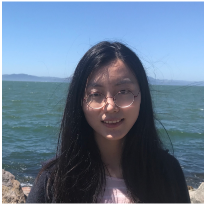
Moments, Random Walks, and Limits for Spectrum Approximation

COLT 2023, Bangalore, India

Apoorv Vikram Singh
(NYU)



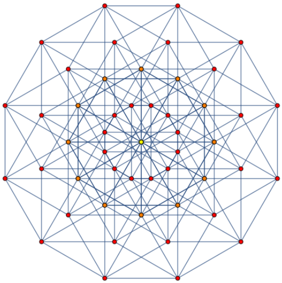
Yujia Jin
(Stanford)

Christopher Musco
(NYU)

Aaron Sidford
(Stanford)

Problem Setup

- Weighted graph G, with normalized adjacency matrix A



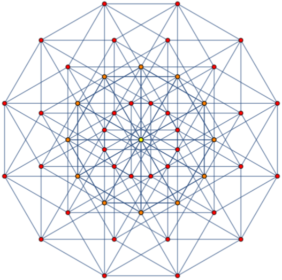
- Spectrum of $G \equiv$ Eigenvalues of A:

$$-1 \leq \lambda_n \leq \cdots \leq \lambda_1 = 1.$$

Goal: "Estimate" the spectrum of G.

Problem Setup

- Weighted graph G, with normalized adjacency matrix A



- Spectrum of $G \equiv$ Eigenvalues of A

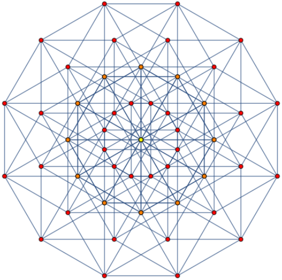
exactly compute $\approx n^{2.35}$

$$-1 \leq \lambda_n \leq \cdots \leq \lambda_1 = 1.$$

Goal: "Estimate" the spectrum of G.

Problem Setup

- Weighted graph G, with normalized adjacency matrix A



- Spectrum of $G \equiv$ Eigenvalues of A:

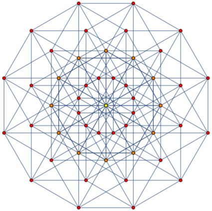
$$-1 \leq \lambda_n \leq \cdots \leq \lambda_1 \leq 1.$$

sublinear time: $o(n^2)$

Goal: "Estimate" the spectrum of G.

Spectral Density of G

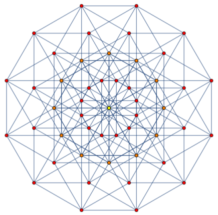
(Graph G with normalized adjacency matrix A)



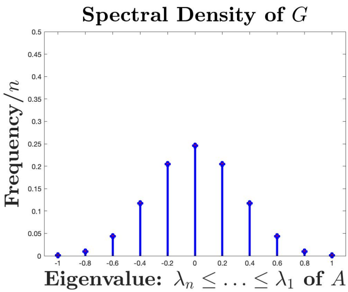
Graph G

Spectral Density of G

(Graph G with normalized adjacency matrix A)



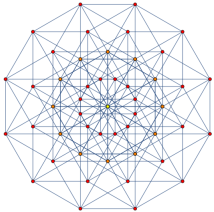
Graph G



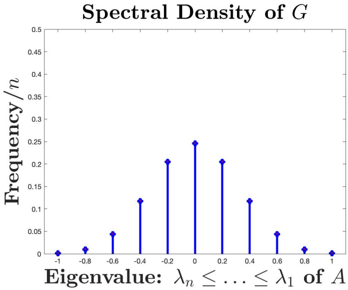
Spectral Density p

Spectral Density of G

(Graph G with normalized adjacency matrix A)



Graph G

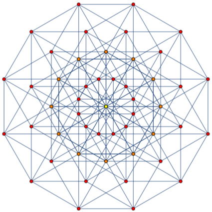


Spectral Density p

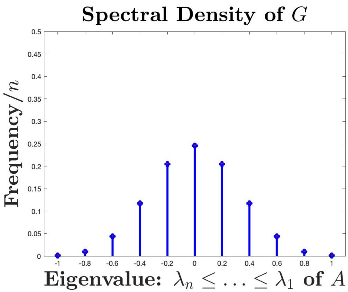
Goal: Output q such that Wasserstein-1 distance: $W_1(p, q) \leq \varepsilon$.

Spectral Density of G

(Graph G with normalized adjacency matrix A)



Graph G



Spectral Density p

Goal: Output q such that Wasserstein-1 distance: $W_1(p,q) \le \varepsilon$.

Equivalently: Output $\lambda'_n \le \cdots \le \lambda'_1$ such that

$$\frac{1}{n}\sum_{i=1}^{n} |\lambda_i - \lambda'_i| \le \varepsilon.$$

Existing Results

Given G with spectral density p and random-walk access to G :
random node and random neighbour

Random start node $+$ Transition to a random neighbour with probability $\propto$ edge weight.

Existing Results

Given G with spectral density p and random-walk access to G :
random node and random neighbour

Random start node $+$ Transition to a random neighbour with probability $\propto$ edge weight.

- [Cohen-Steiner et al. (2018)] Output q: $W_1(p,q) \leq \varepsilon$.
  - Takes $2^{\mathcal{O}(1/\varepsilon)}$ steps of random walks on G.
  - Running time: $2^{\mathcal{O}(1/\varepsilon)}$.

Existing Results

Given G with spectral density p and random-walk access to G :
random node and random neighbour

Random start node $+$ Transition to a random neighbour with probability $\propto$ edge weight.

- [Cohen-Steiner et al. (2018)] Output q: $W_1(p, q) \leq \varepsilon$.
  - Takes $2^{\mathcal{O}(1/\varepsilon)}$ steps of random walks on G.
  - Running time: $2^{\mathcal{O}(1/\varepsilon)}$.

Our Results $\Rightarrow$ [CKSV'18] cannot be improved.

Goal: Learn Distribution p

Explaining [CKSV'18]

Goal: Learn Distribution p

Explaining [CKSV'18]

- **Idea:** Estimate the moments m_j of the distributon p.

$$m_j = \mathbb{E}_{X \sim p}[X^j].$$

- **Question:** How many moments and to what accuracy?

Goal: Learn Distribution p

Explaining [CKSV'18]

- **Idea:** Estimate the moments m_j of the distributon p.

$$m_j = \mathbb{E}_{X \sim p}[X^j].$$

- **Question:** How many moments and to what accuracy?

[Kong and Valiant (2017)]: Estimate $m_1, \ldots, m_{1/\varepsilon}$ moments of p to accuracy $\pm 2^{-\mathcal{O}(1/\varepsilon)}$ to output q such that $W_1(p, q) \leq \varepsilon$.

Goal: Learn Distribution p

Explaining [CKSV'18]

- **Idea:** Estimate the moments m_j of the distributon p.

$$m_j = \mathbb{E}_{X \sim p}[X^j].$$

- **Question:** How many moments and to what accuracy?

[Kong and Valiant (2017)]: Estimate $m_1, \ldots, m_{1/\varepsilon}$ moments of p to accuracy $\pm 2^{-\mathcal{O}(1/\varepsilon)}$ to output q such that $W_1(p, q) \leq \varepsilon$.

Our Results $\Rightarrow$ [KV'17] is tight!

Goal: Learn Distribution p

Explaining [CKSV'18]

- **Idea:** Estimate the moments m_j of the distributon p.

$$m_j = \mathbb{E}_{X \sim p}[X^j].$$

- **Question:** How many moments and to what accuracy?

[Kong and Valiant (2017)]: Estimate $m_1, \ldots, m_{1/\varepsilon}$ moments of p to accuracy $\pm 2^{-\mathcal{O}(1/\varepsilon)}$ to output q such that $W_1(p, q) \leq \varepsilon$.

- **Aim:** Estimate $m_1, \ldots, m_{1/\varepsilon}$ of p to accuracy $\pm 2^{-\mathcal{O}(1/\varepsilon)}$.

Our Results $\Rightarrow$ [KV'17] is tight!

Goal: Learn Distribution p

Explaining [CKSV'18]

- **Idea:** Estimate the moments m_j of the distributon p.

$$m_j = \mathbb{E}_{X \sim p}[X^j].$$

- **Question:** How many moments and to what accuracy?

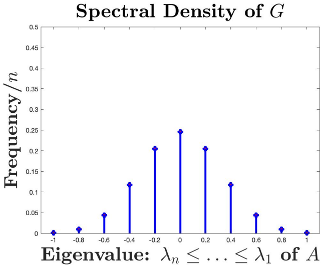
[Kong and Valiant (2017)]: Estimate $m_1, \ldots, m_{1/\varepsilon}$ moments of p to accuracy $\pm 2^{-\mathcal{O}(1/\varepsilon)}$ to output q such that $W_1(p, q) \leq \varepsilon$.

- **Aim:** Estimate $m_1, \ldots, m_{1/\varepsilon}$ of p to accuracy $\pm 2^{-\mathcal{O}(1/\varepsilon)}$.
- **Question:** How to estimate moments of Graph spectrum p? What are your samples?

Our Results $\Rightarrow$ [KV'17] is tight!

Moments and Random Walks

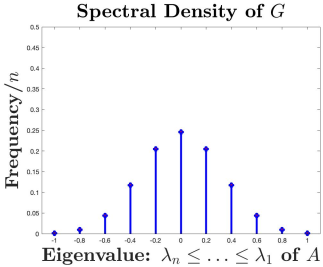
Graph G with normalized Adjacency A.



Distribution p

Moments and Random Walks

Graph G with normalized Adjacency A.

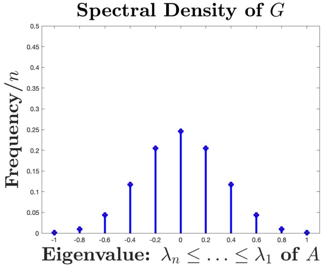


Distribution p

- $m_j := \mathbb{E}_{X\sim p}[X^j] = \frac{1}{n}\sum_{i=1}^{n} \lambda_i^j.$

Moments and Random Walks

Graph G with normalized Adjacency A.

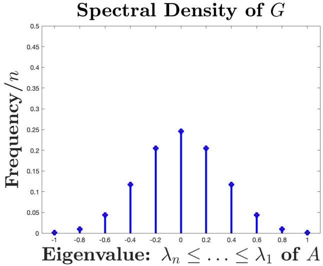


Distribution p

- $m_j := \mathbb{E}_{X\sim p}[X^j] = \frac{1}{n}\sum_{i=1}^n \lambda_i^j$.
- $\mathrm{Tr}(A^j) = \sum_{i=1}^n \lambda_i^j$.
- $\mathrm{Tr}(A^j) = \sum_{i=1}^n \mathbb{P}\left[j\text{-step rw from } i \text{ to } i\right]$.

Moments and Random Walks

Graph G with normalized Adjacency A.



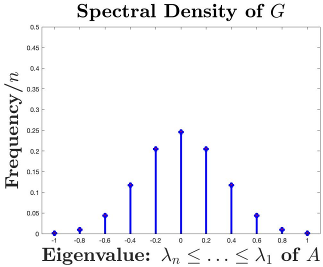
Distribution p

- $m_j := \mathbb{E}_{X\sim p}[X^j] = \frac{1}{n}\sum_{i=1}^{n} \lambda_i^j$.
- $\mathrm{Tr}(A^j) = \sum_{i=1}^{n} \lambda_i^j$.
- $\mathrm{Tr}(A^j) = \sum_{i=1}^{n} \mathbb{P}\left[j\text{-step rw from } i \text{ to } i\right]$.

j-th moment of $p \equiv$
$\mathbb{E}_i\left[\Pr\left(\text{walk of length } j \text{ ends at } i \mid \text{start at } i\right)\right]$.

Moments and Random Walks

Graph G with normalized Adjacency A.



Distribution p

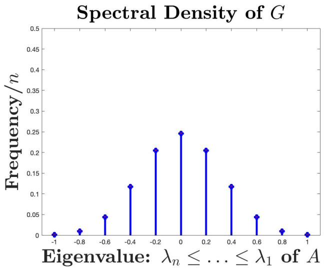
- $m_j := \mathbb{E}_{X\sim p}[X^j] = \frac{1}{n}\sum_{i=1}^n \lambda_i^j$.
- $\mathrm{Tr}(A^j) = \sum_{i=1}^n \lambda_i^j$.
- $\mathrm{Tr}(A^j) = \sum_{i=1}^n \mathbb{P}\,[j\text{-step rw from } i \text{ to } i]$.

$$j\text{-th moment of } p \equiv$$
$$\mathbb{E}_i\left[\Pr\left(\text{walk of length } j \text{ ends at } i \mid \text{start at } i\right)\right].$$

1. Choose a random node i
2. Take a walk of length j
3. Set $X = 1$ if it returns to i, 0 otherwise.

Moments and Random Walks

Graph G with normalized Adjacency A.



Distribution p

- $m_j := \mathbb{E}_{X \sim p}[X^j] = \frac{1}{n} \sum_{i=1}^{n} \lambda_i^j$.
- $\mathrm{Tr}(A^j) = \sum_{i=1}^{n} \lambda_i^j$.
- $\mathrm{Tr}(A^j) = \sum_{i=1}^{n} \mathbb{P}\,[j\text{-step rw from } i \text{ to } i]$.

j-th moment of $p \equiv$
$\mathbb{E}_i\ [\Pr\ (\text{walk of length } j \text{ ends at } i \mid \text{start at } i)]$.

1. Choose a random node i
2. Take a walk of length j
3. Set $X = 1$ if it returns to i, 0 otherwise.

$\mathbb{E}[X] = m_j$

[CKSV'18] Algorithm

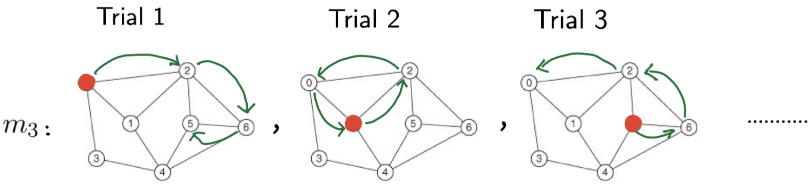
$$m_j = \mathbb{E}_i \left[\Pr \left(\text{walk of length } j \text{ ends at } i \mid \text{start at } i \right) \right].$$

- For each moment $j = 1, \ldots, 1/\varepsilon$
  - Take $2^{\mathcal{O}(1/\varepsilon)}$ trials of rw of length j, starting at random node i.
  - Output the fraction of walks that return to the starting random node.

[CKSV'18] Algorithm

$m_j = \mathbb{E}_i$ [Pr (walk of length j ends at i | start at i)].

- For each moment $j = 1, \ldots, 1/\varepsilon$
  - Take $2^{\mathcal{O}(1/\varepsilon)}$ trials of rw of length j, starting at random node i.
  - Output the fraction of walks that return to the starting random node.



$m_3 \approx$ (# of returns)/(# of trials).

Need $2^{\mathcal{O}(1/\varepsilon)}$ trials because we want accuracy $\pm 2^{-\mathcal{O}(1/\varepsilon)}$.

Scope of Improvement?

Q1 Can we get away with lower accuracy in moment estimation?
([CKSV'18] needed first $1/\varepsilon$ moments to accuracy $2^{-\mathcal{O}(1/\varepsilon)}$)

Scope of Improvement?

Q1 Can we get away with lower accuracy in moment estimation?
([CKSV'18] needed first $1/\varepsilon$ moments to accuracy $2^{-\mathcal{O}(1/\varepsilon)}$)

Q2 Does higher moments help? I.e., Estimate more moments, each to lower accuracy.
(Estimate more than $1/\varepsilon$ moments to lower accuracy)

Scope of Improvement?

Q1 Can we get away with lower accuracy in moment estimation?
([CKSV'18] needed first $1/\varepsilon$ moments to accuracy $2^{-\mathcal{O}(1/\varepsilon)}$)

Q2 Does higher moments help? I.e., Estimate more moments, each to lower accuracy.
(Estimate more than $1/\varepsilon$ moments to lower accuracy)

Q3 Does any other random walk based algorithm works better?
([CKSV'18] considers fraction of walks that return to starting node)

Scope of Improvement?

Q1 Can we get away with lower accuracy in moment estimation?
([CKSV'18] needed first $1/\varepsilon$ moments to accuracy $2^{-\mathcal{O}(1/\varepsilon)}$)

Q2 Does higher moments help? I.e., Estimate more moments, each to lower accuracy.
(Estimate more than $1/\varepsilon$ moments to lower accuracy)

Q3 Does any other random walk based algorithm works better?
([CKSV'18] considers fraction of walks that return to starting node)

Our Results: NO!

Our Results

Lower bound for learning distributions.

Limits of Moment Based Methods

There exists distributions that cannot be approximated to accuracy ε in W_1 distance even if **all** of their moments are known to *multiplicative* accuracy $\left(1 \pm 2^{-\Omega(1/\varepsilon)}\right)$.

Our Results

Lower bound for learning distributions.

Limits of Moment Based Methods

There exists distributions that cannot be approximated to accuracy ε in W_1 distance even if **all** of their moments are known to *multiplicative* accuracy $\left(1 \pm 2^{-\Omega(1/\varepsilon)}\right)$.

Lower bound for learning graph spectrum.

Limit of Random-Walk Based Methods

There exists graphs G such that no algorithm based on $2^{\Omega(1/\varepsilon)}$ steps of random walks can approximate the spectrum of G to accuracy less than ε.

Our Results

Lower bound for learning distributions.

Limits of Moment Based Methods

There exists distributions that cannot be approximated to accuracy ε in W_1 distance even if **all** of their moments are known to *multiplicative* accuracy $\left(1 \pm 2^{-\Omega(1/\varepsilon)}\right)$.

Lower bound for learning graph spectrum.

Limit of Random-Walk Based Methods

There exists graphs G such that no algorithm based on $2^{\Omega(1/\varepsilon)}$ steps of random walks can approximate the spectrum of G to accuracy less than ε.

Both results are constructive!

Lower Bound Goal

Aim:

- Generate G_1, G_2 that are ε away in W_1 distance, and
- Can't distinguish G_1, G_2 by rw of length $2^{\mathcal{O}(1/\varepsilon)}$ wp $> 3/4$.

Implies:

- Two probability distribution that are far away in W_1 distance but **all** of their moments are exponentially close.

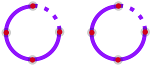
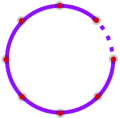
j-th moment of $p \equiv$

$\mathbb{E}_i$ [Pr (walk of length j ends at i | start at i)]

Motivating Example

(Let $\ell = 1/\varepsilon$)

Consider two graphs: Weight of purple edge $= 1/2$.

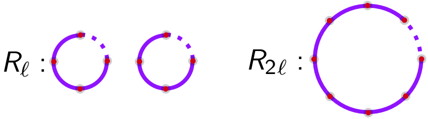
R_ℓ :  $R_{2\ell}$: 

$$\left.\begin{array}{r l} R_\ell : & \text{Two disjoint cycles of length } \ell \\ R_{2\ell} : & \text{Cycle of length } 2\ell \end{array}\right\} W_1(R_\ell, R_{2\ell}) = 1/\ell.$$

Motivating Example

(Let $\ell = 1/\varepsilon$)

Consider two graphs: Weight of purple edge $= 1/2$.



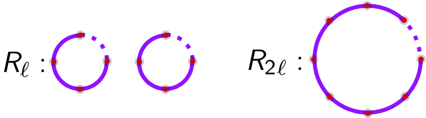
$$\left.\begin{array}{r}R_\ell : \text{Two disjoint cycles of length } \ell \\ R_{2\ell} : \text{Cycle of length } 2\ell\end{array}\right\} W_1(R_\ell, R_{2\ell}) = 1/\ell.$$

- Generate two graphs that are $\Omega(1/\ell) = \Omega(\varepsilon)$ far in W_1 distance and
- Can't distinguish graphs by rw of length $2^{\mathcal{O}(\ell)}$ with constant probability.

Motivating Example

(Let $\ell = 1/\varepsilon$)

Consider two graphs: Weight of purple edge $= 1/2$.



$$\left.\begin{aligned} R_\ell &: \text{Two disjoint cycles of length } \ell \\ R_{2\ell} &: \text{Cycle of length } 2\ell \end{aligned}\right\} W_1(R_\ell, R_{2\ell}) = 1/\ell.$$

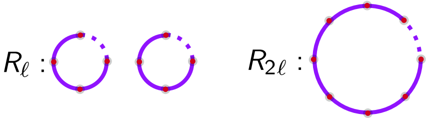
- Generate two graphs that are $\Omega(1/\ell) = \Omega(\varepsilon)$ far in W_1 distance and
- Can't distinguish graphs by rw of length $2^{\mathcal{O}(\ell)}$ with constant probability.

- Length $\ell - 1$ rw cannot distinguish them.

Motivating Example

(Let $\ell = 1/\varepsilon$)

Consider two graphs: Weight of purple edge $= 1/2$.



$$\left.\begin{array}{rl} R_\ell : & \text{Two disjoint cycles of length } \ell \\ R_{2\ell} : & \text{Cycle of length } 2\ell \end{array}\right\} W_1(R_\ell, R_{2\ell}) = 1/\ell.$$

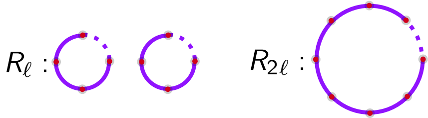
- Generate two graphs that are $\Omega(1/\ell) = \Omega(\varepsilon)$ far in W_1 distance and
- Can't distinguish graphs by rw of length $2^{\mathcal{O}(\ell)}$ with constant probability.

- Length $\ell - 1$ rw cannot distinguish them.
- Length ℓ rw can distinguish with low probability $2^{-\ell}$.

Motivating Example

(Let $\ell = 1/\varepsilon$)

Consider two graphs: Weight of purple edge $= 1/2$.



$$\left.\begin{array}{r l} R_\ell : & \text{Two disjoint cycles of length } \ell \\ R_{2\ell} : & \text{Cycle of length } 2\ell \end{array}\right\} W_1(R_\ell, R_{2\ell}) = 1/\ell.$$

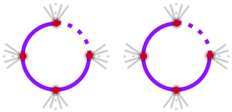
- Generate two graphs that are $\Omega(1/\ell) = \Omega(\varepsilon)$ far in W_1 distance and
- Can't distinguish graphs by rw of length $2^{\mathcal{O}(\ell)}$ with constant probability.

- Length $\ell - 1$ rw cannot distinguish them.
- Length ℓ rw can distinguish with low probability $2^{-\ell}$.
- However, length ℓ^2 rw can distinguish with high probability.

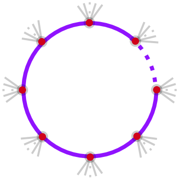
Hypothetical Example

At every step, random walk "disappears" with probability $1/2$.
Weight of purple edges $= 1/4$. Weight of grey "hair" $= 1/2$.

G_1 :

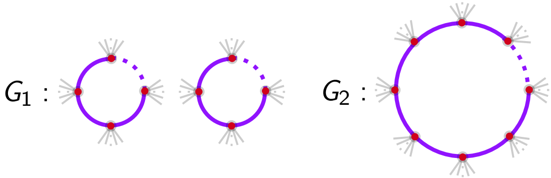


G_2 :



Hypothetical Example

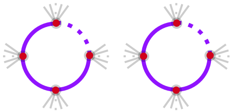
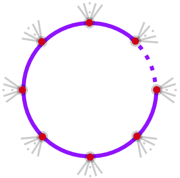
At every step, random walk "disappears" with probability $1/2$.
Weight of purple edges $= 1/4$. Weight of grey "hair" $= 1/2$.



- Suppose spectral density of G_1, G_2 are far in W_1 distance.
- Length $\ell - 1$ rw cannot distinguish them.

Hypothetical Example

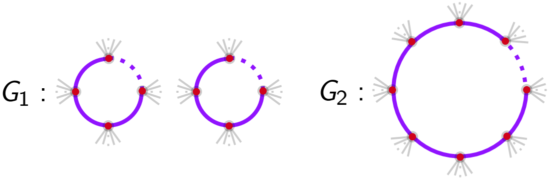
At every step, random walk "disappears" with probability $1/2$.
Weight of purple edges $= 1/4$. Weight of grey "hair" $= 1/2$.

G_1 :  G_2 : 

- Suppose spectral density of G_1, G_2 are far in W_1 distance.
- Length $\ell - 1$ rw cannot distinguish them.
- Length ℓ rw can distinguish with low probability $4^{-\ell}$.

Hypothetical Example

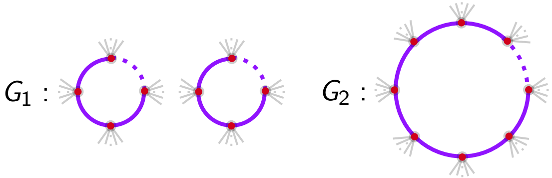
At every step, random walk "disappears" with probability $1/2$.
Weight of purple edges $= 1/4$. Weight of grey "hair" $= 1/2$.



- Suppose spectral density of G_1, G_2 are far in W_1 distance.
- Length $\ell - 1$ rw cannot distinguish them.
- Length ℓ rw can distinguish with low probability $4^{-\ell}$.
- Length ℓ^2 rw can distinguish with low probability $\approx 2^{-\ell}$.

Hypothetical Example

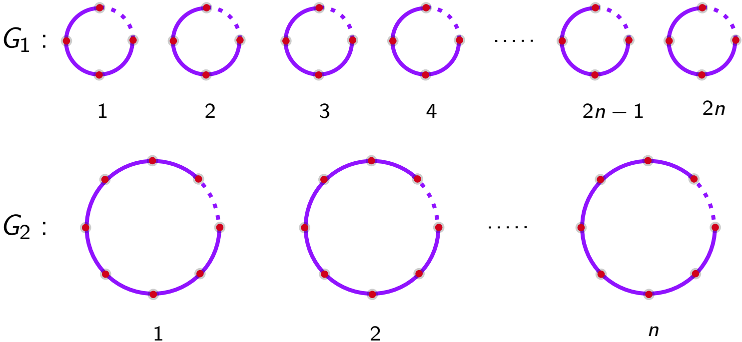
At every step, random walk "disappears" with probability $1/2$.
Weight of purple edges $= 1/4$. Weight of grey "hair" $= 1/2$.



- Suppose spectral density of G_1, G_2 are far in W_1 distance.
- Length $\ell - 1$ rw cannot distinguish them.
- Length ℓ rw can distinguish with low probability $4^{-\ell}$.
- Length ℓ^2 rw can distinguish with low probability $\approx 2^{-\ell}$.
- No random walk can loop around with constant probability!

Making Random Walks "Disappear"

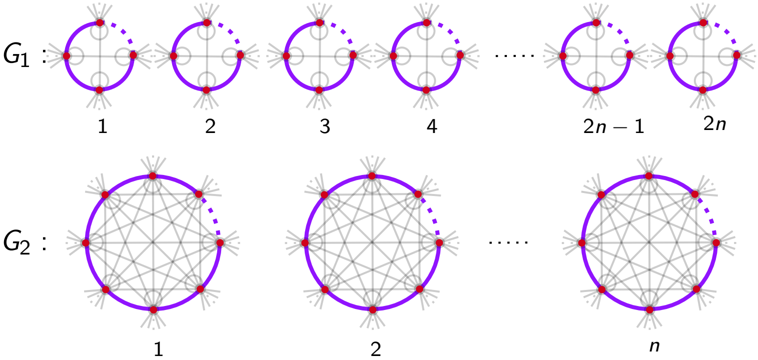
Repeated Cycles +



- $W_1(G_1, G_2) = \varepsilon$.
- Difference of first $1/\varepsilon - 1$ moments is 0.

Making Random Walks "Disappear"

Repeated Cycles + Weighted Complete Graph

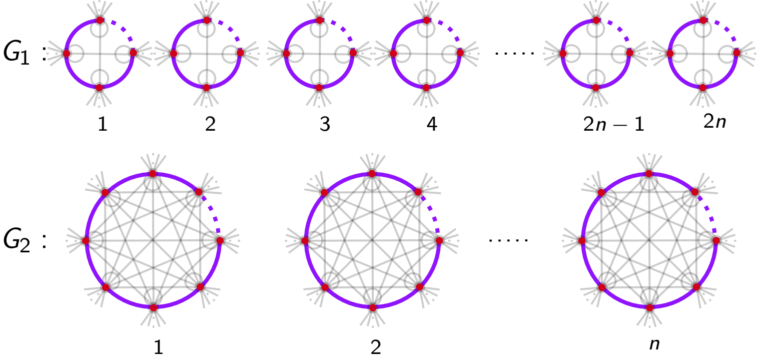


w.p. $1/2$ *rw goes to vertices of the cycle (purple edge), and*
w.p. $1/2$ *rw goes to a random vertex (grey edge)*

- $W_1(G_1, G_2) = \varepsilon/2$.
- Difference of first $1/\varepsilon - 1$ moments is 0.

Making Random Walks “Disappear”

Repeated Cycles + Weighted Complete Graph



w.p. $1/2$ *rw goes to vertices of the cycle (purple edge), and*
w.p. $1/2$ *rw goes to a random vertex (grey edge)*

- $W_1(G_1, G_2) = \varepsilon/2$.
- Difference of first $1/\varepsilon - 1$ moments is 0.
- All other moments differ by at most $2^{-1/\varepsilon}$.

Open Problems

Adaptive Random Walks

- We get tight rates for all random walk based algorithms!
- Starting node of walk was random.

Open Problems

Adaptive Random Walks

- We get tight rates for all random walk based algorithms!
- Starting node of walk was random.
- What if you can specify the starting node of random walk?

Open Problems

Adaptive Random Walks

- We get tight rates for all random walk based algorithms!
- Starting node of walk was random.
- What if you can specify the starting node of random walk?

Adaptive random walks: Specify the starting node of random walk.

(Previously: Starting node was random)

We prove lower bound of $\Omega(1/\varepsilon^2)$ for adaptive random walks.

Open Problems

Adaptive Random Walks

- We get tight rates for all random walk based algorithms!
- Starting node of walk was random.
- What if you can specify the starting node of random walk?

Adaptive random walks: Specify the starting node of random walk.

(Previously: Starting node was random)

We prove lower bound of $\Omega(1/\varepsilon^2)$ for adaptive random walks.

P1: Improve lower bounds in adaptive setting.

P2: Better upper bounds in adaptive setting.

(all existing sublinear time algorithms use non-adaptive random walks)