Lecture 13: Infect (3/4)

How do epidemic and gossip reach people? (i.e., how computer viruses spread?)

COMS 4995-1: Introduction to Social Networks
Part II: Cascade
Tuesday, October 25th
Epidemic model #2: \( S \rightarrow I \rightarrow R \)

- **Thm:** Assuming \( \beta \rho < 1 \), \( \mathbb{E}[ |Y(\infty)| ] \leq C \sqrt{N} / (1 - \beta \rho) \)
  - \( \rho(G) \): largest eigenvalue of G’s adjacency matrix
  - \( C = \sqrt{\#\{\text{initial infected population}\}} \)
- If \( \beta \rho < 1 \) and \( C = o(\sqrt{N}) \), negligible fraction removed

- **Examples:**
  - G is d-regular: \( \rho(G) = d \)
  - Can be applied to bound unif. random graphs
Outline

* Continuous epidemics, “logistic model”
* Discrete epidemics, “graph”
  – Adjacency matrix
  – SI, SIR model
  / – SIS

* Epidemic algorithms
Nodes follow neighbor contamination / recovery
- Node $u \in V$ infectious ($X_u = 1$) or susceptible ($X_u = 0$)
- Node $u$ becomes infected with rate $\beta \cdot \sum_{v \in N(u)} X_v$
- Node $u$ recovers with rate $\gamma = 1$

In a finite graph, all nodes eventually recover
- Because ($X_u = 0 \ \forall \ u \in V$) is the only absorbing state
- Different on infinite graphs (e.g. lattices, trees)
Can we recover fast from an epidemic?

Thm: \( P[X(t) \neq (0, \ldots, 0)] \leq C \sqrt{N} \exp(t \cdot (\beta \rho - 1)) \)
- \( \rho(G) \): largest eigenvalue of \( G \)'s adjacency matrix
- \( C = \sqrt{\#\{\text{initial infected population}\}} \)

Corollary: If we have \( \beta < \rho \)
- \( \mathbb{E}[\text{extinction time}] \leq (1+\ln(n)) / (1-\beta \rho) \)

Bottom line: goes to zero very fast if \( \beta \rho < 1 \)
- complete graph: \( \rho(G) = n-1 \)
- uniform random graph: \( \rho(G) \approx (n-1)p \) (if \( np = \omega(\log n) \))

The effect of network topology on the spread of epidemics,
Proof:

* Step 1: Introduce a random walk process

Intuitively we have $P[X(t) \neq 0] \leq P[Z(t) \neq 0]$.

- This statement can be made precise by coupling.
We have \( \Pr[Z(t) \neq 0] \leq \sum_v \Pr[Z_v(t) \neq 0] \leq \sum_v \mathbb{E}[Z_v(t)] \)

- Using Markov Inequality

We can rewrite evolution of \( Z(t) \)

\[
\frac{d}{dt} \mathbb{E}[Z_v(t)] = \beta \sum_{v' \in \mathcal{N}(v)} \mathbb{E}[Z_{v'}(t)] - \mathbb{E}[Z_v(t)]
\]

So that \( \Pr[Z(t) \neq 0] \leq \|e_1\| \|\exp(t \cdot (\beta A - I)) \cdot X(0)\| \)

\[
\frac{d}{dt} = \frac{d}{d\tau} (\beta A - I) \iff \frac{d}{d\tau} = \exp((\beta A - I)\tau) \cdot Z(0) \quad \frac{dZ}{d\tau} = c dt \]
Finally, we can apply the same bounding technique

\[ \frac{dZ}{dt} = \frac{1}{2} (BA - Jd) \Rightarrow \dot{Z}(t) = \exp(t(BA - Jd)) \cdot X(0) \]

\[ \mathbb{P}[X(t) \neq 0] \leq \sum \mathbb{E}[Z_{\nu}(k)] = \langle e_1, \dot{Z}(t) \rangle = \frac{1}{\sqrt{N}} \exp(t(\beta A - Jd)) \cdot ||X(0)|| \]

\[ A : \varphi(A) = \{ \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N \} \]

\[ \lambda \in \text{Spec}(C(\beta A - Jd)) = \{ 1 \} \]

\[ \exp(t) \cdot \frac{1}{\sqrt{N}} \exp(t(\beta A - Jd)) \cdot e^{(\beta \lambda_2 - 2)} \]

\[ \exp \left( t \cdot (B_0 - J) \right) \cdot \text{exp} \left( \frac{a^T}{a} \cdot \frac{1}{t} \cdot x_1 \cdot e^{\frac{t}{2}} \right) \]
Discrete epidemics: summary

<table>
<thead>
<tr>
<th>Type</th>
<th>Outcomes</th>
</tr>
</thead>
<tbody>
<tr>
<td>S→I</td>
<td>Everyone infected</td>
</tr>
<tr>
<td>S↔I</td>
<td>No infectious nodes</td>
</tr>
<tr>
<td>S→I→R</td>
<td>No infectious node</td>
</tr>
</tbody>
</table>

Follow processes of infection
- Initial conditions: small set infected nodes
- Speed or span depend on graph topology (e.g. spectral analysis)
Outline

* Continuous epidemics, “logistic model”
* Discrete epidemics, “graph”
* Epidemic algorithms
Replicated database maintenance
- Different versions, many locations
- How to handle communication? failures?

1987 “Epidemic alg., rumor spreading, gossip”
- Do not maintain fixed communication topology
- Contact a node unif., spread if one node has a copy

How many rounds $S_n$ before rumor spreads to all
- $S_n = (1+1/\ln(2)) \log(n) + O(1)$ in probability

Epidemic algorithms for replicated database maintenance,
A Demers et. al, ACM PODC. (1987)
A binary tree:
- Also takes time $O(\log(n))$, using $O(n)$ messages
- Seems optimal in both ways, but prone to failure

Gossip:
- Time $O(\log(n))$ (optimal) and $O(n \log n)$ messages
- In fact, unif. gossip requires at least $\omega(n)$ messages, and $\Omega(n \log \log(n))$ if no addresses are kept (the latter can be attained)

Randomized rumor spreading,
R Karp and C Schindelhauer and S Shenker and B Vocking, FOCS. (2000)