An Efficient reconciliation algorithm for social networks

Silvio Lattanzi (Google Research NY)

Joint work with:
Nitish Korula (Google Research NY)

ICERM
Stochastic Graph Models

Outline

- Graph reconciliation
  Model and theoretical results.

- Experimental results
  From theory to practice.

- Open problems and future directions

Graph reconciliation

Real world motivations

WIKIPEDIA

English
The Free Encyclopedia
4 471 000+ articles

Español
La enciclopedia libre
1 087 000+ artículos

日本語
フリー百科事典
900 000+ 記事

Deutsch
Die freie Enzyklopädie
1 697 000+ Artikel

Русский
Свободная энциклопедия
1 096 000+ статей

Français
L'encyclopédie libre
1 485 000+ articles

Italiano
L'enciclopedia libera
1 106 000+ voci

Português
A enciclopédia livre
822 000+ artigos

Polski
Wolna encyklopedia
1 034 000+ haseł

中文
自由的百科全書
755 000+ 條目

Real world motivations

WIKIPEDIA

English
The Free Encyclopedia
4 471 000+ articles

Español
La enciclopedia libre
1 087 000+ artículos

日本語
フリー百科事典
900 000+ 記事

Deutsch
Die freie Enzyklopädie
1 697 000+ Artikel

Русский
Свободная энциклопедия
1 096 000+ статей

Français
L'encyclopédie libre
1 485 000+ articles

Italiano
L'enciclopedia libera
1 106 000+ voci

Português
A enciclopédia livre
822 000+ artigos

Polski
Wolna encyklopedia
1 034 000+ haseł

中文
自由的百科全書
755 000+ 條目

Intra-language network

Real world motivations

WIKIPEDIA

English
The Free Encyclopedia
4 471 000+ articles

Español
La enciclopedia libre
1 087 000+ artículos

日本語
フリー百科事典
900 000+ 記事

Deutsch
Die freie Enzyklopädie
1 697 000+ Artikel

Русский
Свободная энциклопедия
1 096 000+ статей

Français
L'encyclopédie libre
1 485 000+ articles

Italiano
L'enciclopedia libera
1 106 000+ voci

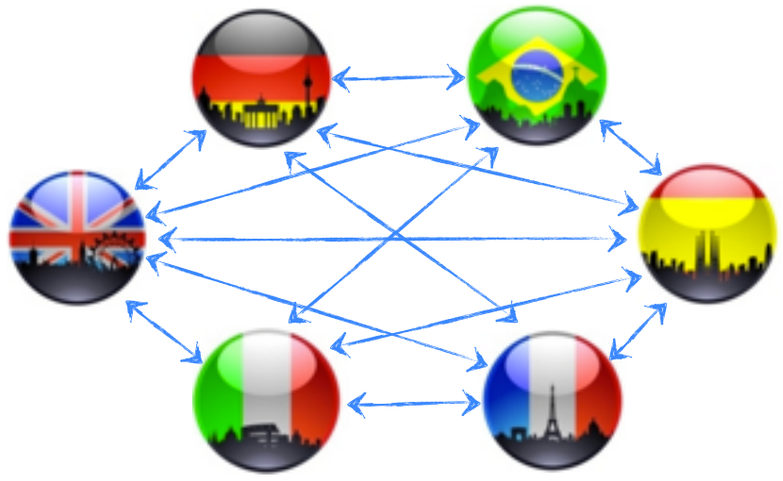
Português
A enciclopédia livre
822 000+ artigos

Polski
Wolna encyklopedia
1 034 000+ haseł

中文
自由的百科全書
755 000+ 條目

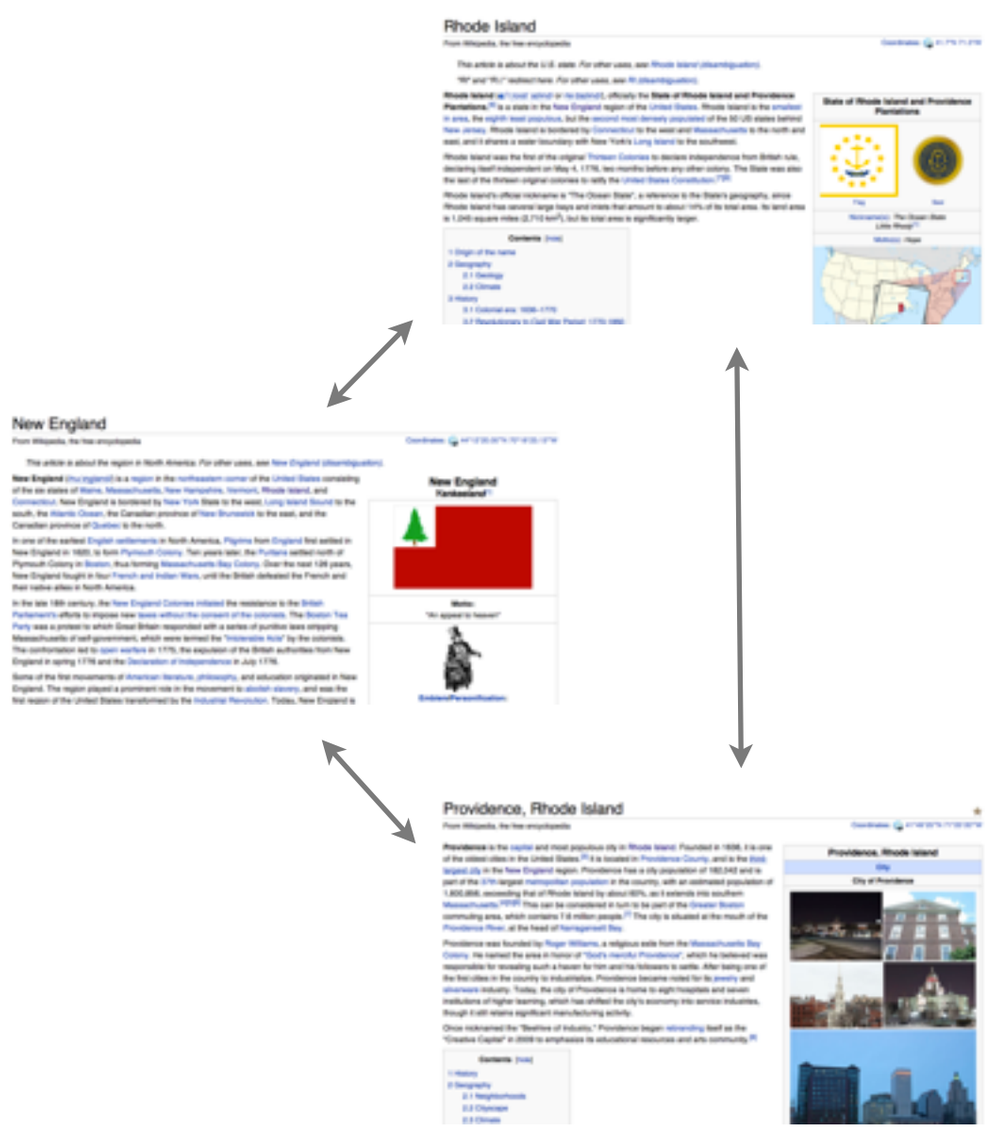
Intra-language network

Inter-language network



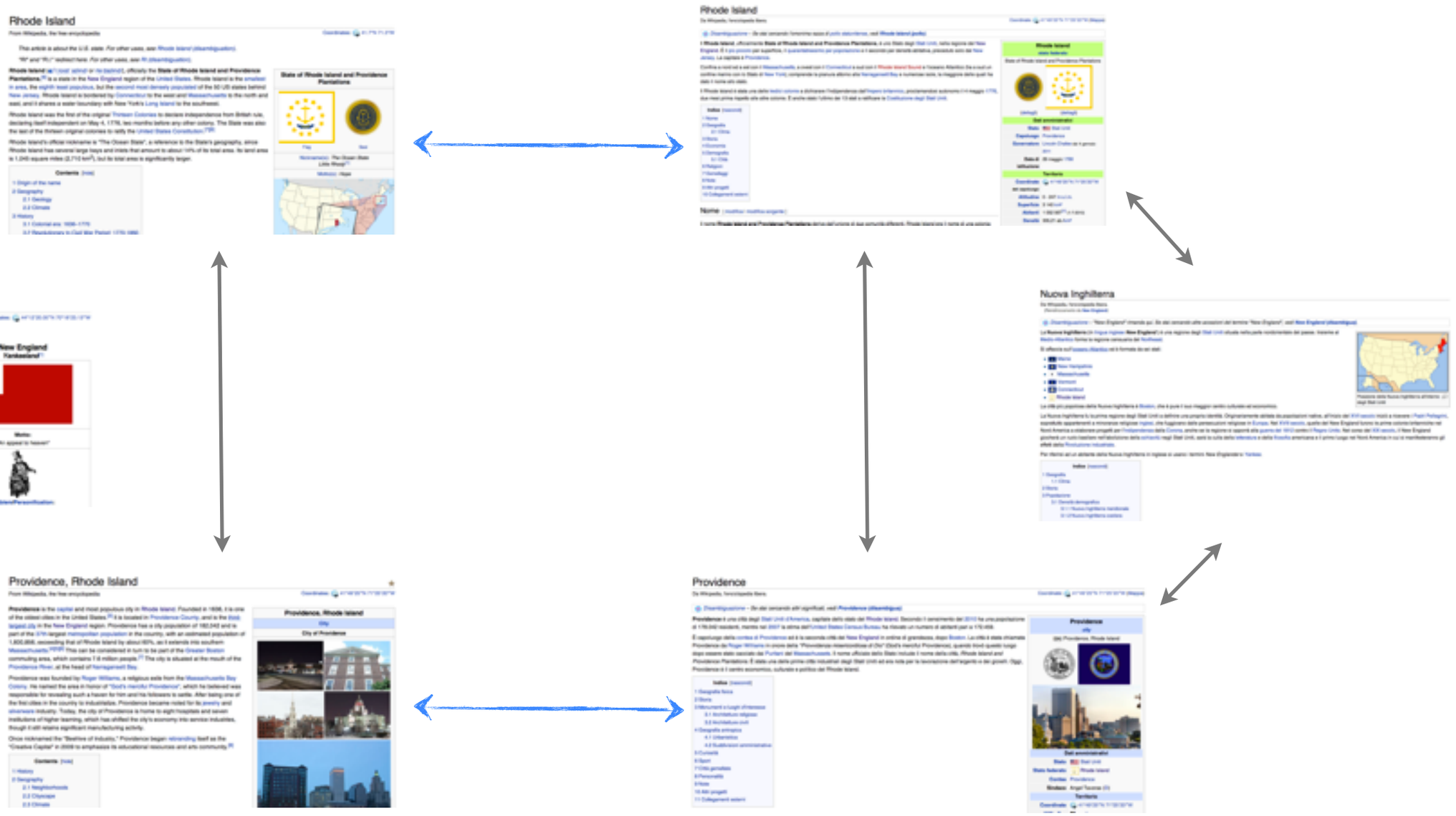
Real world motivations

Can we use intra-language information to improve inter-language graph?



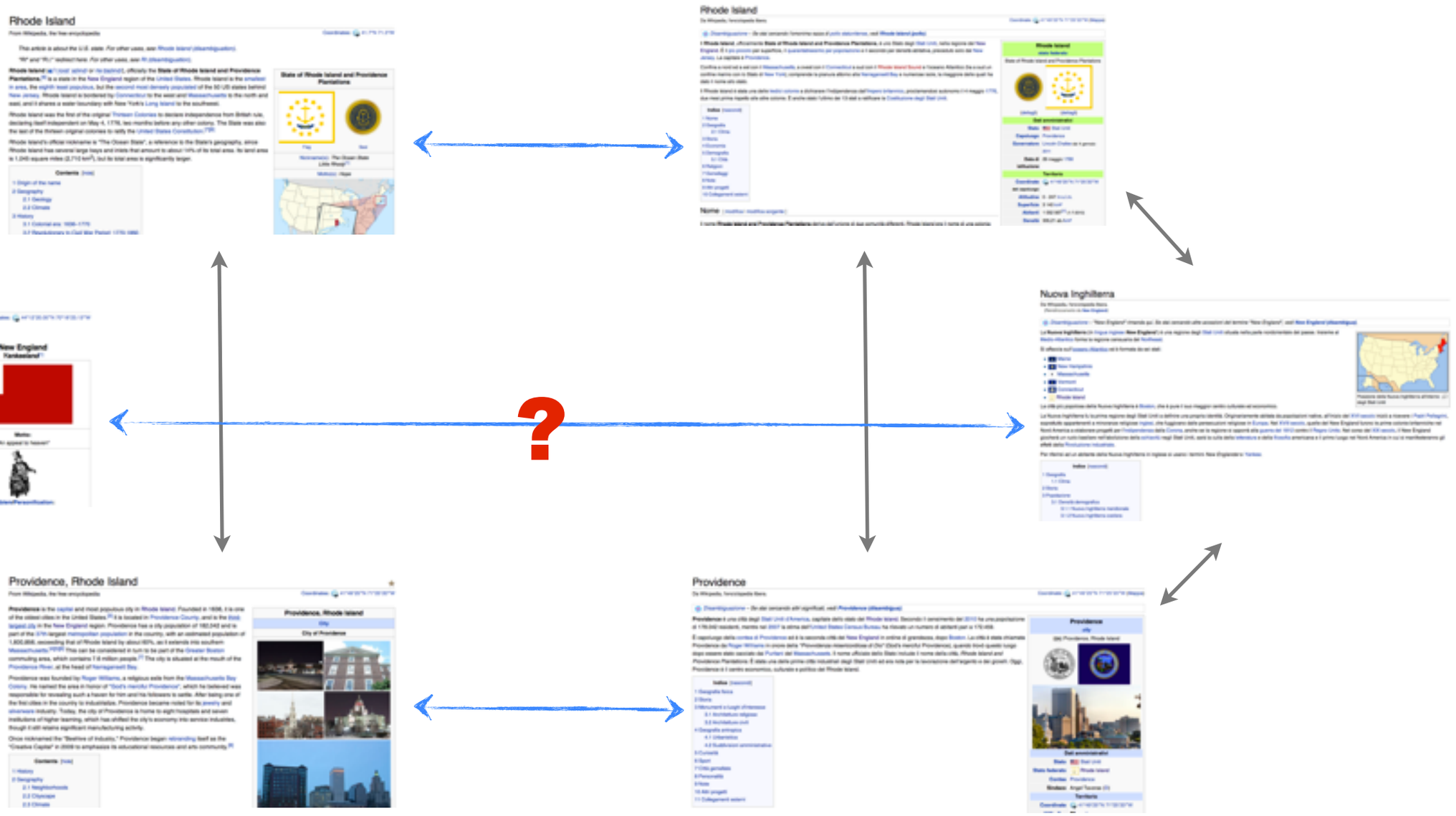
Real world motivations

Can we use intra-language information to improve inter-language graph?

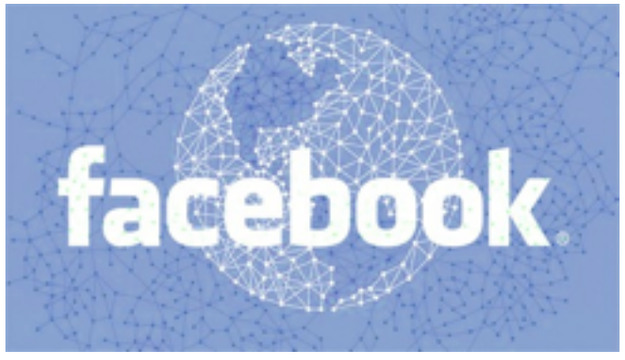


Real world motivations

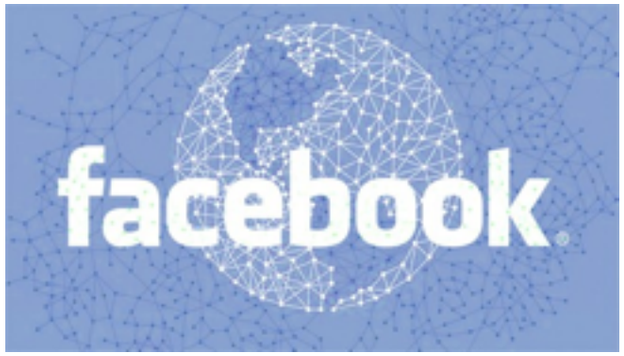
Can we use intra-language information to improve inter-language graph?

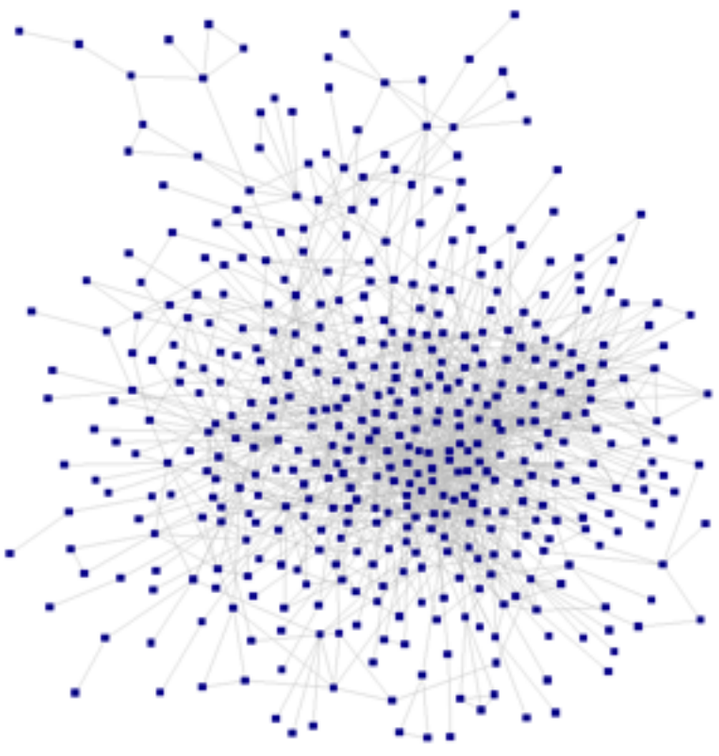


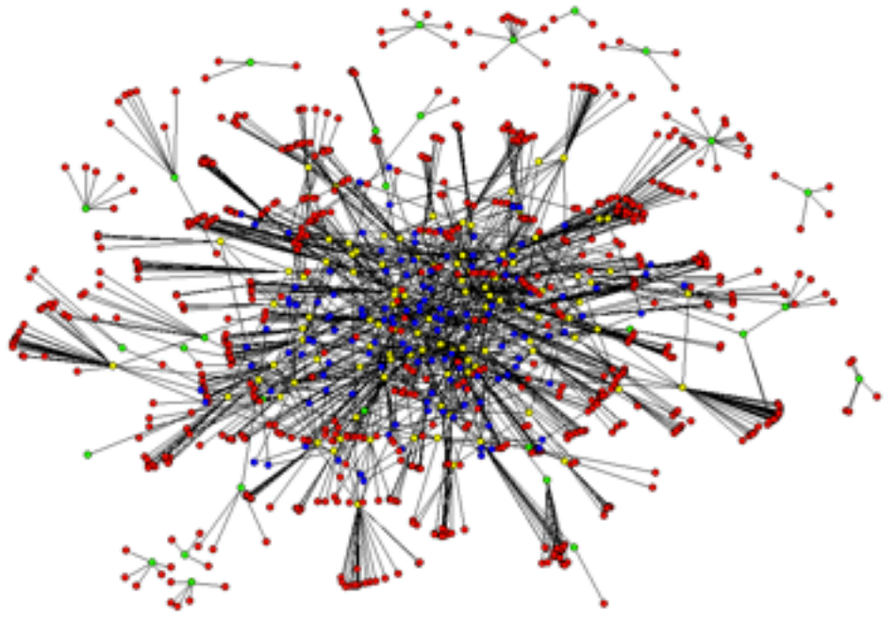
Real world motivations



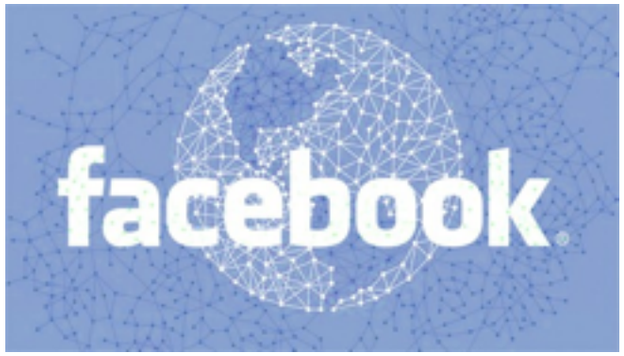
Real world motivations

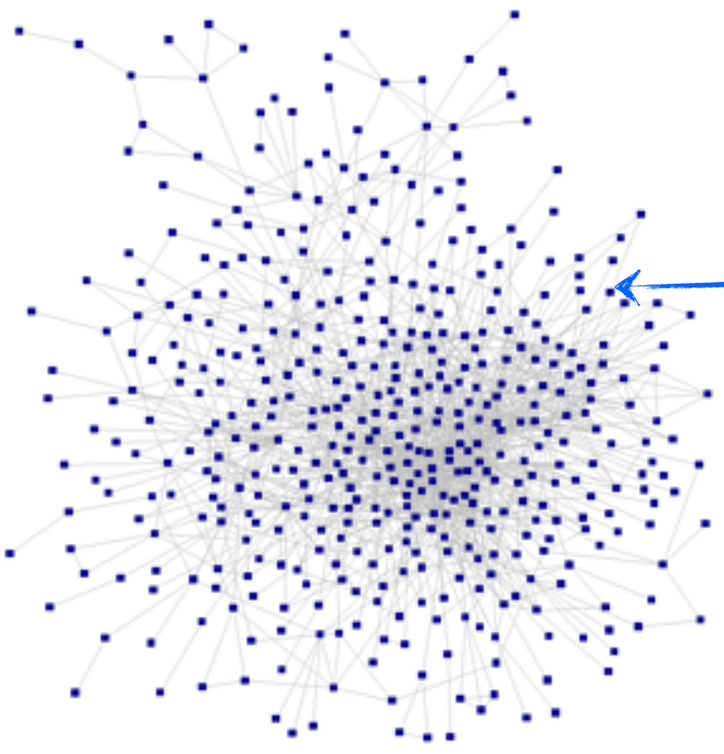


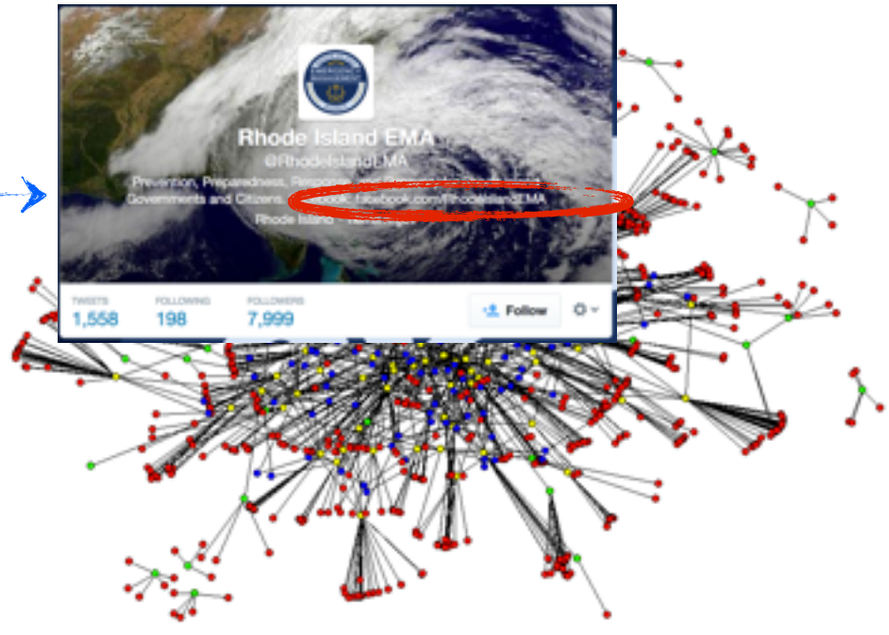




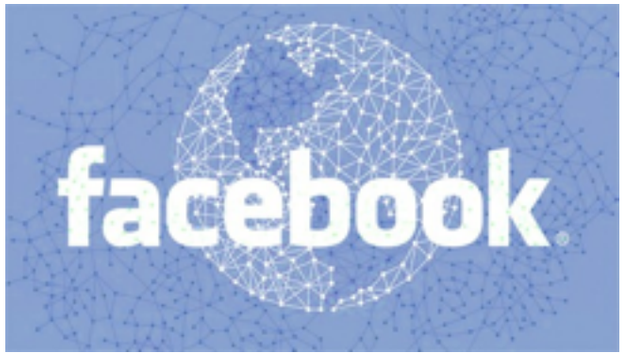
Real world motivations

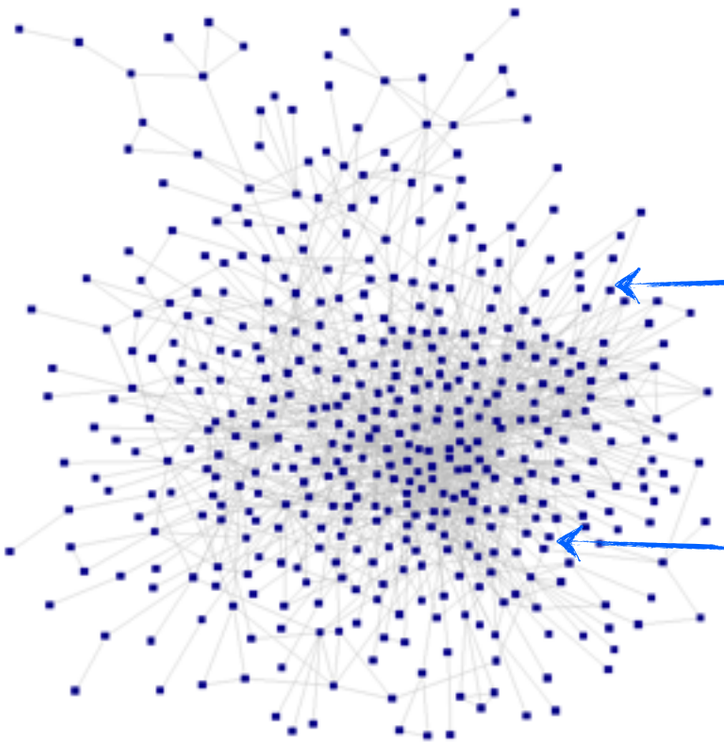






Real world motivations



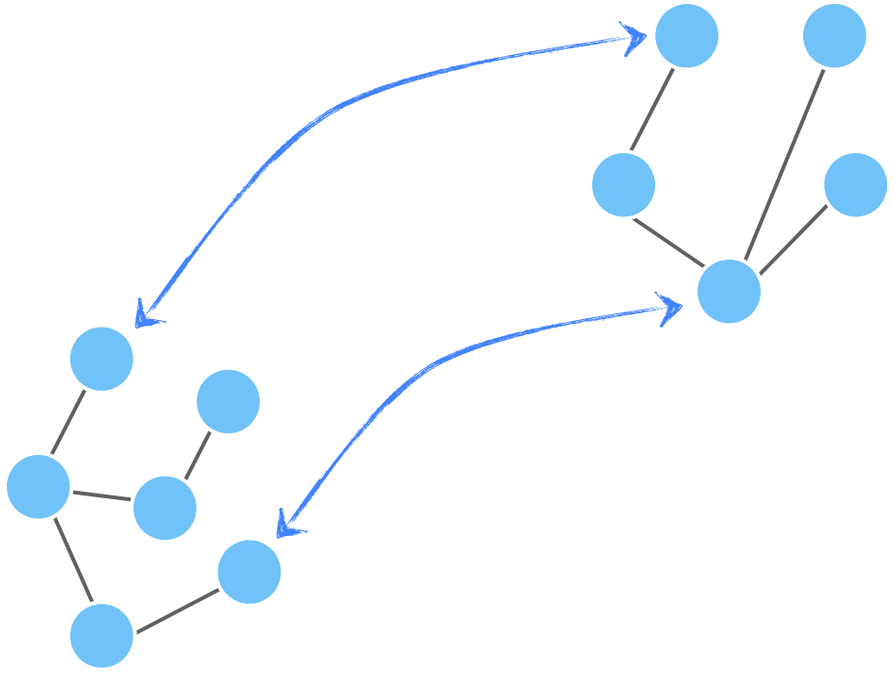


Graph reconciliation problem

Given two networks, identify as many users as possible across them.

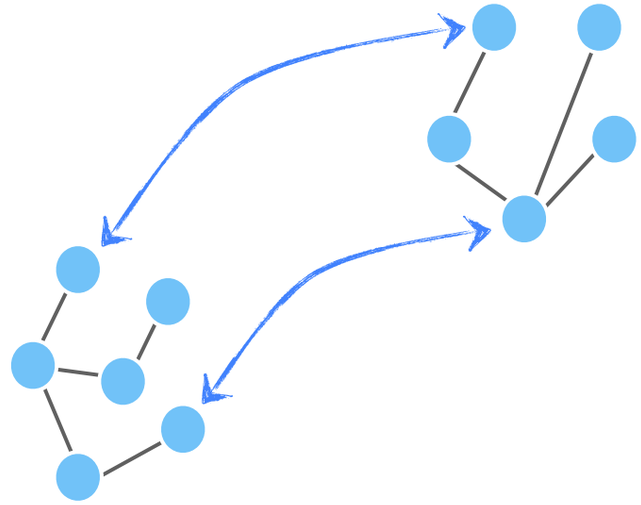
Applications:

- social networks
- ontology reconciliation



Previous work

Problem of reconciliation introduced by Novak et al.



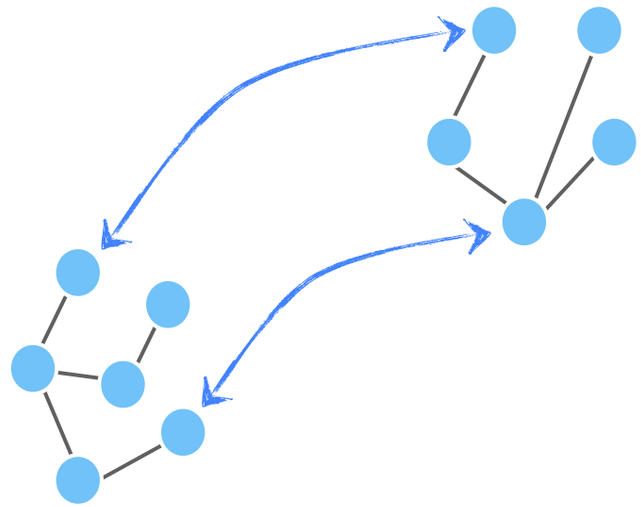
Previous work

Problem of reconciliation introduced by Novak et al.

Two main approaches:

- ML on user profile features

  (name, location, image)



Previous work

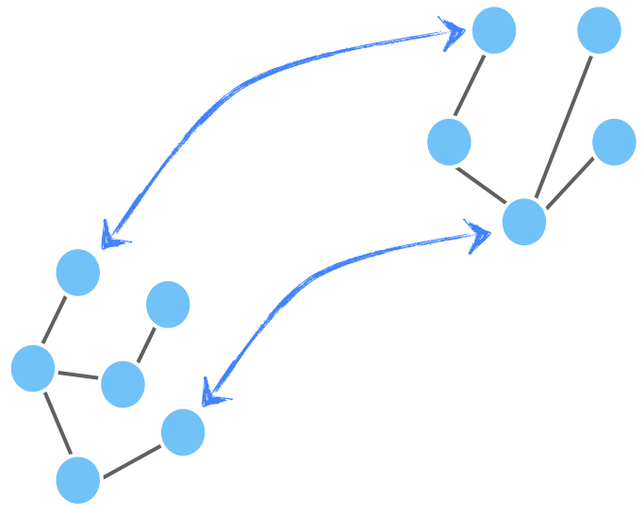
Problem of reconciliation introduced by Novak et al.

Two main approaches:

- ML on user profile features

  (name, location, image)

- ML on neighborhood topology



Previous work

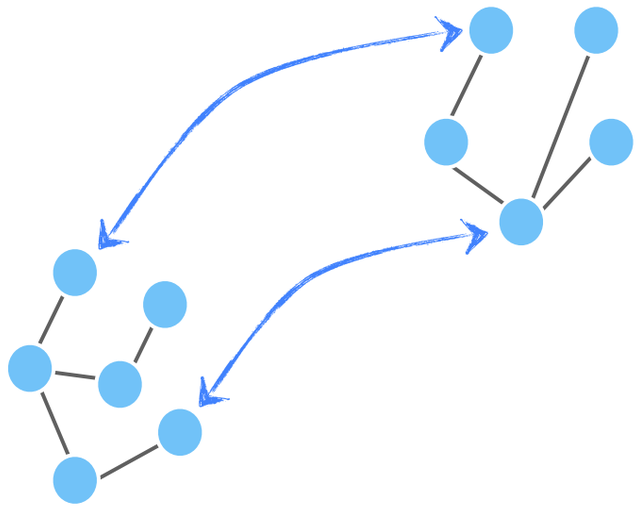
Problem of reconciliation introduced by Novak et al.

Two main approaches:

- ML on user profile features

  (name, location, image)

- ML on neighborhood topology



Limitations:

Previous work

Very rich literature in de-anonymization

Two relevant works:

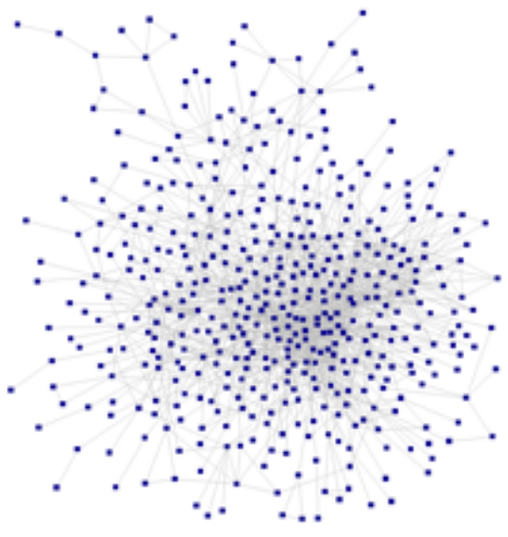
- Backstrom et al. propose an active and passive attack

Previous work

Very rich literature in de-anonymization

Two relevant works:

- Backstrom et al. propose an active and passive attack

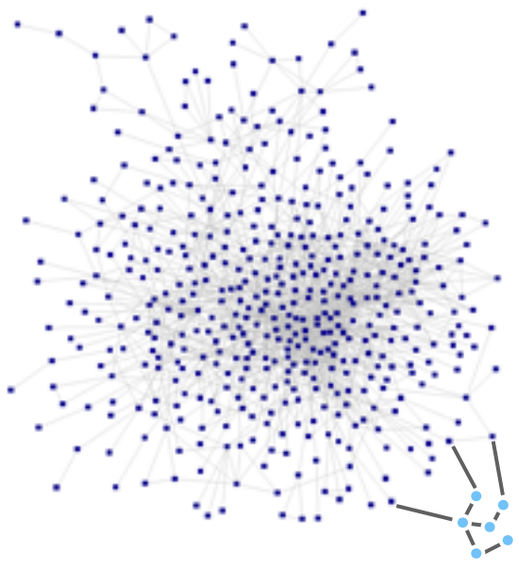


Previous work

Very rich literature in de-anonymization

Two relevant works:

- Backstrom et al. propose an active and passive attack

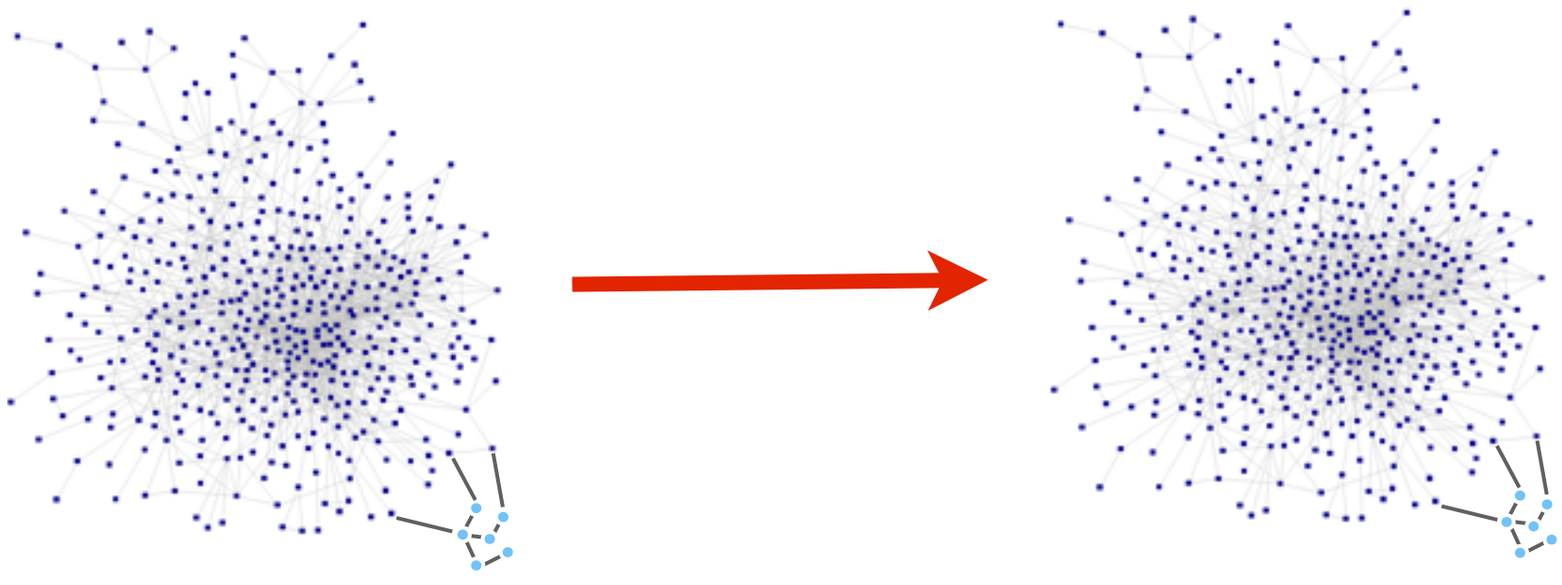


Previous work

Very rich literature in de-anonymization

Two relevant works:

- Backstrom et al. propose an active and passive attack



Previous work

Very rich literature in de-anonymization

Two relevant works:

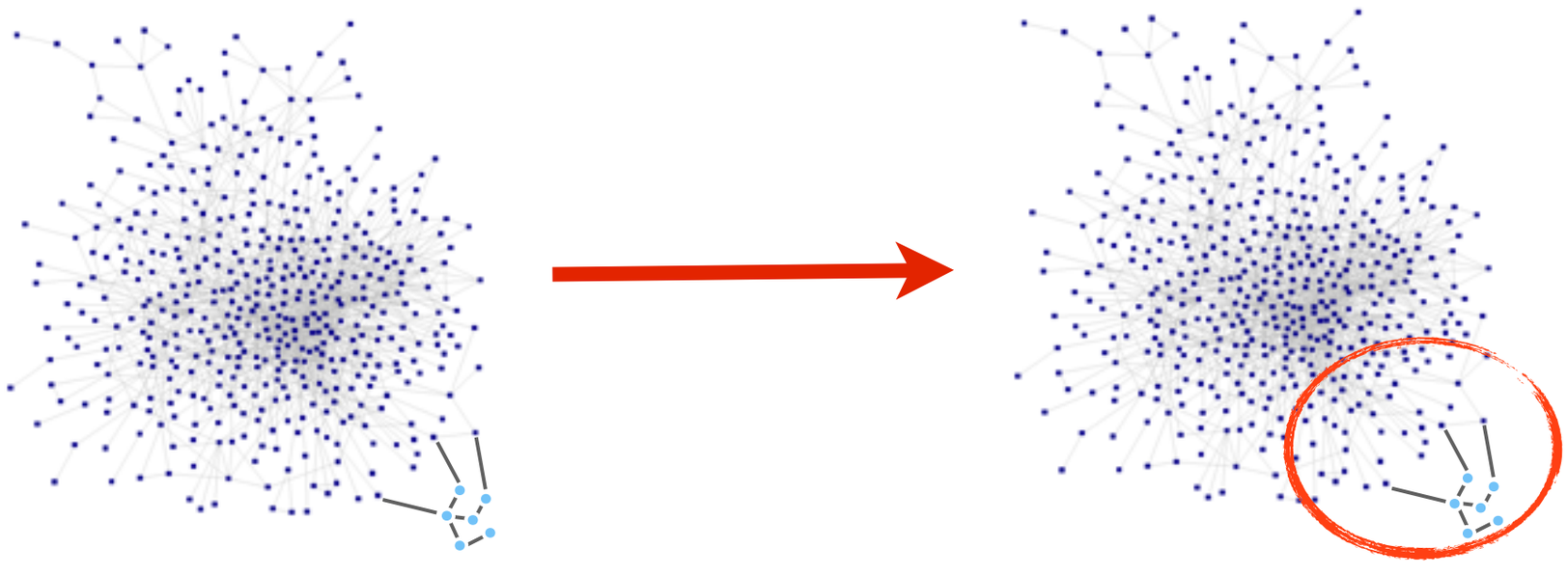
- Backstrom et al. propose an active and passive attack

Previous work

Very rich literature in de-anonymization

Two relevant works:

- Backstrom et al. propose an active and passive attack



Previous work

Very rich literature in de-anonymization

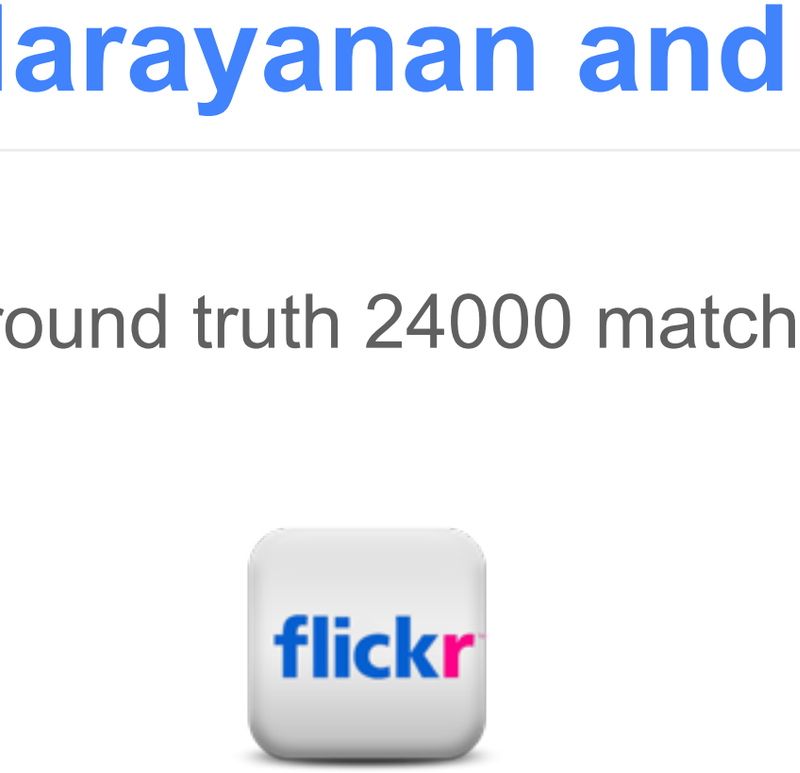
Two relevant works:

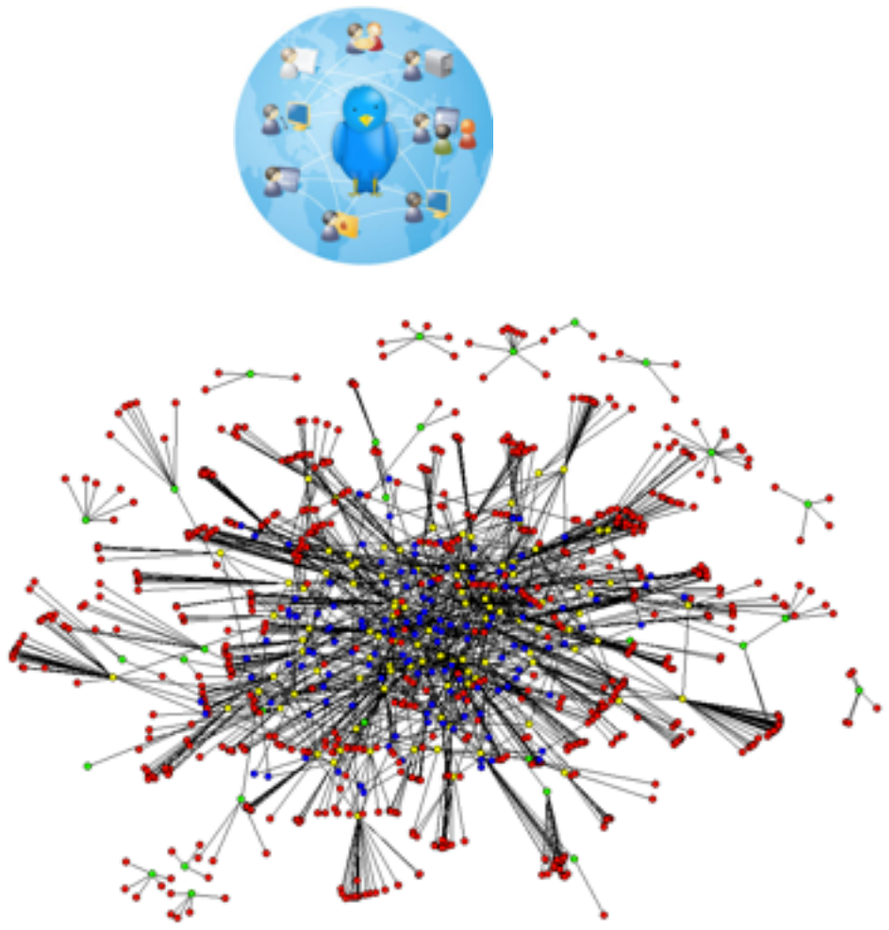
- Backstrom et al. propose an active and passive attack

- Narayanan and Shmatikov successful de-anonymization attack

Narayanan and Shmatikov experiment

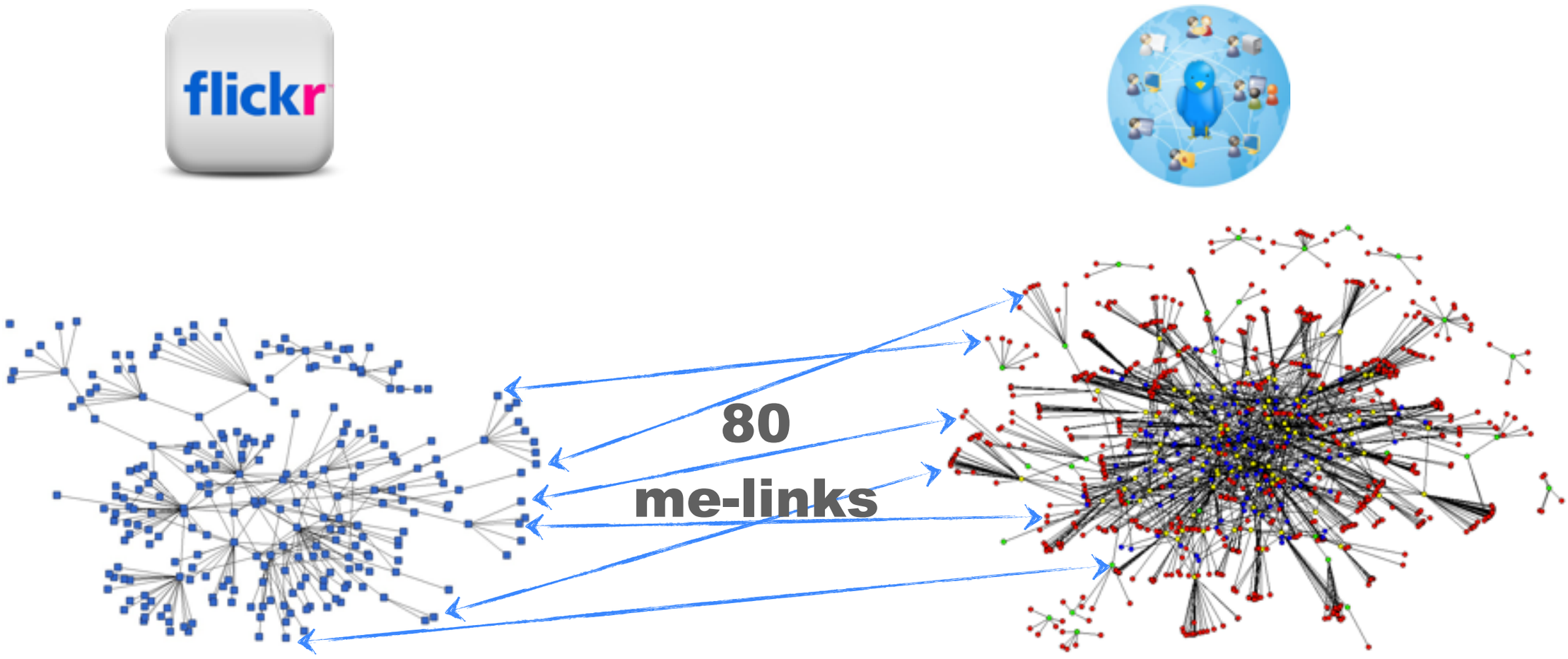
Ground truth 24000 matching across the two social networks





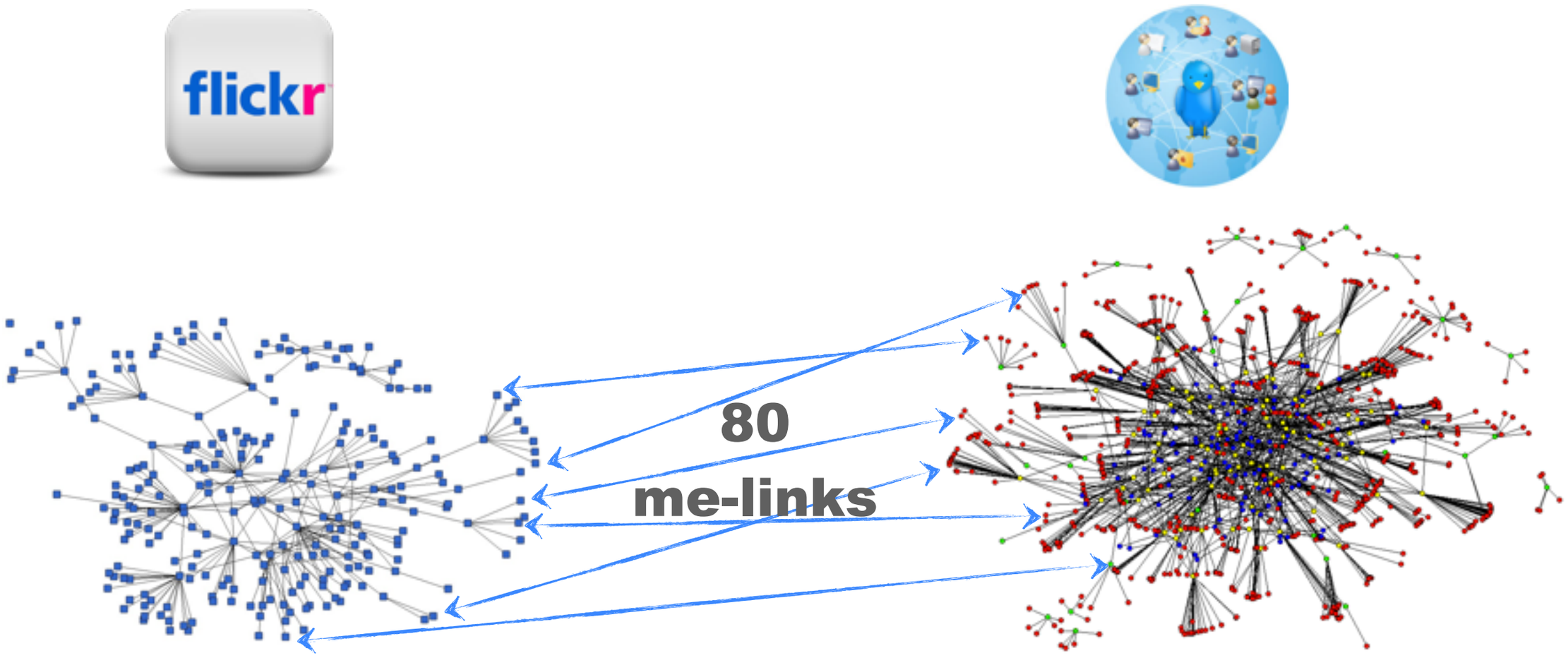
Narayanan and Shmatikov experiment

Ground truth 24000 matching across the two social networks



Narayanan and Shmatikov experiment

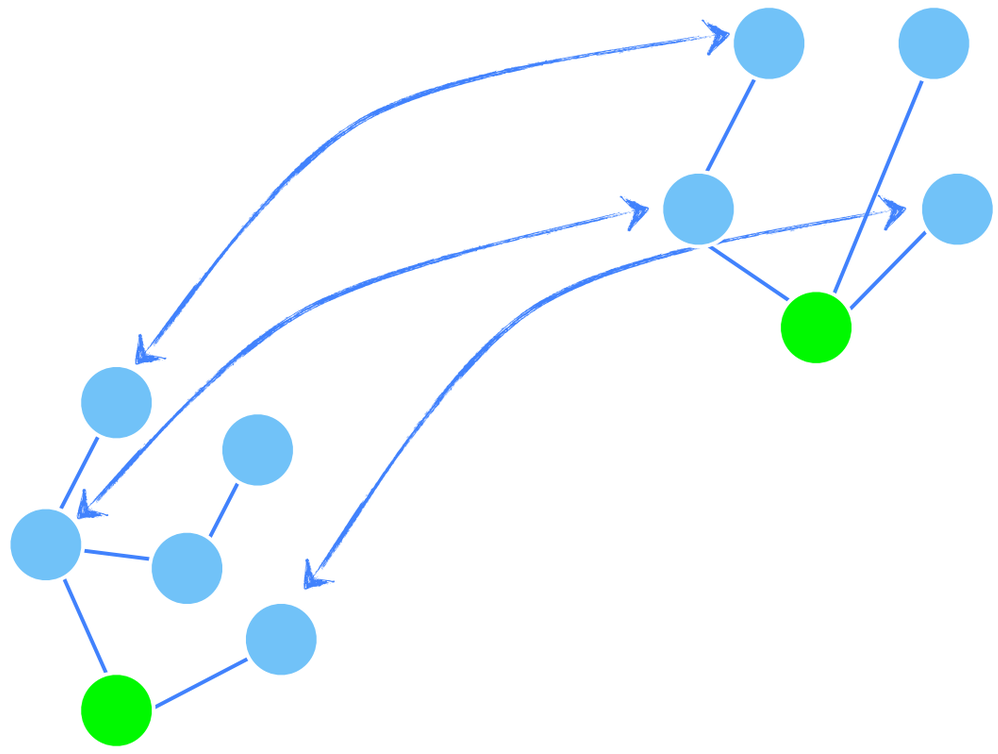
Ground truth 24000 matching across the two social networks



They could re-identify 30.8% of the mappings.

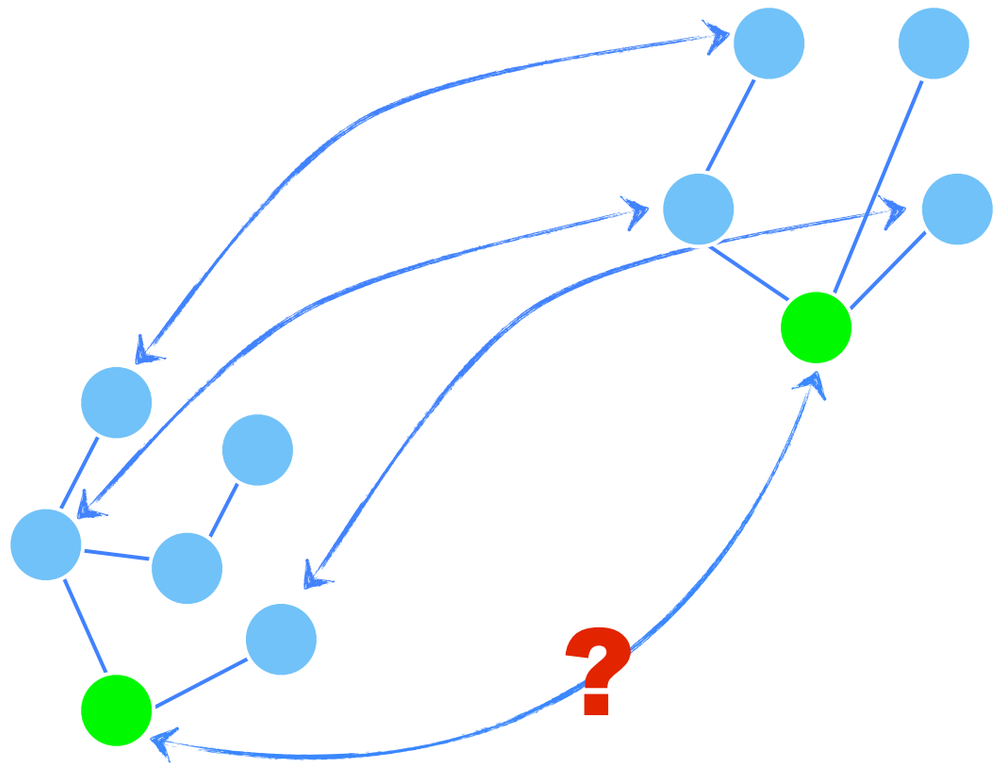
Narayanan and Shmatikov experiment

Algorithm:



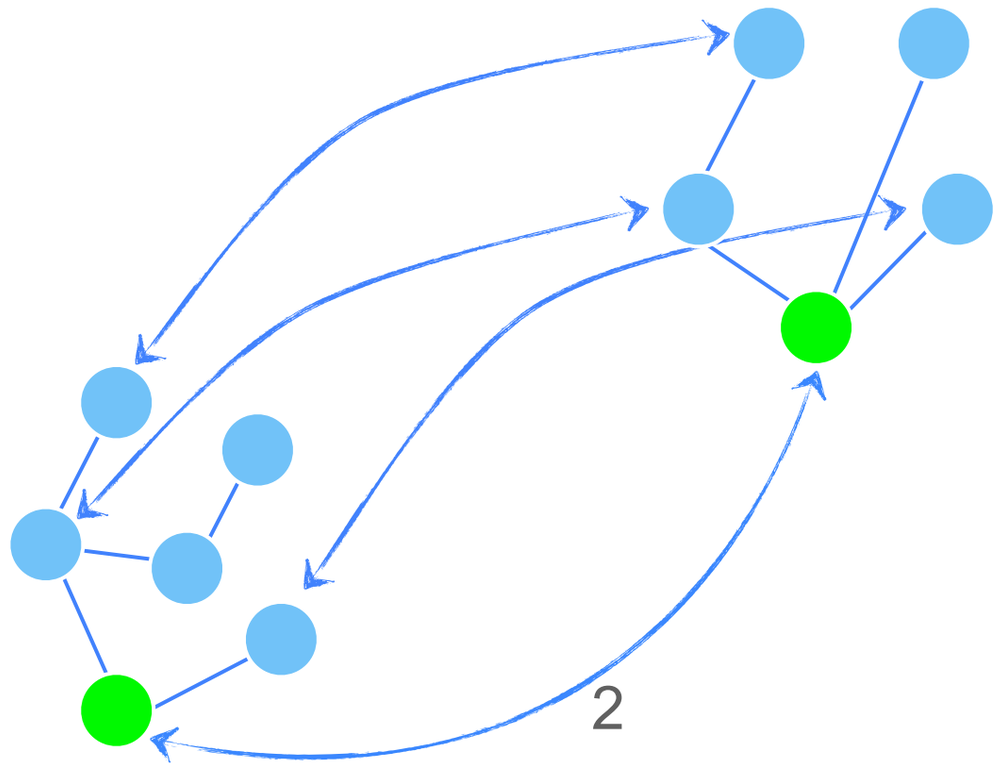
Narayanan and Shmatikov experiment

Algorithm:



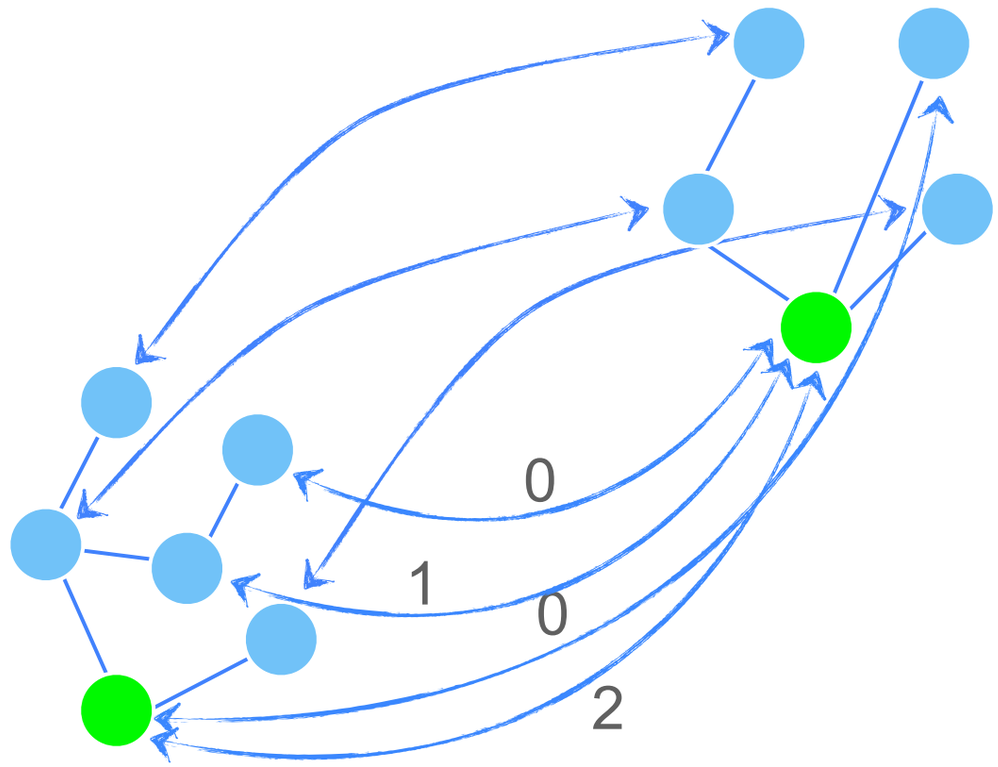
Narayanan and Shmatikov experiment

Algorithm:



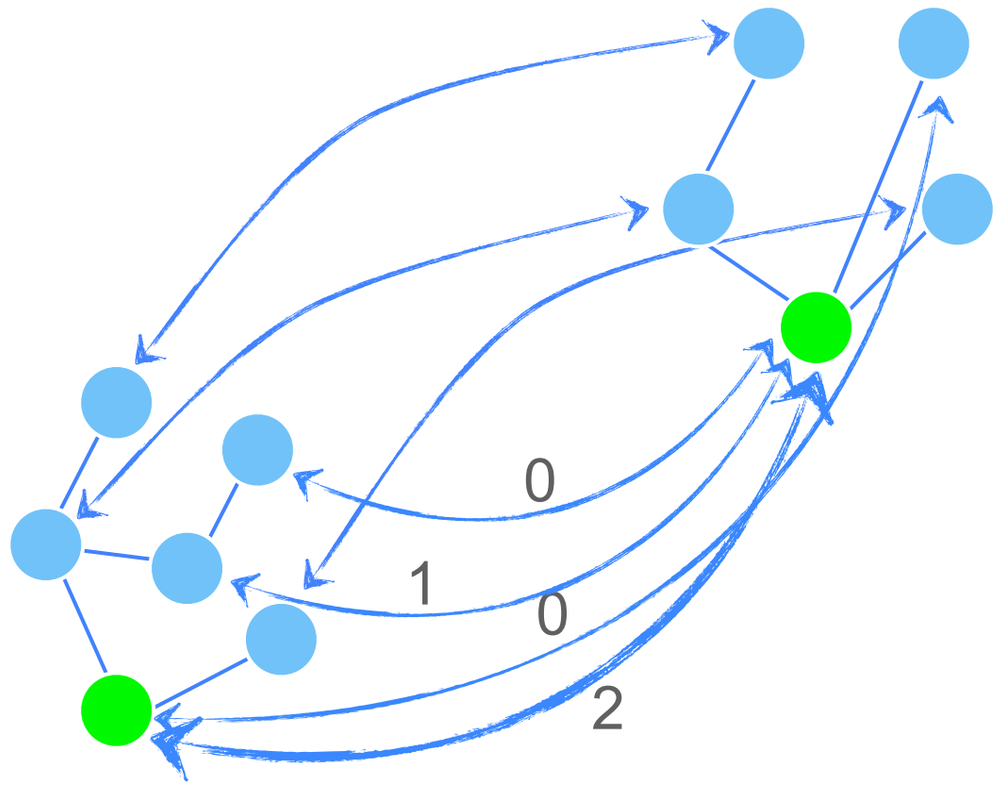
Narayanan and Shmatikov experiment

Algorithm:



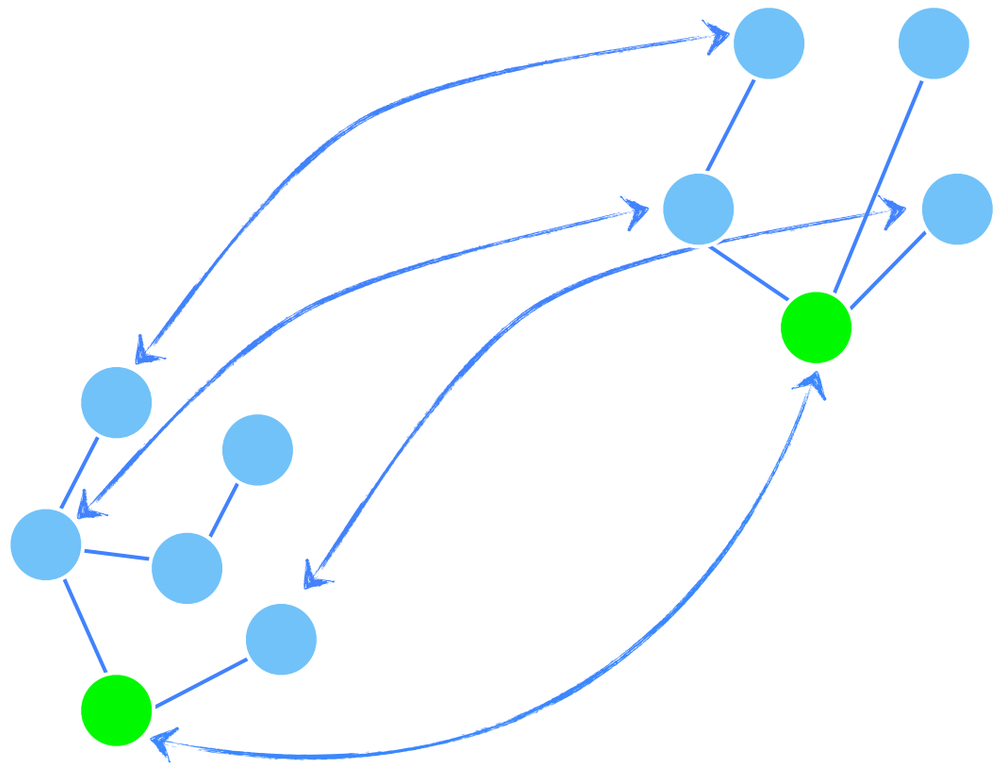
Narayanan and Shmatikov experiment

Algorithm:



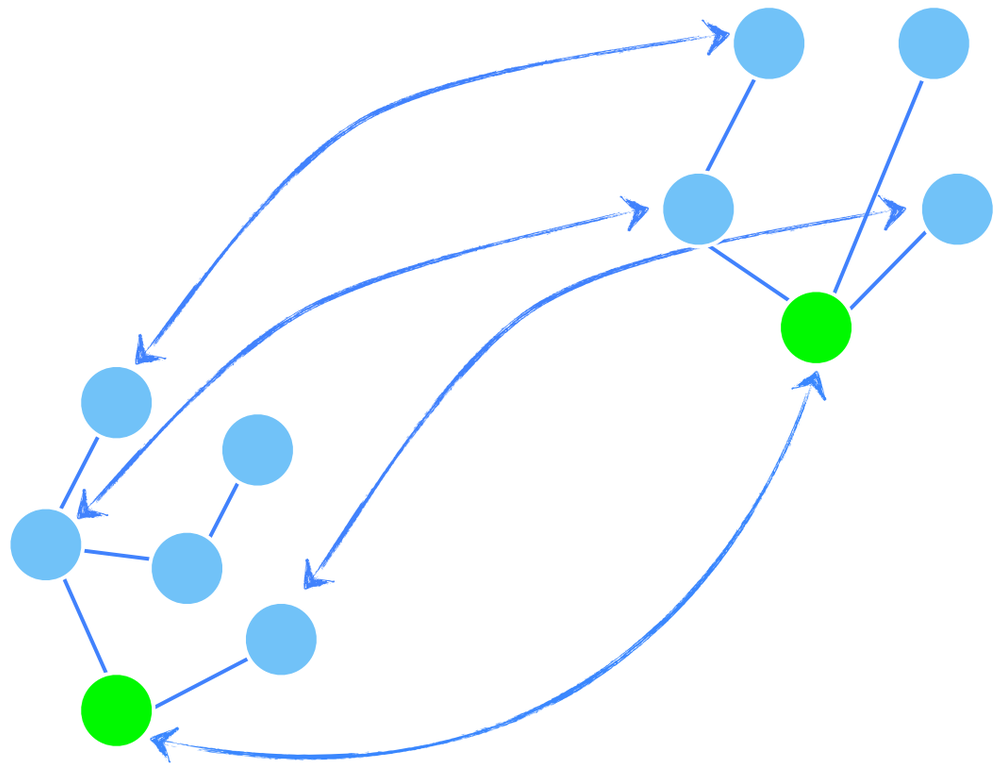
Narayanan and Shmatikov experiment

Algorithm:



Narayanan and Shmatikov experiment

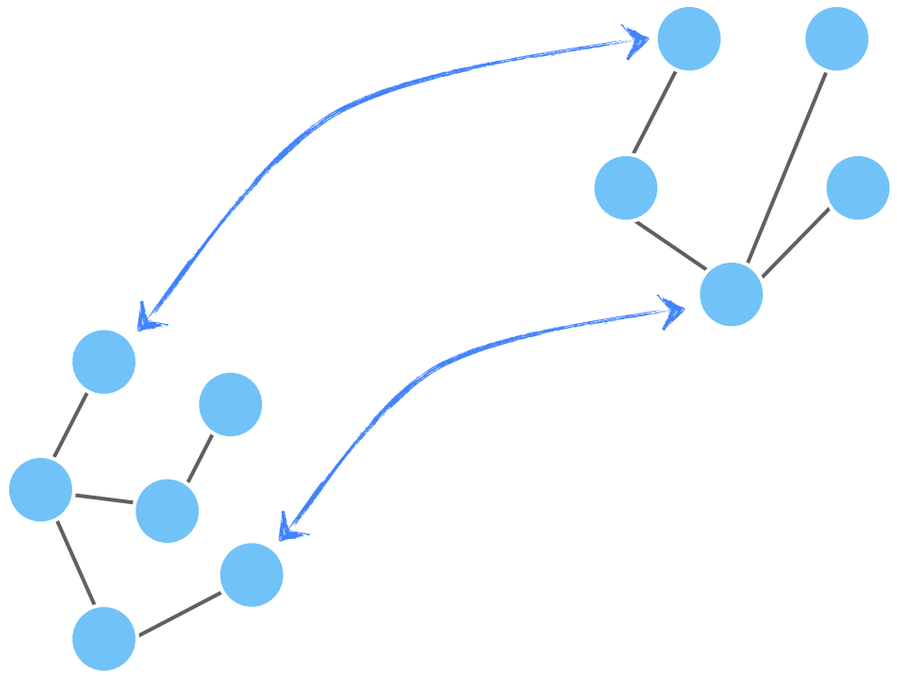
Algorithm:



Why?
Is it necessary to have high degree me-links?

Abstraction

Input: two graphs and a set of trusted matching



We want to maximize the number of final matches.

Is the problem tractable?

Problem is similar to graph isomorphism

Is the problem tractable?

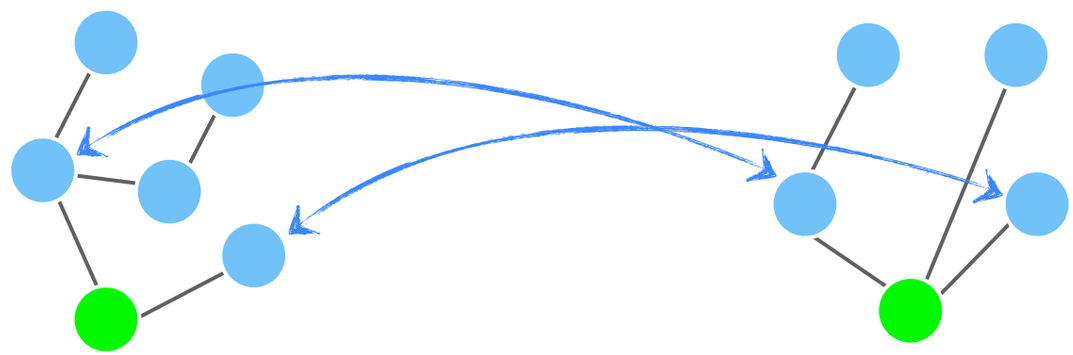
Problem is similar to graph isomorphism

Problem seems even harder because we want to detect similar structure

Is the problem tractable?

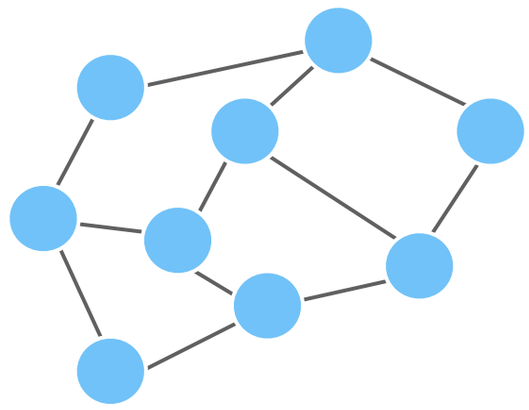
Problem is similar to graph isomorphism

Problem seems even harder because we want to detect similar structure



Abstraction

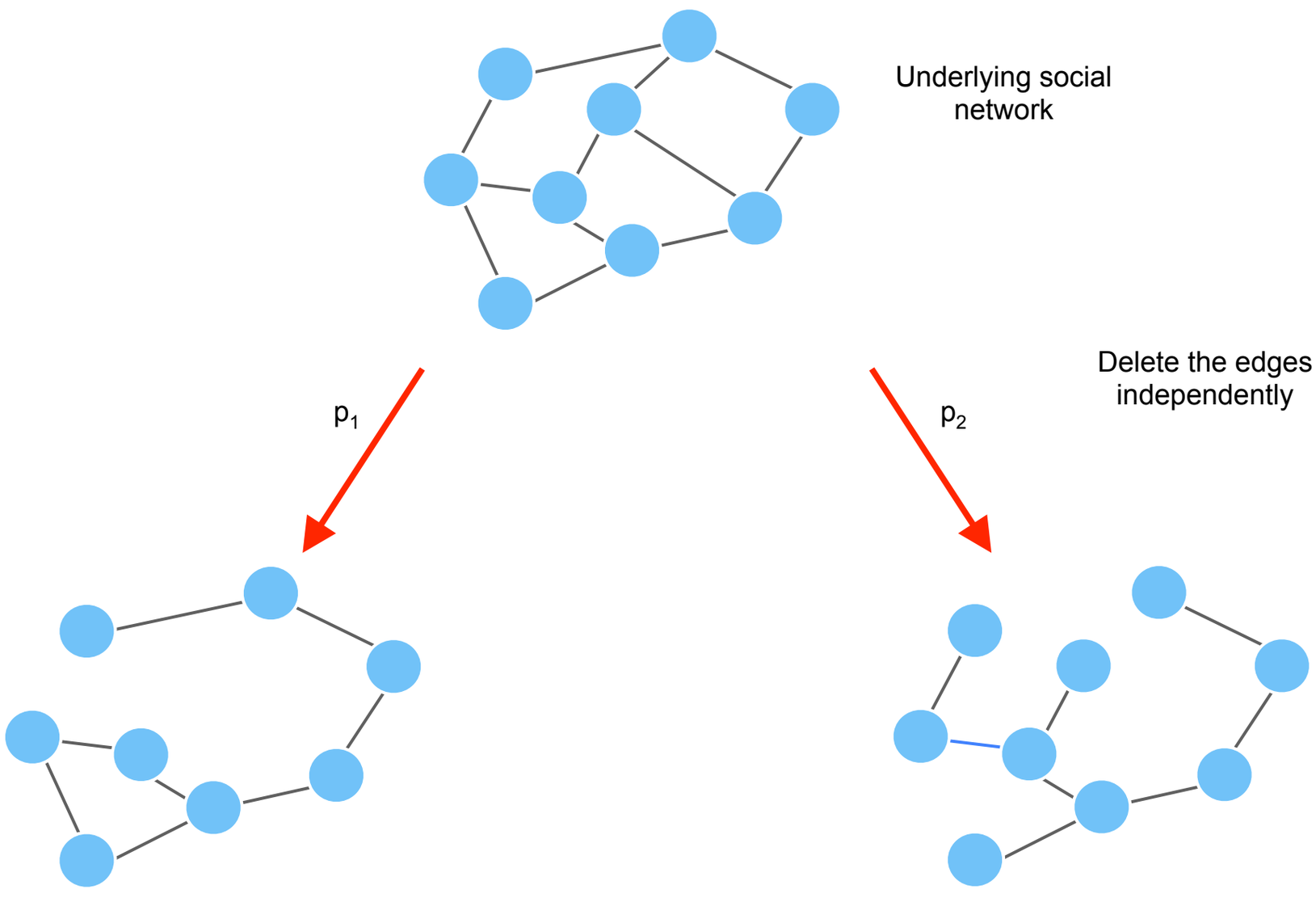
Formalization of the problem:



Underlying social network

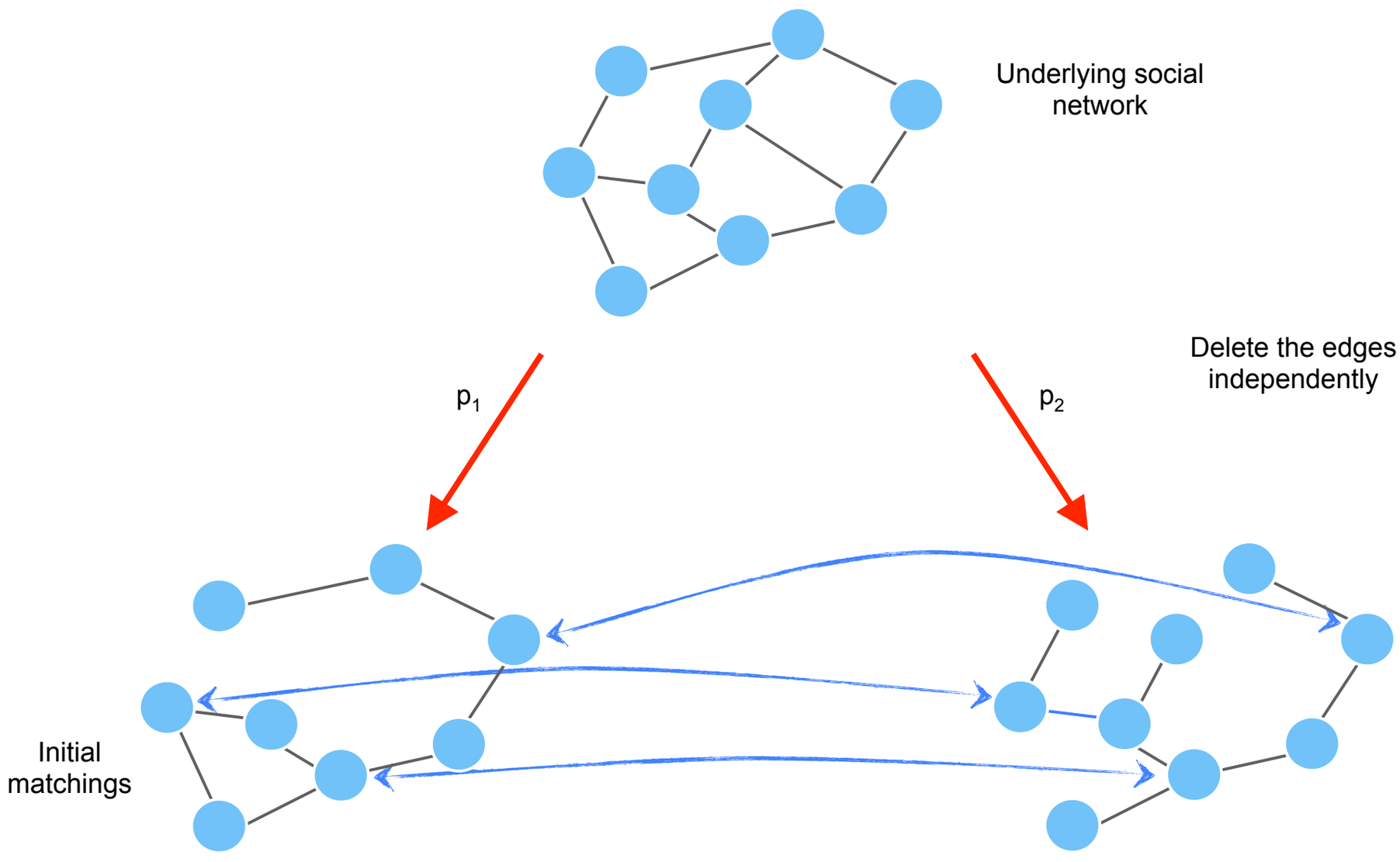
Abstraction

Formalization of the problem:



Abstraction

Formalization of the problem:



Questions

- Having a constant fraction of me-links, can we reconcile the entire network?

- If we have k me-links which fraction of networks can we reconcile?

Underlying social network

Without additional assumption on the underling network problem seems still very hard

Underlying social network

Without additional assumption on the underling network problem seems still very hard

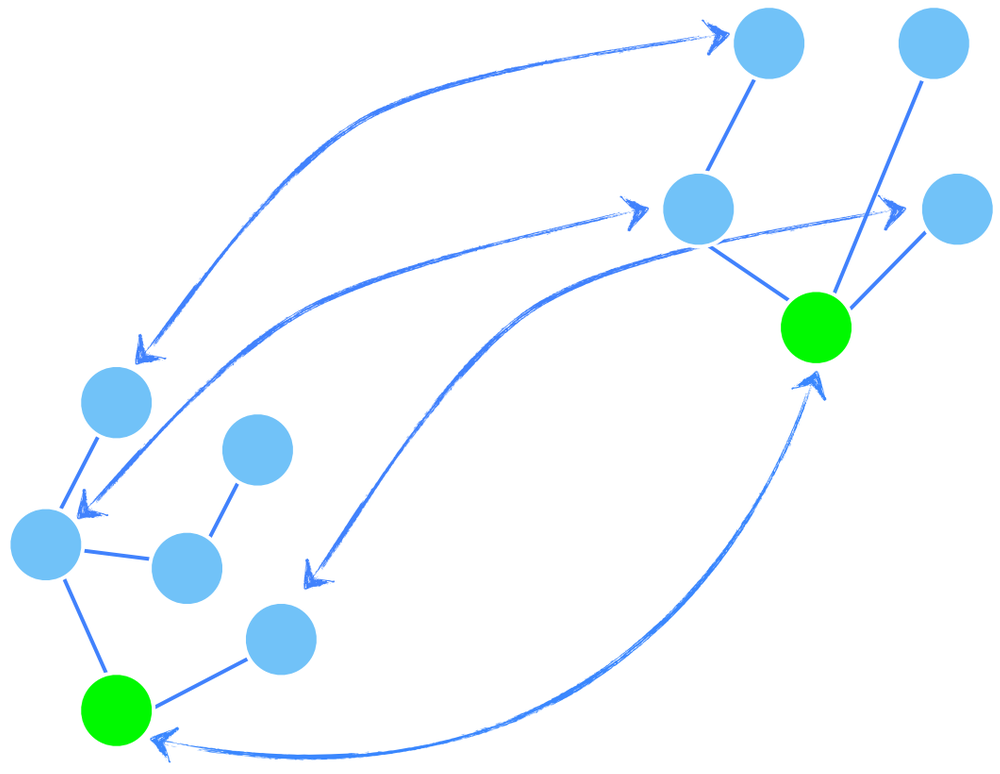
We study two different models for social networks:

- G(n,p)

- Preferential attachment

Our algorithm

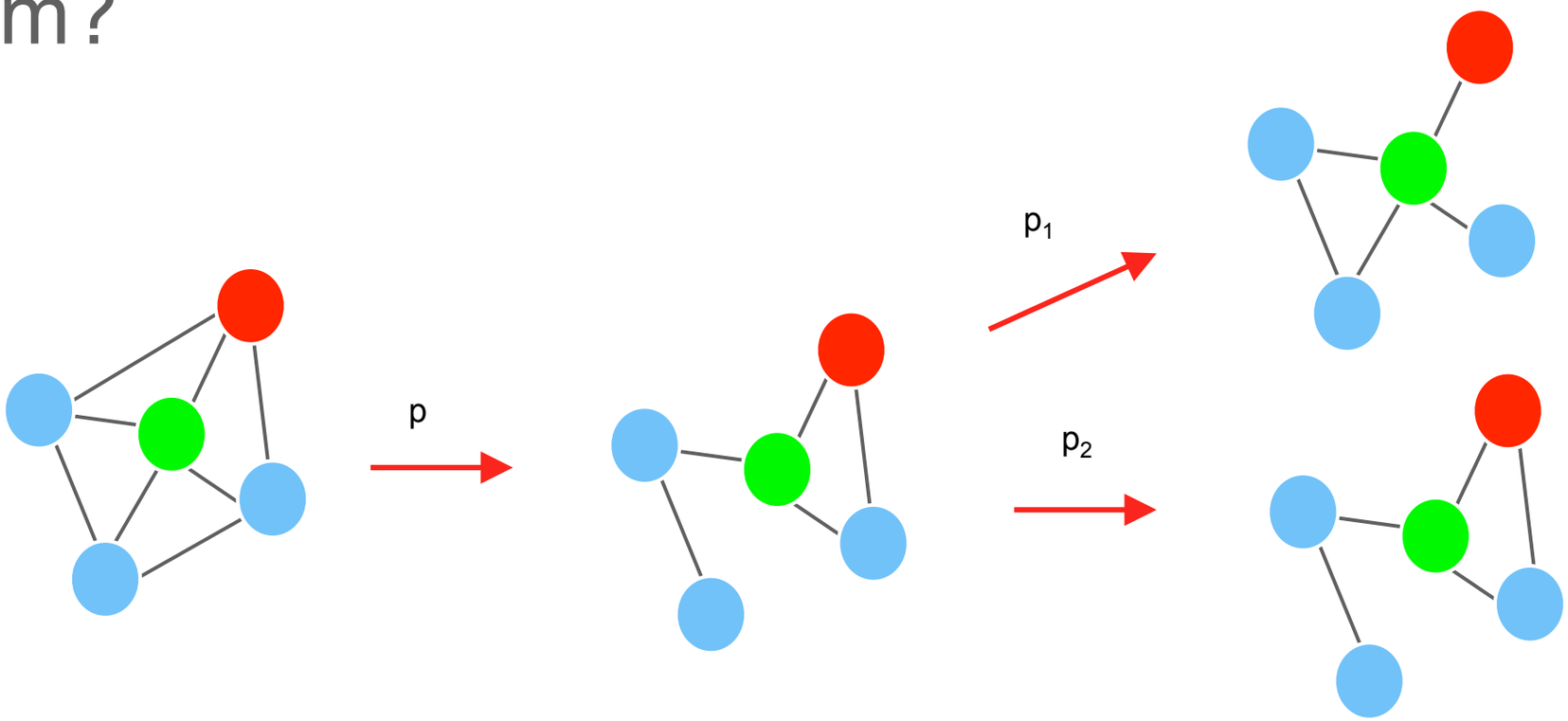
Algorithm:



Narayanan Shmatikov + degree bucketing + acceptance threshold

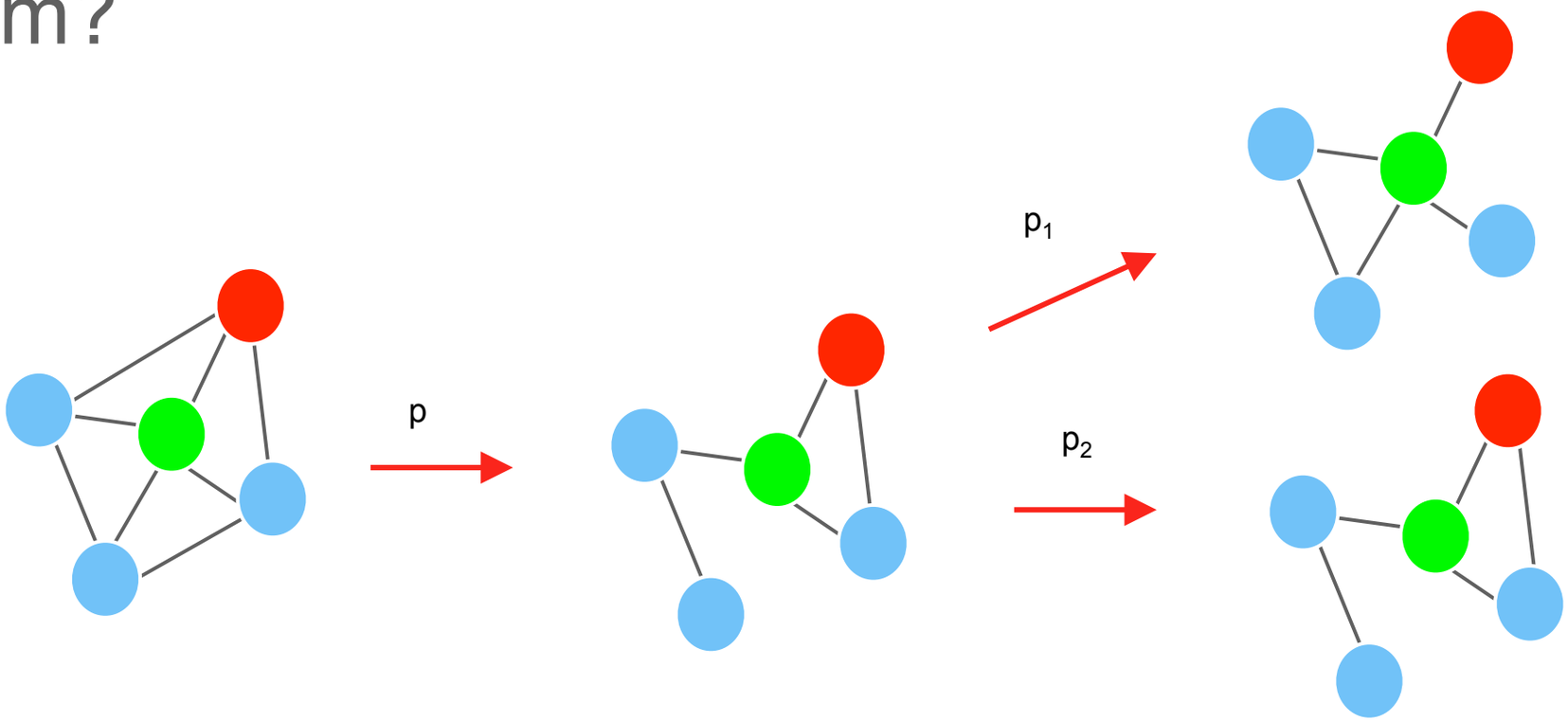
G(n,p)

Does the technique works if the underlying graph is random?



G(n,p)

Does the technique works if the underlying graph is random?



$$E[N_{G_1}(\bullet) \cap N_{G_2}(\bullet)] = (n-1)pp_1p_2$$

$$E[N_{G_1}(\bullet) \cap N_{G_2}(\bullet)] = (n-2)p^2p_1p_2$$

Concentration

We assume $\frac{c \log n}{n} \leq p \leq \frac{1}{6}, l, p_1, p_2 \in O(1)$

Two cases:

- $npp_1p_2l \geq 24 \log n$, Chernoff bound is enough

- $npp_1p_2l \leq 24 \log n$, we never make error

  $x = (n-2)p^2p_1p_2$

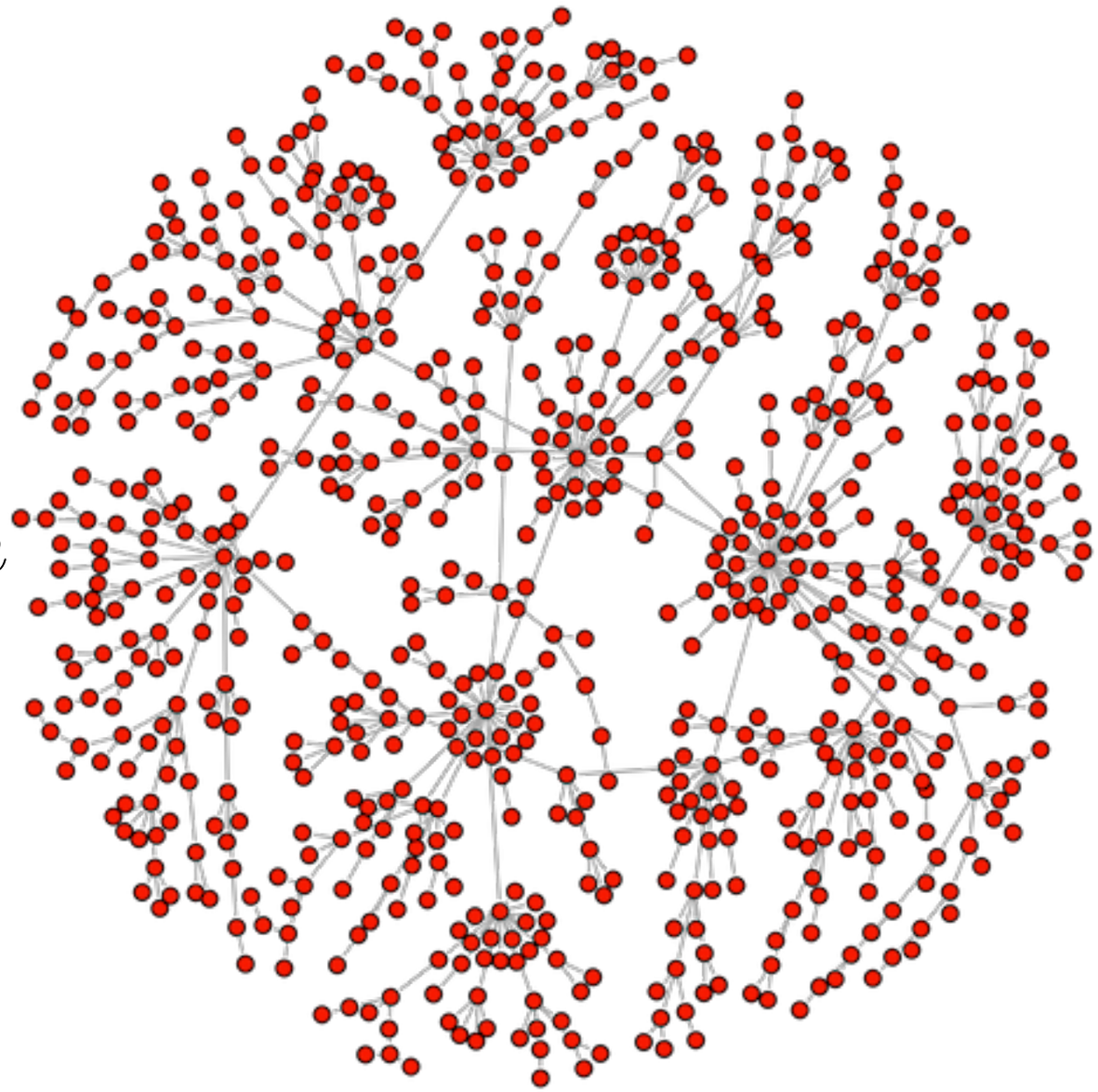
$$P = \left[\sum_{i=1}^{n} B_i \leq 2\right] = (1-x)^n + nx(1-x)^{n-1} + \binom{n}{2}x^2(1-x)^{n-2} = 1 - n^3x^3 - o(n^3x^3)$$

More realistic model

Preferential attachment:

- G_1^m is a single node with m self-loops

- G_n^m adding a node to G_{n-1}^m and m edges with probability proportional to the current degrees



Preferential attachment

A bit harder

- Several nodes of constant degree, we need to have a cascade

- Objective is reconcile a constant fraction of the network

Sketch of the proof

- For high degree node we can use concentration results.

Sketch of the proof

- For high degree node we can use concentration results.

- Different nodes of intermediate degree do not share many neighbors.

Sketch of the proof

- For high degree node we can use concentration results.

- Different nodes of intermediate degree do not share many neighbors.

- High degree nodes help to detect intermediate degree nodes that in turn help to detect small degree nodes.

PA structural lemmas

- *High degree nodes are early birds.*
  Nodes inserted after time ϕn , for constant ϕ , have degree in $o(\log^2 n)$

PA structural lemmas

- *High degree nodes are early birds.*
  Nodes inserted after time ϕn , for constant ϕ , have degree in $o(\log^2 n)$

- *The rich get richer.*
  For nodes of degree greater than $\log^2 n$ a constant fraction of their neighbors has been inserted after time ϵn, for constant ϵ

PA structural lemmas

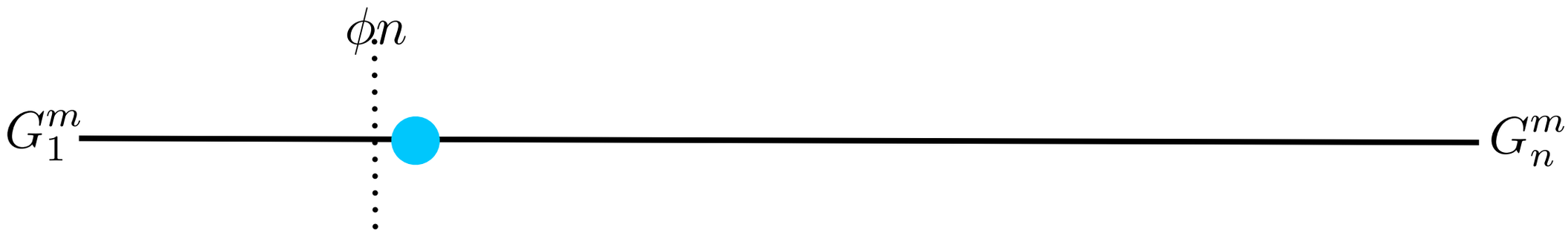
- *High degree nodes are early birds.*
  Nodes inserted after time ϕn , for constant ϕ , have degree in $o(\log^2 n)$

- *The rich get richer.*
  For nodes of degree greater than $\log^2 n$ a constant fraction of their neighbors has been inserted after time ϵn, for constant ϵ

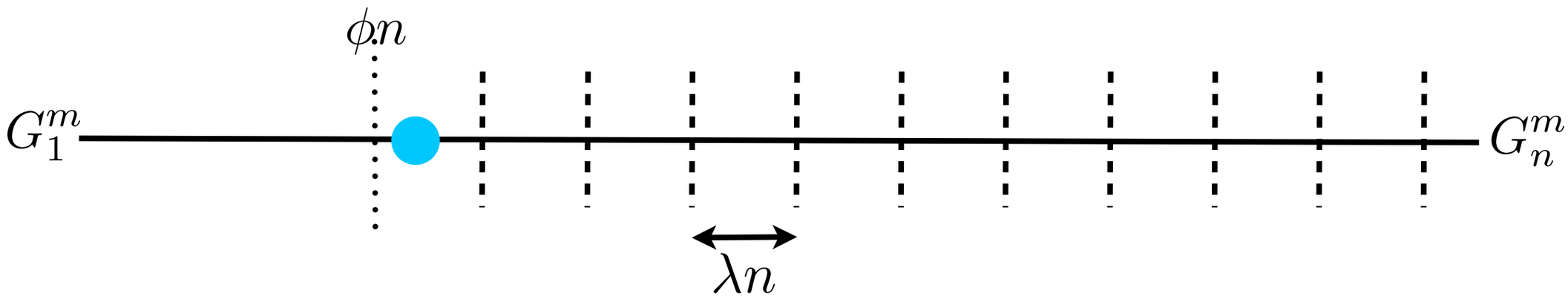
- *First-mover advantage.*
  All nodes inserted before time $n^{0.3}$, have degree at least $\log^3 n$

High degree nodes are early birds

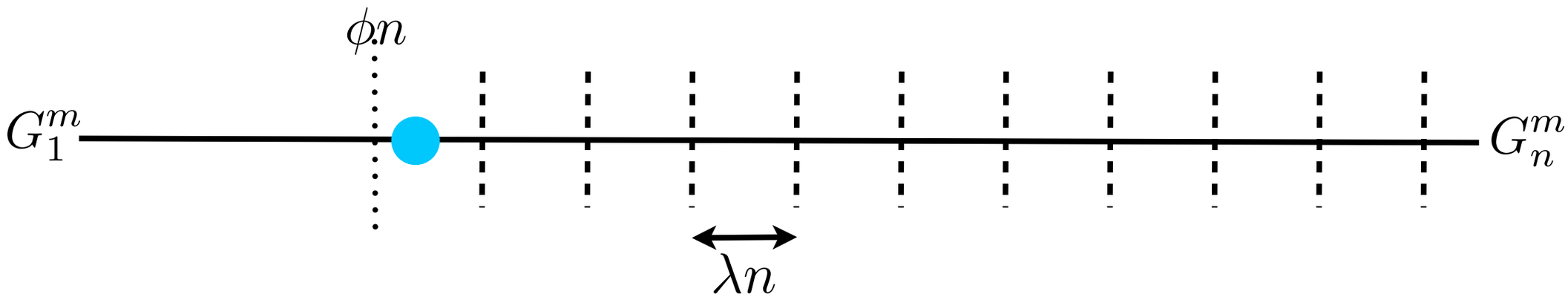
High degree nodes are early birds



High degree nodes are early birds



High degree nodes are early birds

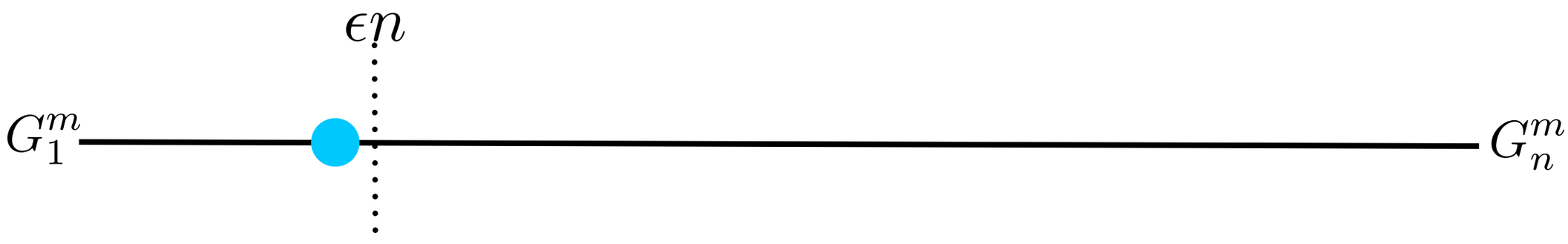


Let d_i be the degree at the beginning of a phase.

The probability that a node increase its degree is dominated by the probability of an head in a coin toss for a biased coin that gives head with probability $\frac{3d_i}{\phi n}$

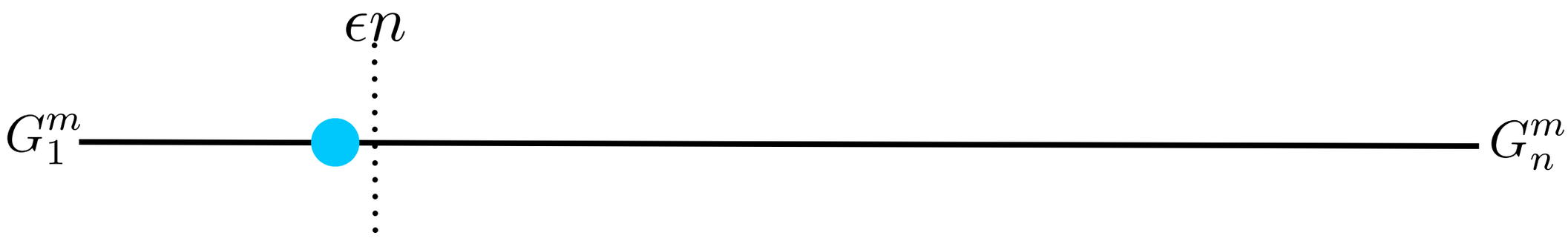
The rich get richer

If at time ϵn, the node has degree less than $\frac{1}{2}d$ we are done



The rich get richer

If at time ϵn, the node has degree less than $\frac{1}{2}d$ we are done



The probability that the node increases its degree is dominated by the probability of an head in a coin toss for a biased coin that gives head with probability $\frac{d}{2nm}$

First-mover advantage

From Cooper and Frieze result on the cover time of PA graphs,

$$D_k = d_{nm}(v_1) + d_{nm}(v_2) + \cdots + d_{nm}(v_k)$$

$$Pr\left(|D_k - 2\sqrt{2kn}| \geq 3\sqrt{mn \log mn}\right) \leq (mn)^{-2}$$

$$Pr(d_n(v_{k+1}) = d + 1 | D_k - 2k = s) \leq \frac{s+d}{2N - 2k - s - d}$$

Playing a bit with algebra we can get the final result.

Sketch of the proof

- **For high degree node we can use concentration results.**

- Different nodes of intermediate degree do not share many neighbors.

- High degree nodes help to detect intermediate degree nodes that in turn help to detect small degree nodes.

Matching high degree nodes

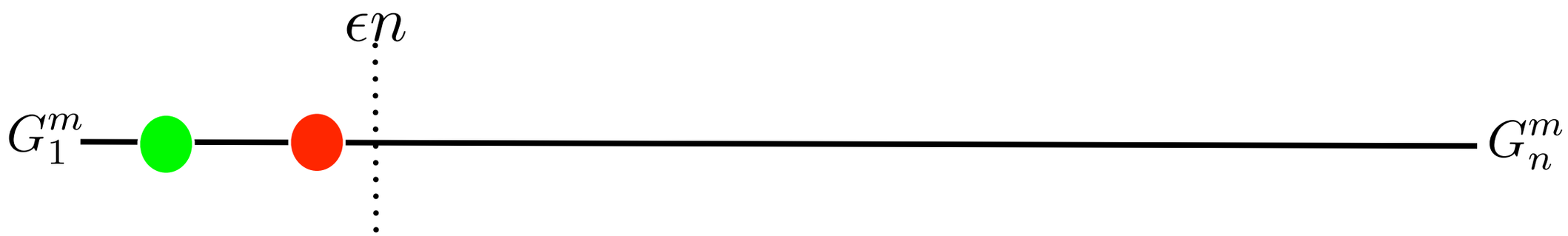
$$E[N_{G_1}(\bullet) \cap N_{G_2}(\bullet)] = d(v)p_1p_2l$$

By Chernoff $N_{G_1}(\bullet) \cap N_{G_2}(\bullet) \geq \frac{7}{8}d(v)p_1p_2l$ w.h.p.

Matching high degree nodes

$$E[N_{G_1}(\bullet) \cap N_{G_2}(\bullet)] = d(v)p_1p_2l$$

By Chernoff $N_{G_1}(\bullet) \cap N_{G_2}(\bullet) \geq \frac{7}{8}d(v)p_1p_2l$ w.h.p.

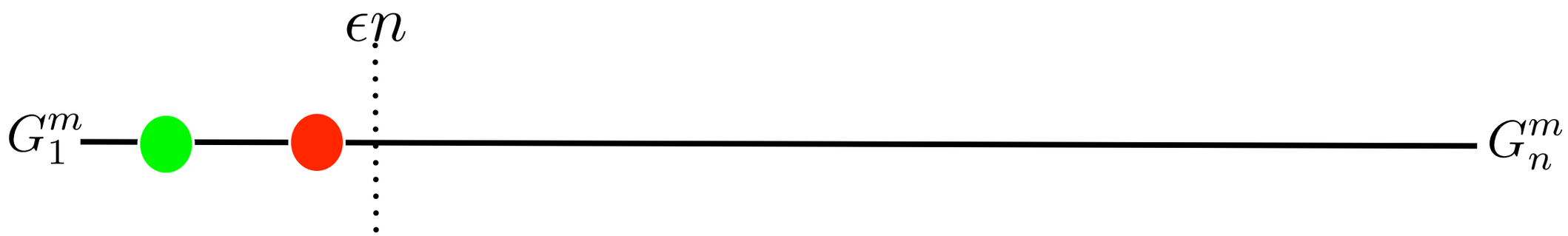


$$N_{G_1}(\bullet) \cap N_{G_2}(\bullet) \leq \left(\frac{2}{3} + \epsilon\right) d(v)p_1p_2l$$

Matching high degree nodes

$$E[N_{G_1}(\bullet) \cap N_{G_2}(\bullet)] = d(v)p_1p_2l$$

By Chernoff $N_{G_1}(\bullet) \cap N_{G_2}(\bullet) \geq \frac{7}{8}d(v)p_1p_2l$ w.h.p.



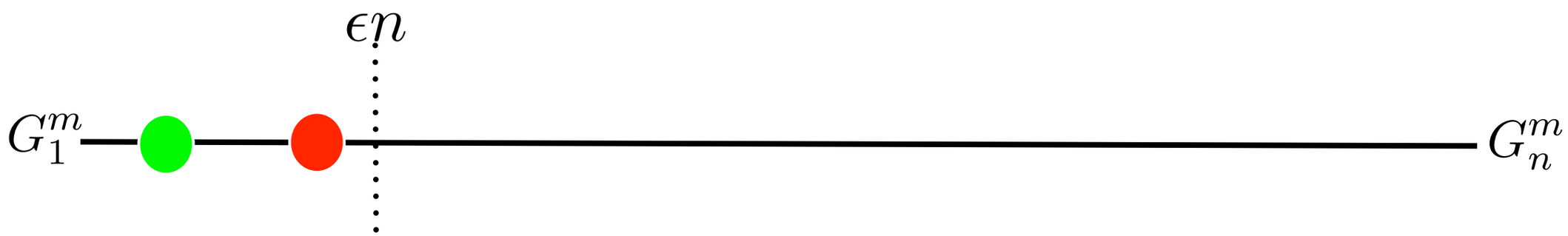
$$N_{G_1}(\bullet) \cap N_{G_2}(\bullet) \leq \left(\frac{2}{3} + \epsilon\right) d(v)p_1p_2l$$

● has degree at most $\tilde{O}(\sqrt{n})$ and so the probability of connecting to it is $o(1)$

Matching high degree nodes

$$E[N_{G_1}(\bullet) \cap N_{G_2}(\bullet)] = d(v)p_1p_2l$$

By Chernoff $N_{G_1}(\bullet) \cap N_{G_2}(\bullet) \geq \frac{7}{8}d(v)p_1p_2l$ w.h.p.



$$N_{G_1}(\bullet) \cap N_{G_2}(\bullet) \leq \left(\frac{2}{3} + \epsilon\right) d(v)p_1p_2l + o(d(v))$$

● has degree at most $\tilde{O}(\sqrt{n})$ and so the probability of connecting to it is $o(1)$

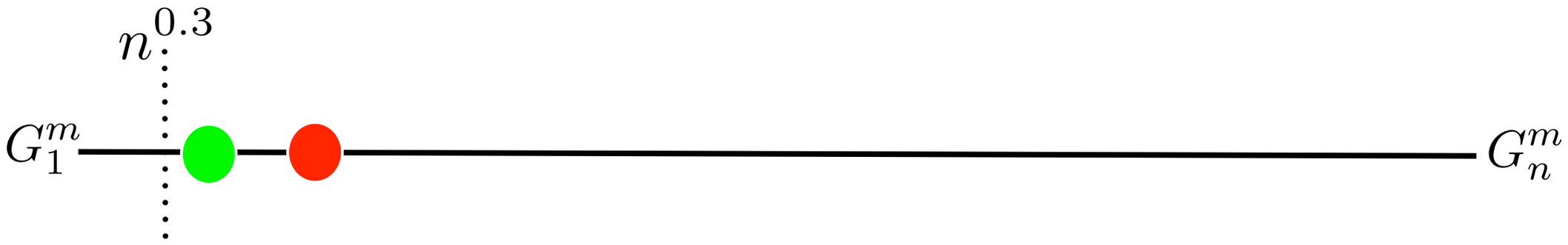
Sketch of the proof

- For high degree node we can use concentration results. 

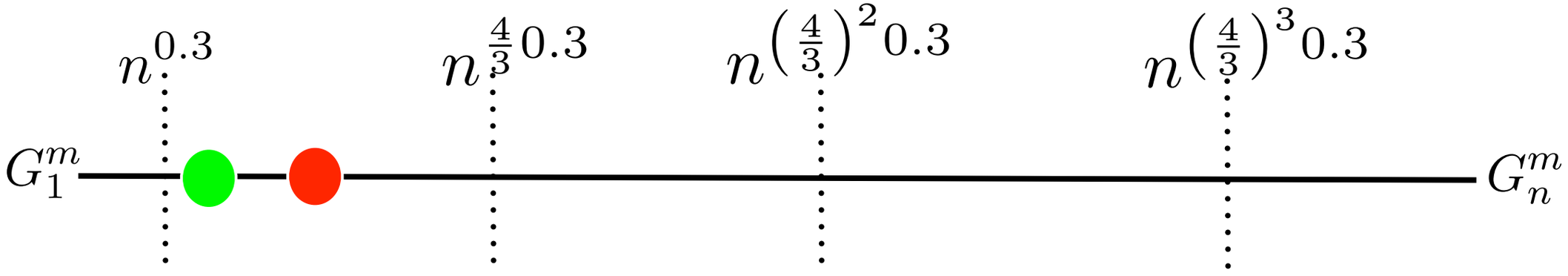
- **Different nodes of intermediate degree do not share many neighbors.**

- High degree nodes help to detect intermediate degree nodes that in turn help to detect small degree nodes.

Bound the mismatch score



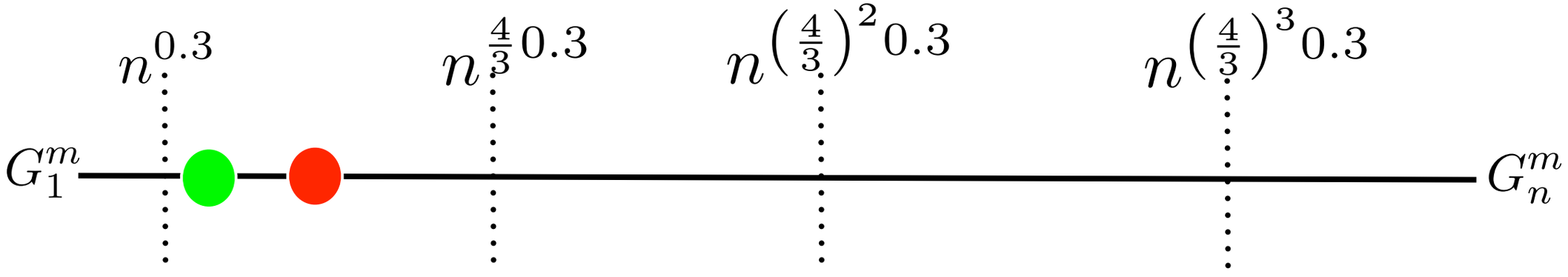
Bound the mismatch score



$$n^a = n^{0.3}, n^b = n^{\frac{4}{3}0.3}$$

$n^{\left(\frac{3}{2}-\epsilon\right)^3 0.3}$ $n^{\left(\frac{3}{2}-\epsilon\right)0.3}$ $n^{\left(\frac{3}{2}-\epsilon\right)^2 0.3}$

Bound the mismatch score



$$n^a = n^{0.3}, n^b = n^{\frac{4}{3}0.3}$$

The probability that 3 nodes coming between n^a and n^b point to ● and ●

$$(n^b)^2 \sum_{i=n^a}^{n^b} \sum_{j=n^a}^{n^b} \sum_{k=n^a}^{n^b} \left(\frac{\log^3 n}{(i-1)}\right)^2 \left(\frac{\log^3 n}{(j-1)}\right)^2 \left(\frac{\log^3 n}{(k-1)}\right)^2 \approx n^{2b-3a} \in o(1)$$

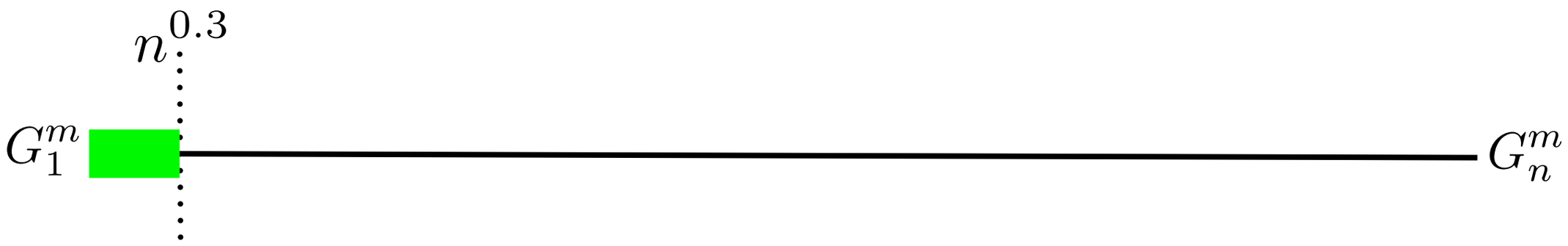
Sketch of the proof

- For high degree node we can use concentration results. 

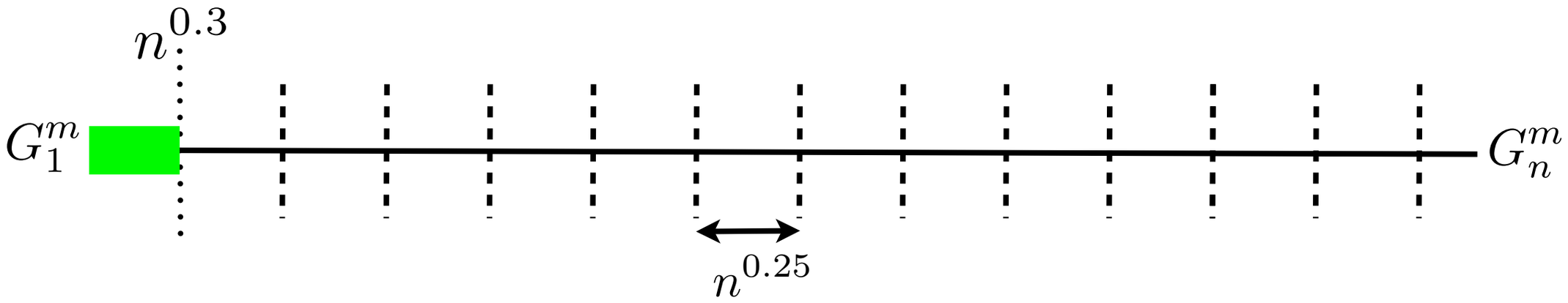
- Different nodes of intermediate degree do not share many neighbors. 

- **High degree nodes help to detect intermediate degree nodes that in turn help to detect small degree nodes.**

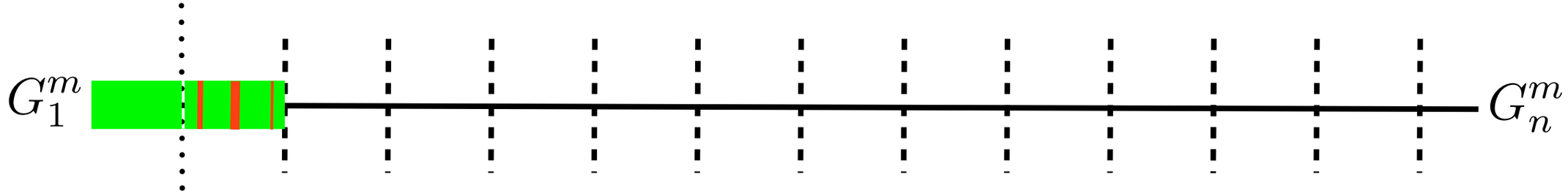
Cascade



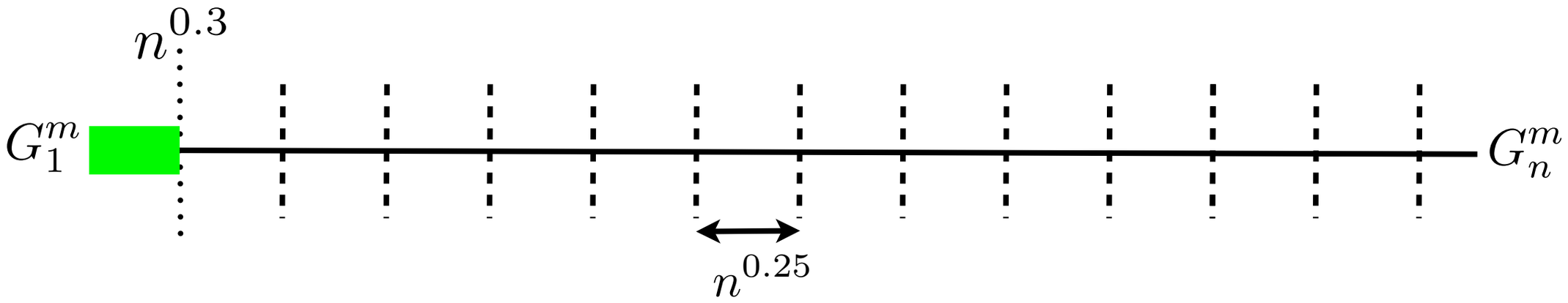
Cascade



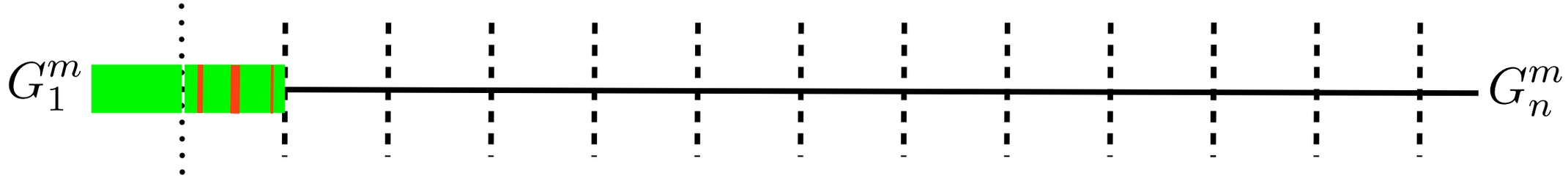
After one phase



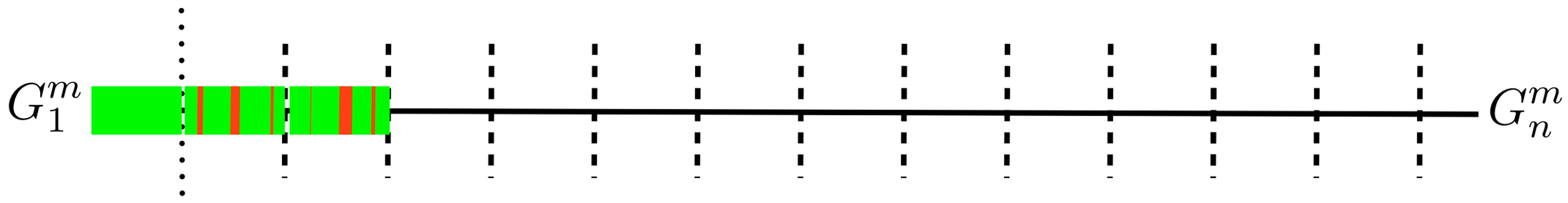
Cascade



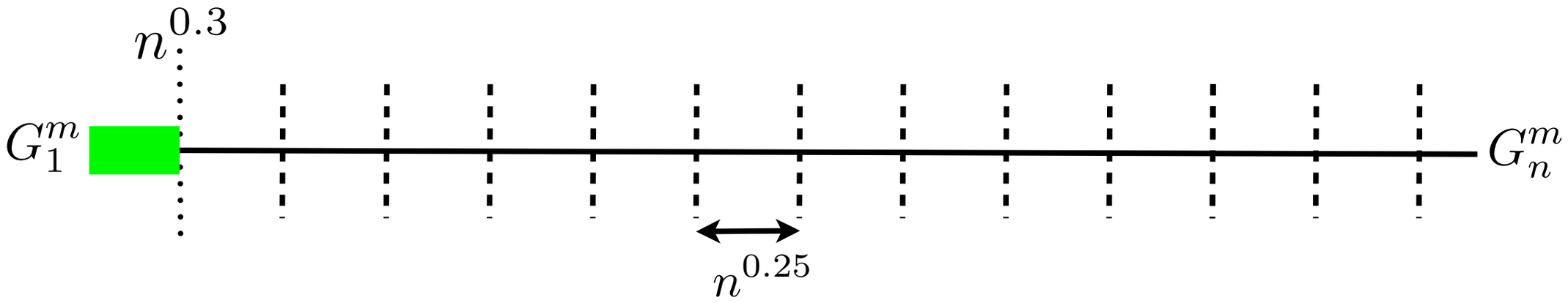
After one phase



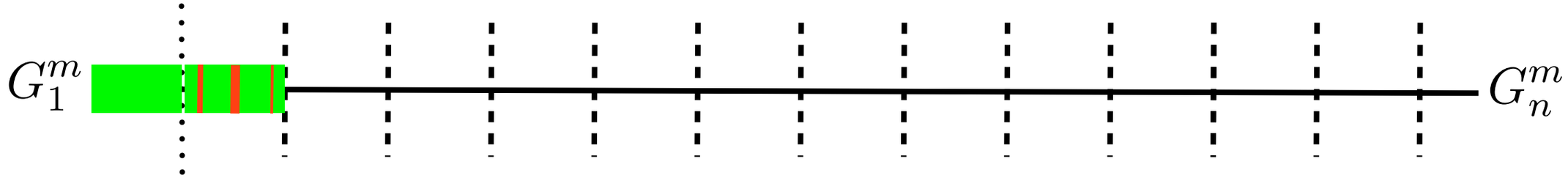
in each phase we do not identify a small fraction, in total we loose a small constant



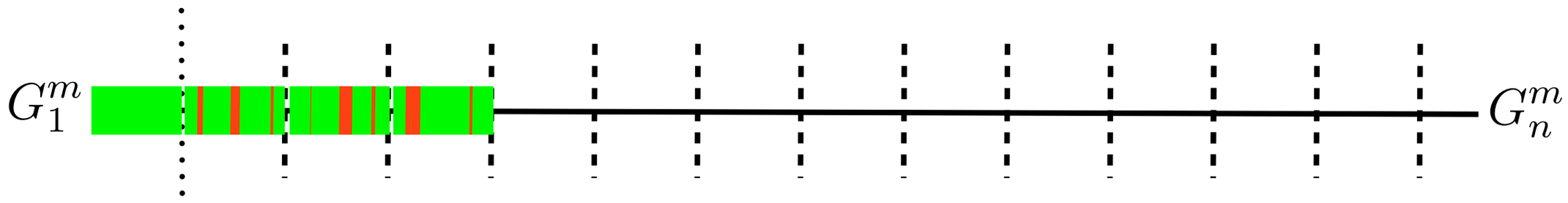
Cascade



After one phase



in each phase we do not identify a small fraction, in total we loose a small constant



Sketch of the proof

- For high degree node we can use concentration results. ✅
- Different nodes of intermediate degree do not share many neighbors. ✅
- High degree nodes help to detect intermediate degree nodes that in turn help to detect small degree nodes. ✅

Results

- *Theorem 1*
  If the underlying network is a G(n,p) graph it is possible to reconcile it completely

- *Theorem 2*
  If the underlying network is a PA graph it is possible to reconcile it a large fraction of it.

Experimental results

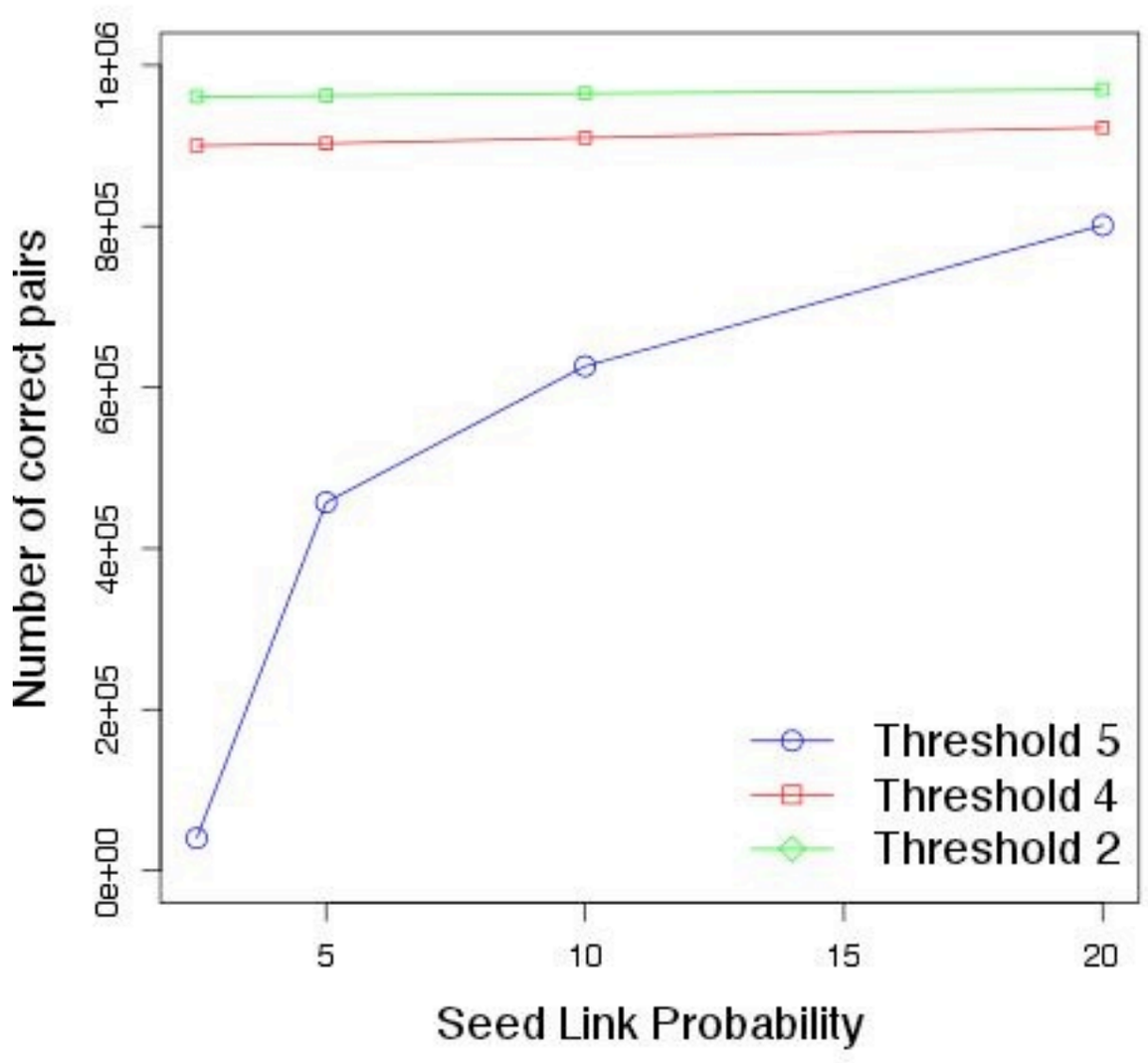
Experiments

Experiments on different graphs:

Network	Number of nodes	Number of edges
PA [5]	1,000,000	20,000,000
RMAT24 [7]	8,871,645	520,757,402
RMAT26 [7]	32,803,311	2,103,850,648
RMAT28 [7]	121,228,778	8,472,338,793
AN [19]	60,026	8,069,546
Facebook [30]	63,731	1,545,686
DBLP [1]	4,388,906	2,778,941
Enron [16]	36,692	367,662
Gowalla [8]	196,591	950,327
French Wikipedia [2]	4,362,736	141,311,515
German Wikipedia [2]	2,851,252	81,467,497

PA experiment

Are our theoretical results robust?



Scalability

How does the algorithm scale with the size of the graph?

Network	Number of nodes	Relative running time
RMAT24	8871645	1
RMAT26	32803311	1.199
RMAT28	121228778	12.544

Facebook experiment

How does the algorithm perform if the underlying graph is a social network?

Pr	Threshold 5		Threshold 4		Threshold 2	
	Good	Bad	Good	Bad	Good	Bad
20%	23915	0	28527	53	41472	203
10%	23832	49	32105	112	38752	213
5%	11091	43	28602	118	36484	236

Facebook experiment

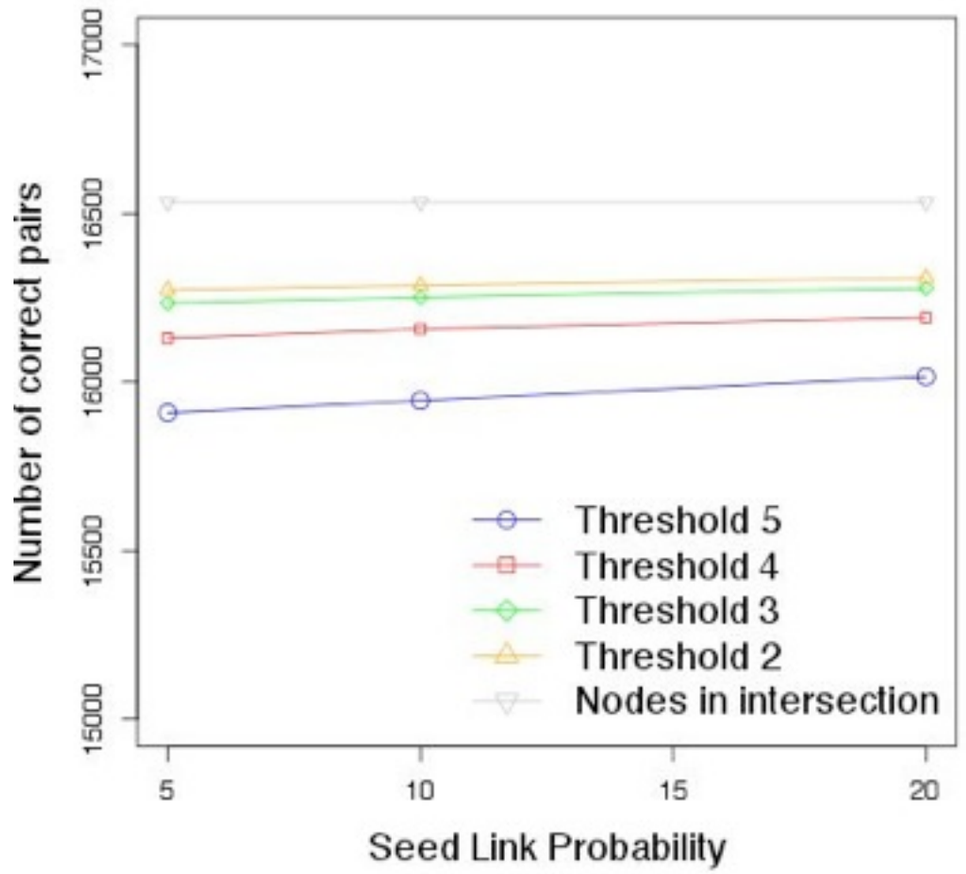
How does the algorithm perform if the underlying graph is a social network?

Pr	Threshold 5		Threshold 4		Threshold 2	
	Good	Bad	Good	Bad	Good	Bad
20%	23915	0	28527	53	41472	203
10%	23832	49	32105	112	38752	213
5%	11091	43	28602	118	36484	236

80% recall!! Can we explain it in theory?

Facebook cascade experiment

What does happen if we generate the underlying network using a cascade process?



Recover almost all the graph in the intersection. Can we explain it in theory?

Affiliation network model

What does happen if we delete all the edges inside a subset of the communities?

Pr	Threshold 4		Threshold 3		Threshold 2	
	Good	Bad	Good	Bad	Good	Bad
10%	54770	0	55863	0	55942	0

More than 80% recall. Can we explain it in theory?

Reconcile different graphs

DBLP: we generate two co-authorship graphs. One considering only publications in even years and the other publication only in odd years.

Reconcile different graphs

DBLP: we generate two co-authorship graphs. One considering only publications in even years and the other publication only in odd years.

Gowalla: we generate two co-checkin graphs. One considering only checkins in even years and the other checkins only in odd years.

Reconcile different graphs

DBLP: we generate two co-authorship graphs. One considering only publications in even years and the other publication only in odd years.

Gowalla: we generate two co-checkin graphs. One considering only checkins in even years and the other checkins only in odd years.

German/French Wikipedia: we crawl the inter-languange links, we use few of them as seed and we check how many links we could recover.

Reconcile different graphs

Pr	Threshold 5		Threshold 4		Threshold 2	
	Good	Bad	Good	Bad	Good	Bad
10	42797	58	53026	641	68641	2985

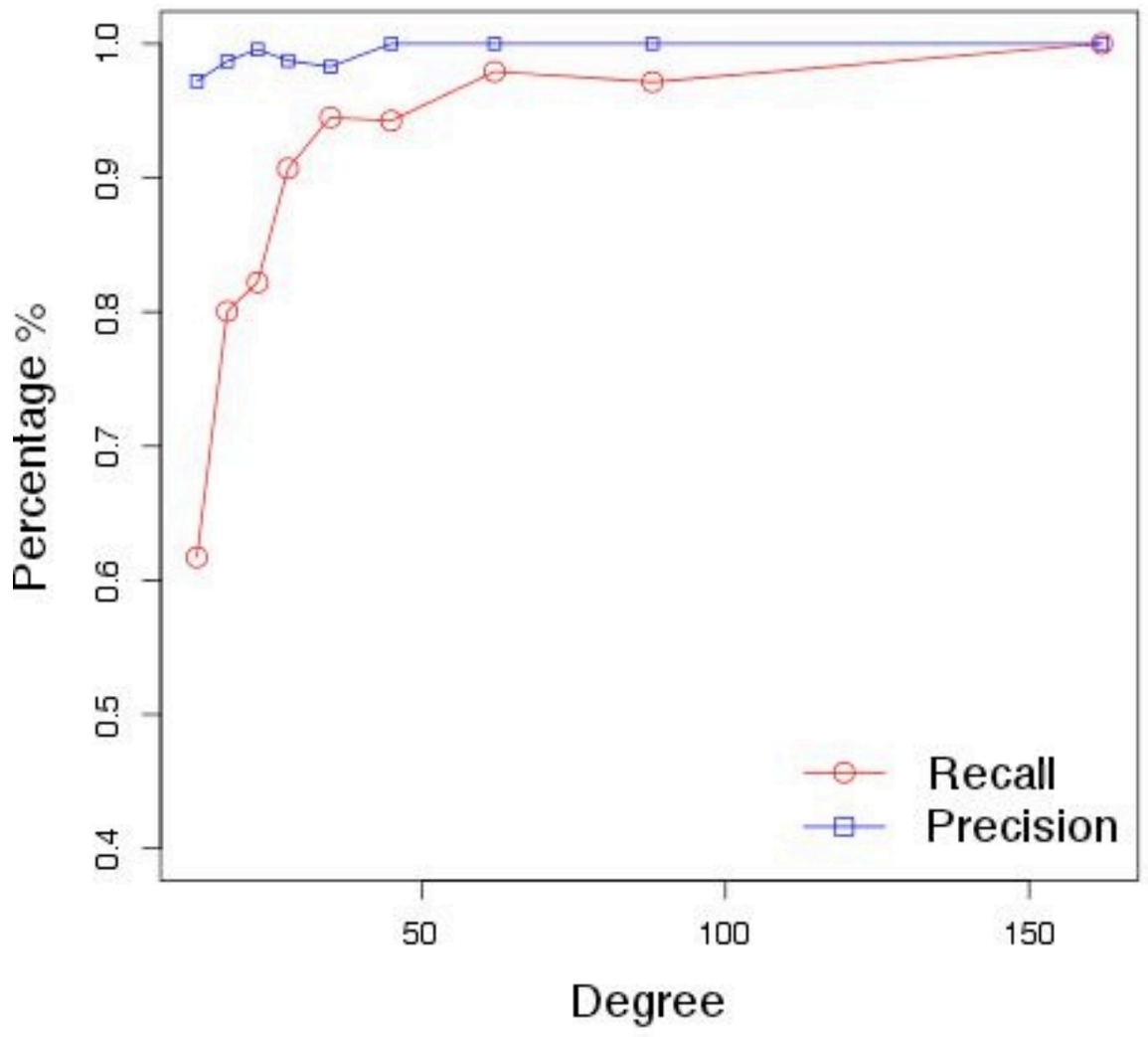
Pr	Threshold 5		Threshold 4		Threshold 2	
	Good	Bad	Good	Bad	Good	Bad
10	5520	29	5917	48	7931	155

Pr	Threshold 5		Threshold 3	
	Good	Bad	Good	Bad
10	108343	9441	122740	14373

Recall for Wikipedia ~30%

Reconcile different graphs

We have really good performance for high degree nodes



Open problems and future directions

Extensions

- Other model of underlying graphs

- Other model of generation of networks

- Adversarial underlying network, error in seed links

Limitation of the current model

- Users' degree depend varies in different social networks
- How can we model this more general setting?

Better algorithm

- Currently exploring only direct neighborhood
- Can we design better algorithms?

Thanks!