Quantum-Sampling of Random Spanning Trees

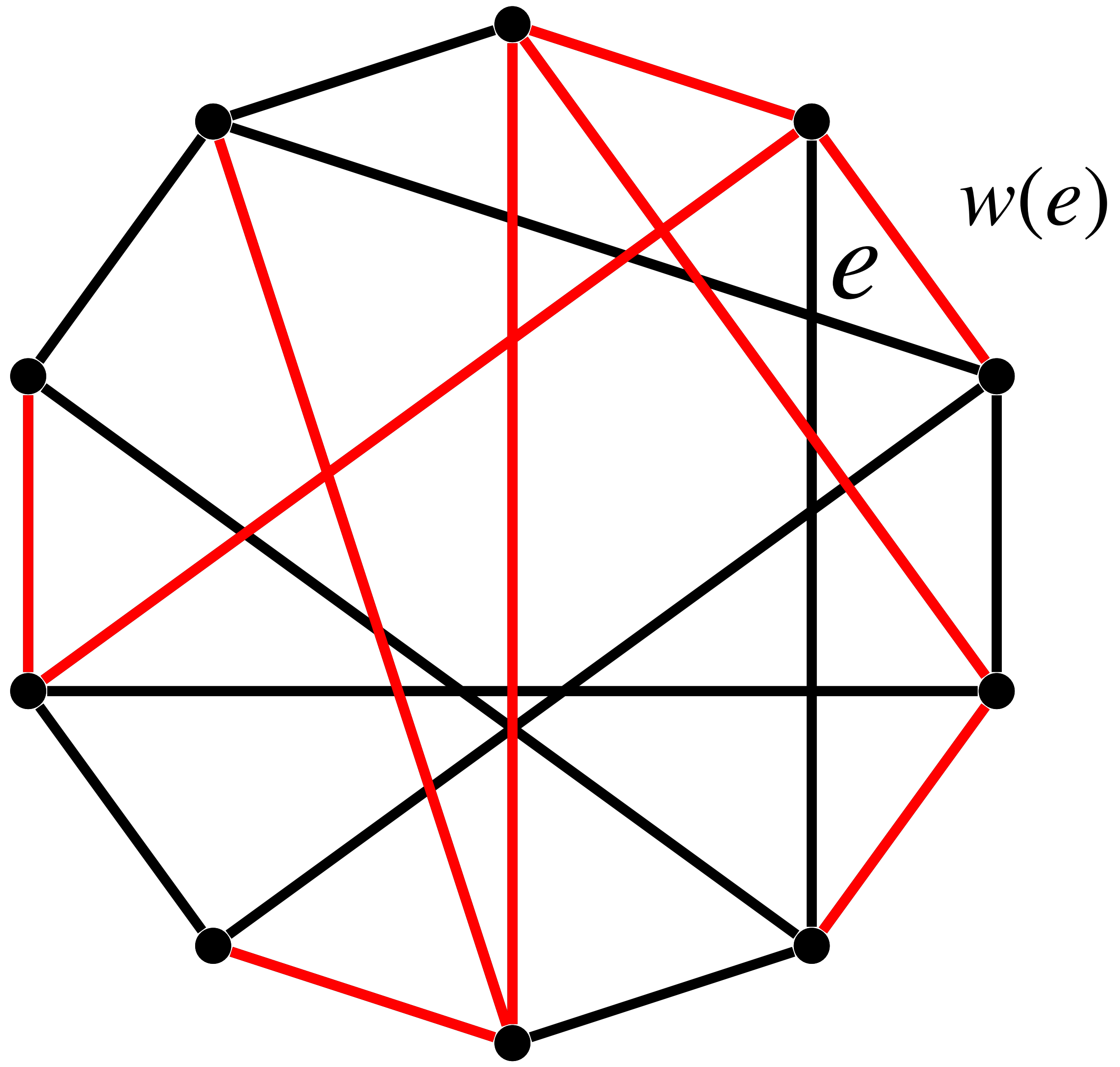
Shrinidhi T S

Based on joint work with *Yassine Hamoudi* and *Adrian Tanasa*

Sampling Spanning Trees

$\pi :=$ Spanning tree distribution of G

$$\pi(T) \propto \prod_{e \in T} w(e)$$



$G(n, m, w)$

Matrix Tree theorem:
$1 \leq \sharp\text{Spanning Trees} \leq n^{n-2}$

How to sample from π efficiently?

Exact sampling

1. Matrix Tree Theorem based:

   Ex: Colbourn et al. (1996) - $O\left(n^{\omega}\right)$ time

2. Random-walk based:

   Ex: Schild (2018) - $m^{1+o(1)}$ time

Approximate sampling

1. Sparsification based:

   Ex: Durfee et al. (2017) - $\tilde{O}\left(n^2 \cdot \epsilon^{-2}\right)$ time

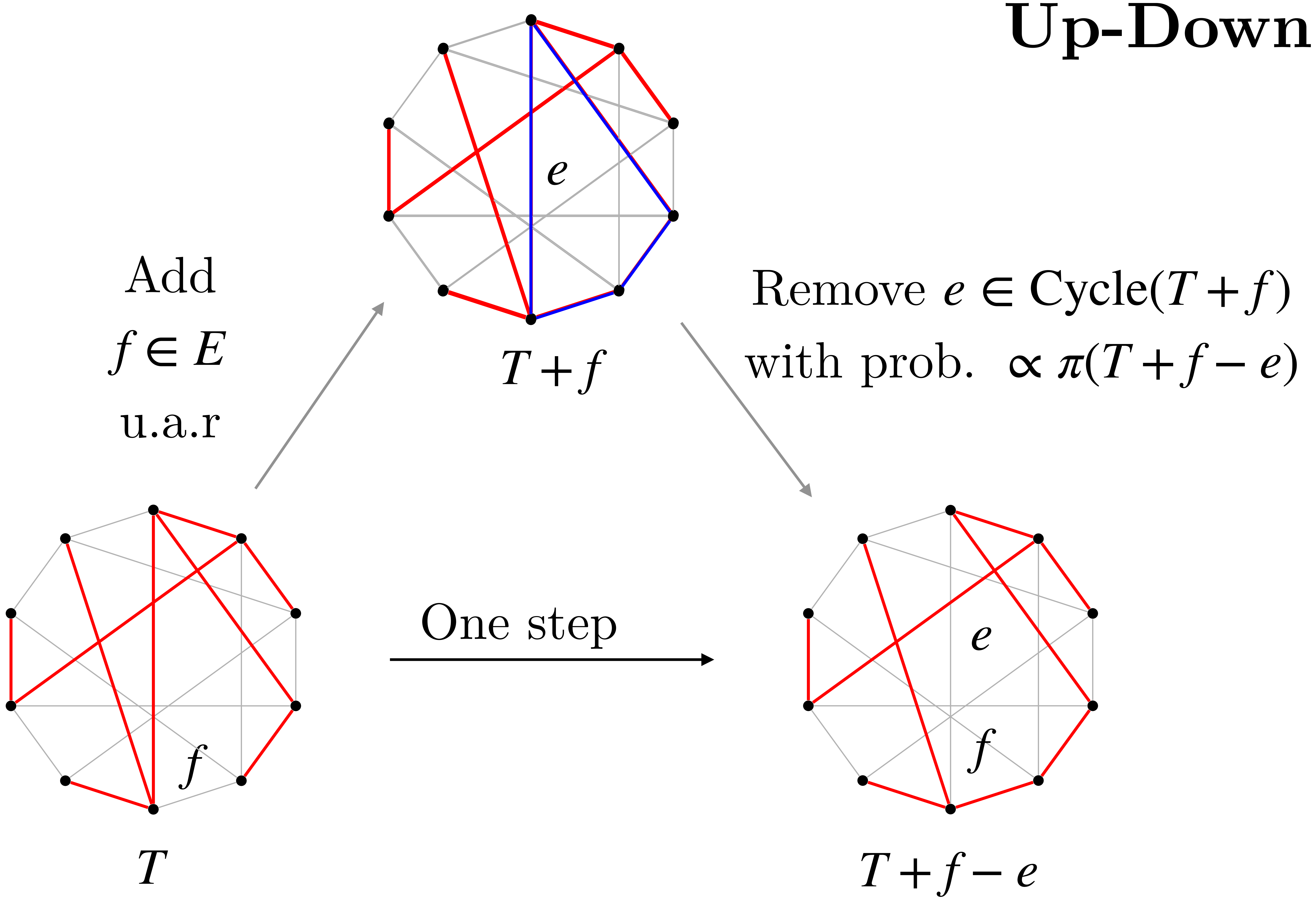
2. Markov chain based:

   Ex: Anari et al. (2020) - $\tilde{O}(m)$ time
   Down-Up and Up-Down walks

$\tilde{O}(\cdot)$ hides poly-logarithmic factors in n

Markov Chains approach and Marginals

Up-Down walk



Marginal of an edge:

$$p_G(e) = \Pr_{T\sim\pi}[e \in T]$$

Probability that an edge e appears in a random $T \sim \pi$

Runtime = mixing time × time per step

$\tilde{O}(m)$ $\quad$ $\tilde{O}(m)$ $\quad\times\quad$ $\tilde{O}(1)$

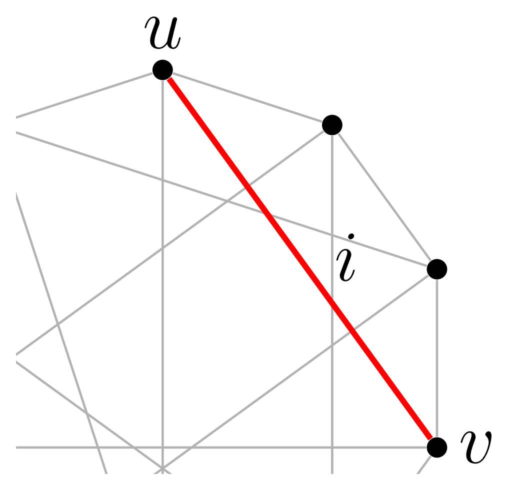
Anari et al. (2020)

Using Link-Cut Trees

Classical Sampling of Random Spanning Trees

Adjacency-list model

G



$\mathscr{O}_G |u\rangle|i\rangle|0\rangle \mapsto |u\rangle|i\rangle|v\rangle$

$\boxed{\text{Can quantum algorithms do better?}}$

Apers, Gao, Ji and Liu (2025)

1. Classical sample using $\tilde{O}\left(\sqrt{mn}\right)$ queries to $\mathscr{O}_G$

Large-step Up-Down walk and bias using marginals **+** Quantum speedup for each step $\longrightarrow$ Classical sample: Description of $T \sim \pi$

$\tilde{O}(1)$ mixing time **×** $\tilde{O}\left(\sqrt{mn}\right)$ queries

$\tilde{O}\left(\sqrt{mn}\right)$ queries

2. $\Omega\left(\sqrt{mn}\right)$ query lower-bound for weighted graphs

Quantum-Sampling of Random Spanning Trees

Quantum sample:
$$|\pi\rangle = \sum_{T \in \Delta} \sqrt{\pi(T)}\,|T\rangle$$

$$\frac{1}{|R|}\sum_{r \in R} |T(r)\rangle\,|r\rangle$$
$$= \sum_{T} \sqrt{\pi(T)}\,|T\rangle\,|g_T\rangle$$
$$\neq |\pi\rangle$$

The hard part:
Uncomputing the garbage $|g_T\rangle$

Can we prepare a quantum sample efficiently?

Why quantum-sample?

Introduced by Aharonov and Ta-Shma (2007)

1. Efficient classical sampler $\not\Rightarrow$ Efficient Q-sample

2. Starting state for Quantum Walk search
$$\tilde{O}\left(S + \frac{1}{\sqrt{\alpha}}\left(\frac{1}{\sqrt{\gamma}}U + C\right)\right)$$
Ex: Element distinctness, MNRS framework etc.,.

3. Can speedup learning of some functions

Ex: Parities, DNF etc.,.

Can solve $G_1 \cong G_2$ if we have $|\mathrm{Aut}(G_1)\rangle$ and $|\mathrm{Aut}(G_2)\rangle$!

Electrical-grid optimisation: minimum-loss spanning trees.

Using Quantum walks to prepare $|\pi\rangle$

Markov chain $\mathscr{P}$

Spectral gap of $\mathscr{P}$: γ

$\xrightarrow{\textbf{Szegedy quantization}}$

Unitary W: phase gap $\Omega\left(1/\sqrt{\gamma}\right)$

Reflection around $|\pi\rangle$: $O(1/\sqrt{\gamma})$ uses of W

Grover based preparation of $|\pi\rangle$:

Start from $|\psi\rangle$, amplify to $|\pi\rangle$ using $\tilde{O}\left(\frac{1}{\sqrt{\gamma}\cdot\langle\psi|\pi\rangle}\right)$ applications of W

$\langle\psi|\pi\rangle$ may be exponentially small

Fast walk $\not\Rightarrow$ fast q-sample preparation

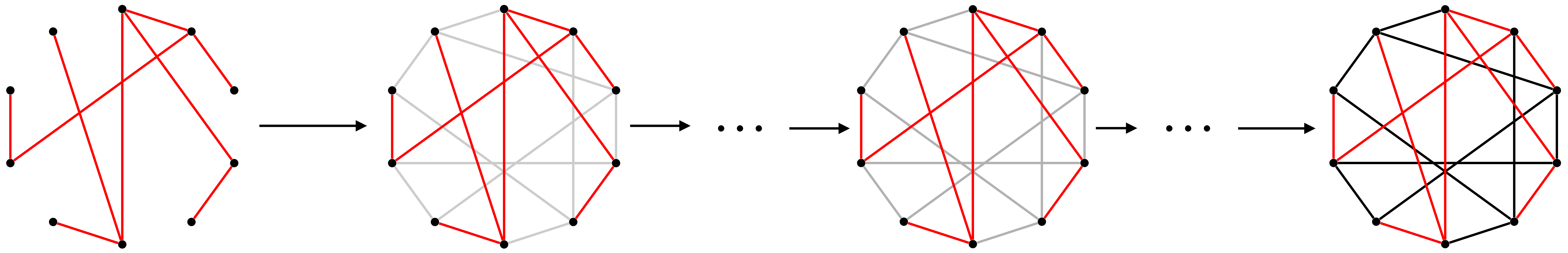
Generally used alternative: simulated annealing

Simulated Annealing

Sequence of slowly changing graphs: from a tree to G

Choose $T_{\max}$ using Apers et al. (2025)

$T_{\max} = G_{\beta_0}$ $\quad$ G_{β_1} $\quad$ G_{β_i} $\quad$ $G_{\beta_l} = G$



$$|T_{\max}\rangle = |\pi_{\beta_0}\rangle \xrightarrow{W_1} |\pi_{\beta_1}\rangle \xrightarrow{W_2} \cdots \xrightarrow{W_i} |\pi_{\beta_i}\rangle \xrightarrow{W_{i+1}} \cdots \xrightarrow{W_l} |\pi_{\beta_l}\rangle = |\pi\rangle$$

$$G_\beta := G \text{ with weights } w_\beta(e) = \begin{cases} w(e), & e \in T_{\max}, \\ e^{-\beta} w(e), & e \notin T_{\max}. \end{cases}$$

Lemma (Short schedule). $\exists$ a sequence $\beta_0 \geq \beta_1 \cdots \geq \beta_l$ with: $\langle \pi_{\beta_i} | \pi_{\beta_{i+1}} \rangle \geq \Omega(1)$ and $l = \tilde{O}\left(\sqrt{n}\right)$

First stop: One quantum sample

Short sequence: $\tilde{O}(\sqrt{n})$ states with $\langle \pi_{\beta_i} | \pi_{\beta_{i+1}} \rangle = \Omega(1)$

Naive Up-Down walk quantization:

W_i can be implemented in $O(1)$ queries

Reflect around $|\pi_{\beta_i}\rangle$ in $\tilde{O}(\sqrt{m})$ queries

One q-sample $|\pi\rangle$: $\tilde{O}(\sqrt{n}) \times \tilde{O}(\sqrt{m}) = \tilde{O}(\sqrt{mn})$

Multiple q-samples:

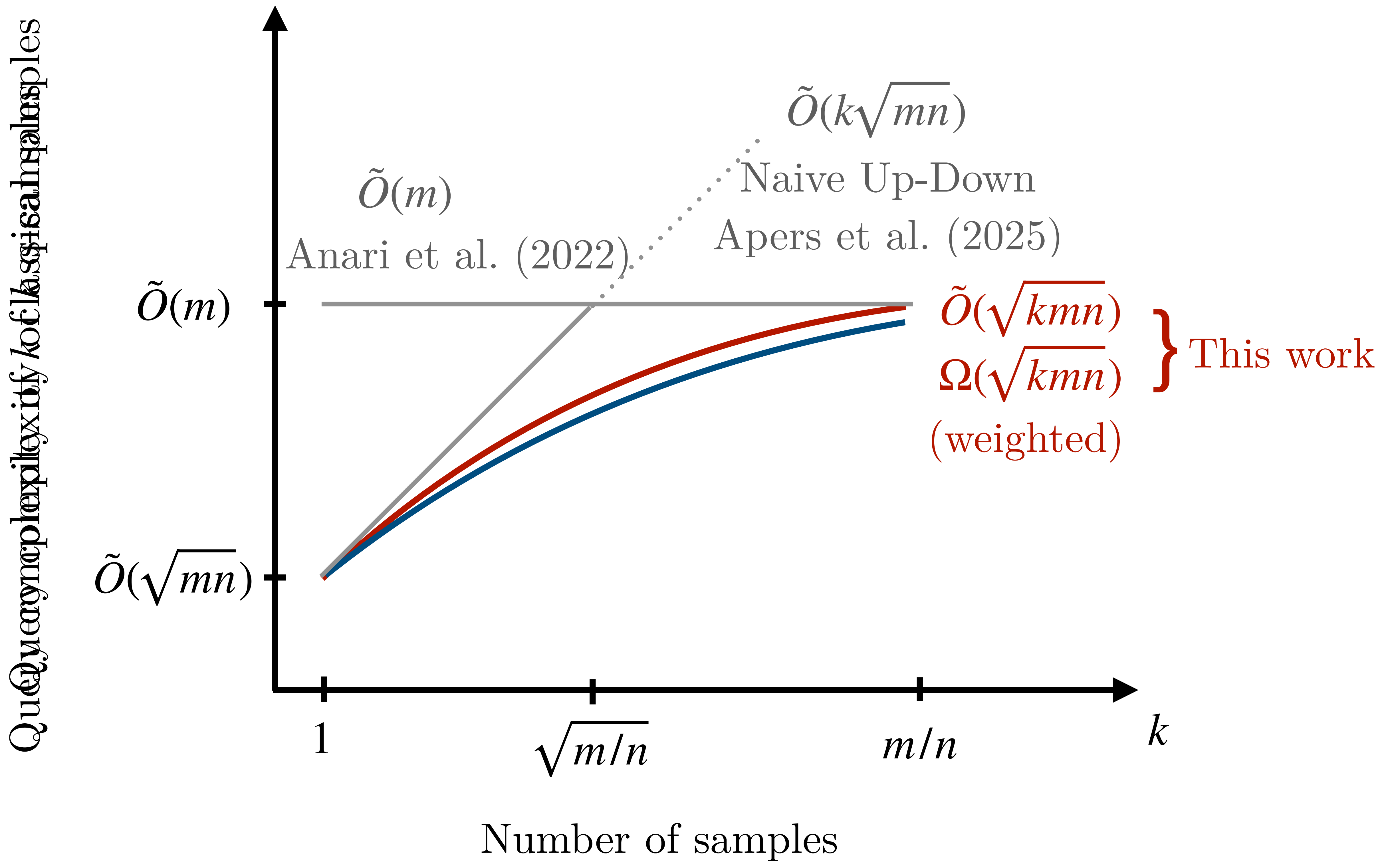
k q-samples: $\tilde{O}(k\sqrt{mn})$

Can we do better?

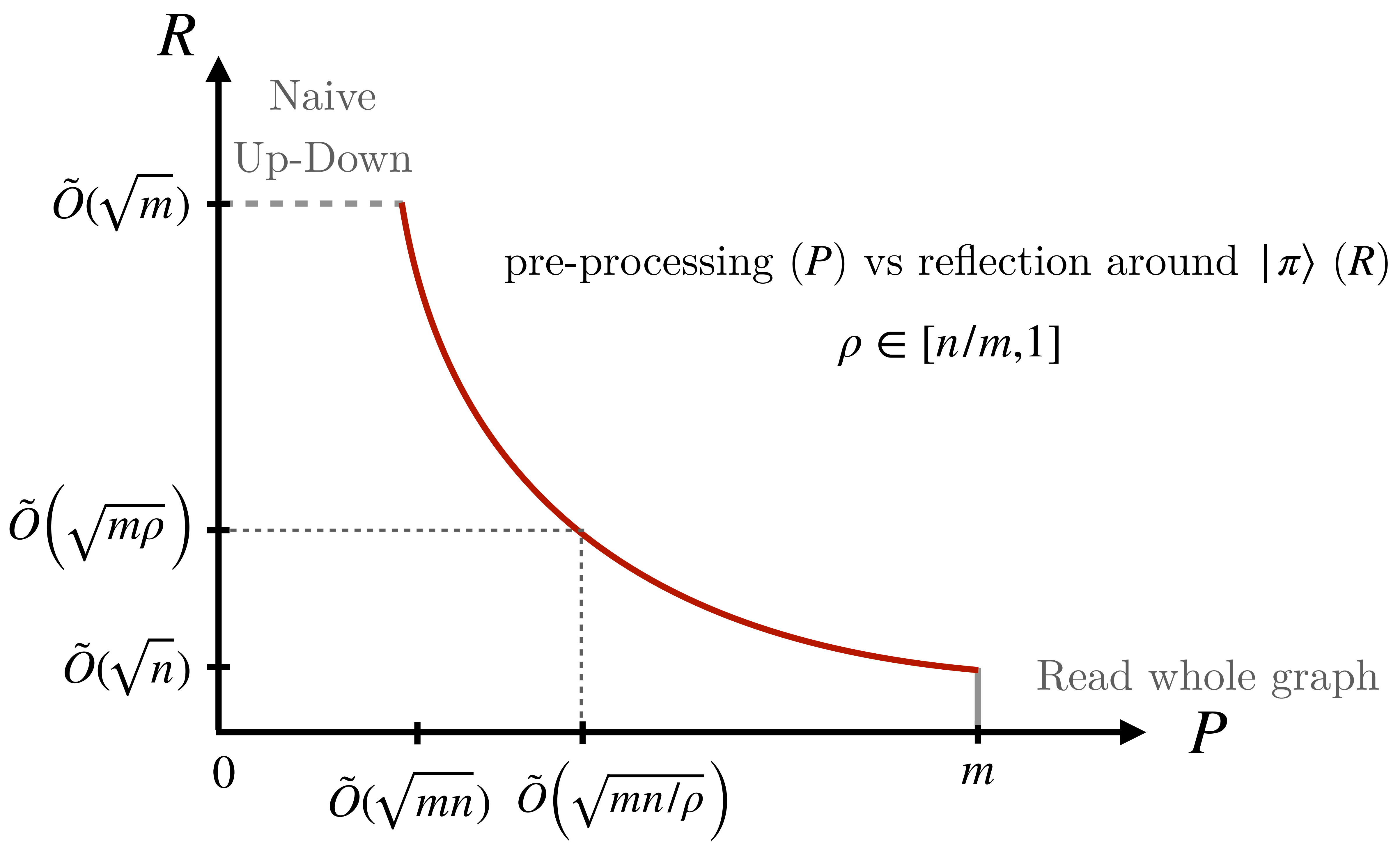
Pre-processing vs Sampling:

Can we beat $\tilde{O}\left(\sqrt{mn}\right)$ with pre-processing?

Our Results



$\tilde{O}(k\sqrt{mn})$
Naive Up-Down
Apers et al. (2025)
$\tilde{O}(m)$
Anari et al. (2022)
$\tilde{O}(m)$
$\tilde{O}(\sqrt{mn})$
$\tilde{O}(\sqrt{kmn})$
$\Omega(\sqrt{kmn})$
(weighted)
This work
1
$\sqrt{m/n}$
m/n
k
Number of samples



R
Naive
Up-Down
$\tilde{O}(\sqrt{m})$
pre-processing (P) vs reflection around $|\pi\rangle$ (R)
$\rho \in [n/m, 1]$
$\tilde{O}\left(\sqrt{m\rho}\right)$
$\tilde{O}(\sqrt{n})$
Read whole graph
0
$\tilde{O}(\sqrt{mn})$
$\tilde{O}\left(\sqrt{mn/\rho}\right)$
m
P

Our Result

Main theorem

1. For any $\rho \in [n/m, 1]$, **preprocessing** (P)**:** $\tilde{O}\left(\sqrt{mn/\rho}\right)$,

   reflect around $|\pi\rangle$ (R)**:** $\tilde{O}(\sqrt{m\rho})$,

   prepare one q-sample (Q)**:** $\tilde{O}(\sqrt{mn\rho})$

2. Weighted graphs: k classical samples require $\Omega(\sqrt{kmn})$ queries

Quantum walk search:

$U = R \cdot \sqrt{\gamma}$ $\qquad$ $S = P + Q$

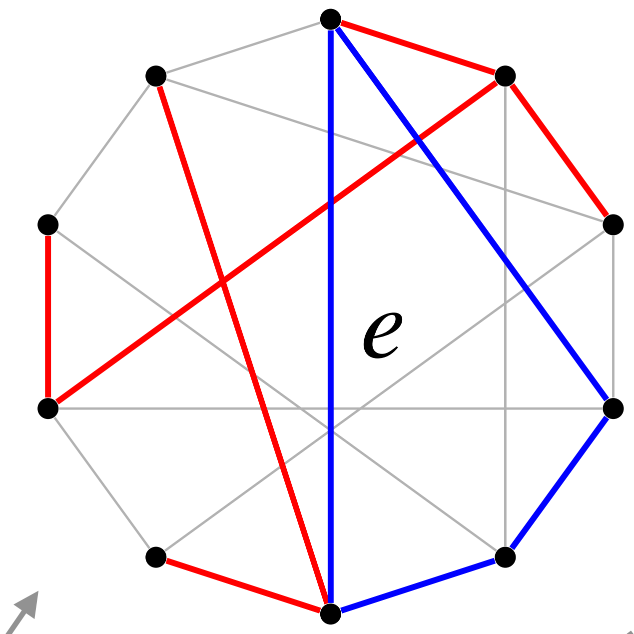
$$\tilde{O}\left(S + \frac{1}{\sqrt{\alpha}}\left(\frac{1}{\sqrt{\gamma}}U + C\right)\right)$$

Counting spanning trees:

Classically: $\Omega(m)$

This work: $\tilde{O}\left(\sqrt{m} \cdot n^{5/8}\right)$

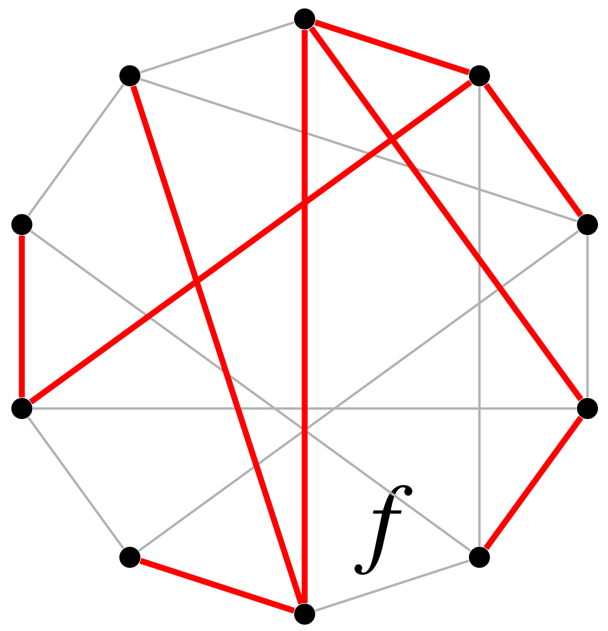
Proof sketch: Marginal-biased Up-Down walk P_{ma}



$T+f$

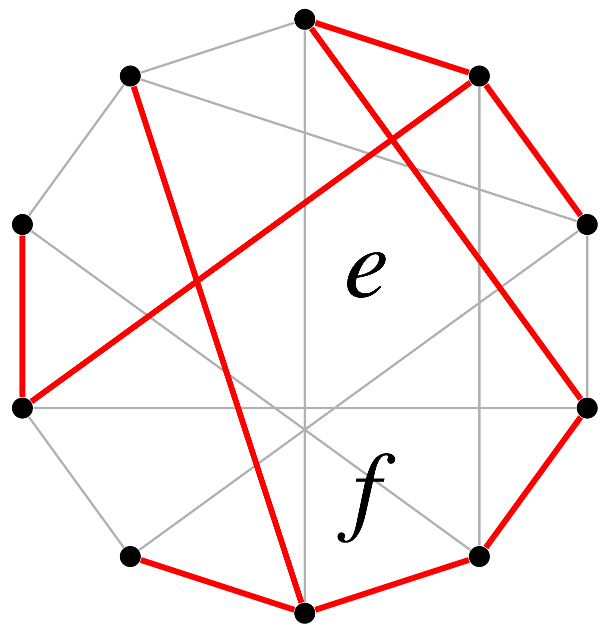
Add $f \in E$. with prob. $\propto p_G(e)$

Remove $e \in \text{Cycle}(T+f)$ with prob. $\propto \dfrac{\pi(T+f-e)}{\prod_{h \in T} p_G(h)}$



T

$\xrightarrow{\text{One step}}$



$T+f-e$

$P_{\text{ma}}(T,T') = \Pr[\text{One Up-Down step from } T \text{ to } T']$

Same state space throughout

$$p_G(e) = \Pr_{T\sim\pi}[e \in T]$$

Use $p_G(e)$ to bias transitions $\longrightarrow$ spectral gap $\tilde{\Omega}(1/n)$

$$P_{\text{ma}} \xrightarrow[\text{Phase gap of } W\text{: } \tilde{\Omega}\left(1/\sqrt{n}\right)]{\text{Szegedy quantisation}} W$$

Cost of one q-sample $|\pi\rangle$:

$$\underbrace{\tilde{O}(\sqrt{n})}_{\text{length of sequence}} \times \underbrace{\tilde{O}(\sqrt{n})}_{\text{uses of } W} \times \text{Cost}(W)$$

Proof sketch: Quantum walk operator

Main challenge: How to efficiently implement W?

Suffices to implement U:

$$|T\rangle|0\rangle \longmapsto |T\rangle \sum_{T' \to T} \sqrt{P_{\mathrm{ma}}(T,T')}\,|T'\rangle$$

Useful primitive for implementing U:

$$|0\rangle \longmapsto \frac{1}{\sqrt{n-1}} \sum_{e \in G} \sqrt{p_{G_\beta}(e)}\,|e\rangle$$

Marginal state

$$\sum_{e \in G} p_{G_\beta}(e) = n-1$$

Given coherent access to $p_G(e)$:

$$|0\rangle \xmapsto{\tilde{O}(1)} \frac{1}{\sqrt{m}} \sum_{e \in G} |e\rangle \xrightarrow[\text{Grover/ Rejection sampling}]{\tilde{O}\left(\sqrt{\frac{m}{n}}\right)} \frac{1}{\sqrt{n-1}} \sum_{e \in G} \sqrt{p_{G_\beta}(e)}\,|e\rangle$$

How do we obtain $p_{G_{\beta_i}}(e)$ for all β_i?

Can we beat $\tilde{O}\left(\sqrt{\frac{m}{n}}\right)$?

Proof sketch: Effective resistances and marginals

Laplacian:

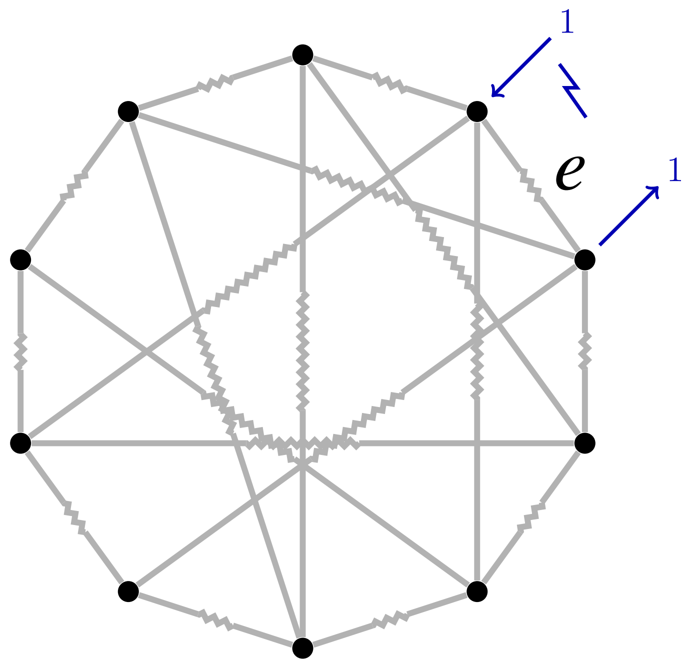
$$L_G = \sum_{e \in E} \underbrace{w(e)\, b_e b_e^\top}_{L_e}$$

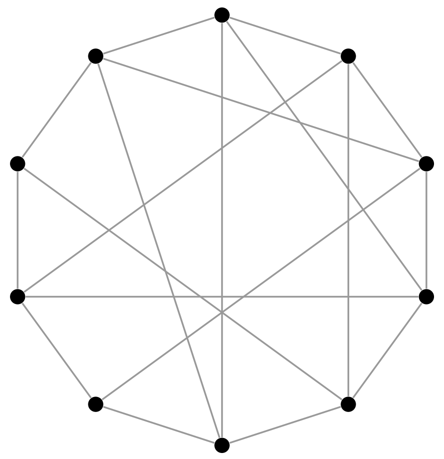
where $b_e = \delta_u - \delta_v$

for $e = \{u, v\}$

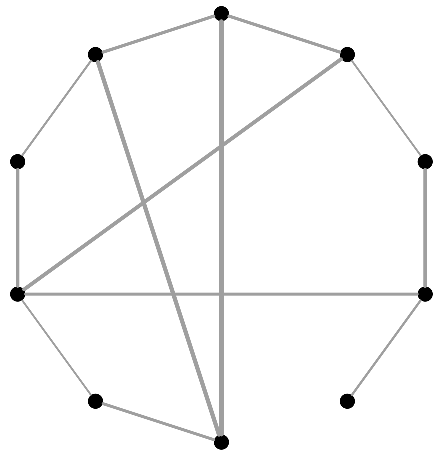
Marginals of π: $\quad p_G(e) = w(e) \cdot R_{\text{eff}}^G(e)$

Effective-resistance: $\quad R_{\text{eff}}^G(u, v) = b_{uv}^\top L_G^+ b_{uv}$





$$G \xrightarrow[L_G \approx L_H]{\text{Sparsification}} H$$



$\tilde{O}(n)$ edges

Quantum speedup by Apers and de Wolf (2020):

$\tilde{O}\left(\sqrt{mn}\right)$ queries $\Longrightarrow$ explicit sparsifier H

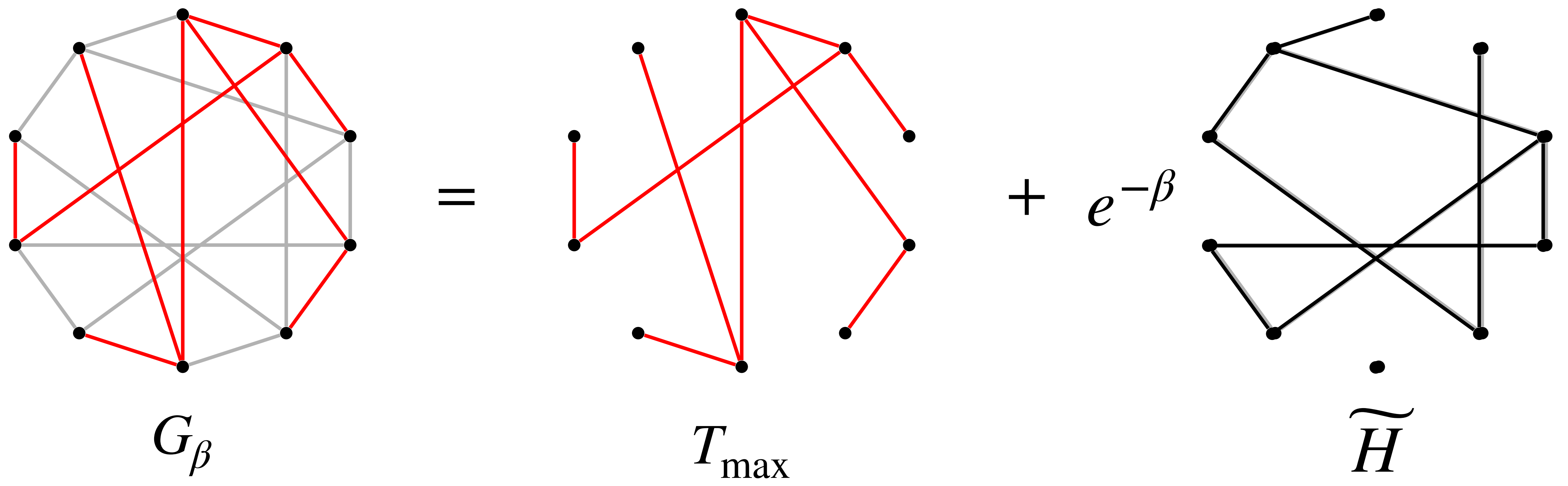
then $R_{\text{eff}}^G(\,\cdot\,,\cdot\,)$ needs **no further queries** to $\mathcal{O}_G$

$\tilde{O}\left(\sqrt{n}\right)$ graphs $\Longrightarrow$ one sparsifier per graph is too costly!

Proof sketch: Reusable sparsifier

Goal: Avoid sparsifying every G_β separately

$$L_{G_\beta} \approx L_{T_{\max}} + e^{-\beta} L_{\widetilde{H}}$$



A-edges fixed;
all others uniformly
rescaled!

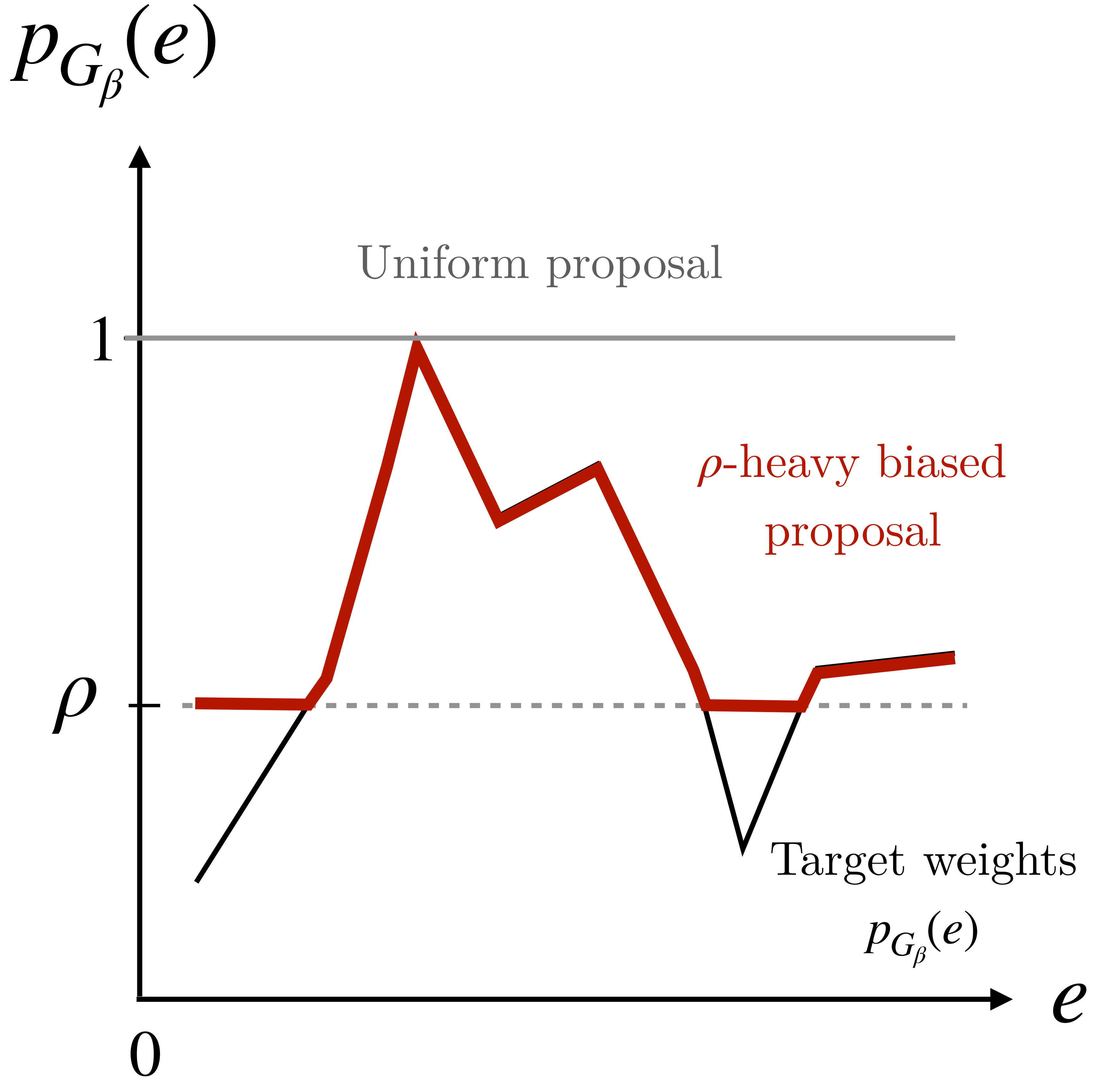
Lemma (Sum of Laplacians). For weighted graphs G_1, G_2, we have $L_{G_1+\alpha G_2} = L_{G_1} + \alpha L_{G_2}$

$$G \backslash T_{\max} \xrightarrow[\tilde{O}\left(\sqrt{mn}\right)]{\text{Sparsify once!}} \widetilde{H}$$

But we can reuse $\widetilde{H}$ for each G_β!

$\tilde{O}\left(\sqrt{mn}\right)$ pre-processing once $\Longrightarrow R_{\text{eff}}^{G_\beta}(\cdot,\cdot)$ needs **no further queries** to $\mathcal{O}_G$ for any β

Proof sketch: Improving the proposal



ρ-Heavy edges

$$\rho\text{−HVY}\left(G_\beta\right) := \{e \in G : p_{G_\beta}(e) \geq \rho\}$$

Uniform proposal

$$\frac{1}{\sqrt{m}} \sum_{e \in G} |e\rangle \xmapsto{\tilde{O}\left(\sqrt{m/n}\right)} \frac{1}{\sqrt{n-1}} \sum_{e \in G} \sqrt{p_G(e)}\, |e\rangle$$

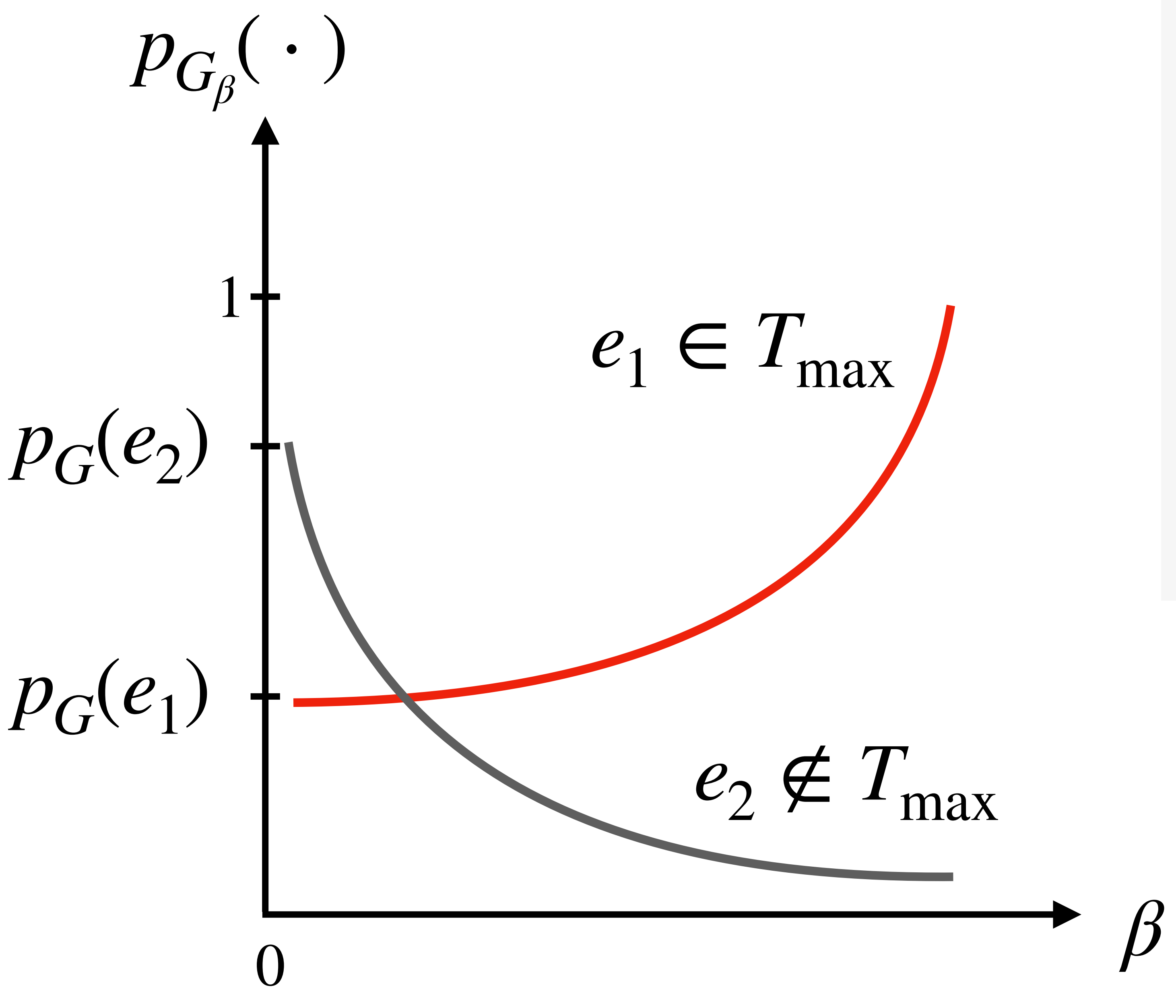
ρ-Heavy biased proposal

$$\frac{1}{\sqrt{Z'}} \left(\sum_{e \in \rho\text{-HVY}(G_\beta)} \sqrt{p_{G_\beta}(e)}\, |e\rangle + \sum_{e \notin \rho\text{-HVY}(G_\beta)} \sqrt{\rho}\, |e\rangle \right) \xmapsto{\tilde{O}\left(\sqrt{m\rho/n}\right)} \frac{1}{\sqrt{n-1}} \sum_{e \in G} \sqrt{p_G(e)}\, |e\rangle$$

Similar ideas in zero sum games, Gibbs sampling..

How do we find ρ−HVY$\left(G_{\beta_i}\right)$ for all β_i?

Proof sketch: Improving the proposal



Lemma (Monotonicity of marginals).

For $e \in A$, marginal of e **increases** with β

For $e \notin A$, marginal of e **decreases** with β

Moreover, for $\beta_1 \leq \beta_2$, $\quad \rho\text{-HVY}(G_{\beta_2}) \subseteq \rho\text{-HVY}(G_{\beta_1}) \cup A$

$$\rho\text{–HVY}\left(G_\beta\right) \subseteq \rho\text{–HVY}(G) \cup A, \text{ for all } \beta$$

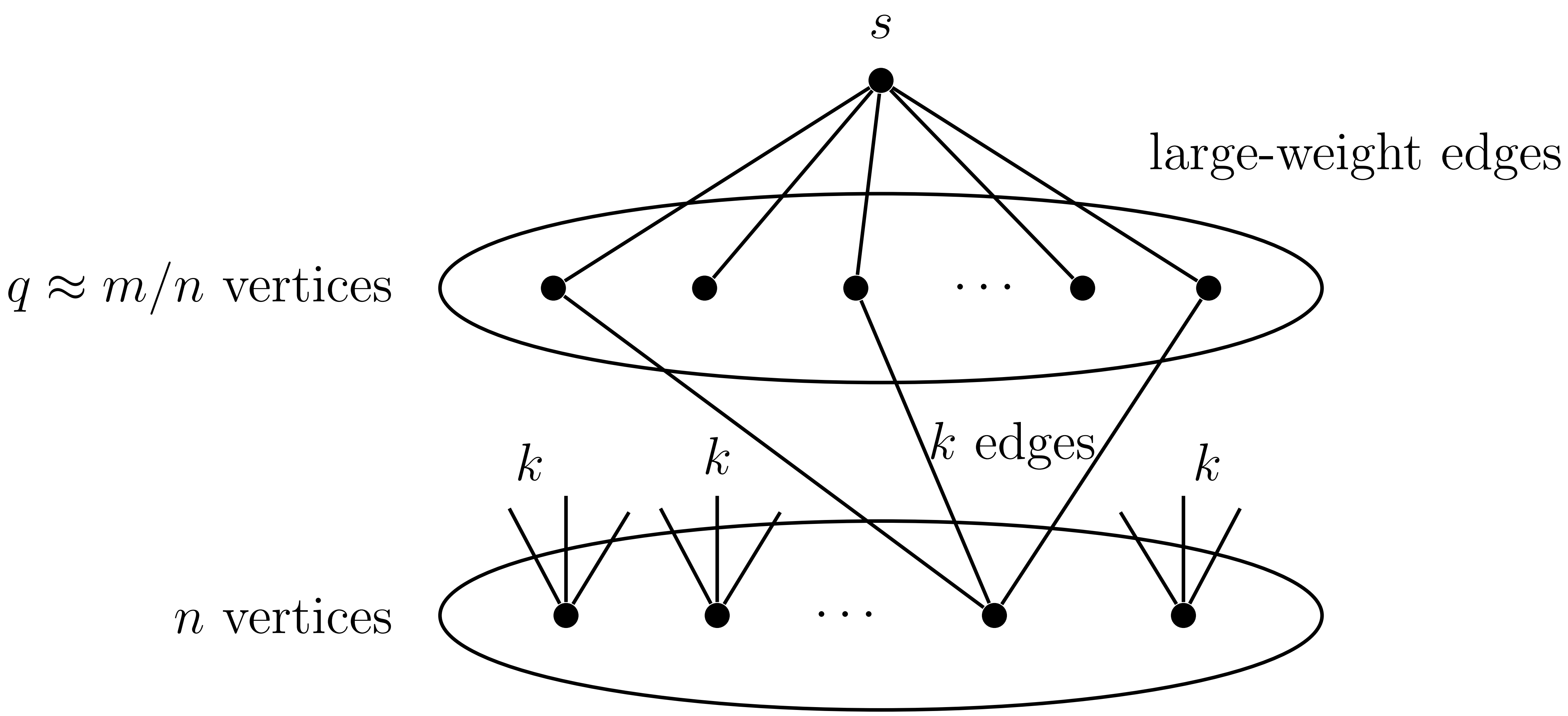
ρ-Heavy edges can be identified in one shot

Find ρ-Heavy edges: $O(\sqrt{mn/\rho})$ queries (repeated Grover search)

For any $\rho \in [n/m, 1]$, $\quad$ Pre-processing (P): $\tilde{O}\left(\sqrt{mn/\rho}\right)$,

Cost(W): $O(\sqrt{m\rho}) \implies$ reflection around $|\pi\rangle$ (R): $\tilde{O}(\sqrt{mn\rho})$

Proof sketch: Lower bound for k samples



Construction similar to the lower bound for one classical sample in Apers et al. (2025)

Encode many search problems:

A hard weighted graph encodes n independent search problems of size $q \approx m/n$

Samples reveal solutions:

k spanning tree samples $\Longrightarrow \Theta(k)$ distinct solutions for each search problem

Invoke multiple-search lower bound:

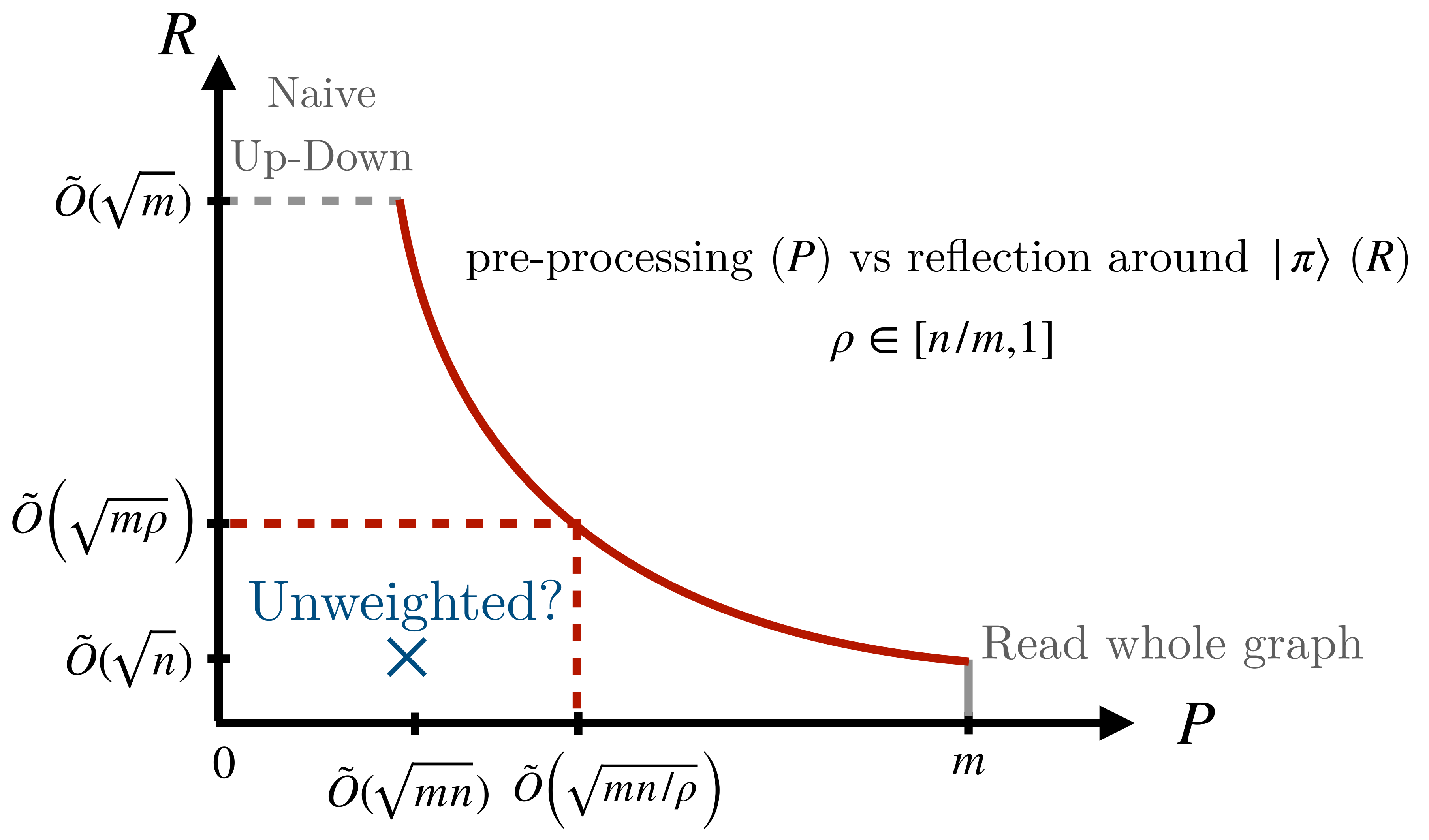
$$\Omega(n\sqrt{kq}) = \Omega(\sqrt{kmn})$$

Open questions

Time complexity:

The time complexity is $\tilde{O}(n^2)$ naively, can we improve?

A history-independent quantum version of the link-cut tree?



Unweighted graphs:

Improved sampling cost for unweighted graphs?

Marginal state for unweighted graphs with fewer queries?

Beyond spanning trees:

Spanning trees $\subset$ DPPs $\subset$ Strongly Rayleigh $\subset$ strongly log-concave

What quantum speedups are possible for DPPs, Bases of Matroids?