Recent advances on random graph matching

Jiaming Xu

The Fuqua School of Business
Duke University

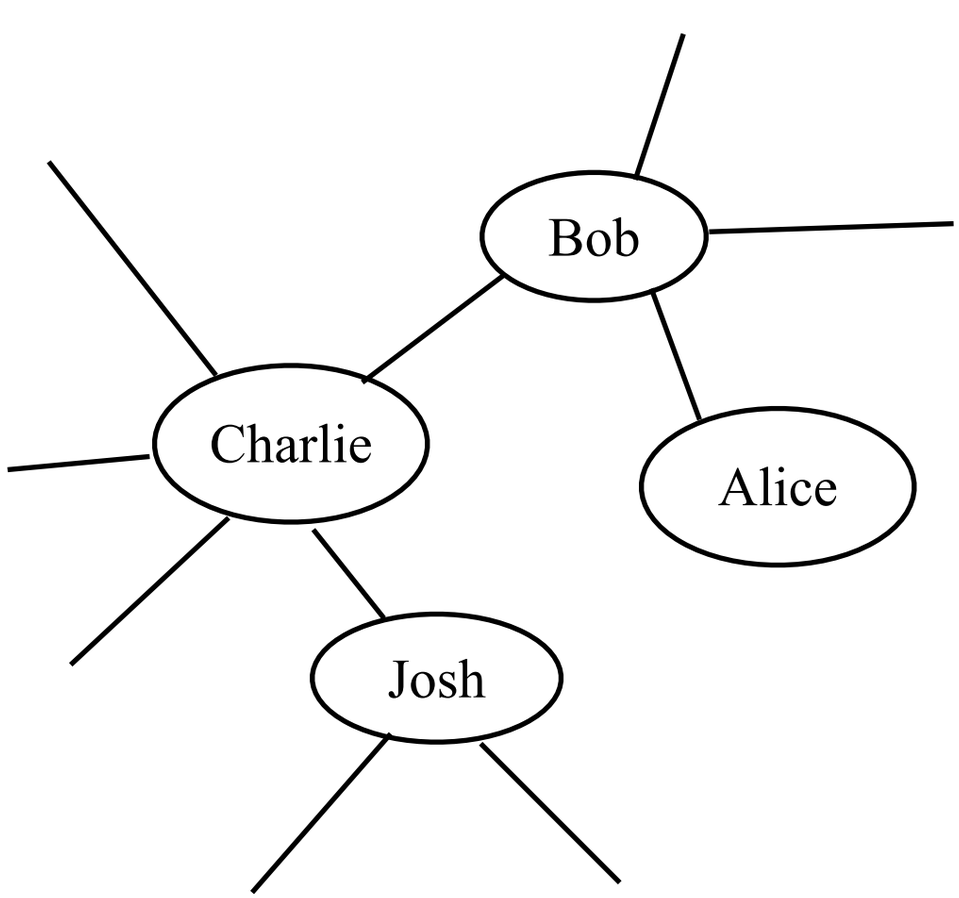
Joint work: Cheng Mao (GaTech), Yihong Wu (Yale), and Sophie H. Yu (Wharton)

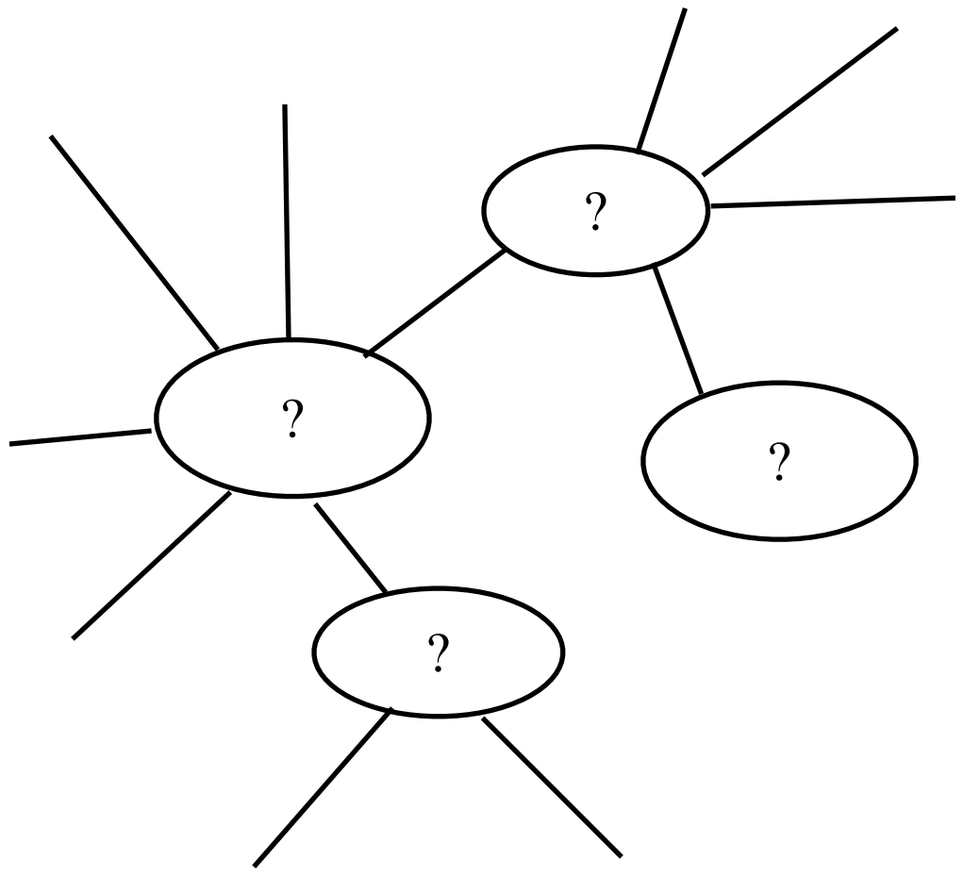
Stochastic Networks Conference 2024

Outline

- Motivation and problem setup
- Information-theoretic limits
- Efficient algorithms
- Concluding remarks

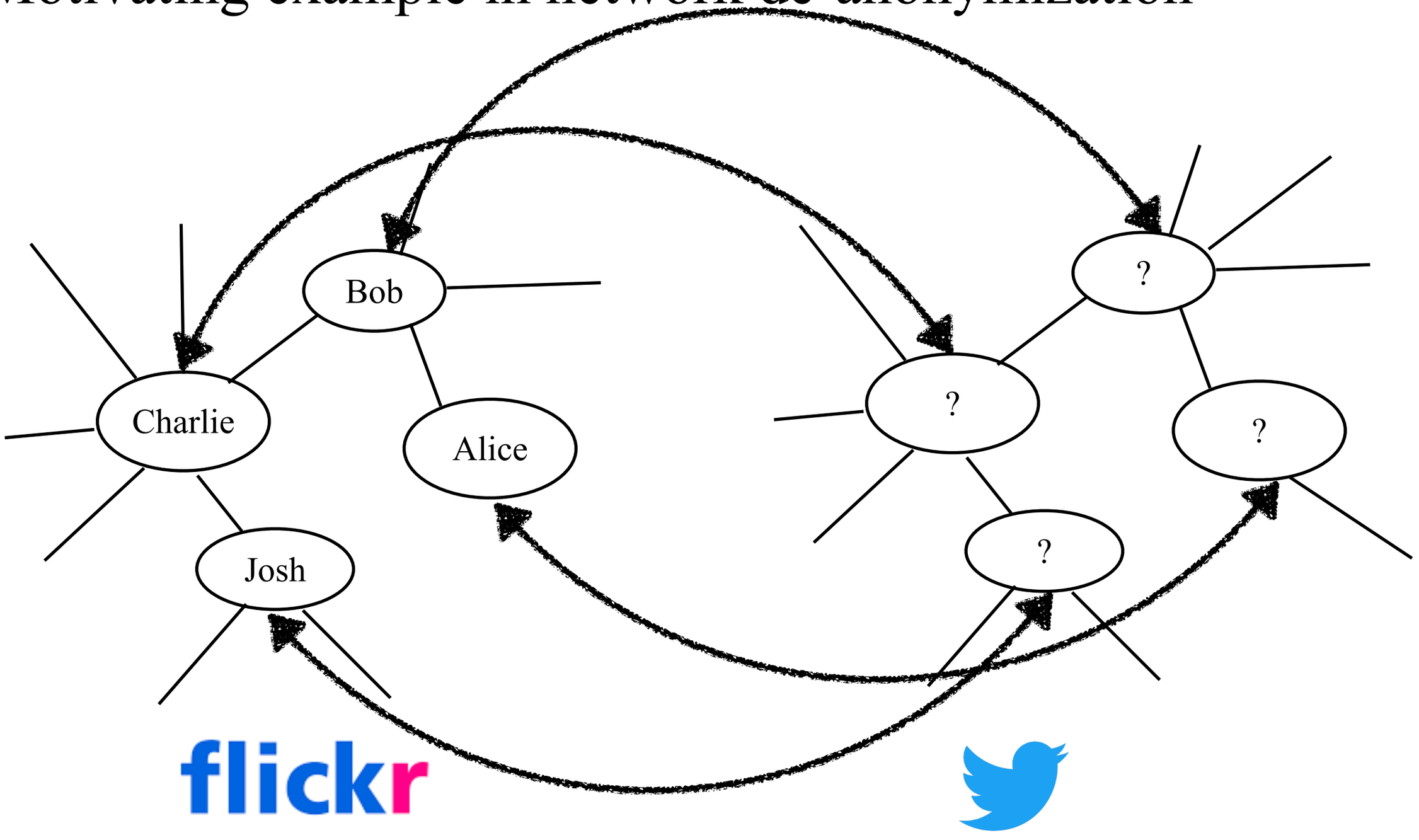
Motivating example in network de-anonymization



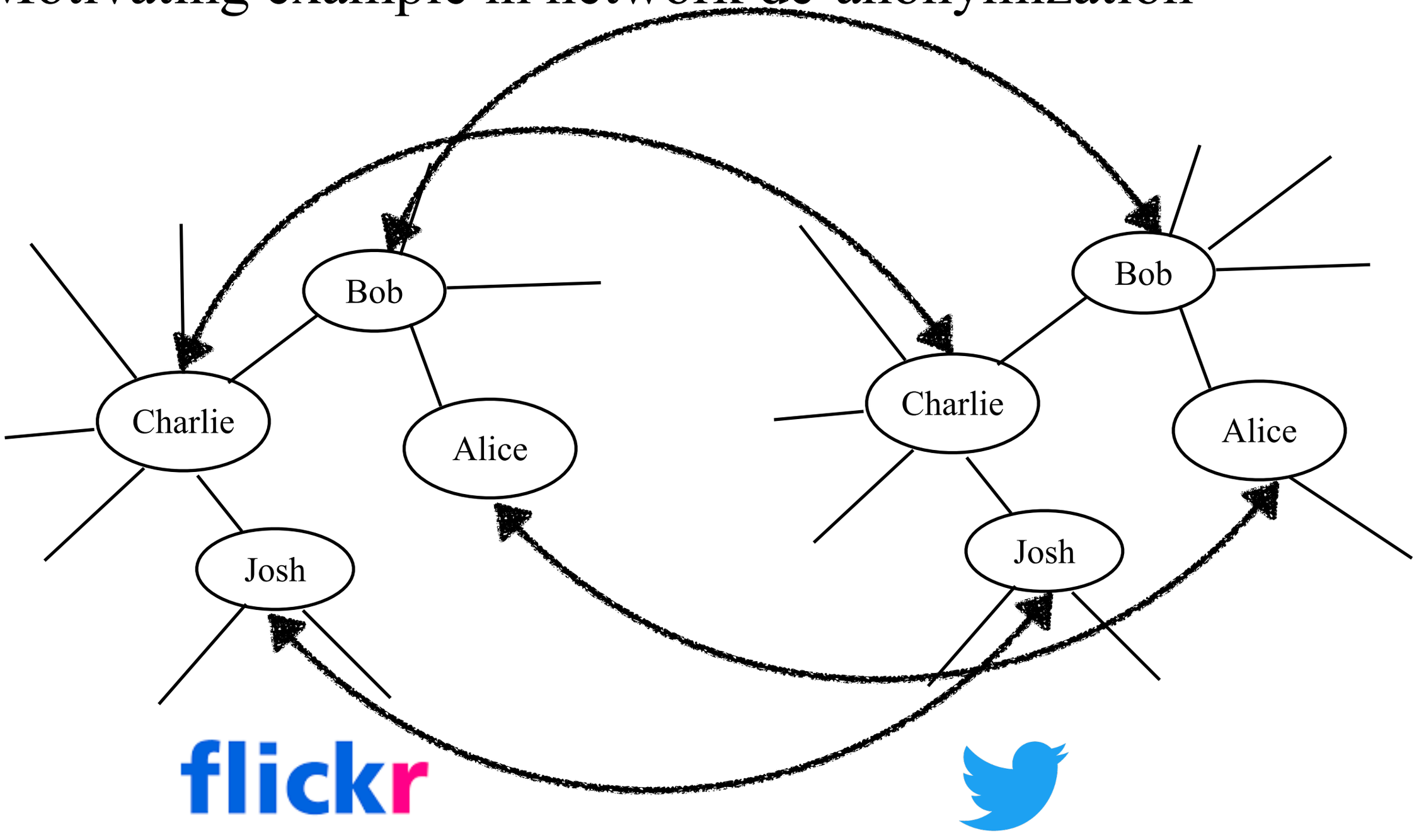


flickr

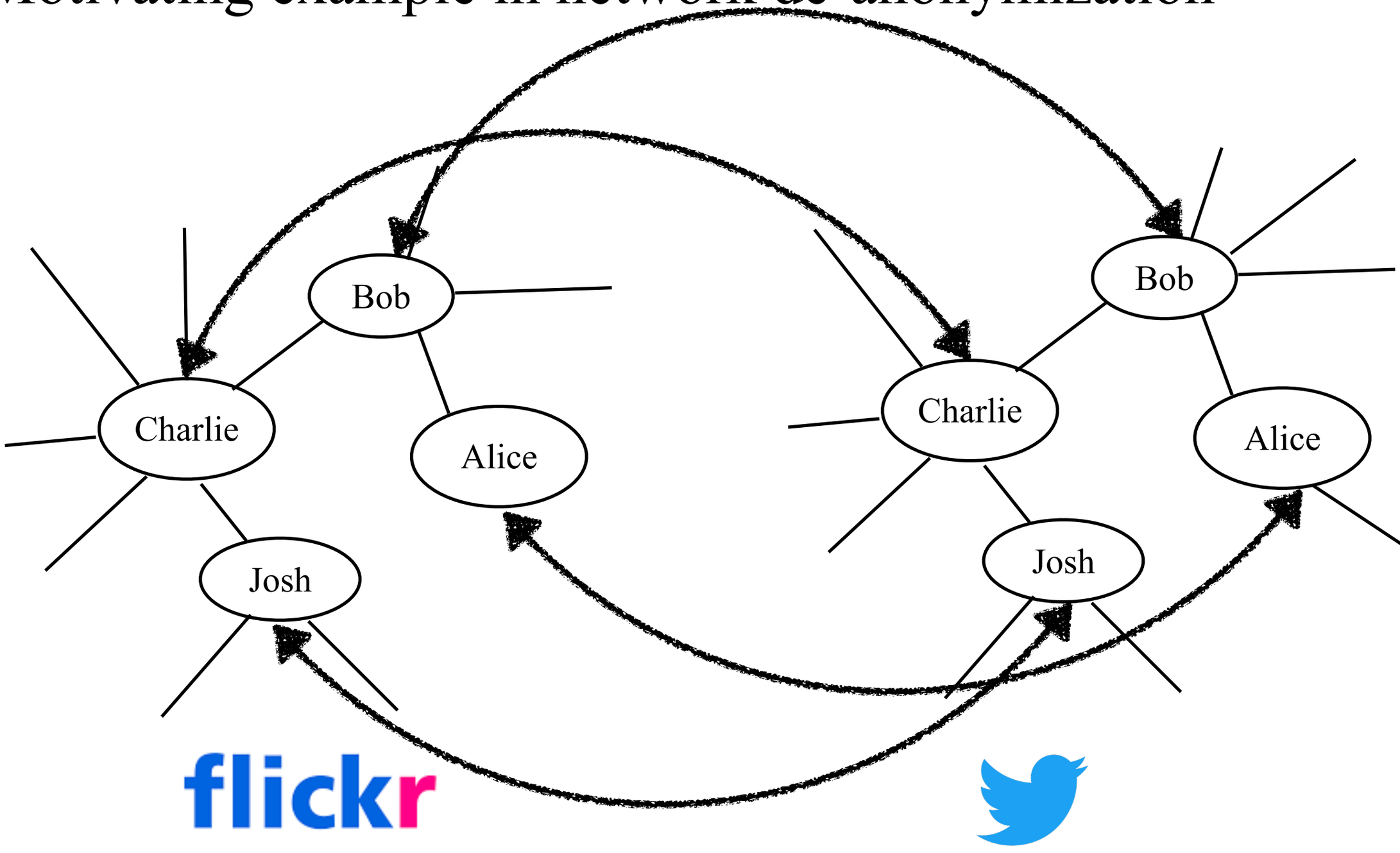
Motivating example in network de-anonymization



Motivating example in network de-anonymization



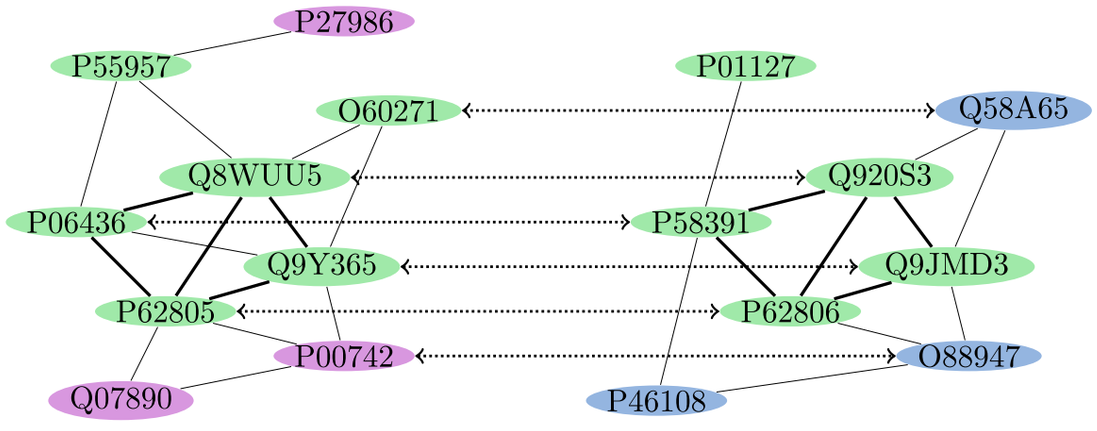
Motivating example in network de-anonymization



- Narayanan and Shmatikov correctly identified a fraction of users between Twitter and Flickr in 2009.

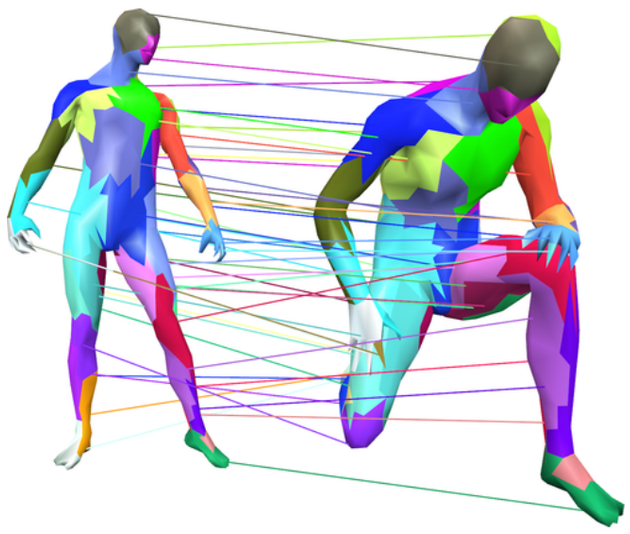
Applications

- Biology [Singh-Xu-Berger ’2008; Kazemi et al. ’2016]

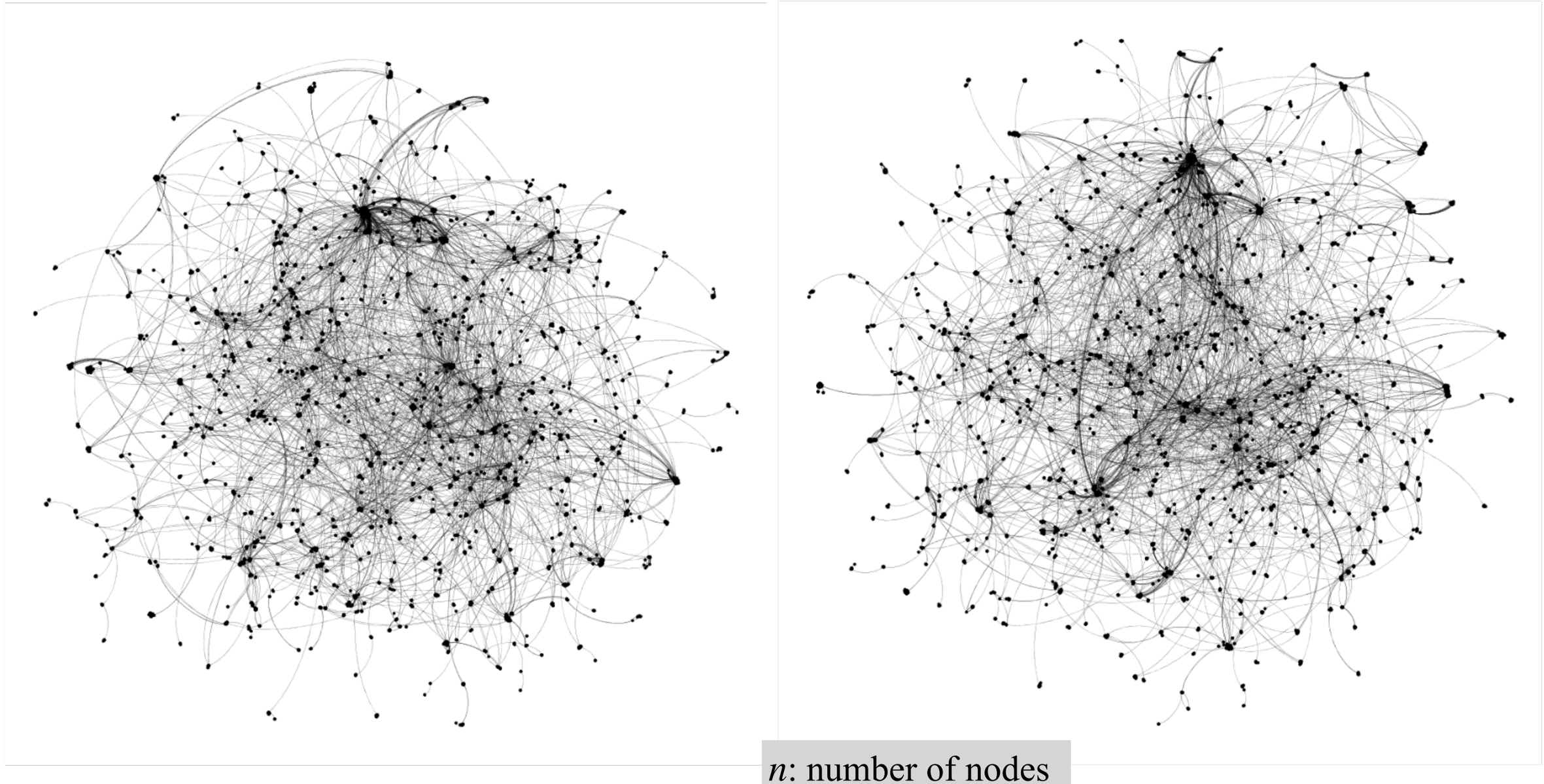


Human network Mouse network

- Computer Vision [Lähner et al. ’2016; Fan-Mao-Wu-Xu ’2020]



Real-world challenges



n: number of nodes

- **Computational:** # of possible node mappings is $n!$ $(100! \approx 10^{158})$;
- **Statistical:** two graphs may be correlated but not exactly the same;

Beyond worst-case intractability

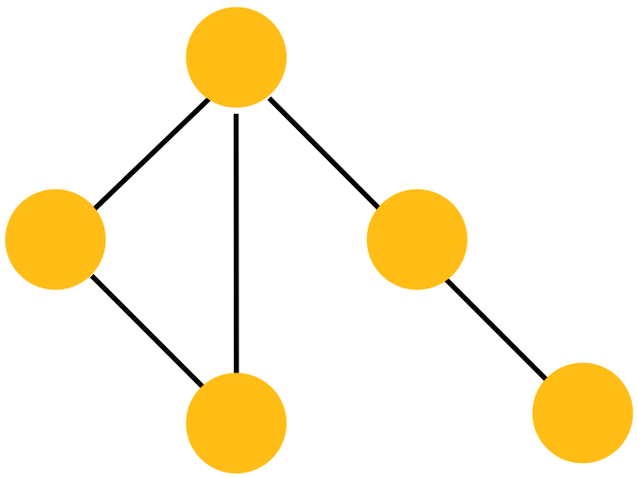
- **NP-hard** for matching two graphs in worst case
- However, real networks are not arbitrary and have latent structures

Beyond worst-case intractability

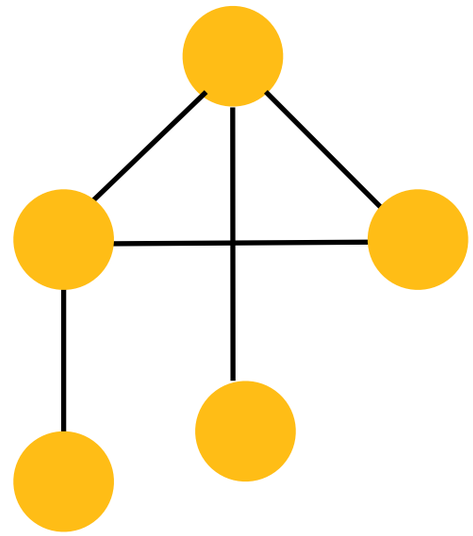
- **NP-hard** for matching two graphs in worst case
- However, real networks are not arbitrary and have latent structures
- Recent surge of interest on the average-case analysis
  - [Cullina-Kiyavash ’16,17]
  - [Cullina-Kiyavash-Mittal-Poor ’19, Dai-Cullina-Kiyavash-Grossglauser ’19, Ding-Ma-Wu-Xu ’18, Barak-Chou-Lei-Schramm-Sheng ’19, Fan-Mao-Wu-Xu ’19a,19b]
  - [Ganassali-Massoulié ’20, Hall-Massoulié ’20, Wu-Xu-Yu ’21, Ganassali-Lelarge-Massoulié ’21, Mao-Rudelson-Tikhomirov ’21a, 21b]
  - [Ding-Du ’21, 22, Mao-Wu-Xu-Yu ’22, Ganassali-Massoulié-Semerjian ’22]
  - [Ding-Li’ 23, Ding-Du-Li’ 23]
  - …

Correlated Erdős–Rényi graph model[Pedarsani-Grossglauser ’11]

Correlated Erdős–Rényi graph model[Pedarsani-Grossglauser '11]

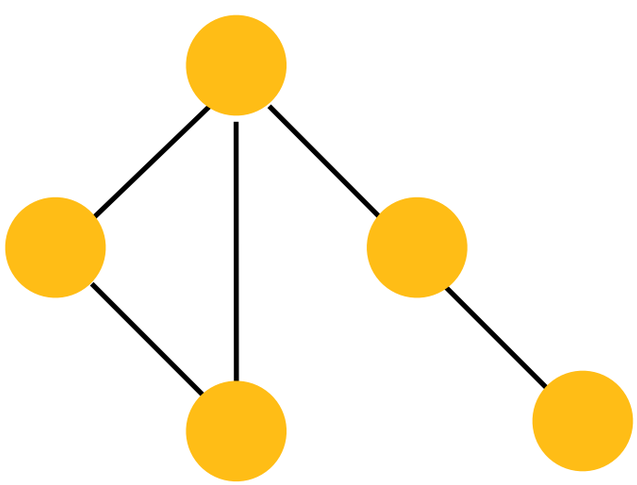


$A \sim \mathcal{G}(n, q)$



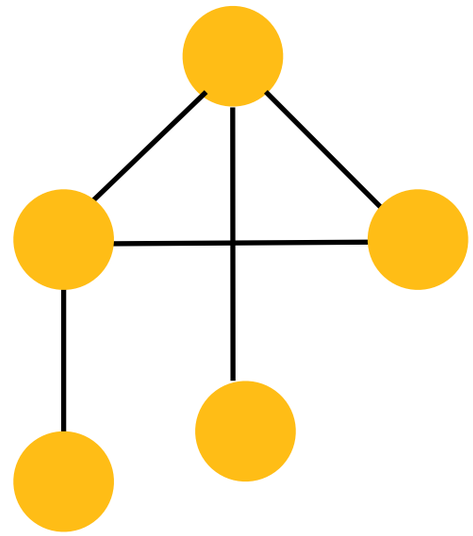
$B \sim \mathcal{G}(n, q)$

Correlated Erdős–Rényi graph model[Pedarsani-Grossglauser ’11]



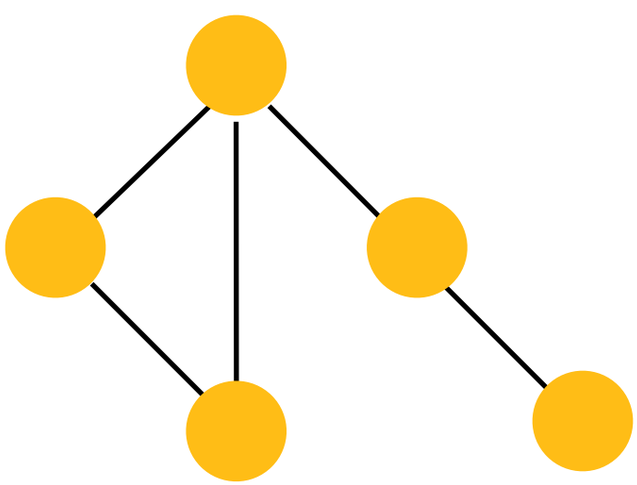
$A \sim \mathcal{G}(n, q)$

n: number of nodes



$B \sim \mathcal{G}(n, q)$

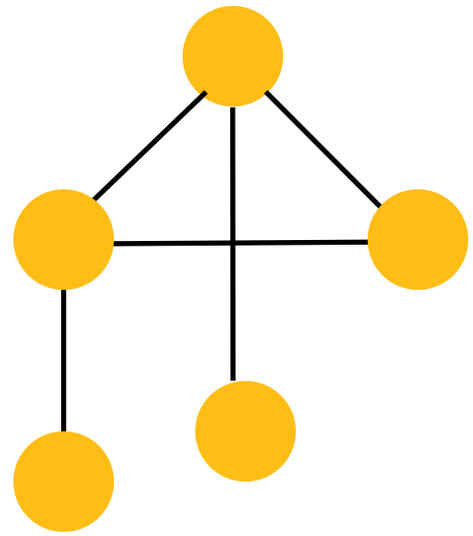
Correlated Erdős–Rényi graph model[Pedarsani-Grossglauser ’11]



$A \sim \mathcal{G}(n, q)$

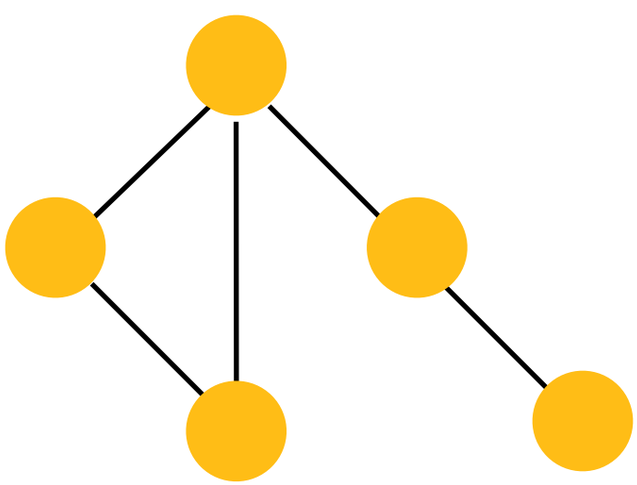
n: number of nodes

Each two nodes are connected with probability q

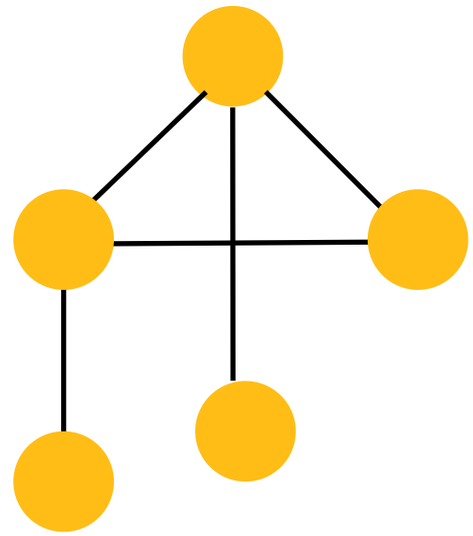


$B \sim \mathcal{G}(n, q)$

Correlated Erdős–Rényi graph model[Pedarsani-Grossglauser '11]



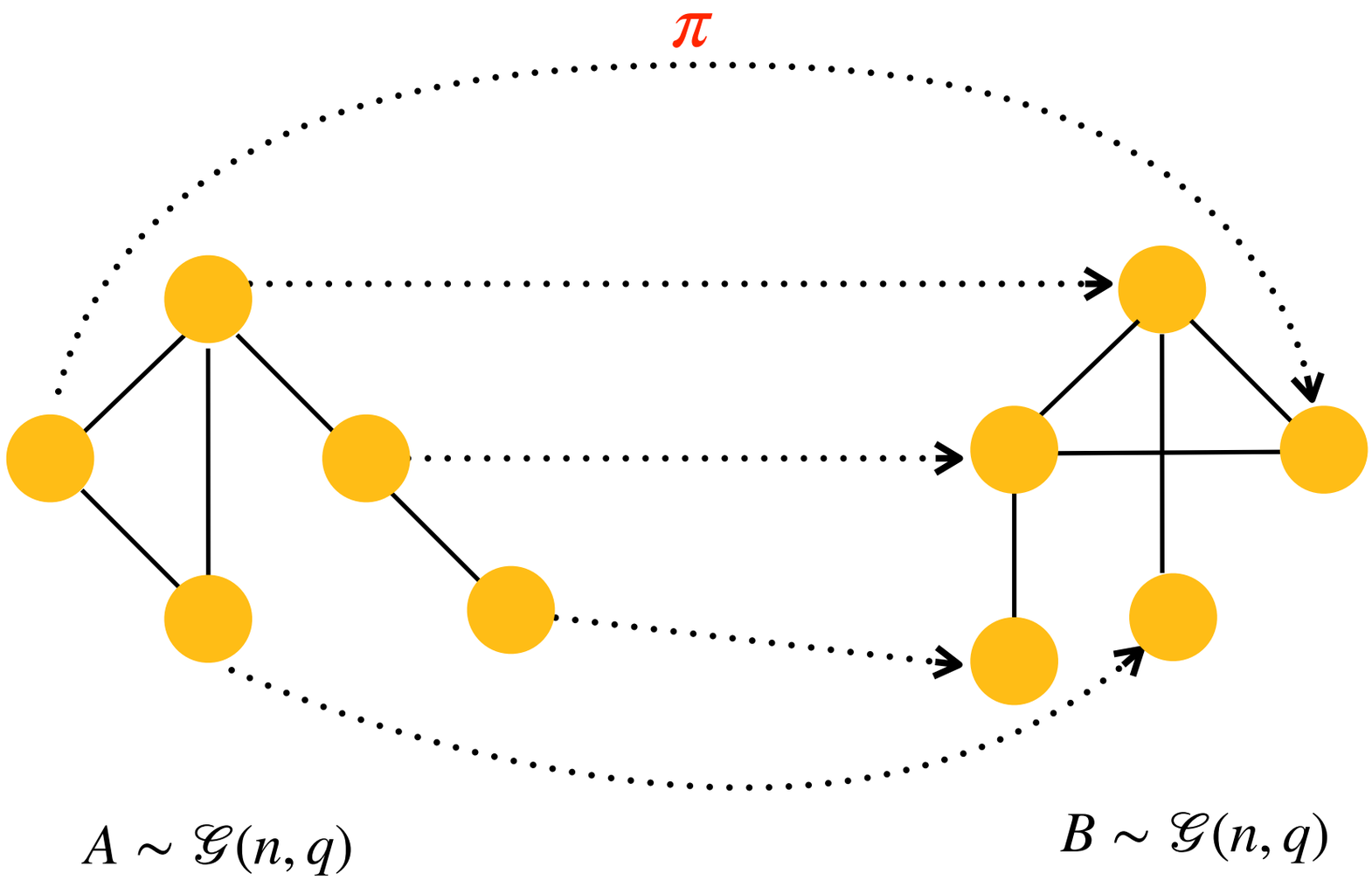
$A \sim \mathcal{G}(n, q)$



$B \sim \mathcal{G}(n, q)$

- A and B are edge-wise correlated (ρ) under the hidden node correspondence π.

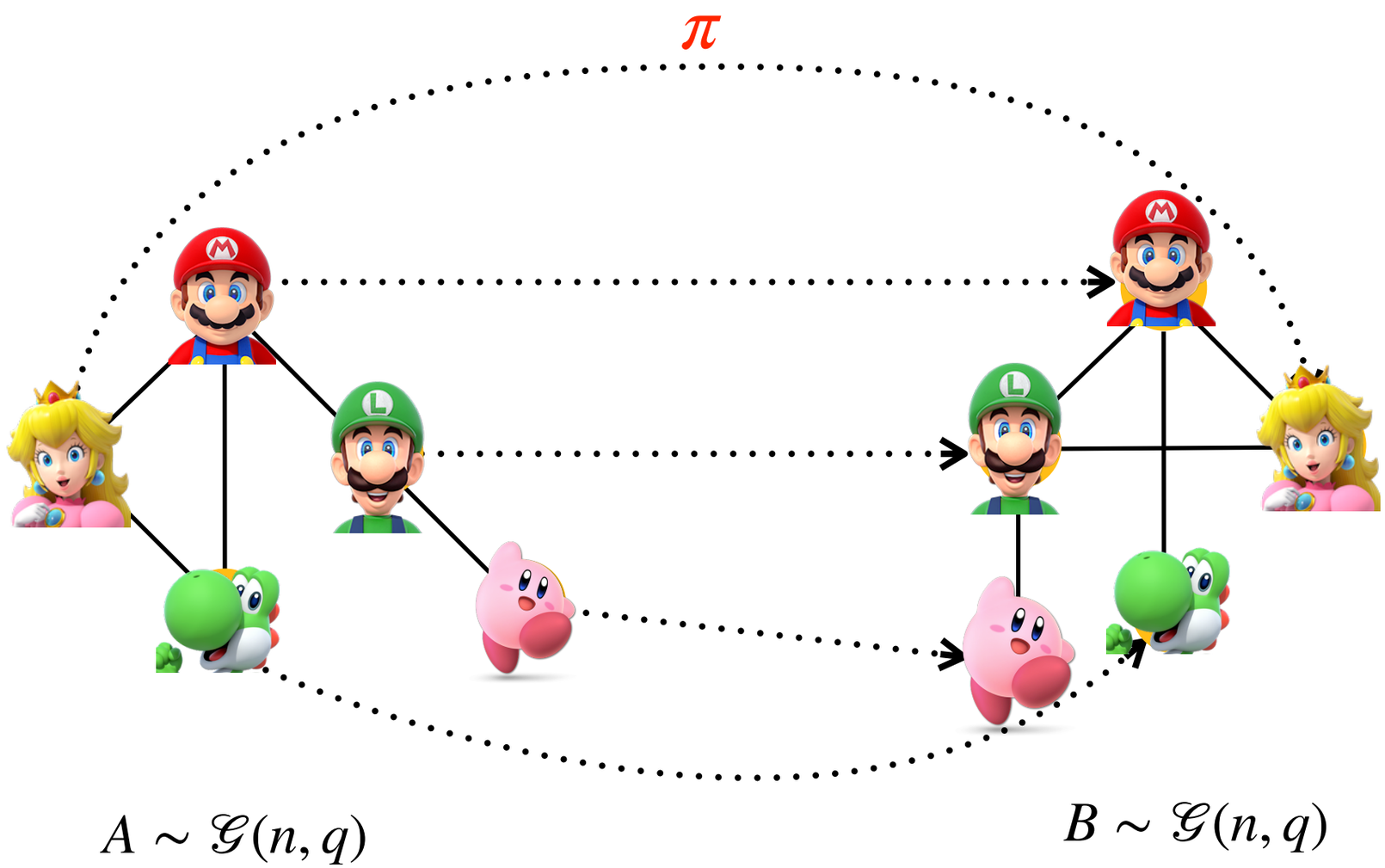
Correlated Erdős–Rényi graph model[Pedarsani-Grossglauser '11]



- A and B are edge-wise correlated (ρ) under the hidden node correspondence π.

$\left(A_{ij}, B_{\pi(i)\pi(j)}\right)$ are i.i.d. pairs of two Bern(q) with correlation ρ

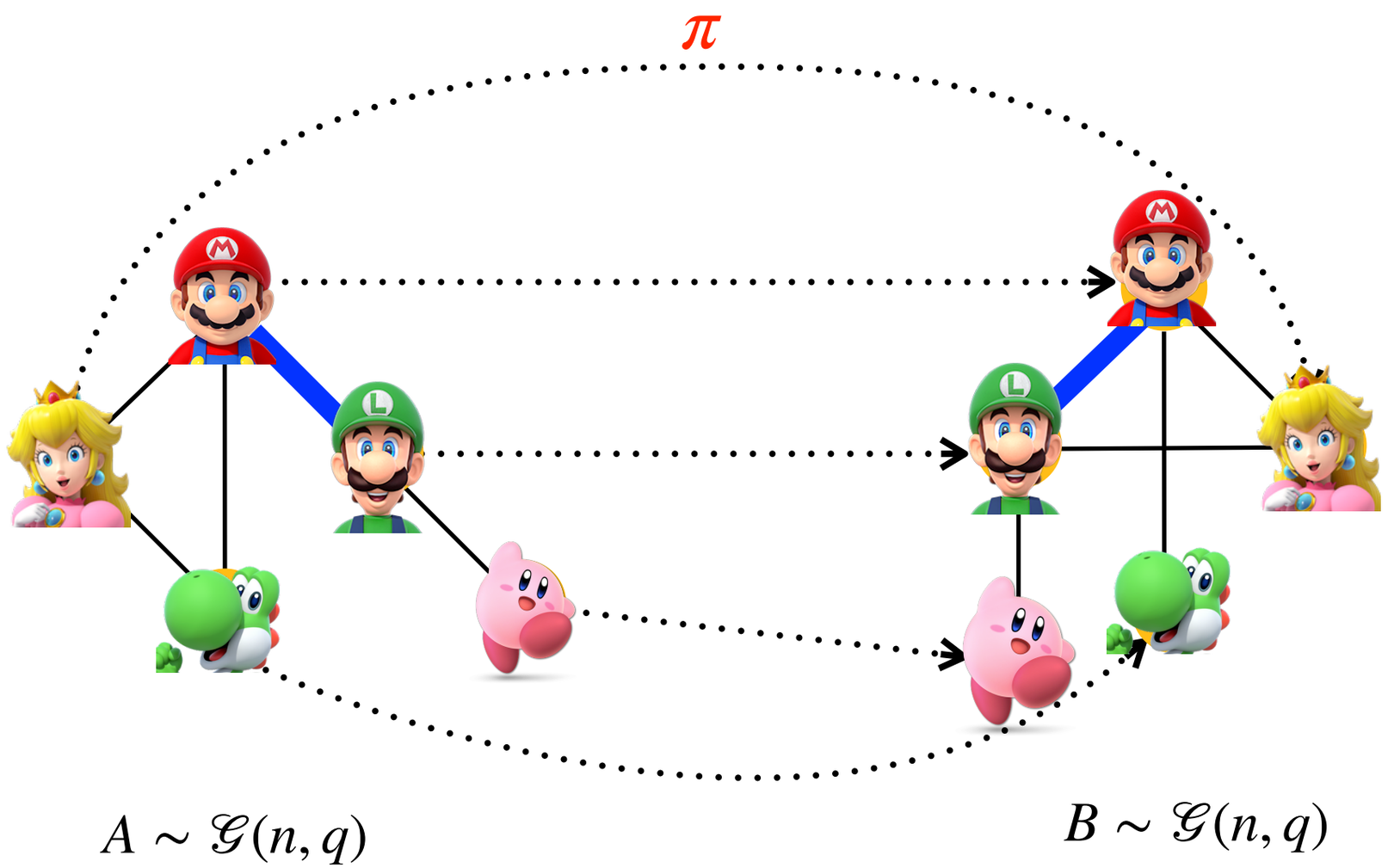
Correlated Erdős–Rényi graph model[Pedarsani-Grossglauser '11]



- A and B are edge-wise correlated (ρ) under the hidden node correspondence π.

$\left(A_{ij}, B_{\pi(i)\pi(j)}\right)$ are i.i.d. pairs of two Bern(q) with correlation ρ

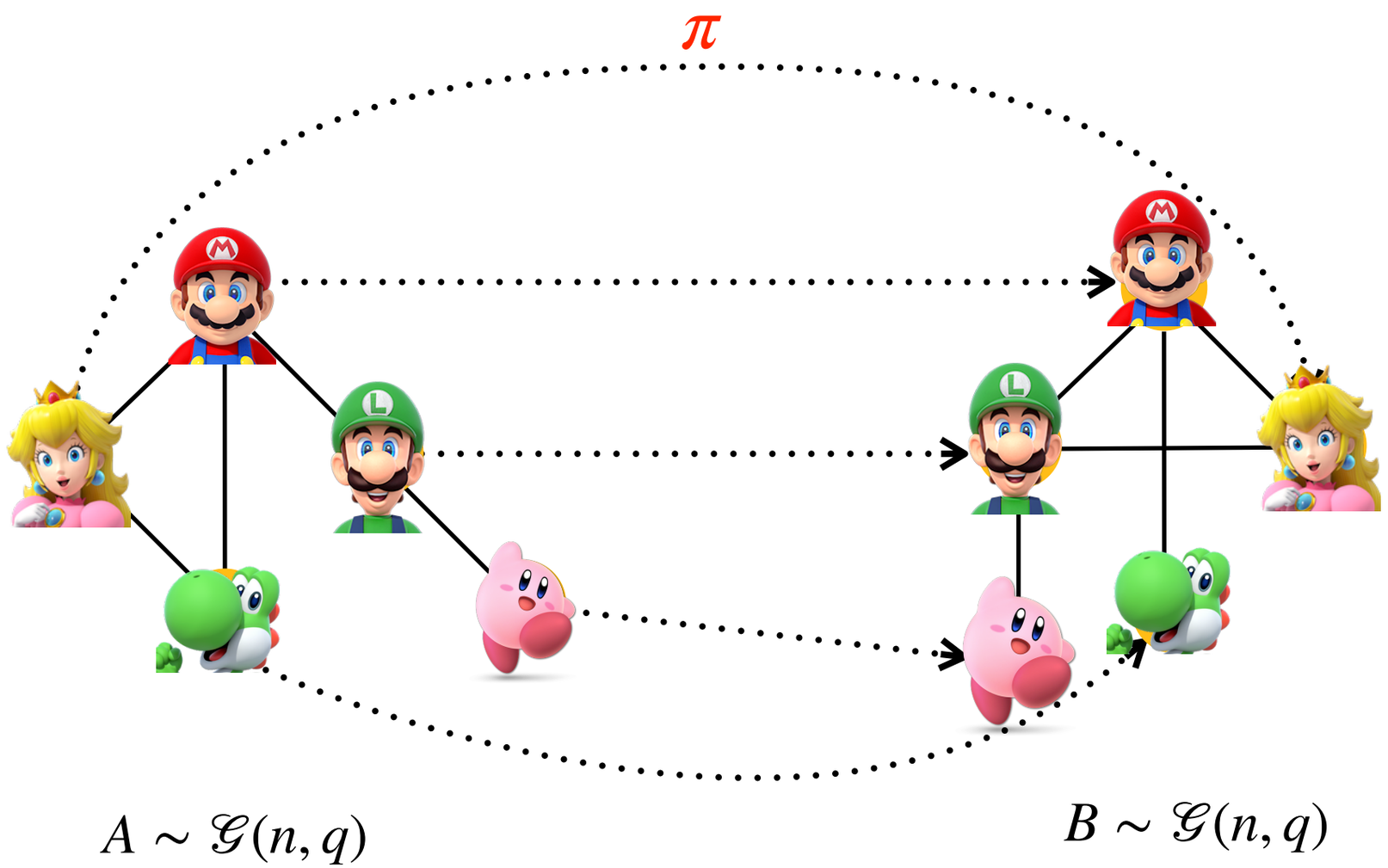
Correlated Erdős–Rényi graph model[Pedarsani-Grossglauser '11]



- A and B are edge-wise correlated (ρ) under the hidden node correspondence π.

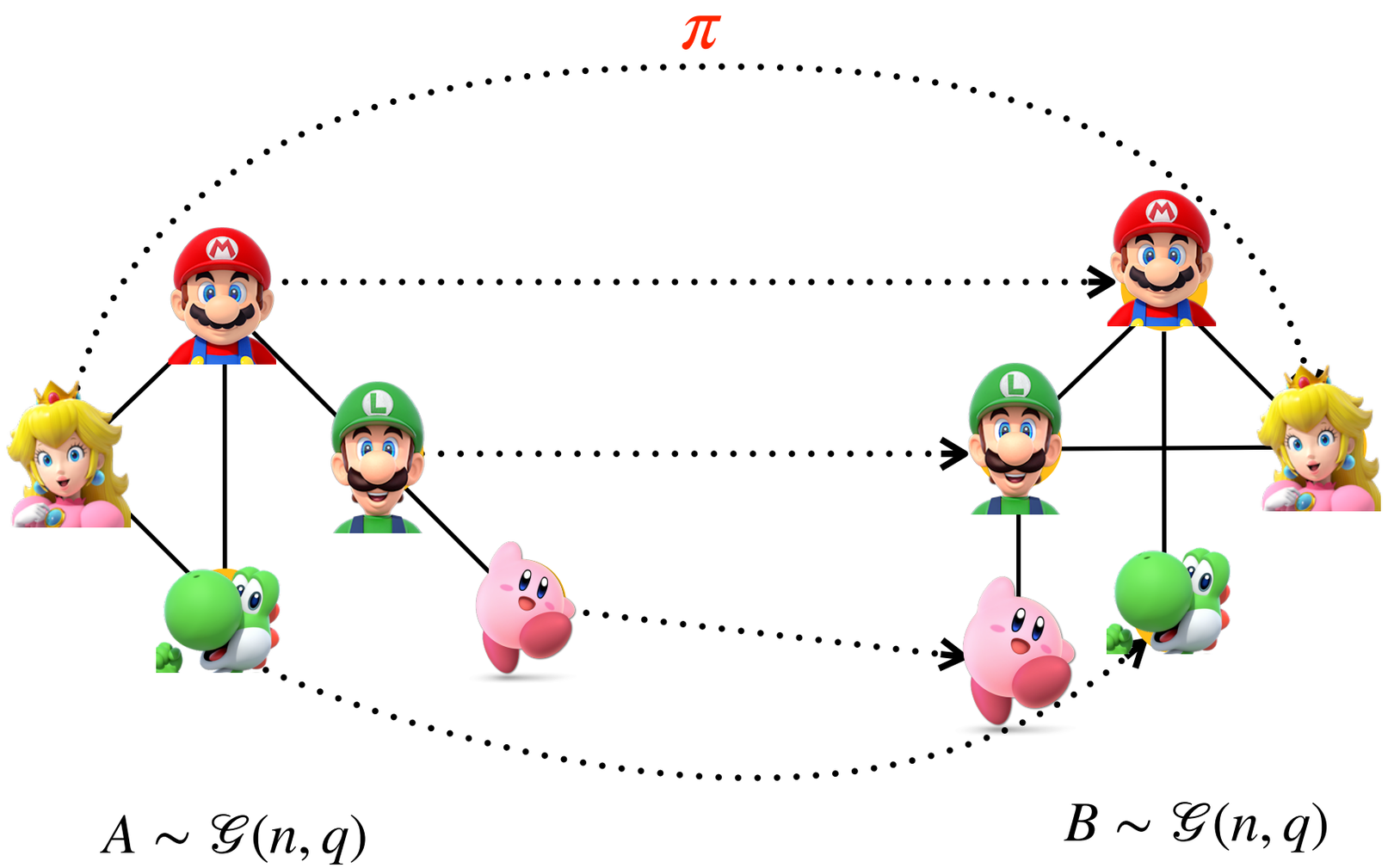
$\left(A_{ij}, B_{\pi(i)\pi(j)}\right)$ are i.i.d. pairs of two Bern(q) with correlation ρ

Correlated Erdős–Rényi graph model[Pedarsani-Grossglauser '11]



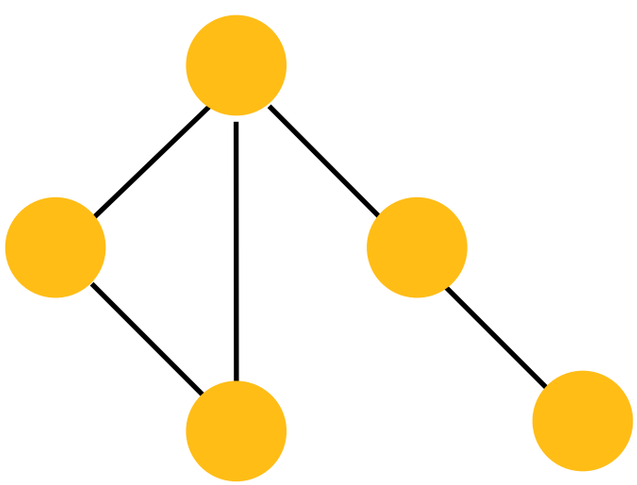
- A and B are edge-wise correlated (ρ) under the hidden node correspondence π.
- ρ is the edge correlation.

Correlated Erdős–Rényi graph model[Pedarsani-Grossglauser '11]

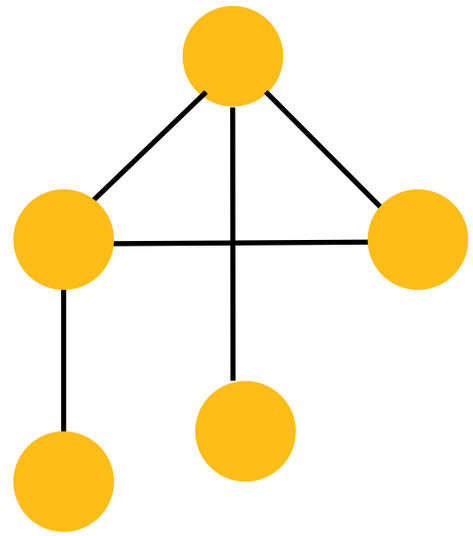


- A and B are edge-wise correlated (ρ) under the hidden node correspondence π.
- ρ is the edge correlation.
- nq is the average degree for A and B.

Correlated Erdős–Rényi graph model[Pedarsani-Grossglauser '11]



$A \sim \mathcal{G}(n, q)$

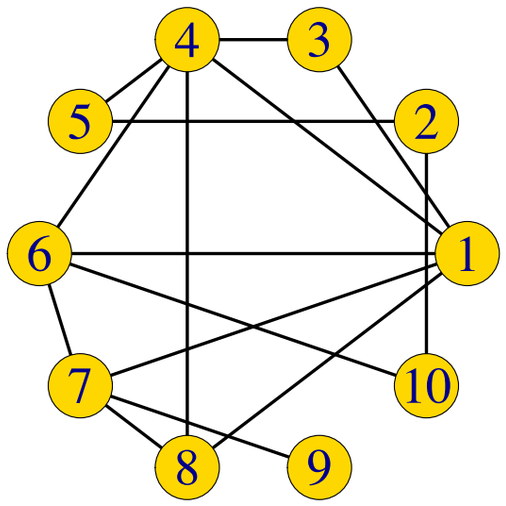


$B \sim \mathcal{G}(n, q)$

- A and B are edge-wise correlated (ρ) under the hidden node correspondence π.
- ρ is the edge correlation.
- nq is the average degree for A and B.

Observe A and B, we want to **recover** the hidden node correspondence π, when $n \to \infty$.

Alternative view: subsampling model

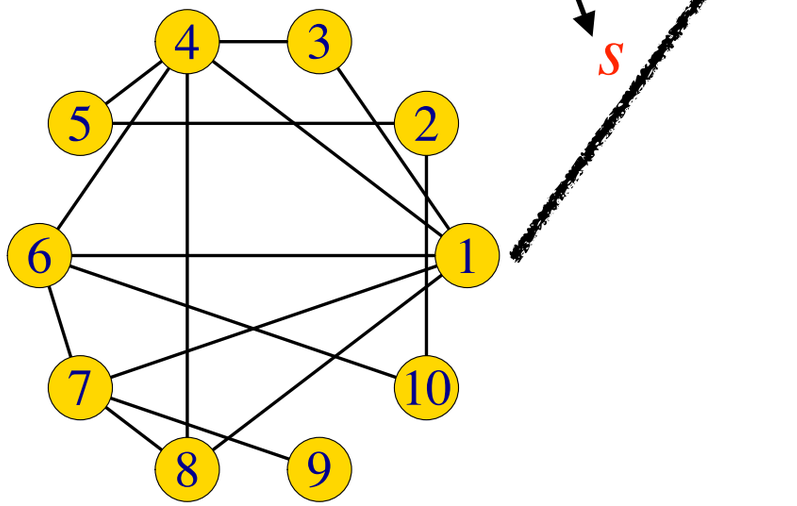


$G \sim \mathcal{G}(n,p)$

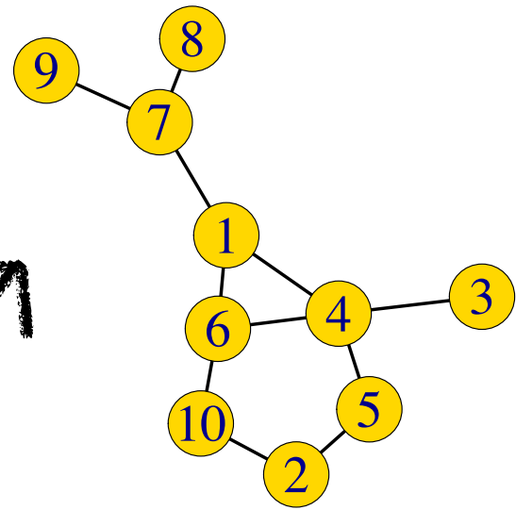
Parent graph edge probability

Alternative view: subsampling model

Edge subsampling probability



s



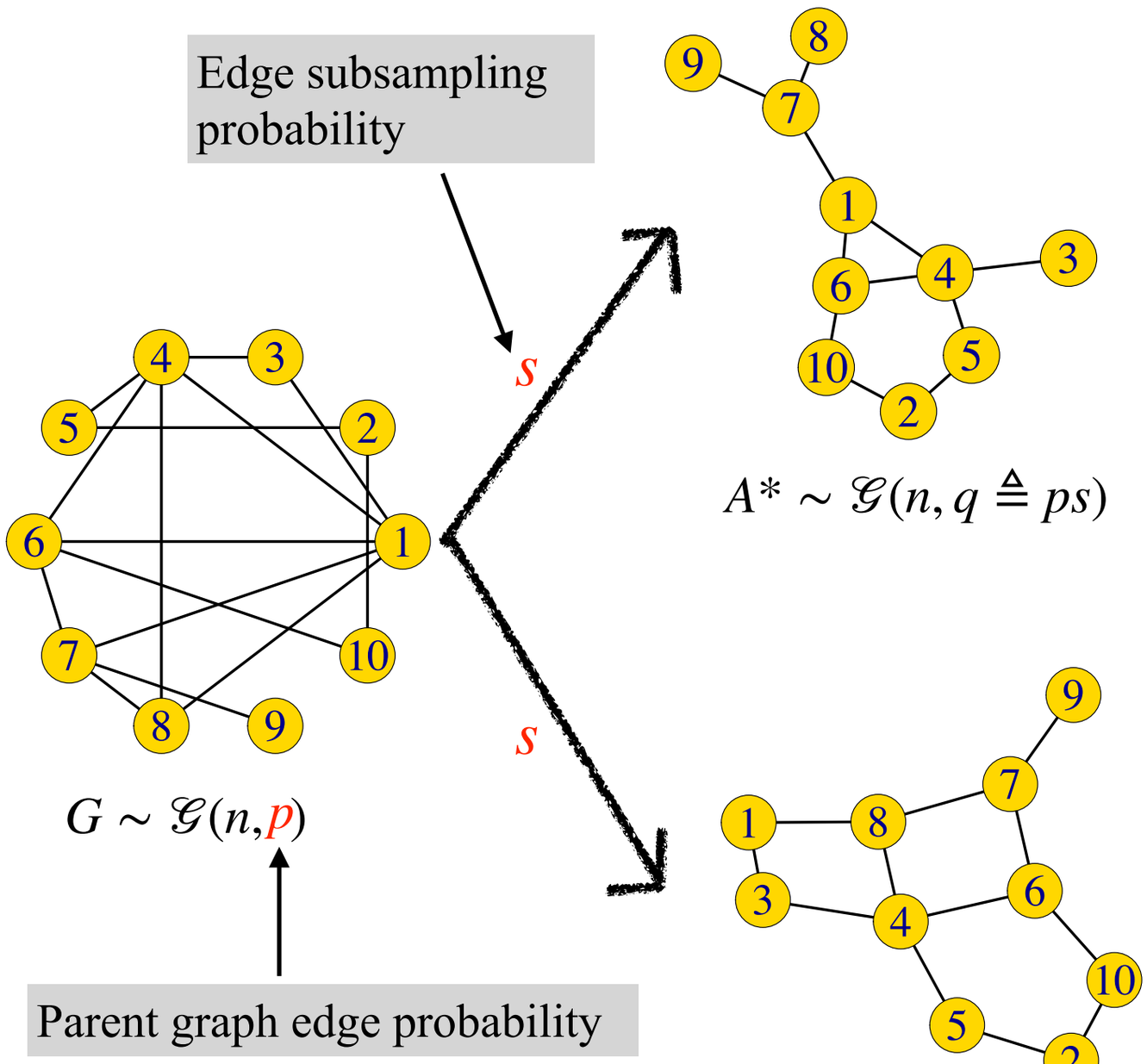
$A^* \sim \mathcal{G}(n, q \triangleq ps)$

$G \sim \mathcal{G}(n,p)$

Parent graph edge probability

Alternative view: subsampling model

Edge subsampling probability



$G \sim \mathcal{G}(n,p)$

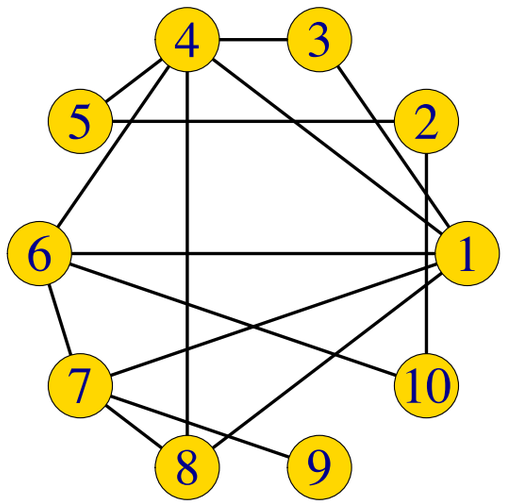
Parent graph edge probability

$A^* \sim \mathcal{G}(n, q \triangleq ps)$

$B \sim \mathcal{G}(n, q \triangleq ps)$

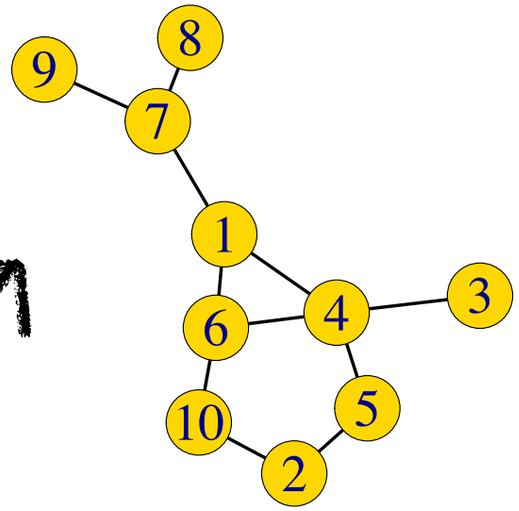
Alternative view: subsampling model

Edge subsampling probability



$G \sim \mathcal{G}(n,p)$

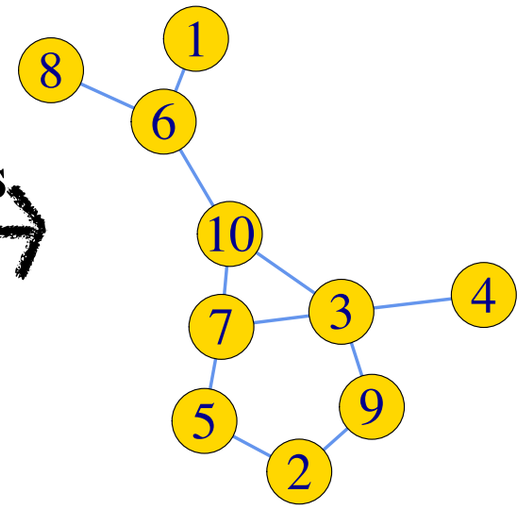
s



$A^* \sim \mathcal{G}(n, q \triangleq ps)$

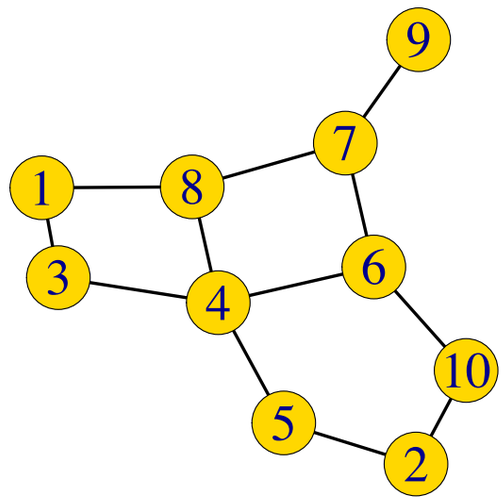
Permute the node labels

By π uniformly



$A \sim \mathcal{G}(n, q \triangleq ps)$

s

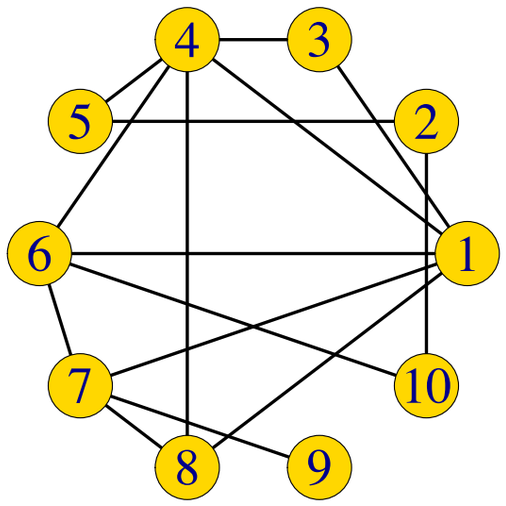


$B \sim \mathcal{G}(n, q \triangleq ps)$

Parent graph edge probability

Alternative view: subsampling model

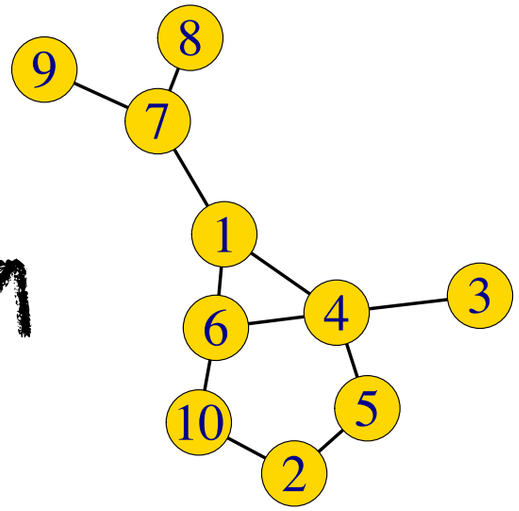
Edge subsampling probability



$G \sim \mathcal{G}(n,p)$

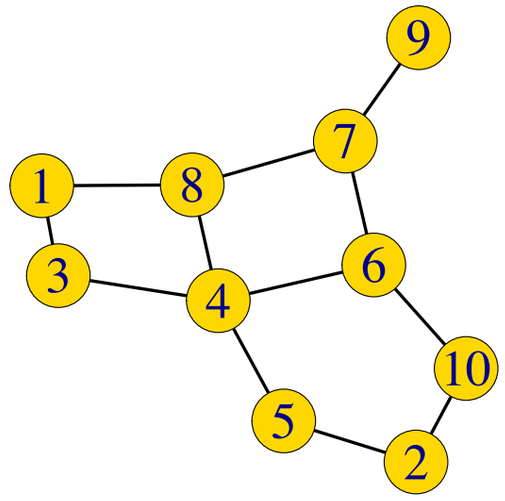
Parent graph edge probability

s



$A^* \sim \mathcal{G}(n, q \triangleq ps)$

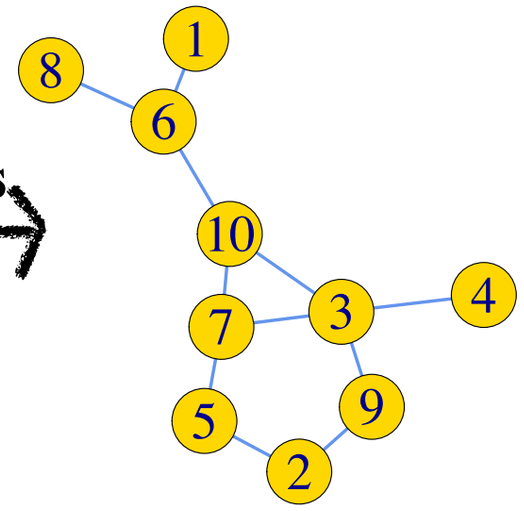
s



$B \sim \mathcal{G}(n, q \triangleq ps)$

Permute the node labels

By π uniformly



$A \sim \mathcal{G}(n, q \triangleq ps)$

- A and B are edge-wise correlated under the latent permutation π with correlation parameter $\rho = \dfrac{s-q}{1-q}$
- $A^* \wedge B \sim \mathcal{G}(n, ps^2)$

Outline

- Motivation and problem setup
- Information-theoretic limits
- Efficient algorithms
- Concluding remarks

MLE: Quadratic Assignment Problem (QAP)

- Maximum likelihood estimator:

$$\pi_{\mathrm{ML}} = \arg\max_{\hat{\pi}} \sum_{i<j} A_{ij} B_{\hat{\pi}(i)\hat{\pi}(j)} \quad \text{(QAP)}$$

MLE: Quadratic Assignment Problem (QAP)

- Maximum likelihood estimator:

$$\pi_{\mathrm{ML}} = \arg\max_{\hat{\pi}} \sum_{i<j} A_{ij} B_{\hat{\pi}(i)\hat{\pi}(j)} \quad \text{(QAP)}$$

  - QAP was introduced by [Koopmans-Beckmann '57]

COWLES FOUNDATION DISCUSSION PAPER, NO. 4*

Assignment Problems and the Location of Economic Activities**

by

Tjalling C. Koopmans and Martin Beckmann

MLE: Quadratic Assignment Problem (QAP)

- Maximum likelihood estimator:

$$\pi_{\mathrm{ML}} = \arg\max_{\hat{\pi}} \sum_{i<j} A_{ij} B_{\hat{\pi}(i)\hat{\pi}(j)} \quad \text{(QAP)}$$

  - QAP was introduced by [Koopmans-Beckmann '57]

COWLES FOUNDATION DISCUSSION PAPER, NO. 4*

Assignment Problems and the Location of Economic Activities**

by

Tjalling C. Koopmans and Martin Beckmann

- It is **NP-hard** to solve or even approximate.
- How much does π_{ML} have in common with π ?

$$\mathrm{overlap}\left(\pi_{\mathrm{ML}}, \pi\right) \triangleq \frac{1}{n} \left| \{ i \in [n] : \pi_{\mathrm{ML}}(i) = \pi(i) \} \right|$$

Fraction of correctly matched vertices

Sharp recovery threshold: dense Erdös-Rényi graphs

Theorem [Wu-Xu-Yu '21]

Suppose $n^{-o(1)} \le p \le 1 - \Omega(1)$. Then

$$\text{If } nps^2 \ge \frac{(2+\epsilon)\log n}{\log\frac{1}{p} - 1 + p} \Rightarrow \text{overlap}\left(\pi_{\text{ML}}, \pi\right) = 1 - o(1) \text{ w.h.p.}$$

$$\text{If } nps^2 \le \frac{(2-\epsilon)\log n}{\log\frac{1}{p} - 1 + p} \Rightarrow \text{overlap}\left(\hat{\pi}, \pi\right) = o(1) \text{ w.h.p. } \forall \hat{\pi}$$

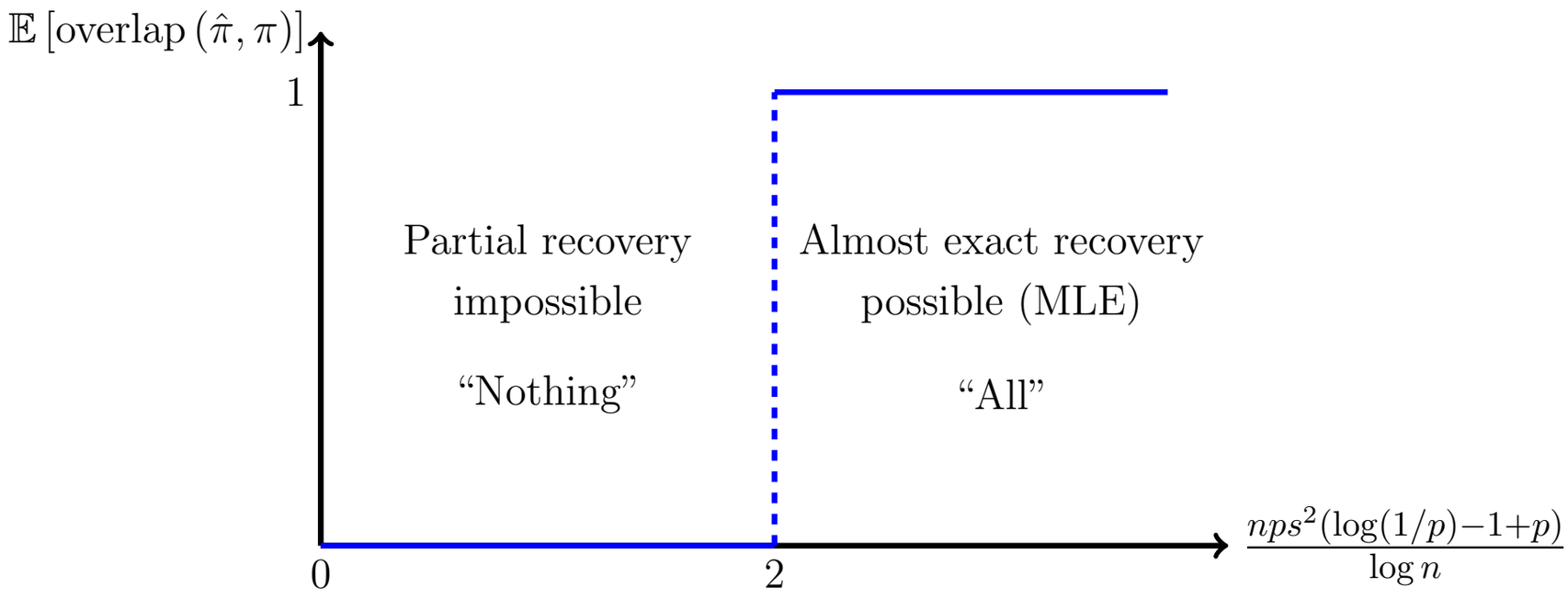
Sharp recovery threshold: dense Erdös-Rényi graphs

Theorem [Wu-Xu-Yu '21]

Suppose $n^{-o(1)} \le p \le 1 - \Omega(1)$. Then

$$\text{If } nps^2 \ge \frac{(2+\epsilon)\log n}{\log\frac{1}{p} - 1 + p} \Rightarrow \text{overlap}\left(\pi_{\text{ML}}, \pi\right) = 1 - o(1) \text{ w.h.p.}$$

$$\text{If } nps^2 \le \frac{(2-\epsilon)\log n}{\log\frac{1}{p} - 1 + p} \Rightarrow \text{overlap}\left(\hat{\pi}, \pi\right) = o(1) \text{ w.h.p. } \forall \hat{\pi}$$



Sharp recovery threshold: dense Erdös-Rényi graphs

Theorem [Wu-Xu-Yu '21]

Suppose $n^{-o(1)} \le p \le 1 - \Omega(1)$. Then

$$\text{If } nps^2 \ge \frac{(2+\epsilon)\log n}{\log\frac{1}{p} - 1 + p} \Rightarrow \text{overlap}\left(\pi_{\text{ML}}, \pi\right) = 1 - o(1) \text{ w.h.p.}$$

$$\text{If } nps^2 \le \frac{(2-\epsilon)\log n}{\log\frac{1}{p} - 1 + p} \Rightarrow \text{overlap}\left(\hat{\pi}, \pi\right) = o(1) \text{ w.h.p. } \forall \hat{\pi}$$

- IT Interpretation of threshold

$$I(\pi; A, B) \approx \binom{n}{2} \times ps^2 \left(\log\frac{1}{p} - 1 + p\right)$$

Mutual info btw two correlated edges

- $H(\pi) \approx n \log n$
- Threshold is at $I(\pi; A, B) \approx H(\pi)$

Sharp recovery threshold: sparse Erdös-Rényi graphs

Theorem [Ding-Du '22]

Suppose $p = n^{-\alpha}$ for $\alpha \in (0,1]$ and $\lambda^* = \gamma^{-1}(1/\alpha)$. Then

If $nps^2 \geq \lambda^* + \epsilon \Rightarrow \exists \hat{\pi}$ s.t. overlap $(\hat{\pi}, \pi) = \Omega(1)$ w.h.p.

If $nps^2 \leq \lambda^* - \epsilon \Rightarrow$ overlap $(\hat{\pi}, \pi) = o(1)$ w.h.p. $\forall \hat{\pi}$

Sharp recovery threshold: sparse Erdös-Rényi graphs

Theorem [Ding-Du '22]

Suppose $p = n^{-\alpha}$ for $\alpha \in (0,1]$ and $\lambda^* = \gamma^{-1}(1/\alpha)$. Then

If $nps^2 \geq \lambda^* + \epsilon \Rightarrow \exists \hat{\pi}$ s.t. overlap $(\hat{\pi}, \pi) = \Omega(1)$ w.h.p.

If $nps^2 \leq \lambda^* - \epsilon \Rightarrow$ overlap $(\hat{\pi}, \pi) = o(1)$ w.h.p. $\forall \hat{\pi}$

- $\gamma : [1,\infty) \to [1,\infty)$ is given by the densest subgraph problem in Erdös-Rényi graphs $\mathcal{G}(n, \lambda/n)$ [Hajek '90, Anantharam-Salez '16]:

$$\max_{\emptyset \neq U \subset [n]} \frac{|E(U)|}{|U|} \to \gamma(\lambda)$$

Sharp recovery threshold: sparse Erdös-Rényi graphs

Theorem [Ding-Du '22]

Suppose $p = n^{-\alpha}$ for $\alpha \in (0,1]$ and $\lambda^* = \gamma^{-1}(1/\alpha)$. Then

If $nps^2 \geq \lambda^* + \epsilon \Rightarrow \exists \hat{\pi}$ s.t. overlap $(\hat{\pi}, \pi) = \Omega(1)$ w.h.p.

If $nps^2 \leq \lambda^* - \epsilon \Rightarrow$ overlap $(\hat{\pi}, \pi) = o(1)$ w.h.p. $\forall \hat{\pi}$

- $\gamma : [1,\infty) \to [1,\infty)$ is given by the densest subgraph problem in Erdös-Rényi graphs $\mathcal{G}(n, \lambda/n)$ [Hajek '90, Anantharam-Salez '16]:

$$\max_{\emptyset \neq U \subset [n]} \frac{|E(U)|}{|U|} \to \gamma(\lambda)$$

- The negative result of $\alpha = 1$ is proved in [Ganassali-Lelarge-Massoulié '21]
- Sharpens our previous threshold $nps^2 = \Theta(1)$ for MLE [Wu-Xu-Yu '21]

Sharp recovery threshold: sparse Erdös-Rényi graphs

Theorem [Ding-Du '22]

Suppose $p = n^{-\alpha}$ for $\alpha \in (0,1]$ and $\lambda^* = \gamma^{-1}(1/\alpha)$. Then

If $nps^2 \geq \lambda^* + \epsilon \Rightarrow \exists \hat{\pi}$ s.t. overlap $(\hat{\pi}, \pi) = \Omega(1)$ w.h.p.

If $nps^2 \leq \lambda^* - \epsilon \Rightarrow$ overlap $(\hat{\pi}, \pi) = o(1)$ w.h.p. $\forall \hat{\pi}$

- $\gamma : [1,\infty) \to [1,\infty)$ is given by the densest subgraph problem in Erdös-Rényi graphs $\mathcal{G}(n, \lambda/n)$ [Hajek '90, Anantharam-Salez '16]:

$$\max_{\emptyset \neq U \subset [n]} \frac{|E(U)|}{|U|} \to \gamma(\lambda)$$

- The negative result of $\alpha = 1$ is proved in [Ganassali-Lelarge-Massoulié '21]
- Sharpens our previous threshold $nps^2 = \Theta(1)$ for MLE [Wu-Xu-Yu '21]
- "All-or-nothing" phenomenon does not exist, as almost exact recovery (overlap $= 1 - o(1)$) requires $nps^2 \to \infty$ [Cullina-Kiyavash-Mittal-Poor '19]

Exact recovery threshold

Theorem [Wu-Xu-Yu '21]

Suppose $p \le 1 - \Omega(1)$. Then

$$\text{If } nps^2 \ge \frac{(1+\epsilon)\log n}{(1-\sqrt{p})^2} \Rightarrow \text{overlap}\left(\pi_{\text{ML}}, \pi\right) = 1 \text{ w.h.p.}$$

$$\text{If } nps^2 \le \frac{(1-\epsilon)\log n}{(1-\sqrt{p})^2} \Rightarrow \text{overlap}\left(\hat{\pi}, \pi\right) \ne 1 \text{ w.h.p} \forall \hat{\pi}$$

Exact recovery threshold

Theorem [Wu-Xu-Yu '21]

Suppose $p \le 1 - \Omega(1)$. Then

$$\text{If } nps^2 \ge \frac{(1+\epsilon)\log n}{(1-\sqrt{p})^2} \Rightarrow \text{overlap}\left(\pi_{\text{ML}}, \pi\right) = 1 \text{ w.h.p.}$$

$$\text{If } nps^2 \le \frac{(1-\epsilon)\log n}{(1-\sqrt{p})^2} \Rightarrow \text{overlap}\left(\hat{\pi}, \pi\right) \ne 1 \text{ w.h.p} \forall \hat{\pi}$$

- $p = o(1)$: reduces to the connectivity threshold of the intersection graph $A^* \wedge B \sim \mathcal{G}(n, ps^2)$ [Cullina-Kiyavash '16 17]

- $p = \Omega(1)$: strictly higher than the connectivity threshold

Summary on information-theoretic thresholds

		Partial recovery & detection	Almost exact recovery	Exact recovery
p	$n^{-o(1)}$	$nps^2 = \frac{2\log n}{\log(1/p)-1+p}$		$\frac{nps^2(1-\sqrt{p})^2}{\log n} = 1$
	$n^{-\alpha}$	$nps^2 = \lambda^*$	$nps^2 = \omega(1)$	
Gaussian		$\frac{n\rho^2}{\log n} = 4$		

Summary on information-theoretic thresholds

		Partial recovery & detection	Almost exact recovery	Exact recovery
p	$n^{-o(1)}$	$nps^2 = \frac{2\log n}{\log(1/p)-1+p}$		$\frac{nps^2(1-\sqrt{p})^2}{\log n} = 1$
	$n^{-\alpha}$	$nps^2 = \lambda^*$	$nps^2 = \omega(1)$	
Gaussian		$\frac{n\rho^2}{\log n} = 4$		

Only a vanishing amount of correlation is needed for recovery information-theoretically!

Summary on information-theoretic thresholds

		Partial recovery & detection	Almost exact recovery	Exact recovery
p	$n^{-o(1)}$	$nps^2 = \frac{2\log n}{\log(1/p)-1+p}$		$\frac{nps^2(1-\sqrt{p})^2}{\log n} = 1$
	$n^{-\alpha}$	$nps^2 = \lambda^*$	$nps^2 = \omega(1)$	
Gaussian		$\frac{n\rho^2}{\log n} = 4$		

Only a vanishing amount of correlation is needed for recovery information-theoretically!

Can we develop a **scalable** algorithm to recover π with a **strong** statistical guarantee?

Outline

- Motivation and problem setup
- Information-theoretic limits
- Efficient algorithms
- Concluding remarks

State of the art (polynomial-time algorithms)

State of the art (polynomial-time algorithms)

ρ \ nq	Sparse graphs $nq = n^{o(1)}$	Dense graphs $nq = n^{\Theta(1)}$
Extremely high correlation $\rho \to 1$		
high correlation ρ close to 1		
Low correlation Constant ρ		

State of the art (polynomial-time algorithms)

ρ \ nq	Sparse graphs $nq = n^{o(1)}$	Dense graphs $nq = n^{\Theta(1)}$
Extremely high correlation $\rho \to 1$		
high correlation ρ close to 1		
Low correlation Constant ρ		

State of the art (polynomial-time algorithms)

ρ \ nq	Sparse graphs $nq = n^{o(1)}$	Dense graphs $nq = n^{\Theta(1)}$
Extremely high correlation $\rho \to 1$		
high correlation ρ close to 1		
Low correlation Constant ρ		

State of the art (polynomial-time algorithms)

ρ \ nq	Sparse graphs $nq = n^{o(1)}$	Dense graphs $nq = n^{\Theta(1)}$
Extremely high correlation $\rho \to 1$		
high correlation ρ close to 1		
Low correlation Constant ρ		

State of the art (polynomial-time algorithms)

ρ \ nq	Sparse graphs $nq = n^{o(1)}$	Dense graphs $nq = n^{\Theta(1)}$
Extremely high correlation $\rho \to 1$		
high correlation ρ close to 1		
Low correlation Constant ρ		

State of the art (polynomial-time algorithms)

ρ \ nq	Sparse graphs $nq = n^{o(1)}$	Dense graphs $nq = n^{\Theta(1)}$
Extremely high correlation $\rho \to 1$		
high correlation ρ close to 1		
Low correlation Constant ρ		

State of the art (polynomial-time algorithms)

ρ \ nq	Sparse graphs $nq = n^{o(1)}$	Dense graphs $nq = n^{\Theta(1)}$
Extremely high correlation $\rho \to 1$	[Babai-Erdős-Selkow '80][Bollobás '82] [Czajka-Pandurangan '07] } $\rho = 1$	
high correlation ρ close to 1		
Low correlation Constant ρ		

State of the art (polynomial-time algorithms)

ρ \ nq	Sparse graphs $nq = n^{o(1)}$	Dense graphs $nq = n^{\Theta(1)}$
Extremely high correlation $\rho \to 1$	[Babai-Erdős-Selkow '80][Bollobás '82] [Czajka-Pandurangan '07] } $\rho = 1$ [Dai-Cullina-Kiyavash-Grossglauser '18] $\rho = 1 - 1/\mathrm{poly}(n)$	
high correlation ρ close to 1		
Low correlation Constant ρ		

State of the art (polynomial-time algorithms)

ρ \ nq	Sparse graphs $nq = n^{o(1)}$	Dense graphs $nq = n^{\Theta(1)}$
Extremely high correlation $\rho \to 1$	[Babai-Erdős-Selkow '80][Bollobás '82] [Czajka-Pandurangan '07] } $\rho = 1$ [Dai-Cullina-Kiyavash-Grossglauser '18] $\rho = 1 - 1/\mathrm{poly}(n)$ [Ding-Ma-Wu-Xu '18] [Fan-Mao-Wu-Xu '19] } $\rho = 1 - 1/\mathrm{polylog}(n)$	
high correlation ρ close to 1		
Low correlation Constant ρ		

State of the art (polynomial-time algorithms)

ρ \ nq	Sparse graphs $nq = n^{o(1)}$	Dense graphs $nq = n^{\Theta(1)}$
Extremely high correlation $\rho \to 1$	[Babai-Erdős-Selkow '80][Bollobás '82] [Czajka-Pandurangan '07] $\rho = 1$ [Dai-Cullina-Kiyavash-Grossglauser '18] $\rho = 1 - 1/\mathrm{poly}(n)$ [Ding-Ma-Wu-Xu '18] [Fan-Mao-Wu-Xu '19] $\rho = 1 - 1/\mathrm{polylog}(n)$ [Mao-Rudelson-Tikhomirov '21] $\rho = 1 - 1/\mathrm{polyloglog}(n)$	
high correlation ρ close to 1		
Low correlation Constant ρ		

State of the art (polynomial-time algorithms)

ρ \ nq	Sparse graphs $nq = n^{o(1)}$	Dense graphs $nq = n^{\Theta(1)}$
Extremely high correlation $\rho \to 1$	[Babai-Erdős-Selkow '80][Bollobás '82] [Czajka-Pandurangan '07] } $\rho = 1$ [Dai-Cullina-Kiyavash-Grossglauser '18] $\rho = 1 - 1/\mathrm{poly}(n)$ [Ding-Ma-Wu-Xu '18] [Fan-Mao-Wu-Xu '19] } $\rho = 1 - 1/\mathrm{polylog}(n)$ [Mao-Rudelson-Tikhomirov '21] $\rho = 1 - 1/\mathrm{polyloglog}(n)$	
high correlation ρ close to 1	[Ganassali-Massoulié '20] [Ganassali-Massoulié-Lelarge'21] [Mao-Rudelson-Tikhomirov '21]	} $\rho = 1 - c$
Low correlation Constant ρ		

State of the art (polynomial-time algorithms)

ρ \ nq	Sparse graphs $nq = n^{o(1)}$	Dense graphs $nq = n^{\Theta(1)}$
Extremely high correlation $\rho \to 1$	[Babai-Erdős-Selkow '80][Bollobás '82] [Czajka-Pandurangan '07] } $\rho = 1$ [Dai-Cullina-Kiyavash-Grossglauser '18] $\rho = 1 - 1/\mathrm{poly}(n)$ [Ding-Ma-Wu-Xu '18] [Fan-Mao-Wu-Xu '19] } $\rho = 1 - 1/\mathrm{polylog}(n)$ [Mao-Rudelson-Tikhomirov '21] $\rho = 1 - 1/\mathrm{polyloglog}(n)$	
high correlation ρ close to 1	[Ganassali-Massoulié '20] [Ganassali-Massoulié-Lelarge'21] [Mao-Rudelson-Tikhomirov '21]	} $\rho = 1 - c$
Low correlation Constant ρ		

$n^{\Theta(\log n)}$-time recovery when $\rho = o(1)$ [Barak-Chou-Lei-Schramm-Sheng '19]

State of the art (polynomial-time algorithms)

ρ \ nq	Sparse graphs $nq = n^{o(1)}$	Dense graphs $nq = n^{\Theta(1)}$
Extremely high correlation $\rho \to 1$	[Babai-Erdős-Selkow '80][Bollobás '82] [Czajka-Pandurangan '07] } $\rho = 1$ [Dai-Cullina-Kiyavash-Grossglauser '18] $\rho = 1 - 1/\mathrm{poly}(n)$ [Ding-Ma-Wu-Xu '18] [Fan-Mao-Wu-Xu '19] } $\rho = 1 - 1/\mathrm{polylog}(n)$ [Mao-Rudelson-Tikhomirov '21] $\rho = 1 - 1/\mathrm{polyloglog}(n)$	
high correlation ρ close to 1	[Ganassali-Massoulié '20] [Ganassali-Massoulié-Lelarge'21] [Mao-Rudelson-Tikhomirov '21]	} $\rho = 1 - c$
Low correlation Constant ρ	?	

State of the art (polynomial-time algorithms)

ρ \ nq	Sparse graphs $nq = n^{o(1)}$	Dense graphs $nq = n^{\Theta(1)}$
Extremely high correlation $\rho \to 1$	$\rho^2 > \alpha \approx 0.338$ [Mao-Wu-Xu-Yu '22]	
high correlation ρ close to 1	$\alpha \triangleq$ Otter's tree counting constant	
Low correlation Constant ρ		

State of the art (polynomial-time algorithms)

ρ \ nq	Sparse graphs $nq = n^{o(1)}$	Dense graphs $nq = n^{\Theta(1)}$
Extremely high correlation $\rho \to 1$	$\rho^2 > \alpha \approx 0.338$ [Mao-Wu-Xu-Yu '22]	
high correlation ρ close to 1	$\alpha \triangleq$ Otter's tree counting constant	
Low correlation Constant ρ		

The number of unlabeled trees with N edges $\approx \alpha^{-N}$ [Otter '1948]

ANNALS OF MATHEMATICS
Vol. 49, No. 3, July, 1948

THE NUMBER OF TREES

RICHARD OTTER[1]

(Received June 10, 1947)

State of the art (polynomial-time algorithms)

ρ \ nq	Sparse graphs $nq = n^{o(1)}$	Dense graphs $nq = n^{\Theta(1)}$
Extremely high correlation $\rho \to 1$		
high correlation ρ close to 1		
Low correlation Constant ρ	$\rho^2 > \alpha \approx 0.338$ [Mao-Wu-Xu-Yu '22] $\alpha \triangleq$ Otter's tree counting constant	

A different (local) algorithm is shown to achieve partial recovery in the sparse regime when $\rho > \sqrt{\alpha}$ [Ganassali-Massoulié-Semerjian '22]

State of the art (polynomial-time algorithms)

ρ \ nq	Sparse graphs $nq = n^{o(1)}$	Dense graphs $nq = n^{\Theta(1)}$
Extremely high correlation $\rho \to 1$		[Ding-Li '23] improves to $\rho =$ arbitrary constant
high correlation ρ close to 1		
Low correlation Constant ρ	$\rho^2 > \alpha \approx 0.338$ [Mao-Wu-Xu-Yu '22] $\alpha \triangleq$ Otter's tree counting constant	

Our results

Our results

Theorem [Mao-Wu-Xu-Yu '22]

When $\rho^2 > \alpha \approx 0.338$ ($\alpha \triangleq$ Otter's tree counting constant), as $n \to \infty$, our polynomial-time matching algorithm with probability $1 - o(1)$ achieves:

- If $nq \geq C$, partial recovery (correctly match a positive constant fraction of vertices)

Our results

Theorem [Mao-Wu-Xu-Yu '22]

When $\rho^2 > \alpha \approx 0.338$ ($\alpha \triangleq$ Otter's tree counting constant), as $n \to \infty$, our polynomial-time matching algorithm with probability $1 - o(1)$ achieves:

- If $nq \geq C$, partial recovery (correctly match a positive constant fraction of vertices)
- If $nq = \omega(1)$, almost exact recovery (correctly match $1 - o(1)$ fraction of vertices)

Our results

Theorem [Mao-Wu-Xu-Yu '22]

When $\rho^2 > \alpha \approx 0.338$ ($\alpha \triangleq$ Otter's tree counting constant), as $n \to \infty$, our polynomial-time matching algorithm with probability $1 - o(1)$ achieves:

- If $nq \geq C$, partial recovery (correctly match a positive constant fraction of vertices)
- If $nq = \omega(1)$, almost exact recovery (correctly match $1 - o(1)$ fraction of vertices)
- If $nq(q + \rho(1-q)) \geq (1+\epsilon)\log n$, exact recovery (correctly match all vertices)

Our results

Theorem [Mao-Wu-Xu-Yu '22]

When $\rho^2 > \alpha \approx 0.338$ ($\alpha \triangleq$ Otter's tree counting constant), as $n \to \infty$, our polynomial-time matching algorithm with probability $1 - o(1)$ achieves:

- If $nq \geq C$, partial recovery (correctly match a positive constant fraction of vertices)
- If $nq = \omega(1)$, almost exact recovery (correctly match $1 - o(1)$ fraction of vertices)
- If $nq(q + \rho(1-q)) \geq (1+\epsilon)\log n$, exact recovery (correctly match all vertices)

- No mismatching error;

Our results

Theorem [Mao-Wu-Xu-Yu '22]

When $\rho^2 > \alpha \approx 0.338$ ($\alpha \triangleq$ Otter's tree counting constant), as $n \to \infty$, our polynomial-time matching algorithm with probability $1 - o(1)$ achieves:

- If $nq \geq C$, partial recovery (correctly match a positive constant fraction of vertices)
- If $nq = \omega(1)$, almost exact recovery (correctly match $1 - o(1)$ fraction of vertices)
- If $nq(q + \rho(1-q)) \geq (1+\epsilon)\log n$, exact recovery (correctly match all vertices)

- No mismatching error;
- The intersection graph between A and B under the hidden node correspondence π $\sim \mathcal{G}\left(n, q(q+\rho(1-q))\right)$;

Our results

Theorem [Mao-Wu-Xu-Yu '22]

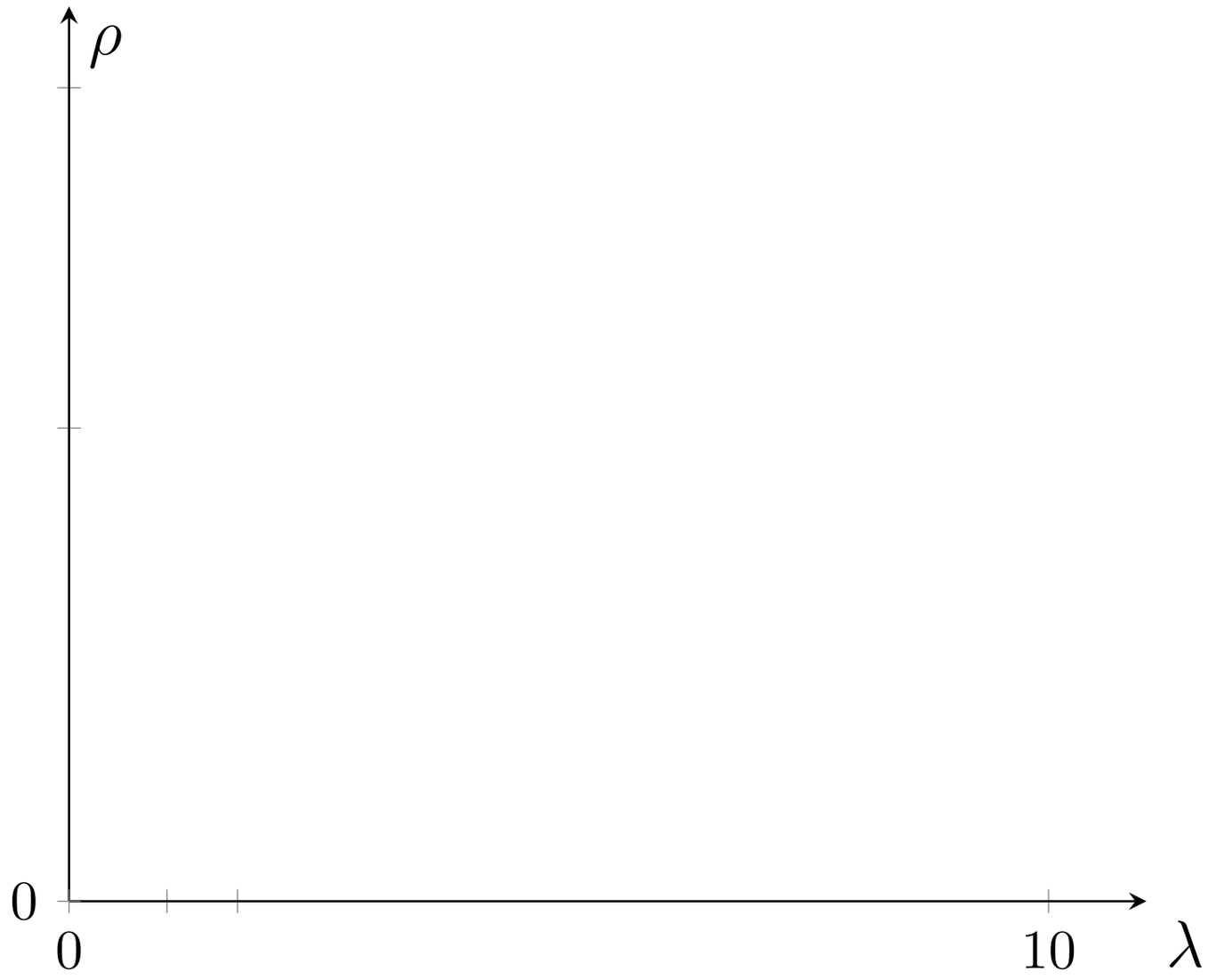
When $\rho^2 > \alpha \approx 0.338$ ($\alpha \triangleq$ Otter's tree counting constant), as $n \to \infty$, our polynomial-time matching algorithm with probability $1 - o(1)$ achieves:

- If $nq \geq C$, partial recovery (correctly match a positive constant fraction of vertices)
- If $nq = \omega(1)$, almost exact recovery (correctly match $1 - o(1)$ fraction of vertices)
- If $nq(q + \rho(1-q)) \geq (1+\epsilon)\log n$, exact recovery (correctly match all vertices)

- No mismatching error;
- The intersection graph between A and B under the hidden node correspondence π $\sim \mathcal{G}\left(n, q(q+\rho(1-q))\right)$;
- $nq(q + \rho(1-q)) \geq (1+\epsilon)\log n$ is the connectivity threshold and information-theoretically necessary for exact recovery;

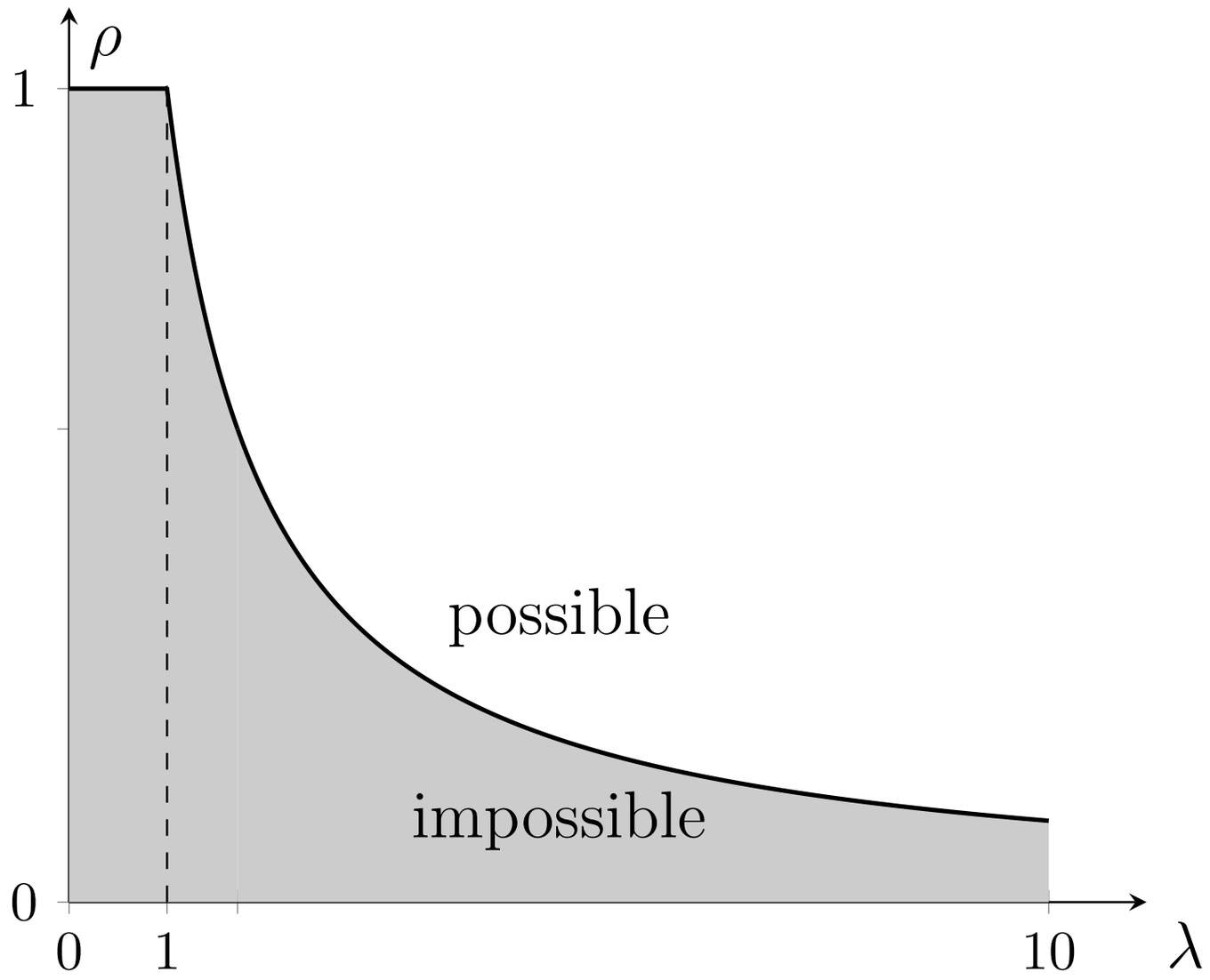
Phase transition diagram for exact recovery

Let's focus on the regime when $nq = \lambda \log n$, where λ is some constant.



Phase transition diagram for exact recovery

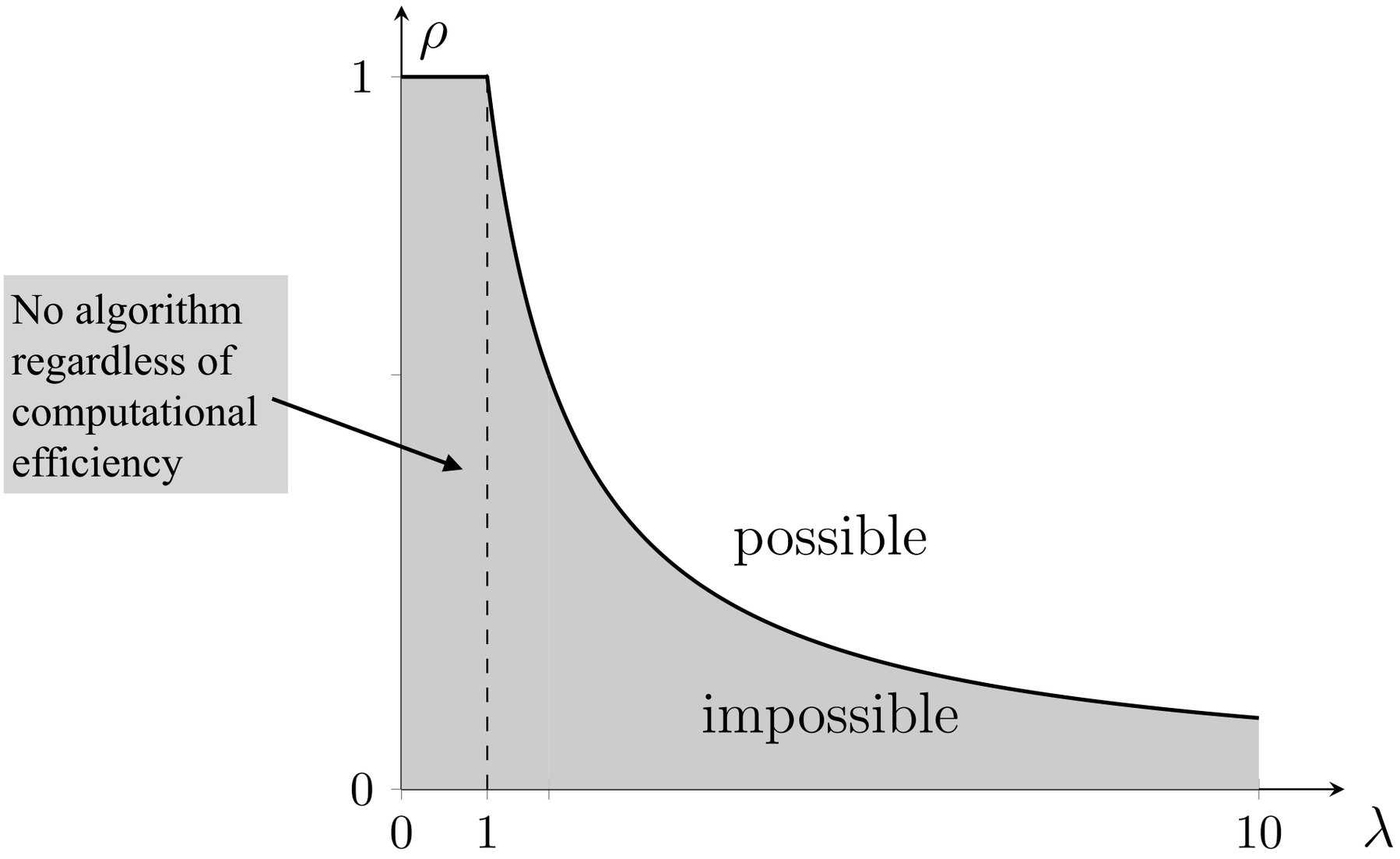
Let's focus on the regime when $nq = \lambda \log n$, where λ is some constant.



The curve is the information theoretical limit where $\rho = \min\left\{1, \frac{1}{\lambda}\right\}$ [Cullina-Kiyavash '16,'17][Wu-Xu-Yu '21].

Phase transition diagram for exact recovery

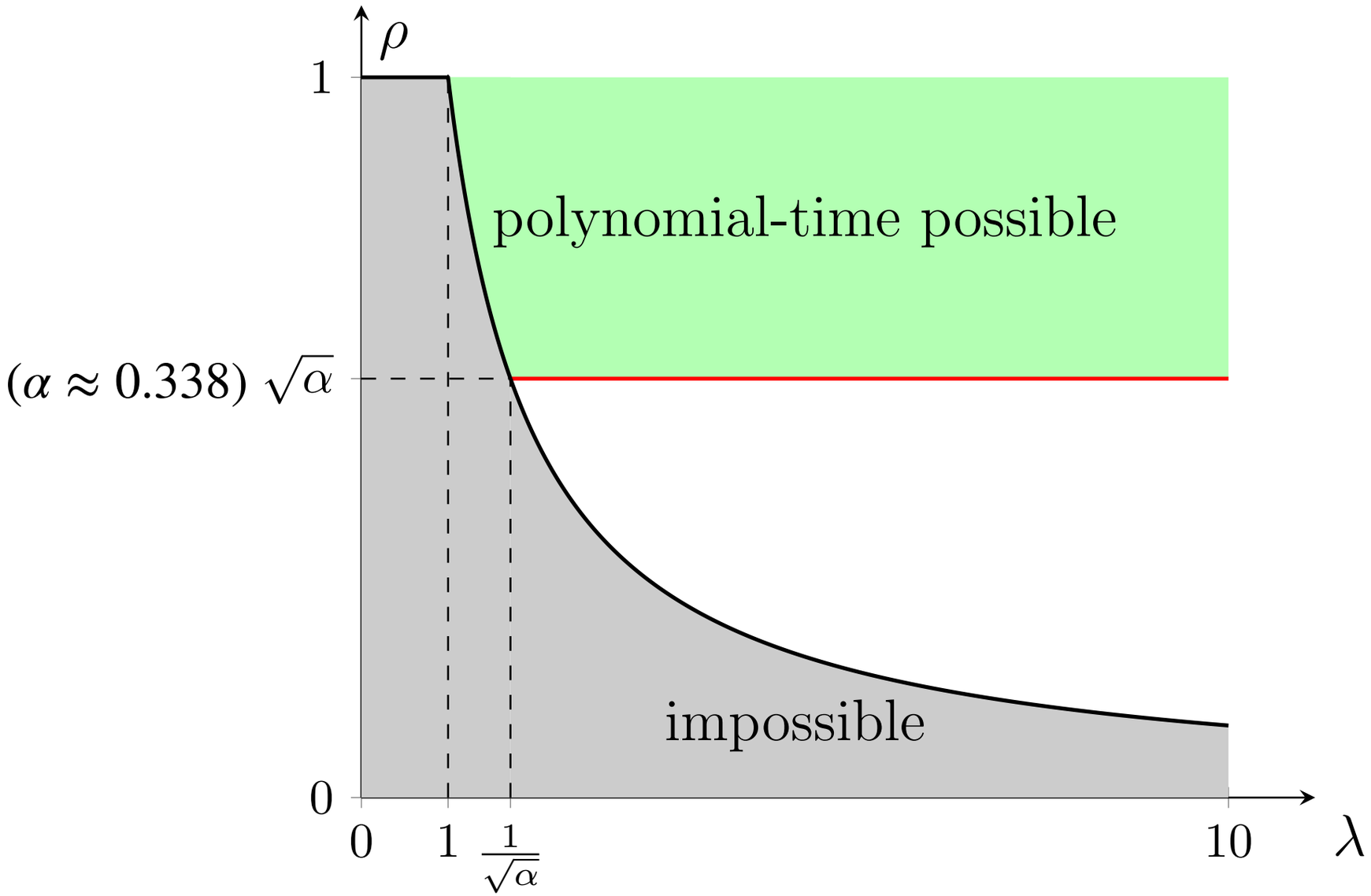
Let's focus on the regime when $nq = \lambda \log n$, where λ is some constant.



The curve is the information theoretical limit where $\rho = \min\left\{1, \frac{1}{\lambda}\right\}$ [Cullina-Kiyavash '16,'17][Wu-Xu-Yu '21].

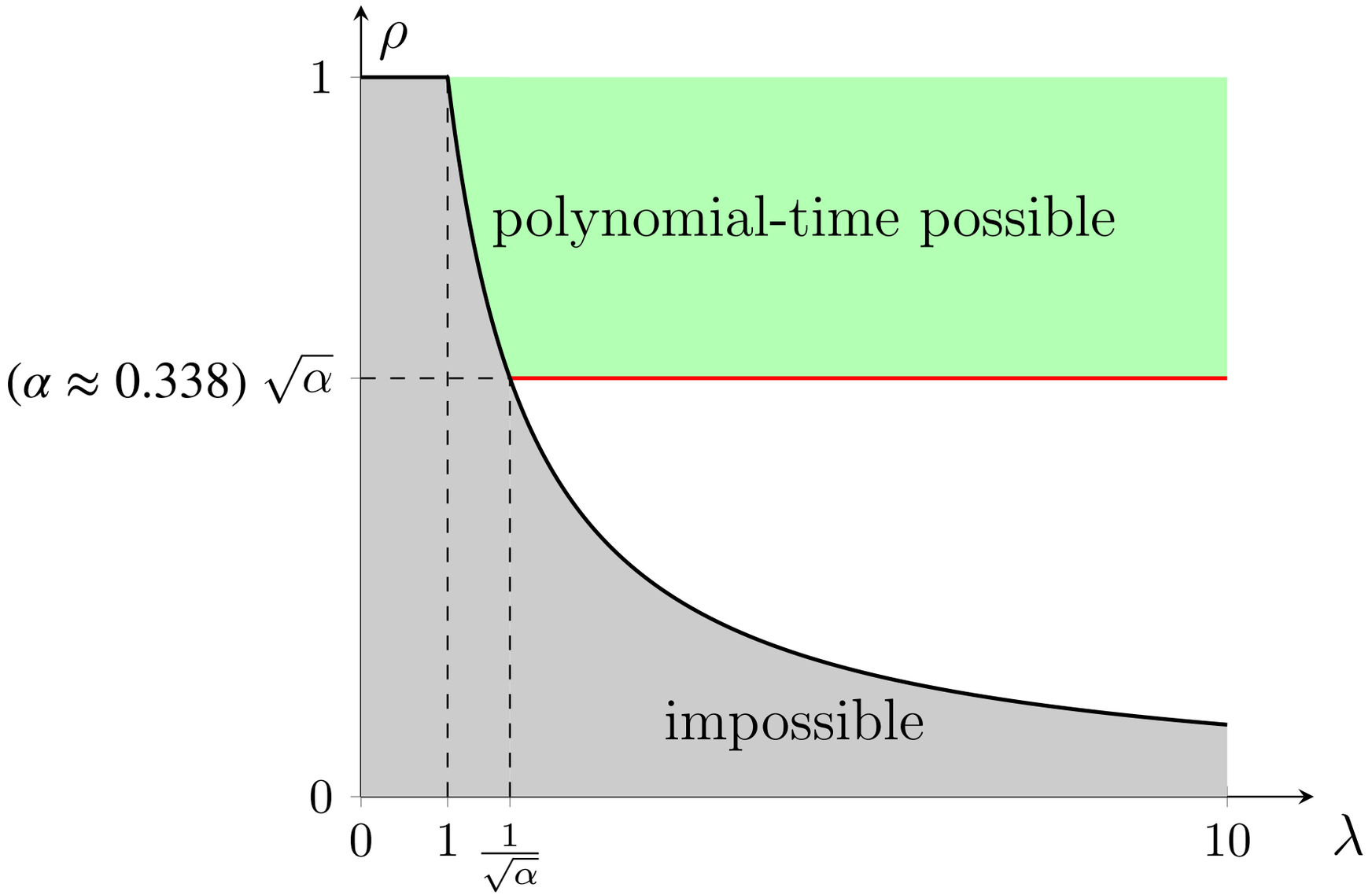
Phase transition diagram for exact recovery

Let's focus on the regime when $nq = \lambda \log n$, where λ is some constant.



Phase transition diagram for exact recovery

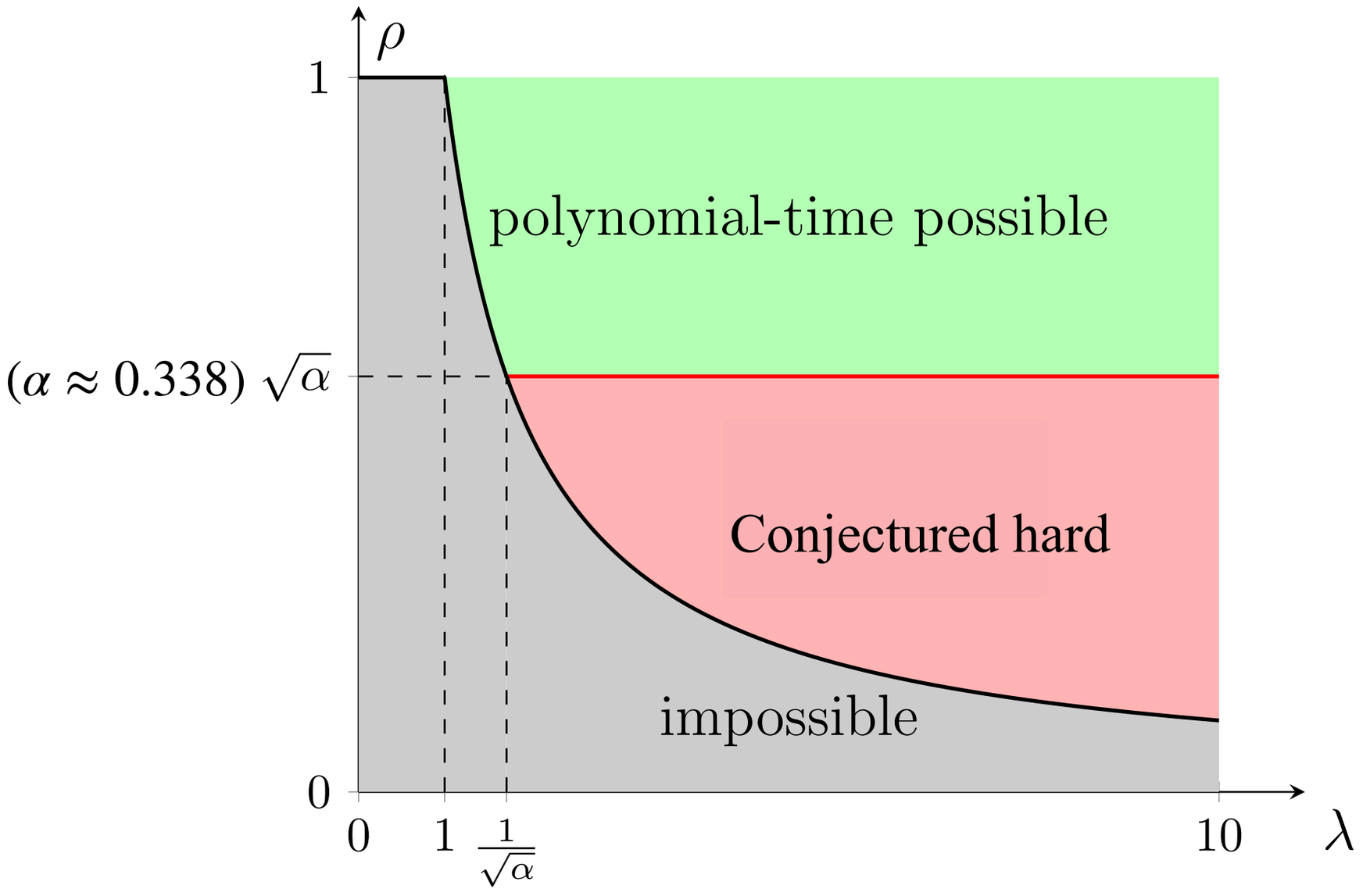
Let's focus on the regime when $nq = \lambda \log n$, where λ is some constant.



Our condition coincides with the information-theoretic limit when $1 < \lambda < \dfrac{1}{\sqrt{\alpha}}$.

Phase transition diagram for exact recovery

Let's focus on the regime when $nq = \lambda \log n$, where λ is some constant.

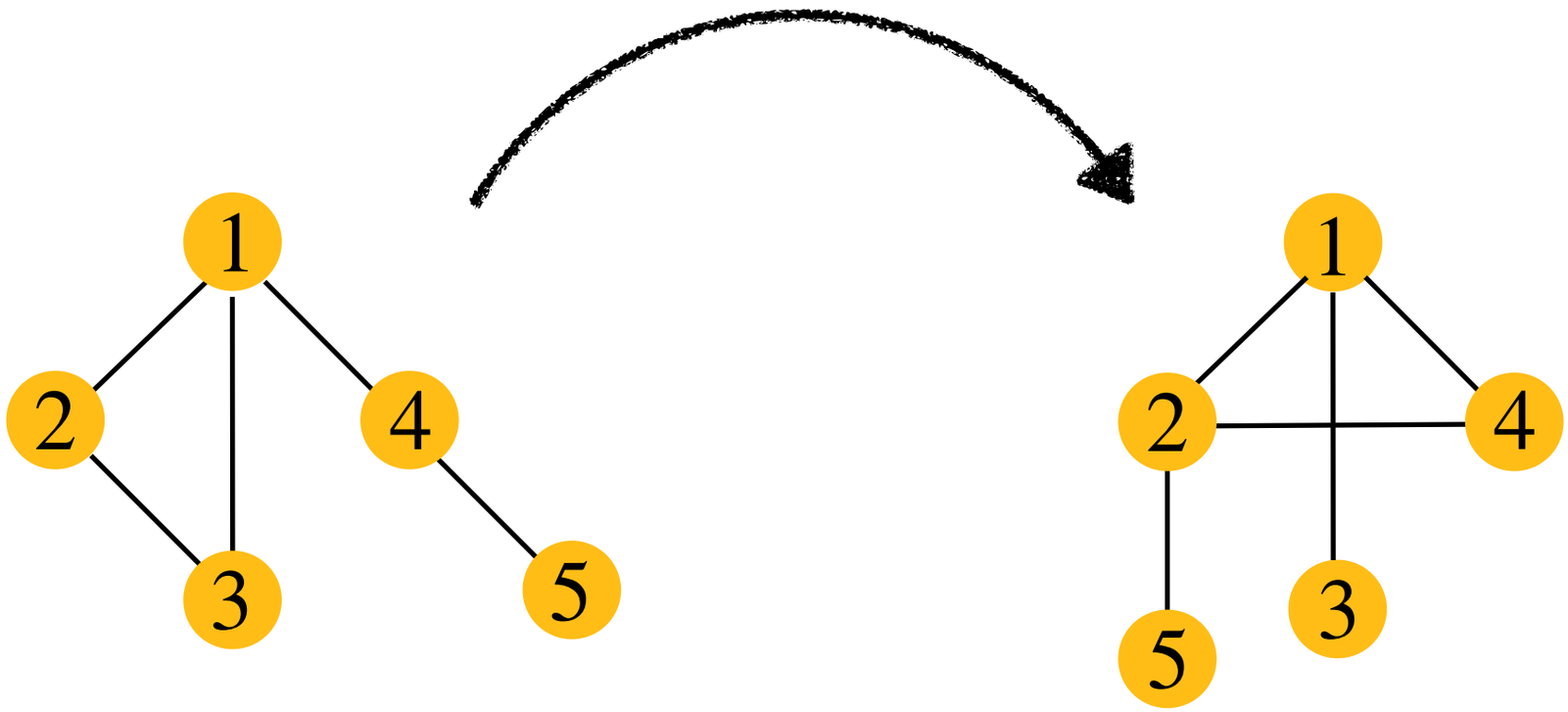


Low-degree polynomial estimators fail when $\rho < \sqrt{\alpha}$ [Ding-Du-Li '23];

Local algorithms fail when $\rho < \sqrt{\alpha}$ in sparse regime [Ganassali-Massoulié-Semerjian '22]

Meta algorithm

Algorithm



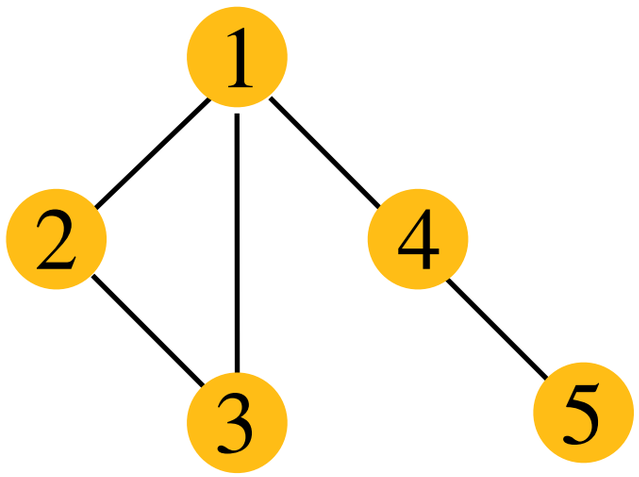
A *B*

Meta algorithm

- Step 1: signature embedding

  Based on the structure of A:

  Construct a **vertex signature** (number or vector) for each vertex in A.



A

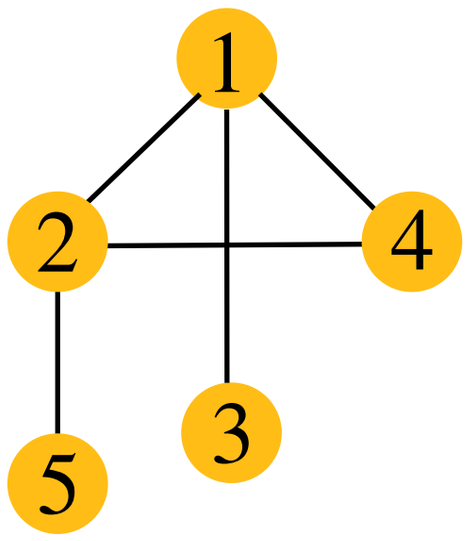
Vertex in A	Vertex signature
1	s_1
2	s_2
3	s_3
4	s_4
5	s_5

Meta algorithm

- Step 1: signature embedding

  Based on the structure of B:

  Construct a **vertex signature** (number or vector) for each vertex in B.



$B \sim \mathcal{G}(n, q)$

Vertex in B	Vertex signature
1	t_1
2	t_2
3	t_3
4	t_4
5	t_5

Meta algorithm

Vertex in A	Vertex signature
1	s_1
2	s_2
3	s_3
4	s_4
5	s_5

Vertex in B	Vertex signature
1	t_1
2	t_2
3	t_3
4	t_4
5	t_5

- Step 2: Similarity score

  For any vertex pair of i in A and j in B, compute **similarity score Φ_{ij}** based on s_i and t_j.

Meta algorithm

Vertex in A	Vertex signature
1	s_1
2	s_2
3	s_3
4	s_4
5	s_5

Vertex in B	Vertex signature
1	t_1
2	t_2
3	t_3
4	t_4
5	t_5

- Step 2: Similarity score

  For any vertex pair of i in A and j in B, compute **similarity score Φ_{ij}** based on s_i and t_j.

  We want

  - For $j = \pi(i)$ (true pair), s_i is close to $t_j \Longrightarrow \Phi_{ij}$ is relatively large;
  - For $j \neq \pi(i)$ (fake pair), s_i is far from $t_j \Longrightarrow \Phi_{ij}$ is relatively small;

Meta algorithm

Vertex in A	Vertex signature
1	s_1
2	s_2
3	s_3
4	s_4
5	s_5

Vertex in B	Vertex signature
1	t_1
2	t_2
3	t_3
4	t_4
5	t_5

- Step 2: Similarity score

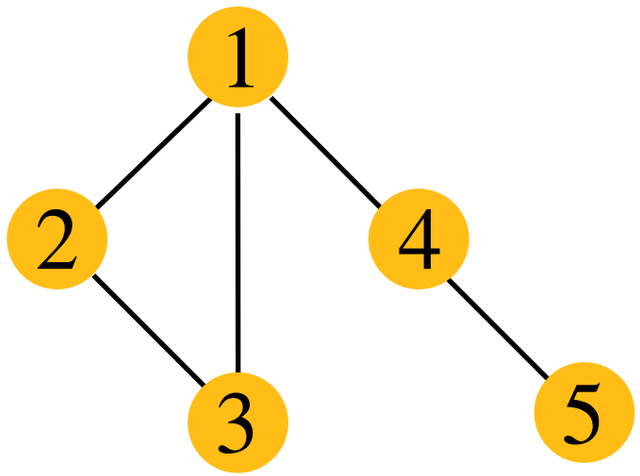
  For any vertex pair of i in A and j in B, compute **similarity score Φ_{ij}** based on s_i and t_j.

  We want

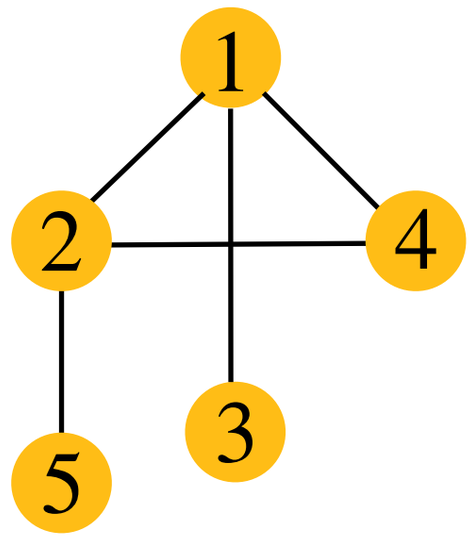
  - For $j = \pi(i)$ (true pair), s_i is close to $t_j \Longrightarrow \Phi_{ij}$ is relatively large;
  - For $j \neq \pi(i)$ (fake pair), s_i is far from $t_j \Longrightarrow \Phi_{ij}$ is relatively small;

How to **construct the vertex signature**?

How to construct the vertex signature?

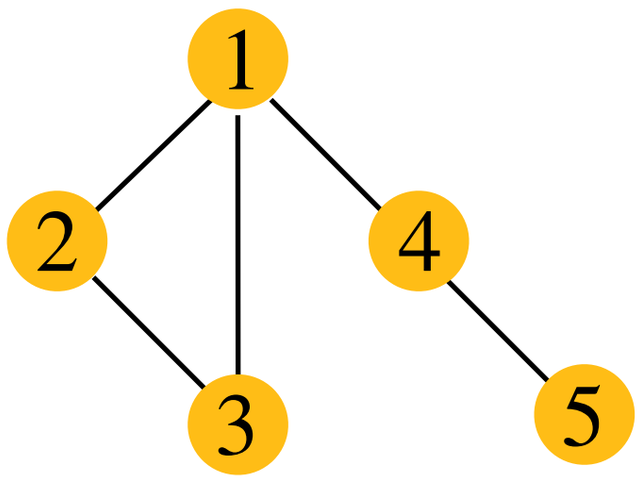


A

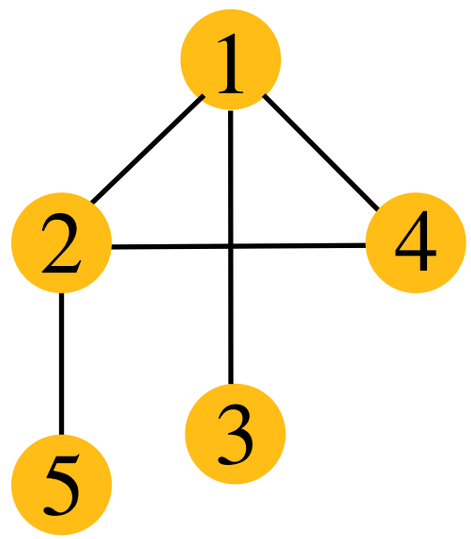


B

How to construct the vertex signature?



A

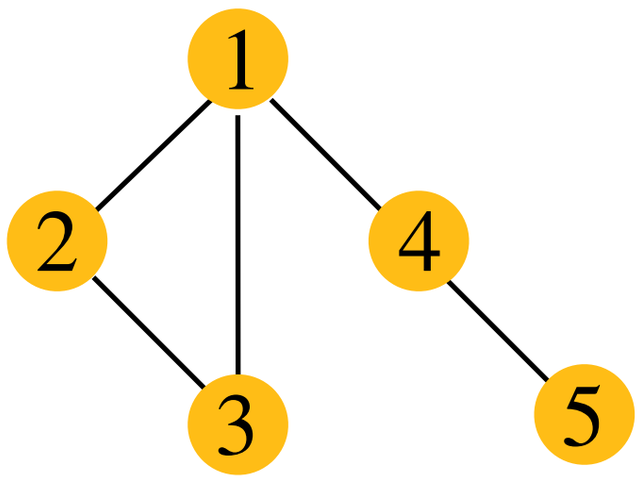


B

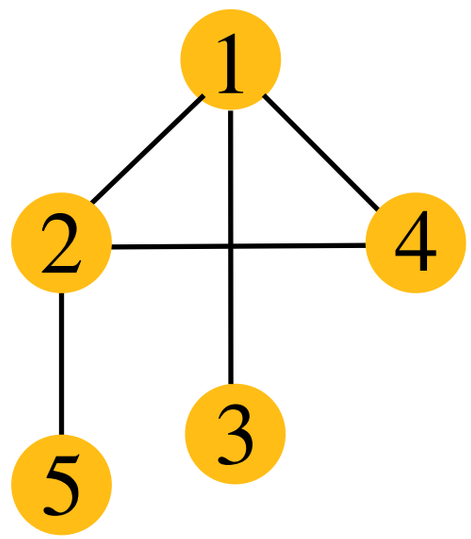
Examples of vertex signature:

- Degree

How to construct the vertex signature?



A

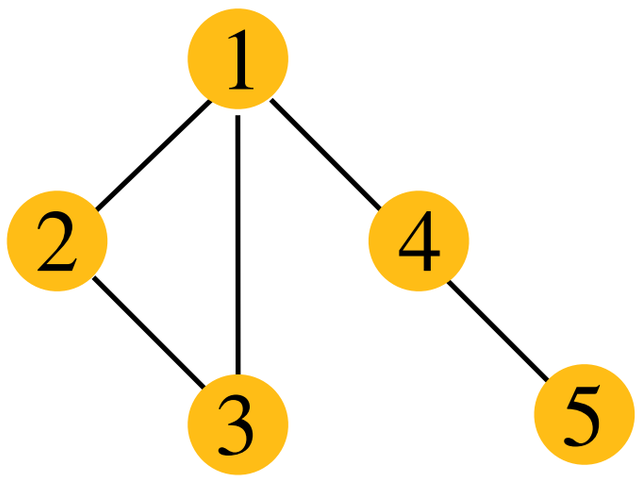


B

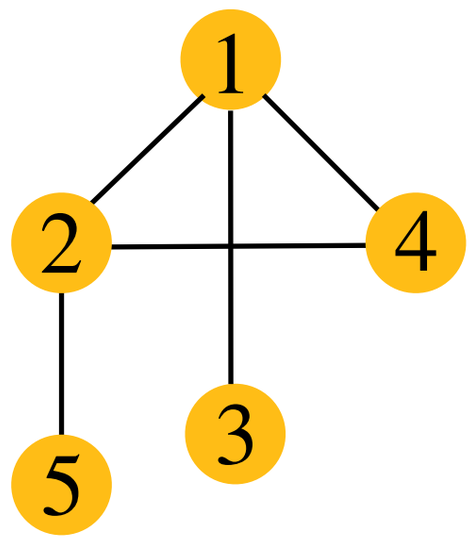
Examples of vertex signature:

- Degree
- Degrees of neighbors: only works for $\rho = 1 - 1/\text{polylog}(n)$ [Ding-Ma-Wu-Xu '18]

How to construct the vertex signature?



A

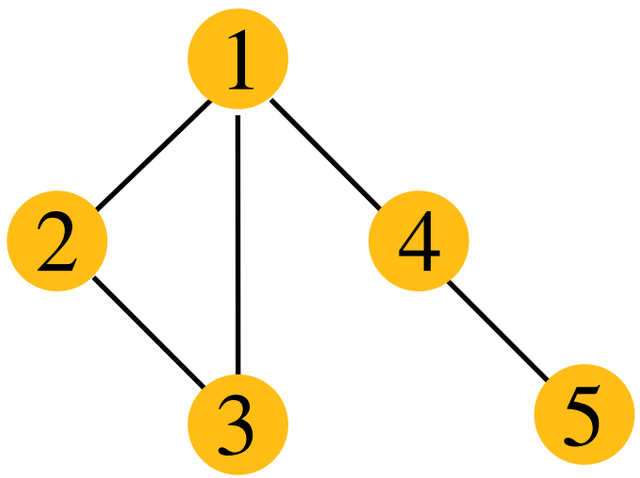


B

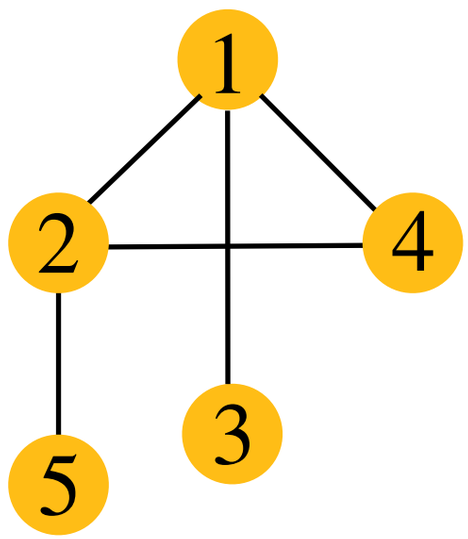
Examples of vertex signature:

- Degree
- Degrees of neighbors: only works for $\rho = 1 - 1/\mathrm{polylog}(n)$ [Ding-Ma-Wu-Xu '18]
- The local tree structure: only works for sparse graphs [Mao-Rudelson-Tikhomirov '21] [Ganassali-Massoulié '20][Ganassali-Massoulié-Lelarge '21]

How to construct the vertex signature?



A



B

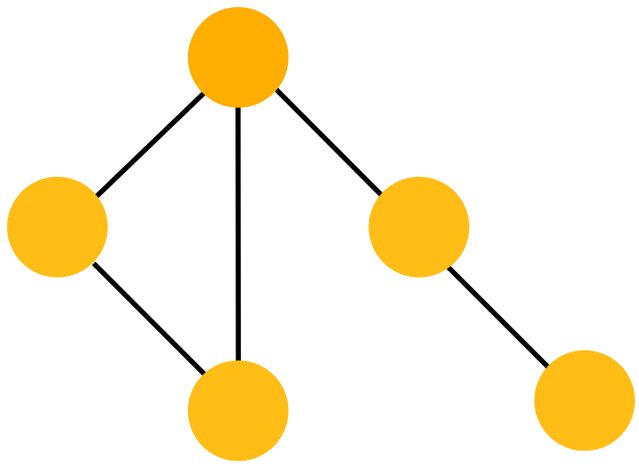
Examples of vertex signature:

- Degree
- Degrees of neighbors: only works for $\rho = 1 - 1/\mathrm{polylog}(n)$ [Ding-Ma-Wu-Xu '18]
- The local tree structure: only works for sparse graphs [Mao-Rudelson-Tikhomirov '21] [Ganassali-Massoulié '20][Ganassali-Massoulié-Lelarge '21]

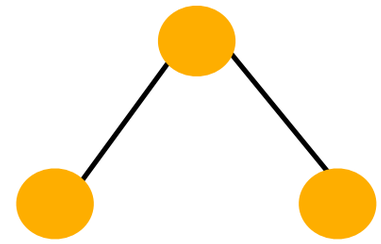
The above vertex signatures are either sensitive to noise or only work in sparse regime.

Our idea: subgraph count

Subgraph count



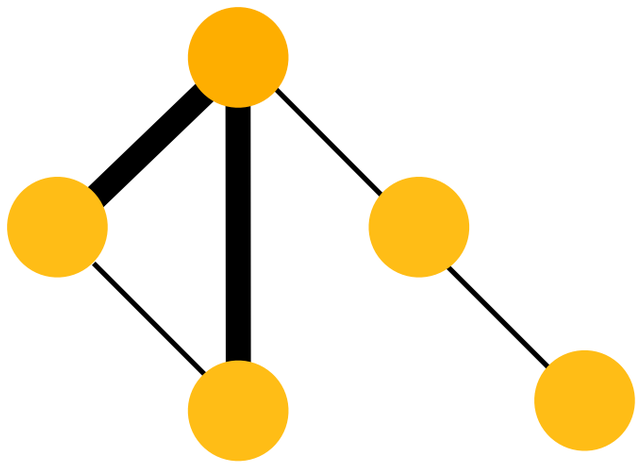
A



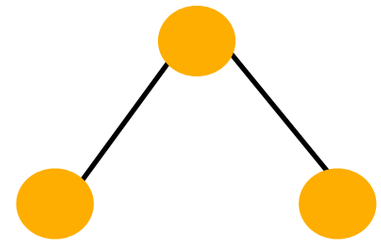
subgraph H

- # copies of H in graph A

Subgraph count



A

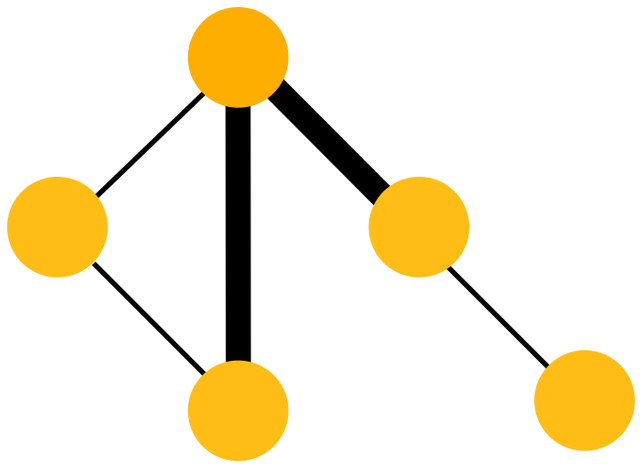


subgraph H

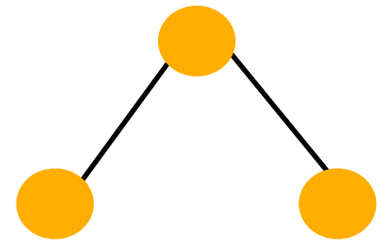
(6 copies of H appear in A)

- # copies of H in graph A

Subgraph count



A

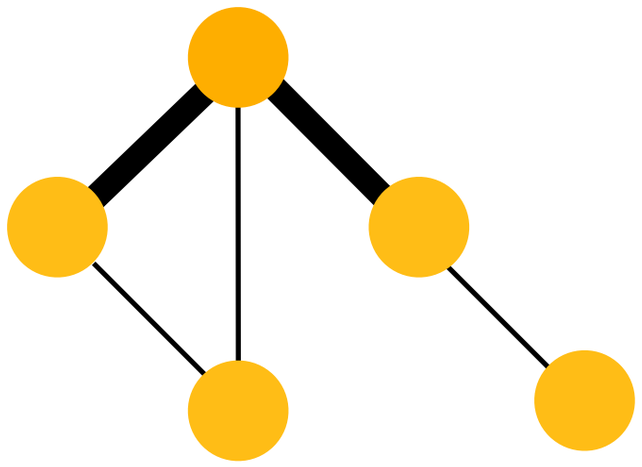


subgraph H

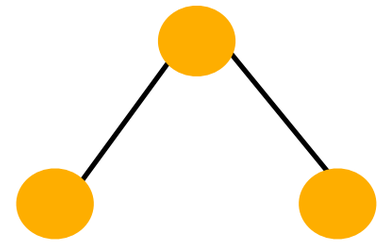
(6 copies of H appear in A)

- # copies of H in graph A

Subgraph count



A

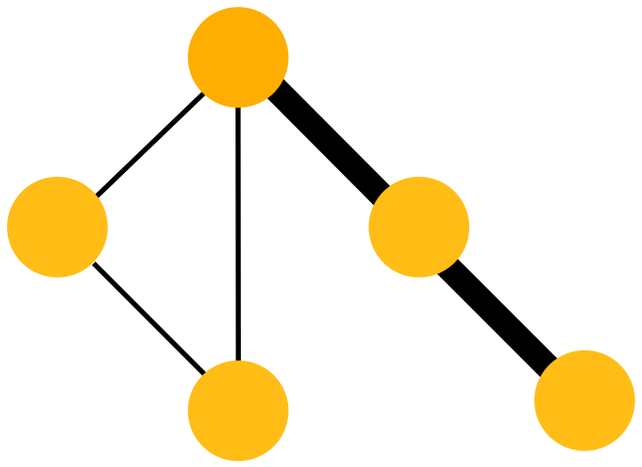


subgraph H

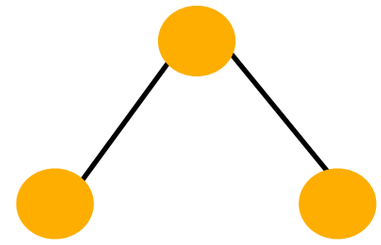
(6 copies of H appear in A)

- # copies of H in graph A

Subgraph count



A

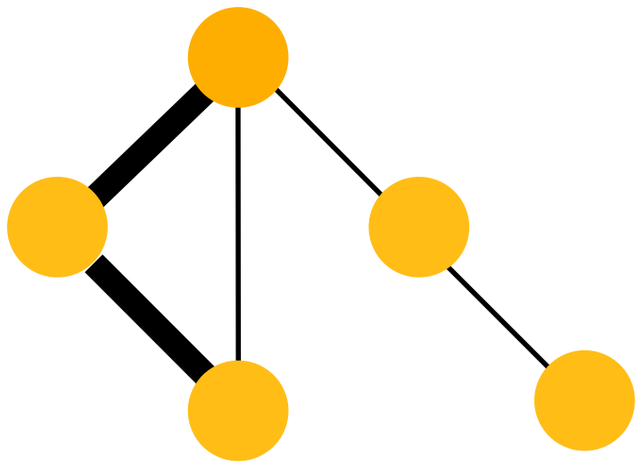


subgraph H

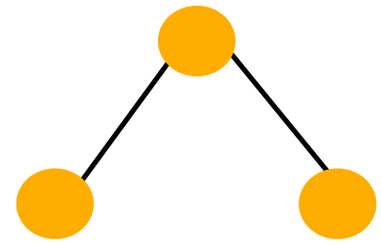
(6 copies of H appear in A)

- # copies of H in graph A

Subgraph count



A

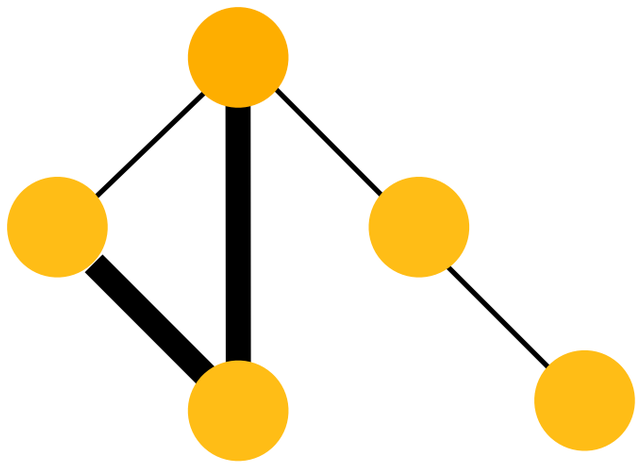


subgraph H

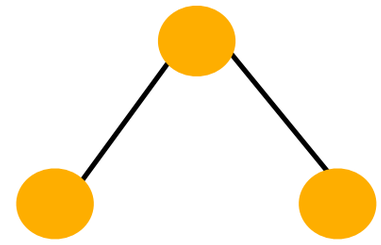
(6 copies of H appear in A)

- # copies of H in graph A

Subgraph count



A

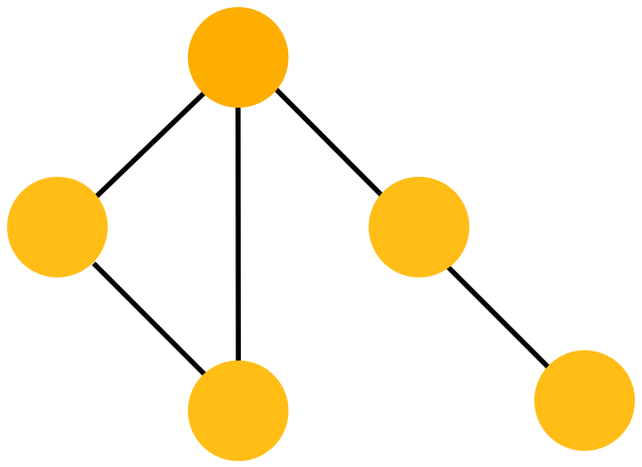


subgraph H

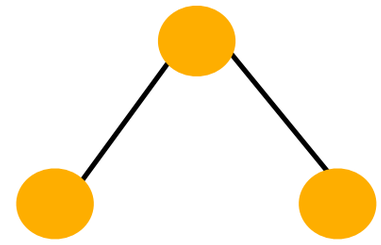
(6 copies of H appear in A)

- # copies of H in graph A

Subgraph count



A

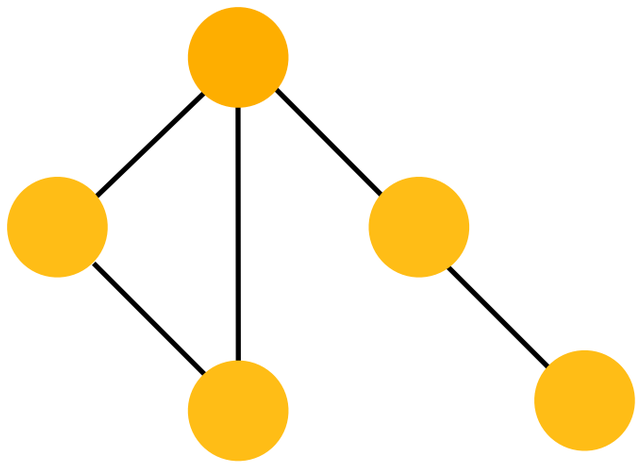


subgraph H

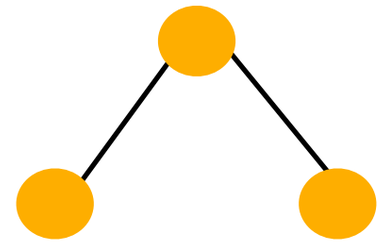
(6 copies of H appear in A)

- # copies of H in graph $A \Longrightarrow$ capture some graph information;

Subgraph count



A

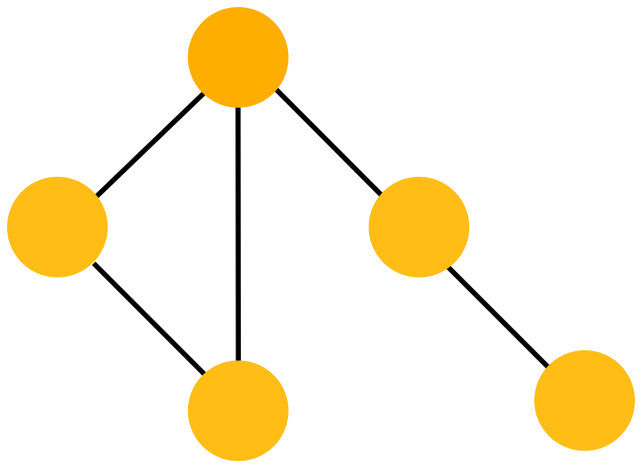


subgraph H

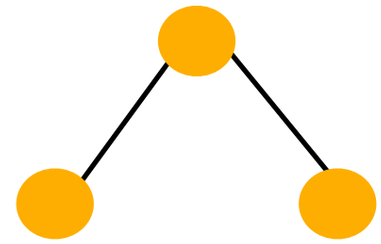
(6 copies of H appear in A)

- # copies of H in graph $A \Longrightarrow$ capture some graph information;
- It is very popular in both theory and practice (motif counting [Milo-Shen-Orr-Itzkovitz-Kashtan '02]);

Subgraph count



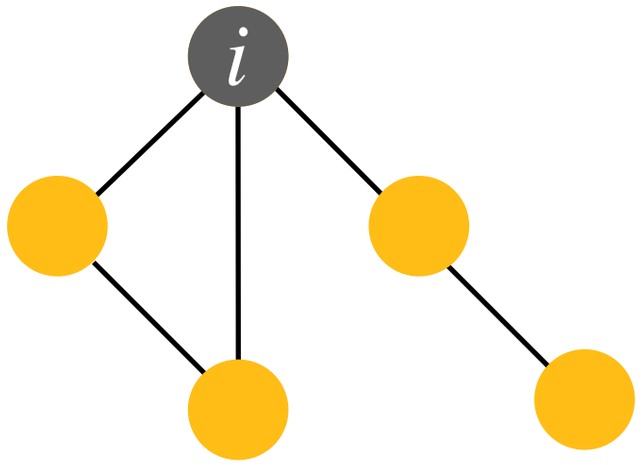
A



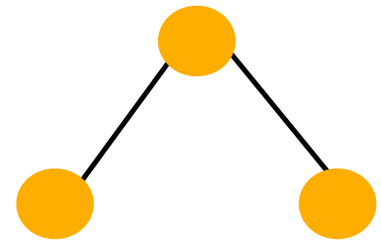
subgraph H

- # copies of H in graph $A \Longrightarrow$ capture some graph information;
- It is very popular in both theory and practice (motif counting [Milo-Shen-Orr-Itzkovitz-Kashtan '02]);
- Has been applied to graph matching [Barak-Chou-Lei-Schramm-Sheng '19]

Subgraph count



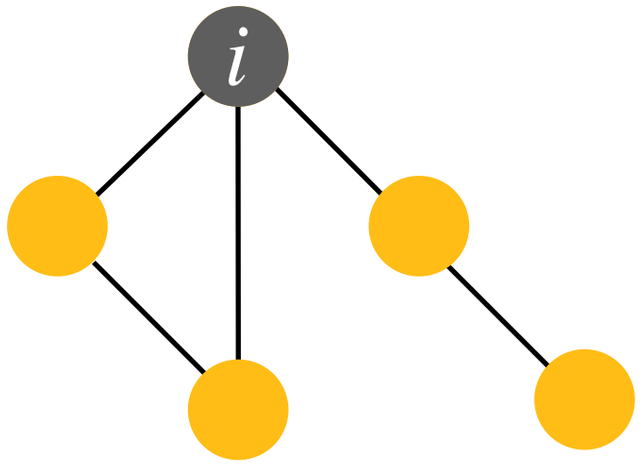
A



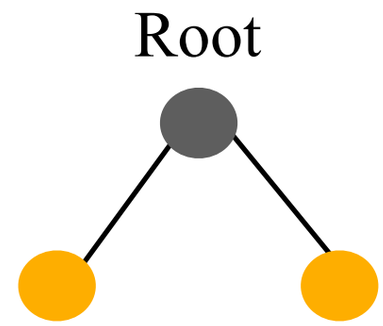
subgraph H

- # copies of H in graph $A \Longrightarrow$ capture some graph information;
- It is very popular in both theory and practice (motif counting [Milo-Shen-Orr-Itzkovitz-Kashtan '02]);
- Has been applied to graph matching [Barak-Chou-Lei-Schramm-Sheng '19]
- How to capture the vertex information of i?

Rooted subgraph count



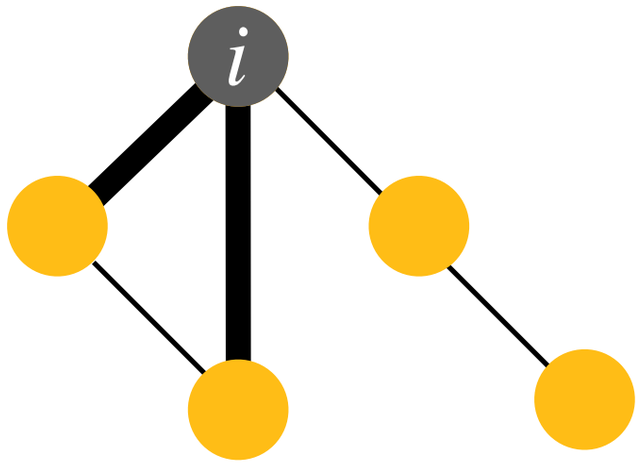
A



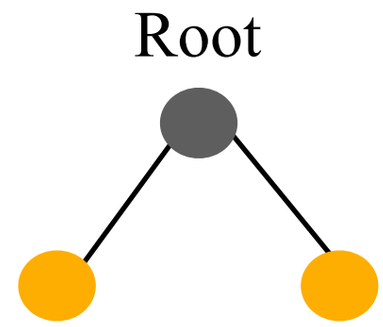
Rooted subgraph H

- # copies of H rooted at i in graph A, denoted as $W_{i,H}(A)$;

Rooted subgraph count



A

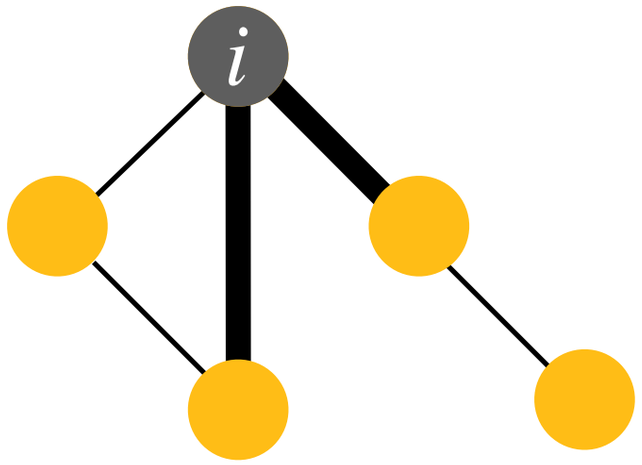


Rooted subgraph H

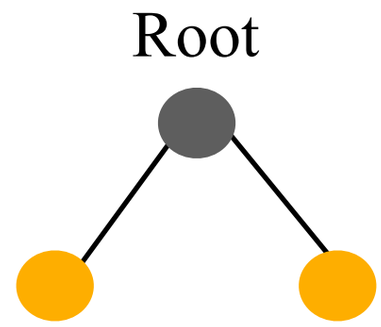
(3 copies of H rooted at node i appear in A)

- # copies of H rooted at i in graph A, denoted as $W_{i,H}(A)$;

Rooted subgraph count



A

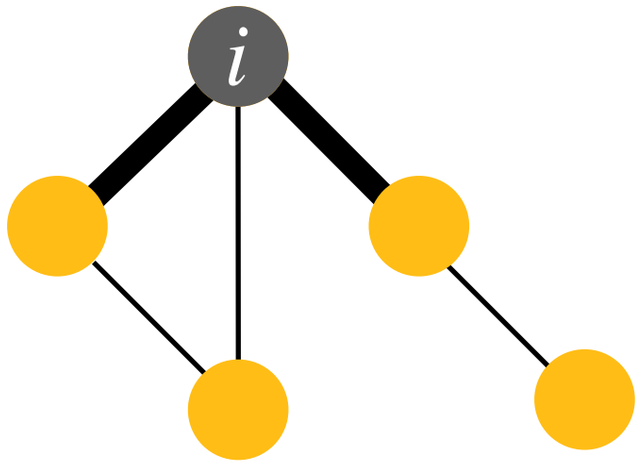


Rooted subgraph H

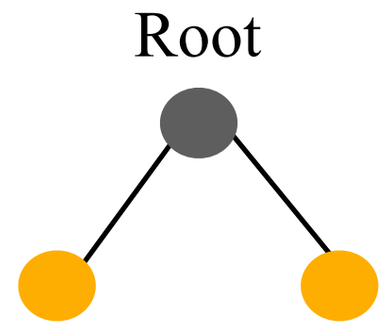
(3 copies of H rooted at node i appear in A)

- # copies of H rooted at i in graph A, denoted as $W_{i,H}(A)$;

Rooted subgraph count



A

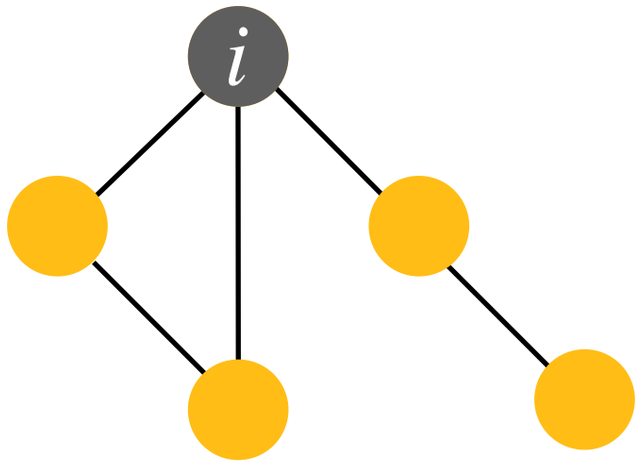


Rooted subgraph H

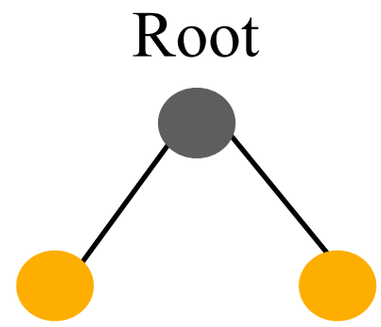
(3 copies of H rooted at node i appear in A)

- # copies of H rooted at i in graph A, denoted as $W_{i,H}(A)$;

Rooted subgraph count



A



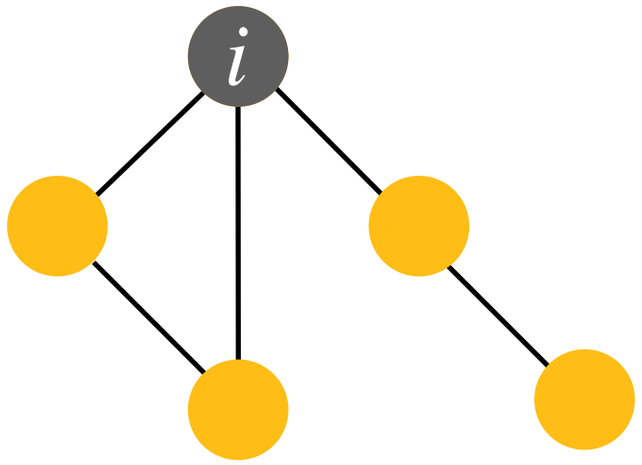
Rooted subgraph H

(3 copies of H rooted at node i appear in A)

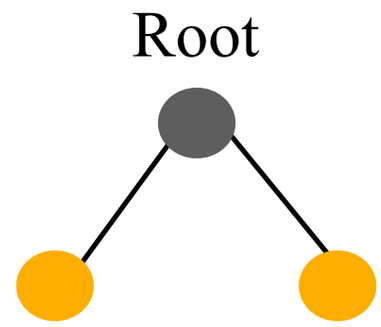
$$W_{i,H}(A) = 3$$

- # copies of H rooted at i in graph A, denoted as $W_{i,H}(A)$;

Rooted subgraph count



A



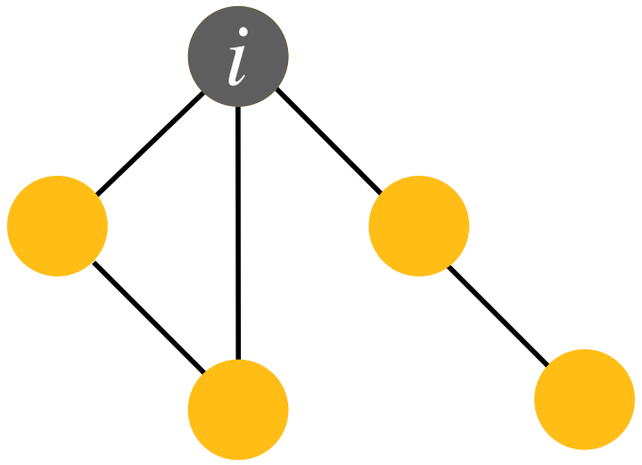
Rooted subgraph H

(3 copies of H rooted at node i appear in A)

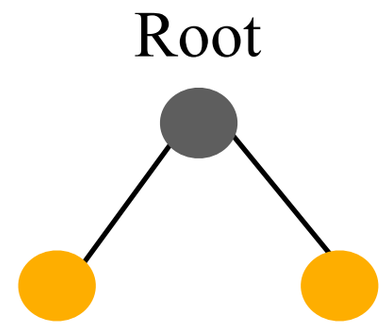
$$W_{i,H}(A) = 3$$

- # copies of H rooted at i in graph A, denoted as $W_{i,H}(A)$;
- Capture some vertex information;

Rooted subgraph count



A



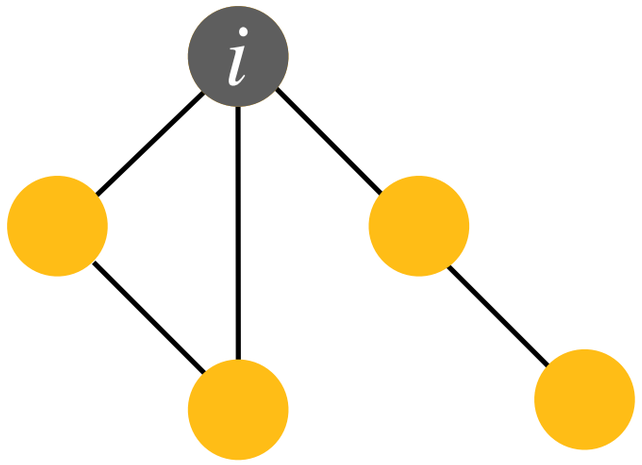
Rooted subgraph H

(3 copies of H rooted at node i appear in A)

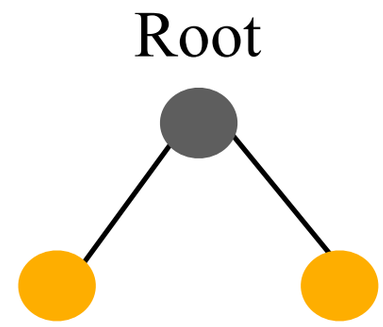
$$W_{i,H}(A) = 3$$

- # copies of H rooted at i in graph A, denoted as $W_{i,H}(A)$;
- Capture some vertex information;
- Idea: construct a rich family of rooted subgraphs:

Rooted subgraph count



A



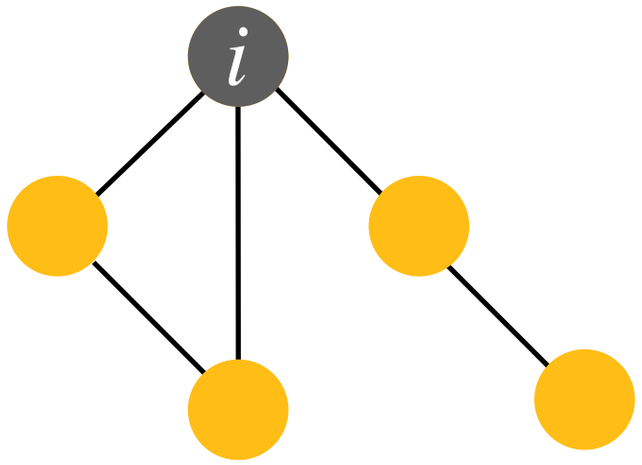
Rooted subgraph H

(3 copies of H rooted at node i appear in A)

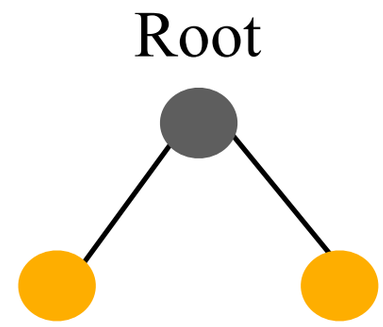
$$W_{i,H}(A) = 3$$

- # copies of H rooted at i in graph A, denoted as $W_{i,H}(A)$;
- Capture some vertex information;
- Idea: construct a rich family of rooted subgraphs:
  - Each rooted subgraph captures some information about the vertex;

Rooted subgraph count



A



Rooted subgraph H

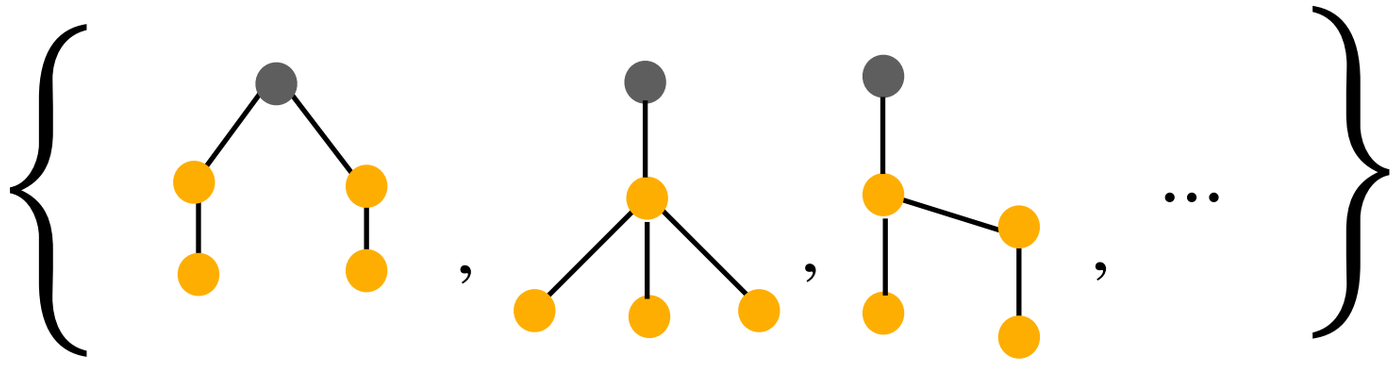
(3 copies of H rooted at node i appear in A)

$$W_{i,H}(A) = 3$$

- # copies of H rooted at i in graph A, denoted as $W_{i,H}(A)$;
- Capture some vertex information;
- Idea: construct a rich family of rooted subgraphs:
  - Each rooted subgraph captures some information about the vertex;
  - **Combine** all the information $\Longrightarrow$ vertex signature to be more informative.

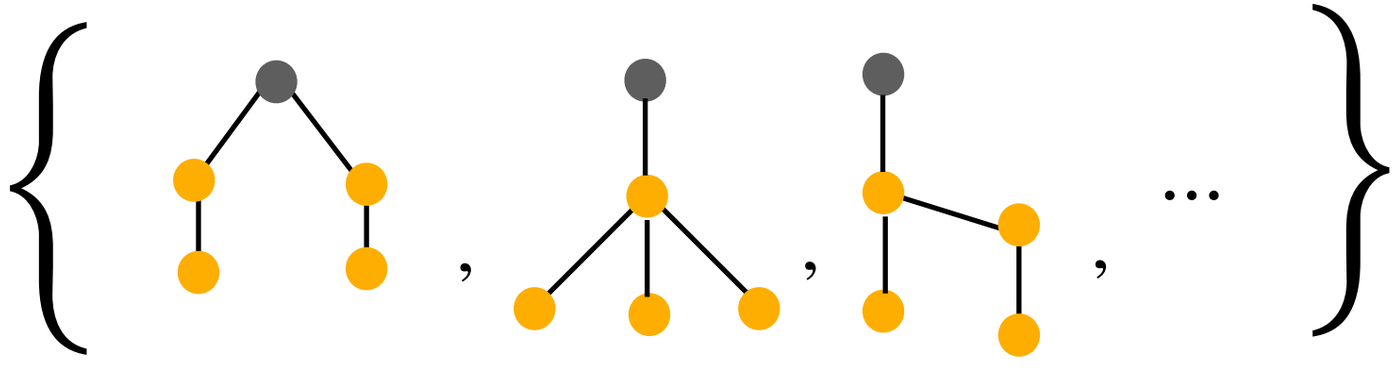
Graph matching via counting rooted subgraphs

Given **a family $\mathscr{H}$ of rooted subgraphs** with N edges:

e.g. $N = 4$, $\mathscr{H} =$ 

Graph matching via counting rooted subgraphs

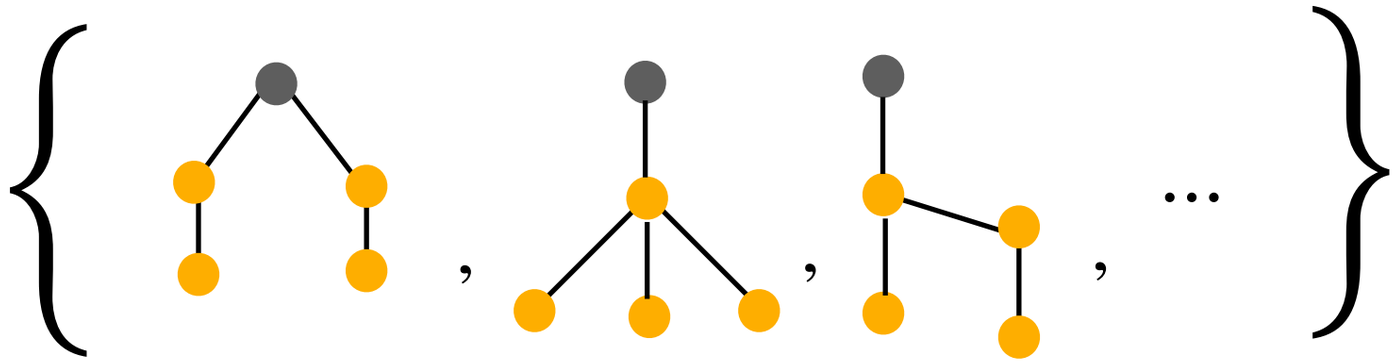
Given **a family $\mathscr{H}$ of rooted subgraphs** with N edges:

e.g. $N = 4$, $\mathscr{H} =$ 

1. For each vertex i in A , its vertex signature is $s_i = \left(W_{i,H}(A)\right)_{H \in \mathscr{H}}$

Graph matching via counting rooted subgraphs

Given **a family $\mathscr{H}$ of rooted subgraphs** with N edges:

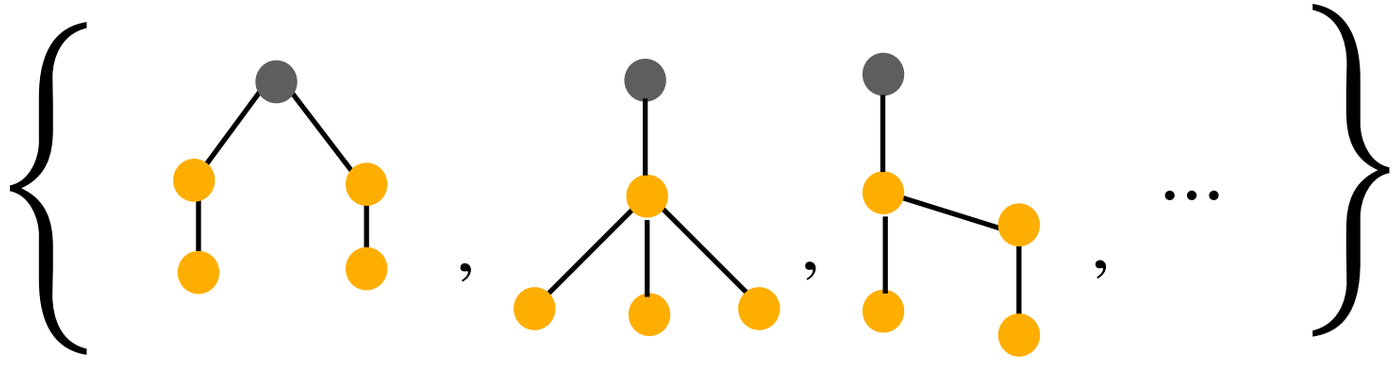
e.g. $N = 4$, $\mathscr{H} =$ 

Highly correlated across different H

1. For each vertex i in A , its vertex signature is $s_i = \left(W_{i,H}(A)\right)_{H\in\mathscr{H}}$

Graph matching via counting rooted subgraphs

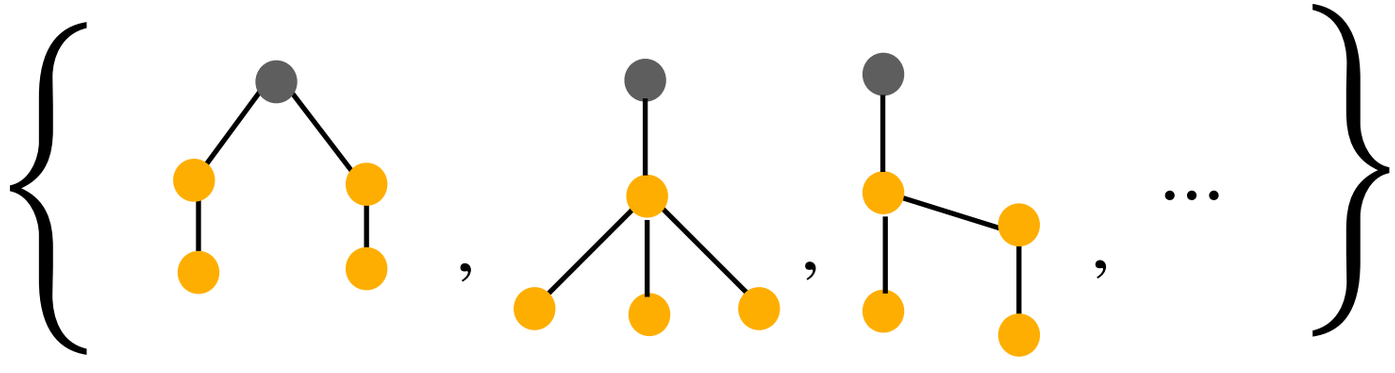
Given **a family $\mathscr{H}$ of rooted subgraphs** with N edges:

e.g. $N = 4$, $\mathscr{H} =$ 

1. For each vertex i in A , its vertex signature is $s_i = \left(W_{i,H}(\overline{A})\right)_{H \in \mathscr{H}}$

Graph matching via counting rooted subgraphs

Given **a family $\mathscr{H}$ of rooted subgraphs** with N edges:

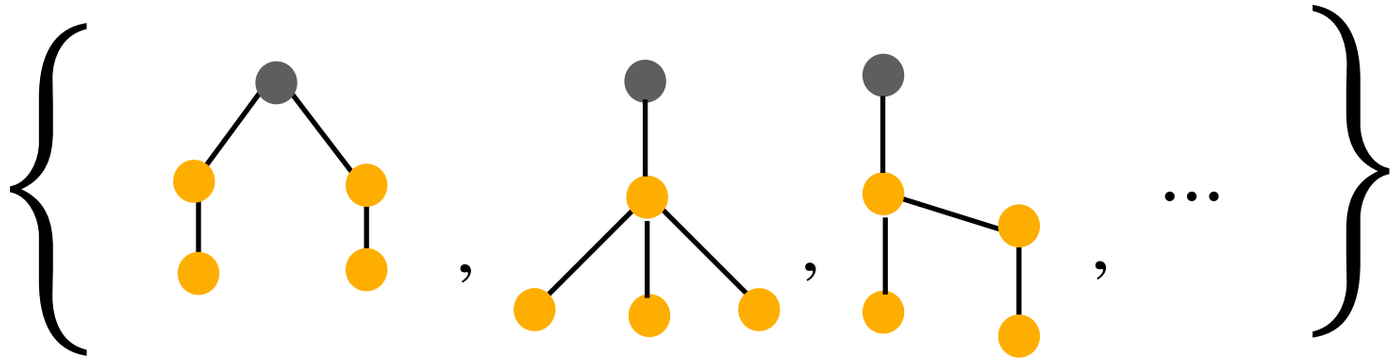
e.g. $N = 4$, $\mathscr{H} =$ 

1. For each vertex i in A , its vertex signature is $s_i = \left(W_{i,H}(\overline{A})\right)_{H \in \mathscr{H}}$

$\overline{A} = A - \mathbb{E}[A]$;
Count the weighted copies in $\overline{A}$
$W_{i,H}(\bar{A}) = \sum_{S(i) \cong H} \prod_{e \in E(S)} \bar{A}_e$

Graph matching via counting rooted **signed** subgraphs

Given **a family $\mathscr{H}$ of rooted subgraphs** with N edges:

e.g. $N = 4$, $\mathscr{H} =$ 

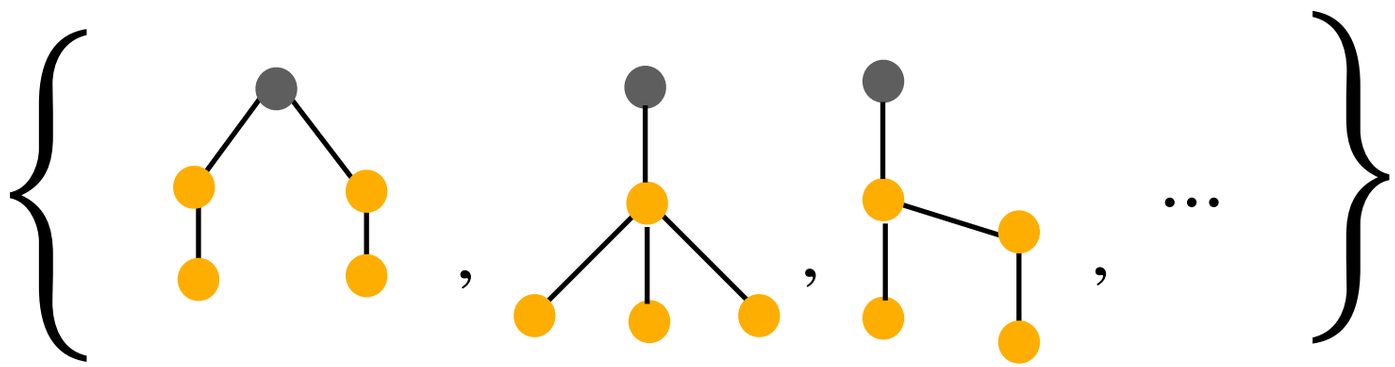
1. For each vertex i in A , its vertex signature is $s_i = \left(W_{i,H}(\overline{A})\right)_{H\in\mathscr{H}}$

rooted "**signed**" subgraph count [Bubeck-Ding-Eldan-Rácz '16]

$\overline{A} = A - \mathbb{E}[A]$;
Count the weighted copies in $\overline{A}$
$W_{i,H}(\bar{A}) = \sum_{S(i)\cong H} \prod_{e\in E(S)} \bar{A}_e$

Graph matching via counting rooted **signed** subgraphs

Given **a family $\mathscr{H}$ of rooted subgraphs** with N edges:

e.g. $N = 4$, $\mathscr{H} =$ 

Uncorrelated across different H

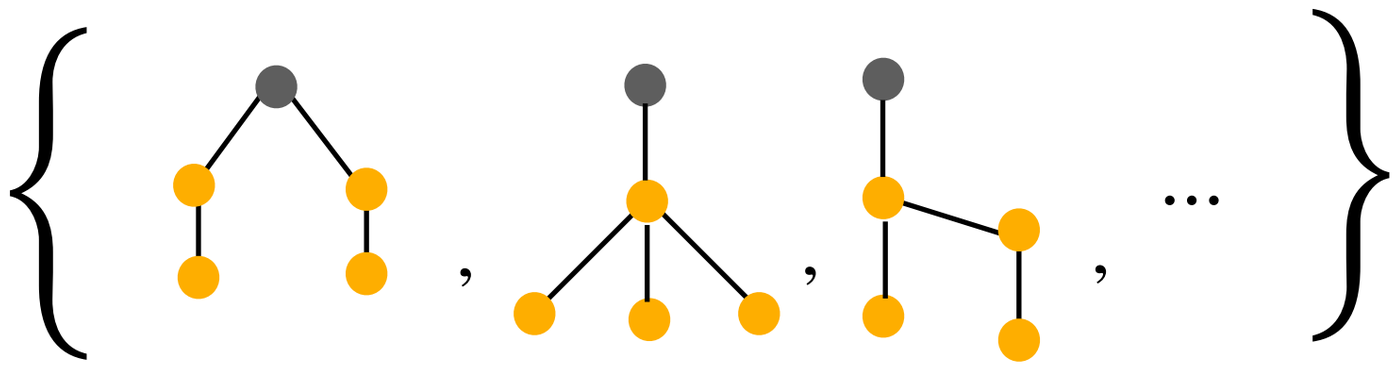
1. For each vertex i in A , its vertex signature is $s_i = \left(W_{i,H}(\overline{A})\right)_{H\in\mathscr{H}}$

rooted "**signed**" subgraph count

$\overline{A} = A - \mathbb{E}[A]$;
Count the weighted copies in $\overline{A}$

Graph matching via counting rooted **signed** subgraphs

Given **a family $\mathscr{H}$ of rooted subgraphs** with N edges:

e.g. $N = 4$, $\mathscr{H} =$ 

Uncorrelated across different H

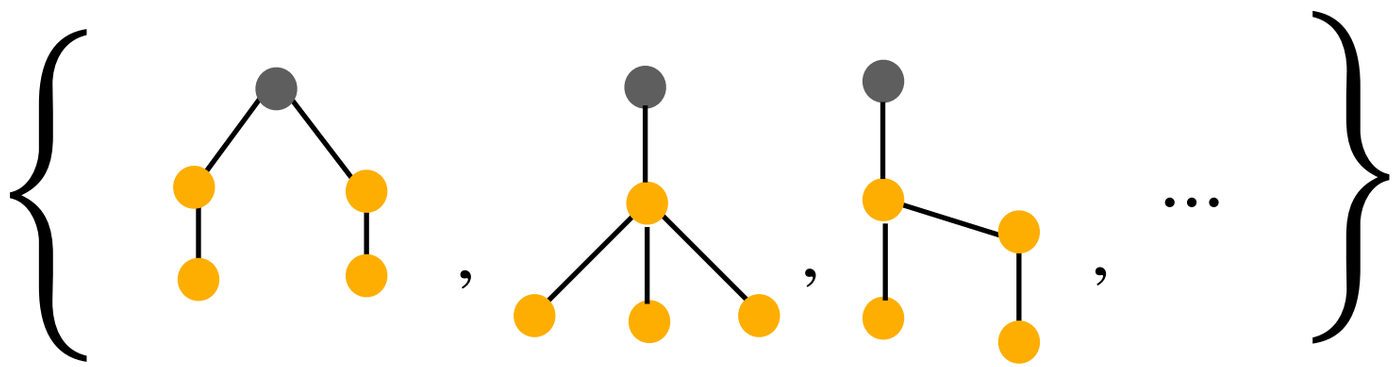
1. For each vertex i in A , its vertex signature is $s_i = \left(W_{i,H}(\overline{A})\right)_{H\in\mathscr{H}}$
2. For each vertex j in B , its vertex signature is $t_j = \left(W_{j,H}(\overline{B})\right)_{H\in\mathscr{H}}$

rooted "**signed**" subgraph count

Uncorrelated across different H

Graph matching via counting rooted **signed** subgraphs

Given **a family $\mathscr{H}$ of rooted subgraphs** with N edges:

e.g. $N = 4$, $\mathscr{H} =$ 

Uncorrelated across different H

1. For each vertex i in A , its vertex signature is $s_i = \left(W_{i,H}(\overline{A})\right)_{H\in\mathscr{H}}$
2. For each vertex j in B , its vertex signature is $t_j = \left(W_{j,H}(\overline{B})\right)_{H\in\mathscr{H}}$
3. Similarity score:

rooted "**signed**" subgraph count

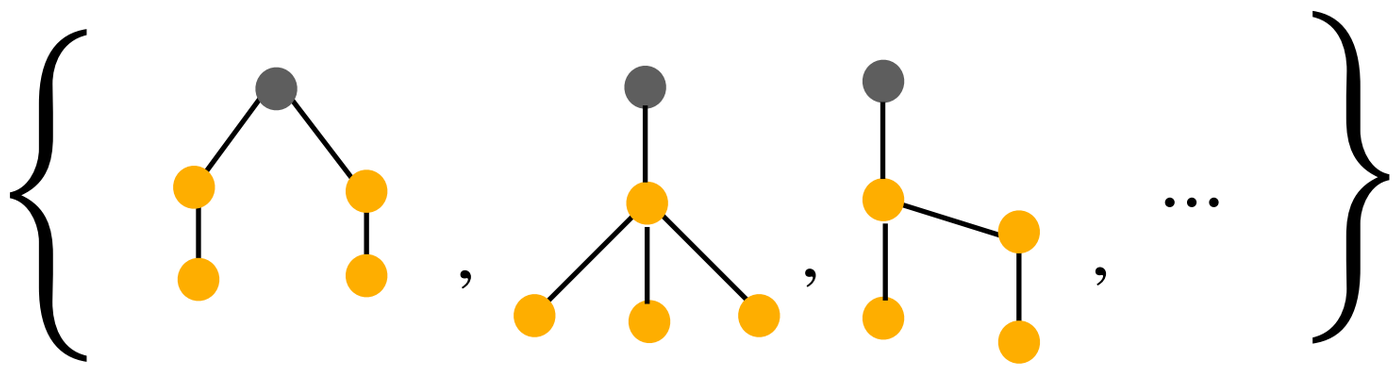
Uncorrelated across different H

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H\in\mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) ,$$

where $\mathrm{aut}(H)$ is the number of automorphism of H.

Graph matching via counting rooted **signed** subgraphs

Given **a family $\mathscr{H}$ of rooted subgraphs** with N edges:

e.g. $N = 4$, $\mathscr{H} =$ 

Uncorrelated across different H

1. For each vertex i in A , its vertex signature is $s_i = \left(W_{i,H}(\overline{A})\right)_{H\in\mathscr{H}}$
2. For each vertex j in B , its vertex signature is $t_j = \left(W_{j,H}(\overline{B})\right)_{H\in\mathscr{H}}$
3. Similarity score:

rooted "**signed**" subgraph count

Uncorrelated across different H

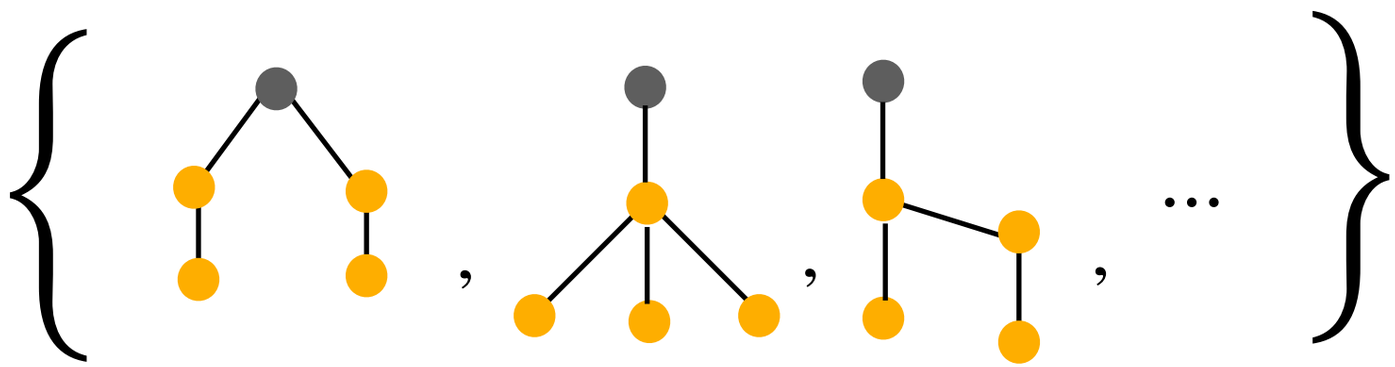
$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H\in\mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) ,$$

where $\mathrm{aut}(H)$ is the number of automorphism of H.

4. Match i to j if $\Phi_{ij} \geq \tau$ for some threshold τ.

Graph matching via counting rooted **signed** subgraphs

Given **a family $\mathscr{H}$ of rooted subgraphs** with N edges:

e.g. $N = 4$, $\mathscr{H} =$ 

Uncorrelated across different H

1. For each vertex i in A , its vertex signature is $s_i = \left(W_{i,H}(\overline{A})\right)_{H\in\mathscr{H}}$
2. For each vertex j in B , its vertex signature is $t_j = \left(W_{j,H}(\overline{B})\right)_{H\in\mathscr{H}}$

rooted "**signed**" subgraph count

Uncorrelated across different H

3. Similarity score:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H\in\mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) ,$$

Polynomials of A_{ij}'s and B_{ij}'s

where $\mathrm{aut}(H)$ is the number of automorphism of H.

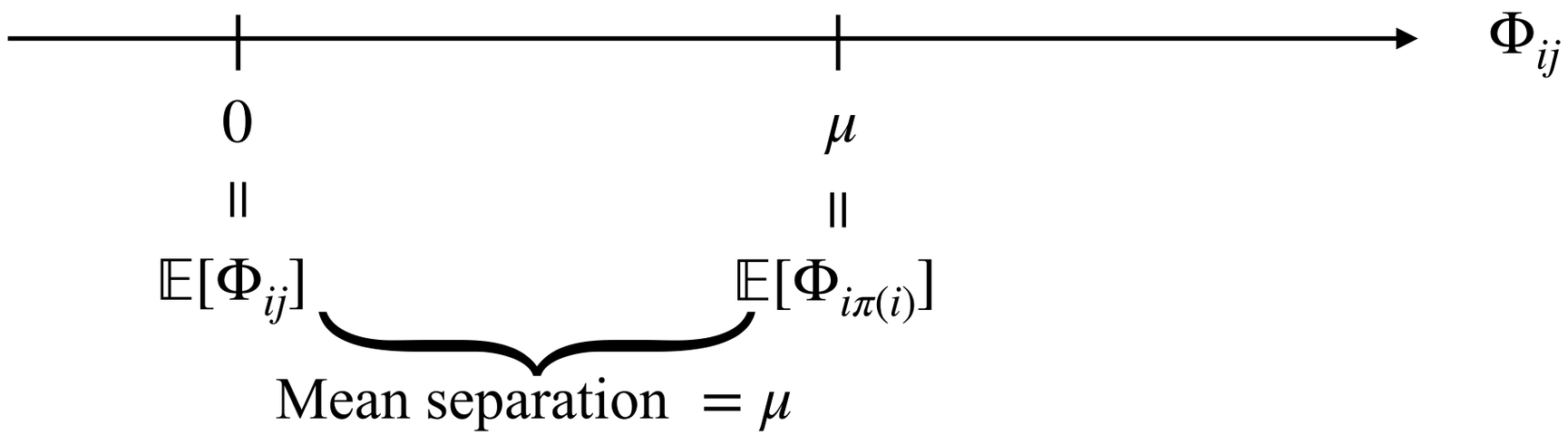
4. Match i to j if $\Phi_{ij} \geq \tau$ for some threshold τ.

Desirable properties of similarity scores (Mean)

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) \underbrace{W_{i,H}(\overline{A}) W_{j,H}(\overline{B})}_{\text{Correlated iff } j = \pi(i) \text{ (true pair)}} .$$

Fake pair: $j \neq \pi(i)$ True pair: $j = \pi(i)$



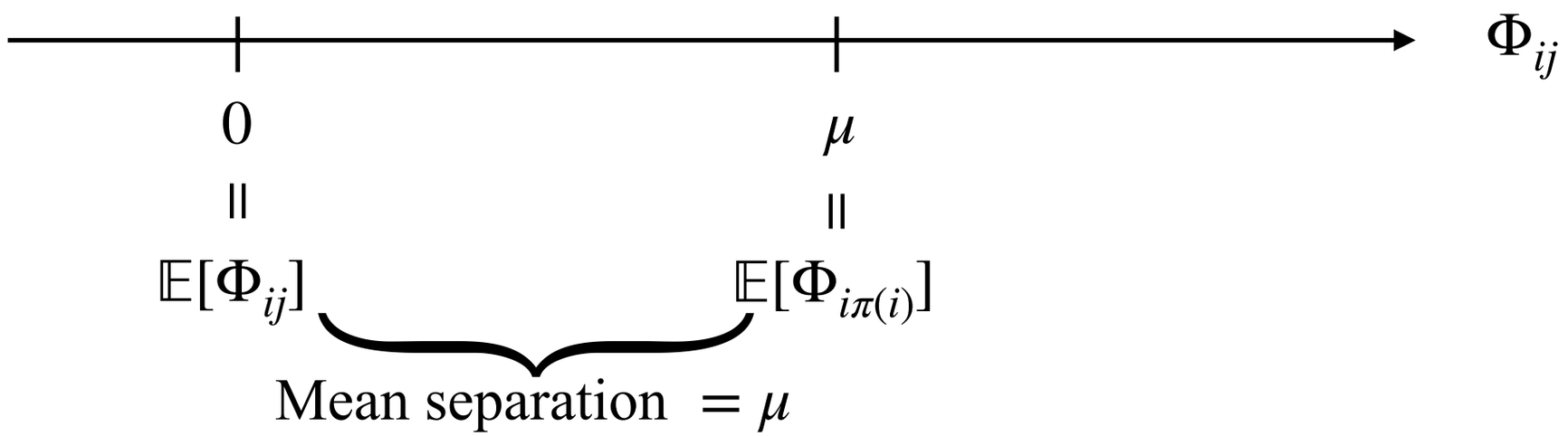
Desirable properties of similarity scores (Variance)

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) \underbrace{W_{i,H}(\overline{A}) W_{j,H}(\overline{B})}_{\text{Correlated iff } j = \pi(i) \text{ (true pair)}} .$$

We want: **fluctuation** of Φ_{ij} to be relatively small compared to μ.

Fake pair: $j \neq \pi(i)$ True pair: $j = \pi(i)$

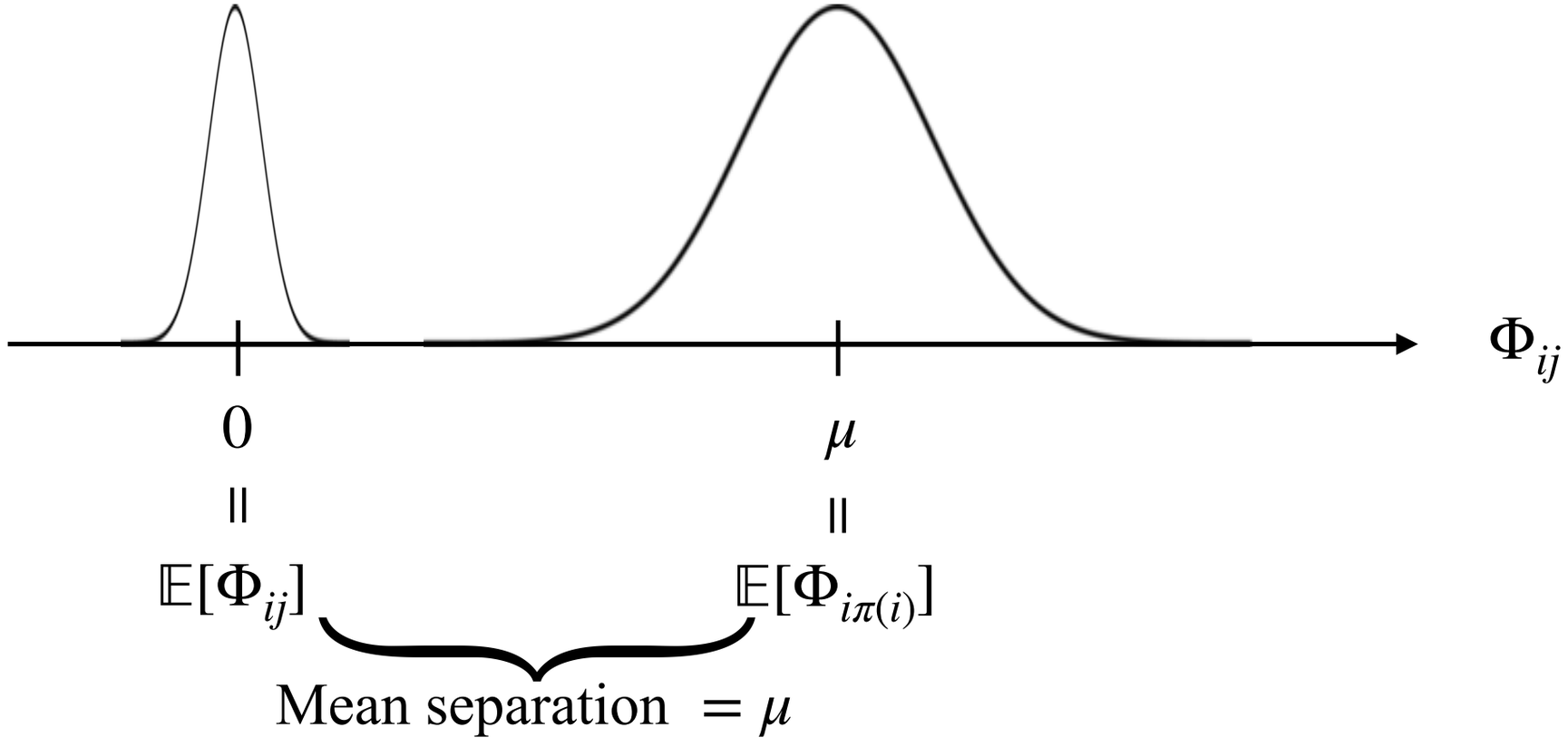


Desirable properties of similarity scores (Variance)

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) \underbrace{W_{i,H}(\overline{A}) W_{j,H}(\overline{B})}_{\text{Correlated iff } j = \pi(i) \text{ (true pair)}} .$$

We want: **fluctuation** of Φ_{ij} to be relatively small compared to μ.

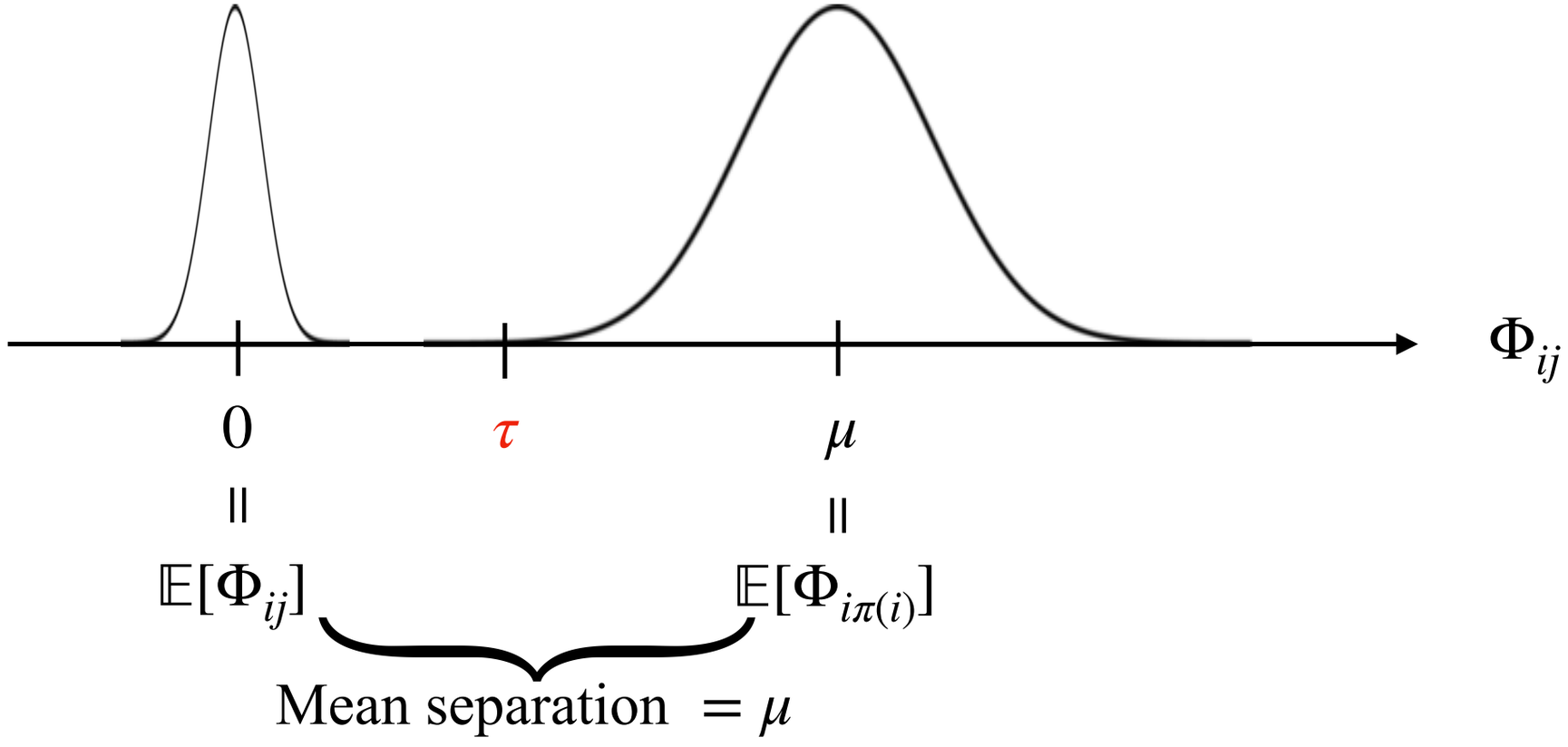


Desirable properties of similarity scores (Variance)

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) \underbrace{W_{i,H}(\overline{A}) W_{j,H}(\overline{B})}_{\text{Correlated iff } j = \pi(i) \text{ (true pair)}} .$$

We want: **fluctuation** of Φ_{ij} to be relatively small compared to μ.

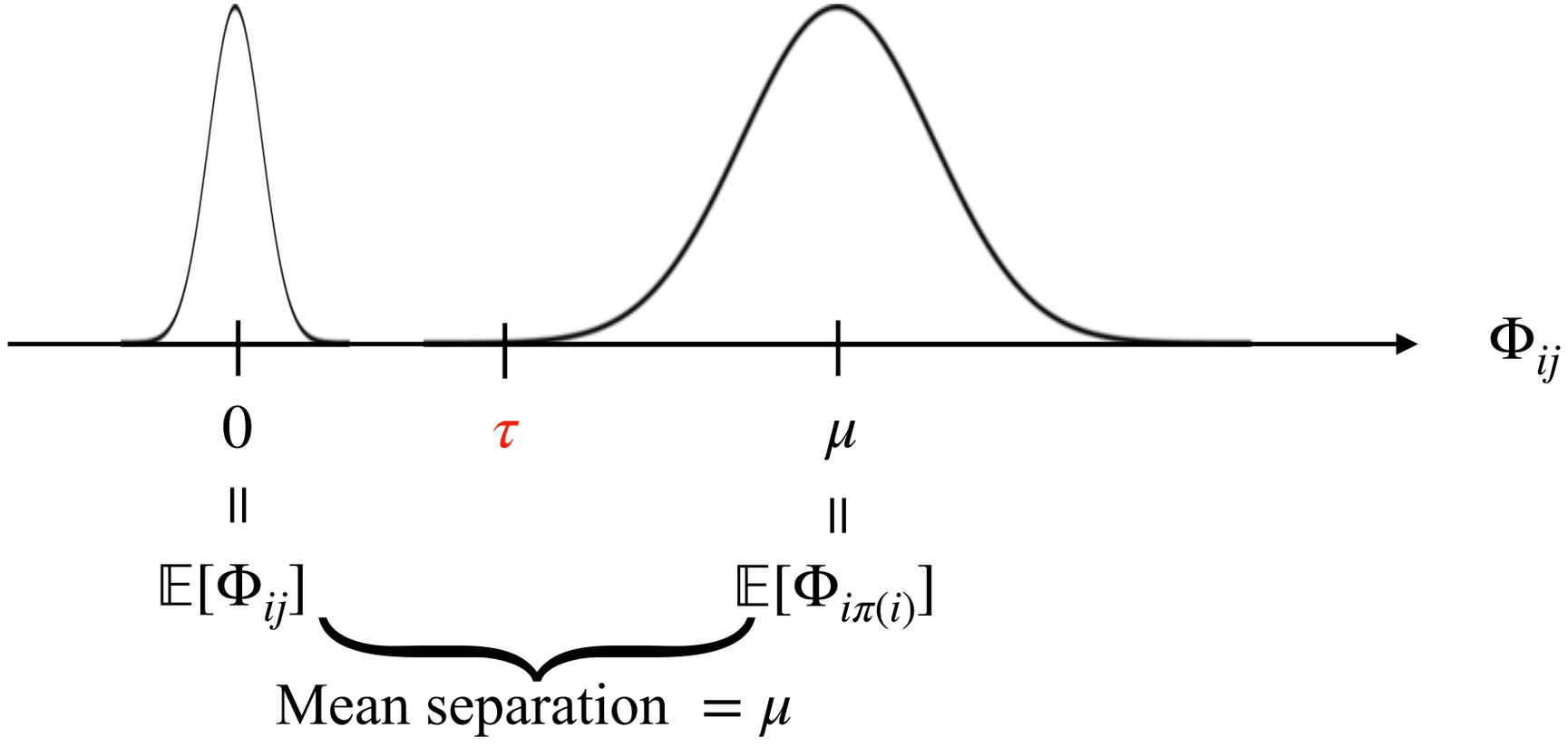


Desirable properties of similarity scores (Variance)

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) \underbrace{W_{i,H}(\overline{A}) W_{j,H}(\overline{B})}_{\text{Correlated iff } j = \pi(i) \text{ (true pair)}} .$$

We want: **fluctuation** of Φ_{ij} to be relatively small compared to μ.



Which family of $\mathscr{H}$ (rooted subgraphs) to choose?

Which family of rooted subgraphs to count?

Suppose $\mathscr{H}$ be a family of rooted subgraphs with N edges.

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) \,.$$

Which family of rooted subgraphs to count?

Suppose $\mathscr{H}$ be a family of rooted subgraphs with N edges.

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) \,.$$

- Wishful thinking: **ignoring the cross-correlations** of $\mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B})$ and $\mathrm{aut}(I) W_{i,I}(\overline{A}) W_{j,I}(\overline{B})$ for different subgraphs H and I:

$$\frac{\mathrm{Var}[\Phi_{ij}]}{\mu^2} \approx \frac{1}{|\mathscr{H}| \rho^{2N}} \xrightarrow{\text{goal}} 0$$

Which family of rooted subgraphs to count?

Suppose $\mathscr{H}$ be a family of rooted subgraphs with N edges.

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) \,.$$

- Wishful thinking: **ignoring the cross-correlations** of $\mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B})$ and $\mathrm{aut}(I) W_{i,I}(\overline{A}) W_{j,I}(\overline{B})$ for different subgraphs H and I:

$$\frac{\mathrm{Var}[\Phi_{ij}]}{\mu^2} \approx \frac{1}{|\mathscr{H}| \rho^{2N}} \xrightarrow{\text{goal}} 0$$

which requires

- The family $\mathscr{H}$ to be "**rich**": $|\mathscr{H}|$ grows at least exponentially in N;
- The subgraph H to be "**large**": N grows in n;

Which family of rooted subgraphs to count?

Suppose $\mathscr{H}$ be a family of rooted subgraphs with N edges.

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) .$$

- Wishful thinking: **ignoring the cross-correlations** of $\mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B})$ and $\mathrm{aut}(I) W_{i,I}(\overline{A}) W_{j,I}(\overline{B})$ for different subgraphs H and I:

$$\frac{\mathrm{Var}[\Phi_{ij}]}{\mu^2} \approx \frac{1}{|\mathscr{H}| \rho^{2N}} \xrightarrow{\text{goal}} 0$$

  which requires
  - The family $\mathscr{H}$ to be "**rich**": $|\mathscr{H}|$ grows at least exponentially in N;
  - The subgraph H to be "**large**": N grows in n;
- We also want H to be **"simple"** to count.

Which family of rooted subgraphs to count?

Suppose $\mathscr{H}$ be a family of rooted subgraphs with N edges.

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) .$$

- Wishful thinking: **ignoring the cross-correlations** of $\mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B})$ and $\mathrm{aut}(I) W_{i,I}(\overline{A}) W_{j,I}(\overline{B})$ for different subgraphs H and I:

$$\frac{\mathrm{Var}[\Phi_{ij}]}{\mu^2} \approx \frac{1}{|\mathscr{H}| \rho^{2N}} \xrightarrow{\text{goal}} 0$$

which requires

- The family $\mathscr{H}$ to be "**rich**": $|\mathscr{H}|$ grows at least exponentially in N;
- The subgraph H to be "**large**": N grows in n;

} **Trees**

- We also want H to be **"simple"** to count.

A family of rooted **trees**

Suppose $\mathscr{H}$ be a family of rooted **trees** with N edges.

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) .$$

- Wishful thinking: **ignoring the cross-correlations** of $\mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B})$ and $\mathrm{aut}(I) W_{i,I}(\overline{A}) W_{j,I}(\overline{B})$ for different subgraphs H and I:

$$\frac{\mathrm{Var}[\Phi_{ij}]}{\mu^2} \approx \frac{1}{|\mathscr{H}| \rho^{2N}} \xrightarrow{\text{goal}} 0$$

which requires

- The family $\mathscr{H}$ to be "**rich**": $|\mathscr{H}| = (\alpha + o(1))^{-N}$;
- The subgraph H to be "**large**": N grows in n;

} **Trees**

- We also want H to be **"simple"** to count.

A family of rooted **trees**

Suppose $\mathscr{H}$ be a family of rooted **trees** with N edges.

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) .$$

- Wishful thinking: **ignoring the cross-correlations** of $\mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B})$ and $\mathrm{aut}(I) W_{i,I}(\overline{A}) W_{j,I}(\overline{B})$ for different subgraphs H and I:

$$\frac{\mathrm{Var}[\Phi_{ij}]}{\mu^2} \approx \frac{1}{|\mathscr{H}| \rho^{2N}} \longrightarrow 0$$

$\rho^2 > \alpha$

which requires

  - The family $\mathscr{H}$ to be "**rich**": $|\mathscr{H}| = (\alpha + o(1))^{-N}$;
  - The subgraph H to be "**large**": N grows in n;

- We also want H to be **"simple"** to count.

} **Trees**

A family of rooted **trees**

Suppose $\mathscr{H}$ be a family of rooted **trees** with N edges.

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) .$$

- Wishful thinking: **ignoring the cross-correlations** of $\mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B})$ and $\mathrm{aut}(I) W_{i,I}(\overline{A}) W_{j,I}(\overline{B})$ for different subgraphs H and I:

$$\frac{\mathrm{Var}[\Phi_{ij}]}{\mu^2} \approx \frac{1}{|\mathscr{H}| \rho^{2N}} \to 0$$

$\rho^2 > \alpha$

which requires

  - The family $\mathscr{H}$ to be "**rich**": $|\mathscr{H}| = (\alpha + o(1))^{-N}$;
  - The subgraph H to be "**large**": N grows in n;

- If H is a tree, it is **"simple"** to count via color coding [Alon-Yuster-Zwick' 95]

} **Trees**

A family of rooted **trees**

Suppose $\mathscr{H}$ be a family of rooted **trees** with N edges.

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) .$$

- Wishful thinking: **ignoring the cross-correlations** of $\mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B})$ and $\mathrm{aut}(I) W_{i,I}(\overline{A}) W_{j,I}(\overline{B})$ for different subgraphs H and I:

$$\frac{\mathrm{Var}[\Phi_{ij}]}{\mu^2} \approx \frac{1}{|\mathscr{H}| \rho^{2N}} \to 0$$

($\rho^2 > \alpha$; $\approx$ marked with a red cross)

which requires

  - The family $\mathscr{H}$ to be "**rich**": $|\mathscr{H}| = (\alpha + o(1))^{-N}$;
  - The subgraph H to be "**large**": N grows in n;

- If H is a tree, it is **"simple"** to count via color coding [Alon-Yuster-Zwick' 95].
- However, we **cannot ignore the cross-correlations.**

We need to **construct** a **special** family of trees.

A **special** family of rooted trees

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) \,.$$

We want to construct a special family of trees:

A **special** family of rooted trees

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) \,.$$

We want to construct a special family of trees:

- **Suppress** the undesirable cross-correlations to control $\mathrm{Var}[\Phi_{ij}]/\mu^2$;

A **special** family of rooted trees

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) \,.$$

We want to construct a special family of trees:

- **Suppress** the undesirable cross-correlations to control $\mathrm{Var}[\Phi_{ij}]/\mu^2$;
- This special family of trees has to be "**rich**" enough;

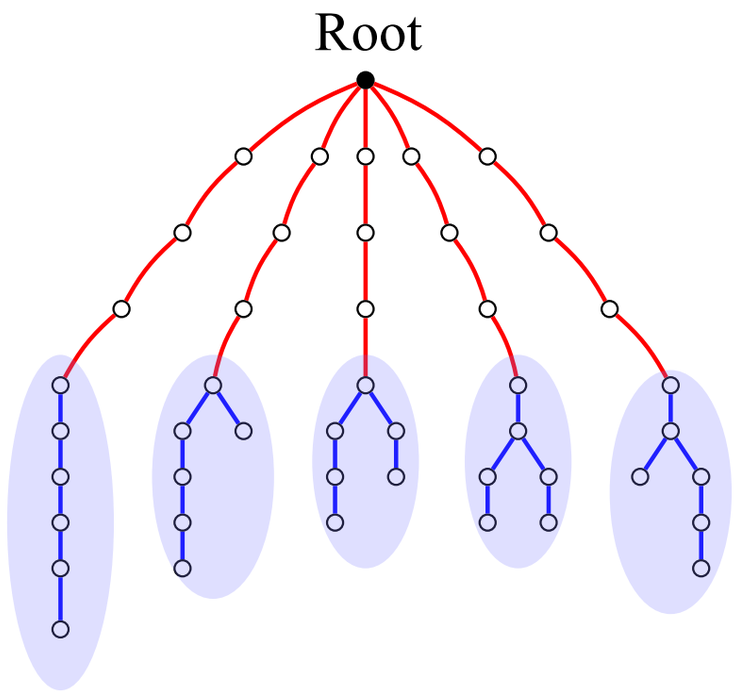
A **special** family of rooted trees - Chandeliers

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) \,.$$

We want to construct a special family of trees:

- **Suppress** the undesirable cross-correlations to control $\mathrm{Var}[\Phi_{ij}]/\mu^2$;
- This special family of trees has to be "**rich**" enough;



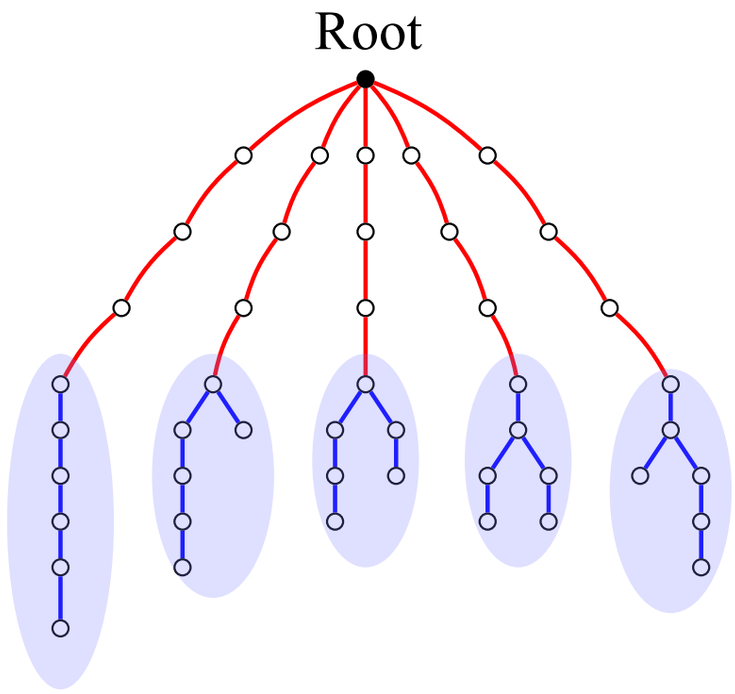
A **special** family of rooted trees - Chandeliers

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) \,.$$

We want to construct a special family of trees:

- **Suppress** the undesirable cross-correlations to control $\mathrm{Var}[\Phi_{ij}]/\mu^2$;
- This special family of trees has to be "**rich**" enough;



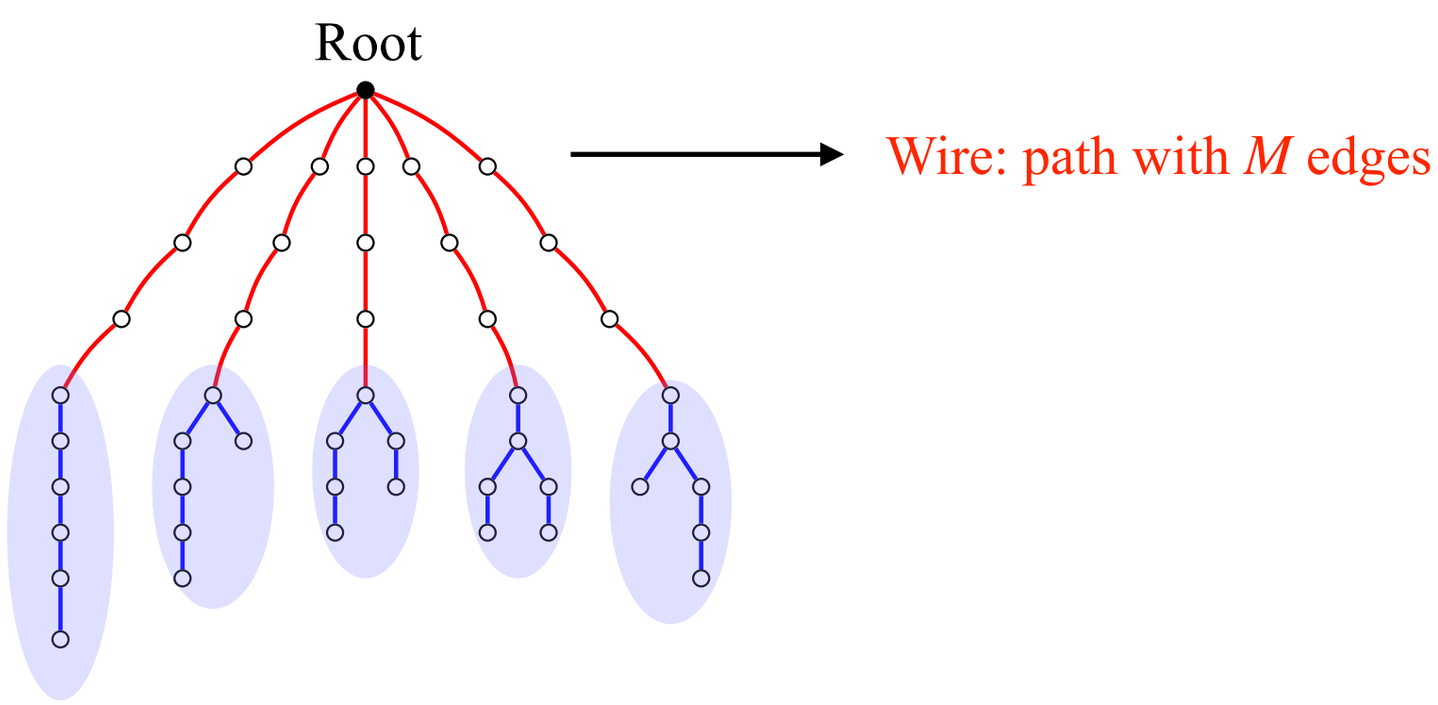
A **special** family of rooted trees - Chandeliers

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) \,.$$

We want to construct a special family of trees:

- **Suppress** the undesirable cross-correlations to control $\mathrm{Var}[\Phi_{ij}]/\mu^2$;
- This special family of trees has to be "**rich**" enough;



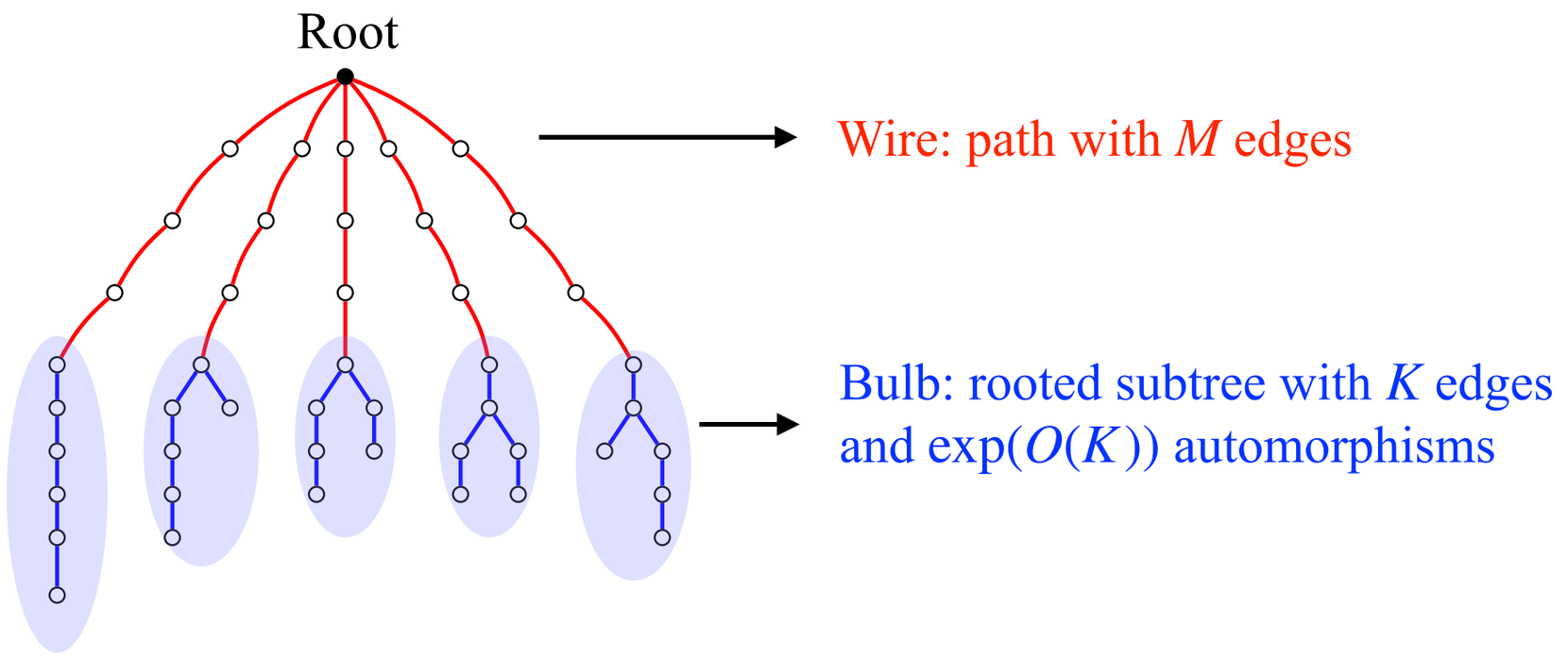
A **special** family of rooted trees - Chandeliers

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) \,.$$

We want to construct a special family of trees:

- **Suppress** the undesirable cross-correlations to control $\mathrm{Var}[\Phi_{ij}]/\mu^2$;
- This special family of trees has to be "**rich**" enough;



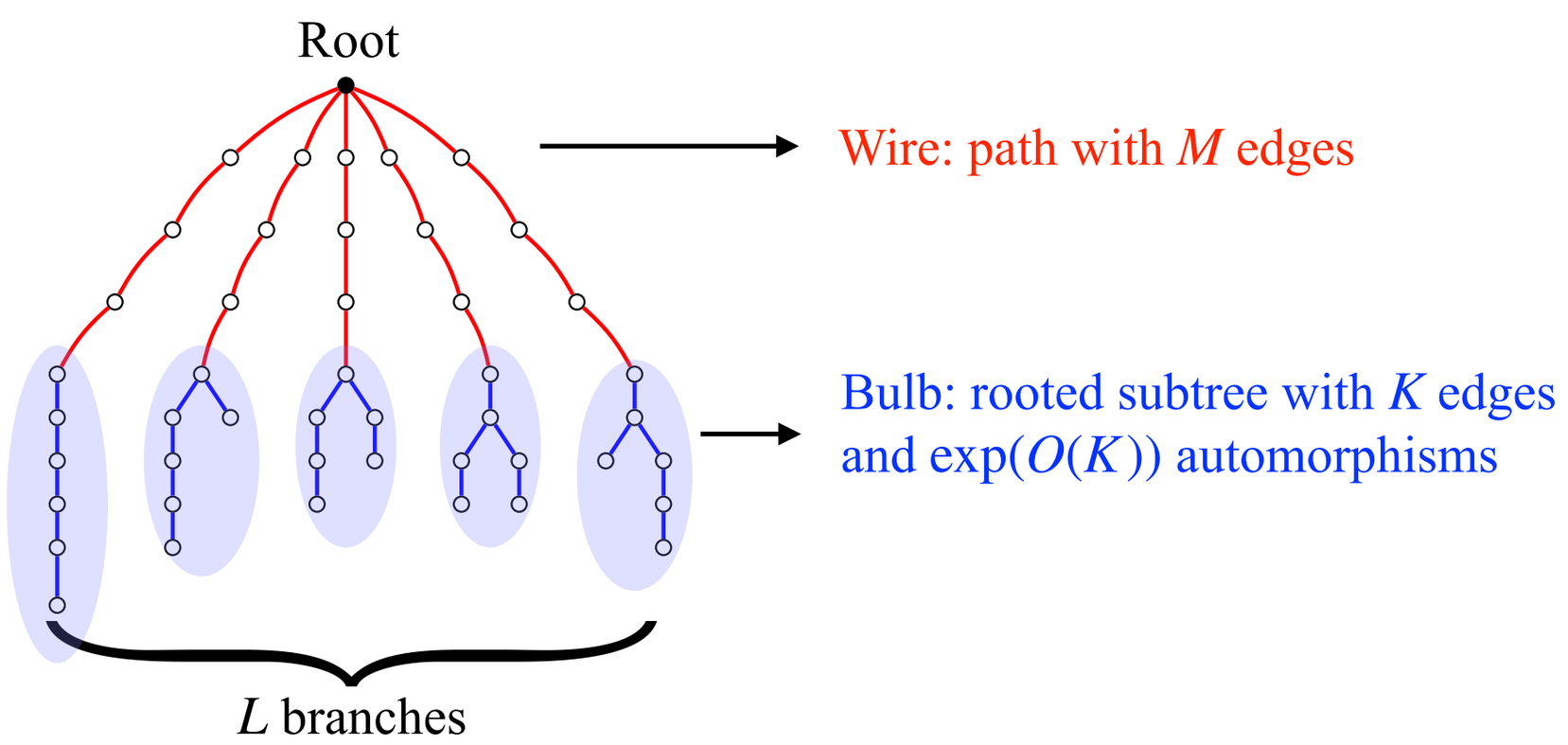
A **special** family of rooted trees - Chandeliers

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) \,.$$

We want to construct a special family of trees:

- **Suppress** the undesirable cross-correlations to control $\mathrm{Var}[\Phi_{ij}]/\mu^2$;
- This special family of trees has to be "**rich**" enough;



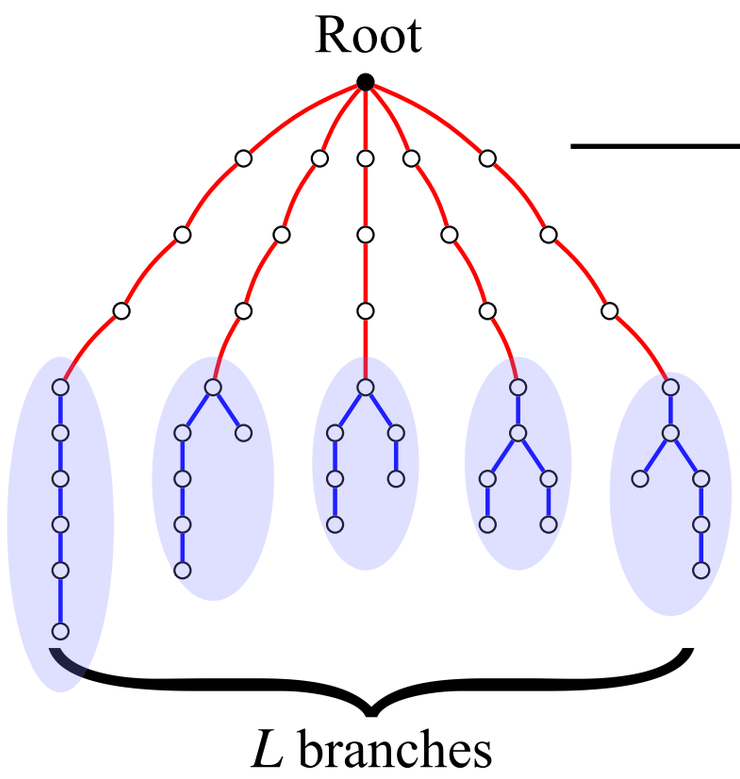
A **special** family of rooted trees - Chandeliers

For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) \,.$$

We want to construct a special family of trees:

- **Suppress** the undesirable cross-correlations to control $\mathrm{Var}[\Phi_{ij}]/\mu^2$;
- This special family of trees has to be "**rich**" enough;



Wire: path with M edges

Bulb: rooted subtree with K edges and $\exp(O(K))$ automorphisms

Chandelier has N edges where $N = (M + K)L$

A **special** family of rooted trees - Chandeliers

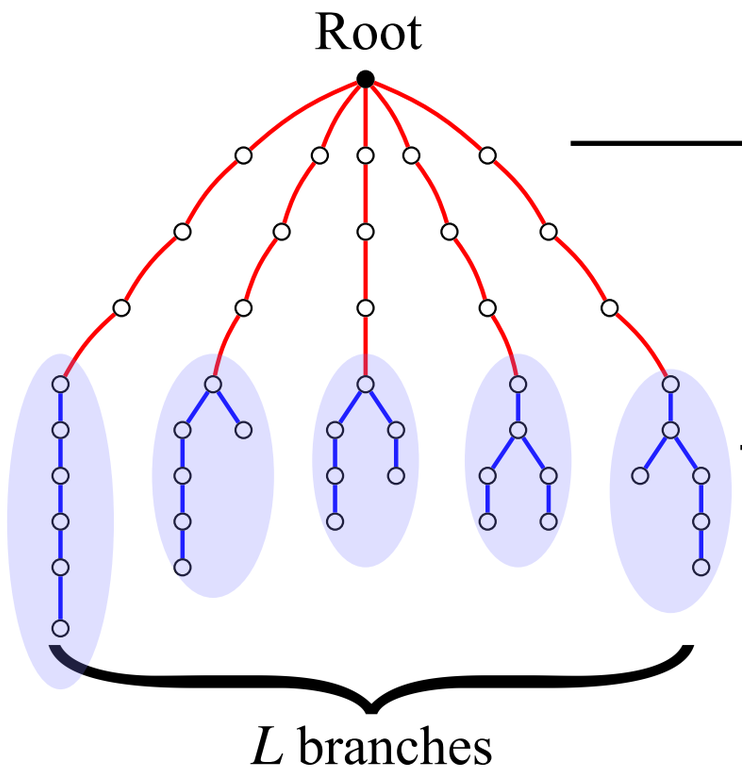
For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) .$$

We want to construct a special family of trees:

- **Suppress** the undesirable cross-correlations to control $\mathrm{Var}[\Phi_{ij}]/\mu^2$;

- Let $\mathscr{H}$ denote the special family of chandeliers with N edges.



Wire: path with M edges

Bulb: rooted subtree with K edges and $\exp(O(K))$ automorphisms

Chandelier has N edges where $N = (M + K)L$

A **special** family of rooted trees - Chandeliers

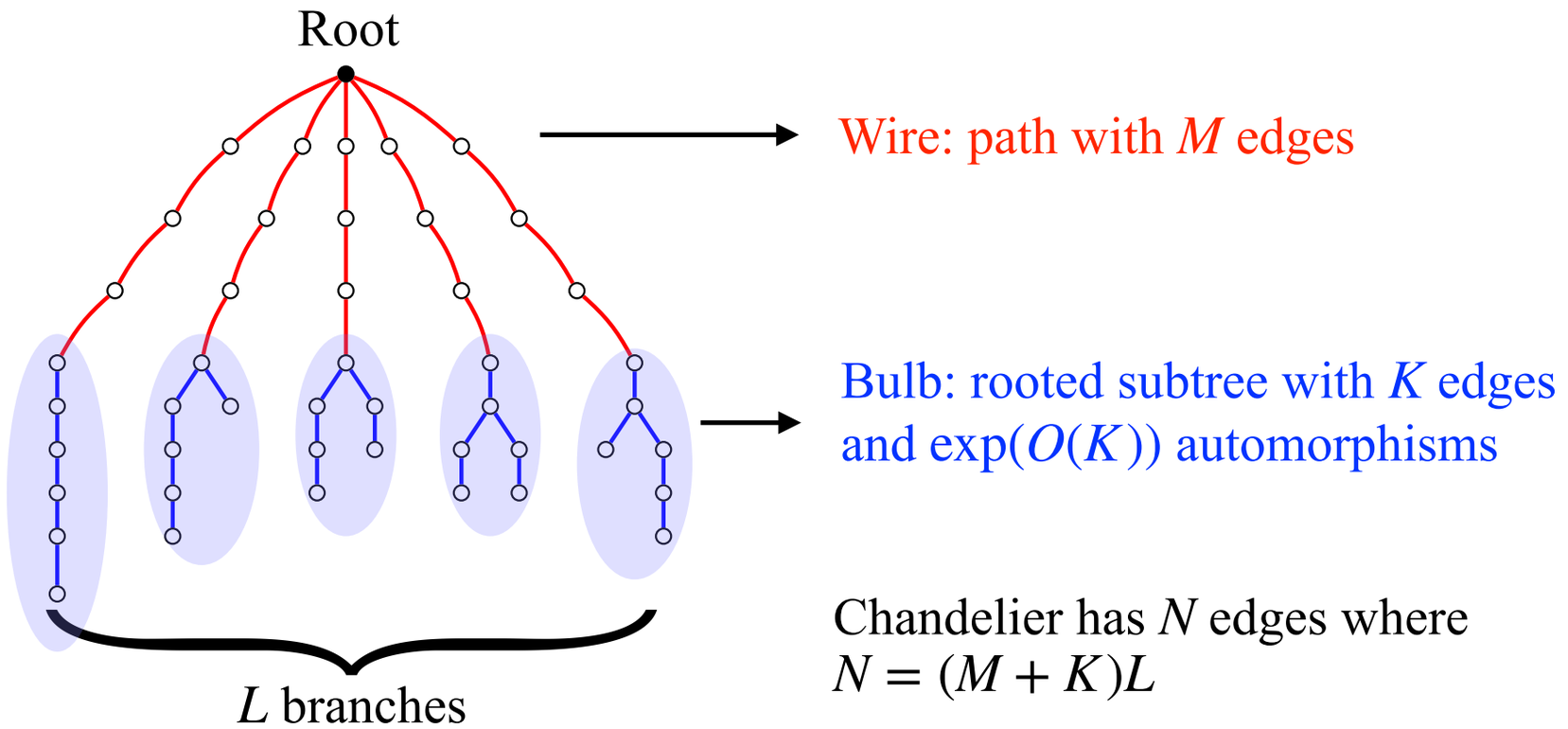
For each vertex i in A and j in B:

$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) .$$

We want to construct a special family of trees:

- **Suppress** the undesirable cross-correlations to control $\mathrm{Var}[\Phi_{ij}]/\mu^2$;

- Let $\mathscr{H}$ denote the special family of chandeliers with N edges.
- Pick $K \gg M$, we can ensure the **richness** of $\mathscr{H}$: $|\mathscr{H}| = (\alpha + o(1))^{-N}$.



A **special** family of rooted trees - Chandeliers

For each vertex i in A and j in B:

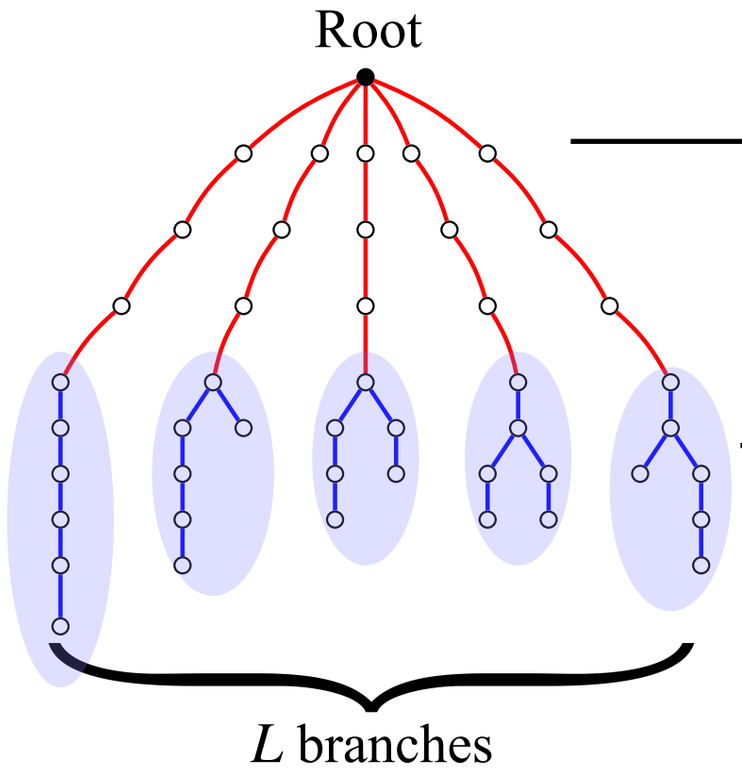
$$\Phi_{ij} = \left\langle s_i, t_j \right\rangle = \sum_{H \in \mathscr{H}} \mathrm{aut}(H) W_{i,H}(\overline{A}) W_{j,H}(\overline{B}) \,.$$

We want to construct a special family of trees:

- **Suppress** the undesirable cross-correlations to control $\mathrm{Var}[\Phi_{ij}]/\mu^2$;

- Let $\mathscr{H}$ denote the special family of chandeliers with N edges.
- Pick $K \gg M$, we can ensure the **richness** of $\mathscr{H}$: $|\mathscr{H}| = (\alpha + o(1))^{-N}$.

Majority of trees have $\exp(O(K))$ automorphisms [Olsson-Wagner '22]



Wire: path with M edges

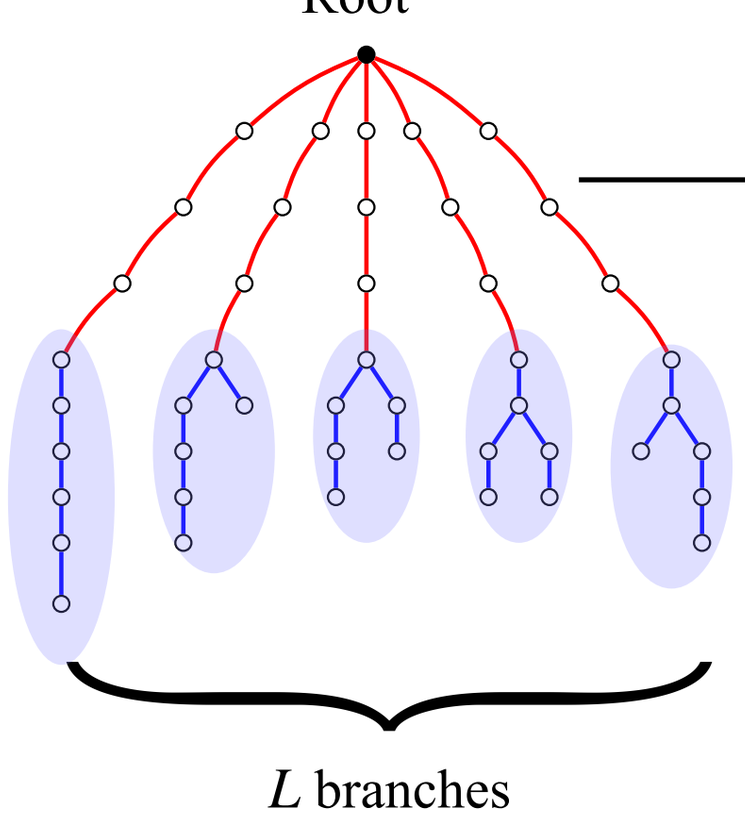
Bulb: rooted subtree with K edges and $\exp(O(K))$ automorphisms

Chandelier has N edges where $N = (M + K)L$

Summary on properties of Chandeliers

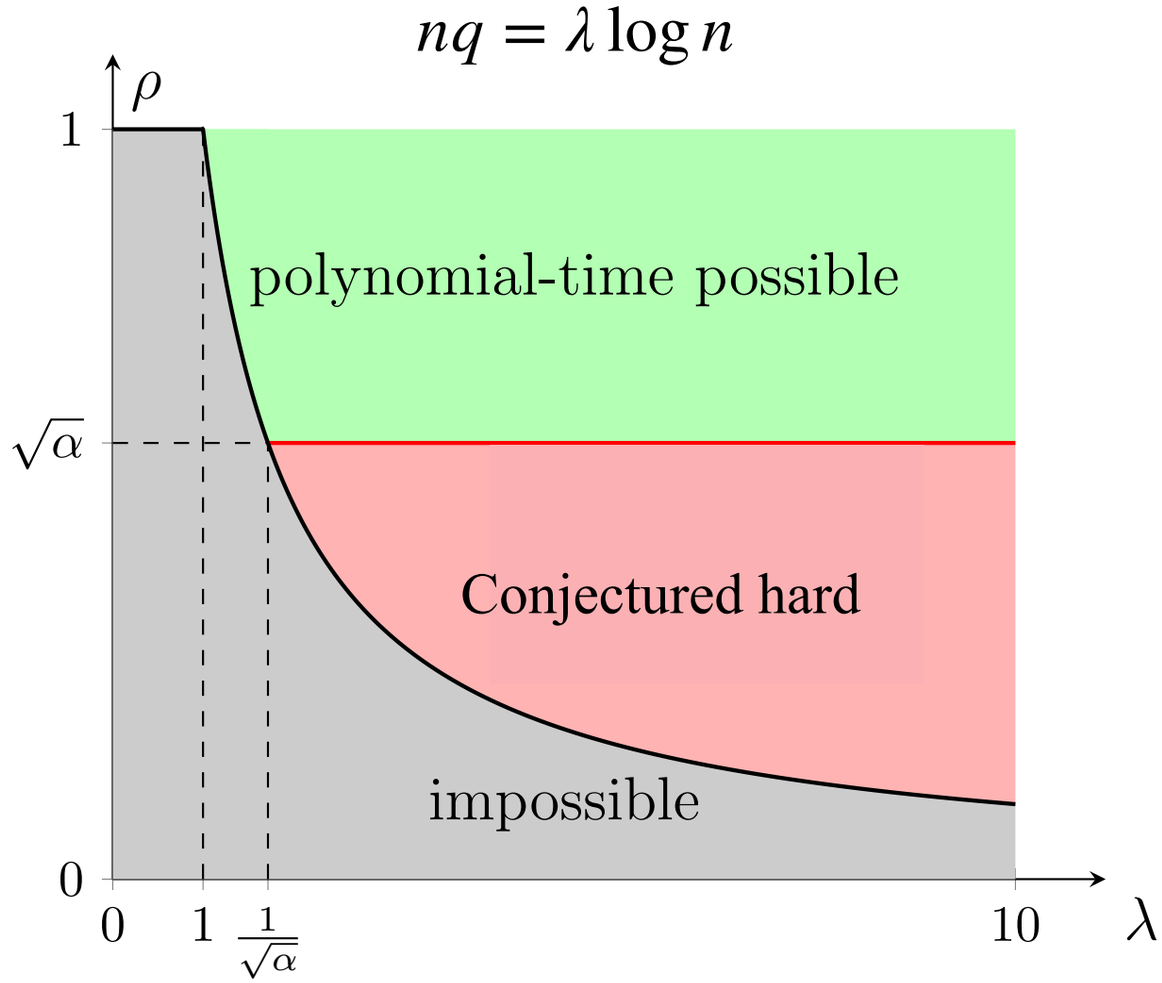
Let $\mathscr{H}$ denote a special family of chandeliers:

- **Large:** The size of chandeliers $N = c \log n$, where c is some small constant.
- **Rich:** Choose $K \gg M$, to ensure $|\mathscr{H}| = (\alpha + o(1))^{-N}$.
  - Almost as rich as the whole family of all rooted trees.
- **Informative:** Choose smaller L and larger M, when graphs are sparser.



Chandelier has N edges where $N = (M + K)L$

Summary on efficient algorithms



Selected references

Information-theoretic limits:

Y. Wu, J. Xu, & S. H. Yu, *Settling the sharp reconstruction thresholds of random graph matching*, IEEE Transactions on Information Theory, arXiv:2102.00082.

J. Ding & H. Du, *Matching recovery threshold for correlated random graphs,* Annals of Statistics, arXiv:2205.14650

Efficient algorithms and computational limits:

C. Mao, Y. Wu, J. Xu, & S. H. Yu, *Random graph matching at Otter's threshold via counting chandeliers*, STOC 2023, arxiv:2209.12313

L. Ganassali, L. Massoulié, & G. Semerjian, *Statistical limits of correlation detection in trees,* to appear in Annals of Applied Probability, arXiv:2209.1373

J. Ding & Z. Li, *A polynomial-time iterative algorithm for random graph matching with non-vanishing correlation*, arXiv:2306.00266.

J. Ding, H. Du & Z. Li, *Low-degree hardness of detection for correlated Erdős-Rényi graphs*, arXiv:2311.15931

Open problems and future directions

- Rigorous evidences for statistical-computational gaps
- Beyond Erdős–Rényi graphs:
  - Random geometric graph matching [Wang-Wu-Xu-Yolou' 22, …]

$$A = \Big(\kappa(x_i, x_j)\Big), \quad B = \Big(\kappa(y_i, y_j)\Big), \quad \text{where } (x_{\pi(i)}, y_i) \overset{iid}{\sim} P$$

  - Community recovery and graph matching in correlated stochastic block models [Racz-Sridhar' 21, Gaudio-Racz-Sridhar '22]
    - Matching preferential attachment graphs [Korula-Lanttanzi '14] or power-law graphs [Yu-Xu-Lin '21]
- Seeded Graph matching with initial "noisy" matched pairs [Kazemi-Hassani-Grossglausser '15, Lubars-Srikant '18, Mossel-Xu '20, Yu-Xu-Lin '20]
- Graph matching with node attribute information [Zhang-Wang-Wang-Wang '20]
- "Robust" graph matching [Ameen-Hajek ' 23]
- Random graph matching with multiple graphs [Ameen-Hajek ' 24]
- Database alignment/record linkage problem [Dai-Cullina-Kiyavash '19, …]

MLE: Quadratic Assignment Problem (QAP)

- Maximum likelihood estimator:

$$\pi_{\mathrm{MLE}} = \arg\max_{\hat{\pi}} \sum_{i<j} A_{ij} B_{\hat{\pi}(i)\hat{\pi}(j)} \quad \text{(QAP)}$$

Suppose $A, B \sim$ correlated Erdős–Rényi graph model under a hidden node mapping $\hat{\pi}$.

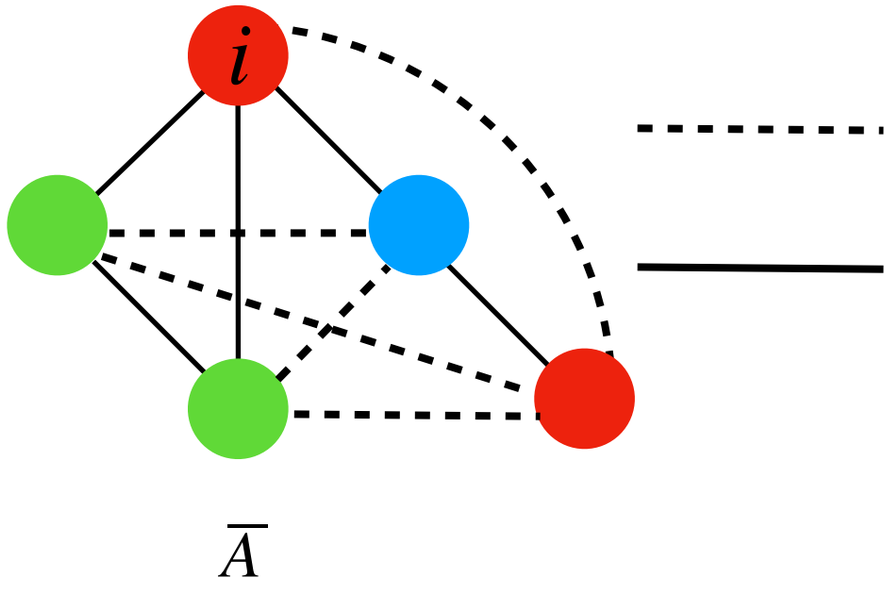
The likelihood function is

$$\begin{aligned}
\mathscr{P}(A, B \mid \hat{\pi}) = \; & (q(q+\rho-q\rho))^{\sum_{i<j} A_{ij} B_{\hat{\pi}(i)\hat{\pi}(j)}} \\
& \Big(q(1-q)(1-\rho)\Big)^{\sum_{i<j} A_{ij}(1-B_{\hat{\pi}(i)\hat{\pi}(j)}) + \sum_{i<j}(1-A_{ij}) B_{\hat{\pi}(i)\hat{\pi}(j)}} \\
& \Big(1-q(2-q-\rho+q\rho)\Big)^{\sum_{i<j}(1-A_{ij})(1-B_{\hat{\pi}(i)\hat{\pi}(j)})} \\
\propto \; & \left(\frac{(q+\rho-q\rho)(1-q(2-q-\rho+q\rho))}{q(1-q)^2(1-\rho)^2}\right)^{\sum_{i<j} A_{ij} B_{\hat{\pi}(i)\hat{\pi}(j)}} .
\end{aligned}$$

Then, we have

$$\mathrm{MLE}_{\pi} = \arg\max_{\hat{\pi}} \mathscr{P}(A, B \mid \hat{\pi}) = \arg\max \sum_{i<j} A_{ij} B_{\hat{\pi}(i)\hat{\pi}(j)} .$$

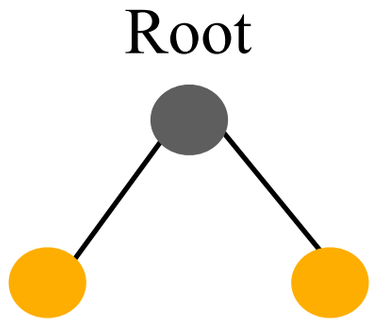
Approximately count signed trees via color coding



$\overline{A}$

$-q$

$1-q$

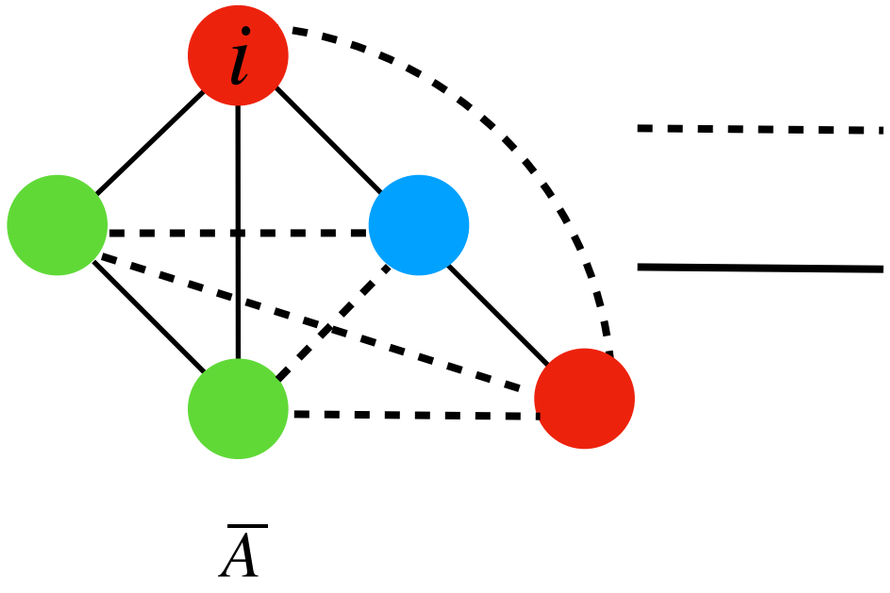


N edges

Rooted tree H

- Exhaustive search takes n^N times: super poly-time when $N \to \infty$
- Solution: approximate count in $n^2 e^{O(N)}$ time via color coding [Alon-Yuster-Zwick '95]
  1. Assign random color μ to each vertex from color set $[N+1]$ uniformly

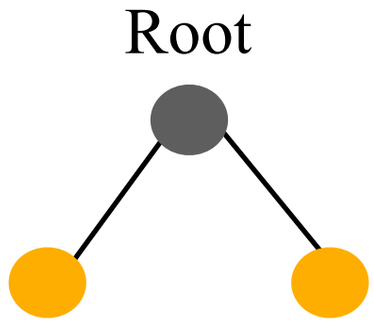
Approximately count signed trees via color coding



$\overline{A}$

$-q$

$1-q$



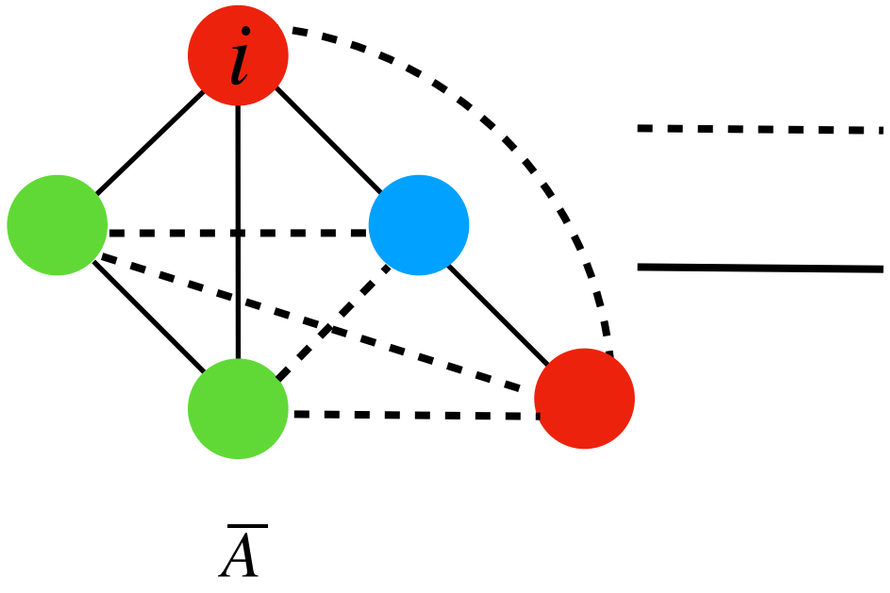
N edges

Rooted tree H

- Exhaustive search takes n^N times: super poly-time when $N \to \infty$
- Solution: approximate count in $n^2 e^{O(N)}$ time via color coding [Alon-Yuster-Zwick '95]
  1. Assign random color μ to each vertex from color set $[N+1]$ uniformly
  2. Count colorful copies of H (all vertices have distinct colors) $\Longrightarrow X_{i,H}(\overline{A}, \mu)$

$$\mathbb{E}_{\mu}[X_{i,H}(\overline{A}, \mu)] = rW_{i,H}(\overline{A}), \quad \text{where } r = (N+1)!/(N+1)^{N+1}$$

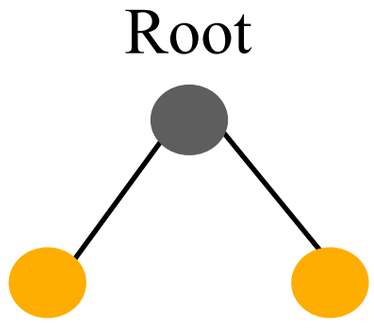
Approximately count signed trees via color coding



$\overline{A}$

$-q$

$1-q$



N edges

Rooted tree H

- Exhaustive search takes n^N times: super poly-time when $N \to \infty$
- Solution: approximate count in $n^2 e^{O(N)}$ time via color coding [Alon-Yuster-Zwick '95]

1. Assign random color μ to each vertex from color set $[N+1]$ uniformly
2. Count colorful copies of H (all vertices have distinct colors) $\Longrightarrow X_{i,H}(\overline{A}, \mu)$

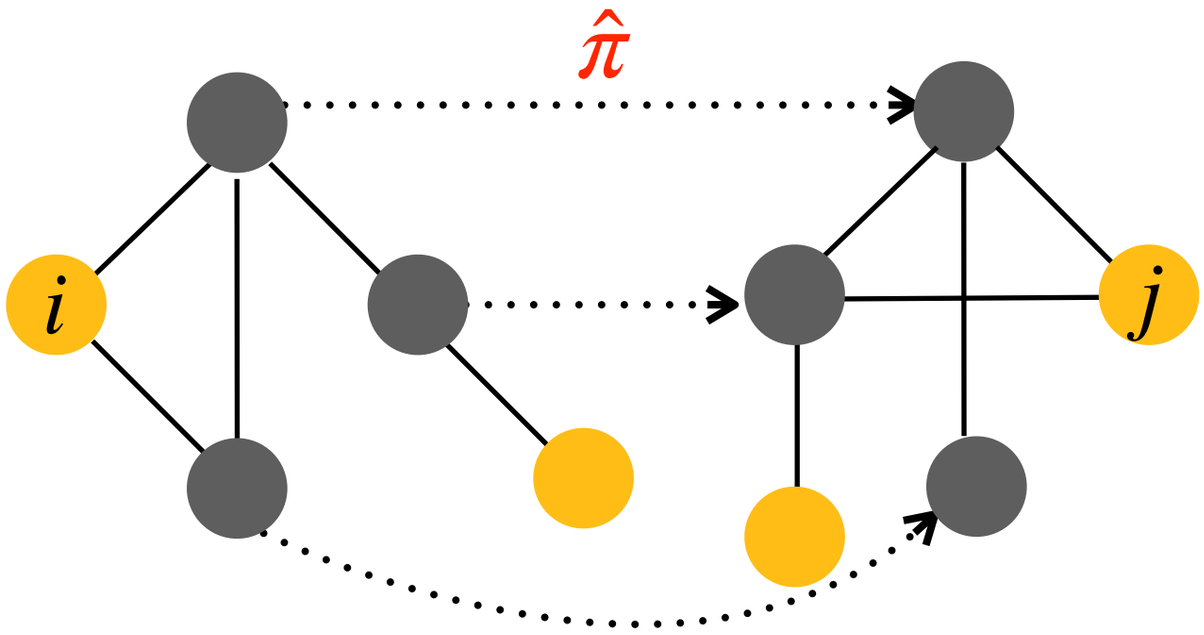
$$\mathbb{E}_{\mu}[X_{i,H}(\overline{A}, \mu)] = rW_{i,H}(\overline{A}), \quad \text{where } r = (N+1)!/(N+1)^{N+1}$$

3. Generate $1/r$ independent random colorings μ_t so that

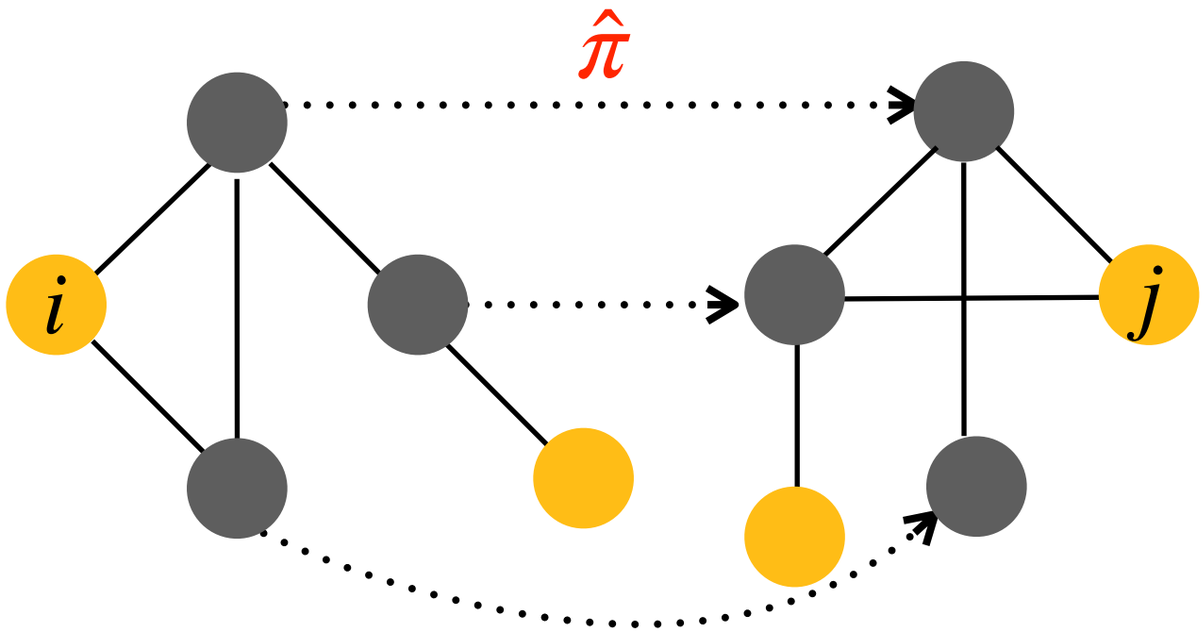
$$W_{i,H}(\overline{A}) \approx \sum_{t=1}^{1/r} X_{i,H}(\overline{A}, \mu_t)$$

Boost almost exact recovery to exact recovery

Boost almost exact recovery to exact recovery



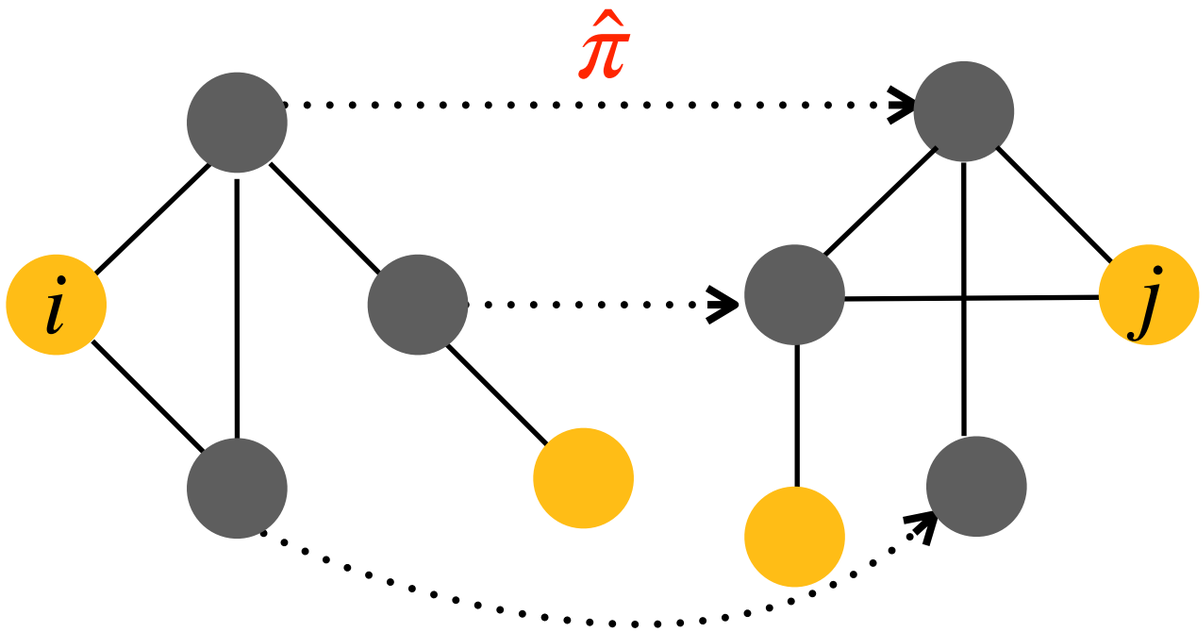
Boost almost exact recovery to exact recovery



Seeded graph matching (SGM)

1. For each unmatched pair (i, j), compute "common" neighbors $N(i, j)$ under $\hat{\pi}$
2. If $N(i, j) \geq \gamma$, match (i, j), append it to $\hat{\pi}$, and repeat

Boost almost exact recovery to exact recovery

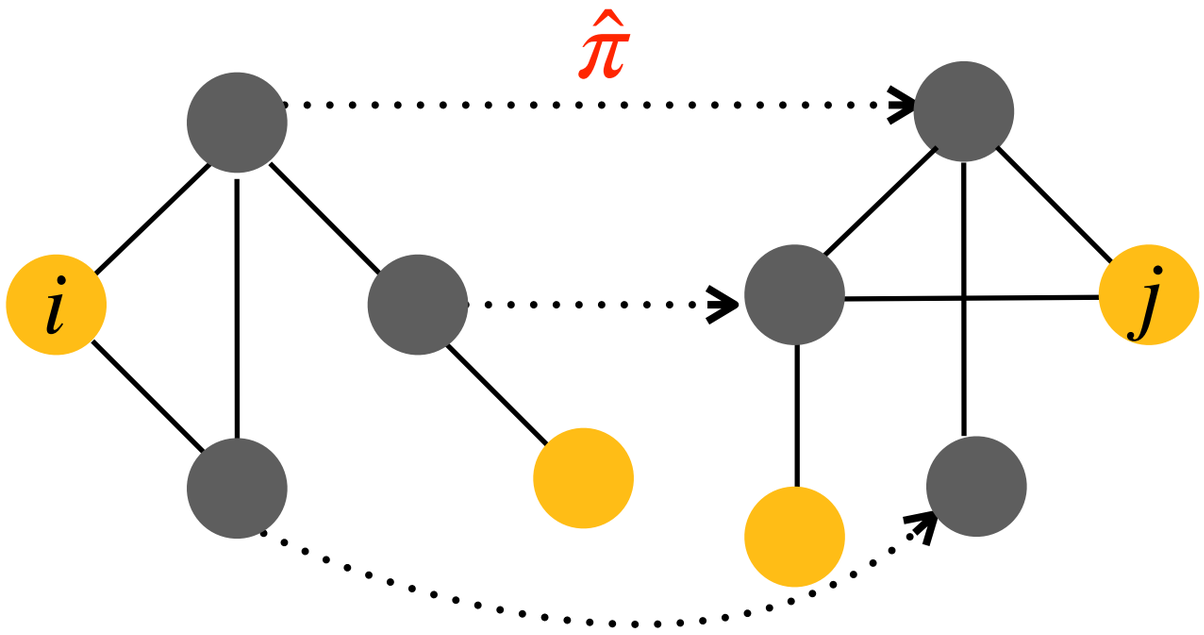


Seeded graph matching (SGM)

1. For each unmatched pair (i, j), compute "common" neighbors $N(i, j)$ under $\hat{\pi}$
2. If $N(i, j) \geq \gamma$, match (i, j), append it to $\hat{\pi}$, and repeat

Similar to percolation graph matching [Yarteva-Grossglauser'13, Barak-Chou-Lei-Schramm-Sheng '19]

Boost almost exact recovery to exact recovery



Seeded graph matching (SGM)

1. For each unmatched pair (i, j), compute "common" neighbors $N(i, j)$ under $\hat{\pi}$
2. If $N(i, j) \geq \gamma$, match (i, j), append it to $\hat{\pi}$, and repeat

Similar to percolation graph matching [Yarteva-Grossglauser'13, Barak-Chou-Lei-Schramm-Sheng '19]

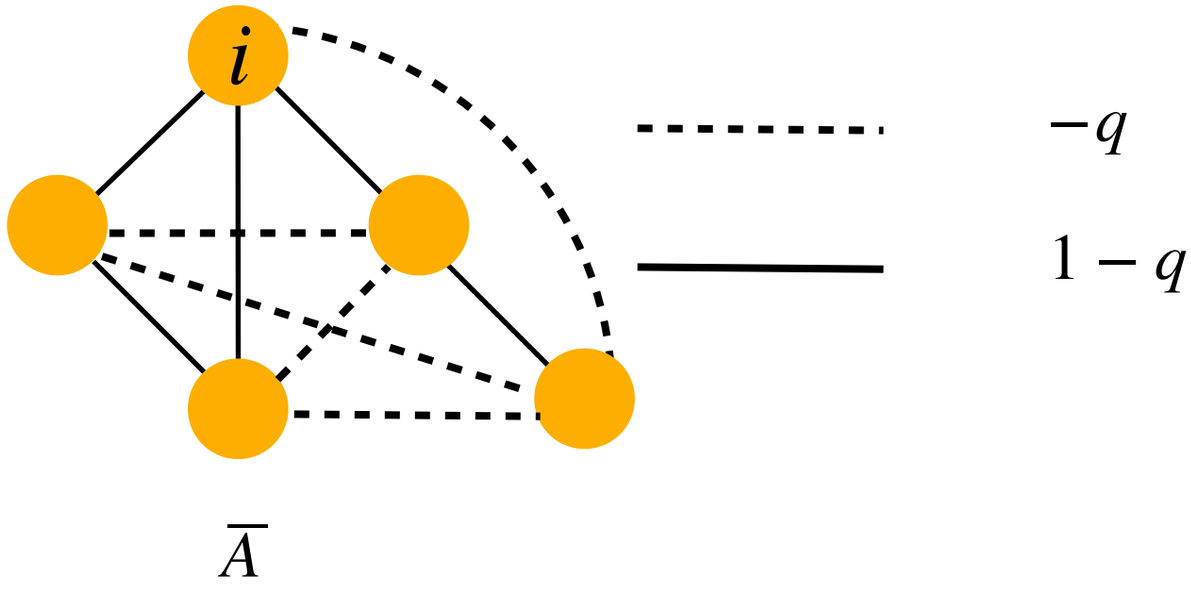
Theorem [Mao-Wu-X.-Yu '22]

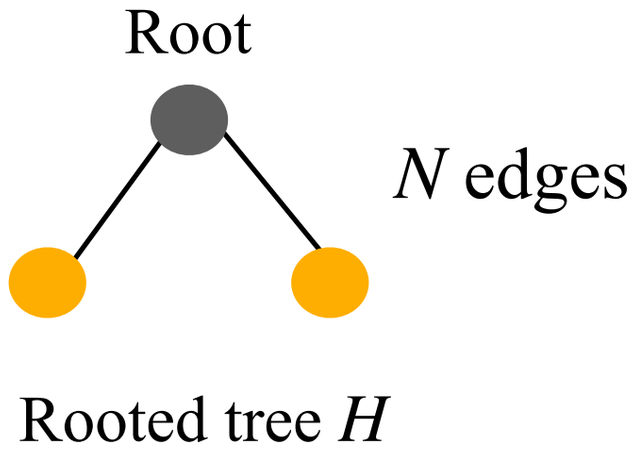
Suppose $nq(q + \rho(1 - q)) \geq (1 + \epsilon)\log n$ and $\rho \geq \epsilon$. With probability $1 - o(1)$, given any input $\hat{\pi}$ that completely coincides with π on at least $(1 - \epsilon)n$ vertices, SGM with an appropriate choice of threshold γ outputs $\tilde{\pi} = \pi$ in time $O(n^3 q^2)$.

- Proof: Intersection graph is an expander, so SGM iteratively matches all vertices

Color coding

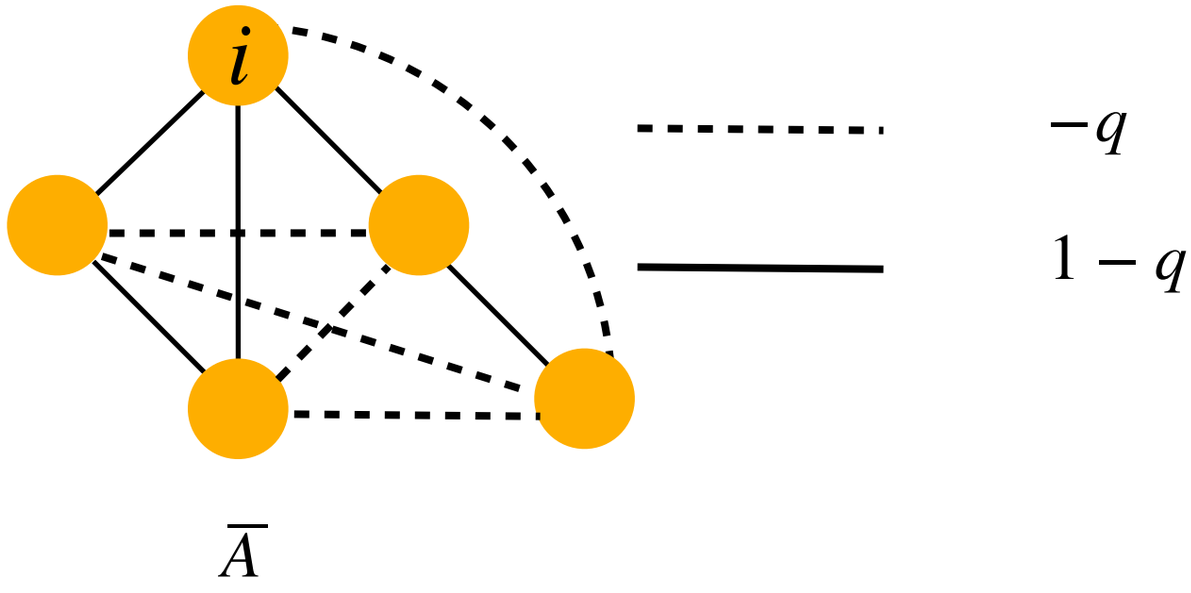
Counting signed trees via color coding

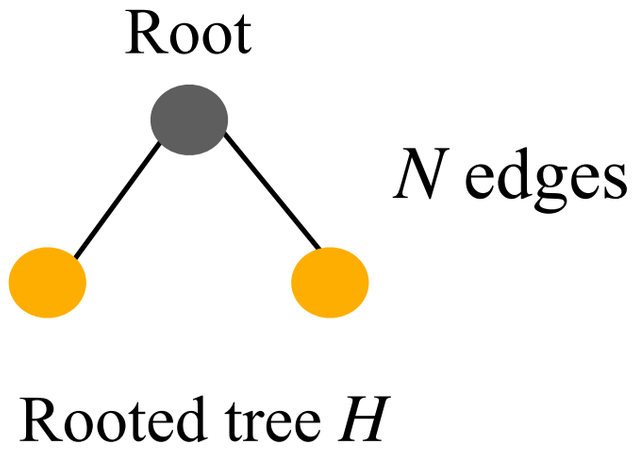




- Exhaustive search takes n^N times: super poly-time when $N \to \infty$

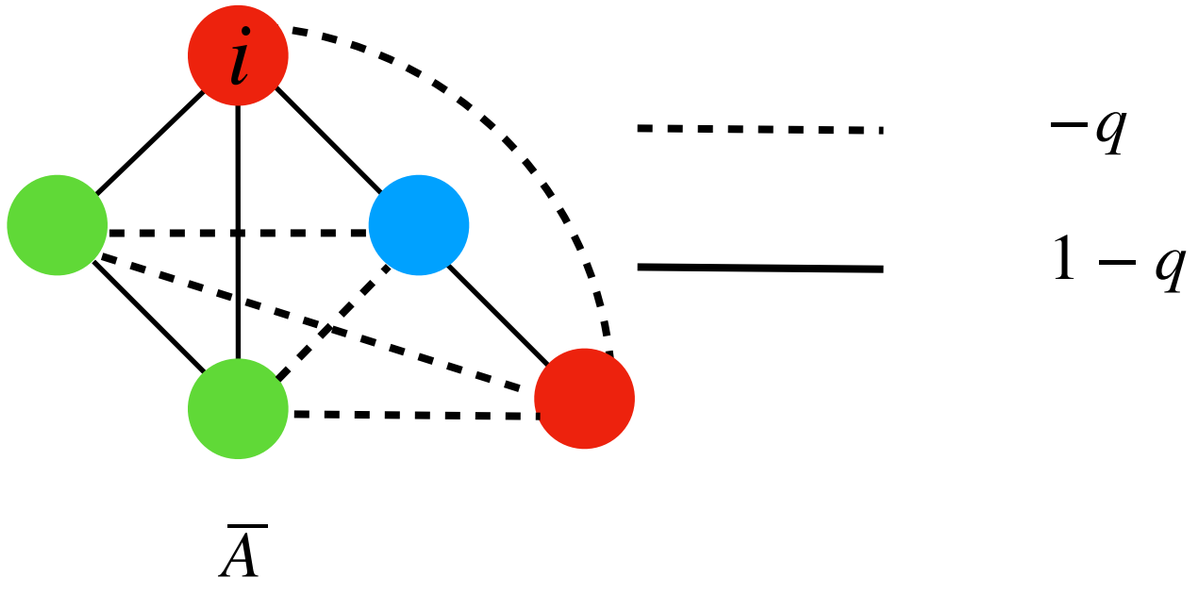
Counting signed trees via color coding



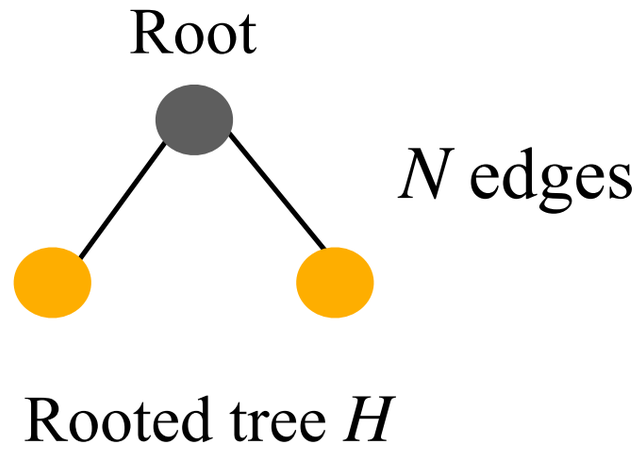


- Exhaustive search takes n^N times: super poly-time when $N \to \infty$
- Our solution: color coding [Alon-Yuster-Zwick '95]

Counting signed trees via color coding



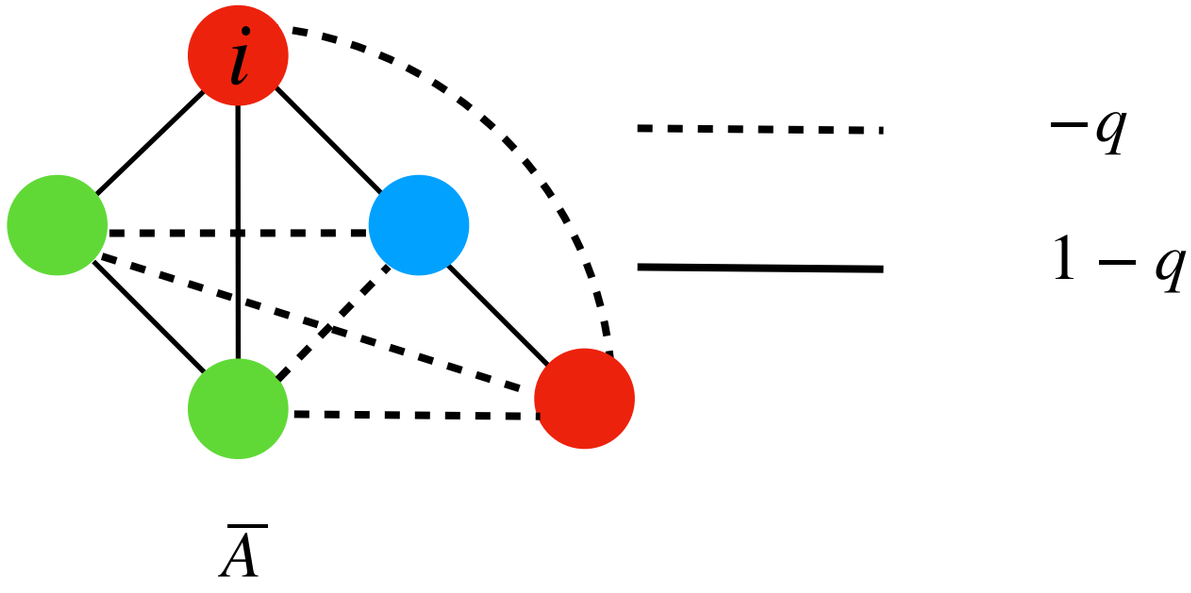
$\overline{A}$

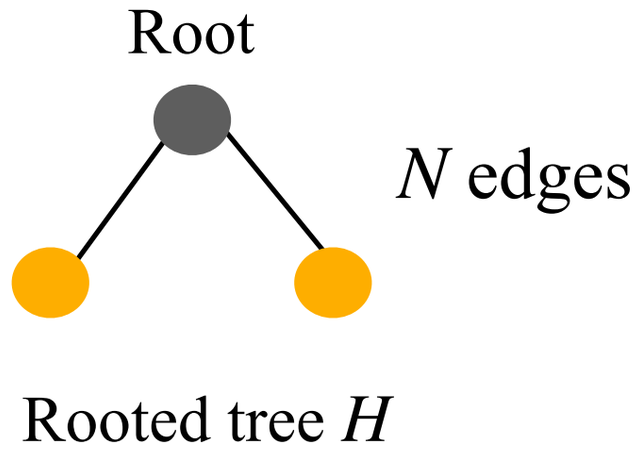


Rooted tree H

- Exhaustive search takes n^N times: super poly-time when $N \to \infty$
- Our solution: color coding [Alon-Yuster-Zwick '95]
  1. Assign random color μ to each vertex from color set $[N + 1]$ uniformly

Counting signed trees via color coding

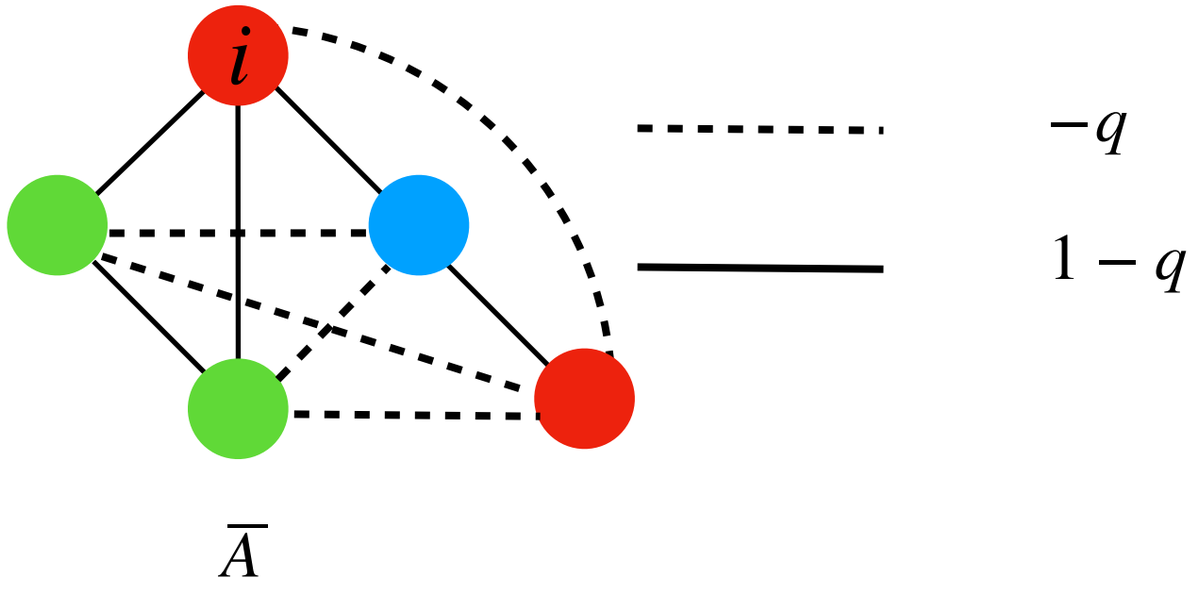


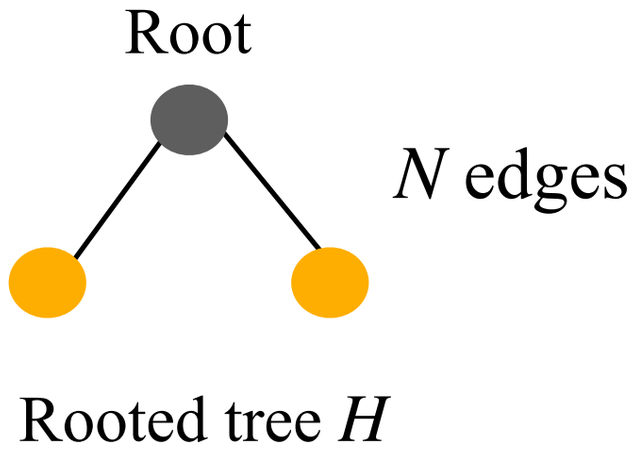


- Exhaustive search takes n^N times: super poly-time when $N \to \infty$
- Our solution: color coding [Alon-Yuster-Zwick '95]
  1. Assign random color μ to each vertex from color set $[N+1]$ uniformly
  2. Count colorful copies of H (all vertices have distinct colors) $\Longrightarrow X_{i,H}(\overline{A}, \mu)$

$$\mathbb{E}_{\mu}[X_{i,H}(\overline{A}, \mu)] = rW_{i,H}(\overline{A}), \text{ where } r = (K+1)!/(K+1)^{K+1}$$

Counting signed trees via color coding





- Exhaustive search takes n^N times: super poly-time when $N \to \infty$
- Our solution: color coding [Alon-Yuster-Zwick '95]

1. Assign random color μ to each vertex from color set $[N+1]$ uniformly
2. Count colorful copies of H (all vertices have distinct colors) $\Longrightarrow X_{i,H}(\overline{A}, \mu)$

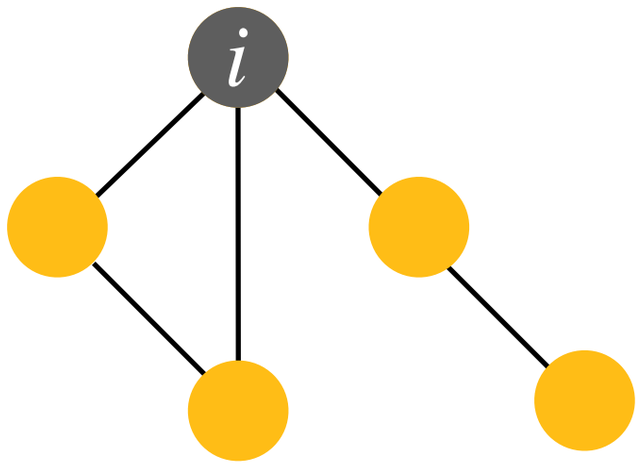
$$\mathbb{E}_{\mu}[X_{i,H}(\overline{A}, \mu)] = rW_{i,H}(\overline{A}), \text{ where } r = (K+1)!/(K+1)^{K+1}$$

3. Generate $1/r$ independent random colorings μ_t so that

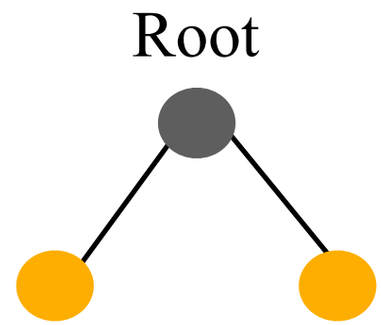
$$W_{i,H}(\overline{A}) \approx \sum_{t=1}^{1/r} X_{i,H}(\overline{A}, \mu_t)$$

Signed rooted tree count

Rooted **signed** subgraph count

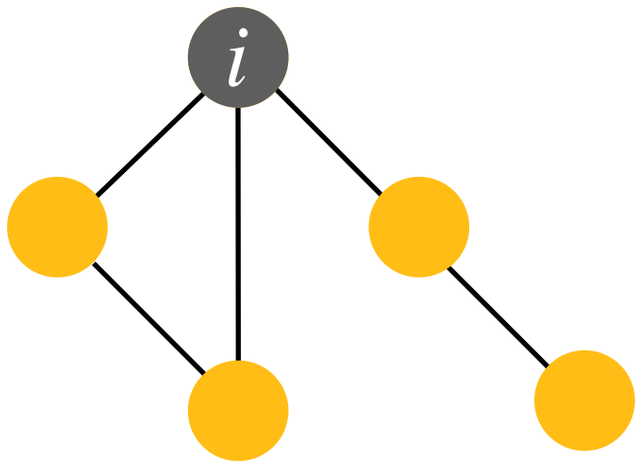


A

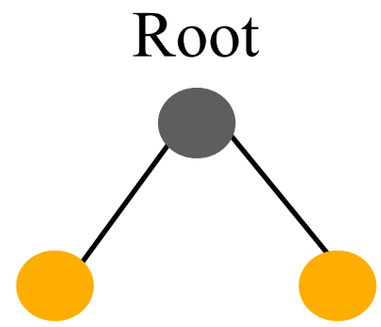


Rooted subgraph H

Rooted **signed** subgraph count



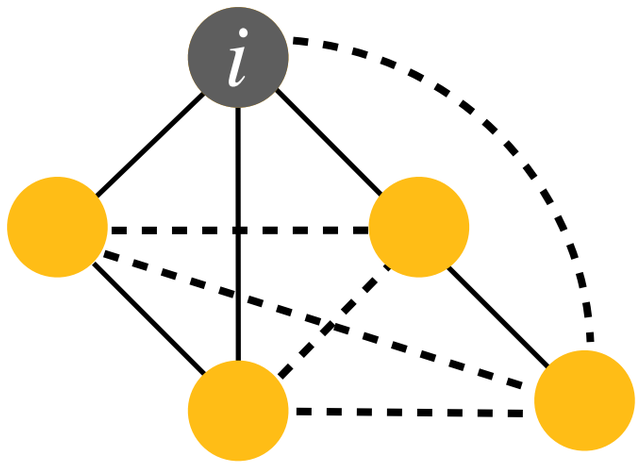
A

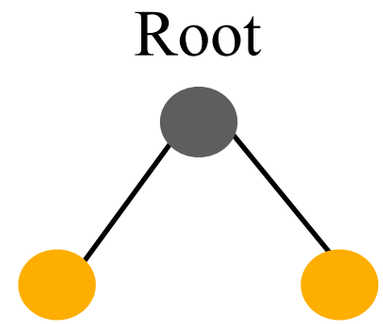


Rooted subgraph H

- Centered adjacency matrices: $\overline{A} = A - \mathbb{E}[A]$;

Rooted **signed** subgraph count





Rooted subgraph H

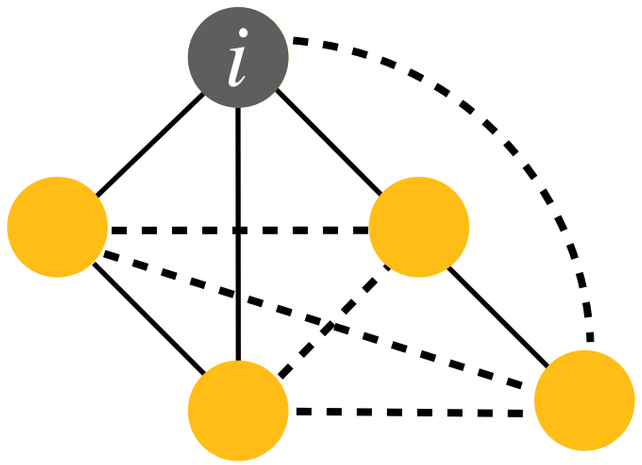
A

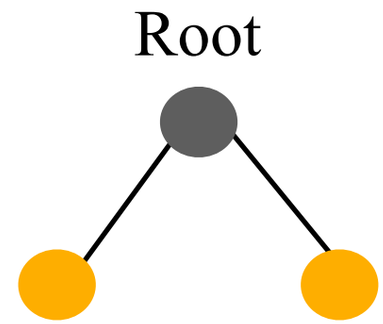
$-q$

$1-q$

- Centered adjacency matrices: $\overline{A} = A - \mathbb{E}[A]$;

Rooted **signed** subgraph count





Rooted subgraph H

A

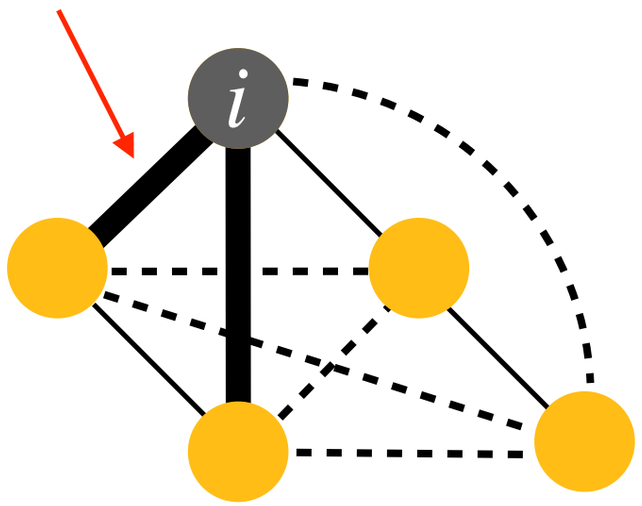
$-q$

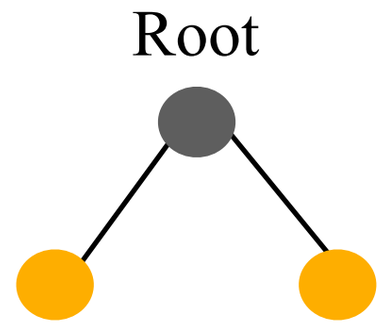
$1-q$

- Centered adjacency matrices: $\overline{A} = A - \mathbb{E}[A]$;
- Count weighted copies of rooted graph in $\overline{A} \implies W_{i,H}(\overline{A})$ (rooted **signed** subgraph count) [Bubeck-Ding-Eldan-Rácz '16];

Rooted **signed** subgraph count

This copy of H has a weight $(1-q)^2$.





Rooted subgraph H

A

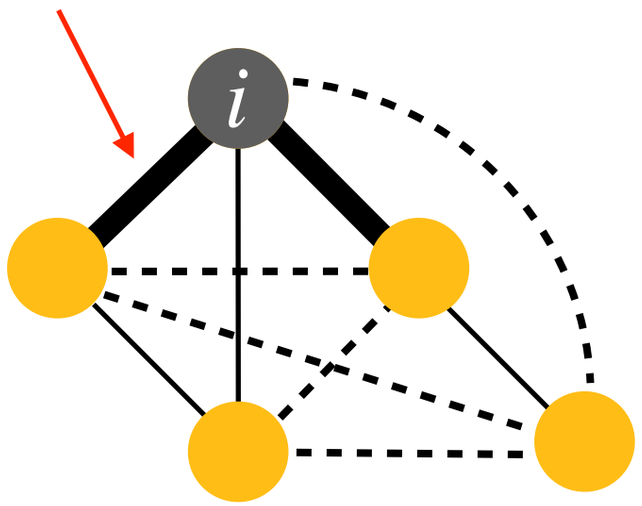
$-q$

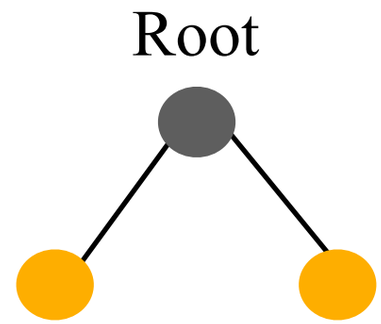
$1-q$

- Centered adjacency matrices: $\overline{A} = A - \mathbb{E}[A]$;
- Count weighted copies of rooted graph in $\overline{A} \implies W_{i,H}(\overline{A})$ (rooted **signed** subgraph count) [Bubeck-Ding-Eldan-Rácz '16];

Rooted **signed** subgraph count

This copy of H has a weight $(1-q)^2$.





Rooted subgraph H

A

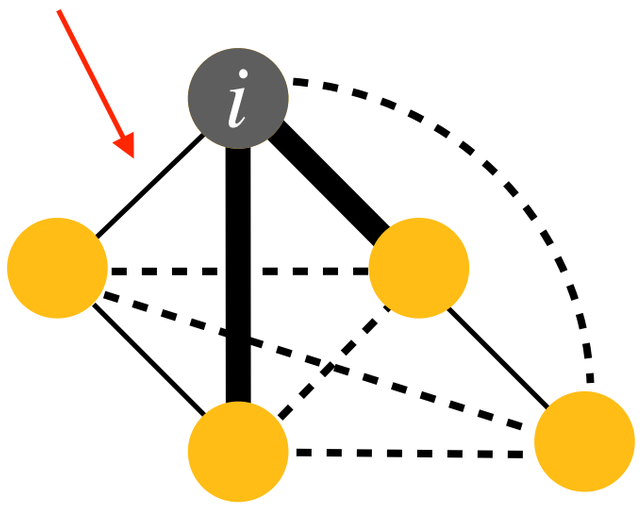
- - - - - - - - $-q$

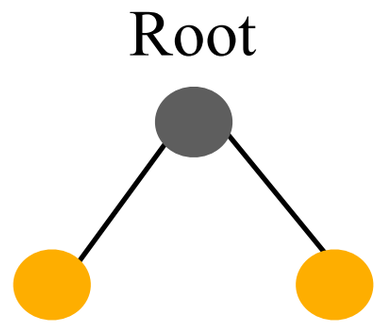
——— $1-q$

- Centered adjacency matrices: $\overline{A} = A - \mathbb{E}[A]$;
- Count weighted copies of rooted graph in $\overline{A} \implies W_{i,H}(\overline{A})$ (rooted **signed** subgraph count) [Bubeck-Ding-Eldan-Rácz '16];

Rooted **signed** subgraph count

This copy of H has a weight $(1-q)^2$.





Rooted subgraph H

A

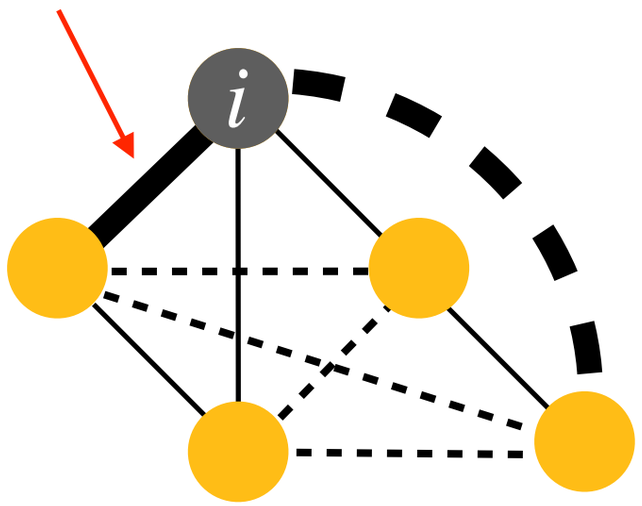
- - - - - - - - $-q$

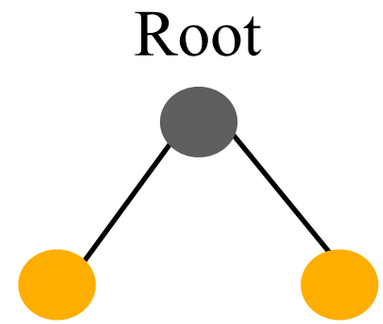
——— $1-q$

- Centered adjacency matrices: $\overline{A} = A - \mathbb{E}[A]$;
- Count weighted copies of rooted graph in $\overline{A} \implies W_{i,H}(\overline{A})$ (rooted **signed** subgraph count) [Bubeck-Ding-Eldan-Rácz '16];

Rooted **signed** subgraph count

This copy of H has a weight $(1-q)(-q)$.





Rooted subgraph H

A

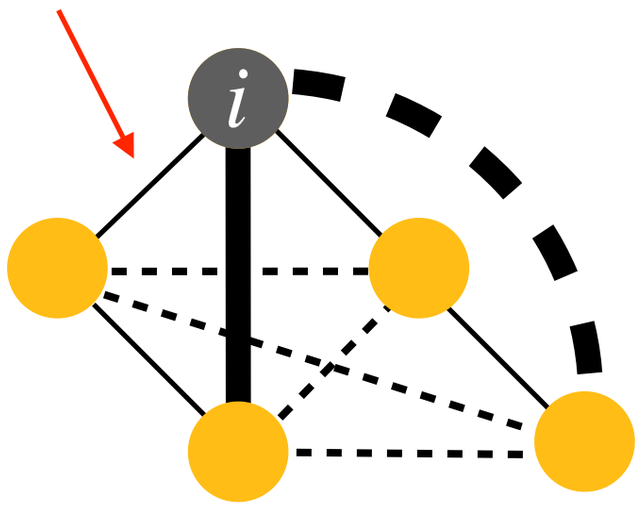
$-q$

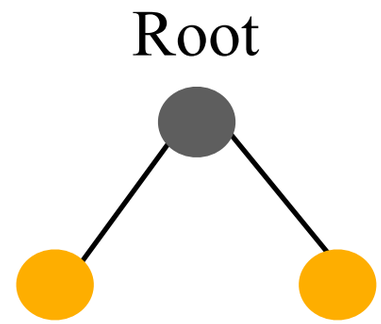
$1-q$

- Centered adjacency matrices: $\overline{A} = A - \mathbb{E}[A]$;
- Count weighted copies of rooted graph in $\overline{A} \implies W_{i,H}(\overline{A})$ (rooted **signed** subgraph count) [Bubeck-Ding-Eldan-Rácz '16];

Rooted **signed** subgraph count

This copy of H has a weight $(1-q)(-q)$.





Rooted subgraph H

A

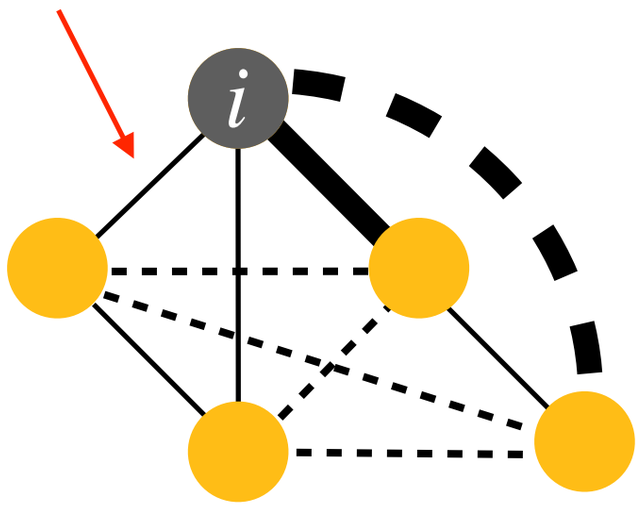
$-q$

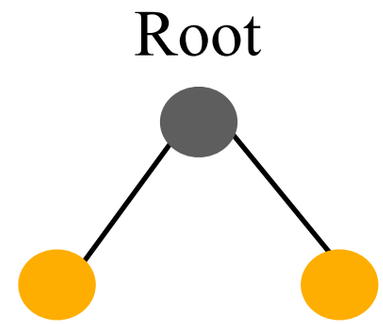
$1-q$

- Centered adjacency matrices: $\overline{A} = A - \mathbb{E}[A]$;
- Count weighted copies of rooted graph in $\overline{A} \implies W_{i,H}(\overline{A})$ (rooted **signed** subgraph count) [Bubeck-Ding-Eldan-Rácz '16];

Rooted **signed** subgraph count

This copy of H has a weight $(1-q)(-q)$.





Rooted subgraph H

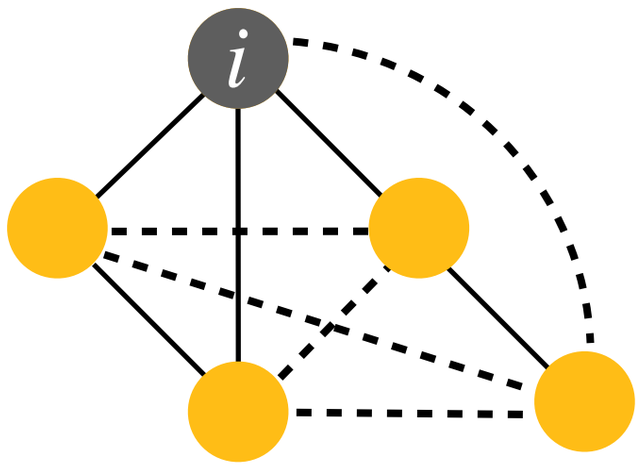
A

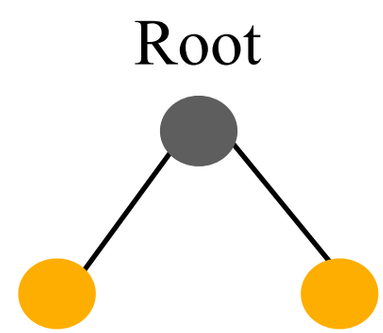
$-q$

$1-q$

- Centered adjacency matrices: $\overline{A} = A - \mathbb{E}[A]$;
- Count weighted copies of rooted graph in $\overline{A} \implies W_{i,H}(\overline{A})$ (rooted **signed** subgraph count) [Bubeck-Ding-Eldan-Rácz '16];

Rooted **signed** subgraph count





Rooted subgraph H

$W_{i,H}(\overline{A}) = 3(1-q)^2 + 3(1-q)(-q)$

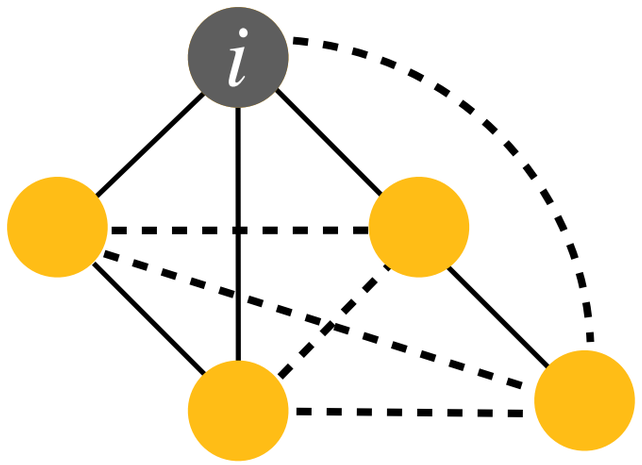
$\overline{A}$

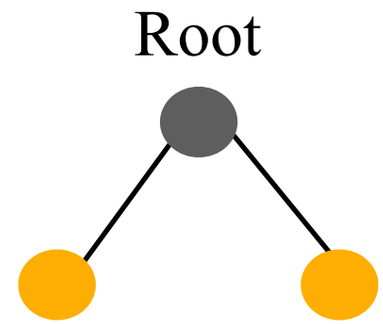
$-q$

$1-q$

- Centered adjacency matrices: $\overline{A} = A - \mathbb{E}[A]$;
- Count weighted copies of rooted graph in $\overline{A} \implies W_{i,H}(\overline{A})$ (rooted **signed** subgraph count) [Bubeck-Ding-Eldan-Rácz '16];

Rooted **signed** subgraph count





Rooted subgraph H

A

$-q$

$1-q$

- Centered adjacency matrices: $\overline{A} = A - \mathbb{E}[A]$;
- Count weighted copies of rooted graph in $\overline{A} \implies W_{i,H}(\overline{A})$ (rooted **signed** subgraph count) [Bubeck-Ding-Eldan-Rácz '16];
- Crucially, $W_{i,H}(\overline{A})$ and $W_{i,I}(\overline{A})$ are uncorrelated for distinct subgraphs H and I.