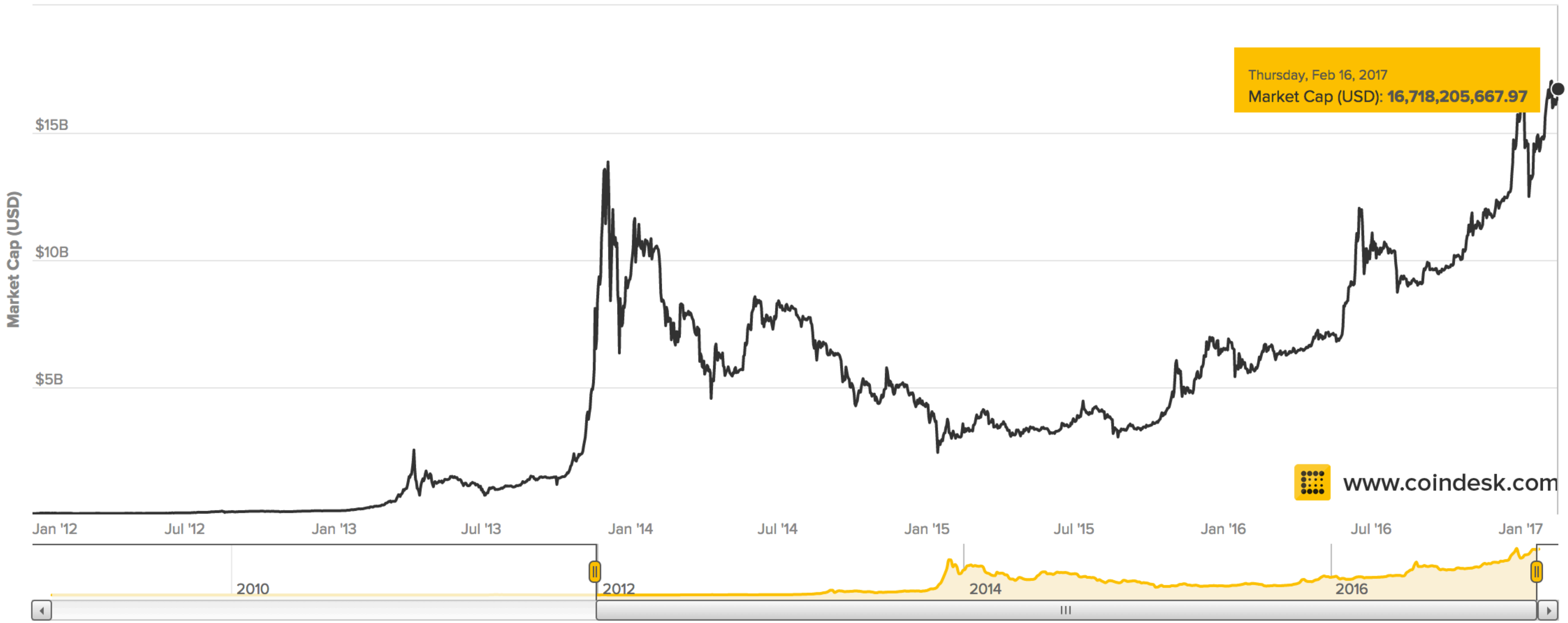
Anonymity in the Bitcoin Peer-to-Peer Network

Shaileshh Bojja Venkatakrishnan, Giulia Fanti,
Andrew Miller, Pramod Viswanath

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Bitcoin Market Cap over Time



Why do People Use Cryptocurrencies?

Currency Stability

Investment

Technical Properties/ Ideology

“Untraceable Bitcoin”

Teenagers using untraceable currency Bitcoin to buy dangerous drugs online

Fears have been raised as children as young as 14 are getting parcels of legal highs delivered to their home

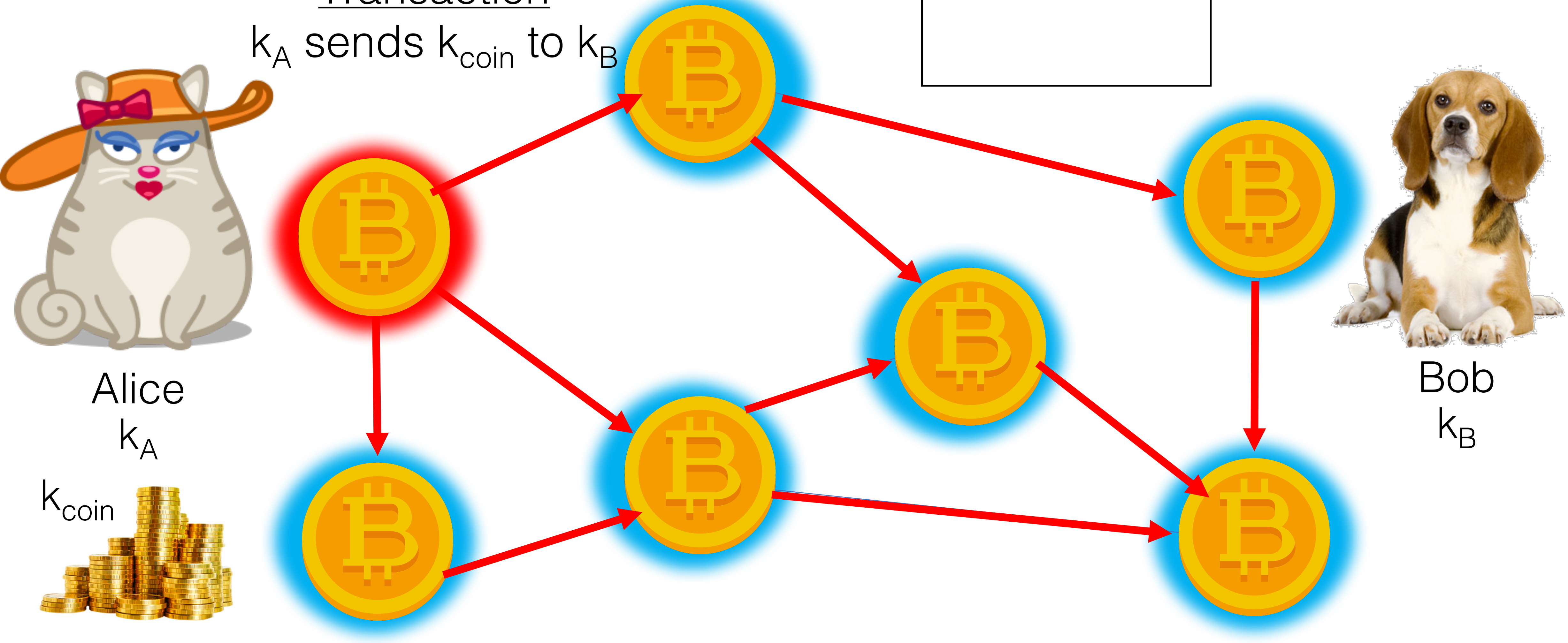
Mirror

This is false.

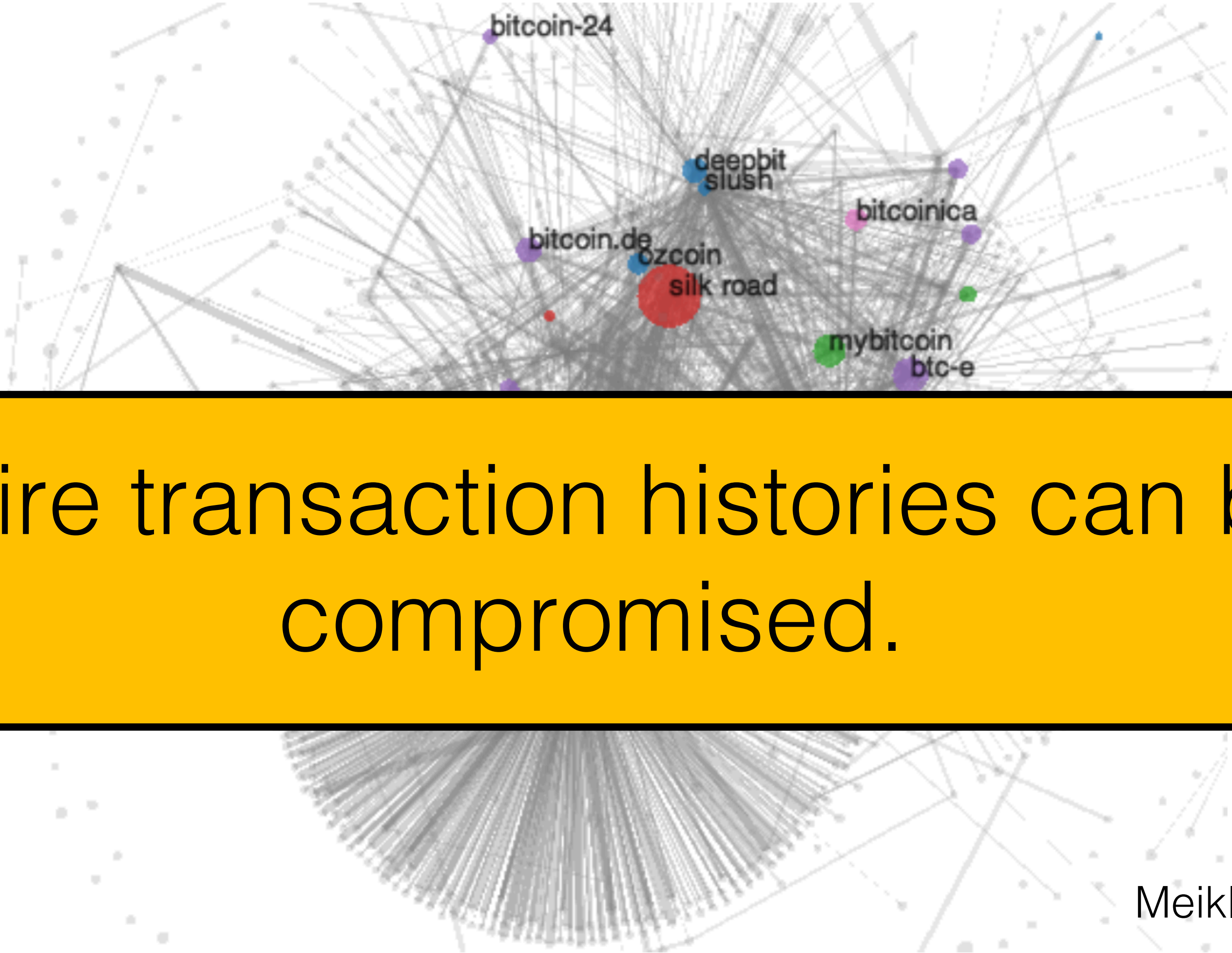
Bitcoin Primer

Blockchain
sd93fjj2
pckrn29
…
our transaction

Transaction
k_A sends k_{coin} to k_B



How can users be deanonymized?



Entire transaction histories can be compromised.

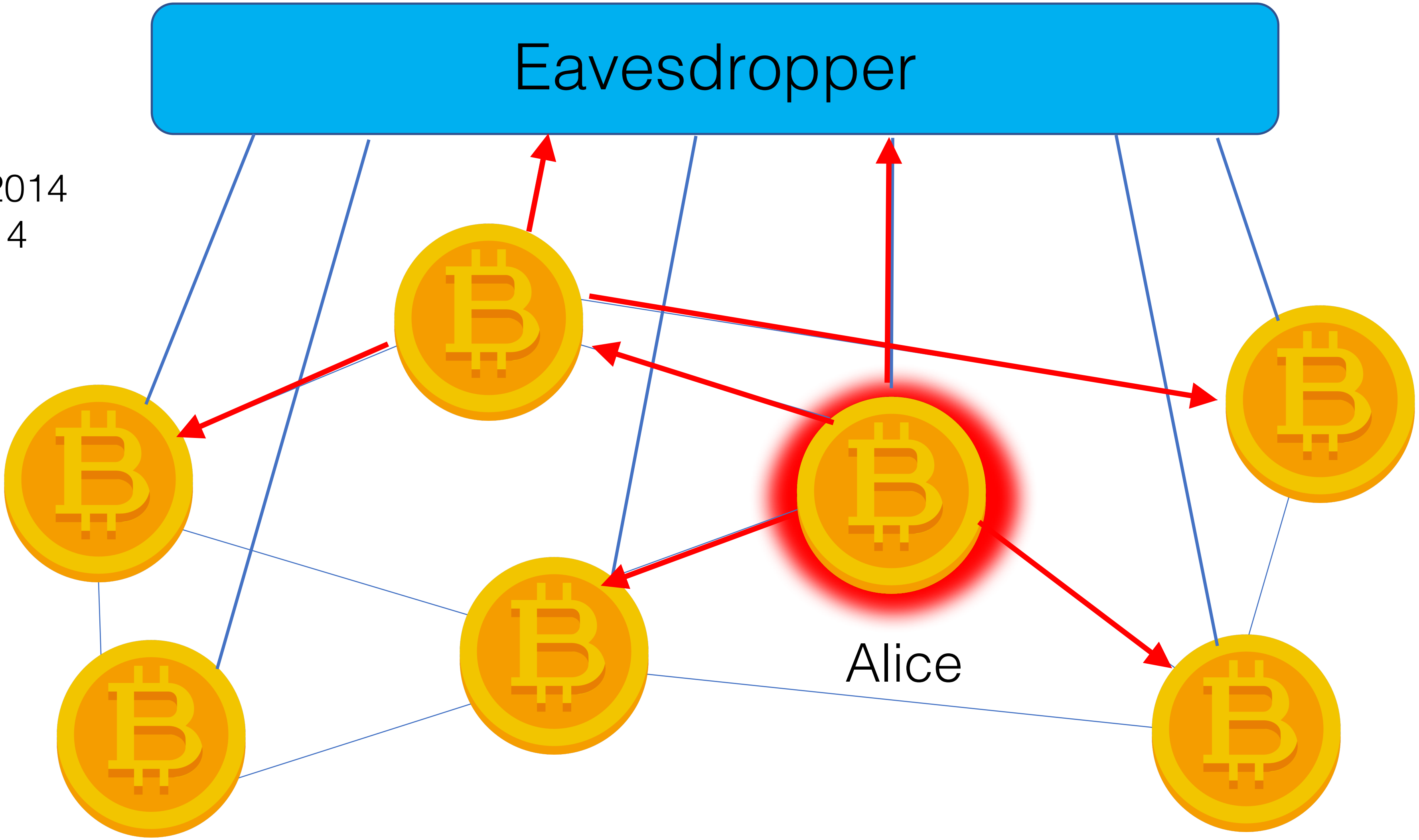
Meiklejohn et al., 2013

What about the peer-to-peer network?

Public Key ⟷ IP Address

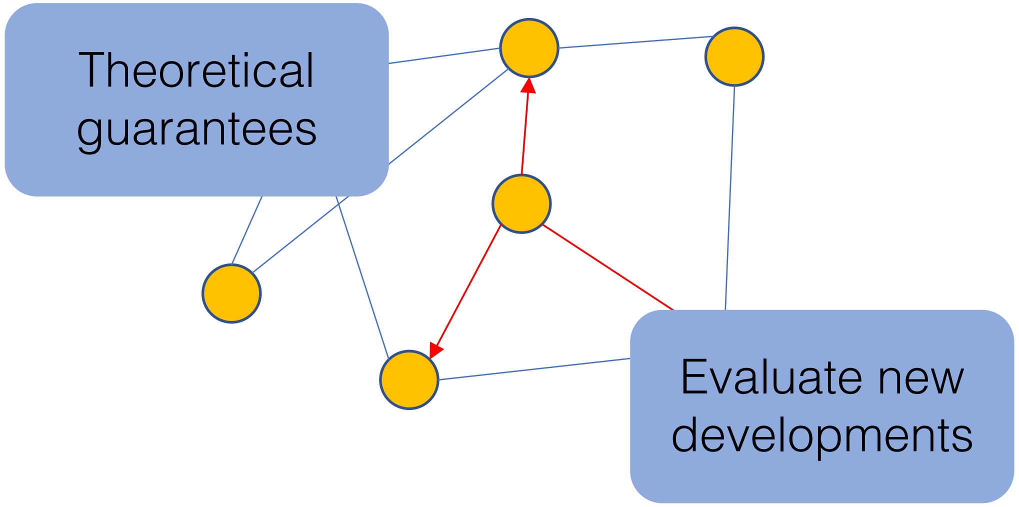
Attacks on the Network Layer

Biryukov et al., 2014
Koshy et al., 2014



Our Work

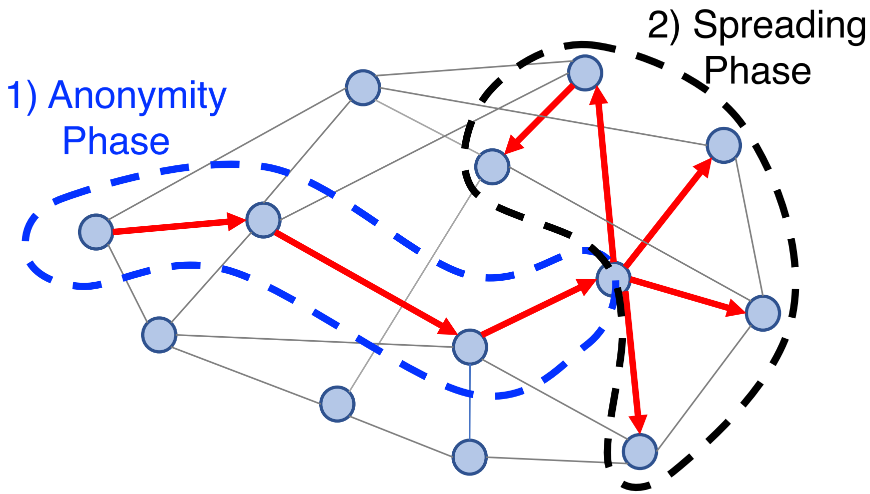
Analysis



Pr(detection)

Under submission, 2017

Redesign



Dandelion

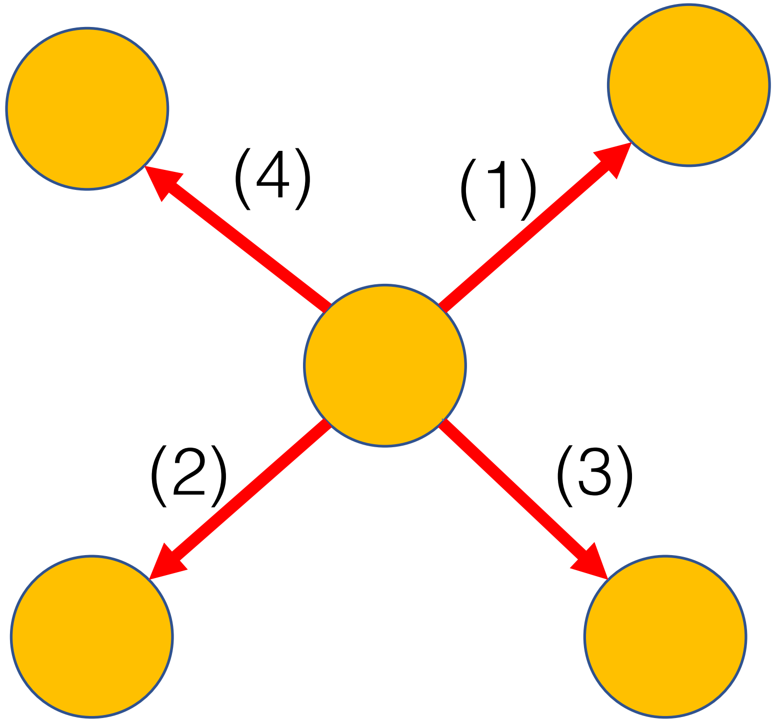
Under submission, 2017

Analysis

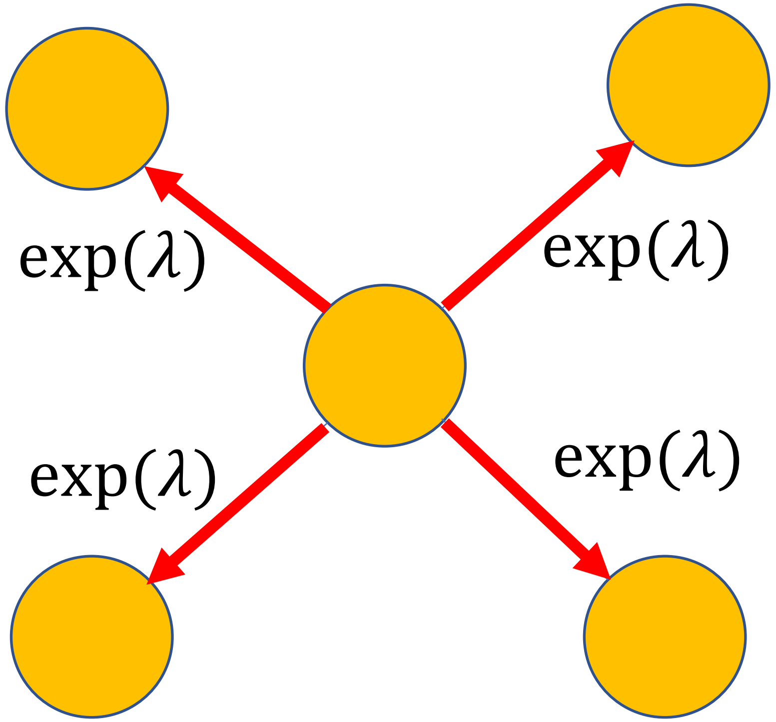
How bad is the problem?

Flooding Protocols

Trickle (pre-2015)

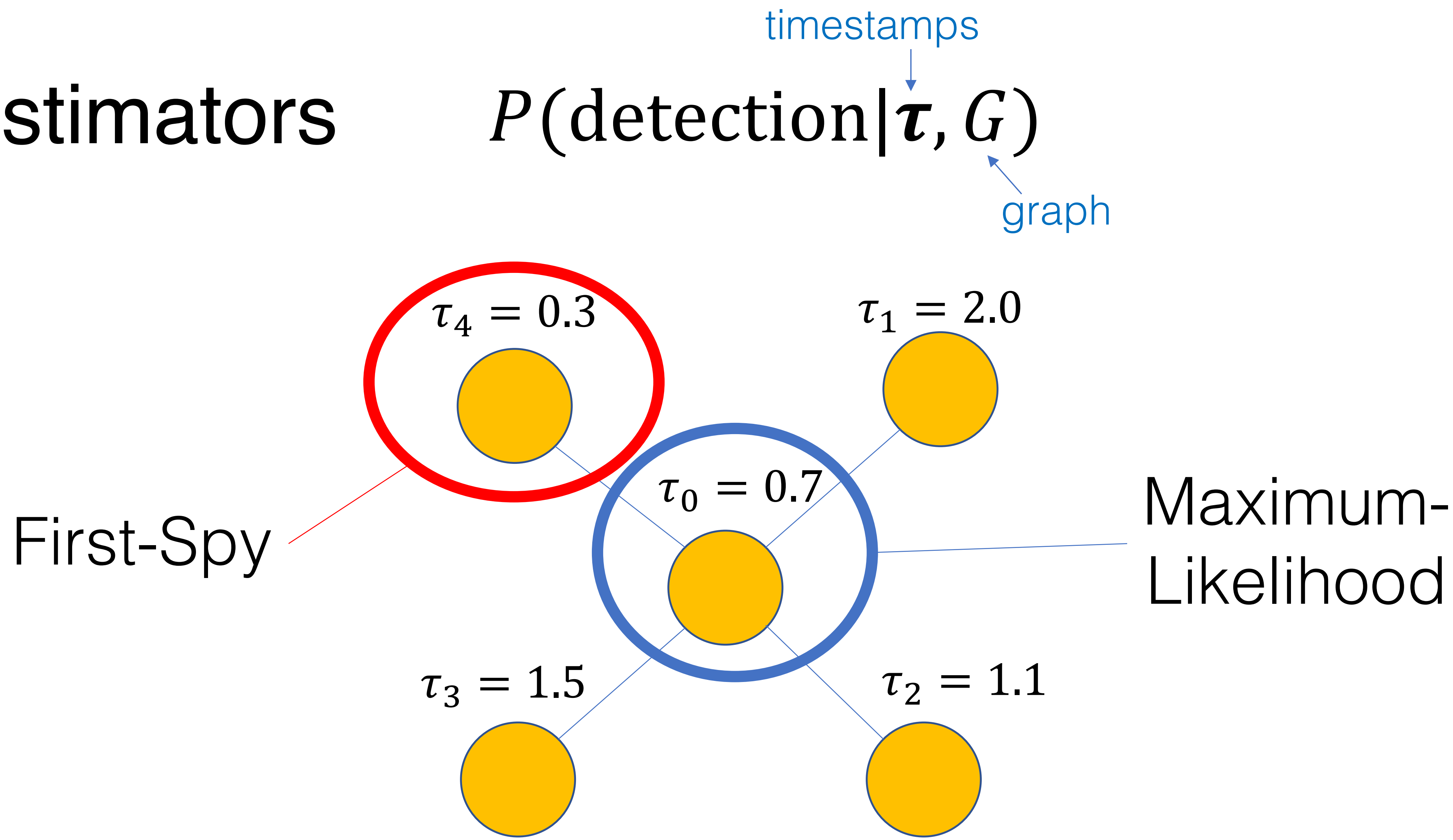


Diffusion (post-2015)



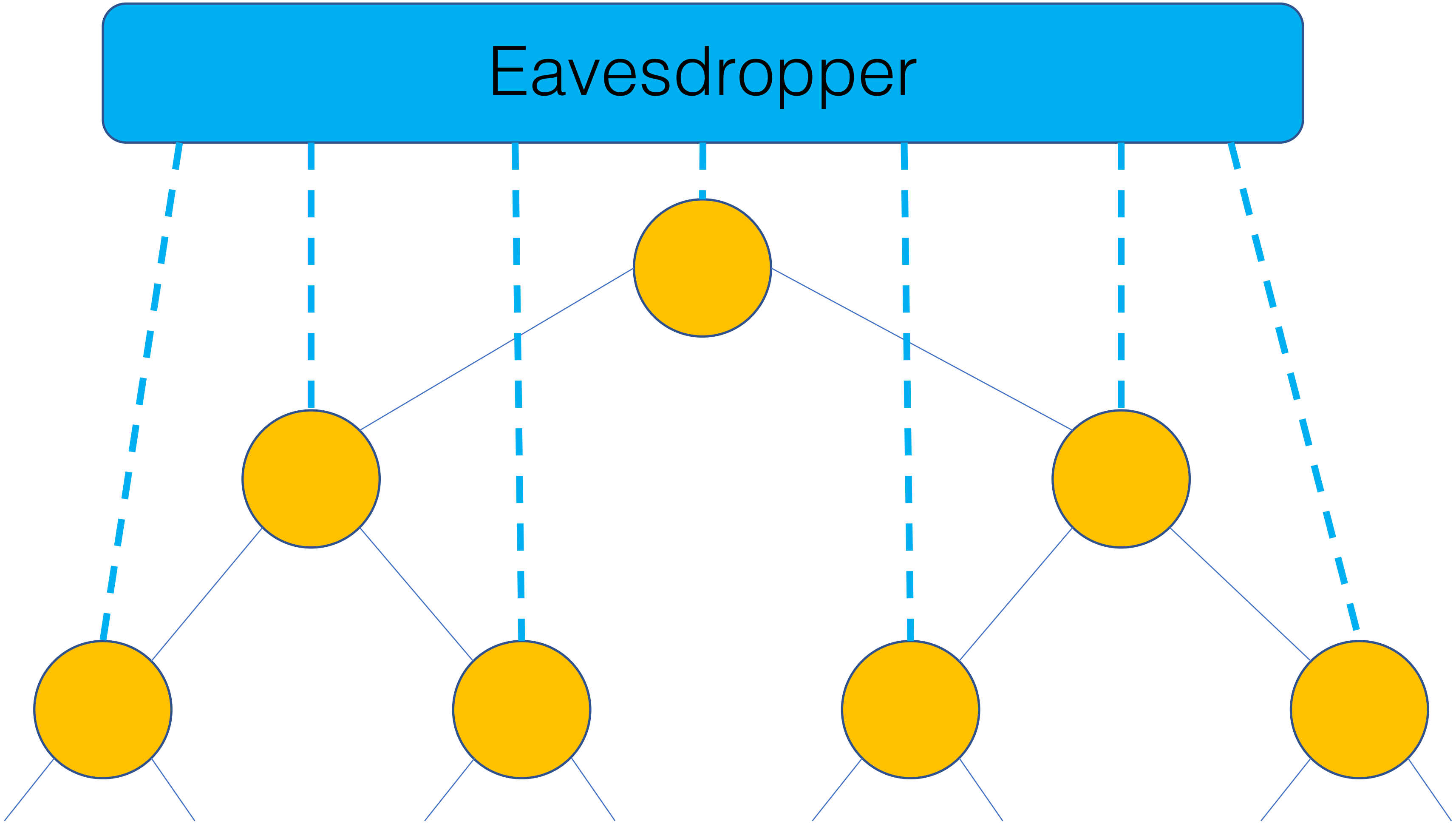
Estimators

$$P(\text{detection}|\boldsymbol{\tau}, G)$$



Does diffusion provide stronger anonymity than trickle spreading?

D-regular trees



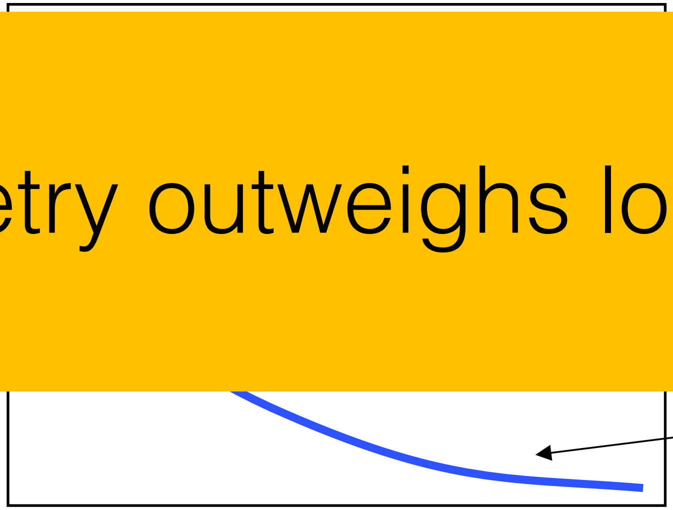
Results: d-Regular Trees

Theorem: The first-spy and maximum-likelihood probabilities of detection for diffusion and trickle are asymptotically identical in d.

Results: d-Regular Trees

	Trickle	Diffusion
First-Timestamp	$O\left(\frac{\log d}{d}\right)$	$O\left(\frac{\log d}{d}\right)$
Maximum-Likelihood	$\Omega(1)$	$\Omega(1)$

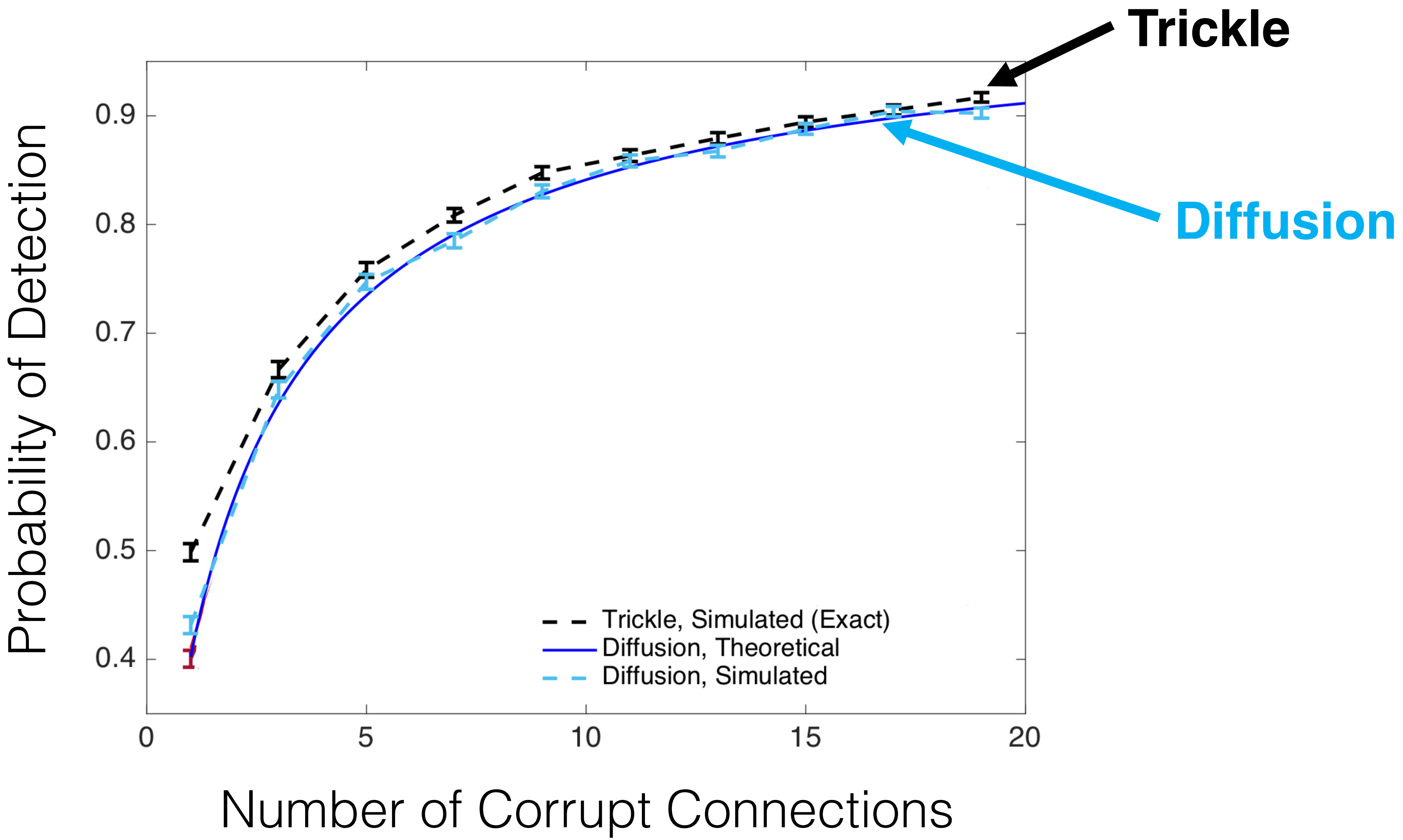
Intuition: Symmetry outweighs local randomness!



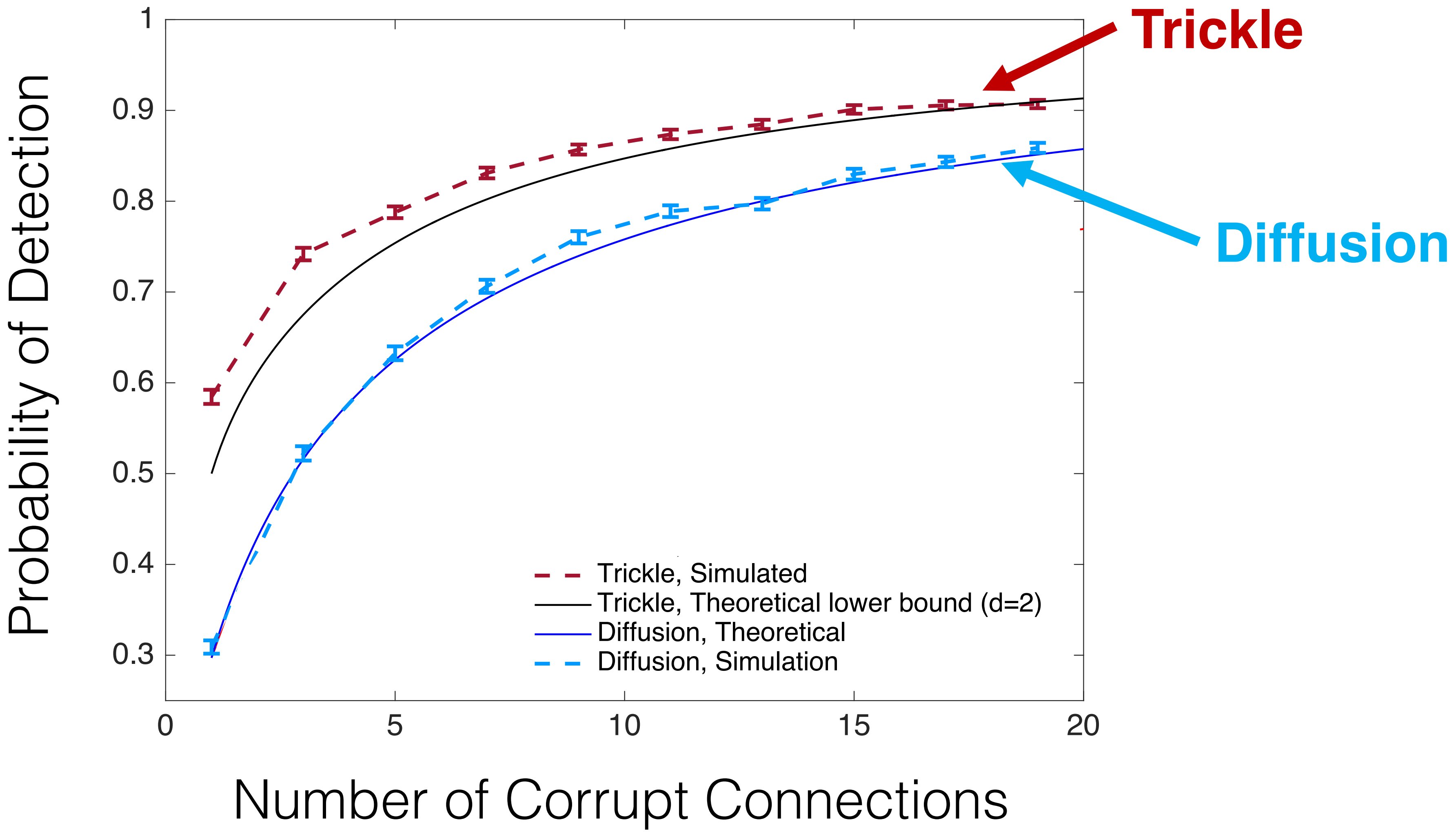
First-timestamp

Degree, d

Results: Trees



Results: Bitcoin Graph

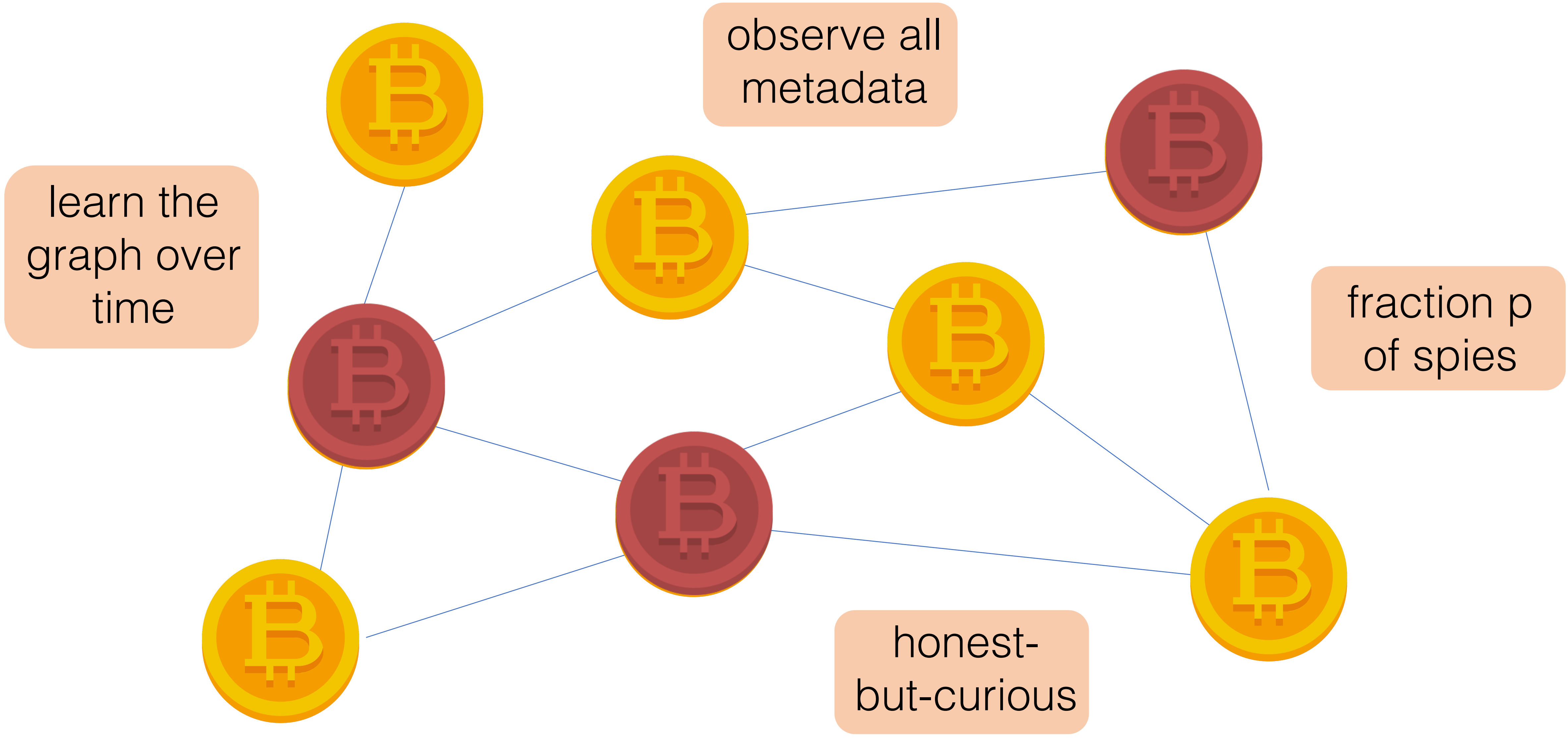


Diffusion does not have (significantly) better anonymity properties than trickle.

Redesign

Can we design a better network?

Adversarial Model



Metric for Anonymity

Recall

$$\frac{1}{n}\sum_{v} 1\{M(v'\text{s tx}) = v\}$$

Number honest users

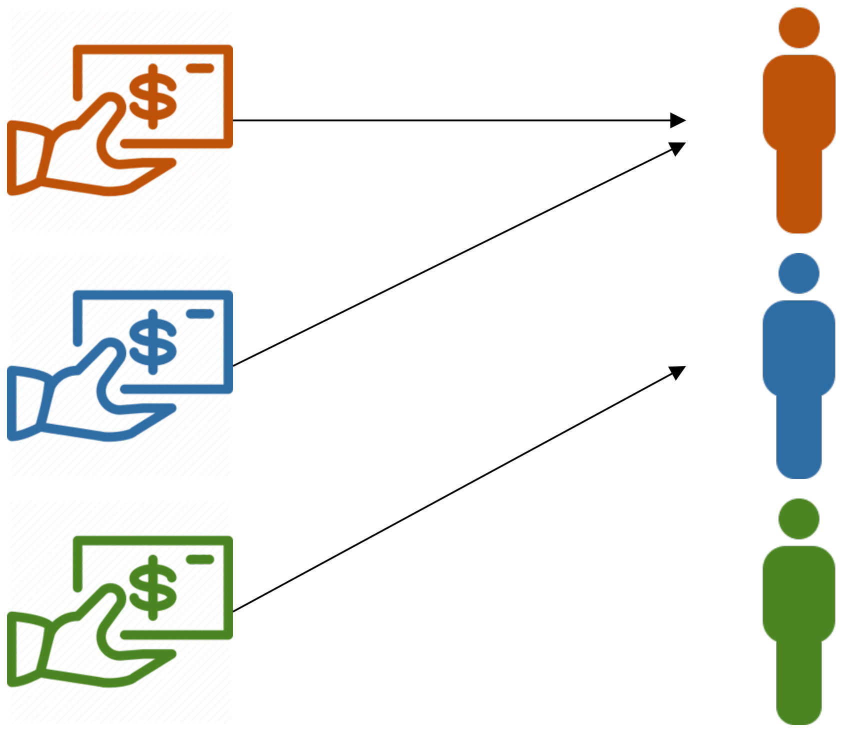
User

Mapping

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

Transactions

Users



Mapping M

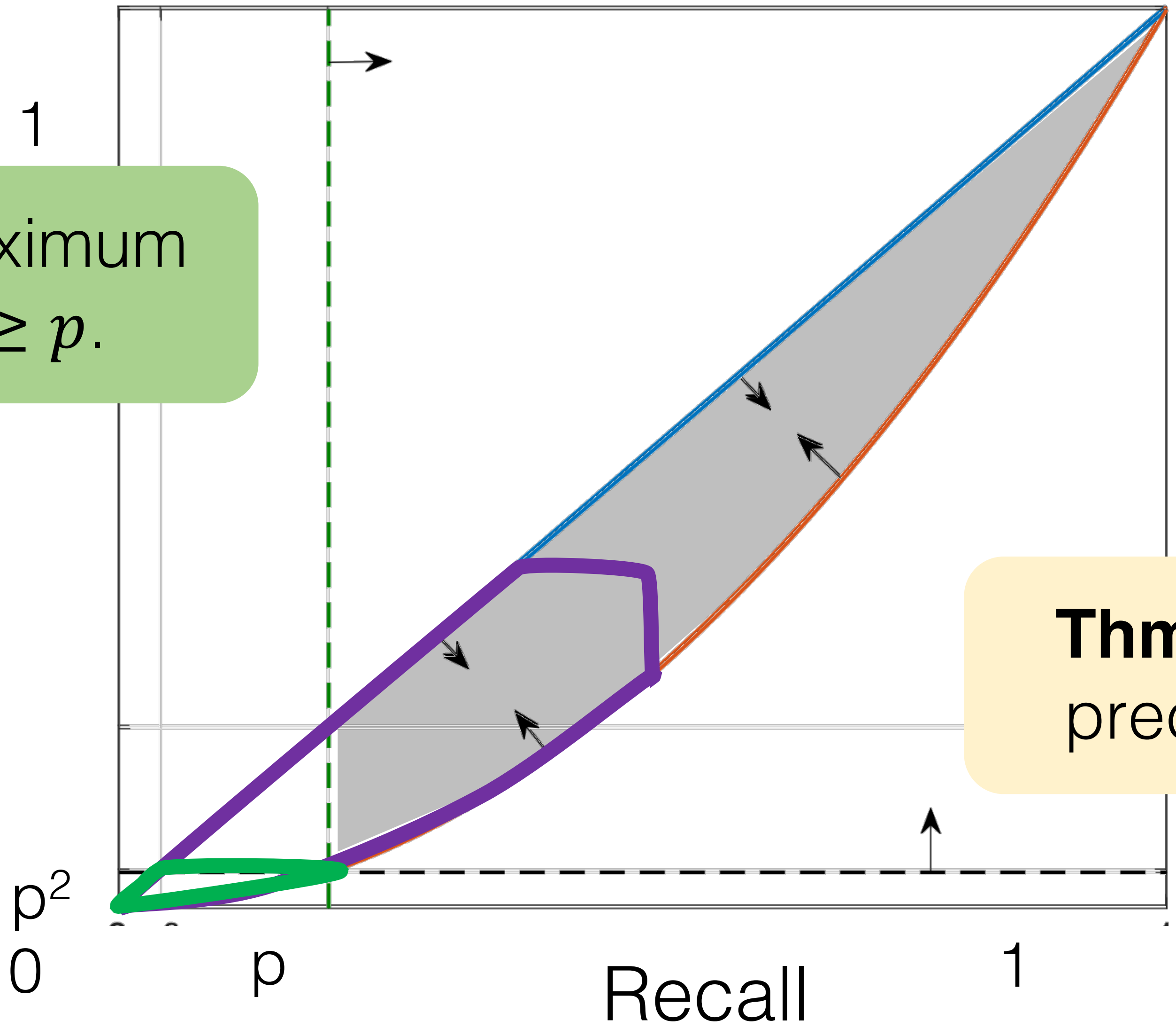
Precision

$$\frac{1}{n}\sum_{v} \frac{1\{M(v'\text{s tx}) = v\}}{\#\text{ tx mapped to v}}$$

Goal:

Design a distributed flooding protocol that minimizes the maximum precision and recall achievable by a computationally-unbounded adversary.

Fundamental Limits

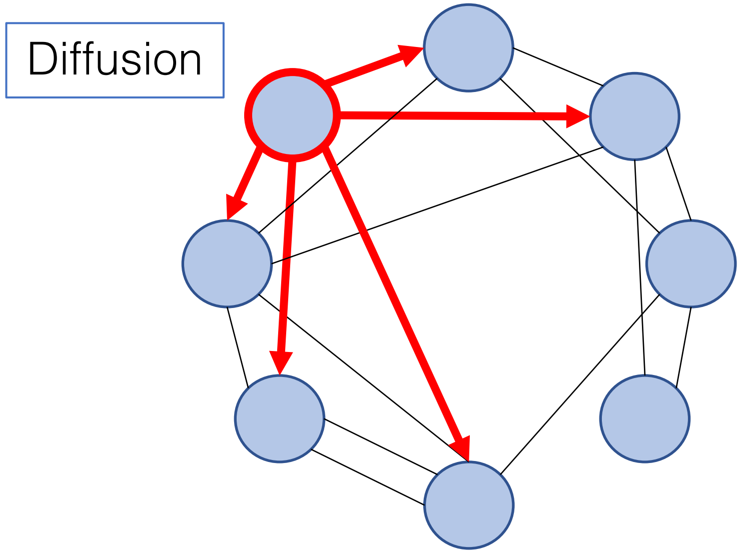


Thm: Maximum recall $\geq p$.

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

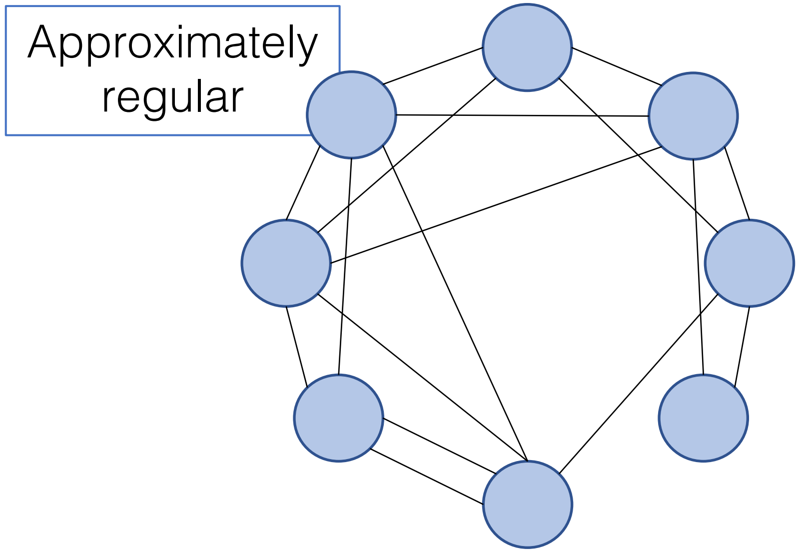
What can we control?

Spreading Protocol



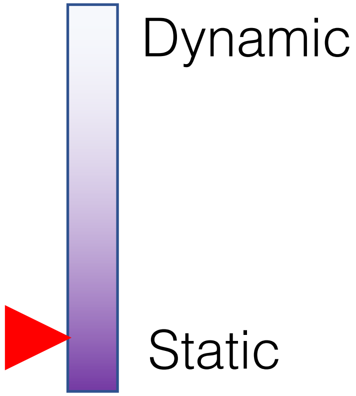
Given a graph, how do we spread content?

Topology



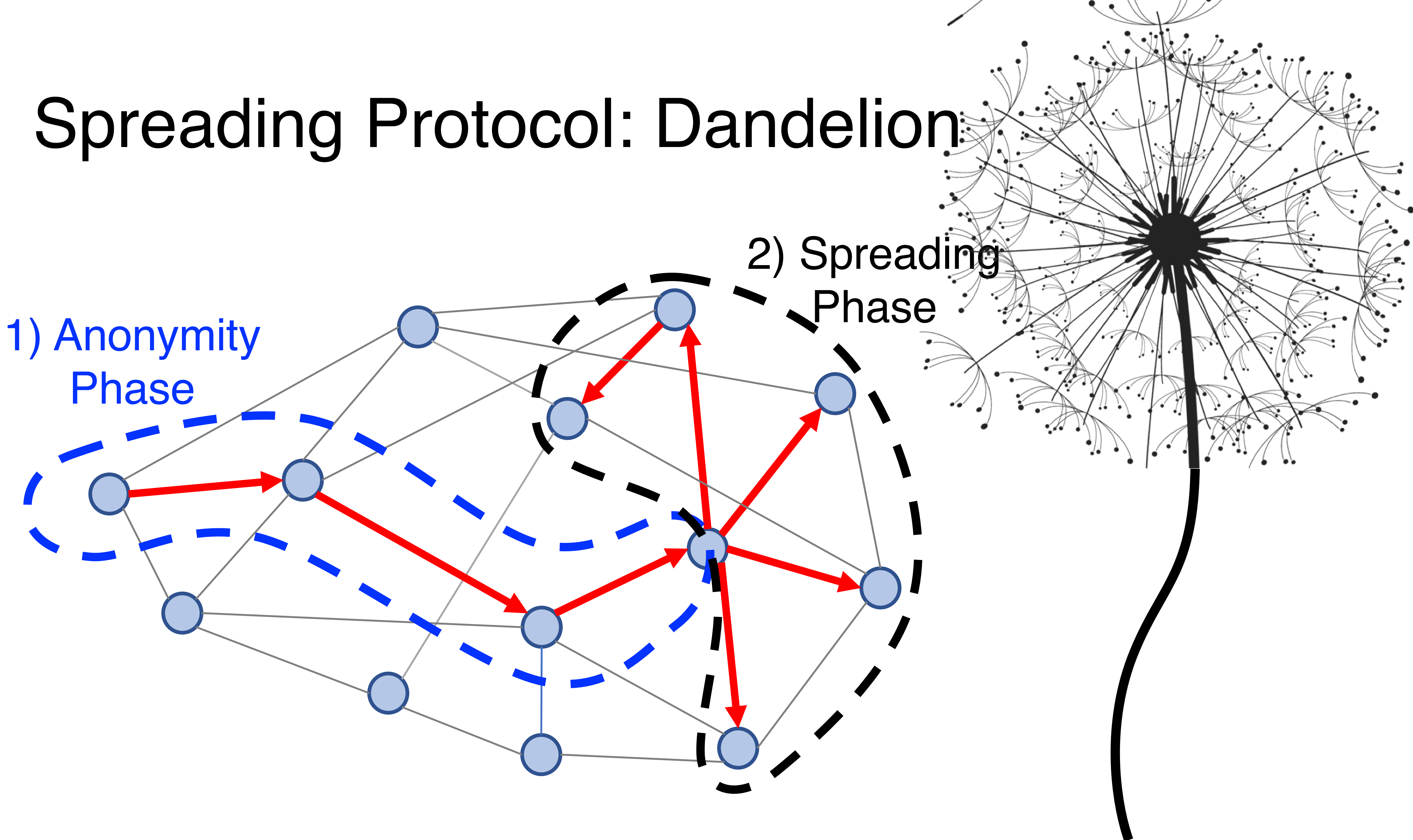
What is the underlying graph topology?

Dynamicity



How often does the graph change?

Spreading Protocol: Dandelion



Why Dandelion spreading?

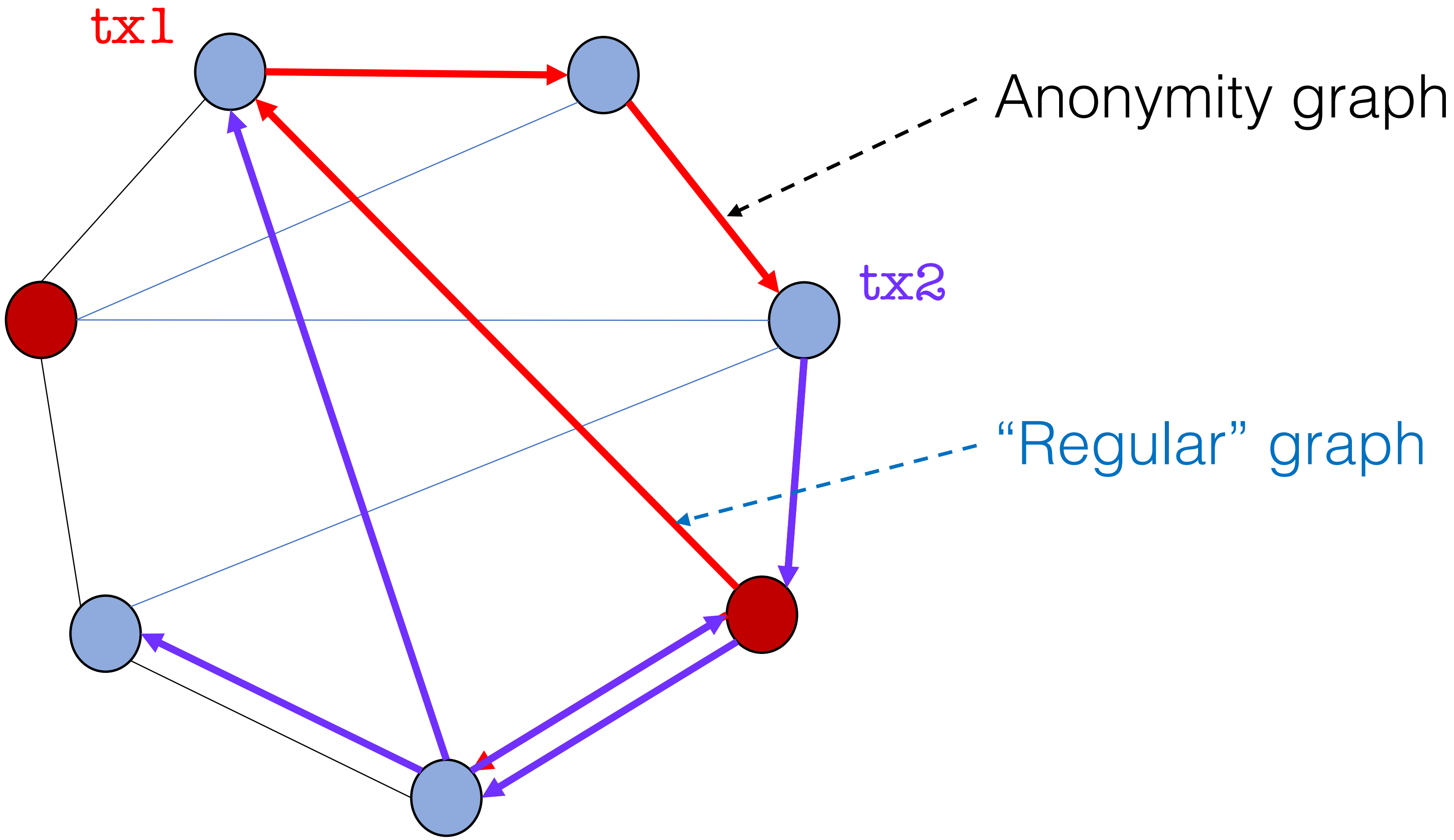
Theorem: Dandelion spreading has an optimally low maximum recall of $p + O\left(\frac{1}{n}\right)$.

lower bound = p

fraction of spies

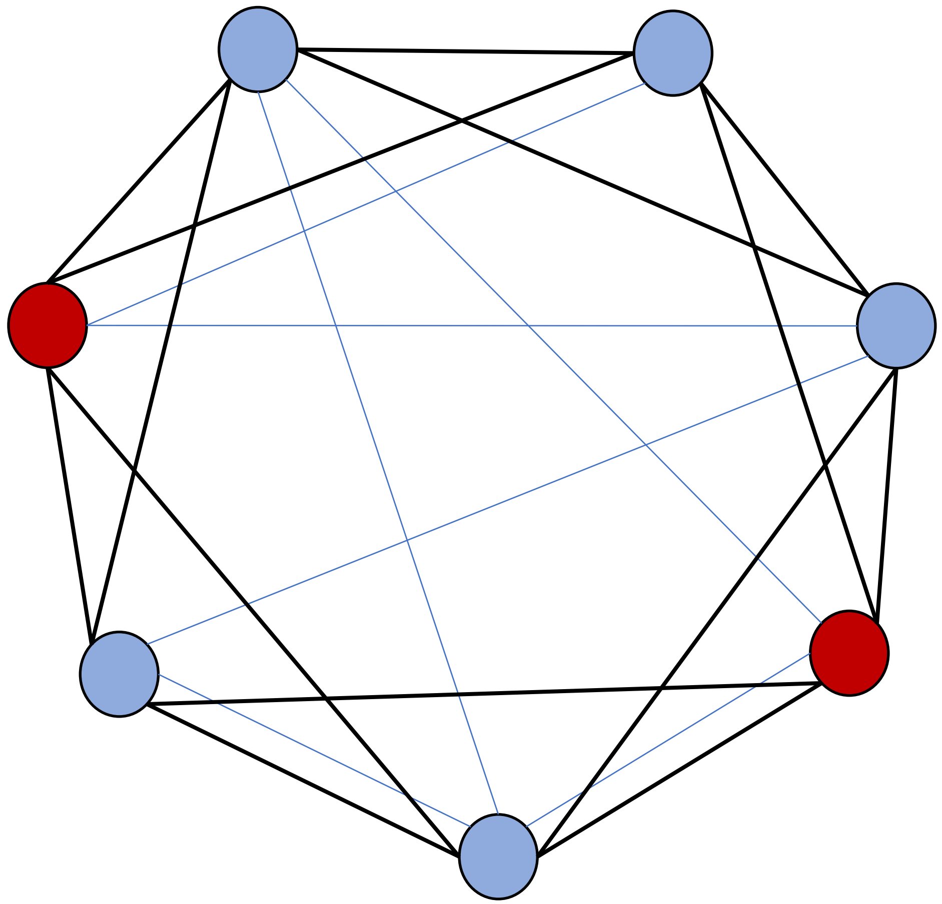
number of nodes

Graph Topology: Line



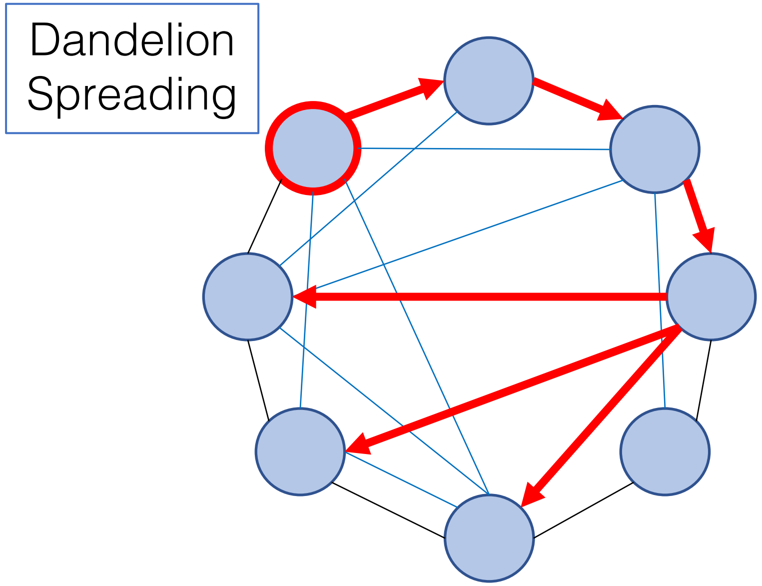
Dynamicity: High

Change the anonymity graph frequently.



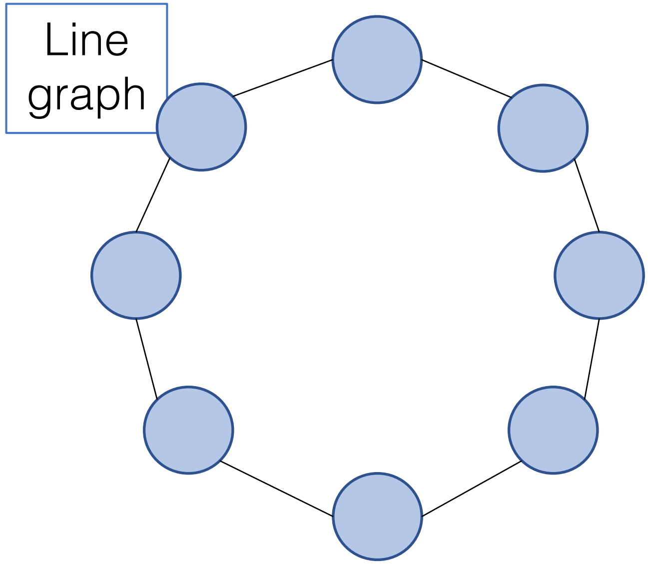
DANDELION Network Policy

Spreading Protocol



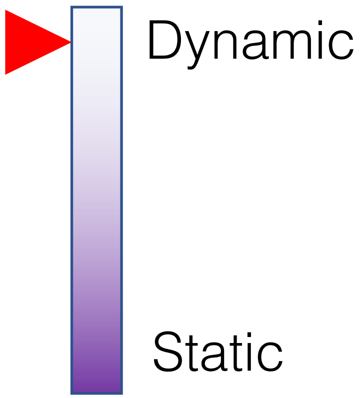
Given a graph, how do we spread content?

Topology



What is the anonymity graph topology?

Dynamicity



How often does the graph change?

What is the precision of DANDELION?

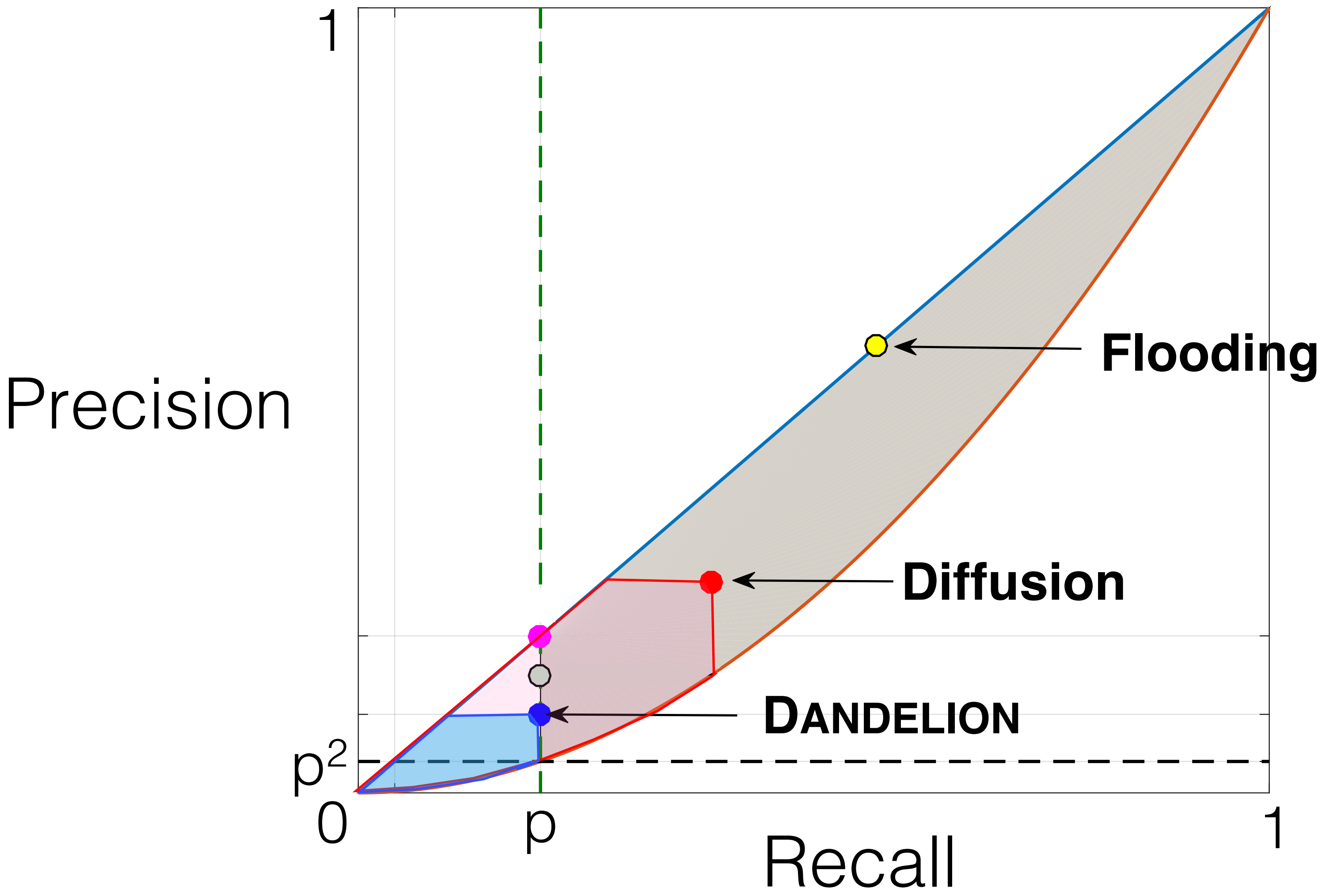
fraction of spies

Theorem: For $p < \frac{1}{3}$, DANDELION has a nearly-optimal maximum precision of $\frac{2p^2}{1-p}\log\left(\frac{2}{p}\right) + O\left(\frac{1}{n}\right)$.

lower bound = p^2

number of nodes

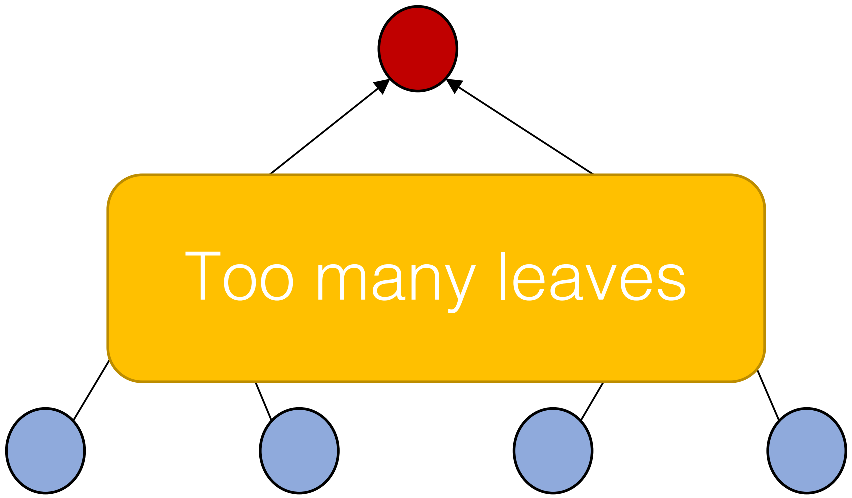
Performance: Achievable Region



Why does DANDELION work?

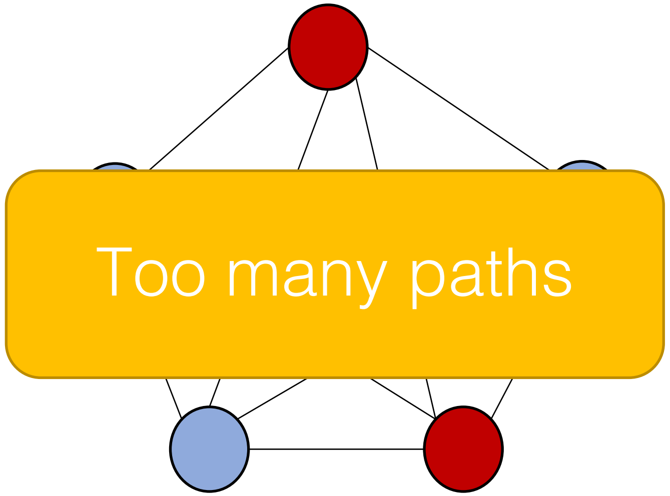
Strong mixing properties.

Tree



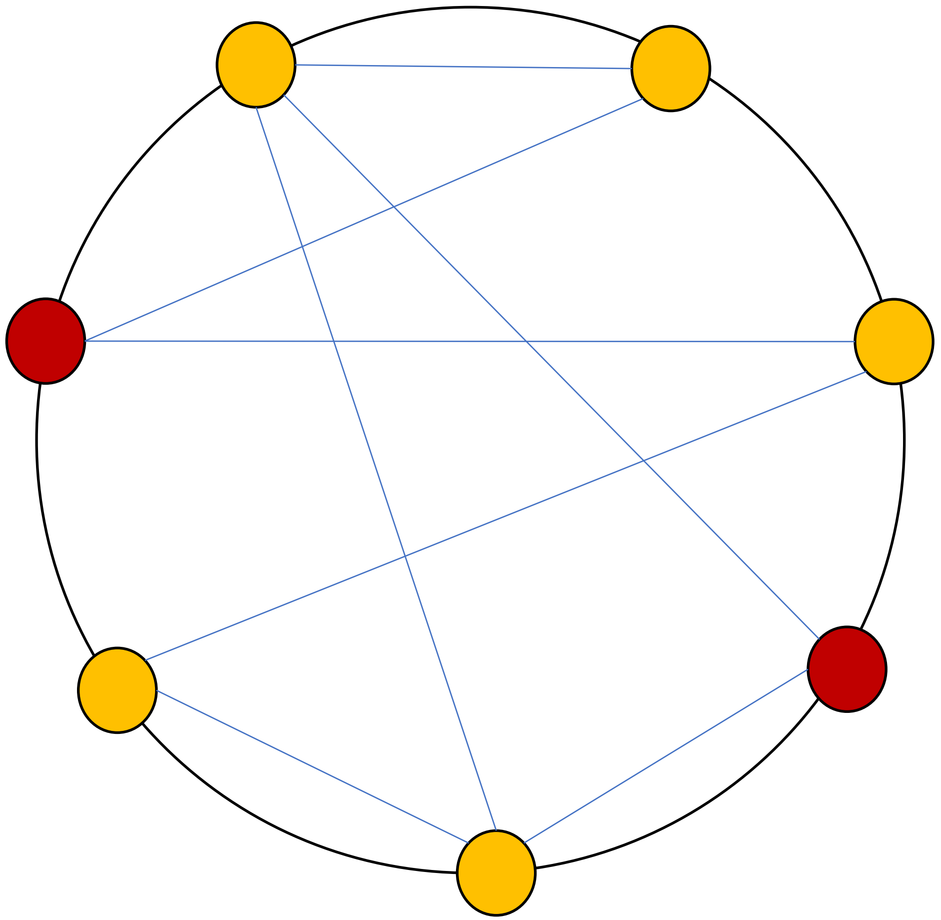
Precision: $O(p)$

Complete graph



Precision: $\frac{p}{1-p}(1-e^{p-1})$

DANDELION vs. Tor, Crowds, etc.



1) Messages propagate over the **same** cycle graph

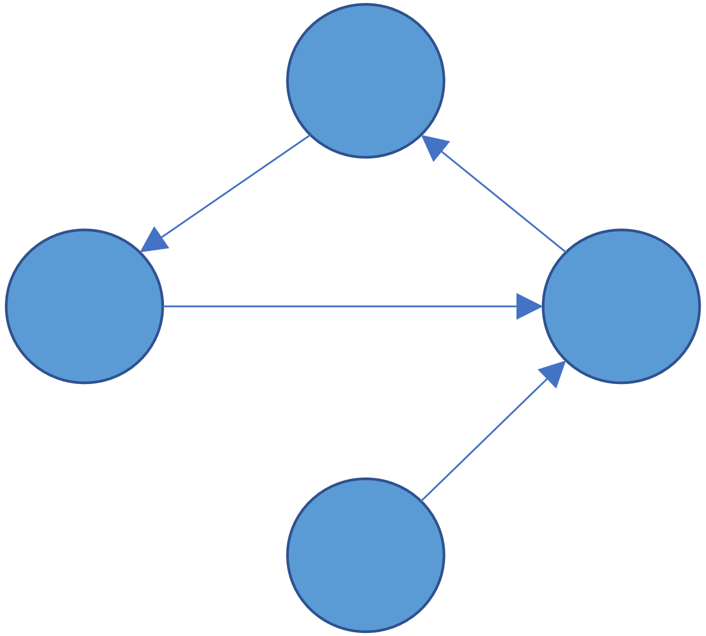
2) Anonymity graph changes dynamically.

3) No encryption required.

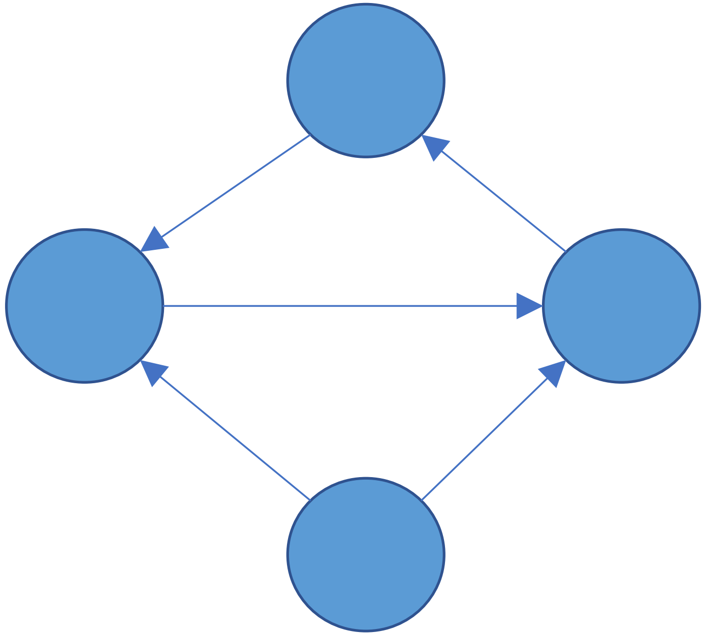
How practical is this?

Implementation

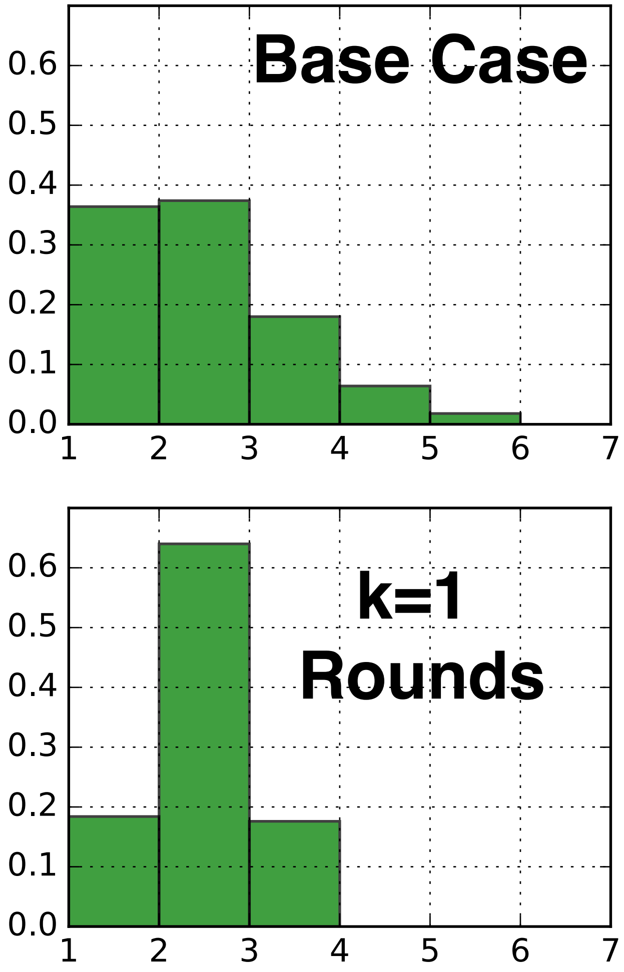
Constructing a Hamiltonian cycle



Base Case



k=1 rounds of Degree-Checking



Degree

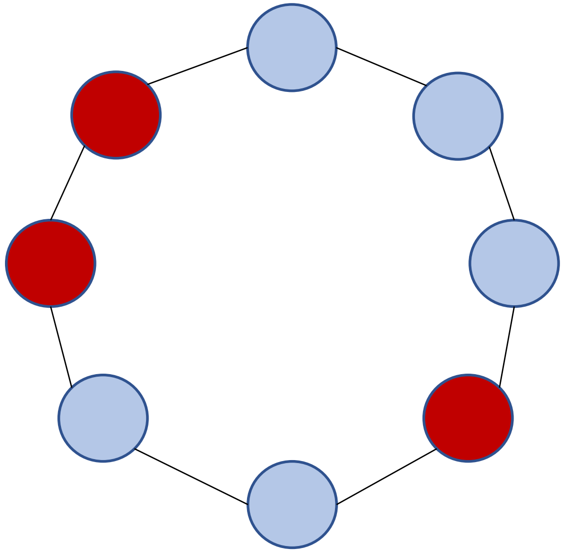
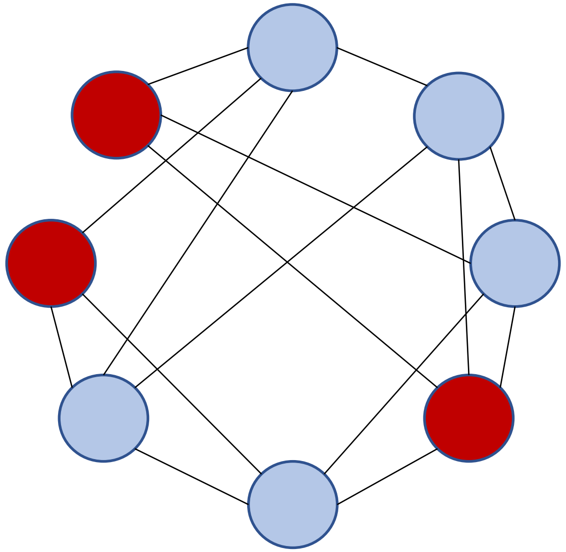
What can the adversary do?

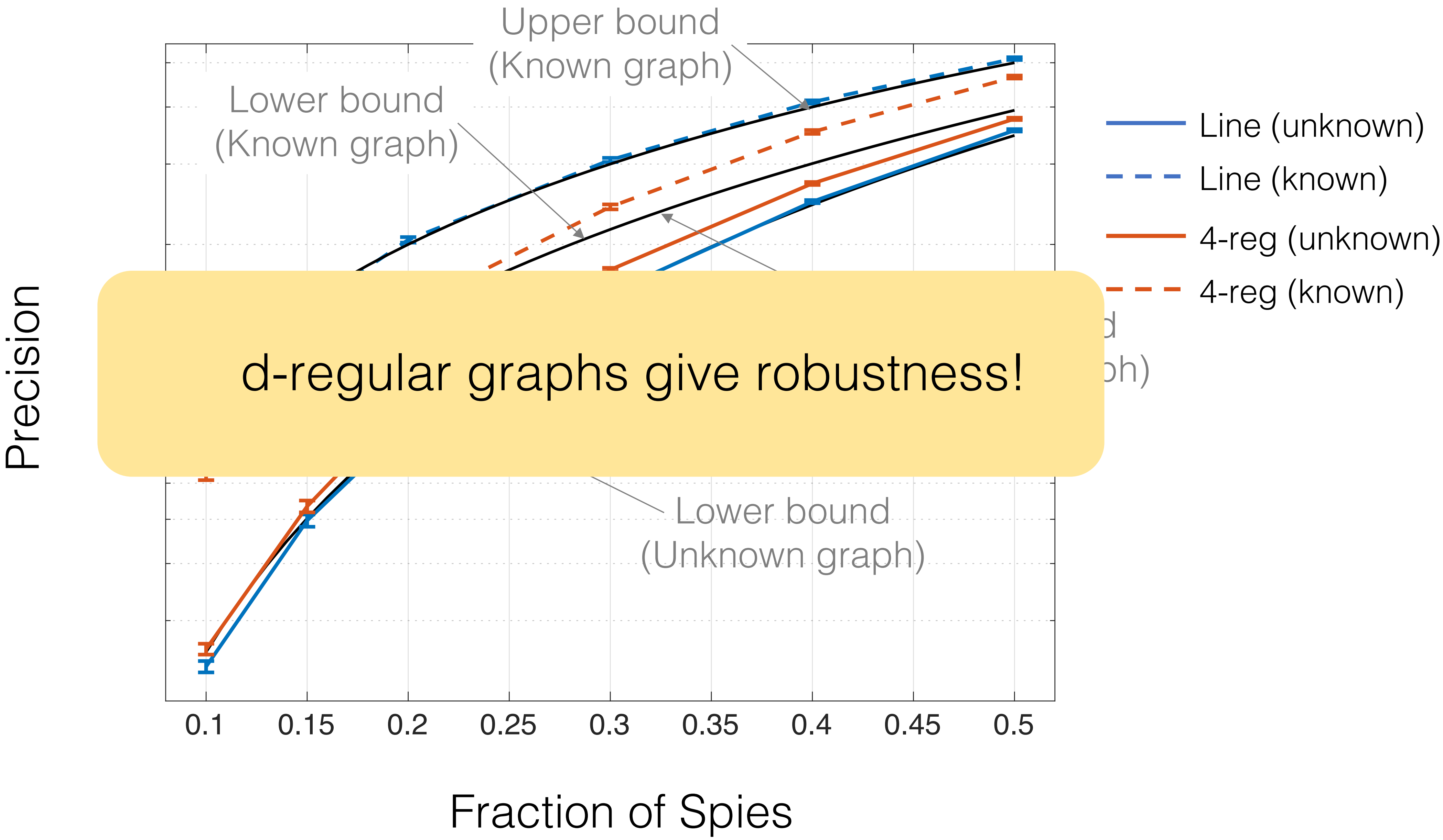
Learn the graph

Misbehave during graph construction

Learning the anonymity graph

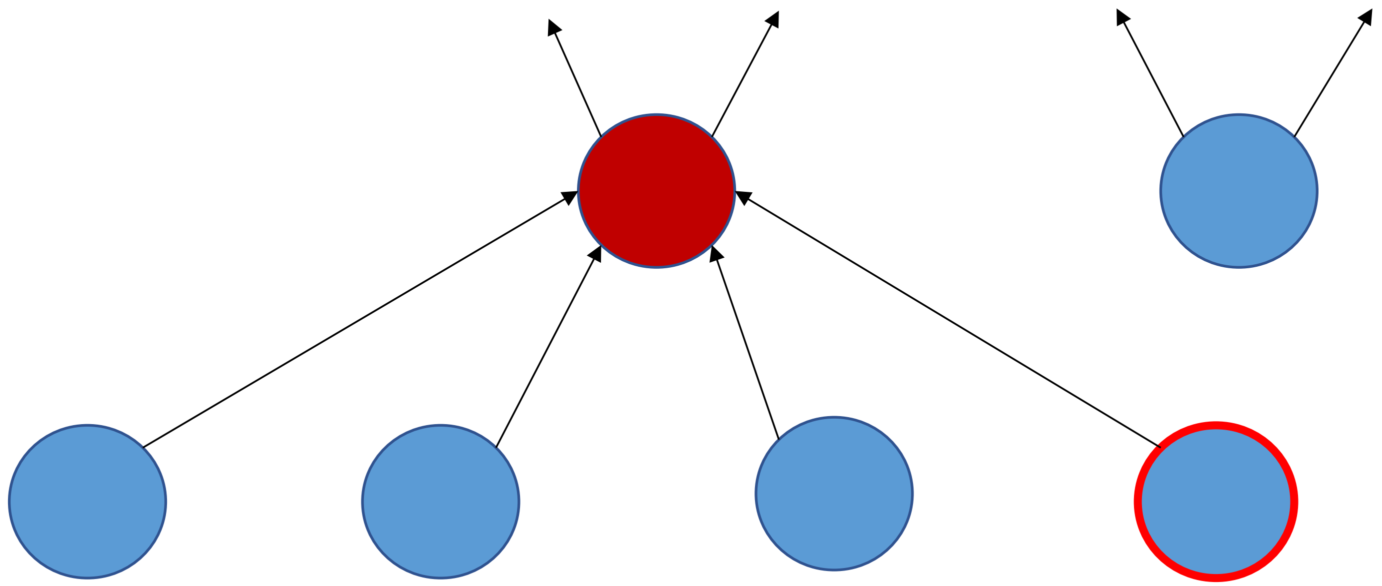
Precision

	Line	Random regular
		
Graph unknown	$O\left(p^2 \log\left(\frac{1}{p}\right)\right)$	**?**
Graph known	$\Omega(p)$	



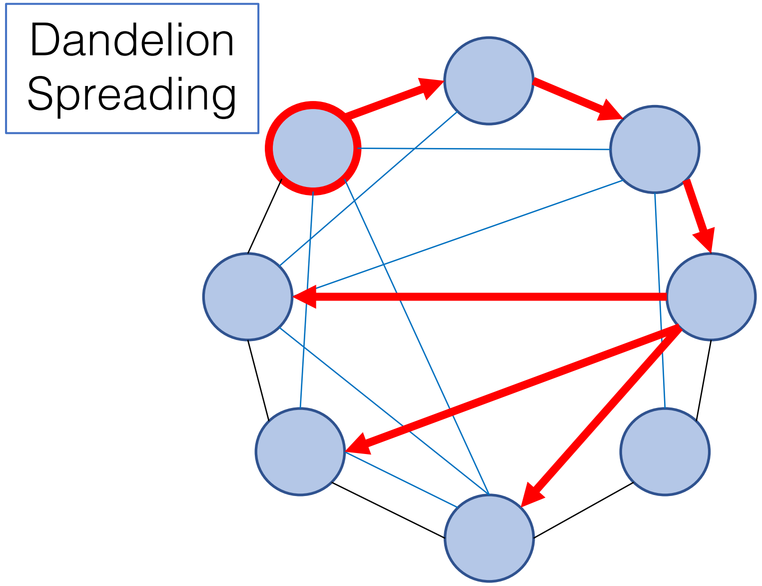
d-regular graphs give robustness!

Manipulating the anonymity graph



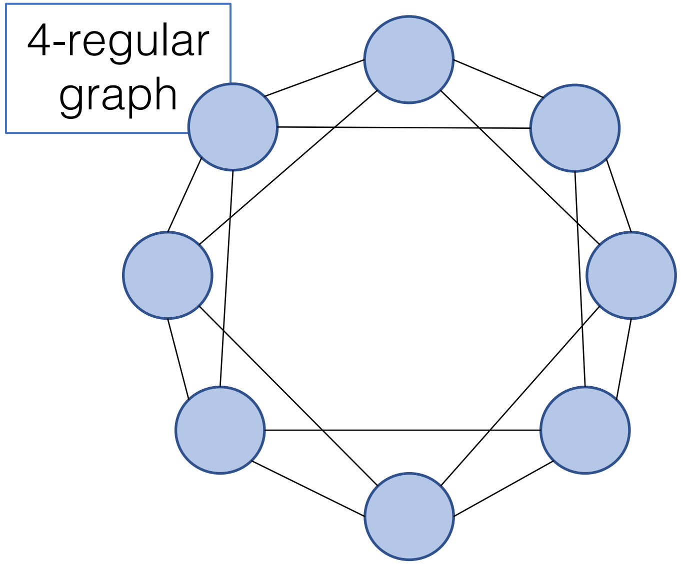
DANDELION++ Network Policy

Spreading Protocol



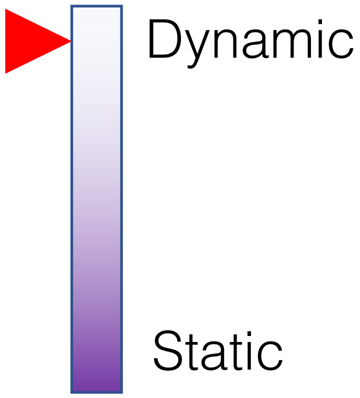
Given a graph, how do we spread content?

Topology



What is the anonymity graph topology?

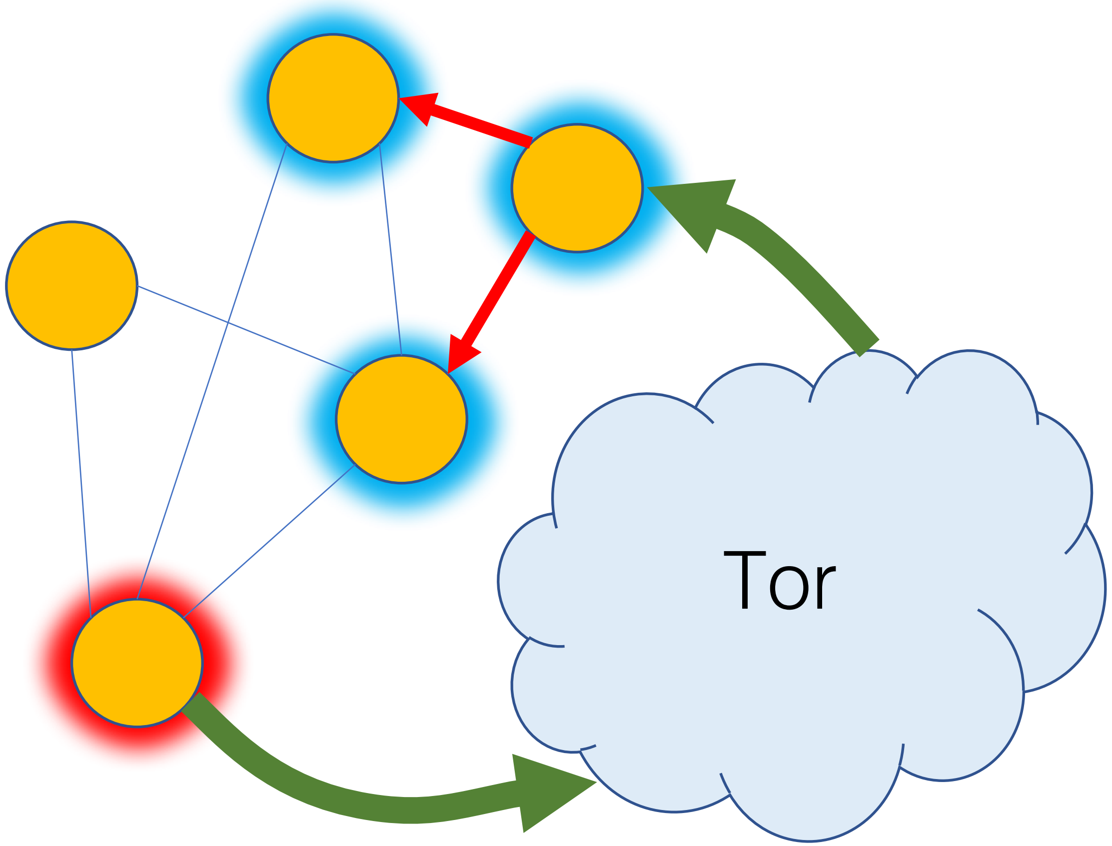
Dynamicity



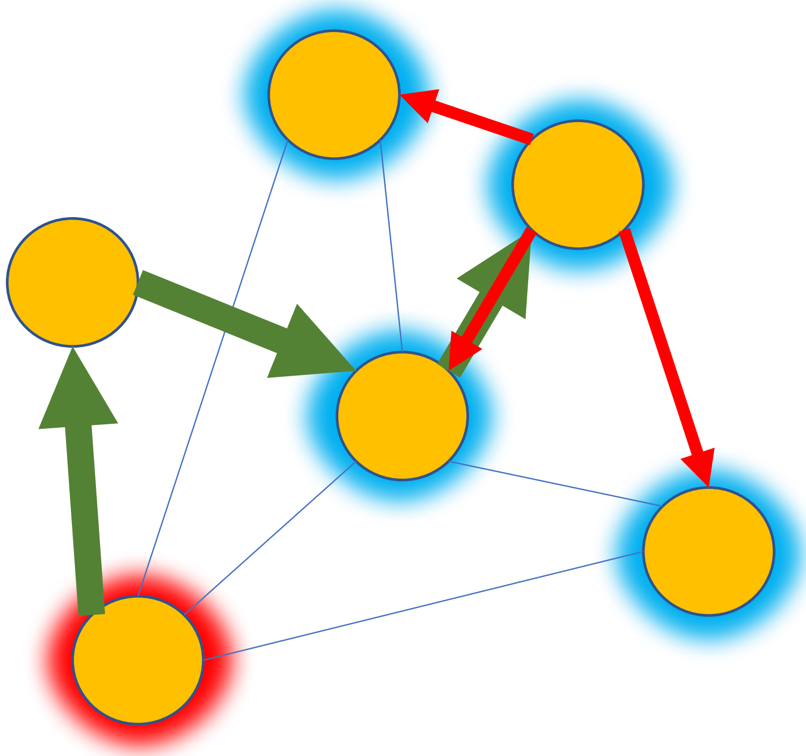
How often does the graph change?

Comparison with Alternative Solutions

Connect through Tor



I2P Integration (e.g. Monero)



Next Steps

Analyze stronger adversaries

Practical demonstration of viability

Take-Home Messages

1) Bitcoin has poor P2P anonymity.

2) Moving from trickle to diffusion did not help.

3) DANDELION++ may be a lightweight solution for certain classes of adversaries.