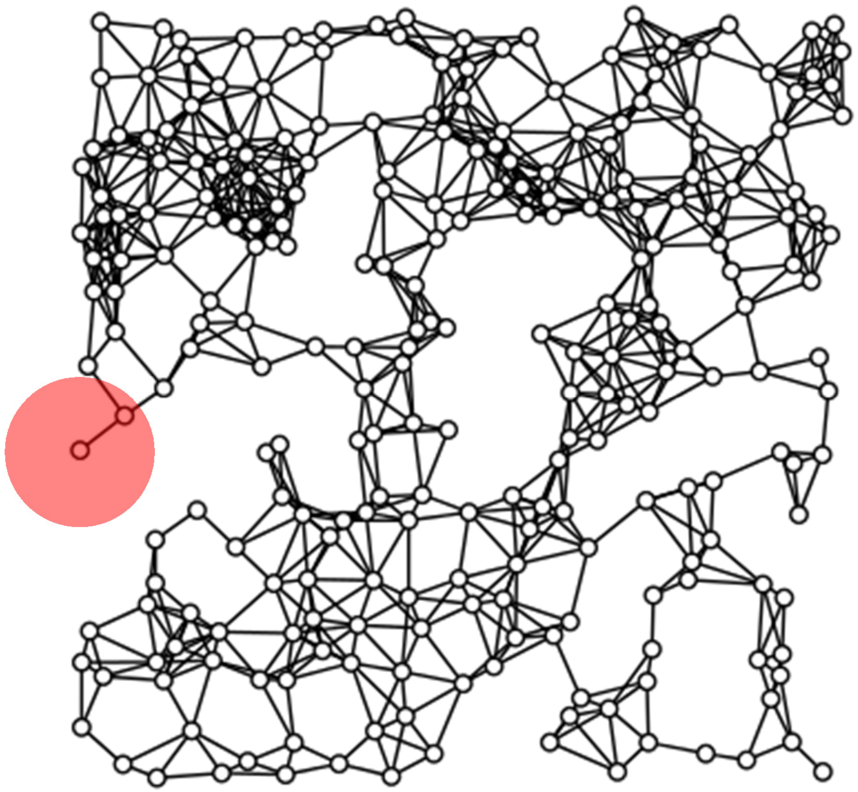


Fig.: A random geometric graph $G(256, 0.1)$ [Wikipedia]

Structural Properties of RGG

- Theorem [Penrose (2003)]: If $r = (\pi n)^{-0.5}\,(\ln(n) + \alpha)^{0.5}$, then
$$\lim_{n\to\infty} \Pr[G(n,r) \text{ is connected}] = \exp(-e^{-\alpha})$$
- Theorem [Penrose (2003)]: There is a constant c such that
  - if $r = \alpha n^{-0.5}$ with $\alpha > c$, then $G(n,r)$ has a "giant component", that is, a connected component consisting of $\Theta(n)$ vertices; the proportion of this component tends to one for $\alpha \to \infty$
  - if $r = \alpha n^{-0.5}$ with $\alpha < c$, then all components have logarithmic size

- Implications for wireless sensor networks: 3 regimes
  - highly disconnected: $r < cn^{-0.5}$ - all nodes can talk to only few others
  - giant component: There a large connected subnetwork, but some nodes are not part of it
  - connected: $r \geq (1+\varepsilon)(\pi n)^{-0.5} \ln^{0.5}(n)$

Weaker Analysis of RGGs: Well-Connected Regime

- Well connected regime: $r \geq Cn^{-0.5}\ln^{0.5}(n)$ for C large enough.

- A typical computational geometry argument shows that G is connected.

- Partition the unit square $[0,1]^2$ into $\Theta(1/r^2)$ squares of side length $l = r/(2\sqrt{2}) = \Theta(r)$
- call two squares adjacent if they touch (vertically, horizontally, diagonally)
- Note: *vertices in adjacent squares are adjacent in the RGG* (choice of l)

- Claim: Each square S contains a vertex!
  - $\Pr[V \cap S = \emptyset] = (1-l^2)^n \leq \exp(-l^2 n) \leq \exp(-(C^2/8)\ln(n)) = n^{-C^2/8}$
  - Union bound: If $C^2 > 8$, then w.p. $1 - n^{-\Theta(1)}$ all squares contain a vertex

- Result: RGG connected and has diameter $O(r^{-1})$
  - sensor view: sensors observe their square and all adjacent ones

Rumor Spreading in Well-Connected RGGs (1)

- Properties of well connected RGGs:
  - diameter $\Theta(1/r)$ [lower bound follows trivially from geometry]
  - all degrees $\Theta(nr^2)$ [expected degree + strong concentration]

- Degree-diameter bound: Rumors spread in time $O(nr)$

- Observation: This bound becomes weaker for larger r ☹
  - if this was true, then a larger communication power would reduce the rumor spreading speed

Rumor Spreading in Well-Connected RGGs (2)

- Let's work in the discrete setting: grid of squares of side-length $l = r/2\sqrt{2}$!
- Each square contains the same expected number $E \geq \frac{C^2}{8}\ln(n)$ of vertices.
- Chernoff bound: with probability $1 - n^{-\Omega(1)}$, each square has $\Theta(E)$ vertices
- Hence each vertex has $\Theta(E)$ neighbors in *each* adjacent square and has $\Theta(E)$ neighbors in total
  - → constant probability to call someone in an adjacent square
- Conditional on this, the rumor spreading process is very similar to 2D grids:
  - expected constant time to call from one square to a neighboring one
  - "follow the path" argument (in the grid of squares): $\Theta(\max\{r^{-1}, \log n\})$ rounds until each square has an informed node
  - additional $O(\log n)$ rounds for each square to inform itself
  - → rumor spreading time is $\Theta(diam(G) + \log n)$ w.h.p.

Detour: Very Disconnected Regime

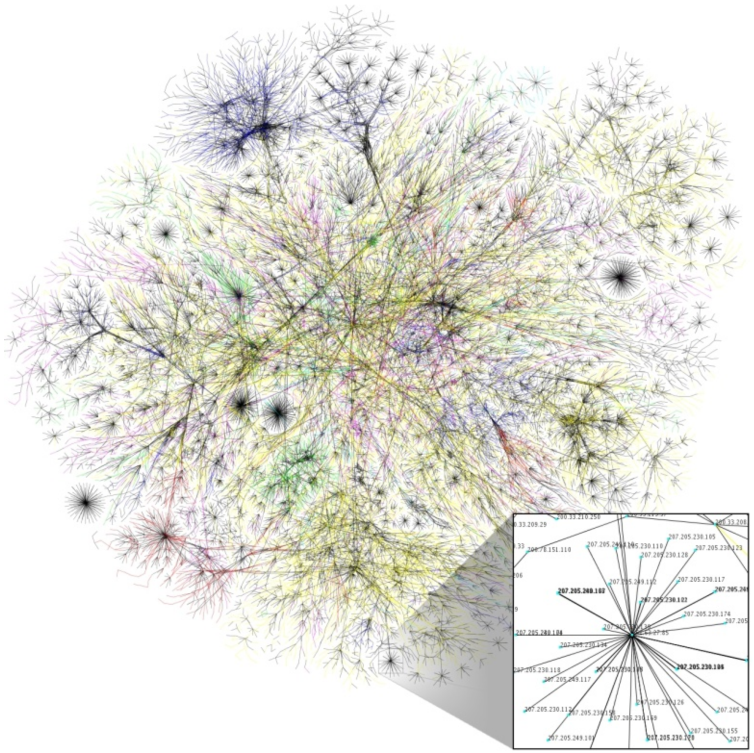
- **Very disconnected regime:** $r = o(n^{-0.5})$. First moment method shows that with high probability, almost all nodes are isolated
  - Probability that x_i and x_j are neighbors: $\approx \pi r^2 = o(n^{-1})$
    - x_j has to land in the radius-r ball around x_i
  - expected number of neighbors of x_i: $n \cdot o(n^{-1}) = o(1)$
  - Markov's inequality: Probability that x_i has a neighbor = $o(1)$
  - expected number of non-isolated vertices: $o(n) =: f(n)$
  - Markov: Probability that more than $f(n) \cdot (n/f(n))^{1/2} = o(n)$ vertices are not isolated is $(f(n)/n)^{1/2} = o(1)$ → with high probability, almost all nodes are isolated!

- **Sensor view:** If you assume that the sensors also have a visibility range of r, then at most an area of size $n \cdot \pi r^2 = o(1)$ is observed by the sensors ☹

Part 3: Rumor Spreading in Social Networks, Real-World Graphs

- <u>“Real-world graph”</u>:
  - airports connected by direct flights
  - scientific authors connected by a joint publication
  - Facebook users being “friends”

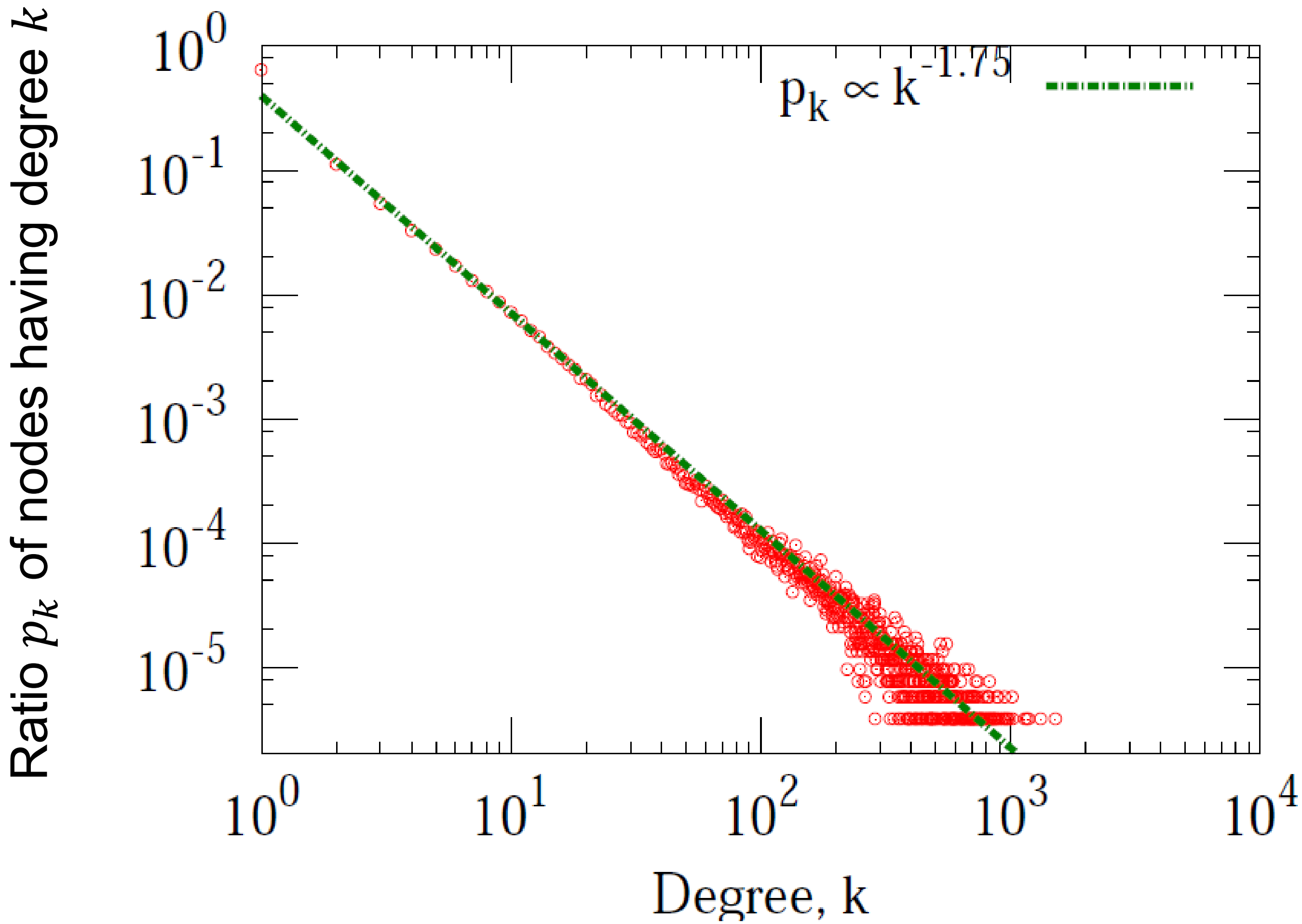
- Big insight of the last 20 year: Real-world graphs have very special properties!
  - small diameter
  - non-uniform degree distribution: few nodes ot high degree (“hubs”), many nodes of small degree
    - power law: number of nodes of degree d is proportional to $d^{-\beta}$ [β a constant, often between 2 and 3]
  - clustering: more triangles than a random graph $G(n, p)$



Map of the internet
[source: the opte project]

Example 1

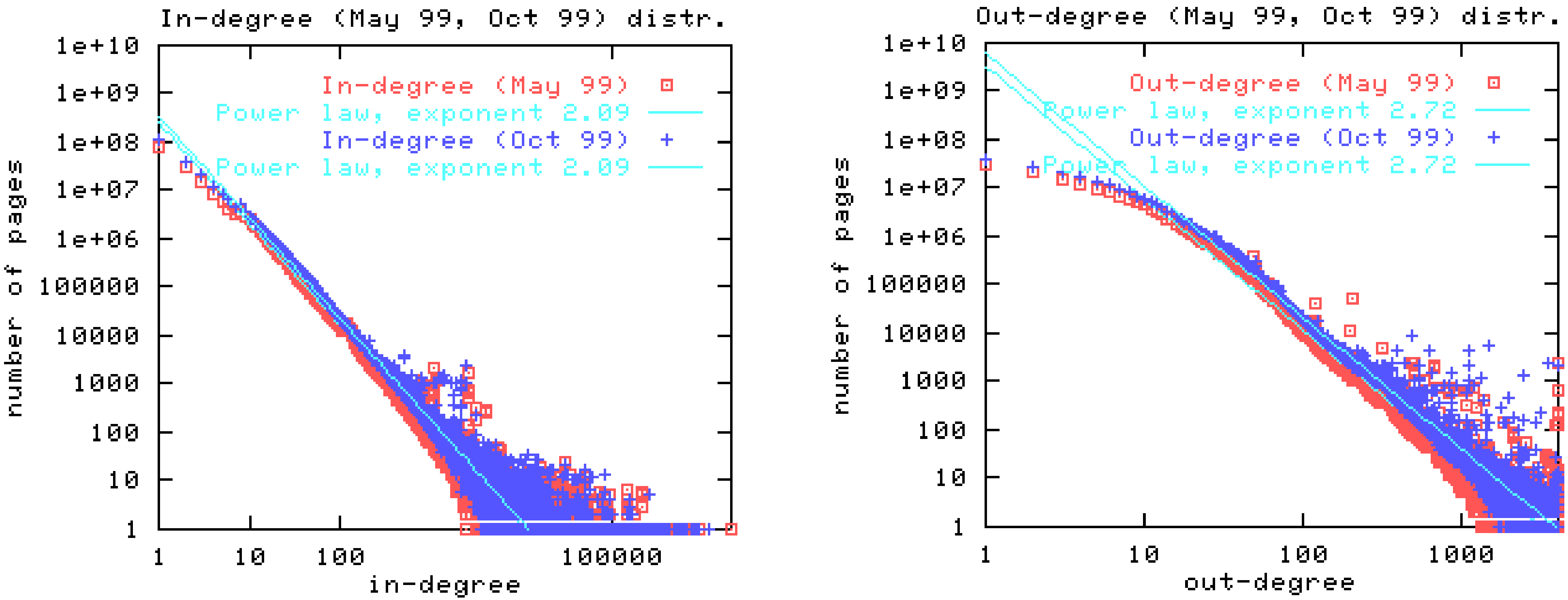
Degree distribution of the Flickr social network with (at that time) $n = 600{,}000$ nodes and $m = 3{,}500{,}000$ edges



[Leskovec et al. (2008)]

Example 2

In- and out-degree of the web-graph (web-pages linking to others)



[Broder et al. (2000)]

Examples 3-14

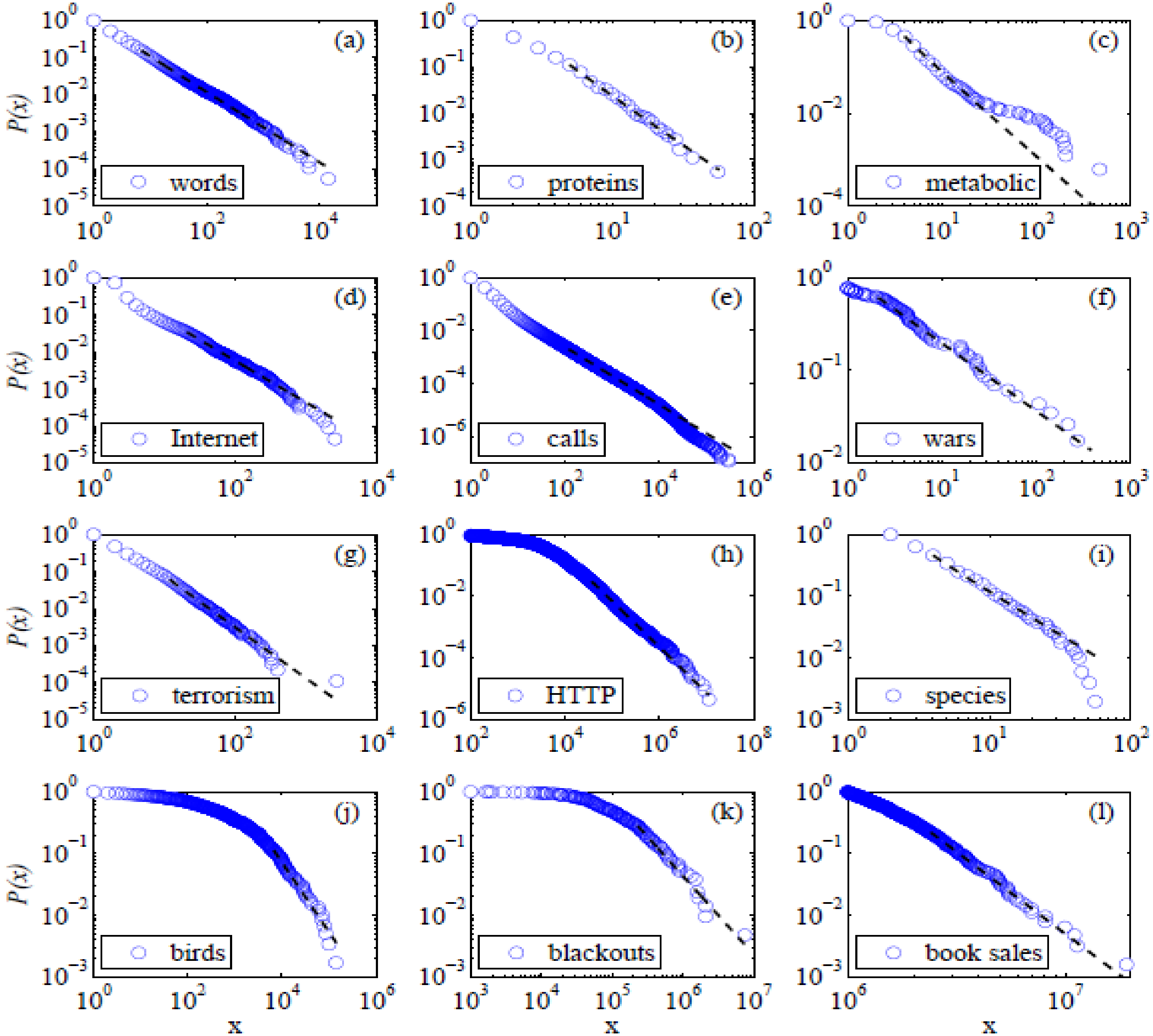


FIG. 6.1. *The cumulative distribution functions $P(x)$ and their maximum likelihood power-law fits for the first twelve of our twenty-four empirical data sets. (a) The frequency of occurrence of unique words in the novel Moby Dick by Herman Melville. (b) The degree distribution of proteins in the protein interaction network of the yeast S. cerevisiae. (c) The degree distribution of metabolites in the metabolic network of the bacterium E. coli. (d) The degree distribution of autonomous systems (groups of computers under single administrative control) on the Internet. (e) The number of calls received by US customers of the long-distance telephone carrier AT&T. (f) The intensity of wars from 1816–1980 measured as the number of battle deaths per 10 000 of the combined populations of the warring nations. (g) The severity of terrorist attacks worldwide from February 1968 to June 2006, measured by number of deaths. (h) The number of bytes of data received in response to HTTP (web) requests from computers at a large research laboratory. (i) The number of species per genus of mammals during the late Quaternary period. (j) The frequency of sightings of bird species in the United States. (k) The number of customers affected by electrical blackouts in the United States. (l) The sales volume of bestselling books in the United States.*

Aaron Clauset, Cosma Rohilla Shalizi, and M. E. J. Newman (2009)

Example 15-26

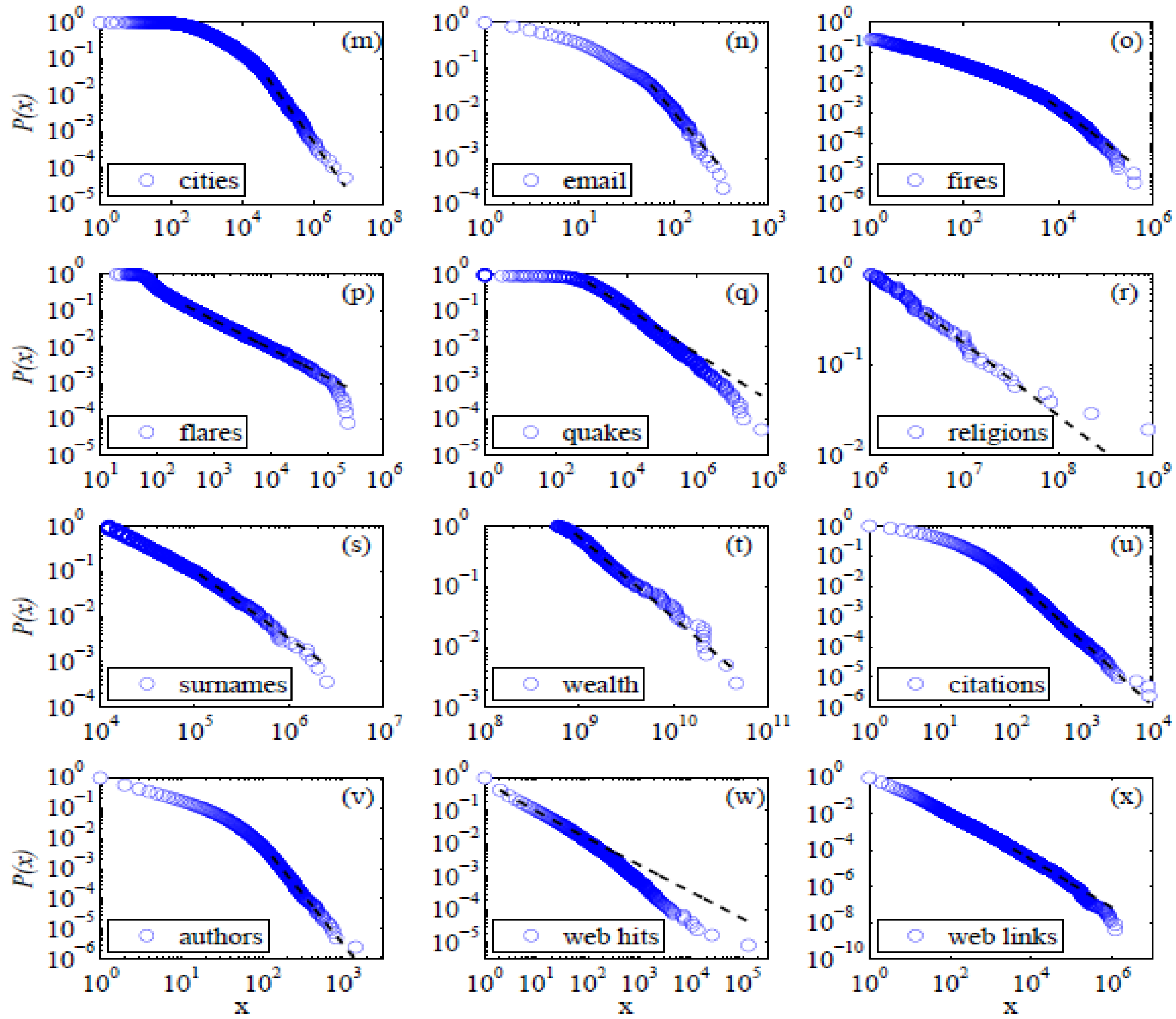


FIG. 6.2. The cumulative distribution functions $P(x)$ and their maximum likelihood power-law fits for the second twelve of our twenty-four empirical data sets. (m) The populations of cities in the United States. (n) The sizes of email address books at a university. (o) The number of acres burned in California forest fires. (p) The intensities of solar flares. (q) The intensities of earthquakes. (r) The numbers of adherents of religious sects. (s) The frequencies of surnames in the United States. (t) The net worth in US dollars of the richest people in America. (u) The numbers of citations received by published academic papers. (v) The numbers of papers authored by mathematicians. (w) The numbers of hits on web sites from AOL users. (x) The numbers of hyperlinks to web sites.

Aaron Clauset, Cosma Rohilla Shalizi, and M. E. J. Newman (2009)

Preferential Attachment (PA) Graphs

- Barabási, Albert (Science 1999):
  - try to explain why social networks could look like this
  - suggest a model for real-world graphs: preferential attachment (PA)

- Preferential attachment paradigm:
  - networks evolve over time
  - when a new node enters a network, it chooses at random a constant number m of neighbors
  - random choice is not uniform, but gives preference to “popular” nodes
    - probability to attach to node x is proportional to the degree of x

- Once made truly precise (by Bollobás and Riordan (2004)), the PA paradigm defines a very useful random graph model (“PA graphs”)
  - Today: One of the most used models for real-world networks

Precise Definition of PA Graphs

- Preferential attachment graph G_m^n
  - n: number of vertices, vertex set $[1..n]$
  - m: density parameter

- The PA graph $\mathrm{G}^n := G_m^n$ is recursively defined:
  - G^1: 1 is the single vertex that has m self-loops
  - G^n: Obtained from adding the new vertex n to G^{n-1}
    - one after the other, the new vertex n chooses m neighbors
    - the probability that some vertex x is chosen, is
      - proportional to the current degree of x, if $x \neq n$
      - proportional to "$1 +$ the current degree of x", if $x = n$ (self-loop probability takes into account the current edge starting in n)

Properties of PA Graphs*

- diameter $\Theta(\log n \,/\, \log \log n)$: less than logarithmic despite $\Theta(n)$ edges!
  - $G(n,p)$ with $p = \Theta(1/n)$: far from connected
  - random regular graphs, k-out graphs: diameter $\Theta(\log n)$

- power law degree distribution: For $d \leq n^{1/5}$, the expected number of vertices having degree d is proportional to d^{-3}.

- clustering coefficient = roughly the probability that two neighbors of some node are connected by an edge
  - PA graphs: $\approx n^{-0.75}$ experimentally
  - real-world graphs: typically constant
  - $G(n, p = \Theta(1/n))$: $\Theta(1/n)$

*All statements hold “with high probability” (whp), that is, with prob. $1 - o(1)$

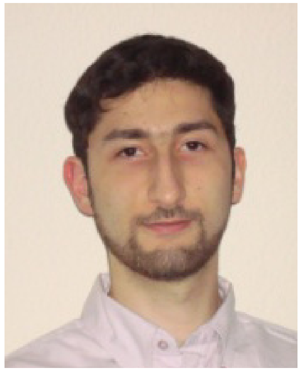
How Fast is Rumor Spreading in Social Networks?

“There's a plane in the Hudson. I'm on the ferry going to pick up the people. Crazy.”

Jan. 15, 2009, 12:36pm: Janis Krums posts the first photo of U.S. Airways flight 1549 on Twitter from his iPhone. Thirty-four minutes later, MSNBC interviewed him live on TV

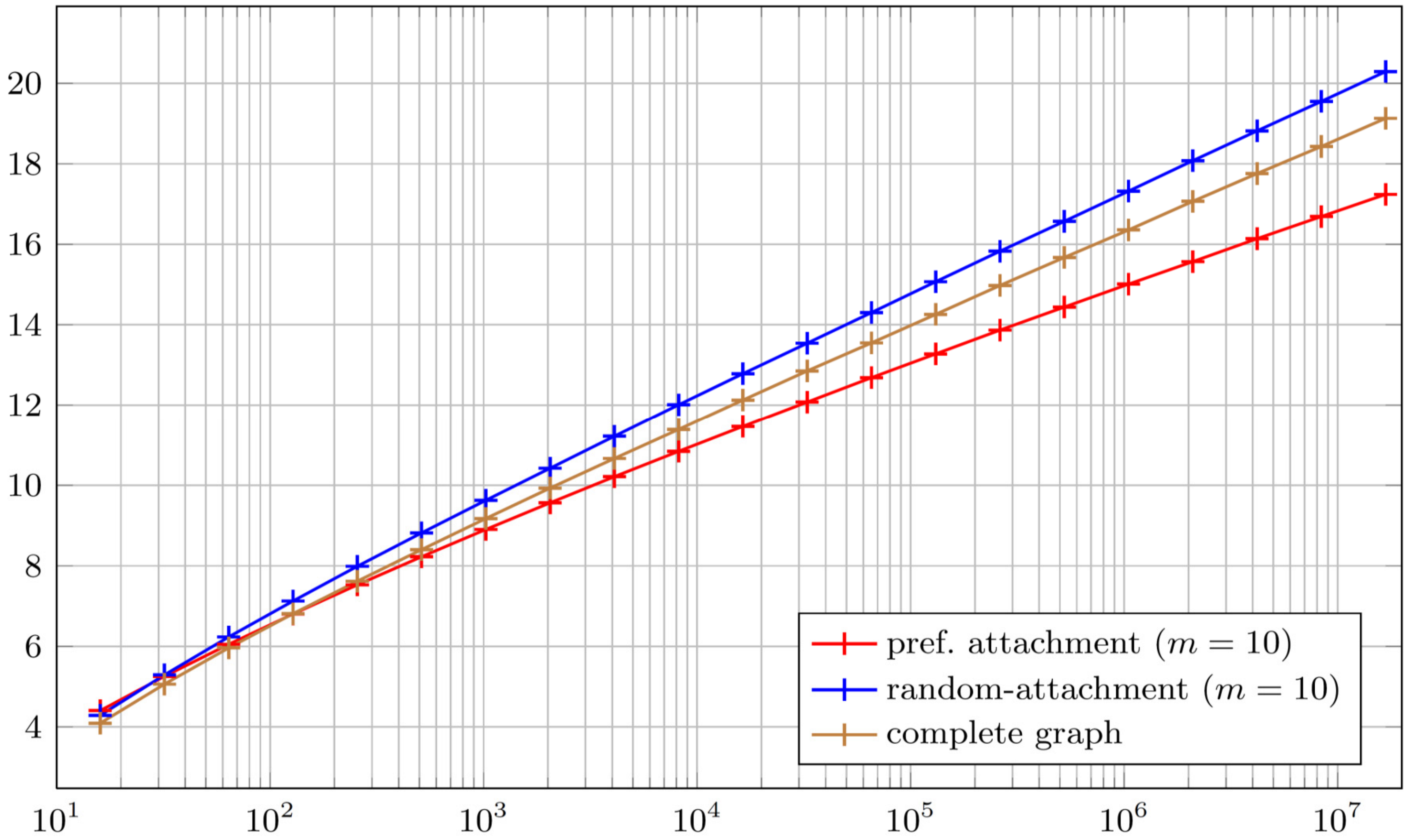
Rumor Spreading in PA Graphs G_m^n

- Chierichetti, Lattanzi, Panconesi (2009):
  - Classic "push" rumor spreading: n^α rounds (α a small constant) with constant probability do not suffice to inform all nodes
  - If both informed and uninformed nodes call random neighbors to spread or seek rumors (push-pull protocol), then $O((\log n)^2)$ rounds inform a PA graph G_m^n, $m \geq 2$, whp.



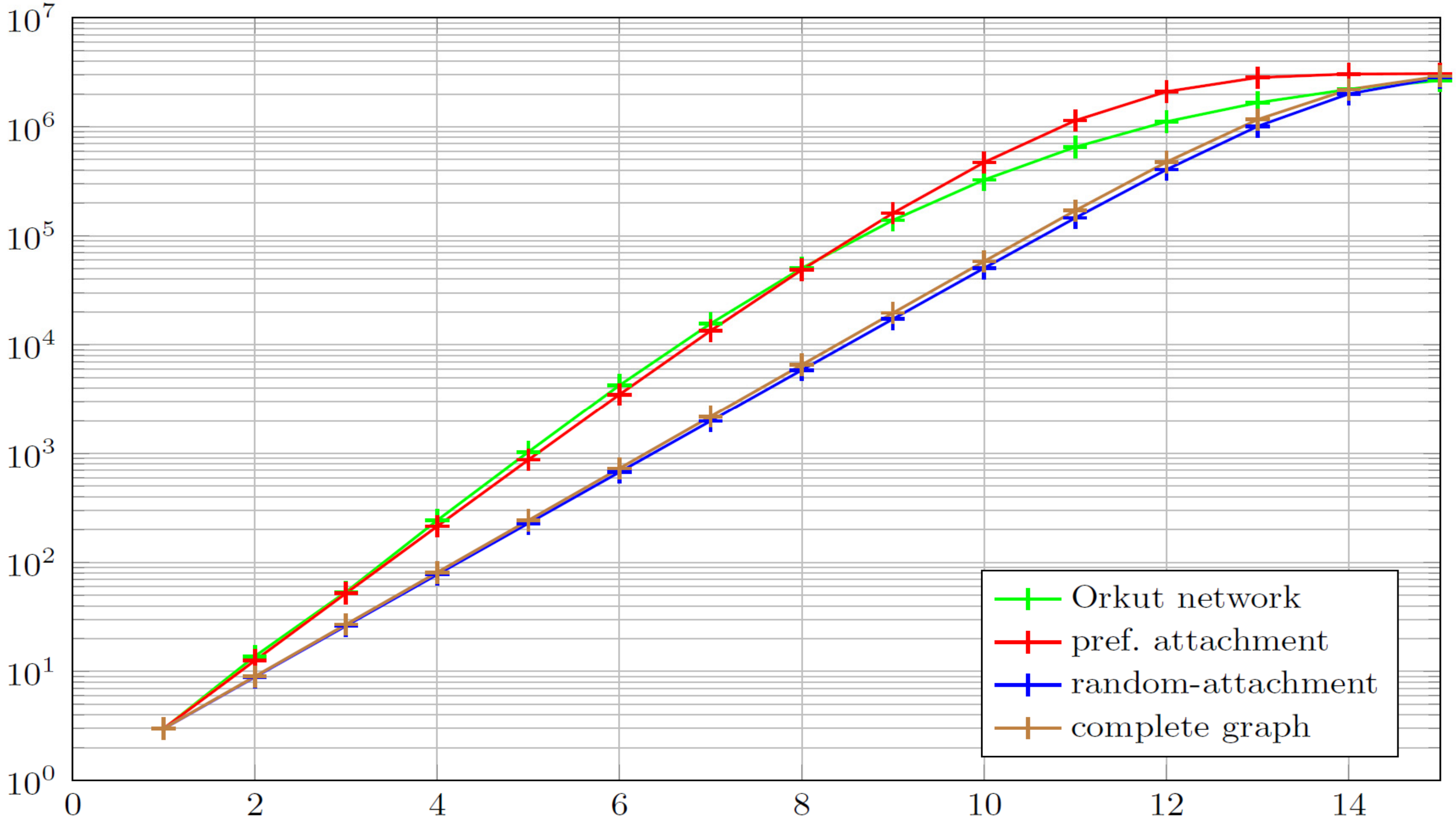
- D, Fouz, Friedrich (2011): In the push-pull protocol, the rumor spreading time is
  - $\Theta(\log n)$ whp
  - $\Theta(\log(n)/\log\log n)$ whp, if contacts are chosen excluding the neighbor contacted in the very previous round (no "double-contacts")
    - Note: Avoiding double-contacts does not improve the $O(\log n)$ times for complete graphs, $G(n,p)$ random graphs, hypercubes, …

Experiments: Time vs. Graph Size



Time to inform all vertices for different graph sizes (no double-contacts).
Observation: Hidden constants don't matter, PA is truly faster.

Experiments: Progress Over Time



Number of nodes informed after t rounds. All graphs: $n = 3{,}072{,}441$; density $m = 38$ (except complete). Orkut: Google's Facebook (100m users in India and Brasil).

Graphs Used in Previous Charts

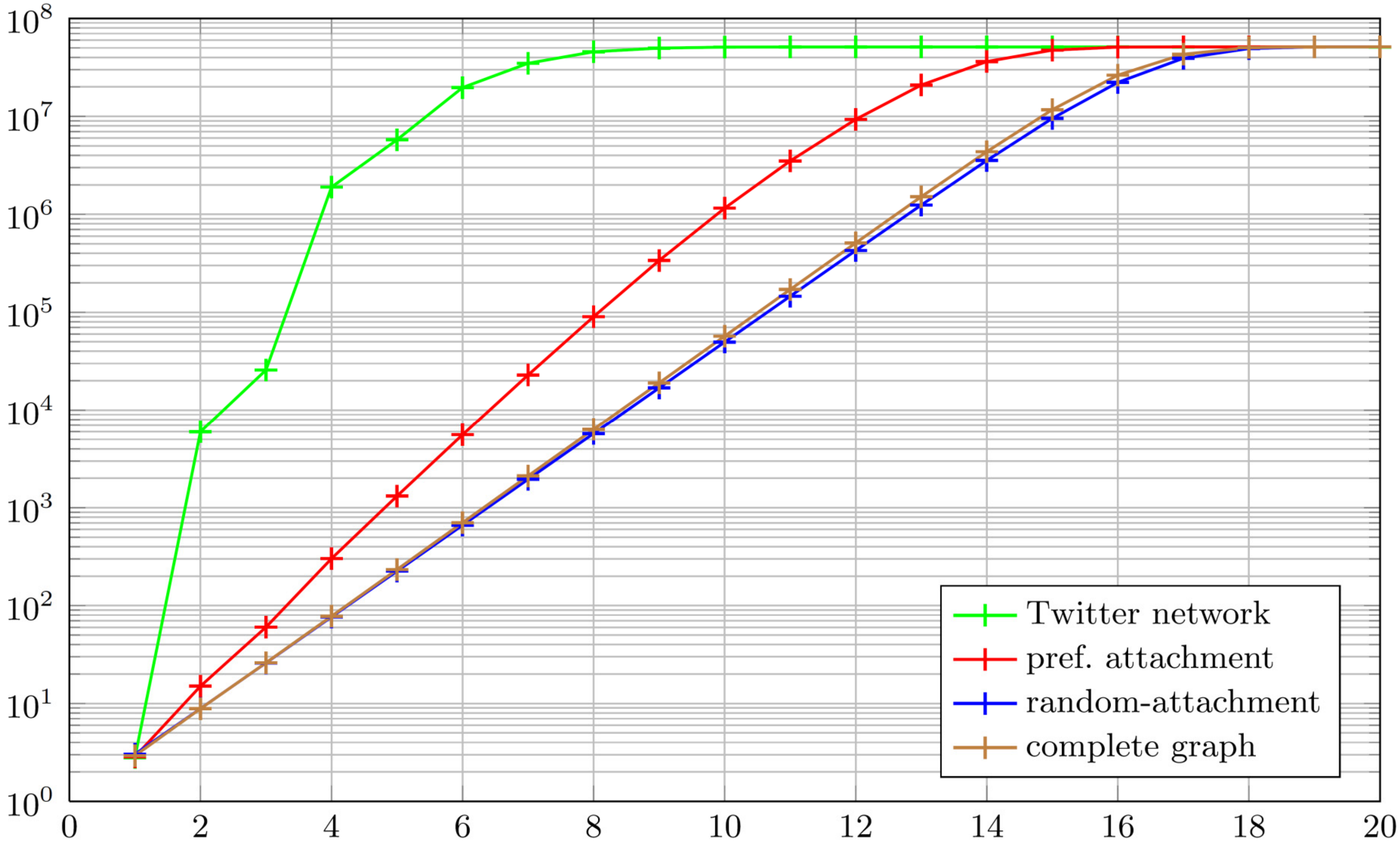
- Orkut: 2006 crawl of around 11% the Orkut social network (Google's alternative to Facebook, today very popular in India and Brazil, ~100,000,000 users, Alexa traffic rank 81st [at that time, now 200,000]): $n = $ 3,072,441 nodes, ~117 million edges (approx. $38n$ edges).

- Preferential attachment (PA) graph: n nodes, each chooses $m = 38$ neighbors, giving higher preference to already popular nodes

- Random-attachment graph (m-out random graph): n nodes, each chooses m neighbors uniformly at random

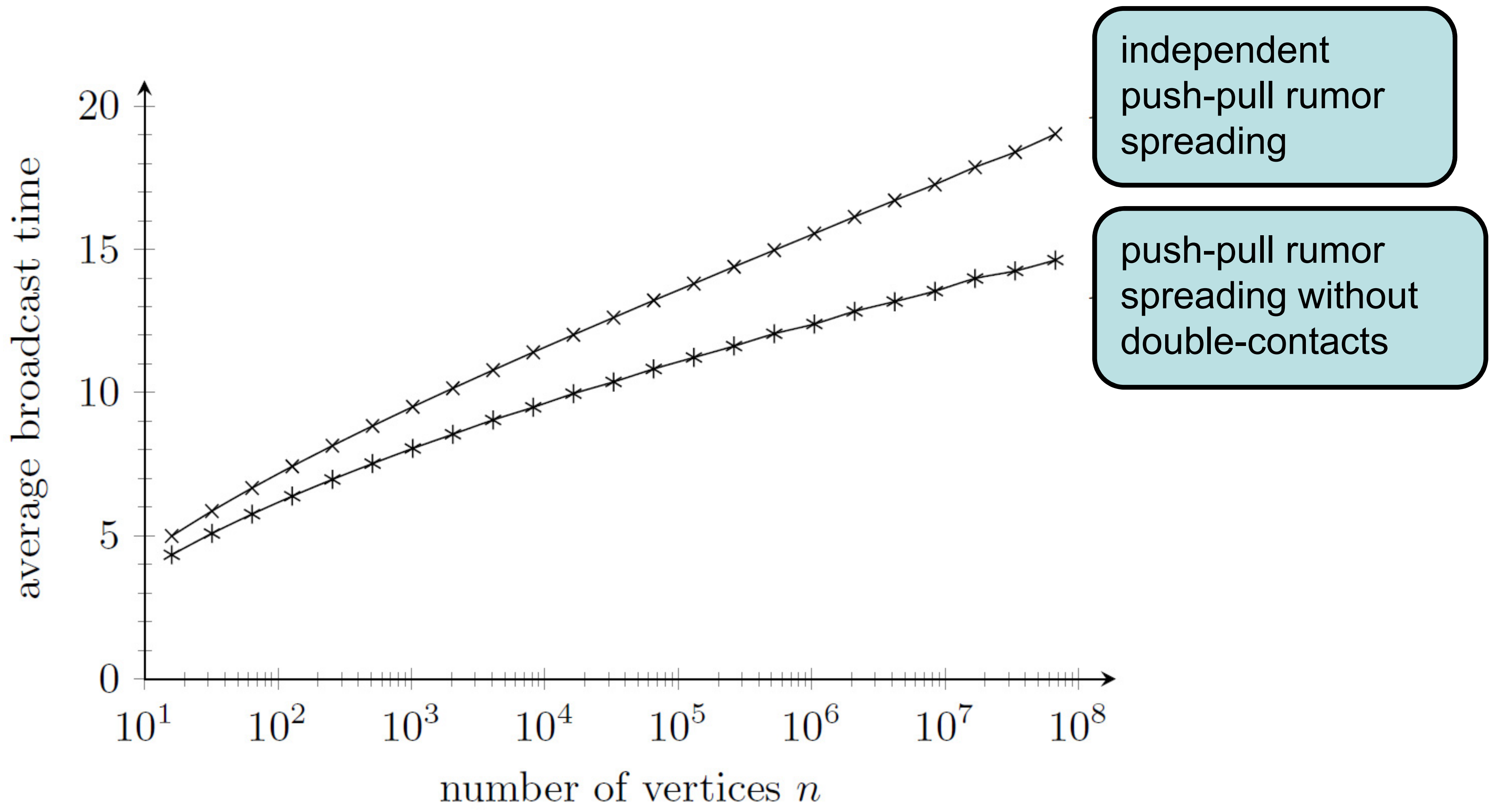
- Complete graph on n vertices

Experiments: Same with Twitter



$n = 51{,}161{,}011$ nodes, $1{,}613{,}927{,}450$ edges, density $m = 32$.

Use of Double Contacts



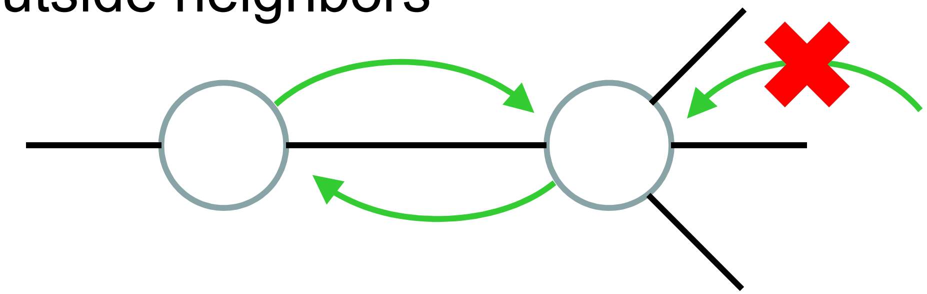
Back to Theory…

- Theorem: Randomized rumor spreading in the push-pull model informs the PA graph G^n (with $m \geq 2$) with high probability in
  - $\Theta(\log n)$ rounds when choosing neighbors uniformly at random
  - $\Theta(\log n \,/\, \log\log n)$ rounds without double-contacts

- Two questions:
  - Why do double-contacts matter?
  - What makes PA graphs spread rumors faster than other graphs?

With Double-Contacts…

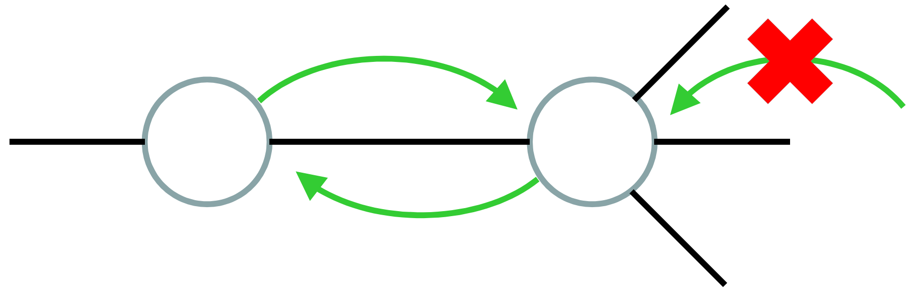
- Critical situation:
  - A pair of uninformed neighboring nodes, each having a constant number of outside neighbors



- With constant probability, the following happens in one round:
  - the two nodes of the pair call each other
  - all their neighbors call someone outside the pair
  - → hence the situation remains critical (pair uninformed)
- Problem: Initially, there are $\Theta(n)$ such critical situations in a PA graph. Since each is solved with constant probability in one round, $\Omega(\log n)$ rounds are necessary

Without Double-Contacts

- The uninformed pair is not critical anymore, because the two nodes cannot call each other twice in a row ☺



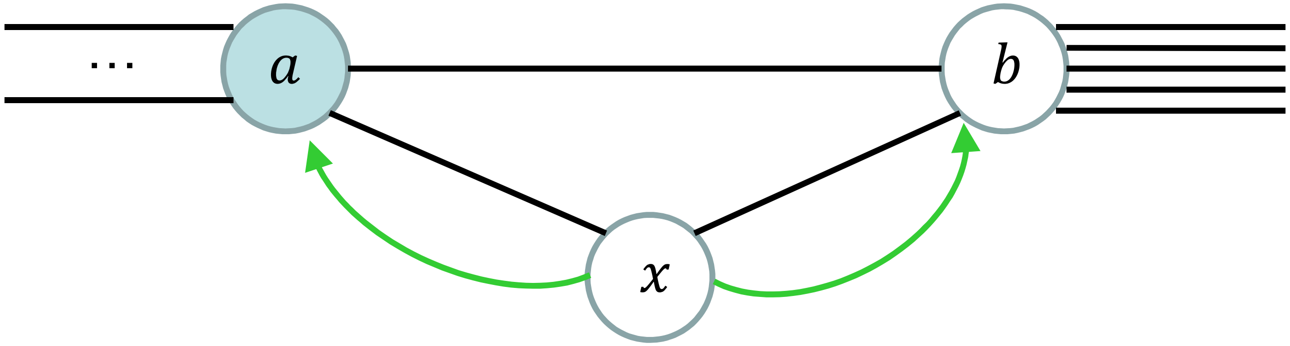
- Remaining critical situations: Uninformed cycles having a constant number of outside neighbors in total.
  - Again, each round, with constant probability the situation remains critical (cycle uninformed)

- No problem! There are only $O(\exp((\log n)^{3/4}))$ such critical situations initially in a PA graph. If each is solved with constant probability, we need $O\big(\log(\exp((\log n)^{3/4}))\big) = O\left((\log n)^{3/4}\right)$ rounds to solve them all ☺

Why Are PA Graphs Faster?

- Large- and small-degree nodes:
  - hub: node with degree $(\log n)^3$ or greater
  - poor node: node with degree exactly m (as small as possible)

- Observation: Poor nodes convey rumors fast!
  - Let a and b be neighbors of a poor node x
  - If a is informed, the expected time for x to pull the rumor from a is less than m
  - After that, it takes another less than m rounds (in expectation) for x to push the news to b



Why Are PA Graphs Faster (2)?

- Large- and small-degree nodes:
  - hub: node with degree $(\log n)^3$ or greater
  - poor node: node with degree exactly m (as small as possible)

- Observation: Poor nodes convey rumors fast!
  - Let a and b be neighbors of a poor node x
  - If a is informed, the expected time for x to pull the rumor from a is less than m
  - After that, it takes another less than m rounds (in expectation) for x to push the news to b

- Key lemma: Between any two hubs, there is a path of length $O(\log n \,/\, \log\log n)$ with every second node a poor node.

- Key lemma + observation + some extra arguments: If one hub is informed, after $O(\log n \,/\, \log\log n)$ rounds all hubs are.

Main Tool: BR'04 Definition of Preferential Attachment Model

Equivalent description of the PA model (also Bollobás & Riordan (2004))

- For $m = 1$
  - Choose $2n$ random numbers in $[0,1]$: $x_1, y_1, \dots, x_n, y_n$
  - If $x_i > y_i$, exchange the two values
    - $\Pr(y_i \leq r) = r^2$
  - Sort the (x, y) pairs by increasing y-value; call them again $(x_1, y_1), (x_2, y_2), \dots$
  - For all k, vertex k chooses the unique $i \leq k$ as neighbor which satisfies $y_{i-1} \leq x_k < y_i$
    - Note: x_k is uniform in $[0, y_k]$
- For $m \geq 2$: Generate G_1^{nm} as above, then merge each m consecutive nodes

- Advantage: Many independent random variables, not a sequential process

Some More Results

- Fountoulakis, Panagiotou, Sauerwald (SODA'12):
  - Chung-Lu graphs: variation of the classic Erdős-Rényi random graphs (independent edges) that yields a power-law degree distribution
  - Synchronized randomized rumor spreading in the push-pull model informs all but $o(n)$ nodes of the Chung-Lu graph
    - in $\Theta(\log n)$ rounds, if the power-law exponent $\beta > 3$
    - in $\Theta(\log\log n)$ rounds, if $2 < \beta < 3$
      [no result for $\beta = 3$, the PA exponent]
  - Asynchronous: Nodes call at times triggered by (their private) Poisson clock (with rate 1 → one call per time unit in expectation)
    - $2 < \beta < 3$: most nodes informed after a *constant* number of rounds!

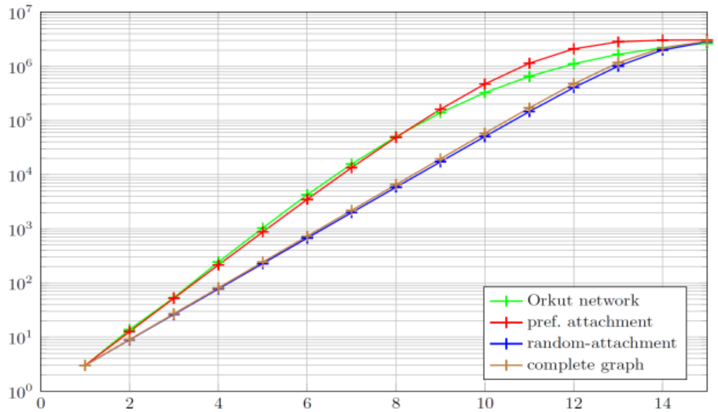
- D, Fouz, Friedrich (SWAT'12):
  - Asynchronous rumor spreading informs most nodes of the PA graph in $O((\log n)^{1/2})$ time [not at all clear is this is sharp]

Summary: Rumor Spreading in Preferential Attachment Graphs

- Theorem: Randomized rumor spreading in the push-pull model informs the PA graph G_m^n (with $m \geq 2$) with high probability in
  - $\Theta(\log n)$ rounds when choosing neighbors uniformly at random
  - $\Theta(\log n \,/\, \log\log n)$ rounds without double-contacts
  - asynchronous: most nodes informed after $O((\log n)^{1/2})$ rounds

- Explanation: Interaction between hubs and poor nodes (constant degree)
  - hubs are available to be called
  - poor nodes quickly transport the news from one neighbor to all others

- Difference visible in experiments:



Summary

- Rumor spreading is an efficient epidemic algorithm to disseminate information in various network topologies
  - hypercubes (as models for man-made communication networks)
  - random geometric graphs (model for wireless sensor networks)
  - preferential attachment graphs (model for social networks)
- Often, we can prove a rumor spreading time of $O(\mathrm{diam}\, G)$, which naturally is asymptotically optimal

Course Summary (1)

Epidemic/gossip-based algorithms&processes

- Models for processes in the real world
  - epidemics (including computer viruses and malware)
  - rumors
  - making acquaintances
  - [viral marketing, influence processes, adoption of new technologies]

- Lightweight and robust distributed algorithms
  - information dissemination
  - network exploration (e.g., computing the number of nodes)
  - computing aggregate information of distributed data

Course Summary (2)

- Power of epidemic/gossip-based algorithmics
  - simple generic algorithm design paradigm: talk to a random neighbor
  - performance often close to the best that can be achieved
  - works well in networks without central organization, stable structure, or reliable communication

- Analysis techniques: Similar to other randomized/distributed algorithms
  - Expectations, birthday paradox, coupon collector
  - Markov chain arguments (adding waiting times)
  - Strong concentration: Chernoff bounds, method of bounded differences

Outlook: Current Research

- Gossip-based algorithms in dynamic networks
  - motivation: these are networks, where you would use gossip-based algorithms
  - challenge: difficult to analyze
  - general difficulty: what is a reasonable model for dynamic networks

- Designing “more clever” gossip-based algorithms:
  - combine “gossip-based” with other algorithmic techniques
  - target: overcome weaknesses of pure gossiping (termination, communication overhead, coupon collector effects, …)

- Epidemic processes in real-world networks

- Lots of recent work on STOC, FOCS, SODA, PODC, ICALP … that is not fully digested yet (→ good topic for an internship, PhD thesis, …)

The End of This Part

- I hope you enjoyed this part of the course, you learned something interesting about epidemic algorithms and beyond.

- Next Thursday, 10:15 here: exam (80min) on this part of the course
  - you can use all *lecture* materials and personal notes
  - no electronic devices except a pocket calculator

- Thursday January 30: reading group – paper assignment

- Good luck in the exam!