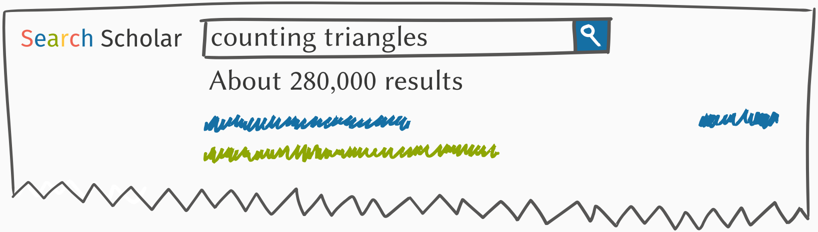
Sampling Arbitrary Subgraphs Exactly Uniformly in Sublinear Time

Hendrik Fichtenberger, Mingze Gao and Pan Peng

ICALP 2020

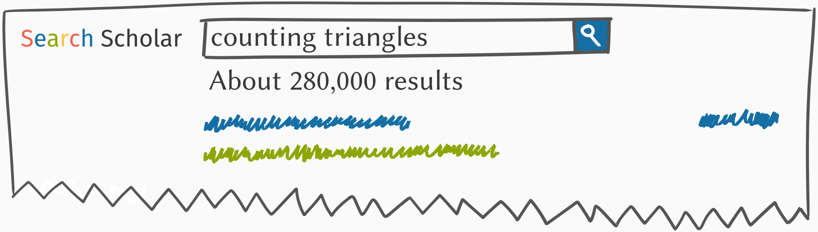
Subgraph Problems

subgraph problems are basic but popular:



Subgraph Problems

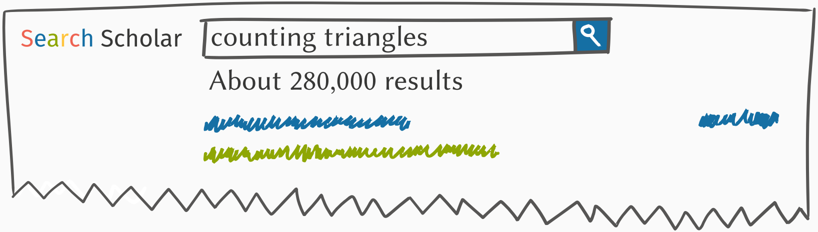
subgraph problems are basic but popular:



in this talk: sampling subgraphs in sublinear time

Subgraph Problems

subgraph problems are basic but popular:



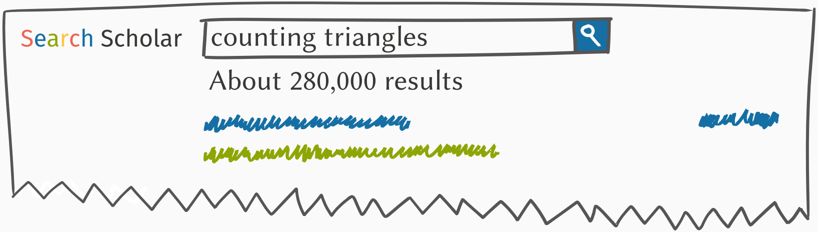
in this talk: sampling subgraphs in sublinear time

↓

fixed subgraph
uniformly from
an input graph

Subgraph Problems

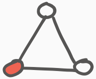
subgraph problems are basic but popular:



in this talk: sampling subgraphs in sublinear time

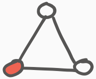
fixed subgraph uniformly from an input graph

need to define model of access to input graph

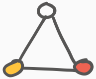


query for uniformly random vertex

Graph Models

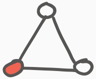


query for uniformly random vertex

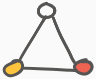


given a vertex,
query for its i-th neighbor

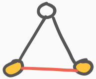
Graph Models



query for uniformly random vertex

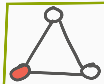


given a vertex,
query for its *i*-th neighbor

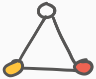


given two vertices,
query if they are adjacent

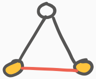
Graph Models



query for uniformly random vertex



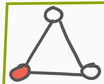
given a vertex,
query for its i-th neighbor



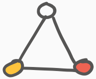
given two vertices,
query if they are adjacent

general model

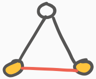
Graph Models



query for uniformly random vertex

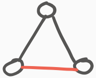


given a vertex,
query for its i-th neighbor



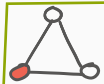
given two vertices,
query if they are adjacent

general model

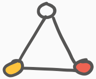


query for uniformly random edge

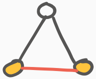
Graph Models



query for uniformly random vertex

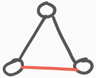


given a vertex,
query for its i-th neighbor



given two vertices,
query if they are adjacent

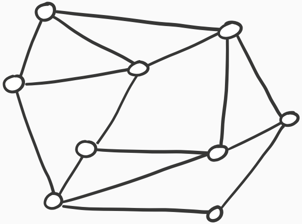
general model



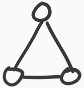
query for uniformly random edge

augmented general model

Simple Algorithm for the Augmented General Model



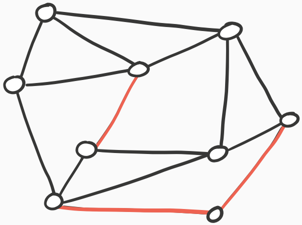
G

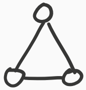


H

repeat until success:

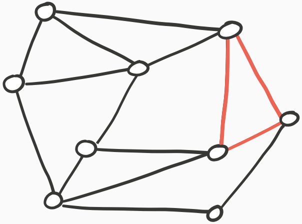
Simple Algorithm for the Augmented General Model



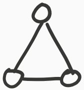


repeat until success:

1. sample $|E(H)|$ edges uniformly at random
2. check wether they form a copy of H



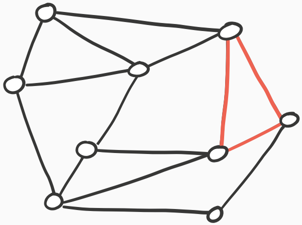
G



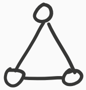
H

repeat until success:

1. sample $|E(H)|$ edges uniformly at random
2. check wether they form a copy of H



G



H

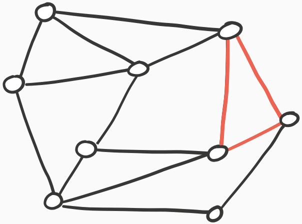
repeat until success:

1. sample $|E(H)|$ edges uniformly at random
2. check wether they form a copy of H

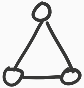
complexity:

- probability to sample a fixed copy of H: $\Theta\left(\frac{1}{m^{|E(H)|}}\right)$
- expected running time: $\Theta\left(\frac{m^{|E(H)|}}{\#H}\right)$

Simple Algorithm for the Augmented General Model



G



H

repeat until success:

1. sample $|E(H)|$ edges uniformly at random
2. check wether they form a copy of H

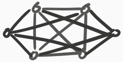
complexity:

- probability to sample a fixed copy of H: $\Theta\left(\frac{1}{m^{|E(H)|}}\right)$
- expected running time: $\Theta\left(\frac{m^{|E(H)|}}{\#H}\right)$

can we do better?

Related Work[1]

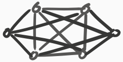
improvements over linear query bounds in $\tilde{O}(\ldots)$, all essentially tight for cliques

subgraph H	approximate counting	sampling *approx.* uniformly	sampling *exactly* uniformly
			
			
			

[1] Goldreich, Ron, RS&A’08; Eden, Rosenbaum, SOSA’18; Eden, Ron, Seshadhri, STOC’18; Assadi, Kapralov, Khanna, ITCS’18

Related Work[1]

improvements over linear query bounds in Õ(...), all essentially tight for cliques

subgraph H	approximate counting	sampling *approx.* uniformly	sampling *exactly* uniformly
	$\frac{n}{\sqrt{m}}$ [GR08]	$\frac{n}{\sqrt{m}}$ [ER18]	
	$\frac{n}{(\#H)^{1/\lvert H\rvert}} + \frac{m^{\rho(H)}}{\#H}$ [ERS18]		
any			

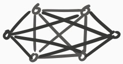
general model

$\rho(H)$ is the frac. edge cover size of H, $|H|/2 \leq \rho(H) \leq |H|$

[1] Goldreich, Ron, RS&A'08; Eden, Rosenbaum, SOSA'18; Eden, Ron, Seshadhri, STOC'18; Assadi, Kapralov, Khanna, ITCS'18

Related Work[1]

improvements over linear query bounds in Õ(...), all essentially tight for cliques

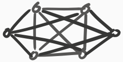
subgraph H	approximate counting	sampling *approx.* uniformly	sampling *exactly* uniformly
	$\frac{n}{\sqrt{m}}$ [GR08]	$\frac{n}{\sqrt{m}}$ [ER18]	
	$\frac{n}{(\#H)^{1/\vert H\vert}} + \frac{m^{\rho(H)}}{\#H}$ [ERS18]		
any			

general model

$\rho(H)$ is the frac. edge cover size of H, $|H|/2 \leq \rho(H) \leq |H|$

[1] Goldreich, Ron, RS&A'08; Eden, Rosenbaum, SOSA'18; Eden, Ron, Seshadhri, STOC'18; Assadi, Kapralov, Khanna, ITCS'18

improvements over linear query bounds in $\tilde{O}(\ldots)$, all essentially tight for cliques

subgraph H	approximate counting	sampling *approx.* uniformly	sampling *exactly* uniformly
	$\frac{n}{\sqrt{m}}$ [GR08]	$\frac{n}{\sqrt{m}}$ [ER18]	trivial
	$\frac{n}{(\#H)^{1/\lvert H\rvert}} + \frac{m^{\rho(H)}}{\#H}$ [ERS18]	$\frac{m^{\rho(H)}}{\#H}$ [ERS18]	$\frac{m^{\rho(H)}}{\#H}$ [ERS18]
any	$\frac{m^{\rho(H)}}{\#H}$ [AKK18]	$\frac{m^{\rho(H)}}{\#H}$ [this work]	$\frac{m^{\rho(H)}}{\#H}$ [this work]

general model / augmented general model

$\rho(H)$ is the frac. edge cover size of H, $|H|/2 \le \rho(H) \le |H|$

[1] Goldreich, Ron, RS&A'08; Eden, Rosenbaum, SOSA'18; Eden, Ron, Seshadhri, STOC'18; Assadi, Kapralov, Khanna, ITCS'18

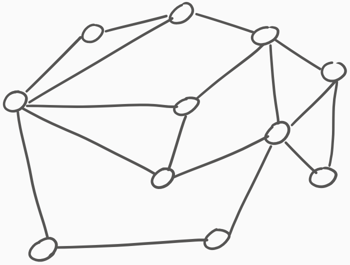
Our Results

Main Theorem

For any subgraph H, sampling *exactly uniformly* from all copies of H in an input graph G has expected query and time complexity $\mathcal{O}\left(\frac{m^{\rho(H)}}{\#H}\right)$ in the augmented general model.

This is essentially tight for cliques, even when we require only *almost* uniform sampling.

Fractional Edge Covers

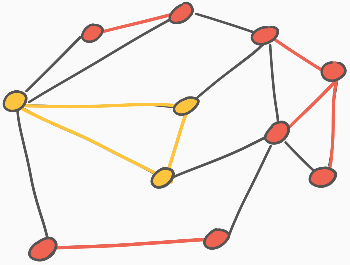


Theorem [AKK18][2]

For every graph H, there is a minimum fractional edge cover by vertex-disjoint odd cycles and stars.

[2] Assadi, Kapralov, Khann, ITCS'18

Fractional Edge Covers



$\psi(H) = 6.5$

Theorem [AKK18][2]

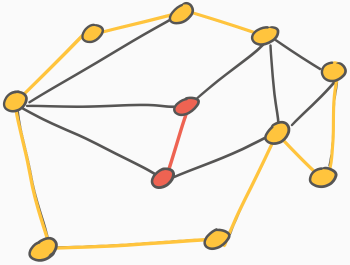
For every graph H, there is a minimum fractional edge cover by vertex-disjoint odd cycles and stars.

The value $\psi_H(C)$ of a cover C is

$$\psi_H(C) = \sum_{\substack{k \in \{3,5,\ldots\} \\ C_k \in C}} \frac{k}{2} + \sum_{\substack{k \in \mathbb{N} \\ S_k \in C}} k.$$

[2] Assadi, Kapralov, Khann, ITCS'18

Fractional Edge Covers



$\psi(H) = 5.5$

Theorem [AKK18][2]

For every graph H, there is a minimum fractional edge cover by vertex-disjoint odd cycles and stars.

The value $\psi_H(C)$ of a cover C is

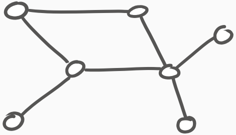
$$\psi_H(C) = \sum_{\substack{k \in \{3,5,\dots\} \\ C_k \in C}} \frac{k}{2} + \sum_{\substack{k \in \mathbb{N} \\ S_k \in C}} k.$$

We define $\rho(H) = \min_C \psi_H(C)$.

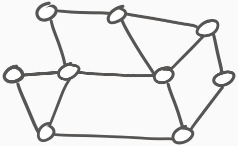
[2]Assadi, Kapralov, Khann, ITCS'18

Sampling in the Augmented General Model

for subgraph H and input G:



H



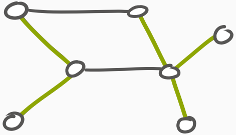
G

$\rho(H) = 5$

Sampling in the Augmented General Model

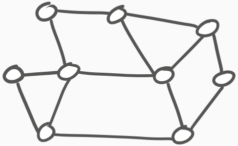
for subgraph H and input G:

1. compute edge cover of H



H

$\rho(H) = 5$

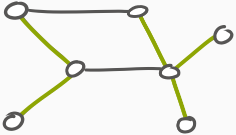


G

Sampling in the Augmented General Model

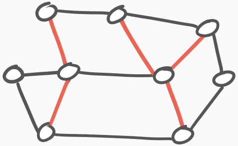
for subgraph H and input G:

1. compute edge cover of H
2. sample $\rho(H)$ edges from G



H

$\rho(H) = 5$

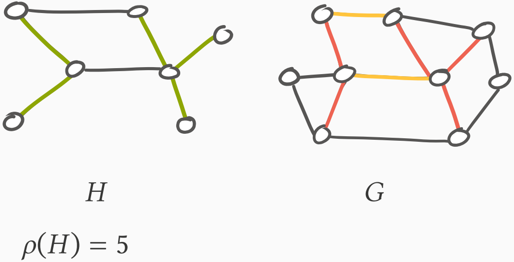


G

Sampling in the Augmented General Model

for subgraph H and input G:

1. compute edge cover of H
2. sample $\rho(H)$ edges from G
3. use pair queries to check for copy of H



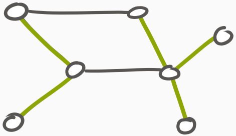
Sampling in the Augmented General Model

for subgraph H and input G:

1. compute edge cover of H
2. sample $\rho(H)$ edges from G
3. use pair queries to check for copy of H

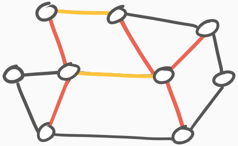
probability to find fixed H: $\Theta\left(\frac{1}{m^{\rho(H)}}\right)$

expected time to find it: $\Theta(m^{\rho(H)})$



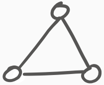
H

$\rho(H) = 5$



G

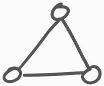
How About Odd Cycles?



H

for odd cycles C_k: $\rho(C_k) = \frac{k}{2}$

How About Odd Cycles?

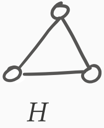


H

for odd cycles C_k: $\rho(C_k) = \frac{k}{2}$

how to sample $\frac{k}{2}$ edges?

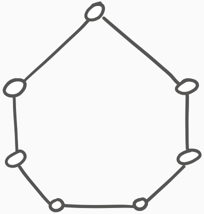
How About Odd Cycles?



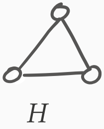
for odd cycles C_k: $\rho(C_k) = \frac{k}{2}$

how to sample $\frac{k}{2}$ edges?

for odd cycles C_k:

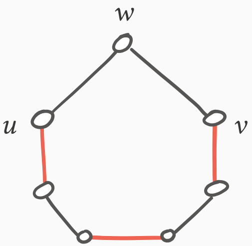


How About Odd Cycles?



for odd cycles C_k: $\rho(C_k) = \frac{k}{2}$

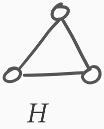
how to sample $\frac{k}{2}$ edges?



for odd cycles C_k:

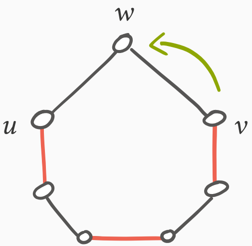
1. sample $\frac{k-1}{2}$ edges

How About Odd Cycles?



for odd cycles C_k: $\rho(C_k) = \frac{k}{2}$

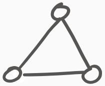
how to sample $\frac{k}{2}$ edges?



for odd cycles C_k:

1. sample $\frac{k-1}{2}$ edges
2. sample a common neighbor of u and v

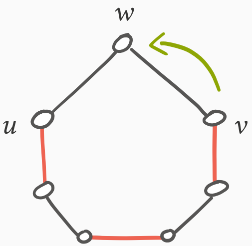
How About Odd Cycles?



H

for odd cycles C_k: $\rho(C_k) = \frac{k}{2}$

how to sample $\frac{k}{2}$ edges?



for odd cycles C_k:

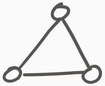
1. sample $\frac{k-1}{2}$ edges
2. sample a common neighbor of u and v

neighbor query probability: $\Pr[w] = \frac{1}{d_v}$

target probability: $\Pr[w] = \Theta\left(\frac{1}{\sqrt{m}}\right)$

for all $w \in \Gamma(u) \cap \Gamma(v)$

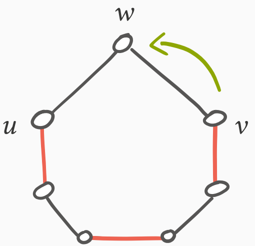
How About Odd Cycles?



H

for odd cycles C_k: $\rho(C_k) = \frac{k}{2}$

how to sample $\frac{k}{2}$ edges?



for odd cycles C_k:

1. sample $\frac{k-1}{2}$ edges
2. sample a common neighbor of u and v

neighbor query probability: $\Pr[w] = \frac{1}{d_v}$

target probability: $\Pr[w] = \Theta\left(\frac{1}{\sqrt{m}}\right)$

for all $w \in \Gamma(u) \cap \Gamma(v)$

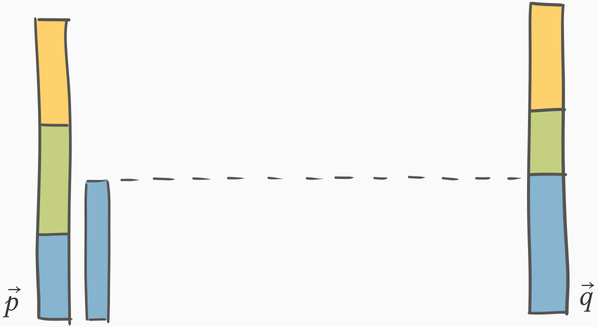
total probability to sample a fixed H: $\Theta\left(\frac{1}{m^{(k-1)/2}} \cdot \frac{1}{\sqrt{m}}\right) = \Theta\left(\frac{1}{m^{\rho(C_k)}}\right)$

Rejection Sampling

Problem

Given samples from distribution $\vec{p}$ on $[n]$, simulate sampling from $\vec{q}$.

Rejection Sampling

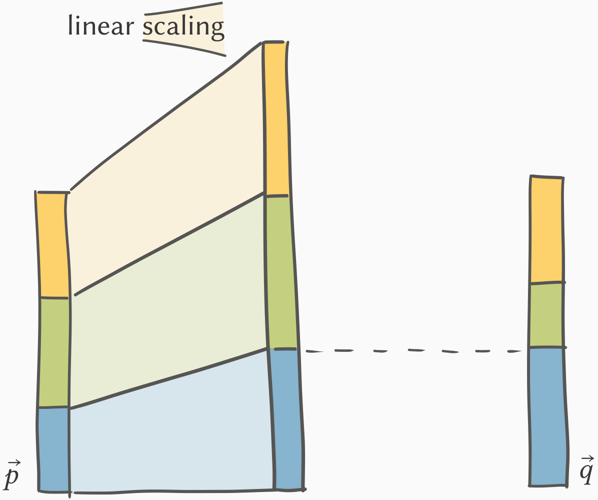


Problem

Given samples from distribution $\vec{p}$ on $[n]$, simulate sampling from $\vec{q}$.

1. scale $\vec{p}$ linearly by factor $s = \max_i \vec{q}(i)/\vec{p}(i)$

Rejection Sampling

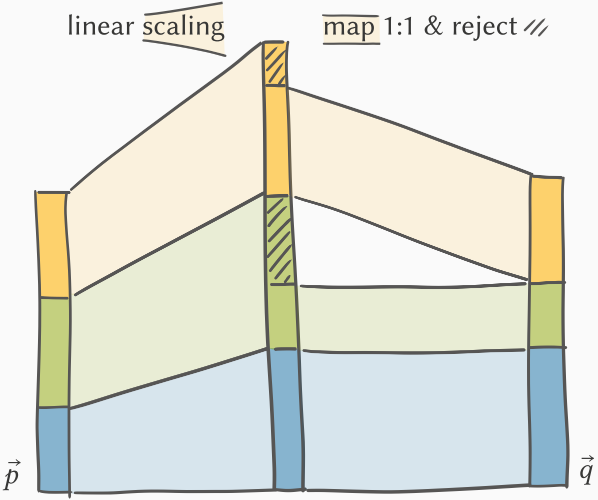


Problem

Given samples from distribution $\vec{p}$ on $[n]$, simulate sampling from $\vec{q}$.

1. scale $\vec{p}$ linearly by factor $s = \max_i \vec{q}(i)/\vec{p}(i)$

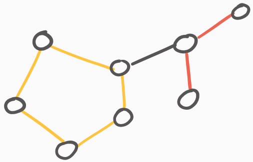
Rejection Sampling



Problem

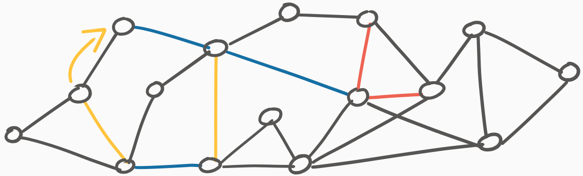
Given samples from distribution $\vec{p}$ on $[n]$, simulate sampling from $\vec{q}$.

1. scale $\vec{p}$ linearly by factor $s = \max_i \vec{q}(i)/\vec{p}(i)$
2. sample o from $\vec{p}$
3. sample x uniformly from $[0, 1]$
4. accept o if $x \leq \vec{q}(o)/(s\vec{p}(o))$, reject and repeat otherwise



H

1. decompose H into odd cycles C and stars S
2. repeat until success:
    a. sample edges from G as described
    b. check whether they form a copy of H using pair queries



G

sample H exactly uniformly in $\mathcal{O}\left(\frac{m^{\rho(H)}}{\#H}\right)$ expected time