Finding and Hiding Message Sources in Networks:

Epidemics, Social Media, Cryptocurrencies

Giulia Fanti and Pramod Viswanath
Broadcasting Information: Then
Broadcasting Information: Now

Donald J. Trump
@realDonaldTrump

Sorry losers and haters, but my I.Q. is one of the highest -and you all know it! Please don't feel so stupid or insecure, it's not your fault

8:37 PM - 8 May 2013
Broadcast communication is easier, cheaper, and more democratic than ever before.
Distributed broadcasting

Epidemics

Social Networks

Cryptocurrencies
Epidemics

Gomes et al. 2014, PLOS
Social Networks
Cryptocurrencies
Broadcasting can impact the robustness, utility, and security of a network.

... but distributed network management poses new challenges!
Relevant Questions

- Who started a broadcast?
- What is the network structure?
- What are the spreading dynamics?
- When do messages go extinct?
- Who is allowed to broadcast?
Attribution is central to communication

“We'll know our disinformation program is complete when everything the American public believes is false.”
- William Casey, CIA Director
(from first staff meeting in 1981)
This talk

• **Part I**: Systems and how to model them (1 hr)
  • Bitcoin primer (30 min)
  • Network models
  • Propagation models
  • Observation models

• **Part II**: Source finding (1 hr)
  • Algorithms for source detection
  • Analysis of these algorithms
  • Open problems

• **Part III**: Source hiding (1 hr)
  • Early results: crypto community
  • Statistical approaches
  • Open problems
Cryptocurrencies Primer
The Origin of Bitcoin

Narayanan et al., *Bitcoin and Cryptocurrency Technologies*, 2016
Financial systems

**Cash**
- Offline transactions
- Anonymous
- Requires initial seed cash

**Credit**
- Exchanges can be digital
- Parties take on risk
Bitcoin Objectives

• **Egalitarianism** → no central trusted party

• **Transparency** → transactions can be verified by all nodes

• **Privacy** → users need not reveal their identity to the currency
## Bitcoin objectives

<table>
<thead>
<tr>
<th></th>
<th>Credit</th>
<th>Cash</th>
</tr>
</thead>
<tbody>
<tr>
<td>Egalitarianism</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Transparency</td>
<td>x</td>
<td>✓</td>
</tr>
<tr>
<td>Privacy</td>
<td>x</td>
<td>✓</td>
</tr>
</tbody>
</table>
Why this problem is hard

Money = string of bits
No central controller

What prevents forgeries?
What prevents double-spending?
Who creates money?

Digital signatures
Global ledger
Users
Append-only ledgers

Haber and Stornetta, 1991

Image from Narayanan et al, 2016
Hierarchical structure

Image from Narayanan et al, 2016
Basic network operation

Blockchain ...

Alice $IP_A$

Bob $IP_B$
Basic network operation

**tx2** \((k_{tx2})\)

Send 1 BTC from \(k_{tx1}\) to \(k_B\).
Signed: \(k_A\)
Adding to the Blockchain

Alice

Bob

tx2
Adding to the Blockchain

What’s wrong with this?
Basic network operation

Alice

\(\text{tx1} \rightarrow \text{Dev}\)

\(\text{tx2'}\)

\(\text{tx1} \rightarrow \text{Bob}\)
Adding to the Blockchain
Distributed Consensus in Bitcoin

**Goal:**
Pick 1 node uniformly at random

- No fixed notion of identity
- Robust to Sybils
Proof-of-Work

Puzzle
Find x: \( H(x) = f(tx, \text{blockchain}) \)
Mining

Alice
How are conflicts managed?
How are conflicts managed?
How are conflicts managed?
Bitcoin Consensus Protocol: Summary

• New transactions are broadcast

• Each node collects transactions into *blocks*

• One random node gets to broadcast its block / round

• Other nodes accept the block iff valid puzzle solution

• Miners “accept” blocks by referencing them in the next block
Probability of transaction reversal

Adversary
Finds blocks with probability $q$

Rest of the network
Finds blocks with probability $p$

$p > q$

Probability of transaction reversal

\[ p = \text{Probability an honest node finds next block} \]

\[ q = \text{Probability attacker finds next block} \]

\[ q_z = \text{Probability attacker overtakes main blockchain starting from } -z \text{ differential} \]

\[ q_z = \begin{cases} 1, & \text{if } p \leq q \\ \left(\frac{q}{p}\right)^{-z}, & \text{if } p > q \end{cases} \]

This does not hold by assumption
# Properties of Proofs of Work

<table>
<thead>
<tr>
<th>Cost</th>
<th>Reward</th>
</tr>
</thead>
<tbody>
<tr>
<td>Measured in:</td>
<td>Computation</td>
</tr>
<tr>
<td></td>
<td>Bitcoins (new-block reward, transaction fees)</td>
</tr>
<tr>
<td>Scales according to:</td>
<td>Network’s mining power (1 block per 10 minutes)</td>
</tr>
<tr>
<td></td>
<td>Geometric scaling</td>
</tr>
</tbody>
</table>
Bitcoin's Controlled Supply is a function of the Block Height and the Block Reward.

The block reward started at 50BTC. The block reward is halved every 210,000 blocks.

Theoretically this would lead to a maximum number of Bitcoins that tends toward 21,000,000.

Due to a limitation in the present data structure of the blockchain, the maximum number of Bitcoins is actually 20,999,999.9769.

This maximum will be reached when block 6,929,999 has been mined.
What purposes does mining serve?

- Distributed consensus protocol
- Limit rate of production
Repeat after me: if you don't need concurrent access to a decentralized, mutable, singleton, you don't need a #blockchain.

— ArthurB (@ArthurB) December 17, 2014
Why should the IT community care?

1. Network is central
2. Distributed storage
3. Game theory

This talk
Figure 1: Privacy-Enhancing Technologies for Bitcoin. The X-axis is the date of invention and the Y-axis is an informal measure that combines the sophistication of the technique and the strength of the privacy guarantee. See Appendix 1 for references.

Narayanan and Moser, 2017
Models

Broadcasting over Networks
System Modeling

Network Models

Propagation Models

Observation/Adversarial Models
Network Models

- Regular trees
- Irregular trees
- Random regular graphs
- General graphs

Theoretical Results

Empirical Results
Propagation Models

**Susceptible-Infected (SI)**

S \rightarrow I

**Susceptible-Infected-Susceptible (SIS)**

\[ S \rightarrow I \rightarrow S \]

**Susceptible-Infected-Recovered (SIR)**

\[ S \rightarrow I \rightarrow R \]
# Propagation Models

<table>
<thead>
<tr>
<th></th>
<th>Susceptible-Infected (SI)</th>
<th>Susceptible-Infected-Susceptible (SIS)</th>
<th>Susceptible-Infected-Recovered (SIR)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Continuous-time</strong></td>
<td><img src="green" alt="Epidemics" />, <img src="orange" alt="Social media" />, <img src="purple" alt="Cryptocurrencies" /></td>
<td><img src="green" alt="Epidemics" />, <img src="orange" alt="Social media" /></td>
<td><img src="green" alt="Epidemics" /></td>
</tr>
<tr>
<td><strong>Discrete-Time</strong></td>
<td><img src="purple" alt="Cryptocurrencies" /></td>
<td><img src="orange" alt="Social media" /></td>
<td><img src="green" alt="Epidemics" /></td>
</tr>
</tbody>
</table>
SI Diffusion (continuous-time)
SI Diffusion (discrete-time)
SI Gossip (discrete-time)
SI Gossip (discrete-time)
Propagation Models: Key attributes

• Fully-distributed protocols

• Infection model can vary (SI, SIR, SIS)

• **Continuous**- vs. discrete-time systems

• Gossip vs. **diffusion**
Snapshot Observer

Epidemics

Cryptocurrencies
Eavesdropping Observer

Luxembourg Researchers Find a Way to Unmask Bitcoin Users

P. H. Madore on 30/11/2014
Eavesdropping Observer

- Epidemics
- Cryptocurrencies

Supernode

\[ \theta \] connections per node
Spy-based Observer

The Facebook Squad: How Israel Police Tracks Activists on Social Media

It follows their Facebook pages, uses fake profiles to 'befriend' them and presents screenshots of posts in court – this is how Israel Police is adding social activists to its virtual surveillance list. 'They know what I write and do,' Ethiopian protest leader says.

Yaniv Kubovich | Feb 06, 2016 9:46 AM
Sampled Observers (Spies)

\[ G \]

\[ t = 06:10:34 \]

\[ t = 06:12:18 \]

- Epidemics
- Social media
- Cryptocurrencies
Observation Models: Key Attributes

• Fraction of nodes that can be observed (all nodes, subset)

• Delay of observation at those nodes (instantaneous / random)

• Nodes’ adherence to protocol (honest-but-curious / malicious)
Summary: Modeling Epidemics

- Network models
  - **Trees**
  - General graphs (social networks, random graphs)

- Spreading models
  - **Diffusion**

- Observation/adversarial models
  - **Snapshot**
  - Spy-based, eavesdropper
Finding the Source

Part II
What you will learn in this hour

• Source detection algorithms
  • Rumor centrality
  • Other heuristics

• Introduction to Pólya urns
  • Definition
  • Convergence results
  • Generalizations

• Using Pólya urn processes to analyze the probability of source detection in diffusion processes
Source Detection Algorithms

Centrality measures
Rumors in networks
Rumors in networks

- a random node is the source of the rumor
Diffusion spreading

- Node 2 spreads the rumor to its neighbors iid along its edges
Rumors in networks
Diffusion Spreading

- Both nodes 1 and 2 spread the message along their edges
Diffusion Spreading

- Node 3 receives the message, say.
Diffusion Spreading
Diffusion Spreading
Snapshot observation

- Get to observe set of nodes with the message
- No timestamps
Source of Rumor

- Use knowledge of underlying graph
- Knowledge of set of nodes with the message
Centrality

• Source is in the center
Rumor centrality

- Specific metric of centrality

Rumor centrality

- Hypothesis: node 1 is the source
Rumor centrality

• Identify a possible spreading pattern
Rumor centrality

• Enumerate all possible spreading patterns
Rumor centrality

1 → 2 → 3 → 4 → 8
1 → 2 → 3 → 8 → 4
1 → 2 → 4 → 3 → 8
Rumor centrality

$R(1) = 3$

• Score = number of possible spreading patterns
Rumor centrality

- Similar score for node 2
Rumor centrality

\[ R(2) = 12 \]

2 → 1 → 4 → 3 → 8
2 → 1 → 3 → 4 → 8
2 → 1 → 3 → 8 → 4
2 → 4 → 1 → 3 → 8
2 → 4 → 3 → 1 → 8
2 → 4 → 3 → 8 → 1
2 → 3 → 1 → 4 → 8
2 → 3 → 1 → 8 → 4
2 → 3 → 8 → 1 → 4
2 → 3 → 4 → 1 → 8
2 → 3 → 4 → 8 → 1
Rumor centrality
Rumor centrality

$R(4) = 3$
Rumor centrality

• Node 2 has the highest centrality score
Rumor centrality

Same as picking node with: smallest sum of distances to all nodes
Jordan centrality

- Maximum distance from a node to another
Jordan centrality
Jordan centrality

$J(1) = 3$

- Node 1's eccentricity is 3
Jordan centrality

\[ J(2) = 2 \]
Jordan centrality

- Both nodes 2 and 3 are equally central
Counting Efficiently

- Naive counting is very inefficient
Naïve implementation of rumor centrality

• Some orderings are valid, others not

2 → 1 → 4 → 3 → 8  2 → 4 → 1 → 8 → 3
2 → 1 → 3 → 4 → 8  2 → 1 → 4 → 8 → 3
2 → 1 → 3 → 8 → 4  2 → 4 → 8 → 1 → 3
2 → 4 → 1 → 3 → 8  2 → 4 → 8 → 3 → 1
2 → 4 → 3 → 1 → 8  2 → 1 → 8 → 4 → 3
2 → 4 → 3 → 8 → 1  2 → 1 → 8 → 3 → 4
2 → 3 → 1 → 4 → 8  2 → 8 → 4 → 1 → 3
2 → 3 → 1 → 8 → 4  2 → 8 → 1 → 3 → 4
2 → 3 → 8 → 1 → 4  2 → 8 → 1 → 4 → 3
2 → 3 → 8 → 4 → 1  2 → 8 → 3 → 1 → 4
2 → 3 → 4 → 1 → 8  2 → 8 → 3 → 4 → 1
2 → 3 → 4 → 8 → 1  2 → 8 → 4 → 3 → 1
Rumor centrality via message passing

- Reuse computations
Rumor centrality via message passing

- Start with a node (1, say) and form a rooted tree
Rumor centrality via message passing

- Tree rooted at node 2
Upward pass

- Messages pass upwards from leaves to the root
Upward pass

- Two types of messages

\[
\begin{align*}
t_{4 \rightarrow 2} &= 1 \\
p_{4 \rightarrow 2} &= 1 \\
t_{8 \rightarrow 3} &= 1 \\
p_{8 \rightarrow 3} &= 1
\end{align*}
\]
Upward pass

- Node 3 processes its message and sends it to its parent
Upward pass

• Node 2 can now process its message and send it
Upward pass

• Node 1 gets to calculate its rumor centrality score

\[ R(1) = \frac{N!}{Np_{2 \rightarrow 1}} = \frac{5!}{5 \times 8} = 3 \]

\[ p_{2 \rightarrow 1} = t_{2 \rightarrow 1}p_{3 \rightarrow 2}p_{4 \rightarrow 2} = 8 \]
\[ t_{2 \rightarrow 1} = t_{3 \rightarrow 2} + t_{4 \rightarrow 2} + 1 = 4 \]

\[ t_{4 \rightarrow 2} = 1 \]
\[ p_{4 \rightarrow 2} = 1 \]

\[ p_{3 \rightarrow 2} = t_{3 \rightarrow 2}p_{8 \rightarrow 3} = 2 \]
\[ t_{3 \rightarrow 2} = t_{8 \rightarrow 3} + 1 = 2 \]

\[ t_{8 \rightarrow 3} = 1 \]
\[ p_{8 \rightarrow 3} = 1 \]
Downward pass

- Messages pass downwards from root

\[
\begin{align*}
R(1) &= 3 \\
2 &
\begin{aligned}
4 &
\begin{aligned}
3 &
\begin{aligned}
8 &
\end{aligned}
\end{aligned}
\end{aligned}
\end{aligned}
\end{align*}
\]

- $t_{4 \rightarrow 2} = 1$
- $t_{3 \rightarrow 2} = 2$
- $t_{8 \rightarrow 3} = 1$
- $t_{2 \rightarrow 1} = 4$
Downward pass

- Pass the rumor centrality score downwards

```
   1
  / \  \
 /   \ /
2     2
  \   / \
   \ /  \
   \  /
  4   3

R(1) = 3
\( t_{2\rightarrow 1} = 4 \)
\( t_{4\rightarrow 2} = 1 \)
\( t_{3\rightarrow 2} = 2 \)
\( t_{8\rightarrow 3} = 1 \)
```
Downward pass

\[ R(2) = R(1) \frac{t_{2 \rightarrow 1}}{N - t_{2 \rightarrow 1}} = 3 \frac{4}{5 - 4} = 12 \]

- Node 2 can compute its rumor centrality score

- \[ t_{4 \rightarrow 2} = 1 \]
- \[ t_{3 \rightarrow 2} = 2 \]
- \[ t_{8 \rightarrow 3} = 1 \]
Downward pass

1
   R(1) = 3

2
   R(2) = 12
   t_{4 \rightarrow 2} = 1
   R(2) = 12
   t_{3 \rightarrow 2} = 2

3

4

8
   t_{8 \rightarrow 3} = 1
Downward pass

\[ R(4) = R(2) \frac{t_{4 \rightarrow 2}}{N - t_{4 \rightarrow 2}} = 12 \frac{1}{5 - 1} = 3 \]
Downward pass

\[ R(3) = R(2) \frac{t_{3 \rightarrow 2}}{N - t_{3 \rightarrow 2}} = 12 \frac{2}{5 - 2} = 8 \]

Diagram:

- Node 1: \( R(1) = 3 \)
- Node 2: \( R(2) = 12 \)
- Node 3: \( t_{3 \rightarrow 2} = 2 \)
- Node 4: \( R(4) = 3 \)
- Node 8: \( t_{8 \rightarrow 3} = 1 \)
Downward pass

\begin{center}
\includegraphics[width=0.5\textwidth]{diagram.png}
\end{center}
Downward pass

\[
R(8) = R(3) \frac{t_{8 \rightarrow 3}}{N - t_{8 \rightarrow 3}} = 8 \frac{1}{5 - 1} = 2
\]

![Diagram with nodes and edges labeled with values.](image-url)
Computational complexity

- 3N computations
Choice of root node

- Root node could have been 2
- Rumor centrality scores remain the same
Graphs with cycles?

- Heuristic: spreading occurs on a **breadth-first tree**
Regular tree

- Theorem: Rumor centrality = Maximum Likelihood
- Positive probability of detection, asymptotically

Shah and Zaman, Rumor Centrality: A Universal Source Detector, Sigmetrics 2012
Analyzing Diffusion Processes

Pólya Urns and More
Introduction to Pólya Urns

What is the fraction of red balls after \( n \) draws?

1) Analyze for 2 colors.
2) Generalize

Does the order of draws matter?

\[ P(r_n = k + 1) = \binom{n}{k} \beta(k + 1, n + 1 - k) \]

\[ \beta(x, y) = \int_0^1 m^{x-1} (1 - m)^{y-1} \, dm \]
Does the fraction of red balls converge?

- $r_n$: Number of red balls
- $R_n$: Fraction of red balls
- $R_n = \frac{r_n}{n+2}$

**Approach**

1) $R_n$ is a martingale.

2) That martingale converges a.s.
1) $R_n$ is a martingale.

- $r_n$: Number of red balls
- $R_n$: Fraction of red balls $R_n = \frac{r_n}{n + 2}$

$$E[R_n \mid R_{n-1}, \ldots, R_1] =$$

$$= \frac{R_{n-1} + r_{n-1}}{n + 2} = \frac{r_{n-1}(n + 2)}{(n + 1)(n + 2)} = R_{n-1}$$
2) This martingale converges a.s.

Martingale Convergence Theorem

\[ R_n \in (0,1) \]

\[ \rightarrow R(\omega) = \lim_{n \to \infty} R_n(\omega) \]
What is the limiting distribution?

Let’s look at the moment-generating function

\[ M_{R_n}(t) = E[\exp(tR_n)] \]

\[ = \sum_{k=0}^{n} \exp(t \frac{k + 1}{n + 2}) P(R_n = \frac{k + 1}{n + 2}) \]

\[ = \sum_{k=0}^{n} \exp \left( t \frac{k + 1}{n + 2} \right) \int_{0}^{1} \binom{n}{k} m^k (1 - m)^{n-k} \, dm \]

\[ \rightarrow \int_{0}^{1} e^{tm} \, dm = \begin{cases} 
\frac{e^t - 1}{t}, & x \neq 0 \\
1, & x = 0
\end{cases} \]
Generalization 1: Number of replacements

\[ R_0 = 3 \]
\[ B_0 = 2 \]

\[ \gamma = \text{new balls added of same color} \]

\[ R \sim \text{Beta}(\frac{R_0}{\gamma}, \frac{B_0}{\gamma}) \]

Depends on initial conditions!
Generalization 2: Number of classes

\[ \alpha = [1 \ 1 \ 2] \quad \text{Initial values} \]
\[ \gamma = 2 \quad \# \text{added balls of same color} \]

\[ R \sim \text{Dirichlet} - \text{Multinomial}(\alpha, n) \]
How can we analyze diffusion?

A nice property

\[ X_{\text{blue}} = 1 \]
\[ X_{\text{orange}} = 2 \]
\[ X_{\text{red}} = 1 \]

\( \nu \) is a rumor center iff

\[ X_i(T) \leq \frac{X_{\text{total}}}{2} \]

\( i \in \{\text{red, orange, blue}\} \)

Number of "infected" nodes

Example:

\[ X_{\text{orange}} = 2 \]

\[ \frac{n}{2} = \frac{5}{2} \]
What does this mean for our urn?

$B_n$: Fraction of blue
$R_n$: Fraction of red
$O_n$: Fraction of other

$\nu$ is a rumor center iff

$$B_n, R_n, O_n \leq \frac{1}{2}$$

Let’s use the convergence results from before.
Let’s consider a slightly different urn.

Want $R_n$ as $n \to \infty$

\[ R \sim \text{Beta}\left(\frac{1}{d-2}, \frac{d-1}{d-2}\right) = \text{Beta}(1, 2) \]
Putting it all together

\[ R \sim \text{Beta}\left(\frac{1}{d-2}, \frac{d-1}{d-2}\right) \]

Want \( R \leq \frac{1}{2} \)

\[ I_1(a, b) \triangleq P(X \in [0, \frac{1}{2}]) \text{ where } X \sim \text{Beta}(a, b) \]

\[
\lim_{t \to \infty} P(\text{detection}) = 1 - d(1 - I_{\frac{1}{2}}\left(\frac{1}{d-2}, \frac{d-1}{d-2}\right))
\]

Example: \( (d = 3) \rightarrow \lim_{t \to \infty} P(\text{detection}) = 0.25 \)

Rumor centrality: A Universal Source Detector, Shah and Zaman, 2012
What about other problems?

\[ \theta = 2 \text{ connections per node} \]
Let’s model this as an urn
Generalized Polya Urns

Replacement Matrix

\[ A = \begin{bmatrix} d - 2 & 1 \\ 0 & -1 \end{bmatrix} \]

Solid  Striped

Example

\[ A = \begin{bmatrix} 1 & 1 \\ 0 & -1 \end{bmatrix} \]
Convergence properties

Conditions
1) \( A_{ij} \geq 0 \) for \( i \neq j \) and \( A_{ii} \geq -1 \)

2) Largest real eigenvalue of \( A (\lambda_1) \) is
   1) positive
   2) simple

3) Start with \( \geq 1 \) ball of a dominating type

Example
1) \( A_{ij} \geq 0 \) and \( A_{ii} \geq -1 \)

2) \( \lambda(A) = \{1,-1\} \)

3) Solids are a dominating type

\[
\begin{pmatrix}
R_n \\
1 - R_n
\end{pmatrix}
\xrightarrow{a.s.} \lambda_1 \begin{pmatrix} v_1 \end{pmatrix}
\]

Fraction of solid balls
Fraction of striped balls
First eigenvector
First (right) eigenvector

Athreya and Ney 1972, Jansen 2003
Comparing the two results

**Classic Pólya Urns**
- Transition matrix
  - Nonsingular
  - Not positive regular

\[ A = \begin{bmatrix}
  d - 2 & \cdots & 0 \\
  \vdots & \ddots & \vdots \\
  0 & \cdots & d - 2
\end{bmatrix} \]

- Converges to a **random variable** (Beta distribution)

**Generalized Pólya Urns**
- Transition matrix
  - Nonsingular
  - Positive regular

\[ A = \begin{bmatrix}
  d - 2 & 1 \\
  0 & -1
\end{bmatrix} \]

- Converges to a **constant**
Back to the eavesdropper

$x_t^i(v) = \# \text{ blue balls in } i\text{th subtree of } v \text{ at time } t$

1. If $\frac{x_t^i(v)}{\sum x_t^i(v)} < \frac{1}{2}$, $\forall i$, then $v$ is a reporting source.

2. Estimate $\hat{v}$ drawn uniformly from the set of reporting sources.
Back to the eavesdropper

Proof Sketch

1) $v^*$ is a rumor center with known probability.

2) Given that $v^*$ is a rumor center, 
   \[ \lim_{t \to \infty} P(v^* \text{ is a reporting center}) = 1 \]

3) There is at most 1 reporting center.

Anonymity Properties of the Bitcoin P2P Network, 2017
Summary of Approach

• Extract a representation of the problem that can be modeled as a Pólya Urn

• Use known convergence results (Athreya and Ney 1972, Jansen 2003)
Spy Adversary

- Spy nodes observe time stamps
Centrality methods

• First spy estimator
  • source = node reporting earliest to spies
  • very easy to implement
  • no knowledge of underlying graph
Centrality methods

• Earliest infection time estimator [Zhu, Chen, Ying, 2014]
  
  • estimate infection times of other nodes

• eccentricity score =

\[
\min_{T \in \mathcal{P}_v} \min_{(u,v) \in T} \sum_{u,v,\mu} (t_u - t_v - \mu)^2
\]

• pick node with smallest eccentricity

• related estimator [Pinto, Thiran, Vetterli, 2012]
Thoughts on how to handle spies

- Use the same counting-based estimator
- Use randomized Polya urns
Open Problems

Moving Forward
Other related questions

- Number of sources
- Detecting more than one source
- Combination of adversaries: snapshot+eavesdropper+spy
- Inferring the underlying network
Inferring diffusion networks
Inferring diffusion networks
Inferring diffusion networks
Inferring diffusion networks

t=0
Inferring diffusion networks
Inferring diffusion networks
Inferring diffusion networks

$t=0$

only time-stamps are observed
Inferring diffusion networks
Inferring diffusion networks
Inferring diffusion networks
Inferring diffusion networks

Goal:
Estimate underlying graph topology
Models

- independent cascades model [Kempe, Kleinberg, Tardos ’03]
  - discrete-time
  - susceptible $\rightarrow$ active for one time-slot $\rightarrow$ inactive
  - node i infects j with probability $p_{ij}$ if i is active
Algorithms

- estimate $p_{i,j}$ for all pairs $(i,j)$:
  - log likelihood decouples, each term convex
- threshold to output graph
- sample complexity $O(d^2 \log n)$ for degree bound $d$

[Netrapalli, Sanghavi ’12],
[Daneshmand, Gomez-Rodriguez, Song, Scholkopf ’14]
Algorithms

- submodularity

- greedy algorithm; add one edge at a time to the graph estimate

[Gomez-Rodriguez, Leskovec, Krause ’12]
Hiding the Source

Part III
What you will learn in this hour

• Classical approach from the crypto community
  • Dining cryptographer networks

• Statistical approaches
  • Static graph is given
  • Dynamic graph can be chosen

• Open problems
General-Purpose Hiding
Dining Cryptographer Networks
Dining Cryptographer Networks

Chaum, *The Dining Cryptographers Problem*, 1988
What are some problems?

• High communication costs

• Cannot handle collisions

• Fragile to misbehaving nodes

Golle and Juels, *Dining Cryptographers Revisited*, 2004
Sirer et al., *Eluding Carnivores: File Sharing with Strong Anonymity*, 2004
Franck, *New Directions for Dining Cryptographers*, 2008
Corrigan-Gibbs et al., *Dissent: Accountable Group Anonymity*, 2013

...
Worst-case solutions can be **too heavy** to be practical.
Hiding on a Static Network

Applications in Social Networks
Information flow in social networks

Diffusion has statistical symmetry
Breaking symmetry: Adaptive diffusion

Provides provable anonymity guarantees

[Spy vs. Spy: Rumor Source Obfuscation, ACM Sigmetrics 2015]
$d$-regular trees: adaptive diffusion

Initially, the author is also the “virtual source”
$d$-regular trees: adaptive diffusion

Break directional symmetry
$d$-regular trees: adaptive diffusion

chosen neighbor = new virtual source
$d$-regular trees: *adaptive diffusion*
$d$-regular trees: adaptive diffusion

Break temporal symmetry

keep the virtual source token

pass the virtual source token
keep the virtual source token
pass the virtual source token

new virtual source
pass the virtual source token
Results

<table>
<thead>
<tr>
<th></th>
<th>d-Regular trees</th>
<th>Irregular trees</th>
<th>Facebook graph</th>
</tr>
</thead>
<tbody>
<tr>
<td>Snapshot</td>
<td>[1]</td>
<td>[2]</td>
<td>[1]</td>
</tr>
</tbody>
</table>

[1] Spy vs. Spy: Rumor Source Obfuscation, Sigmetrics 2015
When to keep the virtual source token?

Virtual source token is kept with probability $\alpha = (d - 1)^{-h}$
Maximum likelihood detection

**Theorem:** Probability of detection \( \frac{1}{N-1} \)
hop distance from source

\[ h = 1 \]

\[ h = 2 \]

Likelihood = \( \mathcal{L} \) \( \alpha \)

Likelihood = \( \frac{1}{d} \cdot \frac{1 - \alpha}{d - 1} \)

Want these to be equal:

\[ \alpha = \frac{1}{d} \]
Irregular trees

\[ d_v = \begin{cases} 
3 & \text{w.p.} \ 0.7 \\
5 & \text{w.p.} \ 0.3 
\end{cases} \]

\[ d_{\text{max}} = 5 \]

\[ d_{\text{min}} = 3 \]
How do we analyze this?

\[ d_v = \begin{cases} d_{\min} & \text{w.p. } p_{\min} \\ d_{\max} & \text{w.p. } p_{\max} \end{cases} \]

\[ \hat{\theta}_{ML} = \arg \max_{v \in \text{leaves}} \frac{1}{d_v} \prod_{w \in P(v, v_T)} \frac{1}{d_w - 1} \]

Path from \( v \) to virtual source

Degree of node \( w \)

\[ P(\text{detection} \mid \text{snapshot}) = \min_{v \in \text{leaves}} \frac{1}{d_v \prod_{w \in P(v, v_T)} (d_w - 1)} \]
Main result (special case)

\[
\Lambda_{GT} \triangleq \min_{v \in \text{leaves}} d_v \prod_{w \in P(v,v_T)} (d_w - 1)
\]

If \( p_{min}(d_{min} - 1) > 1 \)

\[
P \left( \left| \frac{\log(\Lambda_{GT})}{T} - \log(d_{min} - 1) \right| > \delta \right) \leq e^{-c_1 T}
\]

**Theorem:** Probability of detection \( \approx \frac{1}{(d_{min} - 1)^T} \)
**Theorem:** Probability of detection \( \approx \frac{1}{(d_{\text{min}} - 1)^T} \)
Proof sketch for

\[
\min_{v \in \text{leaves}} d_v \prod_{w \in P(v, v_T)} (d_w - 1) \approx (d_{\text{min}} - 1)^T
\]

\[
d_v = \begin{cases} 
3 & \text{w.p. 0.7} \\
5 & \text{w.p. 0.3}
\end{cases}
\]

If \( p_{\text{min}}(d_{\text{min}} - 1) > 1 \) then the pruned process survives.
If $p_{\text{min}}(d_{\text{min}} - 1) > 1$:

$$\min_{v \in \text{leaves}} d_v \prod_{w \in P(v, v_T)} d_w - 1 \approx (d_{\text{min}} - 1)^T$$
Main result

\[ \Lambda_{G_T} \triangleq \min_{v \in \text{leaves}} d_v \prod_{w \in P(v,v_T)} (d_w-1) \]

In general,

\[ P \left( \left| \frac{\log(\Lambda_{G_T})}{T} - r^* \right| > \delta \right) \leq e^{-c_1 r} \]

\[ \beta_i = \frac{p_i(d_i - 1)}{\sum_j p_j(d_j - 1)} \]

\[ \mu = \log \sum_i p_i(d_i - 1) \]

\[ R_D = \{ r \in S \mid \mu \geq D_{KL}(r \parallel \beta) \} \]

\[ r^* = \min_{r \in R_D} \langle r, \log(d - 1) \rangle \]
Results

<table>
<thead>
<tr>
<th>Snapshot</th>
<th>$d$-Regular trees</th>
<th>Irregular trees</th>
<th>Facebook graph</th>
</tr>
</thead>
</table>

[1] Spy vs. Spy: Rumor Source Obfuscation, Sigmetrics 2015
Facebook graph

![Graph showing the relationship between Probability of Detection and Nodes Infected (N). The graph illustrates the diffusion and perfect hiding scenarios.]
Results

<table>
<thead>
<tr>
<th></th>
<th>$d$-Regular trees</th>
<th>Irregular trees</th>
<th>Facebook graph</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Spy-based</strong></td>
<td>[3]</td>
<td>[3]</td>
<td>[3]</td>
</tr>
</tbody>
</table>

[1] Spy vs. Spy: Rumor Source Obfuscation, Sigmetrics 2015
Spy-based adversary

With probability $p$

Adversary sees metadata at spy nodes
Facebook Graph

![Graph showing spy probability vs. probability of detection with different diffusion models: Adaptive diffusion, Diffusion q=0.1, Diffusion q=0.5.](image)

- **Lower bound, p**
- **Adaptive diffusion**
- **Diffusion, q=0.1**
- **Diffusion, q=0.5**

Lower bound on detection
THEOREM: Probability of detection = \( p + o(p) \)
Hiding on a Dynamic Network

Applications in Cryptocurrencies
Bitcoin Reminder

Transaction

$k_A$ sends $k_{coin}$ to $k_B$

Blockchain
sd93fjj2
pckrn29

... our transaction

Alice

$k_A$

$k_{coin}$

Bob

$k_B$
Botnet (spy-based) adversarial model

- Spies collude
- Observe all metadata
- Identities unknown
- Fraction $p$ of spies
- Honest-but-curious
Metric for Anonymity

Recall

\[ \frac{1}{n} \sum_v 1\{M(v's\ tx) = v\} \]

\( \sum \) represents the total number of transactions.\( \forall \)v, \( M(v's\ tx) \) is the mapping of transaction to user.\( \forall \)v, \( 1\{M(v's\ tx) = v\} \) is an indicator function that returns 1 if the transaction is mapped to user v, otherwise 0.

\[ \mathbb{E}[\text{Recall}] = \text{Probability of Detection} \]

Precision

\[ \frac{1}{n} \sum_v 1\{M(v's\ tx) = v\} \]

\( \sum \) represents the total number of transactions.\( \forall \)v, \( M(v's\ tx) \) is the mapping of transaction to user.\( \forall \)v, \( 1\{M(v's\ tx) = v\} \) is an indicator function that returns 1 if the transaction is mapped to user v, otherwise 0.

\( \# \) tx mapped to v represents the number of transactions mapped to user v.
Goal:

Design a distributed flooding protocol that minimizes the maximum precision and recall achievable by a computationally-unbounded adversary.
**Thm:** Maximum recall $\geq p$.

**Thm:** Maximum precision $\geq p^2$. 

Fraction of spies
What are we looking for?

Asymmetry

Mixing

1 2 3 4 spy
What can we control?

**Spreading Protocol**
- Diffusion

**Topology**
- Approximately regular

**Dynamicity**
- Dynamic
- Static

*Given a graph, how do we spread content?*

*What is the underlying graph topology?*

*How often does the graph change?*

Dandelion: Redesigning the Bitcoin Network for Anonymity, Sigmetrics 2017
Spreading Protocol: Dandelion

1) Anonymity Phase

2) Spreading Phase
Why Dandelion spreading?

**Theorem**: Dandelion spreading has an optimally low maximum recall of \( p + O \left( \frac{1}{n} \right) \).

- lower bound = \( p \)
- fraction of spies
- number of nodes
Graph Topology: Line

Anonymity graph

“Regular” graph

\texttt{tx1}

\texttt{tx2}
Dynamicity: High

Change the anonymity graph frequently.
DANDELION Network Policy

**Spreading Protocol**

- Given a graph, how do we spread content?

**Topology**

- What is the anonymity graph topology?

**Dynamicity**

- How often does the graph change?

- Dynamic

- Static
**Theorem**: DANDELION has a nearly-optimal maximum precision of \( \frac{2p^2}{1-p} \log \left( \frac{2}{p} \right) + O \left( \frac{1}{n} \right). \)

*For \( p < \frac{1}{3} \)
Performance: Achievable Region
How practical is this?
Dandelion spreading

1) Anonymity Phase

2) Spreading Phase
Anonymity graph construction
Dealing with stronger adversaries

- Learn the graph
- Misbehave during graph construction
- Misbehave during propagation

- 4-regular graphs
- Only send messages on outgoing edges
- Multiple nodes diffuse
Latency Overhead: Estimate

Information Propagation in the Bitcoin Network, Decker and Wattenhofer, 2013
Why not alternative solutions?

Connect through Tor

I2P Integration (e.g. Monero)
Open Problems

• Static graph
  • Modeling user preferences
  • Using cliques for better anonymity on general graphs

• Dynamic graph
  • Characterizing graph learning rate

• Both
  • Intersection attacks!
Conclusion

• Broadcasting information
  • common primitive
  • modern applications

• Performance metrics
  • latency, spreading rate, coverage, anonymity

• Engineering choices
  • underlying topology, spreading protocol

• **Finding** the source
  • Inferring the network topology

• **Hiding** the source