

Mining Large Graphs

ECML/PKDD 2007 tutorial

Part 2: Diffusion and cascading behavior

Jure Leskovec and Christos Faloutsos

Machine Learning Department



Carnegie Mellon

Joint work with: Lada Adamic, Deepay Chakrabarti, Natalie Glance, Carlos Guestrin, Bernardo Huberman, Jon Kleinberg, Andreas Krause, Mary McGlohon, Ajit Singh, and Jeanne VanBriesen.

Tutorial outline

- Part 1: Structure and models for networks
 - What are properties of large graphs?
 - How do we model them?
- Part 2: Dynamics of networks
 - Diffusion and cascading behavior
 - How do viruses and information propagate?
- Part 3: Case studies
 - 240 million MSN instant messenger network
 - Graph projections: how does the web look like

Part 2: Outline

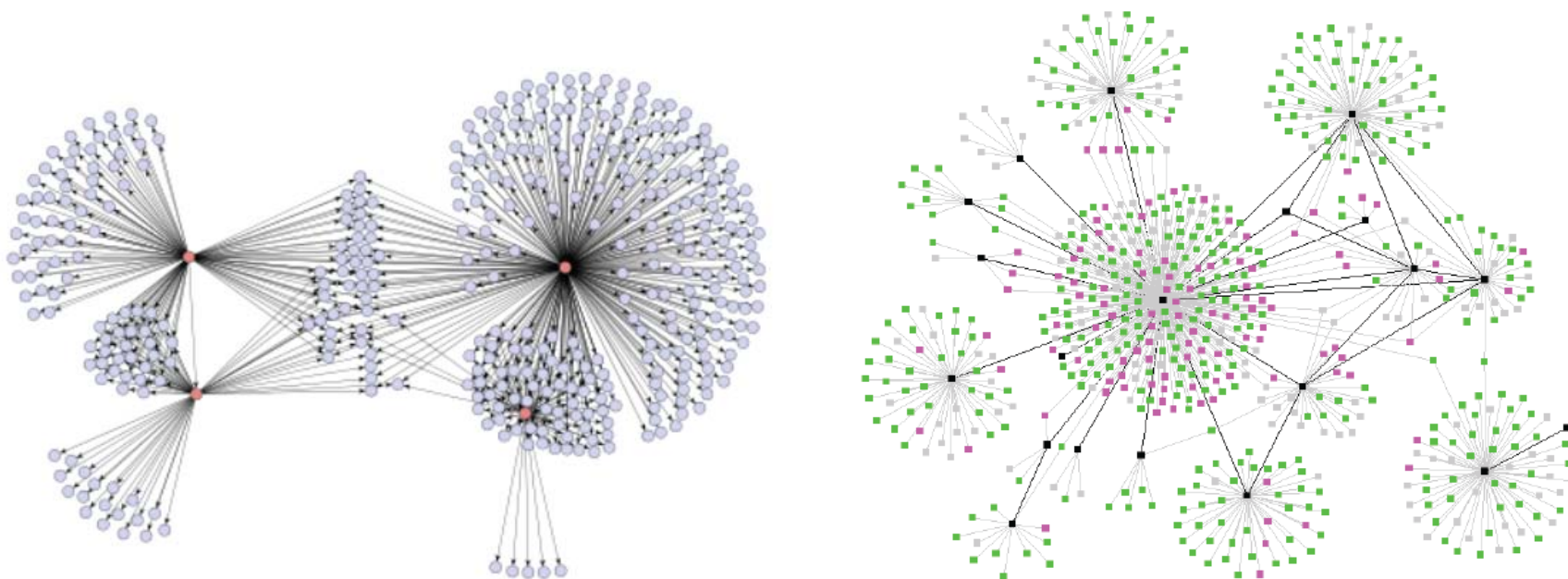
Diffusion and Cascading Behavior

- Part 1: Basic mathematical models
 - Virus propagation and Diffusion (cascading behavior)
 - Finding influential nodes
- Part 2: Empirical studies on large networks
 - Viral Marketing and Blogging
- Part 3: More algorithms and consequences
 - Detecting cascades effectively
- Conclusion and reflections

Structure vs. Process

- What have we learned about large networks?
- We know a lot about the **structure**: Many recurring patterns
 - Scale-free, small-world, locally clustered, bow-tie, hubs and authorities, communities, bipartite cores, network motifs, highly optimized tolerance
- We know much less about **processes** and **dynamics**

Diffusion in Social Networks



- One of the networks is a spread of a disease, the other one is product recommendations
- Which is which? 😊

Diffusion in Social Networks

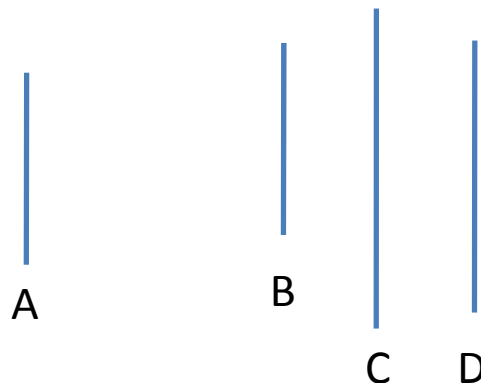
- A fundamental process in social networks:
Behaviors that cascade from node to node like an epidemic
 - News, opinions, rumors, fads, urban legends, ...
 - Word-of-mouth effects in marketing: rise of new websites, free web based services
 - Virus, disease propagation
 - Change in social priorities: smoking, recycling
 - Saturation news coverage: topic diffusion among bloggers
 - Internet-energized political campaigns
 - Cascading failures in financial markets
 - Localized effects: riots, people walking out of a lecture

Empirical Studies of Diffusion (1)

- Experimental studies of diffusion have long history:
 - Spread of new agricultural practices [Ryan-Gross 1943]
 - Adoption of a new hybrid-corn between the 259 farmers in Iowa
 - Classical study of diffusion
 - Interpersonal network plays important role in adoption
 - Diffusion is a social process
 - Spread of new medical practices [Coleman et al 1966]
 - Studied the adoption of a new drug between doctors in Illinois
 - Clinical studies and scientific evaluations were not sufficient to convince the doctors
 - It was the social power of peers that led to adoption

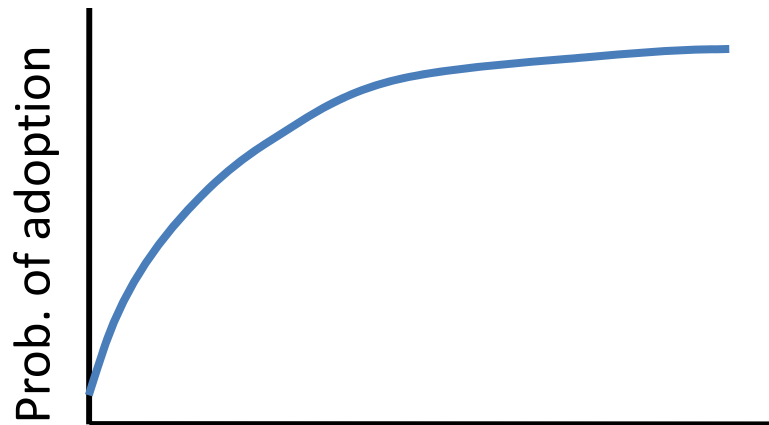
Empirical Studies of Diffusion (2)

- Diffusion has many (very interesting) 😊 flavors, *e.g.*:
 - The contagion of obesity [Christakis et al. 2007]
 - If you have an overweight friend your chances of becoming obese increases by 57%
 - Psychological effects of others' opinions, *e.g.*:
Which line is closest in length to A? [Asch 1958]



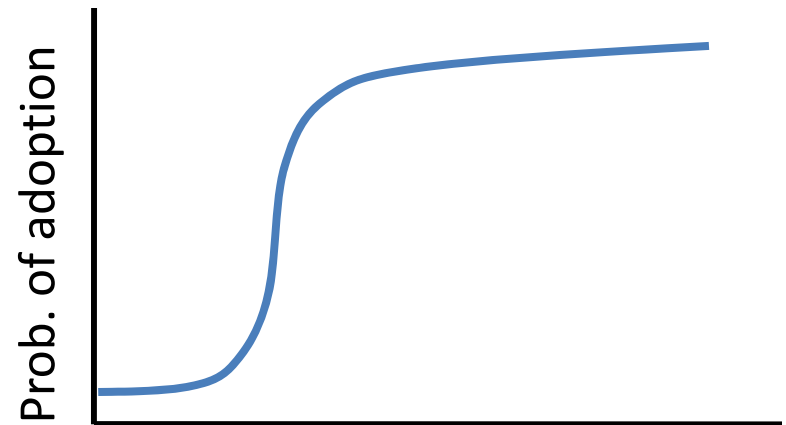
Diffusion Curves (1)

- Basis for models:
 - Probability of adopting new behavior depends on the number of friends who have adopted [Bass '69, Granovetter '78, Shelling '78]
- What's the dependence?



k = number of friends adopting

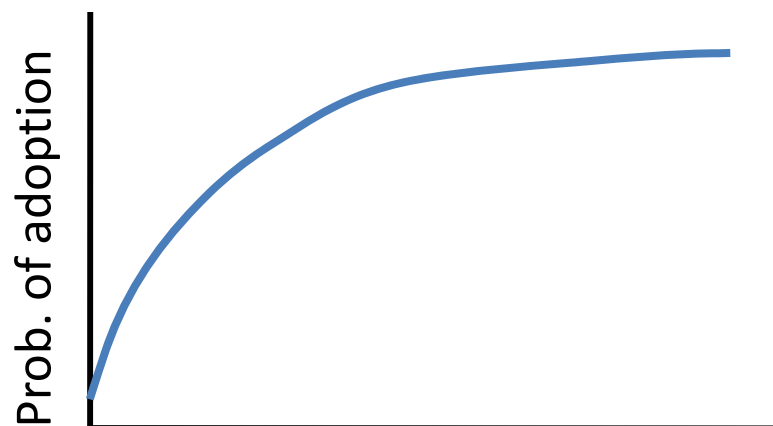
Diminishing returns?



k = number of friends adopting

Critical mass?

Diffusion Curves (2)



k = number of friends adopting

Diminishing returns?



k = number of friends adopting

Critical mass?

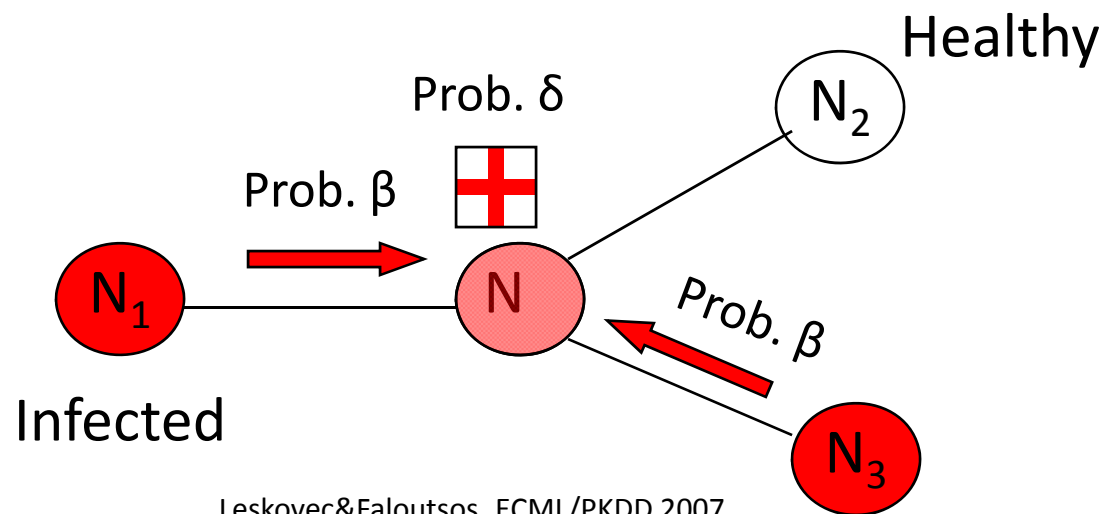
- **Key issue:** qualitative shape of diffusion curves
 - Diminishing returns? Critical mass?
 - Distinction has consequences for models of diffusion at population level

Part 1: Mathematical Models

- Two flavors, two types of questions:
 - A) Models of Virus Propagation:
 - SIS: Susceptible – Infective – Susceptible (*e.g.*, flu)
 - SIR: Susceptible – Infective – Recovered (*e.g.*, chicken-pox)
 - **Question:** Will the virus take over the network?
 - B) Models of Diffusion:
 - Independent contagion model
 - Threshold model
 - **Questions:**
 - Finding influential nodes
 - Detecting cascades

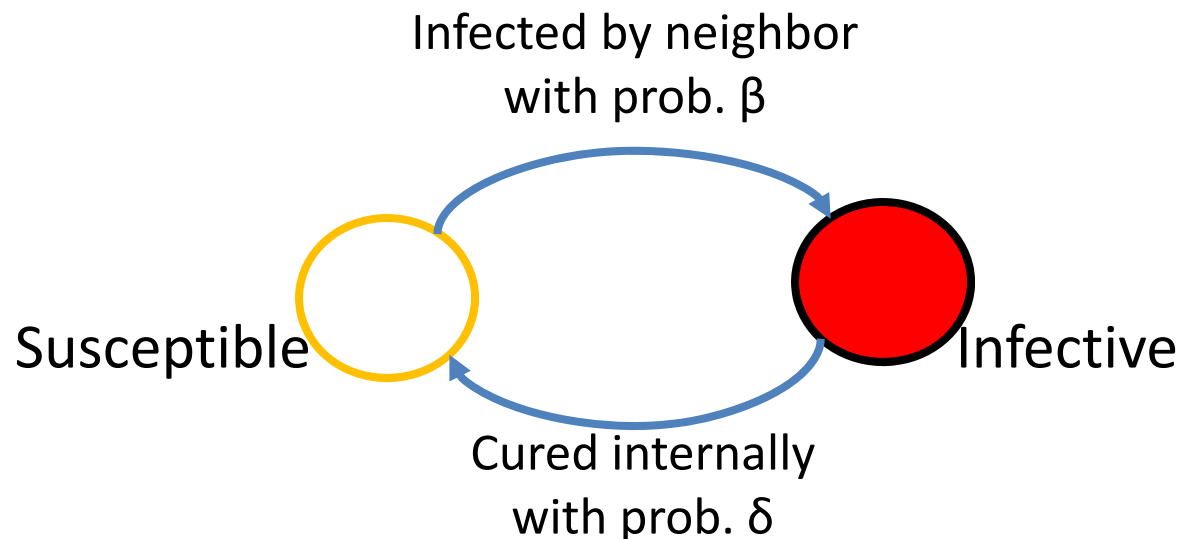
A) Models of Virus Propagation

- How do viruses/rumors propagate?
- Will a flu-like virus linger, or will it become extinct?
- (Virus) birth rate β : probability than an infected neighbor attacks
- (Virus) death rate δ : probability that an infected node heals



The Model

- Susceptible-Infective-Susceptible (SIS) model
- Cured nodes immediately become susceptible
- Virus “strength”: $s = \beta / \delta$



Question: Epidemic Threshold τ

of a graph: the value of τ , such that

$$\text{If strength } s = \beta / \delta < \tau$$

epidemic can not happen

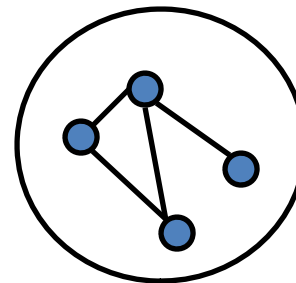
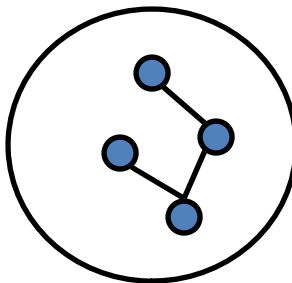
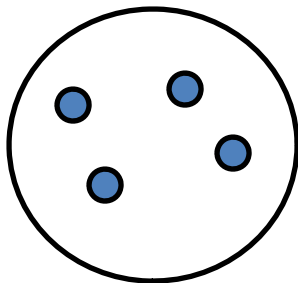
Thus,

- given a graph
- compute its epidemic threshold

Epidemic Threshold τ

What should τ depend on?

- avg. degree? and/or highest degree?
- and/or variance of degree?
- and/or third moment of degree?
- and/or diameter?



Epidemic Threshold

Theorem [Wang+ 2003]:

We have no epidemic if:

The diagram shows the equation $\beta/\delta < \tau = 1/\lambda_{1,A}$ enclosed in a red rectangular box. Annotations include: an arrow from '(Virus) Death rate' pointing to δ ; an arrow from '(Virus) Birth rate' pointing to β ; an arrow from 'Epidemic threshold' pointing to τ ; and a red arrow from 'largest eigenvalue of adj. matrix A ' pointing to $\lambda_{1,A}$.

(Virus) Death rate

Epidemic threshold

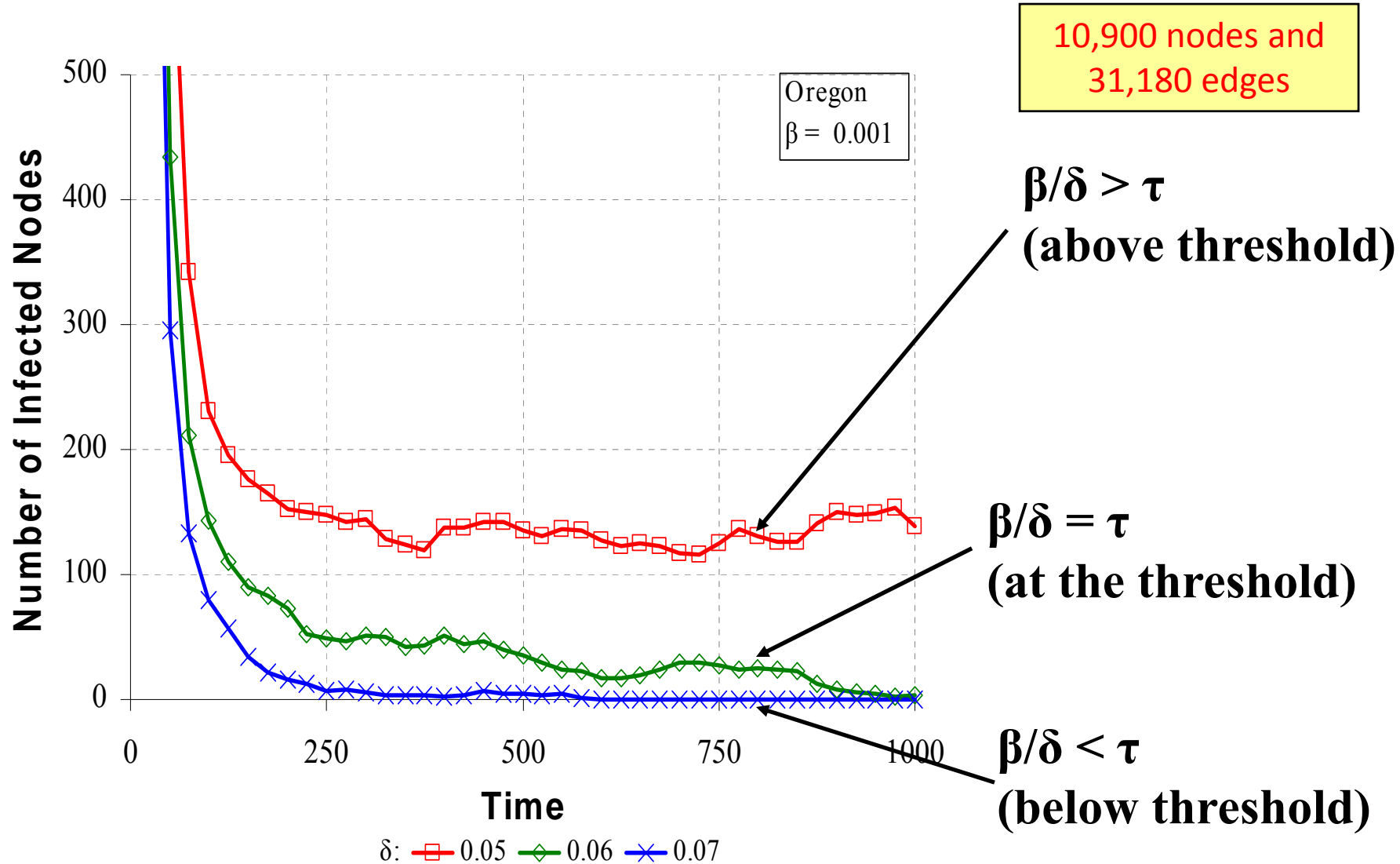
$$\beta/\delta < \tau = 1/\lambda_{1,A}$$

(Virus) Birth rate

largest eigenvalue of adj. matrix A

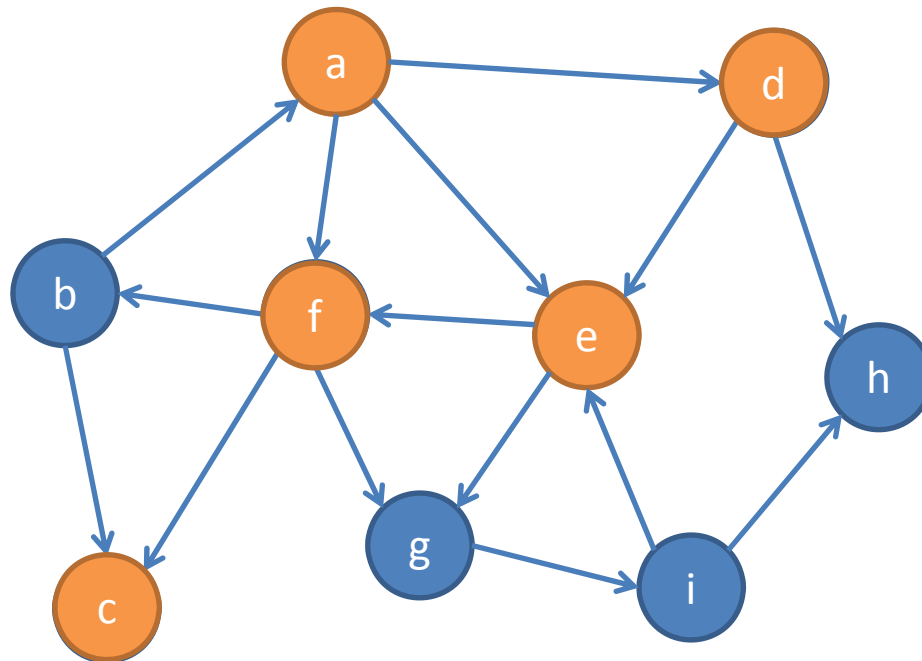
► $\lambda_{1,A}$ *alone captures the property of the graph!*

Experiments (AS graph)



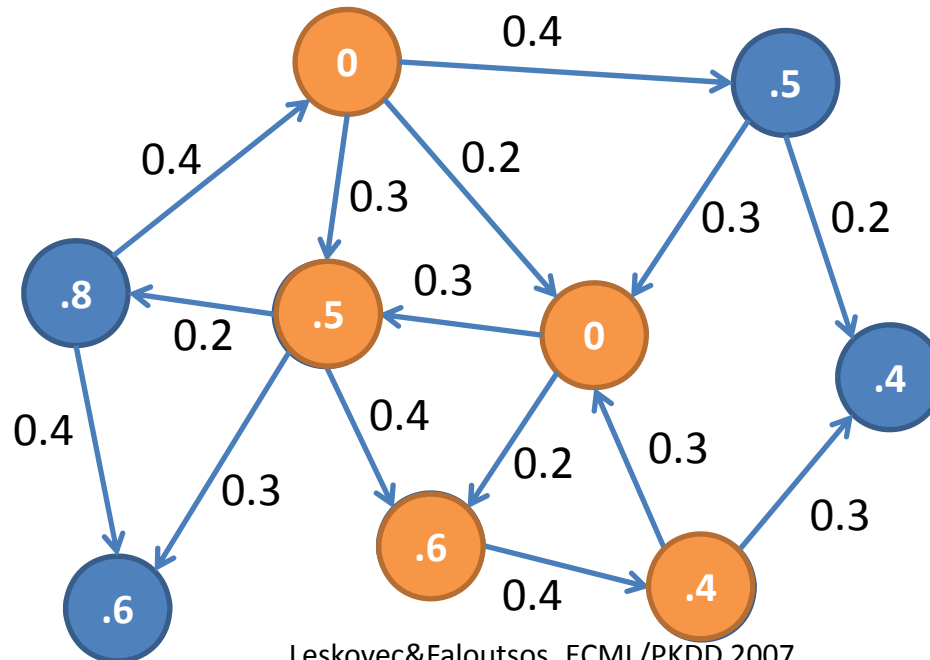
B) Models of Diffusion in Networks

- Initially some nodes are active
- Active nodes spread their influence on the other nodes, and so on ...



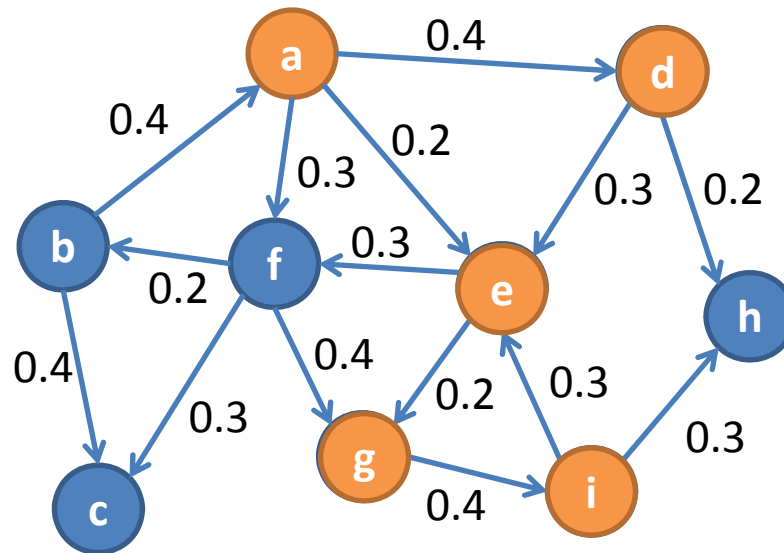
Threshold Model [Granovetter '78]

- Initially some nodes are active
- Each edge (u,v) has weight w_{uv}
- Each node has a threshold t
- Node u activates if $t < \sum_{active(u)} w_{uv}$



Independent Contagion Model

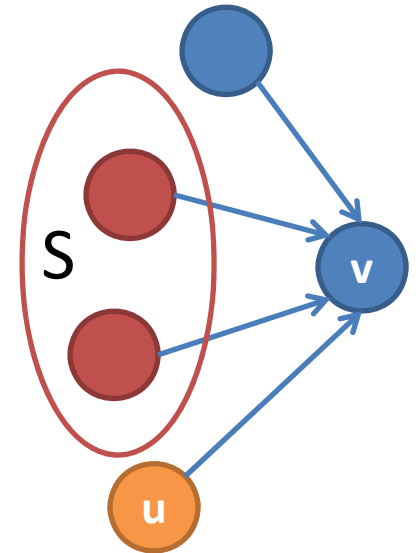
- Initially some nodes are active
- Each edge (u,v) has probability (weight) p_{uv}



- Node a becomes active: activates node b with prob. p_{uv}
- Activations spread through the network

General Contagion Model

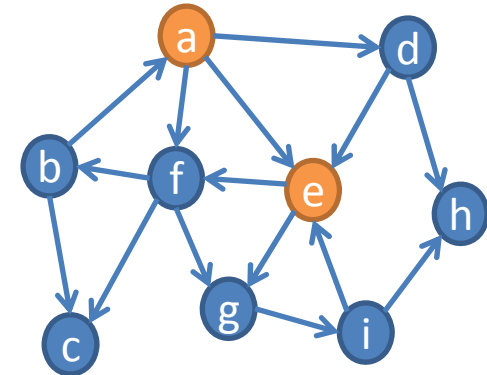
- What general language do we need to describe diffusion?
- [Kempe et al. '03, Dodds-Watts '04]
 - When u tries to influence v : success based on set of nodes S that already tried and failed
 - Success functions $p_v(u, S)$



- Independent cascades: $p_v(u, S) = p_{uv}$
- Threshold: if $|S| = k$: $p_v(u, S) = 1$ else 0
- Diminishing returns: $p_v(u, S) \geq p_v(u, T)$ if $S \subseteq T$

Most Influential Subset of Nodes

- If S is initial active set, let $f(S)$ denote expected size of final active set
- **Most influential set of size k** : the set S of k nodes producing largest expected cascade size $f(S)$ if activated [Domingos-Richardson 2001]

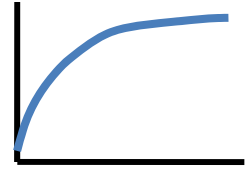


- As a discrete optimization problem

$$\max_{S \text{ of size } k} f(S)$$

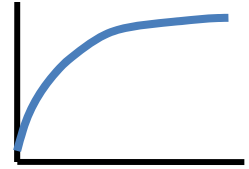
- NP-hard and highly inapproximable
 - Proof relies on critical mass. **Is it necessary?**

An Approximation Result



- Diminishing returns: $p_v(u, S) \geq p_v(u, T)$ if $S \subseteq T$
- Hill-climbing: repeatedly select node with maximum marginal gain
- Performance guarantee: hill-climbing algorithm is within $(1-1/e) \sim 63\%$ of optimal [Kempe et al. 2003]

An Approximation Result



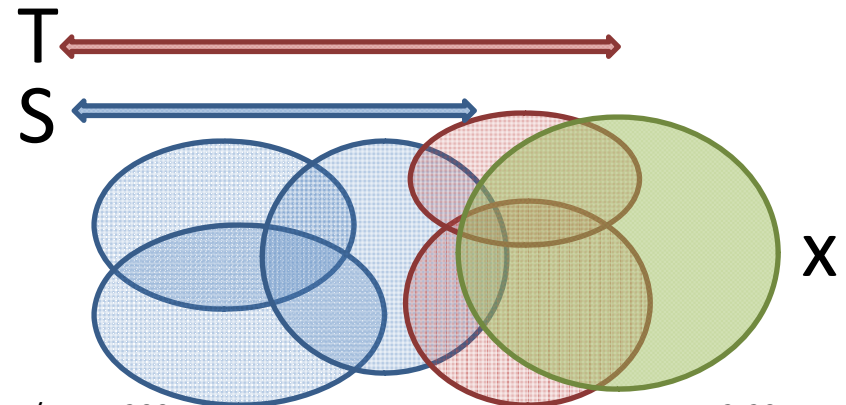
- Analysis: diminishing returns at individual nodes implies diminishing returns at a “global” level
 - Cascade size $f(S)$ grows slower and slower with S .
 - f is submodular: if $S \subseteq T$ then
$$f(S \cup \{x\}) - f(S) \geq f(T \cup \{x\}) - f(T)$$
 - Theorem [Nehmhauser et al. ‘78]:
If f is a function that is monotone and submodular, then k -step hill-climbing produces set S for which $f(S)$ is within $(1-1/e)$ of optimal.

Analysis: Independent Contagion

- Our function f is clearly monotone; we must show that it is submodular:

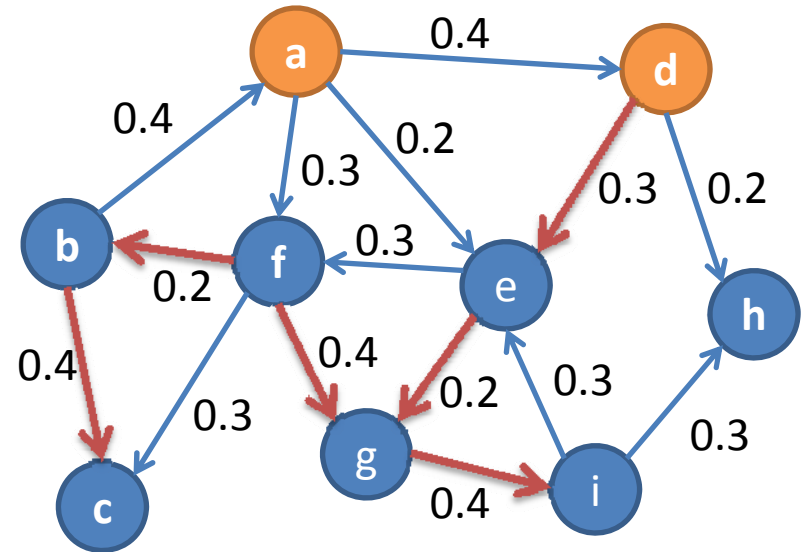
$$f(S \cup \{x\}) - f(S) \geq f(T \cup \{x\}) - f(T)$$

- What do we know about submodular functions?
 - 1) If f_1, f_2, \dots, f_k are submodular, and $a_1, a_2, \dots, a_k > 0$ then $\sum a_i f_i$ is also submodular
 - 2) Natural example:
 - Sets A_1, A_2, \dots, A_n :
 - $f(S) = \text{size of union of } A_i$



Analysis: Alternative View

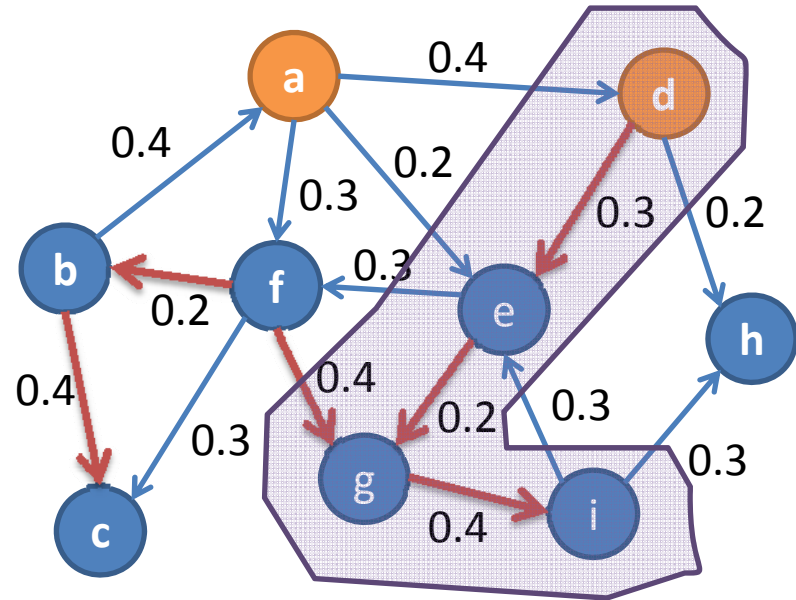
- Alternative view:
 - Generate the randomness ahead of time



- Flip a coin for each edge to decide whether it will succeed when (if ever) it attempts to transmit
- Edges on which activation will succeed are live
- $f(S)$ = size of the set reachable by live-edge paths

Analysis: Alternative View

- Fix outcome i of coin flips
- Let $f_i(S)$ be size of cascade from S given these coin flips

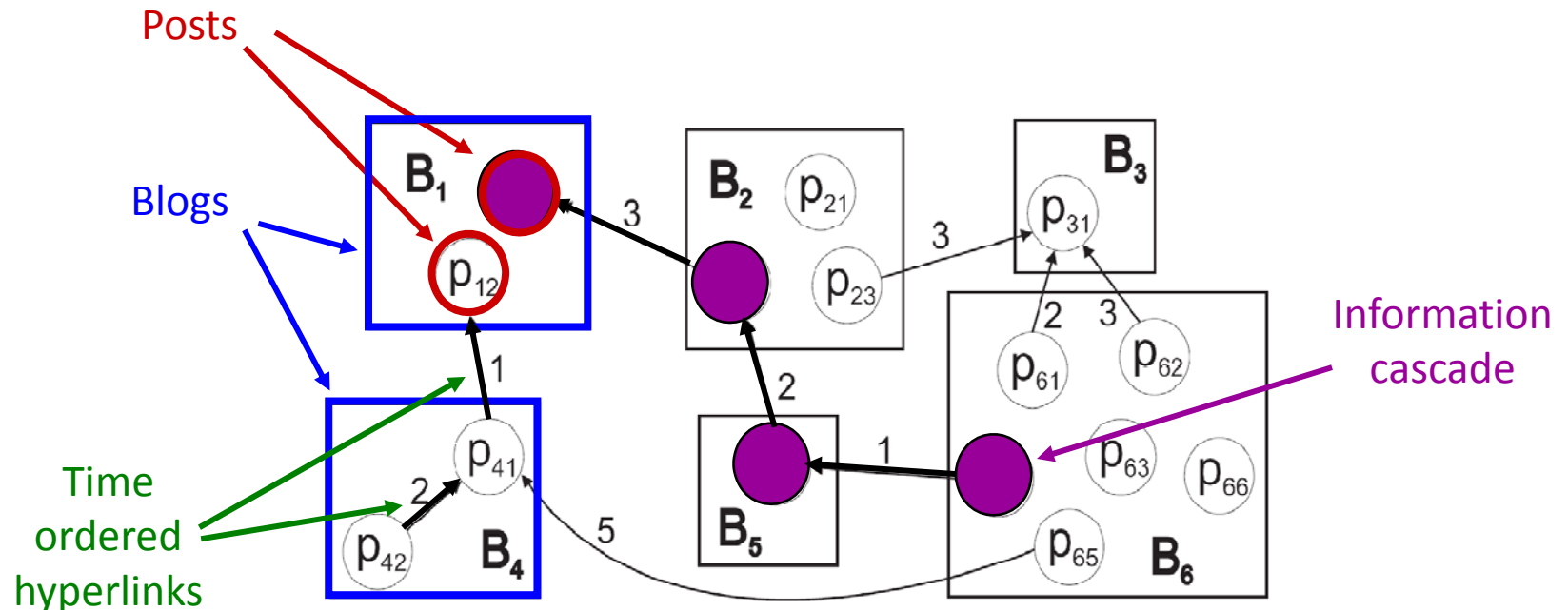


- Let $R_i(v)$ = set of nodes reachable from v on **live-edge** paths
- $f_i(S)$ = size of union $R_i(v) \rightarrow f_i$ is submodular
- $f = \sum \text{Prob}[i] f_i \rightarrow f$ is submodular

Part 2: Empirical Analysis

- What do diffusion curves look like?
- How do cascades look like?
- **Challenge:**
 - Large dataset where diffusion can be observed
 - Need social network links and behaviors that spread
- We use:
 - **Blogs:** How information propagates? [Leskovec et al. 2007]
 - **Product recommendations:** How recommendations and purchases propagate? [Leskovec-Adamic-Huberman 2006]
 - **Communities:** How community membership propagates? [Backstrom et al. 2006]

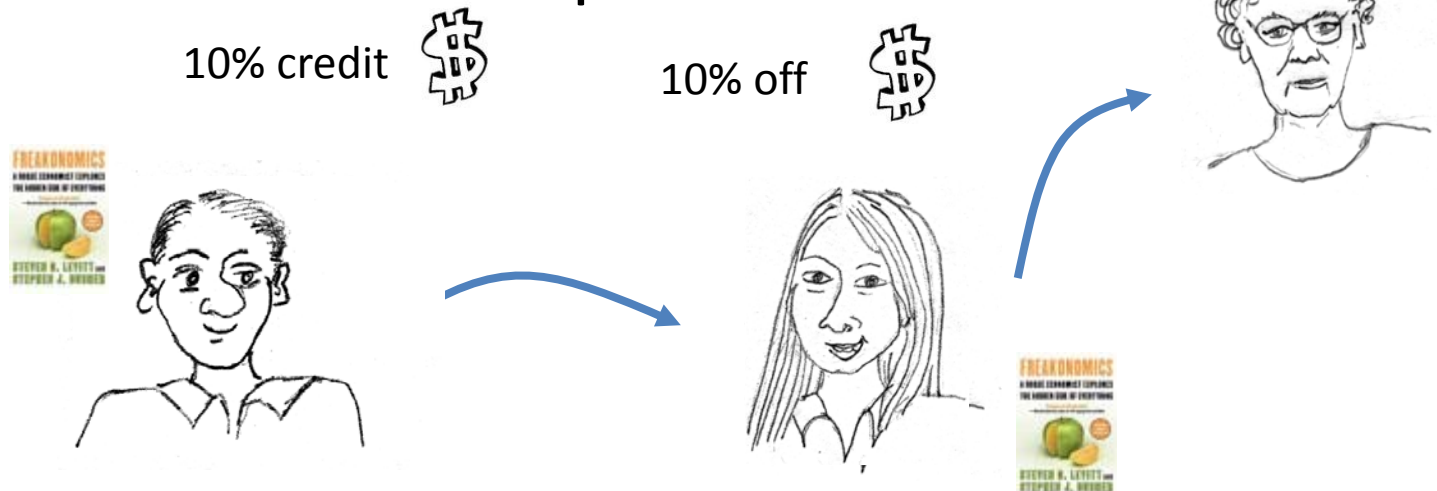
Diffusion in Blogs



- Data – Blogs:
 - We crawled 45,000 blogs for 1 year
 - 10 million posts and 350,000 cascades

Diffusion in Viral Marketing

- Senders and followers of recommendations receive discounts on products



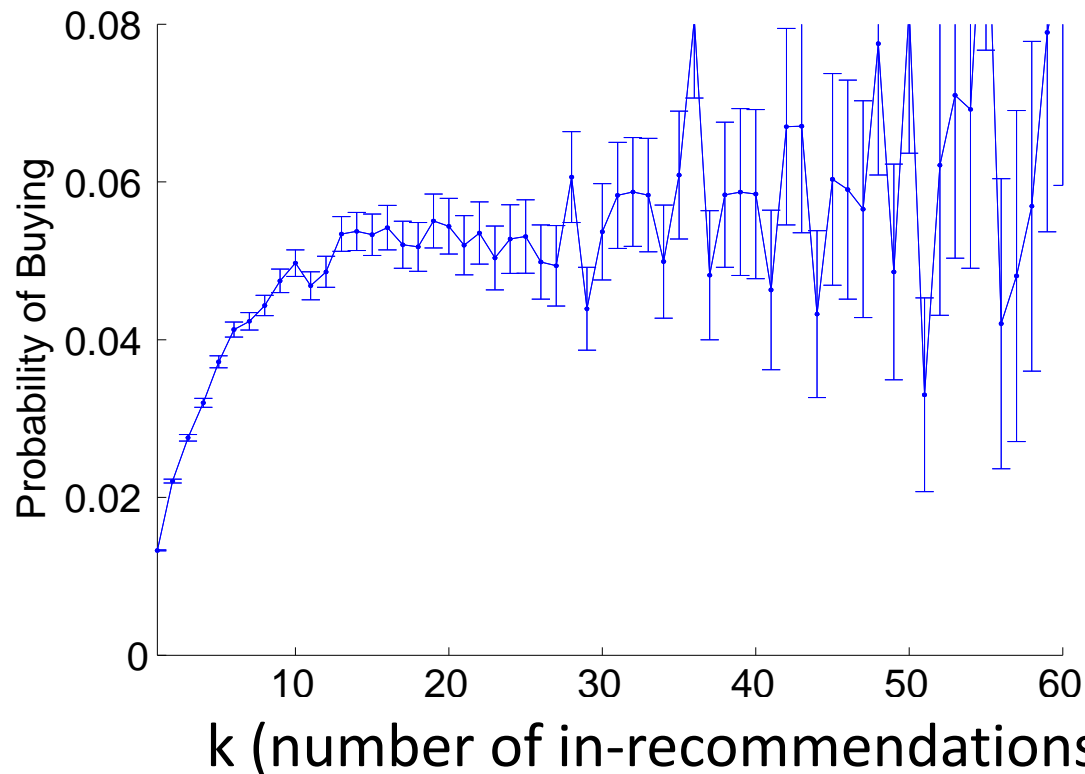
- Data – Incentivized Viral Marketing program
 - 16 million recommendations
 - 4 million people
 - 500,000 products

Diffusion of Community Membership

- Use social networks where people belong to explicitly defined groups
- Each group defines a behavior that diffuses
- Data – LiveJournal:
 - On-line blogging community with friendship links and user-defined groups
 - Over a million users update content each month
 - Over 250,000 groups to join

How do diffusion curves look like?

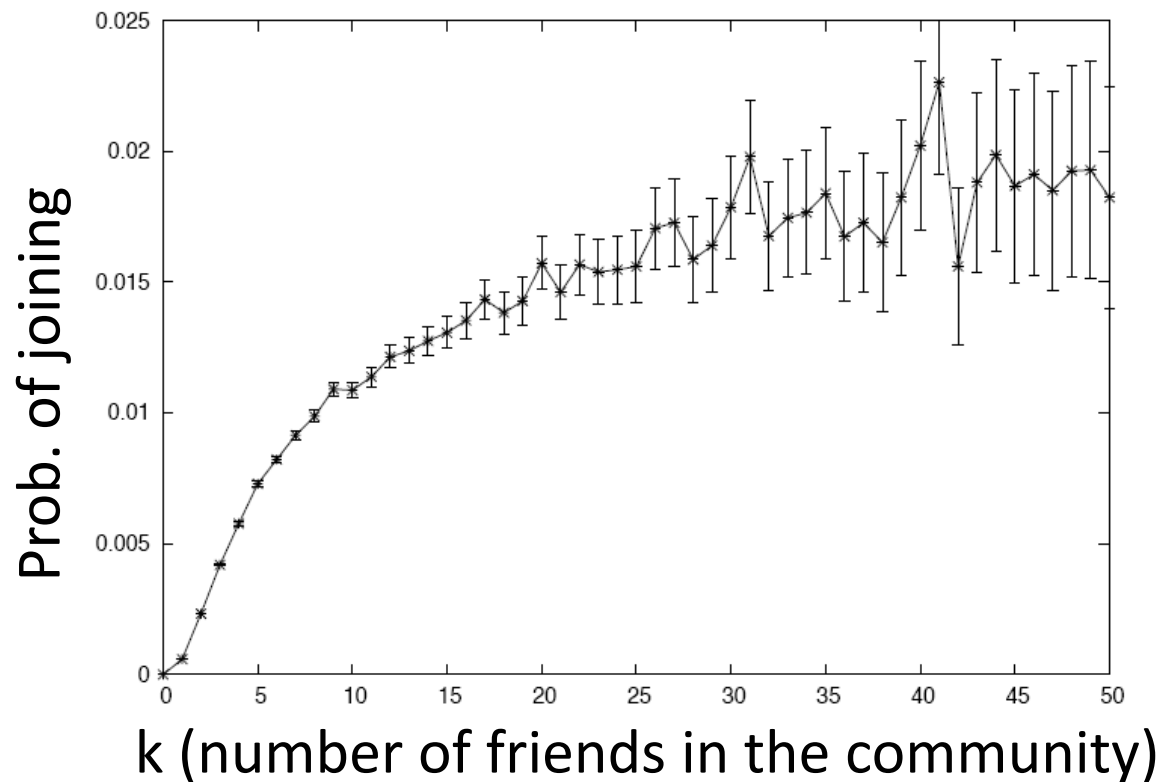
- Viral marketing – DVD purchases:



- Mainly diminishing returns (**saturation**)
- Turns upward for $k = 0, 1, 2, \dots$

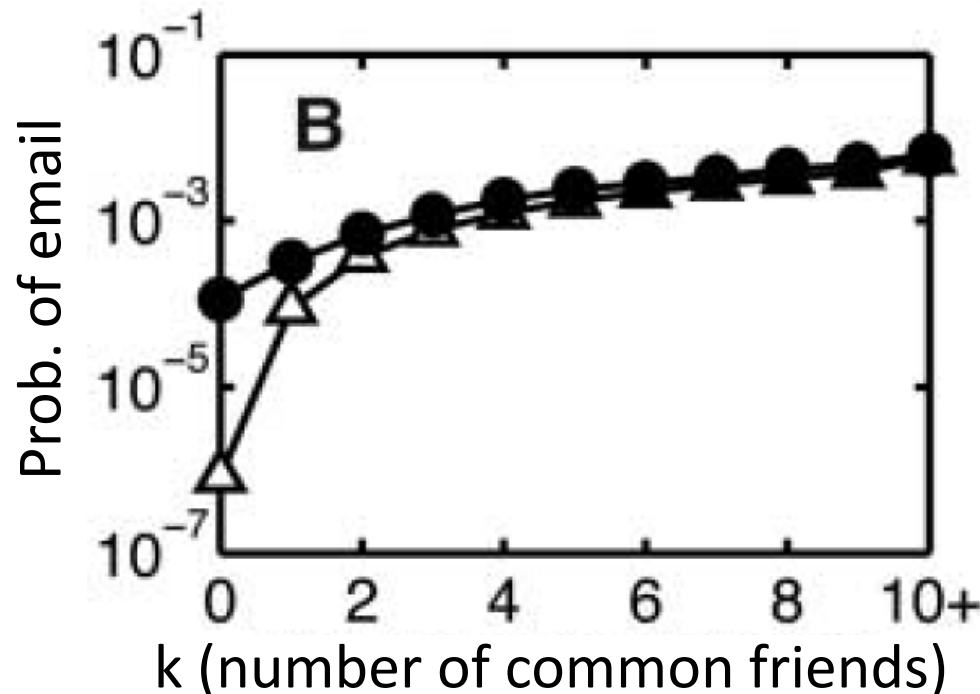
How do diffusion curves look like?

- LiveJournal community membership:



How do diffusion curves look like?

- Email probability [Kossinets-Watts 2006]:
 - Email network of large university
 - Prob. of a link as a function of # of common friends

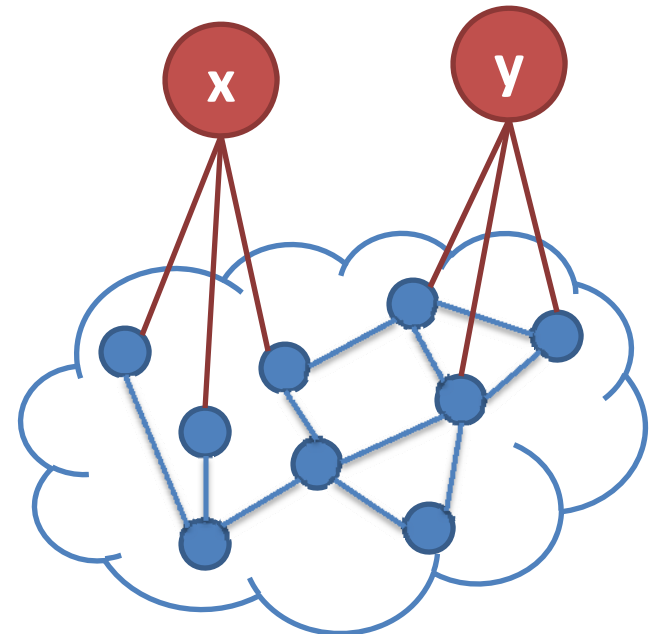


What are we really measuring?

- For viral marketing:
 - We see that node v receiving the i -th recommendation and then purchased the product
- For communities:
 - At time t we see the behavior of node v 's friends
- Questions:
 - When did v become aware of recommendations or friends' behavior?
 - When did it translate into a decision by v to act?
 - How long after this decision did v act?

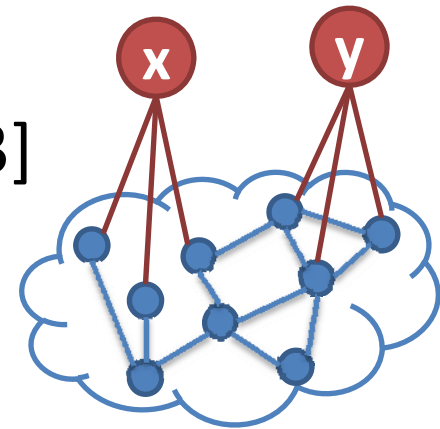
More subtle features: Communities

- Dependence on number of friends
- Consider: connectedness of friends
 - x and y have three friends in the group
 - x's friends are independent
 - y's friends are all connected
 - Who is more likely to join?



Connectedness of Friends

- Competing sociological theories
 - Information argument [Granovetter '73]
 - Social capital argument [Coleman '88]
- Information argument:
 - Unconnected friends give independent support
- Social capital argument:
 - Safety>truest advantage in having friends who know each other

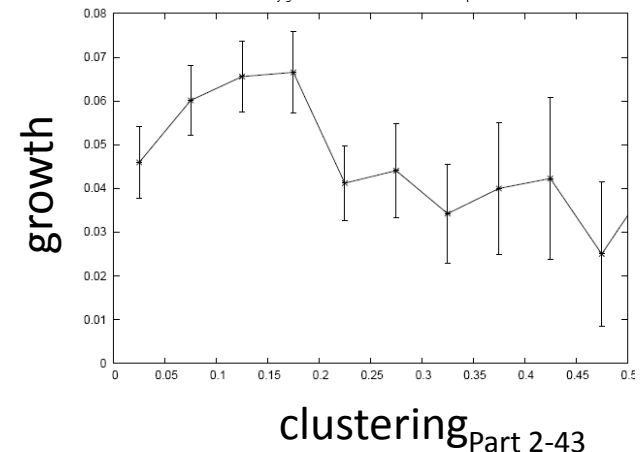


Connectedness of Friends

- In LiveJournal, community joining probability increases with more connections among friends in group
- Number and connectedness of friends are most crucial features when formulated as prediction task

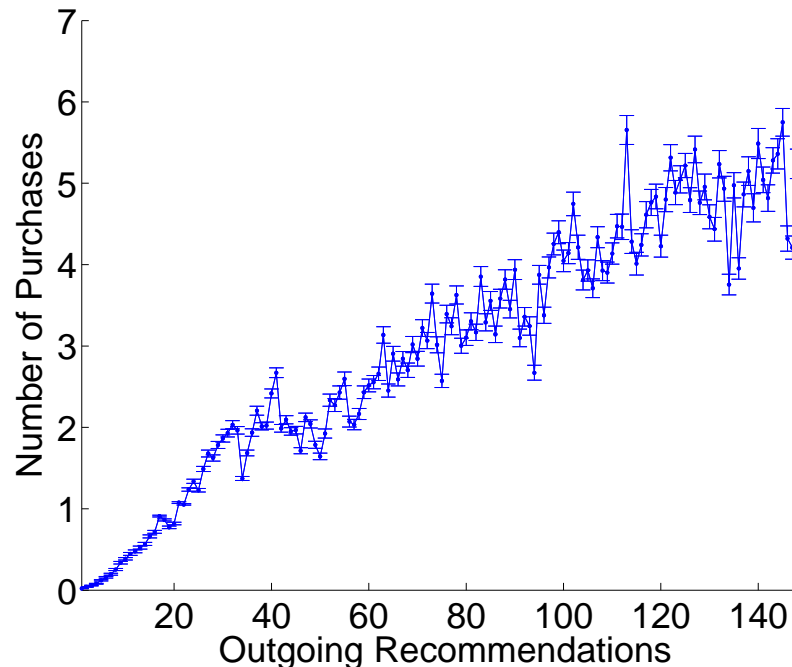
A Puzzle

- If connectedness among friends promotes joining, do highly “clustered” groups grow more quickly?
 - Define clustering = # triangles / # open triads
 - Look at growth from t_1 to t_2 as a function of clustering
 - Groups with large clustering grow slower
 - But not just because clustered groups had fewer nodes one step away



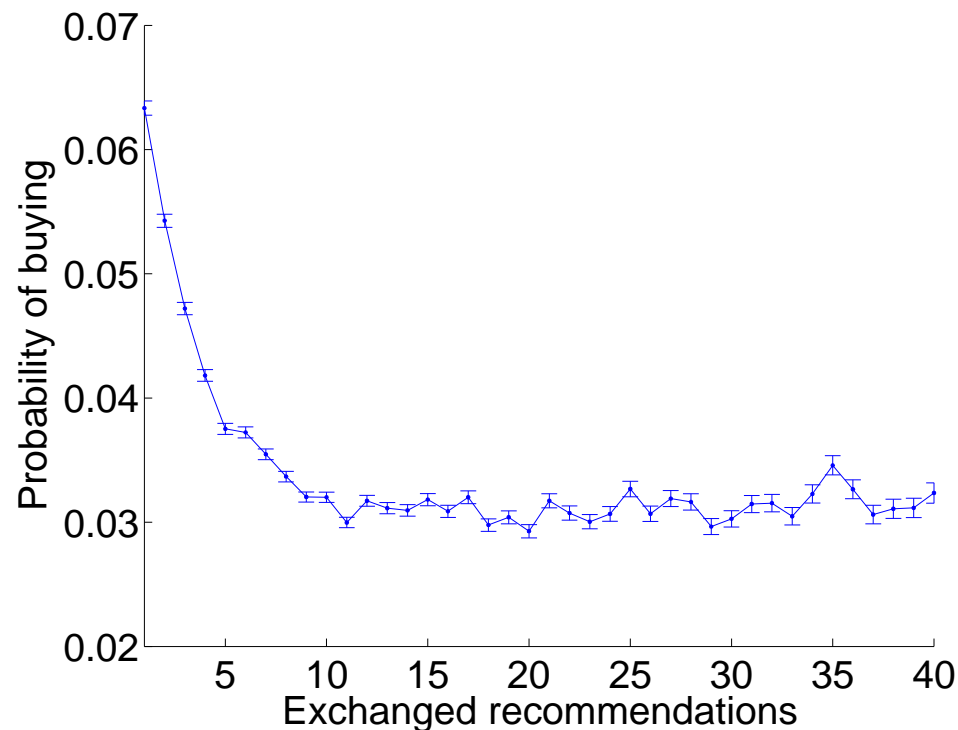
More subtle features: Viral marketing

- Does sending more recommendations influence more purchases?



More subtle features: Viral marketing

- What is the effectiveness of subsequent recommendations?

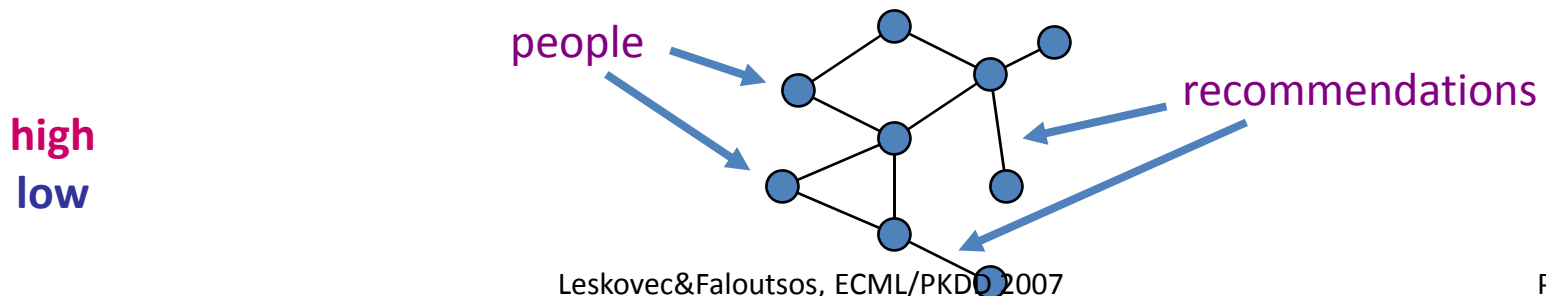


More subtle features: Viral marketing

- What role does the product category play?

[Leskovec-Adamic-Huberman 2006]

	products	customers	recommendations	edges	buy + get discount	buy + no discount
Book	103,161	2,863,977	5,741,611	2,097,809	65,344	17,769
DVD	19,829	805,285	8,180,393	962,341	17,232	58,189
Music	393,598	794,148	1,443,847	585,738	7,837	2,739
Video	26,131	239,583	280,270	160,683	909	467
Full	542,719	3,943,084	15,646,121	3,153,676	91,322	79,164



Cascading of Recommendations

- Some products are easier to recommend than others

product category	number of buy bits	forward recommendations	percent
Book	65,391	15,769	24.2
DVD	16,459	7,336	44.6
Music	7,843	1,824	23.3
Video	909	250	27.6
Total	90,602	25,179	27.8

Viral Marketing: More subtleties

- 47,000 customers responsible for the 2.5 out of 16 million recommendations in the system
- 29% success rate per recommender of an anime DVD
- Giant component covers 19% of the nodes
- Overall, recommendations for DVDs are more likely to result in a purchase (7%), but the anime community stands out

Predicting recommendation success

Variable	transformation	Coefficient
const		-0.940 ***
# recommendations	$\ln(r)$	0.426 ***
# senders	$\ln(n_s)$	-0.782 ***
# recipients	$\ln(n_r)$	-1.307 ***
product price	$\ln(p)$	0.128 ***
# reviews	$\ln(v)$	-0.011 ***
avg. rating	$\ln(t)$	-0.027 *
R^2		0.74

significance at the 0.01 (***), 0.05 (**) and 0.1 (*) levels

Viral Marketing: Why?

- Viral marketing successfully utilizes social networks for adoption of some services
- Hotmail gains 18 million users in 12 months, spending only \$50,000 on traditional advertising
- GMail rapidly gains users although referrals are the only way to sign up
- Customers becoming less susceptible to mass marketing
- Mass marketing impractical for unprecedented variety of products online
- Google AdSense helps sellers reach buyers with targeted advertising
- But how do buyers get good recommendations?

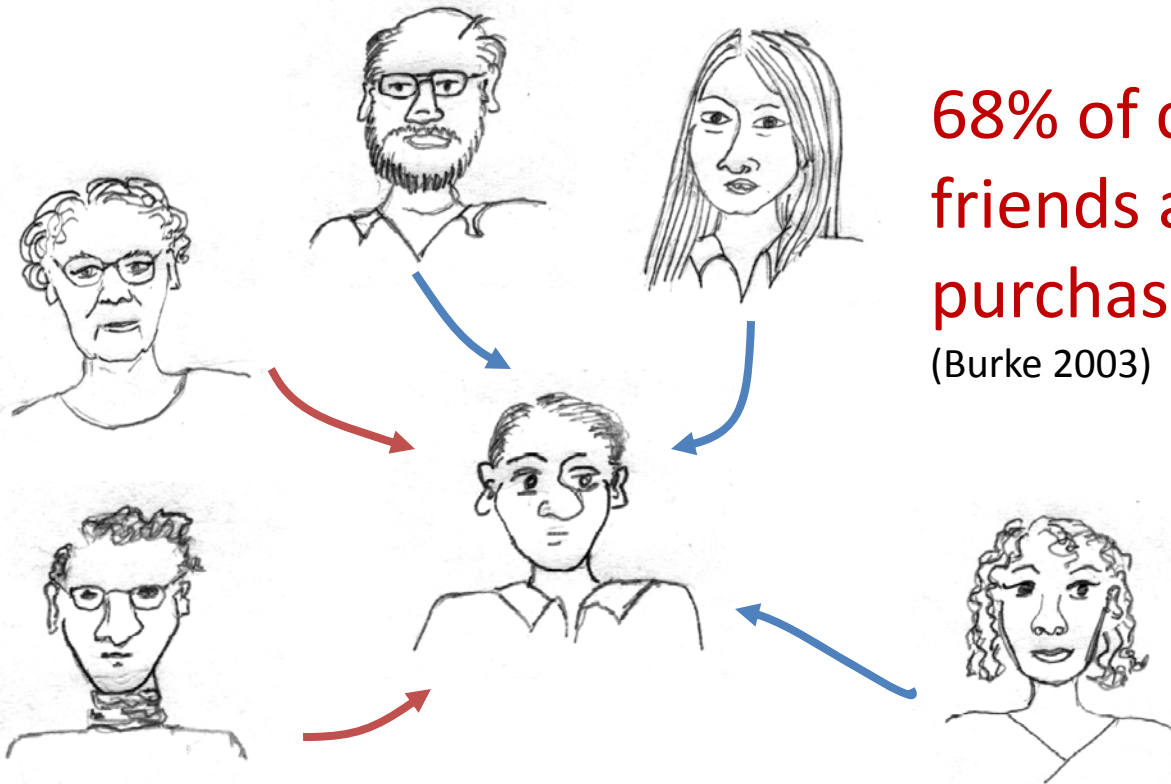
How do people get recommendations?

- > 50% of people do research online before purchasing electronics
- Personalized recommendations based on prior purchase patterns and ratings
- Amazon, “people who bought x also bought y”
- MovieLens, “based on ratings of users like you...”
- Is there still room for viral marketing?

Is there still room for Viral Marketing?

- We are more influenced by our friends than strangers

68% of consumers consult friends and family before purchasing home electronics
(Burke 2003)



Viral Marketing: Not spreading virally

- 94% of users make first recommendation without having received one previously
- Size of giant connected component increases from 1% to 2.5% of the network (100,420 users) – small!
- Some sub-communities are better connected
 - 24% out of 18,000 users for westerns on DVD
 - 26% of 25,000 for classics on DVD
 - 19% of 47,000 for anime (Japanese animated film) on DVD
- Others are just as disconnected
 - 3% of 180,000 home and gardening
 - 2-7% for children's and fitness DVDs

Viral Marketing: Consequences

Products suited for Viral Marketing

- small and tightly knit community
 - few reviews, senders, and recipients
 - but sending more recommendations helps
- pricey products
- rating doesn't play as much of a role

Observations for future diffusion models

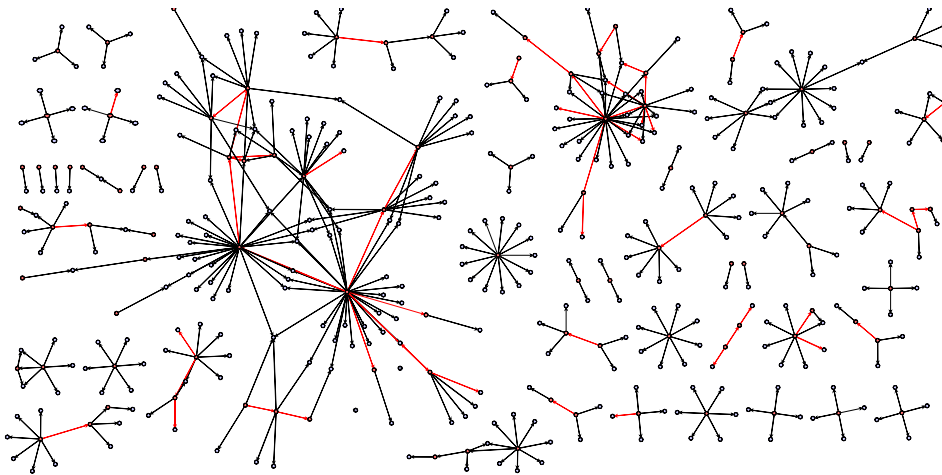
- purchase decision more complex than threshold or simple infection
- influence saturates as the number of contacts expands
- links user effectiveness if they are overused

Conditions for successful recommendations

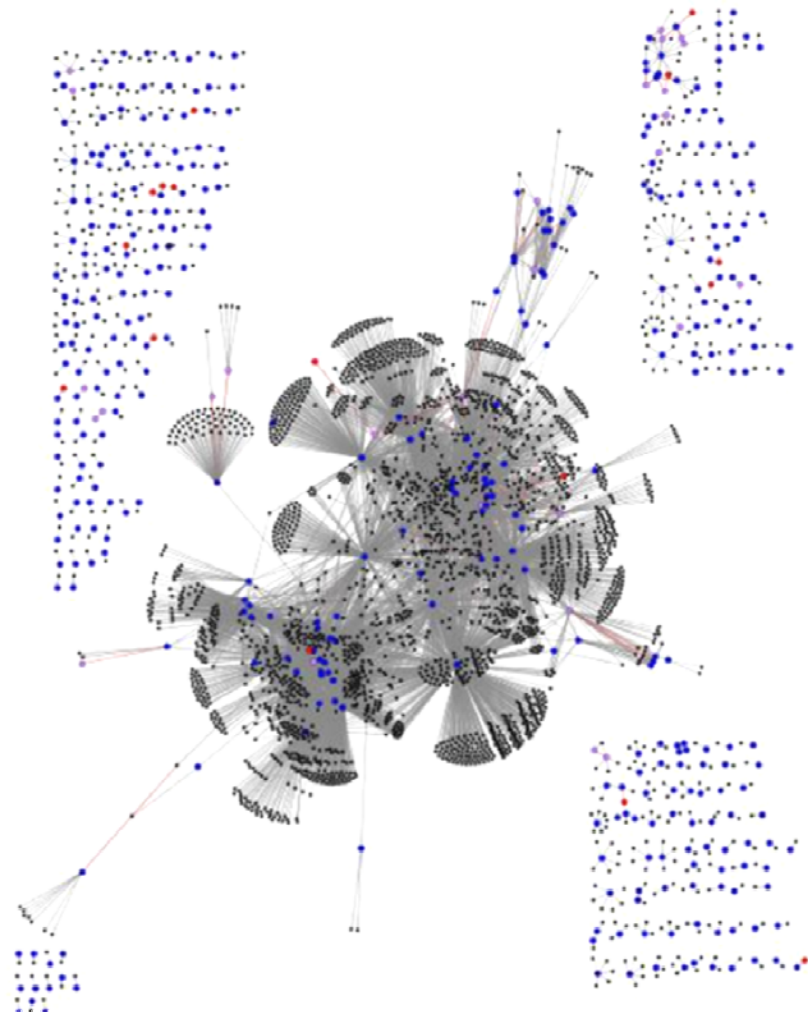
- professional and organizational contexts
- discounts on expensive items
- small, tightly knit communities

How Do Cascades Look Like?

- How big are cascades?
- What are the building blocks of cascades?



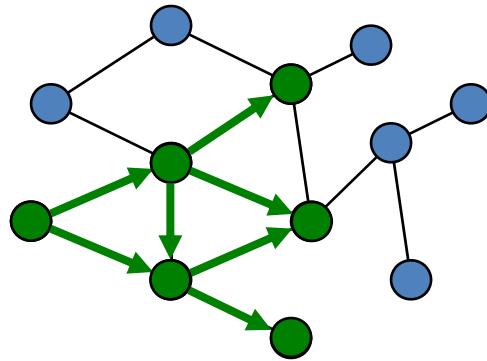
Medical guide book



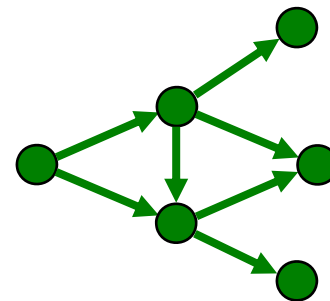
DVD

Cascades as Graphs

- Given a (social) network
- A process by spreading over the network creates a graph (a tree)



Social network



Cascade

(propagation graph)

Let's count cascades

Viral Marketing: Frequent Cascades

●→● is the most common cascade subgraph

- It accounts for ~75% cascades in books, CD and VHS, only 12% of DVD cascades

●→● is 6 (1.2 for DVD) times more frequent than ●→●

- For DVDs ●→● is more frequent than ●→●

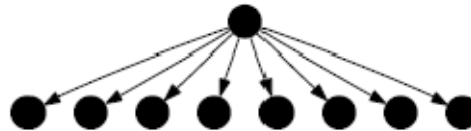
- Chains (●→●→●) are more frequent than ●→●

- is more frequent than a collision (●→●) (but collision has **less edges**)

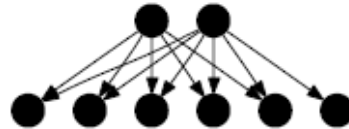
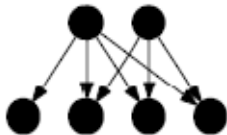
- Late split (●→●→●) is more frequent than ●→●→●

Viral Marketing Cascades

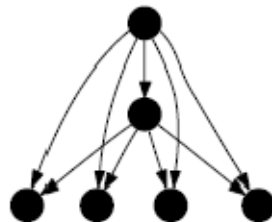
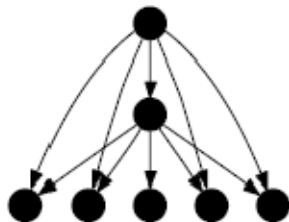
- Stars (“no propagation”)



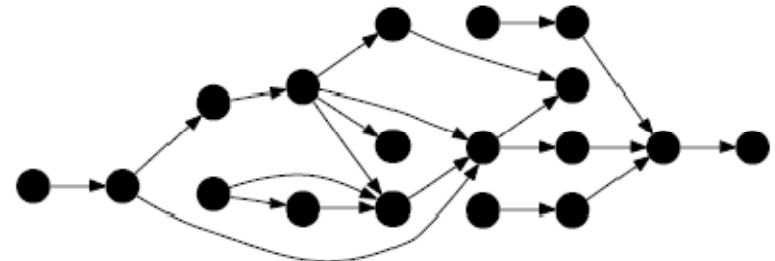
- Bipartite cores (“common friends”)



- Nodes having same friends

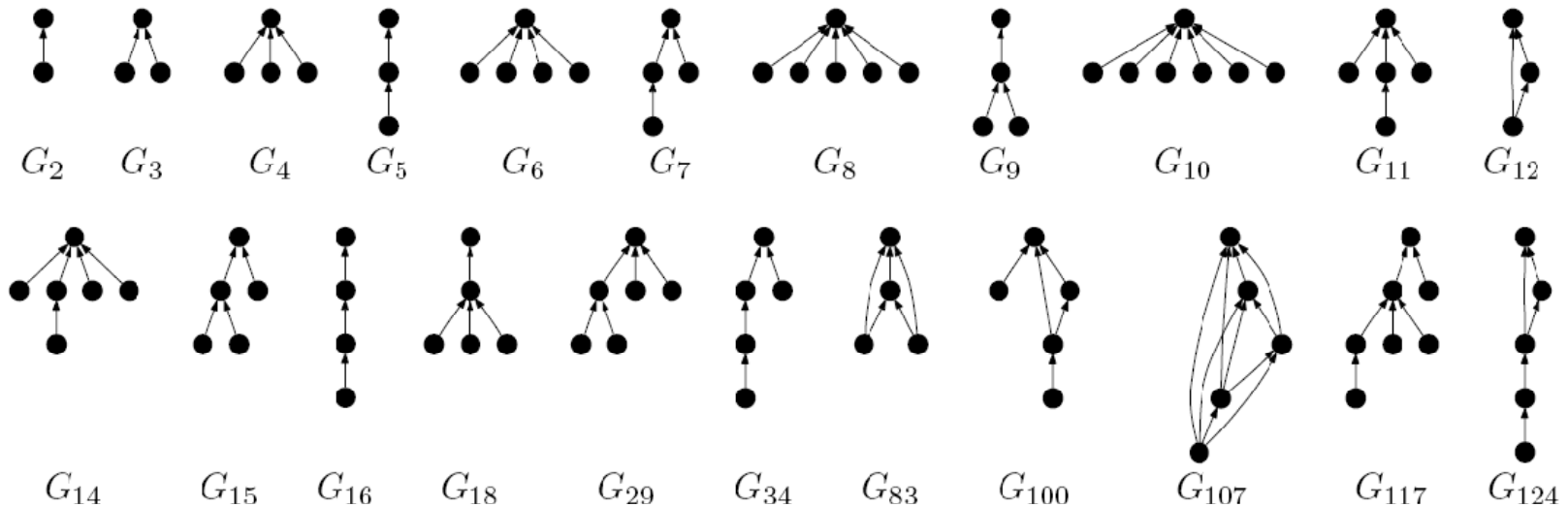


- A complicated cascade







Information Cascades in Blogs

- Cascade shapes (ordered by frequency)

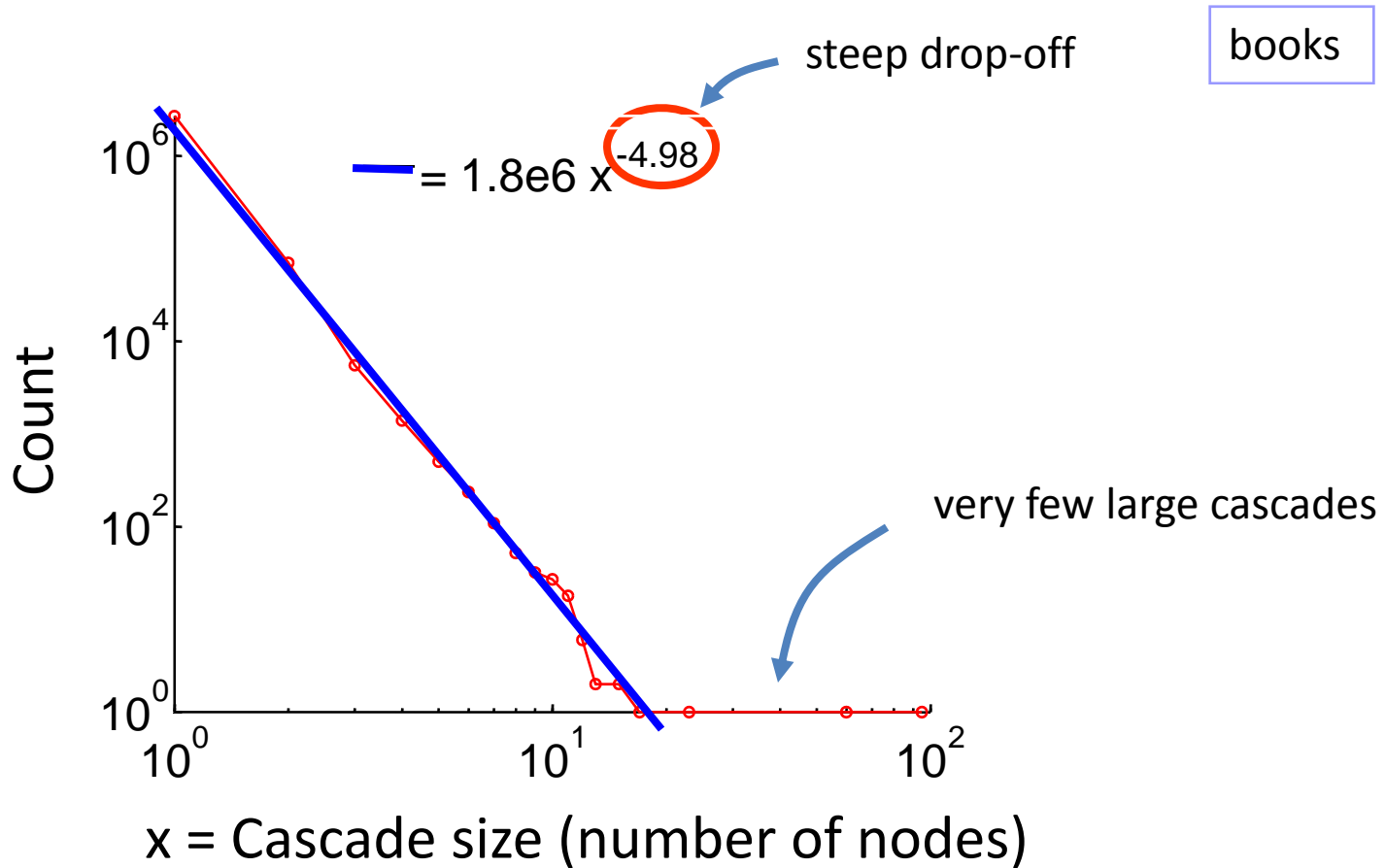


- Cascades are mainly **stars** (trees)
- Interesting relation between the cascade frequency and structure

Cascades: Shape and Frequency

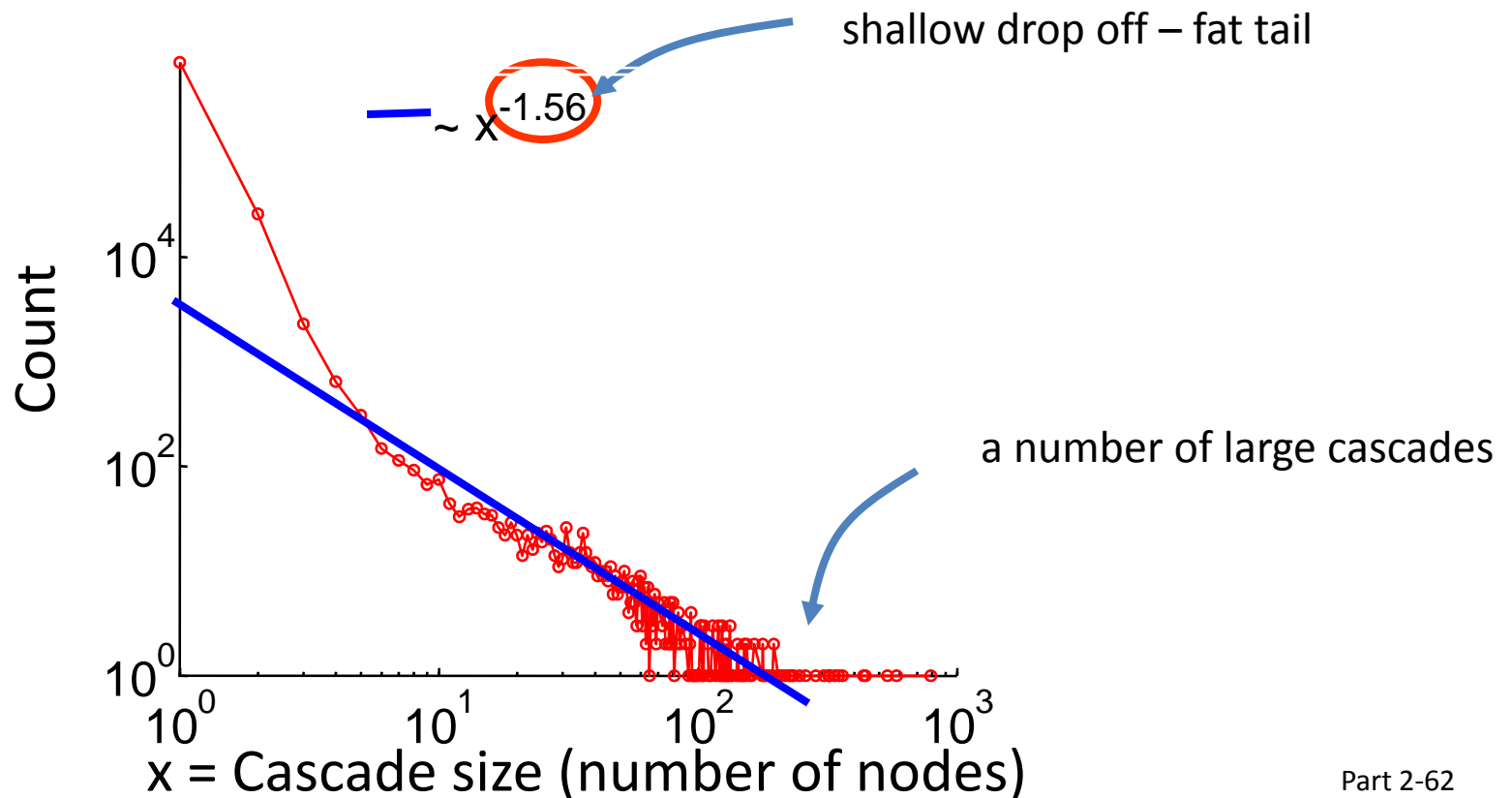
- How do the
 - social context
 - the cascade shape and
 - frequencyrelate?
- What are characteristics that determine cascade frequency?
- Why is it the case that
 -  is more frequent than 
 -  is more frequent than 

Cascade Size: Viral Marketing – Books

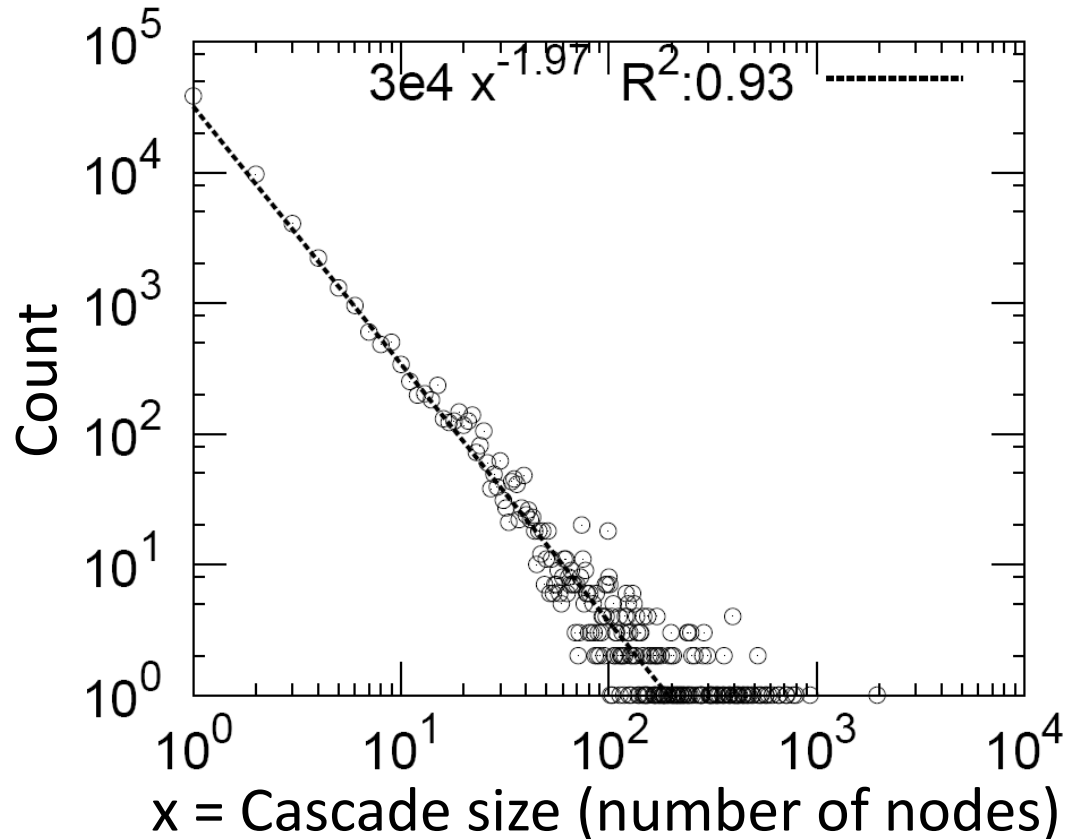


Cascade Size: Viral Marketing – DVDs

- DVD cascades can grow large
- Possibly as a result of websites where people sign up to exchange recommendations



Cascade Size: Blogs



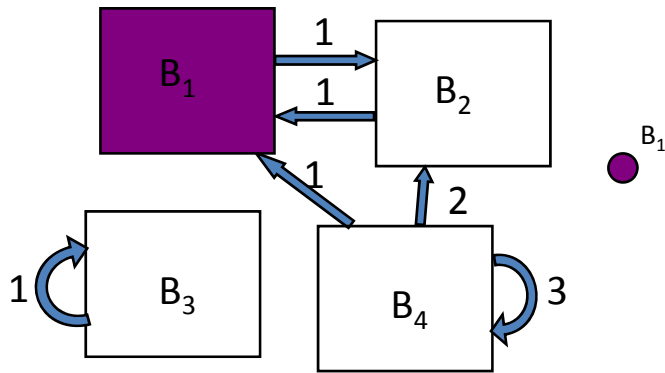
- The probability of observing a cascade on x nodes follows a Zipf distribution: $p(x) \sim x^{-2}$

Cascade Size: Consequences

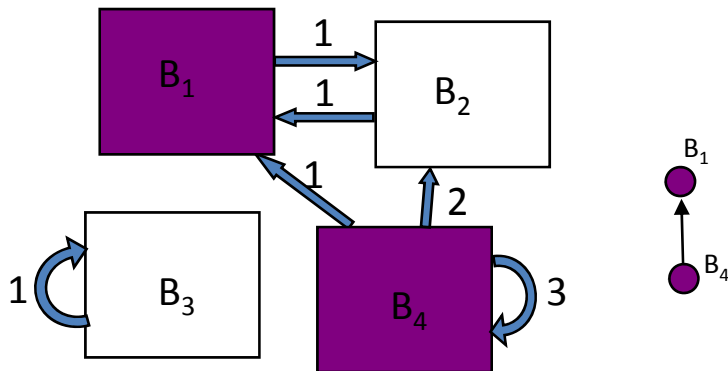
- Cascade sizes follow a heavy-tailed distribution
 - Viral marketing:
 - Books: steep drop-off: power-law exponent -5
 - DVDs: larger cascades: exponent -1.5
 - Blogs:
 - Zipf's law: power-law exponent -2
- However, it is not a branching process
 - A simple branching process (a on k -ary tree):
 - Every node infects each of k of its neighbors with prob. p gives exponential cascade size distribution
- Questions:
 - What role does the underlying social network play?
 - Can make a step towards more realistic cascade generation (propagation) model?

Towards a Better Cascade Model

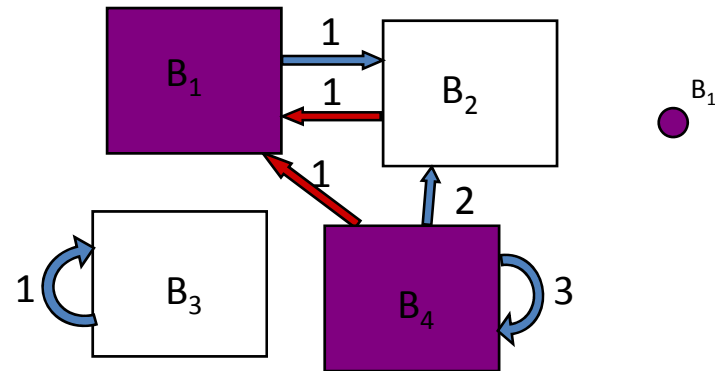
1) Randomly pick blog to infect, add to cascade.



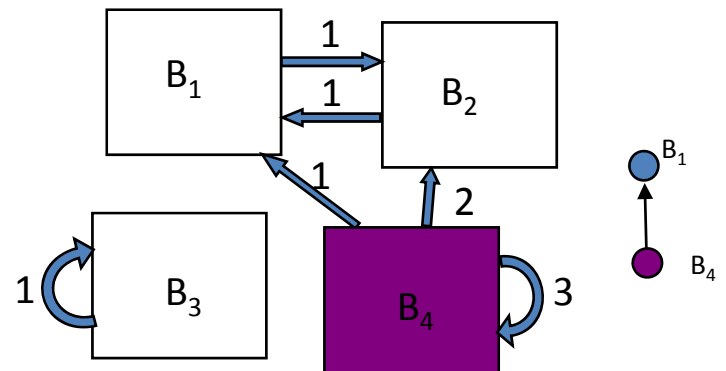
3) Add infected neighbors to cascade.



2) Infect each in-linked neighbor with probability β .



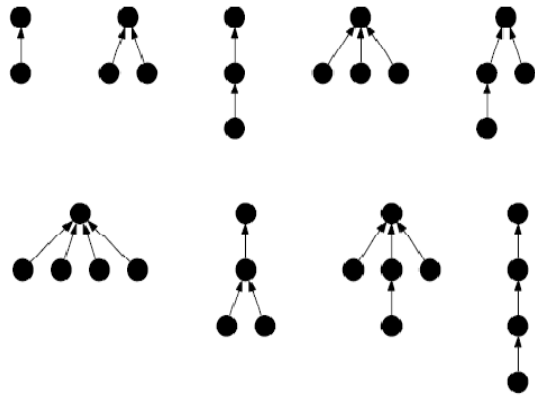
4) Set node infected in (i) to uninfected.



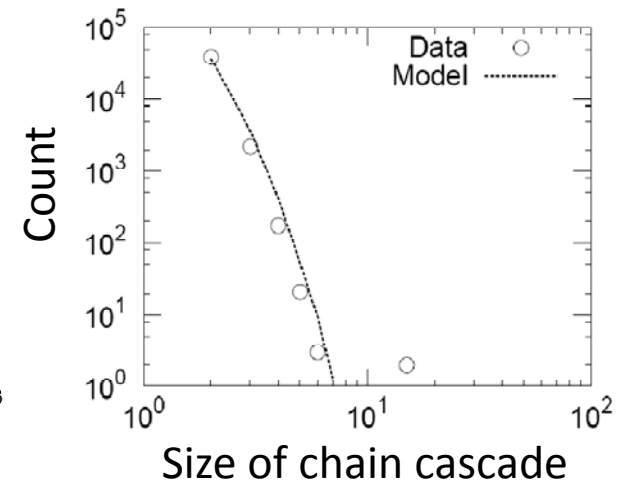
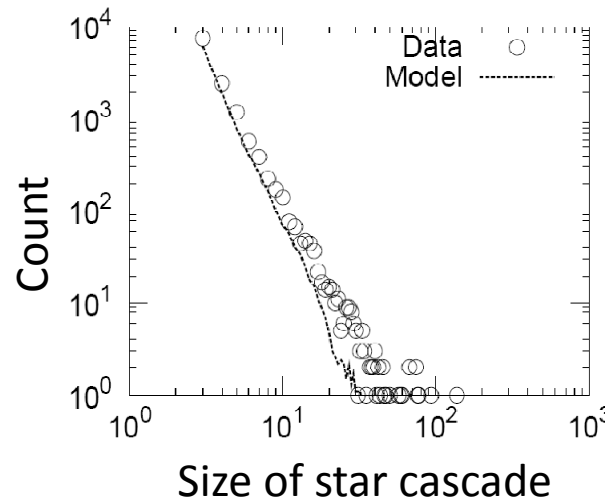
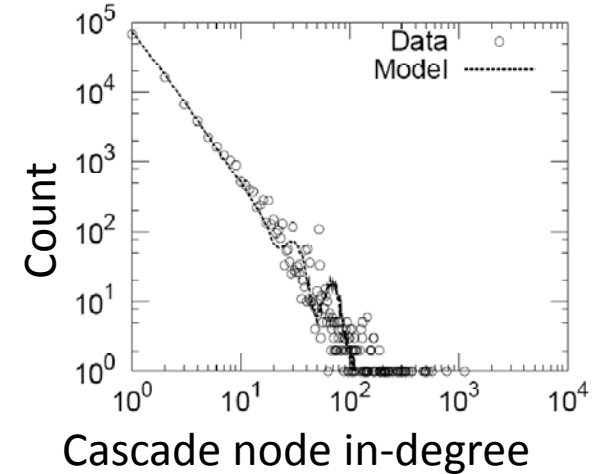
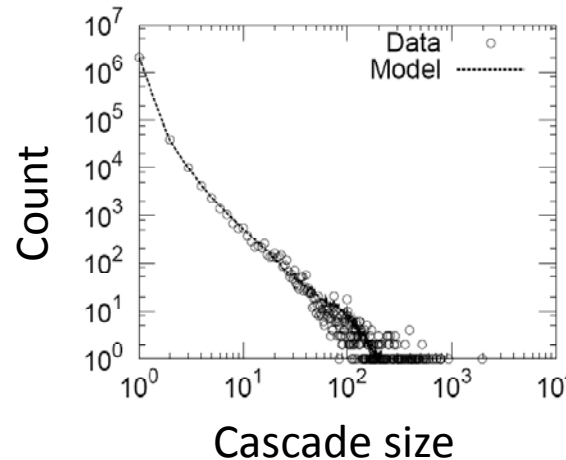
Cascade Model: Results

Generative model
produces realistic
cascades

$$\beta=0.025$$



Most frequent cascades



Part 3: Detecting Cascades, Outbreaks

[Leskovec, Krause, Guestrin, Faloutsos, Glance, VanBriesen 2007]:

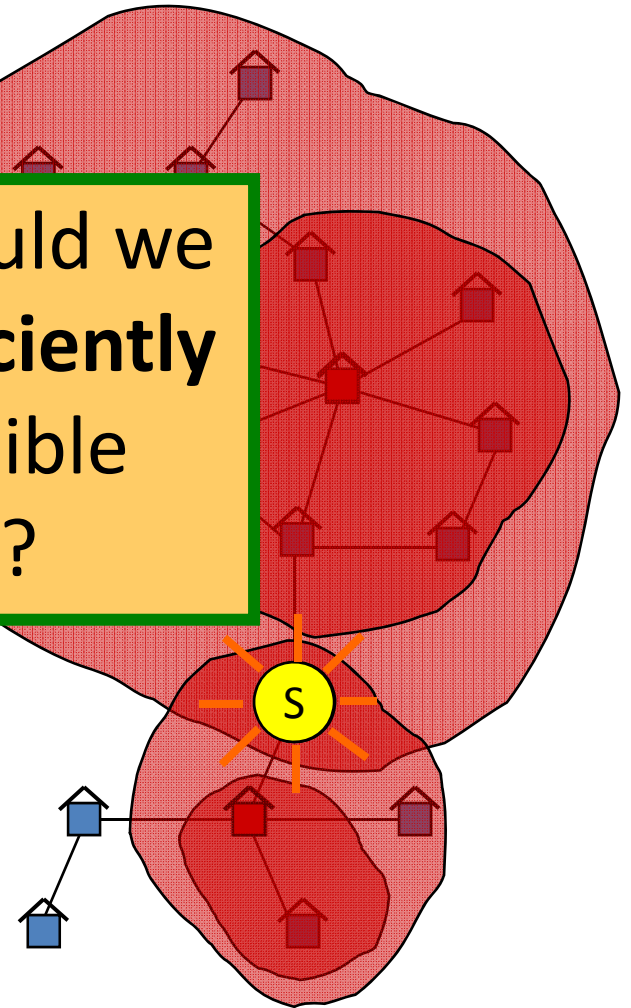
- Given
 - a network
 - and a set of cascades
- Which nodes shall we monitor to detect cascades (outbreaks) effectively?

Scenario 1: Water Network

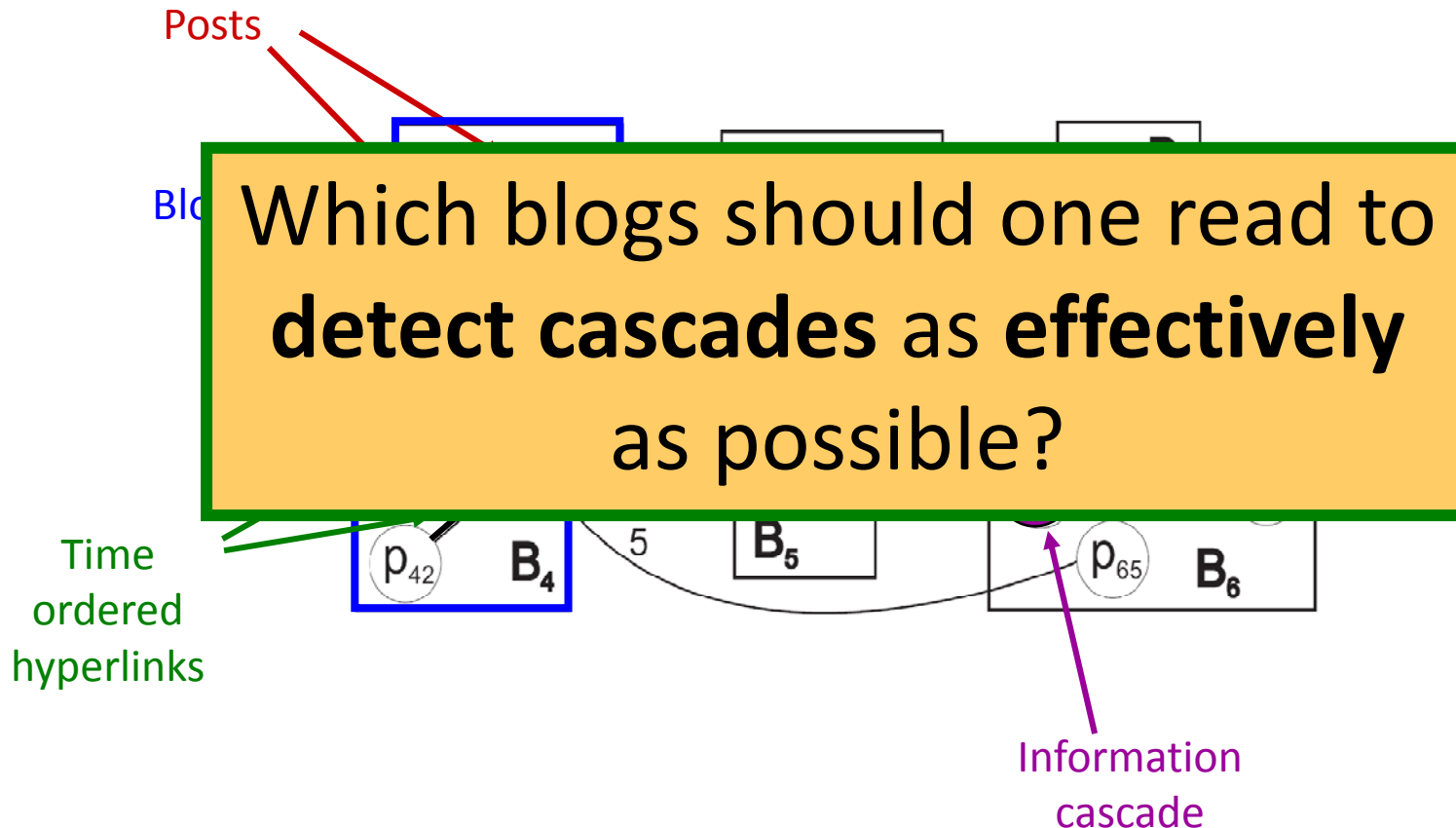
- Given a real city water distribution network
- And data on past contaminations in the network
- Problem: How can we efficiently detect all possible contaminations?

On which nodes should we place **sensors** to **efficiently** detect the all possible contaminations?

*Environmental
Protection Agency*



Scenario 2: Cascades in Blogs



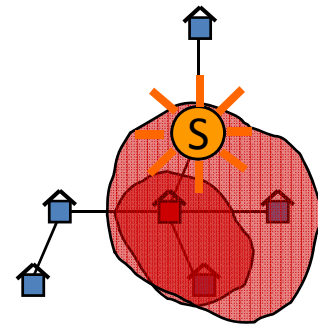
Cascade Detection: General Problem

- Given a dynamic process spreading over the network
- We want to **select a set of nodes to detect the process effectively**
- Note:
 - The problem is different from selecting “influential nodes”
 - We aim to select nodes that are most easily influenced, *i.e.*, cascades (outbreaks) hit them soon
- Many other applications:
 - Epidemics
 - Network security

Two Parts to the Problem

- **Reward**, *e.g.*:
 - 1) Minimize time to detection
 - 2) Maximize number of detected propagations
 - 3) Minimize number of infected people
- **Cost** (location dependent):
 - Reading big blogs is more time consuming
 - Placing a sensor in a remote location is expensive

Problem Setting



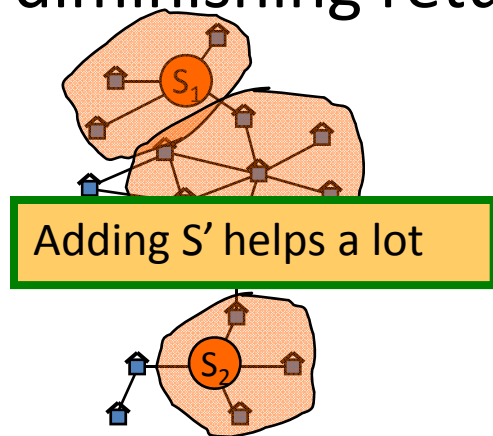
- Given a graph $G(V, E)$
- and a budget B for sensors
- and data on how contaminations spread over the network:
 - for each contamination i we know the time $T(i, u)$ when it contaminated node u
- Select a **subset** of nodes A that **maximize** the **expected reward**

$$\max_{A \subseteq V} R(A) \equiv \sum_i P(i) \underbrace{R_i(T(i, A))}_{\text{Reward for detecting contamination } i}$$

subject to $cost(A) < B$

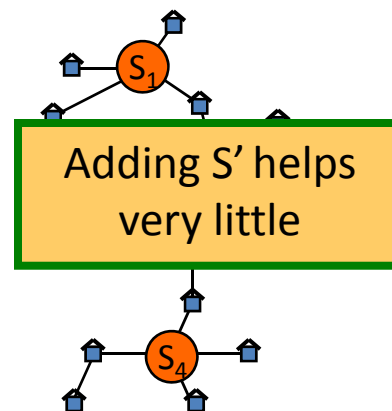
Structure of the Problem

- Solving the problem exactly is **NP-hard**
 - Set cover (or vertex cover)
- Observation:
 - Objective (reward) functions are **submodular**, *i.e.* diminishing returns



Placement $A = \{S_1, S_2\}$

New sensor:



Placement $A = \{S_1, S_2, S_3, S_4\}$

Reward Functions: Submodularity

- For all placement $\mathcal{A} \subseteq \mathcal{B} \subseteq \mathcal{V}$ it holds
$$\underbrace{R(\mathcal{A} \cup \{s\}) - R(\mathcal{A})}_{\text{Benefit of adding a sensor to a small placement}} \geq \underbrace{R(\mathcal{B} \cup \{s\}) - R(\mathcal{B})}_{\text{Benefit of adding a sensor to a large placement}}$$

- Similar argument as in influence maximization:

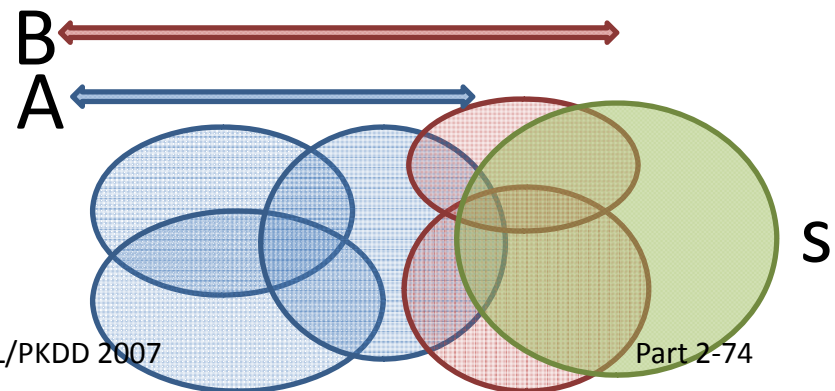
- Linear combinations of submodular functions are

- submodular: $\max_{\mathcal{A} \subseteq \mathcal{V}} R(\mathcal{A}) \equiv \sum_i P(i) R_i(T(i, \mathcal{A}))$

- Individual functions

- R_i are submodular:

- Size of the union of sets



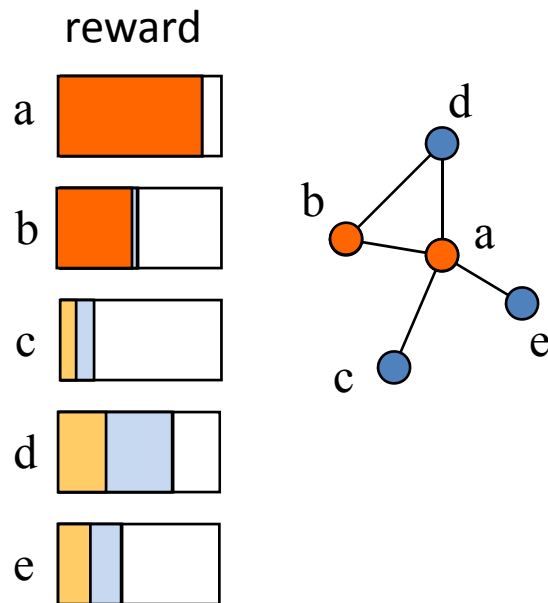
Reward Functions are Submodular

- Objective functions from Battle of Water Sensor Networks competition [Ostfeld et al]:
 - 1) Time to detection (DT)
 - How long does it take to detect a contamination?
 - 2) Detection likelihood (DL)
 - How many contaminations do we detect?
 - 3) Population affected (PA)
 - How many people drank contaminated water?

are all **submodular**

Background: Submodular functions

Hill-climbing



Add sensor with highest
marginal gain

What do we know about optimizing submodular functions?

- A hill-climbing (*i.e.*, greedy) is near optimal ($1 - 1/e$ ($\sim 63\%$) of optimal)
- But
 - 1) this only works for **unit cost** case (each sensor/location costs the same)
 - 2) Hill-climbing algorithm is **slow**
 - At each iteration we need to re-evaluate marginal gains
 - It scales as $O(|V|/B)$

Towards a New Algorithm

- Possible algorithm: **hill-climbing ignoring the cost**
 - Repeatedly select sensor with highest marginal gain
 - Ignore sensor cost
- It always prefers more expensive sensor with reward r to a cheaper sensor with reward $r - \epsilon$
 - For variable cost it can **fail arbitrarily badly**
- **Idea:**
 - What if we optimize **benefit-cost ratio**?

$$s_k = \operatorname{argmax}_{s \in \mathcal{V} \setminus \mathcal{A}_{k-1}} \frac{R(\mathcal{A}_{k-1} \cup \{s\}) - R(\mathcal{A}_{k-1})}{c(s)}$$

More Problems with Benefit-Cost

- Bad news: Optimizing benefit-cost ratio can fail arbitrarily badly
- Example: Given a budget B , consider:
 - 2 locations s_1 and s_2 :
 - Costs: $c(s_1)=\epsilon$, $c(s_2)=B$
 - Only 1 cascade with reward: $R(s_1)=2\epsilon$, $R(s_2)=B$
 - Then benefit-cost ratio is
 - $bc(s_1)=2$ and $bc(s_2)=1$
 - So, we first select s_1 and then can not afford s_2
 - We get reward 2ϵ instead of B

Now send ϵ to 0 and we get arbitrarily bad

Solution: CELF Algorithm

- CELF (cost-effective lazy forward-selection) algorithm
 - A two pass greedy algorithm:
 - Set (solution) A: use benefit-cost greedy
 - Set (solution) B: use unit cost greedy
 - Final solution: $\operatorname{argmax}(R(A), R(B))$
- How far is CELF from (unknown) optimal solution?
- Theorem: CELF is near optimal
 - CELF achieves $\frac{1}{2}(1-1/e)$ factor approximation
- CELF is much faster than standard hill-climbing

Tighter Algorithmic Bound

- Traditional bound $(1-1/e)$ tells us:
How far from optimal are we even before seeing the data and running the algorithm
- Can we do better? Yes!
- We develop a new **tighter bound**. Intuition:
 - Marginal gains are decreasing with the solution size
 - We use this to get tighter bound on the solution

Scaling up CELF algorithm

- Observation:

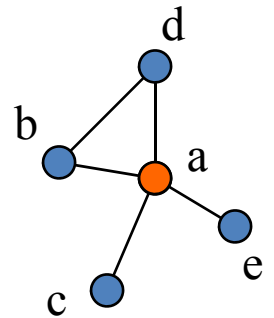
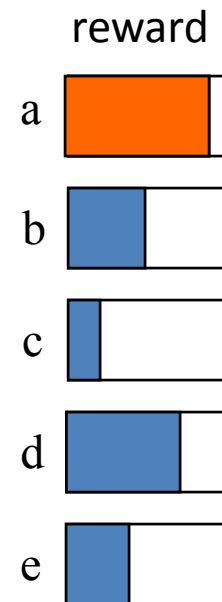
Submodularity guarantees that marginal benefits **decrease** with the solution size



- Idea: exploit submodularity, doing **lazy evaluations!**
(considered by Robertazzi et al. for unit cost case)

Scaling up CELF

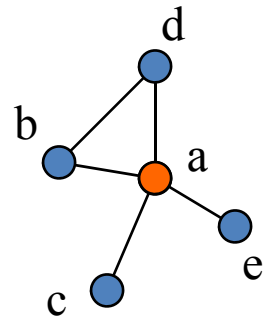
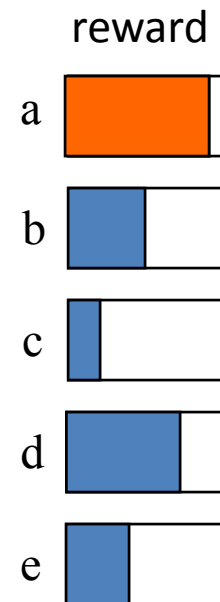
- **CELF** algorithm – **hill-climbing**:
 - Keep an ordered list of marginal benefits b_i from previous iteration
 - Re-evaluate b_i **only** for top sensor
 - Re-sort and prune



Scaling up CELF

- **CELF** algorithm – **hill-climbing**:

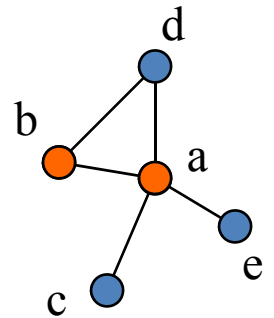
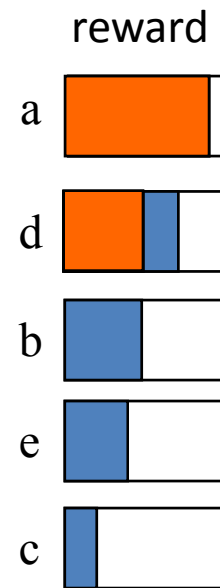
- Keep an ordered list of marginal benefits b_i from previous iteration
- Re-evaluate b_i **only** for top sensor
- Re-sort and prune



Scaling up CELF

- **CELF** algorithm – **hill-climbing**:

- Keep an ordered list of marginal benefits b_i from previous iteration
- Re-evaluate b_i **only** for top sensor
- Re-sort and prune

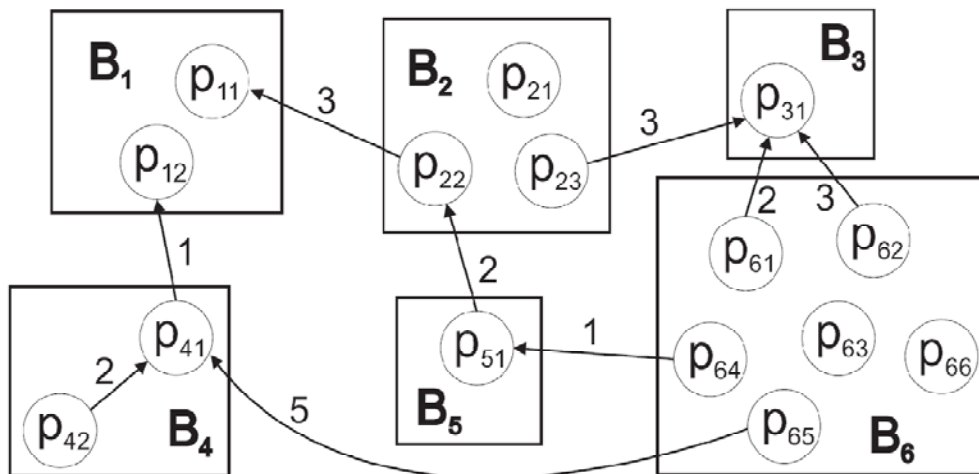


Experiments: 2 Case Studies

- We have **real propagation data**
 - Blog network:
 - We crawled blogs for 1 year
 - We identified cascades – temporal propagation of information
 - Water distribution network:
 - Real city water distribution networks
 - Realistic simulator of water consumption provided by US Environmental Protection Agency

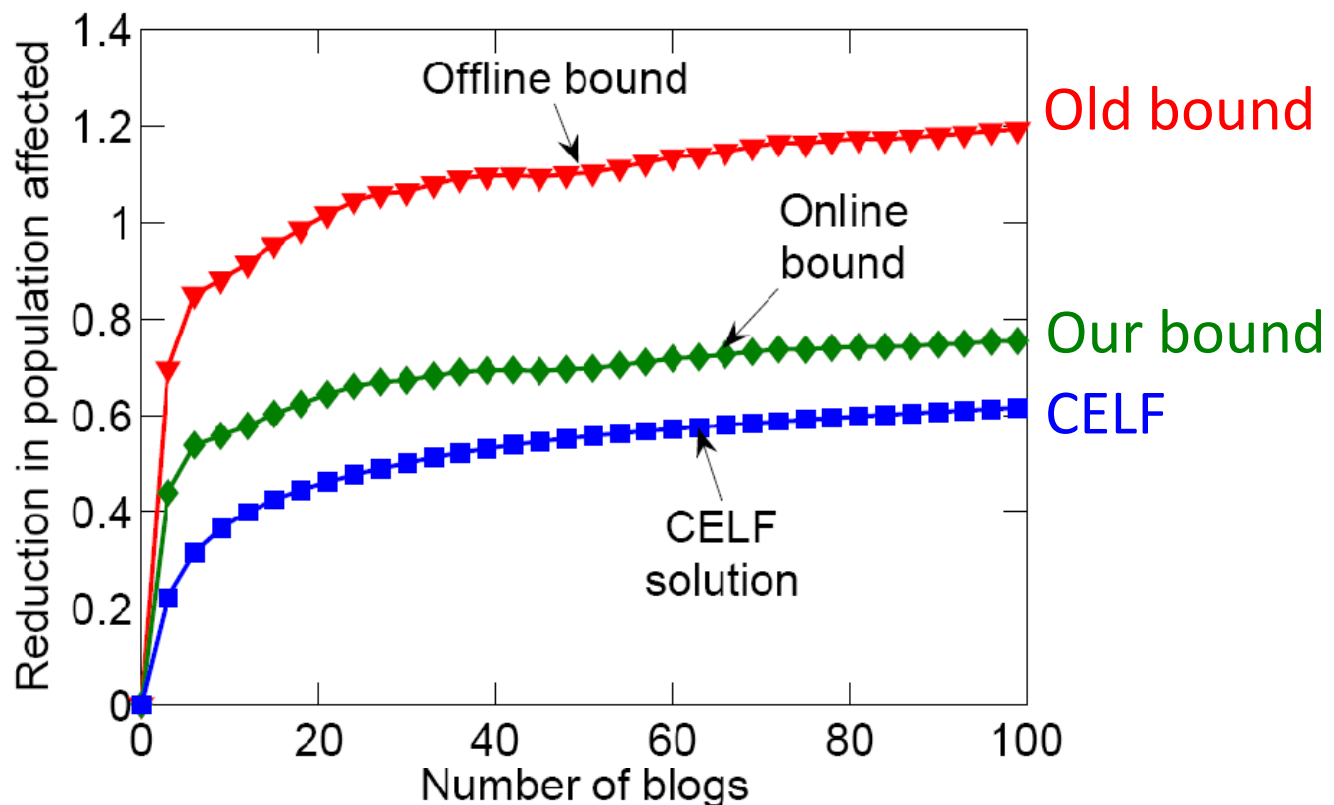
Case study 1: Cascades in Blogs

- (Same data as in part 2 of the talk)
 - We crawled 45,000 blogs for 1 year
 - We obtained 10 million posts
 - And identified 350,000 cascades



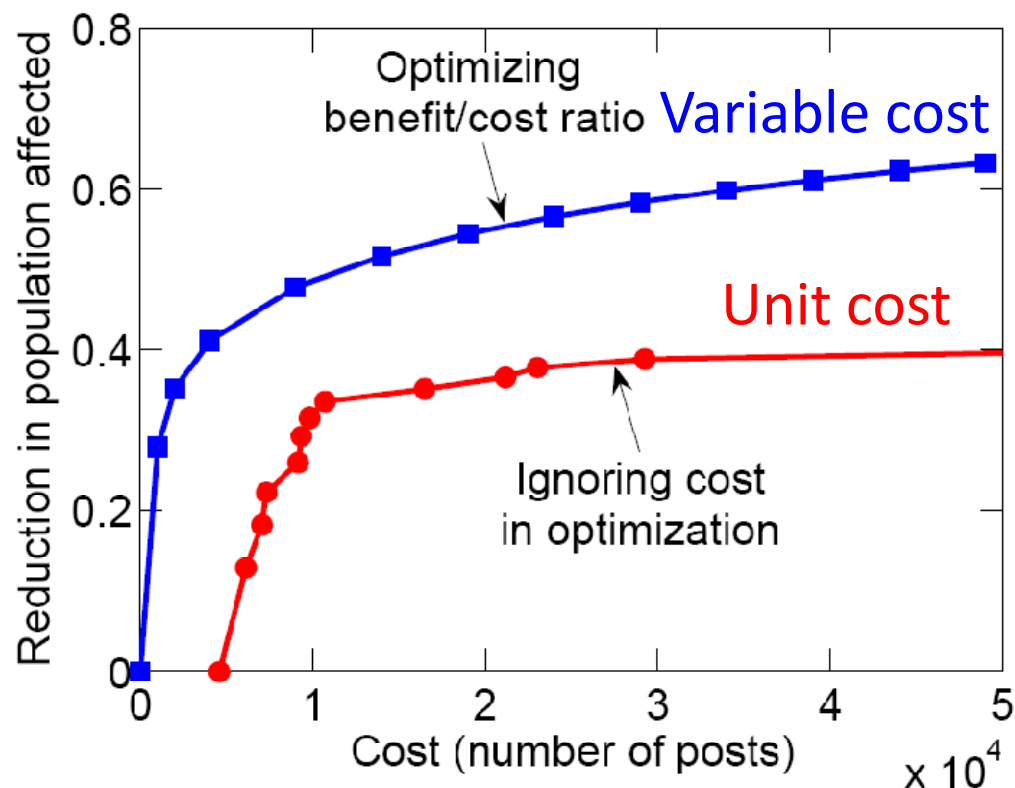
Q1: Blogs: Solution Quality

- Our bound is much tighter
 - 13% instead of 37%



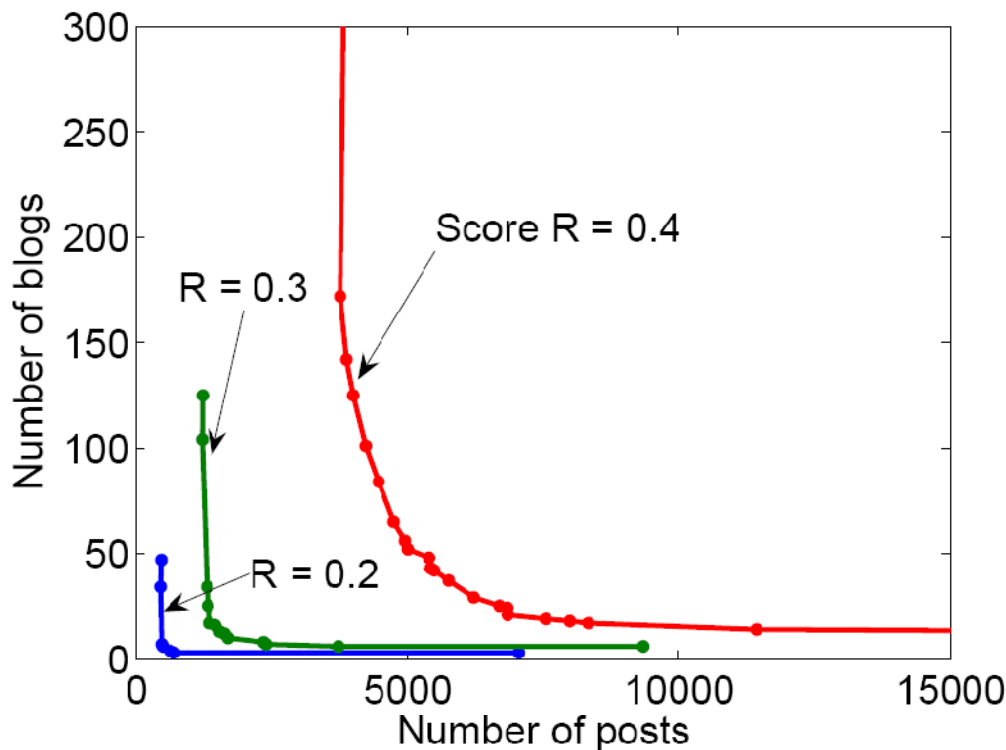
Q2: Blogs: Cost of a Blog

- Unit cost:
 - algorithm picks **large popular blogs**:
instapundit.com,
michellemalkin.com
- Variable cost:
 - proportional to the **number of posts**
- We can do much better when considering costs



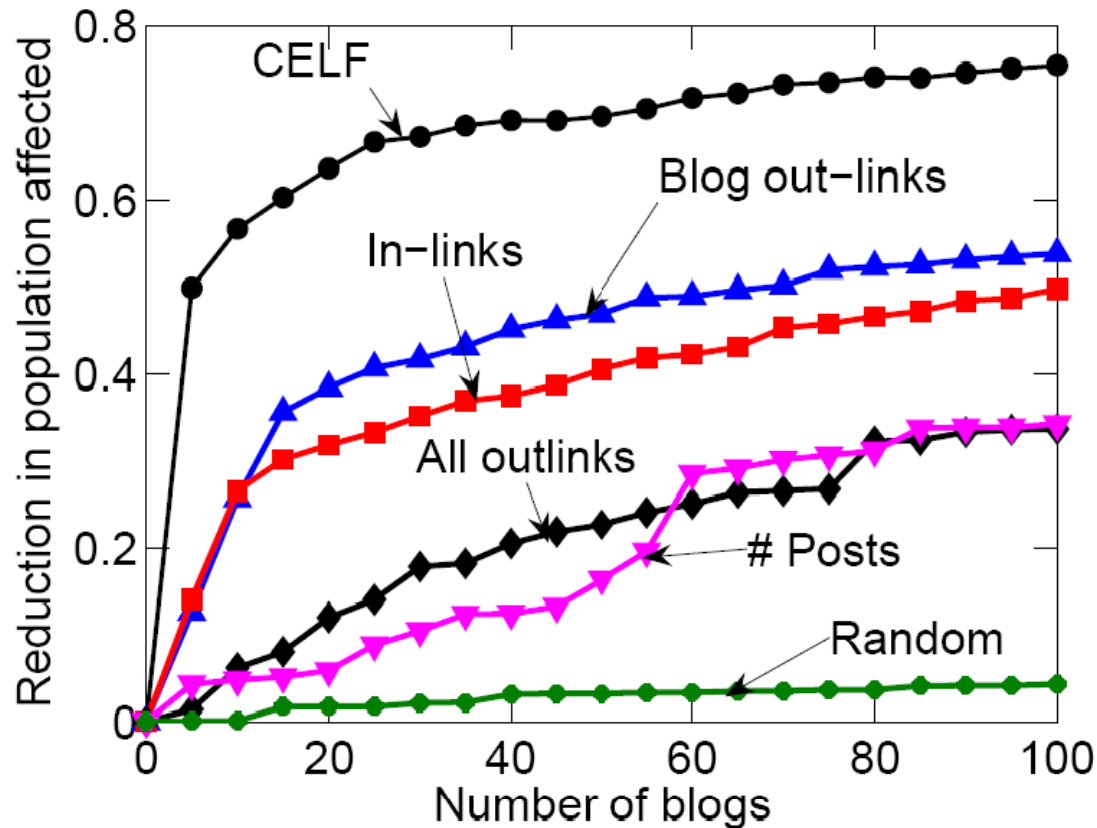
Q2: Blogs: Cost of a Blog

- But then algorithm picks **lots of small blogs** that participate in few cascades
- We pick best solution that interpolates between the costs
- We can get good solutions with **few blogs and few posts**



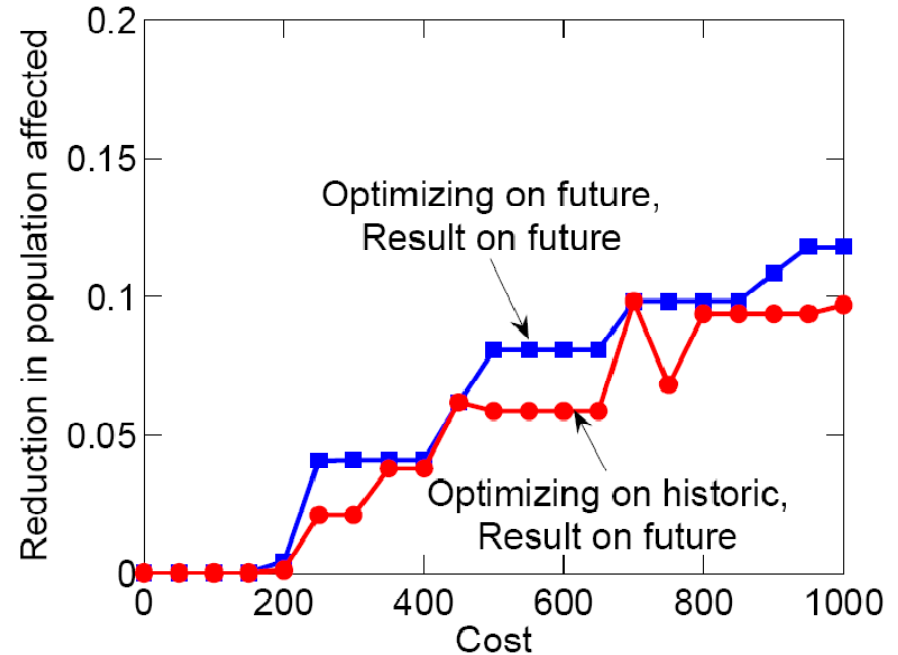
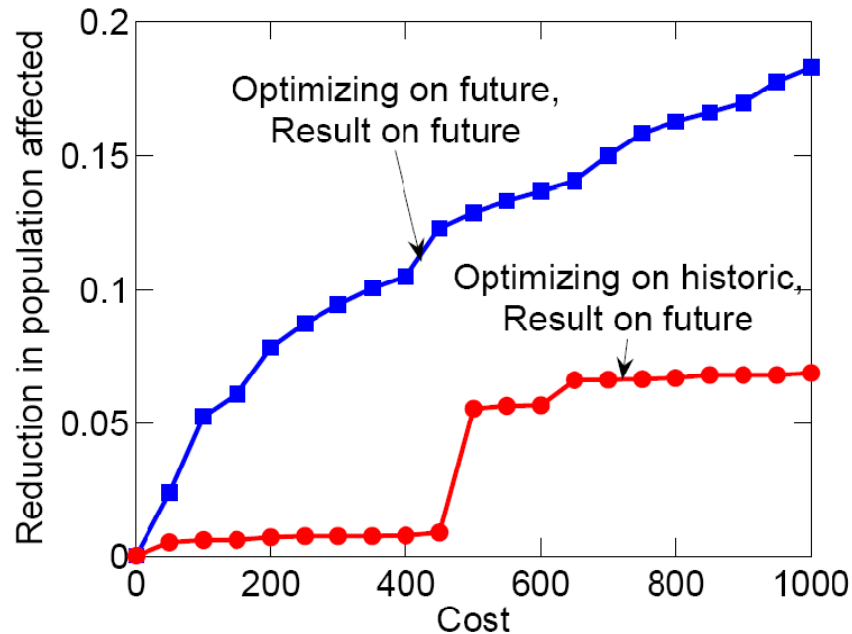
Each curve represents solutions with same final reward

Q4: Blogs: Heuristic Selection



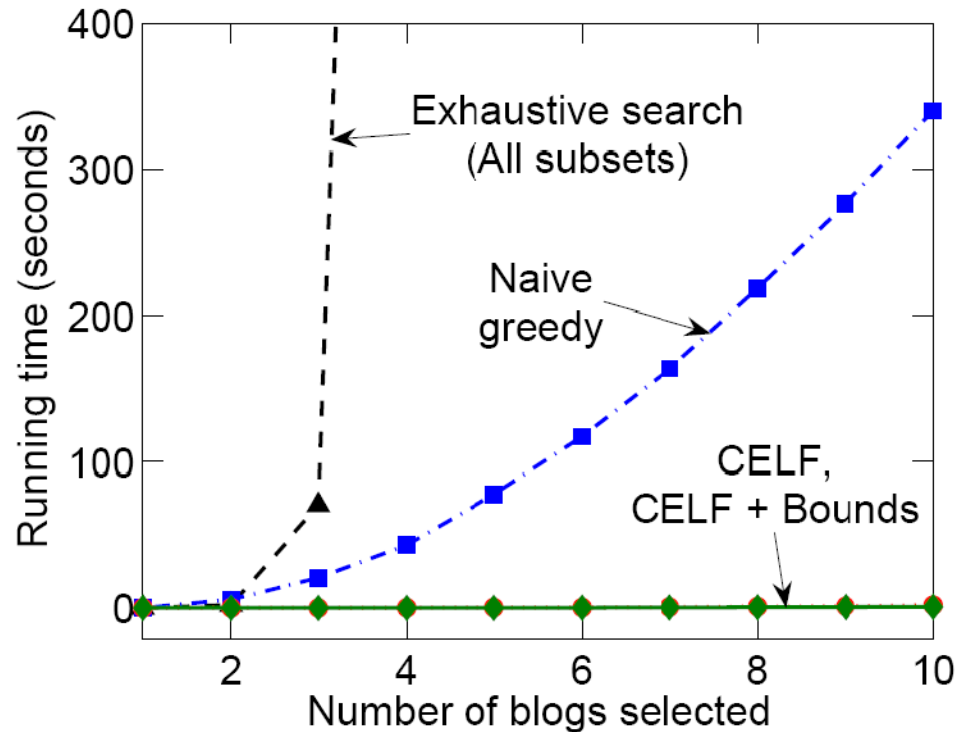
- Heuristics perform much worse
- One really needs to perform optimization

Blogs: Generalization to Future



- We want to generalize well to future (unknown) cascades
- Limiting selection to bigger blogs improves generalization

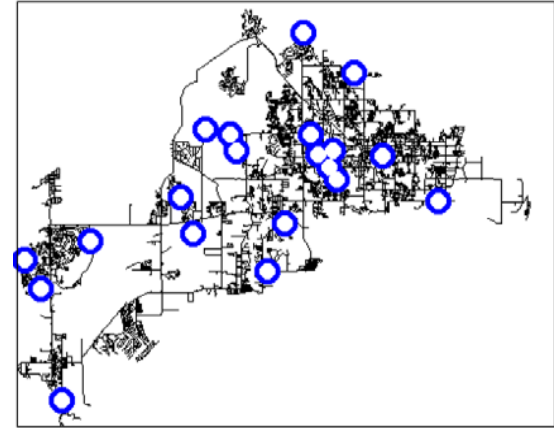
Q5: Blogs: Scalability



- **CELF** runs **700** times faster than simple hill-climbing algorithm

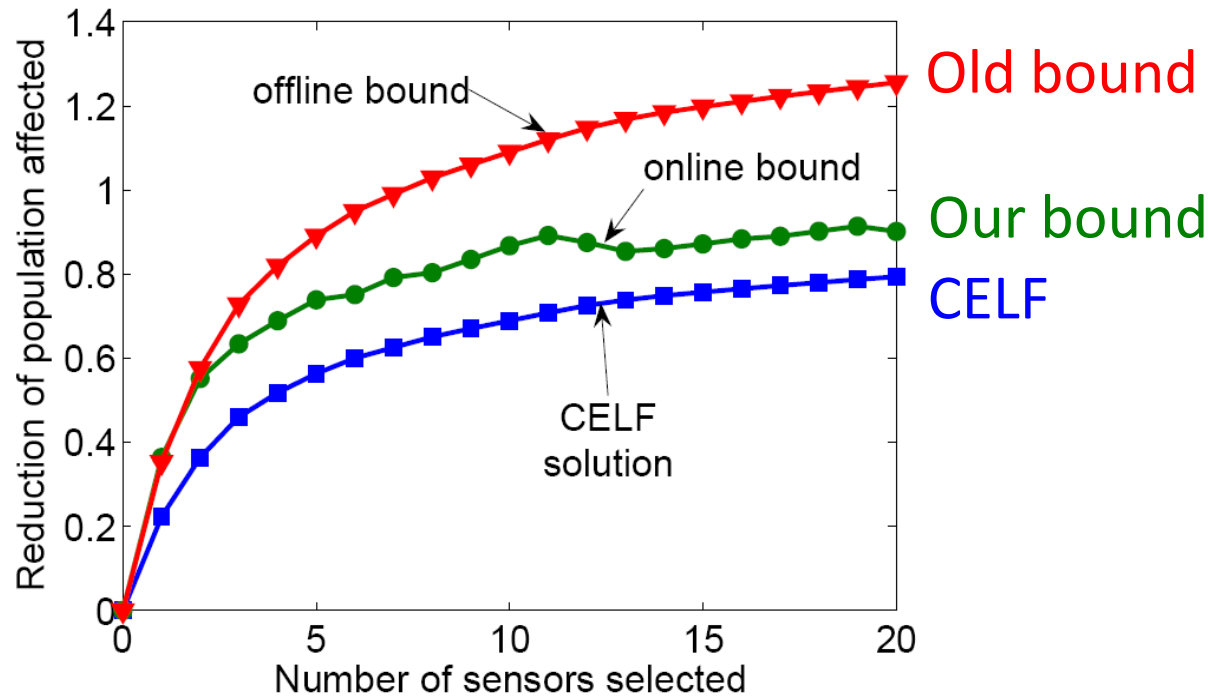
Case study 2: Water Network

- Real metropolitan area water network
 - $V = 21,000$ nodes
 - $E = 25,000$ pipes



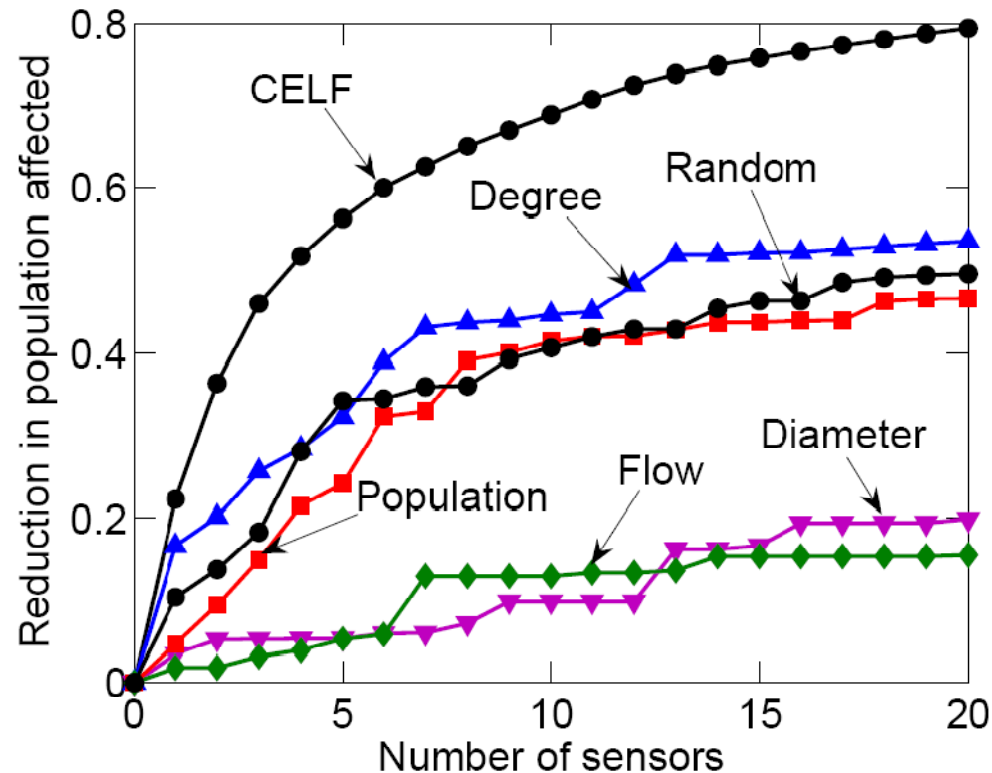
- Use a cluster of 50 machines for a month
- Simulate 3.6 million epidemic scenarios (152 GB of epidemic data)
- By exploiting sparsity we fit it into main memory (16GB)

Water: Solution Quality



- The new bound gives much better estimate of solution quality

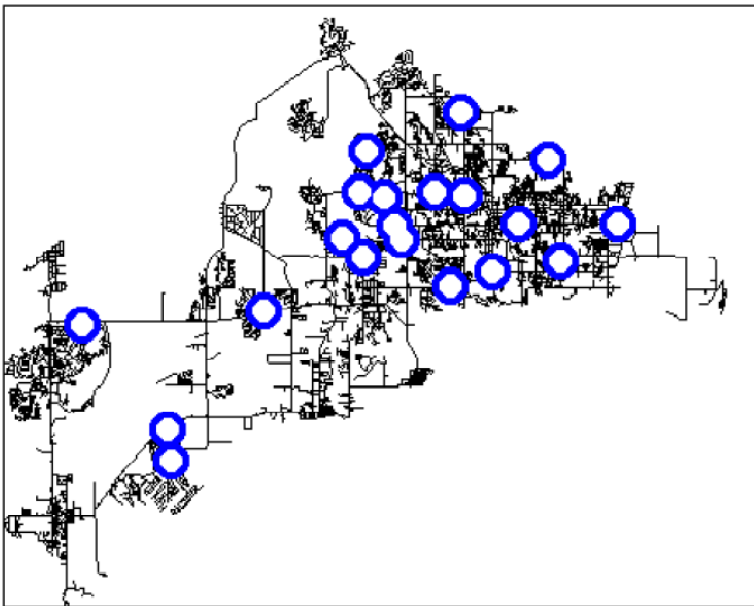
Water: Heuristic Placement



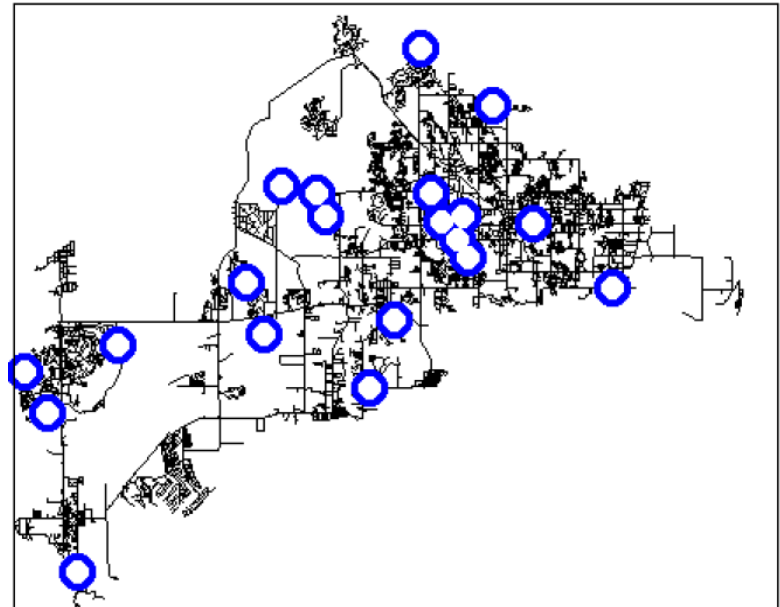
- Heuristics placements perform much worse
- One really needs to consider the spread of epidemics

Water: Placement Visualization

- Different reward functions give different sensor placements

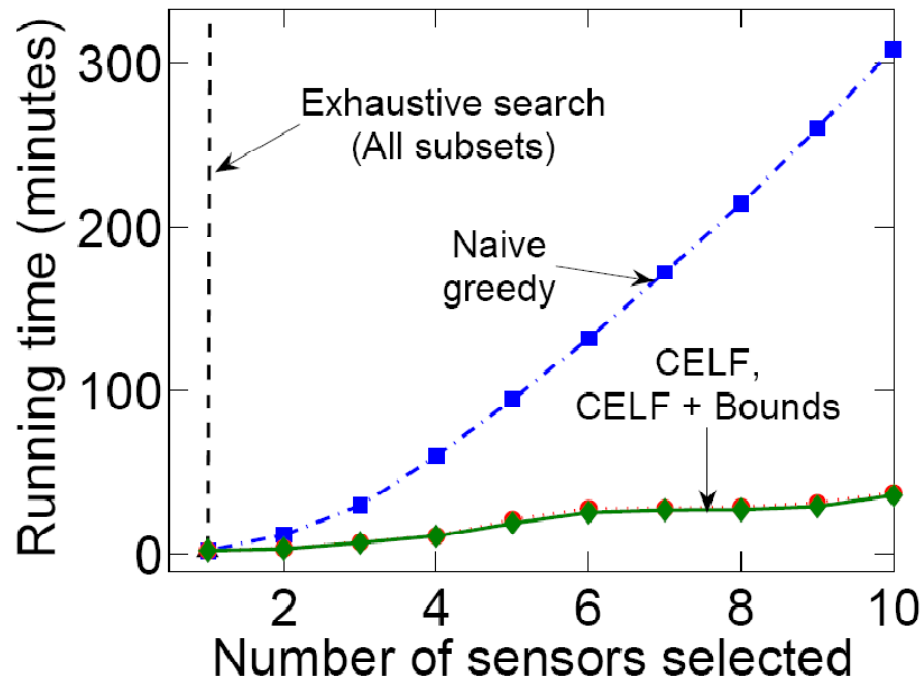


Population affected



Detection likelihood

Water: Algorithm Scalability



- CELF is an order of magnitude faster than hill-climbing

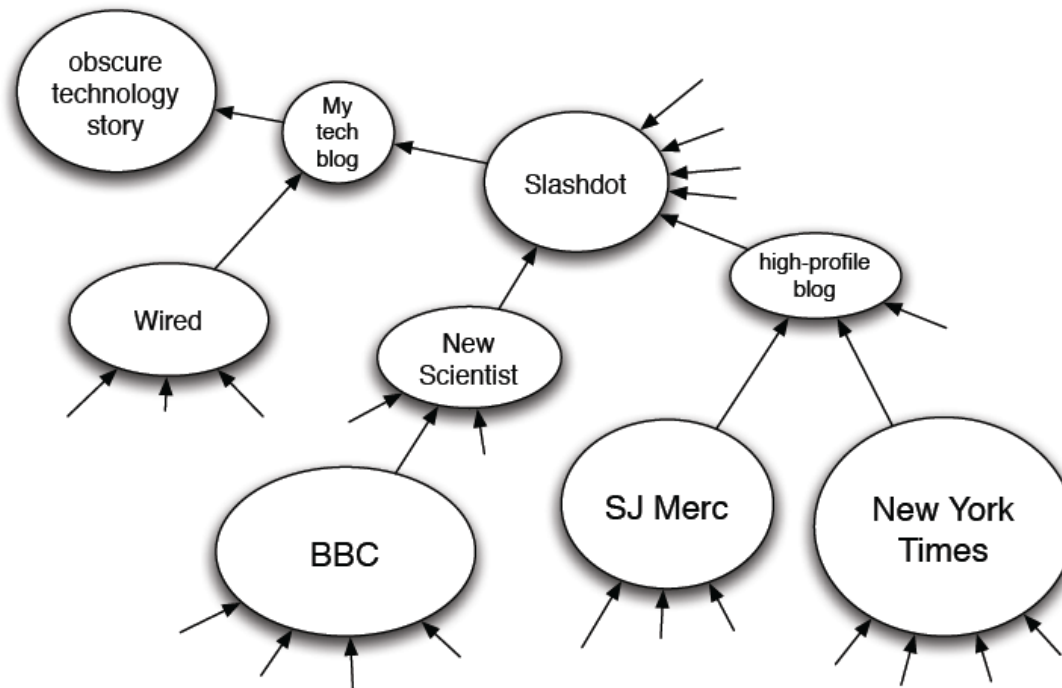
Conclusion and Connections

- **Diffusion of Topics**
 - How news cascade through on-line networks
 - Do we need new notions of rank?
- **Incentives and Diffusion**
 - Using diffusion in the design of on-line systems
 - Connections to game theory
- **When will one product overtake the other?**

Further Connections

- **Diffusion of topics** [Gruhl et al '04, Adar et al '04]:
 - News stories cascade through networks of bloggers
 - How do we track stories and rank news sources?
- **Recommendation incentive networks** [Leskovec-Adamic-Huberman '07]:
 - How much reward is needed to make the product “work-of-mouth” success?
- **Query incentive networks** [Kleinberg-Raghavan '05]:
 - Pose a request to neighbors; offer reward for answer
 - Neighbors can pass on request by offering (smaller) reward
 - How much reward is needed to produce an answer?

Topic Diffusion



- News and discussion spreads via diffusion:
 - Political cascades are different than technological cascades
- Suggests new ranking measures for blogs

Reflections

Starting to see some basic social network processes

- Diffusion is a model that captures many different processes:
 - In the on-line world: communities, topics, popularity, commerce
- Only recently have basic properties been observed on a large scale:
 - Confirms some social science intuitions; calls others into question
 - Interplay between theoretical consequences of diffusion properties and empirical studies
- A number of novel opportunities:
 - Predictive modeling of the spread of new ideas and behaviors
 - Opportunity to design systems that make use of diffusion process

References

- D. Kempe, J. Kleinberg, E. Tardos. Maximizing the Spread of Influence through a Social Network. ACM KDD, 2003.
- Jure Leskovec, Lada Adamic, Bernardo Huberman. The Dynamics of Viral Marketing. ACM TWEB 2007.
- Jure Leskovec, Mary McGlohon, Christos Faloutsos, Natalie Glance, Matthew Hurst. Cascading Behavior in Large Blog Graphs. SIAM Data Mining 2007.
- Jure Leskovec, Ajit Singh, Jon Kleinberg. Patterns of Influence in a Recommendation Network. PAKDD 2006.
- Jure Leskovec, Andreas Krause, Carlos Guestrin, Christos Faloutsos, Jeanne VanBriesen, Natalie Glance. Cost-effective Outbreak Detection in Networks. ACM KDD, 2007.

Acknowledgement

Some slides and drawings borrowed from Jon Kleinberg

Coming up next...

Case studies

- Microsoft Instant Messenger communication network
 - How does the whole world communicate?
- How to find fraudsters on eBay?
- Graph projections
 - How do we predict the quality of search results without looking at the content?