## An FPRAS FOR TWO TERMINAL RELIABILITY IN DIRECTED ACYCLIC GRAPHS

Heng Guo (University of Edinburgh)
Based on joint work with Weiming Feng (UC Berkeley $\rightarrow$ ETH Zürich)

Institute of Software, Chinese Academy of Sciences, Dec 11, 2023

## The complexity of computing quantities

Complexity class \#P by Valiant (1979):
a counting analogue of NP
E.g. counting the number of solutions to CNF formulas

Other examples:
determinants / permanents of matrices;
evaluation of probabilities;
partition functions in statistical physics; counting discrete structures ...


## THE COMPLEXITY OF APPROXIMATE COUNTING

What about (multiplicatively) approximating \#P-complete problems?

- at most NP-hard (Stockmeyer 1983; Valiant and Vazirani, 1986);
- typically, polynomial approximation can be amplified into $\varepsilon$-approximation with only polynomial runtime overhead.

Thus, we strive to classify approximate counting problems as either NP-hard or FPRASable.

FPRASes do exist! Famous examples include:

- the number of solutions to DNF formulae
- the volume of convex bodies
(Karp and Luby, 1983); (Dyer, Frieze, and Kannan, 1991)
- the partition function of ferromagnetic Ising models
- the permanent of non-negative matrices
(Jerrum and Sinclair, 1993); (Jerrum, Sinclair, and Vigoda, 2004).
There are still many open problems in approximate counting!


## THE COMPLEXITY OF APPROXIMATE COUNTING

What about (multiplicatively) approximating \#P-complete problems?

- at most NP-hard (Stockmeyer 1983; Valiant and Vazirani, 1986);
- typically, polynomial approximation can be amplified into $\varepsilon$-approximation with only polynomial runtime overhead.

Thus, we strive to classify approximate counting problems as either NP-hard or FPRASable.

FPRASes do exist! Famous examples include:

- the number of solutions to DNF formulae (Karp and Luby, 1983);
- the partition function of ferromagnetic Ising models (Jerrum and Sinclair, 1993);
- the volume of convex bodies (Dyer, Frieze, and Kannan, 1991);
- the permanent of non-negative matrices (Jerrum, Sinclair, and Vigoda, 2004).

There are still many open problems in approximate counting!

Network reliability

## Network reliability

Given a directed or undirected graph (a.k.a. network) $G=(V, E)$, define a random subgraph $G(p)$ by removing each edge independently with probability $p$.

Two-terminal reliability: given $s, t \in V$,

$$
\operatorname{Pr}[s \xrightarrow{\mathrm{G}(\mathrm{p})} \mathrm{t}]
$$

Directed and undirected Two-terminal reliability (and a few other variants) are featured in the original list of 13 \# P-complete problems by Valiant (1979).

One may ask the probability of other properties of $G(p)$, such as whether $G(p)$ is connected (Allterminal reliability).

## NeTwORK RELIABILITY

Two-terminal reliability: $\operatorname{Pr}[s \xrightarrow{\mathrm{G}(\mathrm{p})} \mathrm{t}]$
In other words, we want to compute

For example:

$$
\begin{aligned}
& =(1-p)^{4}+4 p(1-p)^{3}+2 p^{2}(1-p)^{2}
\end{aligned}
$$

## COMPUTATIONAL COMPLEXITY OF RELIABILITY

Exact evaluation of almost all variants of reliability is \#P-complete, shown by the pioneer work of Valiant (1979), Jerrum (1981), Provan and Ball (1983), etc.

Much less is known about their approximation complexity.

A famous breakthrough is by Karger (1999), who gave an FPRAS for All-terminal unreliability. (However, good approximation of unreliability is not necessarily a good approximation for reliability when reliability is exponentially small.)

All-terminal reliability is shown to have an FPRAS by G. and Jerrum (2019), resolving positively conjectures by Provan and Ball (1983), Welsh (1993), Karger (1999), etc.

## Main Results

We gave an FPRAS for Two-terminal reliability in directed acyclic graphs (DAGs).

## Theorem (Feng and G., 2023)

Let G be a DAG and $\mathbf{q}$ denote failure probabilities. There is a randomized algorithm that takes $(\mathrm{G}, \mathrm{q}, \mathrm{s}, \mathrm{t}, \varepsilon)$ as inputs and outputs a $(1 \pm \varepsilon)$-approximation to $\mathrm{s}-\mathrm{t}$ RELIABILITY with probability at least $3 / 4$ in time $\widetilde{\mathrm{O}}\left(\mathrm{n}^{6} \mathrm{~m}^{4} \max \left\{\mathrm{~m}^{4}, \varepsilon^{-4}\right\}\right)$.

This answers positively a conjecture by Zenklusen and Laumanns (2011).
On the flip side, the corresponding unreliability problem is unlikely to have an FPRAS.

## Theorem (Feng and G., 2023)

There is no FPRAS to estimate $\mathrm{s}-\mathrm{t}$ unReLIAbility in DAGs unless there is an FPRAS for \#BIS.
Here \#BIS is the problem of counting independent sets in bipartite graphs. It is conjectured to have no FPRAS.

## Simultaneous work

Independently, Amarilli, van Bremen, and Meel (2023) reduce $s-t$ reLIABILITY in DAGs to counting the number of accepting paths of a given length for non-deterministic automata (\#NFA). The latter problem has an FPRAS by Arenas, Croquevielle, Jayaram, and Riveros (2021).

ACJR21's algorithm runs in time $O\left(\left(\frac{n \ell}{\varepsilon}\right)^{17}\right)$ for an $n$-state NFA and strings of length $\ell$. AvBM23 reduces a DAG with $n$ vertices and $m$ edges to a \#NFA instance counting length $m$ accepting strings for an NFA with $\mathrm{O}\left(\mathrm{m}^{2}\right)$ states. Thus their running time is like $\mathrm{O}\left(\frac{\mathrm{m}^{51}}{\varepsilon^{17}}\right)$.

| Terminal | Graphs | Type | Complexity | Best run-time |
| :---: | :---: | :---: | :---: | :---: |
| All | Undirected | Unrel | FPRAS (K99) | $\frac{m^{1+o(1)}}{\varepsilon^{2}}+\widetilde{\mathrm{O}}\left(\frac{n^{1.5}}{\varepsilon^{3}}\right)(\mathrm{CHLP23})$ |
| All | Undirected | Rel | FPRAS (GJ19) | $\widetilde{\mathrm{O}}\left(\frac{\mathrm{mn}}{\varepsilon^{2}}\right)(\mathrm{CGZZ23)}$ |
| $\mathrm{s}-\mathrm{t}$ | DAG | Rel | FPRAS (FG23, AvBM23) | $\widetilde{\mathrm{O}\left(n^{6} \mathrm{~m}^{4} \max \left\{\mathrm{~m}^{4}, \varepsilon^{-4}\right\}\right)(\mathrm{FG} 23)}$ |
| $\mathrm{s}-\mathrm{t}$ | DAG | Unrel | \#BIS-hard (FG23) | - |
| $\mathrm{S}-\mathrm{t}$ | DAG | Rel | NP-hard (upcoming) | - |

CHLP23: Cen, He, Li, and Panigrahi (2023)
CGZZ23: Chen, G., Zhang, and Zou (2023)

## Some natural attempts

(AND WHY THEY DO Not SUCCEED)

## Naive Monte Carlo

## A natural unbiased estimator $\widetilde{Z}$ of $Z_{\text {rel }}$ :

1. Draw $k$ independent subgraphs $\left(S_{i}\right)_{i \in[k]}$ of $G(p)$.
2. Let

$$
\tilde{Z}:=\frac{1}{k} \sum_{i \in[k]} \mathbb{1}_{s \rightarrow t}\left(S_{i}\right),
$$

where $\mathbb{1}_{s \rightarrow t}(S)$ is the indicator variable whether $s \xrightarrow{(\mathrm{~V}, \mathrm{~S})} \mathrm{t}$.

It is easy to see that $\mathbb{E} \widetilde{Z}=Z_{\text {rel }}$.
However, if $Z_{r e l}$ is exponentially small (e.g. $\left.Z_{r e l}\left(P_{n 2}, p\right)=(1-p)^{n-1}\right)$, then we will almost never see a connected $S_{i}$

In that case, the relative variance of $\mathbb{1}_{s \rightarrow t}(S)$ is exponentially large, and $k$ has to be exponentially large to yield a good approximation.

## Naive Monte Carlo

## A natural unbiased estimator $\widetilde{Z}$ of $Z_{\text {rel }}$ :

1. Draw $k$ independent subgraphs $\left(S_{i}\right)_{i \in[k]}$ of $G(p)$.
2. Let

$$
\tilde{Z}:=\frac{1}{k} \sum_{i \in[k]} \mathbb{1}_{s \rightarrow t}\left(S_{i}\right),
$$

where $\mathbb{1}_{s \rightarrow t}(S)$ is the indicator variable whether $s \xrightarrow{(V, S)} \mathrm{t}$.

It is easy to see that $\mathbb{E} \tilde{Z}=Z_{\text {rel }}$.
However, if $Z_{r e l}$ is exponentially small (e.g. $\left.Z_{r e l}\left(P_{n}, p\right)=(1-p)^{n-1}\right)$, then we will almost never see a connected $S_{i}$

In that case, the relative variance of $1_{S \rightarrow t}(S)$ is exponentially large, and $k$ has to be exponentially large to yield a good approximation.

## Naive Monte Carlo

A natural unbiased estimator $\widetilde{Z}$ of $Z_{\text {rel }}$ :

1. Draw $k$ independent subgraphs $\left(S_{i}\right)_{i \in[k]}$ of $G(p)$.
2. Let

$$
\widetilde{Z}:=\frac{1}{k} \sum_{i \in[k]} \mathbb{1}_{s \rightarrow t}\left(S_{i}\right),
$$

where $\mathbb{1}_{s \rightarrow t}(S)$ is the indicator variable whether $s \xrightarrow{(\mathrm{~V}, \mathrm{~S})} \mathrm{t}$.

It is easy to see that $\mathbb{E} \tilde{Z}=Z_{\text {rel }}$.
However, if $Z_{\text {rel }}$ is exponentially small (e.g. $\left.Z_{r e l}\left(P_{n}, p\right)=(1-p)^{n-1}\right)$, then we will almost never see a connected $S_{i}$.

In that case, the relative variance of $\mathbb{1}_{s \rightarrow t}(S)$ is exponentially large, and $k$ has to be exponentially large to yield a good approximation.

## Previous techniques

Karger's method for (undirected) All-terminal unreliability:
Naive Monte Carlo + random contraction
(Undirected) All-terminal reliability:
via the counting to sampling reduction (Jerrum, Valiant, and Vazirani, 1986) and then either

- Partial rejection sampling (G. and Jerrum, 2019); or
- Markov chain Monte Carlo (Anari, Liu, Oveis Gharan, and Vinzant, 2019),

All these techniques rely on some nice structure of the solution space, which Two-TERMINAL RELIABILITY (even in DAG) does not have.

## Previous techniques

Karger's method for (undirected) All-terminal unreliability:
Naive Monte Carlo + random contraction
(Undirected) All-terminal reliability:
via the counting to sampling reduction (Jerrum, Valiant, and Vazirani, 1986) and then either

- Partial rejection sampling (G. and Jerrum, 2019); or
- Markov chain Monte Carlo (Anari, Liu, Oveis Gharan, and Vinzant, 2019).

All these techniques rely on some nice structure of the solution space, which Two-TERMINAL RELI-
ABILITY (even in DAG) does not have.

## Previous techniques

Karger's method for (undirected) All-terminal unreliability:
Naive Monte Carlo + random contraction
(Undirected) All-terminal reliability:
via the counting to sampling reduction (Jerrum, Valiant, and Vazirani, 1986) and then either

- Partial rejection sampling (G. and Jerrum, 2019); or
- Markov chain Monte Carlo (Anari, Liu, Oveis Gharan, and Vinzant, 2019).

All these techniques rely on some nice structure of the solution space, which Two-terminal reliAbility (even in DAG) does not have.

## Markov chain Monte Carlo

Markov chains are the "off the shelf" approach to sampling from complicated distributions. Here we want to sample $R \sim G(p)$ conditioned on $s \xrightarrow{R} t$.

There is a natural Markov chain converging to the desired distribution:

1. Let $C_{0}=E$.
2. Given $C_{t}$, randomly pick an edge $e \in E$.

$$
\begin{aligned}
& \text { If } \neg\left(s \xrightarrow{C_{t} \backslash\{e\}} t\right) \text { then } C_{t+1}=C_{t} \text {. Otherwise, } \\
& \qquad C_{t+1}= \begin{cases}C_{t} \cup\{e\} & \text { with prob. } 1-p ; \\
C_{t} \backslash\{e\} & \text { with prob. } p .\end{cases}
\end{aligned}
$$

This chain is slow for $s$ and $t$ connected via two length $n / 2$ paths.

# Dynamic programming meets Monte Carlo 

## DYNAMIC PROGRAMMING

Our main inspiration is the \#NFA algorithm of Arenas, Croquevielle, Jayaram, and Riveros (2021), which in turn builds upon a quasi-polynomial time algorithm for generating words in context-free grammars (\#CFG) by Gore, Jerrum, Kannan, Sweedyk, and Mahaney (1997).

Given the topological ordering $s=v_{1}, v_{2}, \ldots, v_{n}=t$ of the input DAG, we estimate $R_{i}:=$ $\operatorname{Pr}\left[v_{i} \xrightarrow{\mathrm{G}(\mathrm{p})} \mathrm{t}\right]$ from $\mathrm{i}=\mathrm{n}$ to 1 .

Note that for $\nu_{i}$, we can safely ignore all vertices unreachable from $v_{i}$. Call the resulting graph $\mathrm{G}_{v_{i}}$, and we only need to consider $\mathrm{G}_{\nu_{i}}(\mathrm{p})$.

## The inductive step



Let $v_{i}$ 's neighbours be $u_{1}, \ldots, u_{d}$. Then,

$$
\operatorname{Pr}\left[v_{i} \xrightarrow{\mathrm{G}_{v_{i}}(p)} \mathrm{t}\right]=\operatorname{Pr}\left[\exists j \in[d],\left(\left(v_{i}, u_{j}\right) \in G_{v_{i}}(p)\right) \wedge\left(u_{j} \xrightarrow{\mathrm{G}_{u_{j}}(p)} \mathrm{t}\right)\right] .
$$

This reminds us to use the DNF counting technique of Karp and Luby (1983) and Karp, Luby, and Madras (1989).

## IN ACTION

Given DNF $\Phi=C_{1} \vee C_{2} \vee \cdots \vee C_{m}$, counting the number of solutions to $\Phi$ is equivalent to evaluating

$$
\operatorname{Pr}[\sigma \models \Phi]=\operatorname{Pr}\left[\exists i \in[m], \sigma \models C_{i}\right],
$$

where $\sigma$ is a uniformly at random assignment.
Let $\Omega_{i}$ be the set of solutions to $C_{i}$, and $p_{i}:=\operatorname{Pr}\left[\sigma \models C_{i}\right]=\frac{\left|\Omega_{i}\right|}{2^{n}}=2^{-\left|C_{i}\right|}$.
Karp-Luby goes as follows:

1. Draw $i$ with probability proportional to $p_{i}$;
2. draw uniformly $\sigma$ such that $\sigma \models C_{\mathfrak{i}}$ (namely uniformly from $\Omega_{\mathfrak{i}}$ );
3. Let $t_{\sigma}$ be the number of clauses $\sigma$ satisfies. Output $Z:=\frac{1}{t_{\sigma}}$.

Then $\mathbb{E}[Z]=\sum \sigma C_{i}: \sigma \left\lvert\,=C_{i} \frac{p_{i}}{\sum_{i \in[m]} p_{i}} \cdot \frac{1}{\left|\Omega_{i}\right|} \cdot \frac{1}{t_{\sigma}}\right.$.

$$
\mathbb{E}[Z]=\sum_{\sigma} \sum_{C_{i}: \sigma \models C_{i}} \frac{p_{i}}{\sum_{i \in[m]} p_{i}} \cdot \frac{1}{\left|\Omega_{i}\right|} \cdot \frac{1}{\mathrm{t}_{\sigma}}
$$



Also, $\mathbb{E}[Z]$
implying that $\frac{\operatorname{Var}(Z)}{\mathbb{E}[Z]^{2}} \leqslant m^{2}$

$$
\begin{aligned}
\mathbb{E}[Z] & =\sum_{\sigma} \sum_{C_{i}: \sigma \mid=C_{i}} \frac{p_{i}}{\sum_{i \in[m]} p_{i}} \cdot \frac{1}{\left|\Omega_{\mathfrak{i}}\right|} \cdot \frac{1}{\mathrm{t}_{\sigma}} \\
& =\frac{1}{\sum_{i \in[m]} p_{i}} \sum_{\sigma} \sum_{c_{i}: \sigma \mid=C_{i}} \frac{\left|\Omega_{i}\right|}{2^{n}} \cdot \frac{1}{\left|\Omega_{i}\right|} \cdot \frac{1}{\mathrm{t}_{\sigma}}
\end{aligned}
$$

$$
\begin{aligned}
\mathbb{E}[Z] & =\sum_{\sigma} \sum_{c_{i}: \sigma \models C_{i}} \frac{p_{i}}{\sum_{i \in[m]} p_{i}} \cdot \frac{1}{\left|\Omega_{i}\right|} \cdot \frac{1}{t_{\sigma}} \\
& =\frac{1}{\sum_{i \in[m]} p_{i}} \sum_{\sigma} \sum_{c_{i}: \sigma \mid=c_{i}} \frac{\left|\Omega_{i}\right|}{2^{n}} \cdot \frac{1}{\left|\Omega_{i}\right|} \cdot \frac{1}{t_{\sigma}} \\
& =\frac{1}{\sum_{i \in[m]} p_{i}} \cdot \frac{1}{2^{n}} \sum_{\sigma} \frac{1}{t_{\sigma}} \sum_{c_{i}: \sigma \mid=c_{i}} 1
\end{aligned}
$$

$$
\begin{aligned}
\mathbb{E}[Z] & =\sum_{\sigma} \sum_{c_{i}: \sigma \models C_{i}} \frac{p_{i}}{\sum_{i \in[m]} p_{i}} \cdot \frac{1}{\left|\Omega_{i}\right|} \cdot \frac{1}{t_{\sigma}} \\
& =\frac{1}{\sum_{i \in[m]} p_{i}} \sum_{\sigma} \sum_{c_{i}: \sigma \mid=c_{i}} \frac{\left|\Omega_{i}\right|}{2^{n}} \cdot \frac{1}{\left|\Omega_{i}\right|} \cdot \frac{1}{t_{\sigma}} \\
& =\frac{1}{\sum_{i \in[m]} p_{i}} \cdot \frac{1}{2^{n}} \sum_{\sigma} \frac{1}{t_{\sigma}} \sum_{C_{i}: \sigma \mid=c_{i}} 1 \\
& =\frac{1}{\sum_{i \in[m]} p_{i}} \cdot \frac{1}{2^{n}} \sum_{\sigma: \exists i \in[m], \sigma \mid=c_{i}} \frac{t_{\sigma}}{t_{\sigma}}
\end{aligned}
$$

$$
\begin{aligned}
\mathbb{E}[Z] & =\sum_{\sigma} \sum_{c_{i}: \sigma \mid=c_{i}} \frac{p_{i}}{\sum_{i \in[m]} p_{i}} \cdot \frac{1}{\left|\Omega_{i}\right|} \cdot \frac{1}{t_{\sigma}} \\
& =\frac{1}{\sum_{i \in[m]} p_{i}} \sum_{\sigma} \sum_{c_{i}: \sigma \mid=c_{i}} \frac{\left|\Omega_{i}\right|}{2^{n}} \cdot \frac{1}{\left|\Omega_{i}\right|} \cdot \frac{1}{t_{\sigma}} \\
& =\frac{1}{\sum_{i \in[m]} p_{i}} \cdot \frac{1}{2^{n}} \sum_{\sigma} \frac{1}{t_{\sigma}} \sum_{c_{i}: \sigma \models c_{i}} 1 \\
& =\frac{1}{\sum_{i \in[m]} p_{i}} \cdot \frac{1}{2^{n}} \sum_{\sigma: \exists i \in[m],}{ }_{\sigma \models=c_{i}} \frac{t_{\sigma}}{t_{\sigma}} \\
& =\frac{1}{\sum_{i \in[m]} p_{i}} \cdot \operatorname{Pr}\left[\exists i \in[m], \sigma \models C_{i}\right]
\end{aligned}
$$

$$
\begin{aligned}
\mathbb{E}[Z] & =\sum_{\sigma} \sum_{c_{i}: \sigma \mid=C_{i}} \frac{p_{i}}{\sum_{i \in[m]} p_{i}} \cdot \frac{1}{\left|\Omega_{i}\right|} \cdot \frac{1}{t_{\sigma}} \\
& =\frac{1}{\sum_{i \in[m]} p_{i}} \sum_{\sigma} \sum_{c_{i}: \sigma \models=c_{i}} \frac{\left|\Omega_{i}\right|}{2^{n}} \cdot \frac{1}{\left|\Omega_{i}\right|} \cdot \frac{1}{\mathrm{t}_{\sigma}} \\
& =\frac{1}{\sum_{i \in[m]} p_{i}} \cdot \frac{1}{2^{n}} \sum_{\sigma} \frac{1}{\mathrm{t}_{\sigma}} \sum_{c_{i}: \sigma \mid=c_{i}} 1 \\
& =\frac{1}{\sum_{i \in[m]} p_{i}} \cdot \frac{1}{2^{n}} \sum_{\sigma: \exists i \in[m], \sigma \models c_{i}} \frac{t_{\sigma}}{t_{\sigma}} \\
& =\frac{1}{\sum_{i \in[m]} p_{i}} \cdot \operatorname{Pr}\left[\exists i \in[m], \sigma \models C_{i}\right]
\end{aligned}
$$

Also, $\mathbb{E}[Z] \geqslant \frac{1}{m}$, implying that $\frac{\operatorname{Var}(Z)}{\mathbb{E}[Z]^{2}} \leqslant m^{2}$.

We want to estimate $\operatorname{Pr}\left[\exists j \in[d],\left(\left(v_{i}, u_{j}\right) \in G_{v_{i}}(p)\right) \wedge\left(u_{j} \xrightarrow{G_{u_{j}}(p)} t\right)\right]=\operatorname{Pr}\left[\exists j \in[d], \mathcal{E}_{j}\right]$.
Let $p_{j}:=\operatorname{Pr}\left[\mathcal{E}_{j}\right]=\operatorname{Pr}\left[\left(v_{i}, u_{j}\right) \in G_{v_{i}}(p)\right] \times \operatorname{Pr}\left[u_{j} \xrightarrow{G_{u_{j}}(p)} t\right]=(1-p) R_{u_{j}}$.

## Then we can

1. draw $j \in[d]$ with probability proportional to $p_{j}$;
2. draw $S \sim G_{u_{j}}(p)$ conditioned on $u_{j} \xrightarrow{s} t ;$ for $j^{\prime} \neq j$, draw $\left(v_{i}, u_{j^{\prime}}\right)$ independently with probability $1-p$; let the set of these edges together with $\left(\nu_{i}, u_{j}\right)$ be $S^{\prime}$;
3. let $t_{S^{\prime}}$ be the number of events $\mathcal{E}_{j}$ occurring under $S^{\prime}$; output $Z:=\frac{1}{t_{\mathrm{s}^{\prime}}}$

The same analysis implies that the expectation of this estimator is what we want and the relative variance is small.

We want to estimate $\operatorname{Pr}\left[\exists j \in[d],\left(\left(v_{i}, u_{j}\right) \in G_{v_{i}}(p)\right) \wedge\left(u_{j} \xrightarrow{G_{u_{j}}(p)} t\right)\right]=\operatorname{Pr}\left[\exists j \in[d], \varepsilon_{j}\right]$. Let $p_{j}:=\operatorname{Pr}\left[\varepsilon_{j}\right]=\operatorname{Pr}\left[\left(v_{i}, u_{j}\right) \in G_{v_{i}}(p)\right] \times \operatorname{Pr}\left[u_{j} \xrightarrow{G_{u_{j}}(p)} t\right]=(1-p) R_{u_{j}}$.
Then we can

1. draw $\mathrm{j} \in[\mathrm{d}]$ with probability proportional to $\mathrm{p}_{\mathrm{j}}$;
2. draw $S \sim G_{u_{j}}(\mathfrak{p})$ conditioned on $\mathfrak{u}_{\mathfrak{j}} \xrightarrow{s} \mathfrak{t}$; for $\mathfrak{j}^{\prime} \neq \mathfrak{j}$, draw $\left(v_{i}, \mathfrak{u}_{\mathfrak{j}^{\prime}}\right)$ independently with probability $1-p$; let the set of these edges together with $\left(v_{i}, u_{j}\right)$ be $S^{\prime}$;
3. let $t_{S^{\prime}}$ be the number of events $\varepsilon_{j}$ occurring under $S^{\prime}$; output $Z:=\frac{1}{t_{S^{\prime}}}$.

The same analysis implies that the expectation of this estimator is what we want and the relative variance is small.

We want to estimate $\operatorname{Pr}\left[\exists j \in[d],\left(\left(v_{i}, u_{j}\right) \in G_{v_{i}}(p)\right) \wedge\left(u_{j} \xrightarrow{G_{u_{j}}(p)} t\right)\right]=\operatorname{Pr}\left[\exists j \in[d], \varepsilon_{j}\right]$. Let $p_{j}:=\operatorname{Pr}\left[\varepsilon_{j}\right]=\operatorname{Pr}\left[\left(v_{i}, u_{j}\right) \in G_{v_{i}}(p)\right] \times \operatorname{Pr}\left[u_{j} \xrightarrow{G_{u_{j}}(p)} t\right]=(1-p) R_{u_{j}}$.
Then we can

1. draw $j \in[d]$ with probability proportional to $p_{j}$;
2. draw $S \sim G_{u_{j}}(p)$ conditioned on $u_{j} \xrightarrow{s} t ;$ for $\mathfrak{j}^{\prime} \neq \mathfrak{j}$, draw $\left(v_{i}, u_{j^{\prime}}\right)$ independently with probability $1-p$; let the set of these edges together with $\left(v_{i}, u_{j}\right)$ be $S^{\prime}$;
3. let $t_{S^{\prime}}$ be the number of events $\varepsilon_{j}$ occurring under $S^{\prime}$; output $Z:=\frac{1}{t_{S^{\prime}}}$.

The same analysis implies that the expectation of this estimator is what we want and the relative variance is small.

We want to estimate $\operatorname{Pr}\left[\exists j \in[d],\left(\left(v_{i}, u_{j}\right) \in G_{v_{i}}(p)\right) \wedge\left(u_{j} \xrightarrow{G_{u_{j}}(p)} t\right)\right]=\operatorname{Pr}\left[\exists j \in[d], \varepsilon_{j}\right]$. Let $p_{j}:=\operatorname{Pr}\left[\varepsilon_{j}\right]=\operatorname{Pr}\left[\left(v_{i}, u_{j}\right) \in G_{v_{i}}(p)\right] \times \operatorname{Pr}\left[u_{j} \xrightarrow{G_{u_{j}}(p)} t\right]=(1-p) R_{u_{j}}$.
Then we can

1. draw $\mathrm{j} \in[\mathrm{d}]$ with probability proportional to $\mathrm{p}_{\mathrm{j}}$;
2. draw $S \sim G_{u_{j}}(p)$ conditioned on $u_{j} \xrightarrow{S} t ;$ for $\mathfrak{j}^{\prime} \neq \mathfrak{j}$, draw $\left(v_{i}, u_{\mathfrak{j}^{\prime}}\right)$ independently with probability $1-p$; let the set of these edges together with $\left(v_{i}, u_{j}\right)$ be $S^{\prime}$;
3. let $t_{S^{\prime}}$ be the number of events $\mathcal{E}_{j}$ occurring under $S^{\prime}$; output $Z:=\frac{1}{t_{S^{\prime}}}$.

The same analysis implies that the expectation of this estimator is what we want and the relative variance is small.

We want to estimate $\operatorname{Pr}\left[\exists j \in[d],\left(\left(v_{i}, u_{j}\right) \in G_{v_{i}}(p)\right) \wedge\left(u_{j} \xrightarrow{G_{u_{j}}(p)} t\right)\right]=\operatorname{Pr}\left[\exists j \in[d], \varepsilon_{j}\right]$. Let $p_{j}:=\operatorname{Pr}\left[\mathcal{E}_{j}\right]=\operatorname{Pr}\left[\left(v_{i}, u_{j}\right) \in G_{v_{i}}(p)\right] \times \operatorname{Pr}\left[u_{j} \xrightarrow{\mathrm{G}_{u_{j}}(p)} \mathrm{t}\right]=(1-\mathfrak{p}) \mathrm{R}_{\mathbf{u}_{j}}$.
Then we can

1. draw $\mathrm{j} \in[\mathrm{d}]$ with probability proportional to $\mathrm{p}_{\mathrm{j}}$;
2. draw $S \sim G_{u_{j}}(p)$ conditioned on $u_{j} \xrightarrow{s}$ t; for $\mathfrak{j}^{\prime} \neq \mathfrak{j}$, draw $\left(v_{i}, u_{j^{\prime}}\right)$ independently with probability $1-p$; let the set of these edges together with $\left(v_{i}, u_{j}\right)$ be $S^{\prime}$;
3. let $t_{S^{\prime}}$ be the number of events $\mathcal{E}_{j}$ occurring under $S^{\prime}$; output $Z:=\frac{1}{t_{S^{\prime}}}$.

The same analysis implies that the expectation of this estimator is what we want and the relative variance is small.

We want to estimate $\operatorname{Pr}\left[\exists j \in[d],\left(\left(v_{i}, u_{j}\right) \in G_{v_{i}}(p)\right) \wedge\left(u_{j} \xrightarrow{G_{u_{j}}(p)} t\right)\right]=\operatorname{Pr}\left[\exists j \in[d], \varepsilon_{j}\right]$. Let $p_{j}:=\operatorname{Pr}\left[\varepsilon_{j}\right]=\operatorname{Pr}\left[\left(v_{i}, u_{j}\right) \in G_{v_{i}}(p)\right] \times \operatorname{Pr}\left[u_{j} \xrightarrow{G_{u_{j}}(p)} t\right]=(1-p) R_{u_{j}}$.
Then we can

1. draw $\mathrm{j} \in[\mathrm{d}]$ with probability proportional to $\mathrm{p}_{\mathrm{j}}$;
2. draw $S \sim G_{u_{j}}(p)$ conditioned on $u_{j} \xrightarrow{s} t ;$ for $\mathfrak{j}^{\prime} \neq \mathfrak{j}$, draw $\left(v_{i}, u_{j^{\prime}}\right)$ independently with probability $1-p$; let the set of these edges together with $\left(v_{i}, u_{j}\right)$ be $S^{\prime}$;
3. let $t_{S^{\prime}}$ be the number of events $\mathcal{E}_{j}$ occurring under $S^{\prime}$; output $Z:=\frac{1}{t_{S^{\prime}}}$.

The same analysis implies that the expectation of this estimator is what we want and the relative variance is small.

We want to estimate $\operatorname{Pr}\left[\exists j \in[d],\left(\left(v_{i}, u_{j}\right) \in G_{v_{i}}(p)\right) \wedge\left(u_{j} \xrightarrow{G_{u_{j}}(p)} t\right)\right]=\operatorname{Pr}\left[\exists j \in[d], \varepsilon_{j}\right]$. Let $p_{j}:=\operatorname{Pr}\left[\varepsilon_{j}\right]=\operatorname{Pr}\left[\left(v_{i}, u_{j}\right) \in G_{v_{i}}(p)\right] \times \operatorname{Pr}\left[u_{j} \xrightarrow{G_{u_{j}}(p)} t\right]=(1-p) R_{u_{j}}$.
Then we can

1. draw $\mathrm{j} \in[\mathrm{d}]$ with probability proportional to $\mathrm{p}_{\mathrm{j}}$;
2. draw $S \sim G_{u_{j}}(\mathfrak{p})$ conditioned on $\mathfrak{u}_{\mathfrak{j}} \xrightarrow{s} \mathfrak{t}$; for $\mathfrak{j}^{\prime} \neq \mathfrak{j}$, draw $\left(v_{i}, \mathfrak{u}_{\mathfrak{j}^{\prime}}\right)$ independently with probability $1-p$; let the set of these edges together with $\left(v_{i}, u_{j}\right)$ be $S^{\prime}$;
3. let $t_{S^{\prime}}$ be the number of events $\varepsilon_{j}$ occurring under $S^{\prime}$; output $Z:=\frac{1}{t_{S^{\prime}}}$.

The same analysis implies that the expectation of this estimator is what we want and the relative variance is small.

We want to estimate $\operatorname{Pr}\left[\exists j \in[d],\left(\left(v_{i}, u_{j}\right) \in G_{v_{i}}(p)\right) \wedge\left(u_{j} \xrightarrow{G_{u_{j}}(p)} t\right)\right]=\operatorname{Pr}\left[\exists j \in[d], \varepsilon_{j}\right]$. Let $p_{j}:=\operatorname{Pr}\left[\varepsilon_{j}\right]=\operatorname{Pr}\left[\left(v_{i}, u_{j}\right) \in G_{v_{i}}(p)\right] \times \operatorname{Pr}\left[u_{j} \xrightarrow{G_{u_{j}}(p)} t\right]=(1-p) R_{u_{j}}$.
Then we can

1. draw $j \in[d]$ with probability proportional to $p_{j}$;
2. draw $S \sim G_{u_{j}}(p)$ conditioned on $u_{j} \xrightarrow{s} t ;$ for $\mathfrak{j}^{\prime} \neq \mathfrak{j}$, draw $\left(v_{i}, u_{j^{\prime}}\right)$ independently with probability $1-p$; let the set of these edges together with $\left(v_{i}, u_{j}\right)$ be $S^{\prime}$;
3. let $t_{S^{\prime}}$ be the number of events $\varepsilon_{j}$ occurring under $S^{\prime}$; output $Z:=\frac{1}{t_{S^{\prime}}}$.

The same analysis implies that the expectation of this estimator is what we want and the relative variance is small.

## ARE WE THERE YET?

To implement Karp-Luby, we need to do two things:

- calculate $p_{j}=(1-p) R_{u_{j}}$, which can be done as $R_{\mathfrak{u}_{j}}$ is known in previous steps;
- sample $S \sim G_{u_{j}}(p)$ conditioned on $u_{j} \xrightarrow{S} t$.

How do we sample?

The sampling to counting reduction a la Jerrum, Valiant, and Vazirani (1986) to the rescue!

We maintain a set of samples $S_{v_{i}}$ in addition to $R_{v_{i}}$

## ARE WE THERE YET?

To implement Karp-Luby, we need to do two things:

- calculate $p_{j}=(1-p) R_{u_{j}}$, which can be done as $R_{\mathfrak{u}_{j}}$ is known in previous steps;
- sample $S \sim G_{u_{j}}(p)$ conditioned on $u_{j} \xrightarrow{s} t$.

How do we sample?

The sampling to counting reduction a la Jerrum, Valiant, and Vazirani (1986) to the rescue!

We maintain a set of samples $S_{v_{i}}$ in addition to $R_{v_{i}}$.

## SAMPLING TO COUNTING REDUCTION

## JVV'86 self-reduction:

sample edges one-by-one, using marginal probabilities conditioned on previous outcomes


The marginal of $e=\left(v_{i}, u_{1}\right)$ is
$\frac{\operatorname{Pr}\left[e \in G_{v_{i}}(p)\right] \operatorname{Pr}\left[\nu_{i} \xrightarrow{G_{v_{i}}(p)} t \mid e \in G_{v_{i}}(p)\right]}{\operatorname{Pr}\left[e \in G_{v_{i}}(p)\right] \operatorname{Pr}\left[v_{i} \xrightarrow{G_{v_{i}}(p)} t \mid e \in G_{v_{i}}(p)\right]+\operatorname{Pr}\left[e \notin G_{v_{i}}(p)\right] \operatorname{Pr}\left[v_{i} \xrightarrow{G_{v_{i}}(p)} t \mid e \notin G_{v_{i}}(p)\right]}$

## SAMPLING TO COUNTING REDUCTION

## JVV'86 self-reduction:

sample edges one-by-one, using marginal probabilities conditioned on previous outcomes


The marginal of $e=\left(v_{i}, u_{1}\right)$ is

$$
\frac{\operatorname{Pr}\left[e \in G_{v_{i}}(p)\right] \operatorname{Pr}\left[v_{i} \xrightarrow{G_{v_{i}}(p)} t \mid e \in G_{v_{i}}(p)\right]}{\operatorname{Pr}\left[e \in G_{v_{i}}(p)\right] \operatorname{Pr}\left[v_{i} \xrightarrow{\mathrm{G}_{v_{i}}(p)} \mathrm{t} \mid \mathrm{e} \in \mathrm{G}_{v_{i}}(p)\right]+\operatorname{Pr}\left[e \notin \mathrm{G}_{v_{i}}(p)\right] \operatorname{Pr}\left[v_{i} \xrightarrow{\mathrm{G}_{v_{i}}(p)} \mathrm{t} \mid e \notin \mathrm{G}_{v_{i}}(p)\right]}
$$

## SAMPLING TO COUNTING REDUCTION

## JVV'86 self-reduction:

sample edges one-by-one, using marginal probabilities conditioned on previous outcomes


The marginal of $e=\left(v_{i}, u_{1}\right)$ is

$$
\frac{(1-p) \operatorname{Pr}\left[v_{i} \xrightarrow{\mathrm{G}_{v_{i}}(p)} \mathrm{t} \mid \mathrm{e} \in \mathrm{G}_{v_{i}}(p)\right]}{(1-p) \operatorname{Pr}\left[v_{i} \xrightarrow{\mathrm{G}_{v_{i}}(p)} \mathrm{t} \mid \mathrm{e} \in \mathrm{G}_{v_{i}}(p)\right]+\mathrm{p} \operatorname{Pr}\left[v_{i} \xrightarrow{\mathrm{G}_{v_{i}}(p)} \mathrm{t} \mid \mathrm{e} \notin \mathrm{G}_{v_{i}}(p)\right]}
$$

## Computing the marginal

How do we estimate $\operatorname{Pr}\left[\nu_{i} \xrightarrow{\mathrm{G}_{v_{i}}(p)} \mathrm{t} \mid \mathrm{e} \in \mathrm{G}_{v_{i}}(\mathrm{p})\right]$ ?


## Computing the marginal

How do we estimate $\operatorname{Pr}\left[\nu_{i} \xrightarrow{\mathrm{G}_{v_{i}}(\mathrm{p})} \mathrm{t} \mid \mathrm{e} \in \mathrm{G}_{v_{i}}(\mathrm{p})\right]$ ?


Contract $v_{i}$ and $u_{1}$. Call it $v_{i}^{\prime}$. Note that we would have already computed $\left(R_{u_{j}}, S_{u_{j}}\right)$ for $\mathfrak{j} \geqslant 2$.

## Computing the marginal

How do we estimate $\operatorname{Pr}\left[\nu_{i} \xrightarrow{\mathrm{G}_{v_{i}}(\mathrm{p})} \mathrm{t} \mid \mathrm{e} \in \mathrm{G}_{v_{i}}(\mathrm{p})\right]$ ?


Contract $v_{i}$ and $u_{1}$. Call it $v_{i}^{\prime}$. Note that we would have already computed ( $\left.R_{u_{j}}, S_{u_{j}}\right)$ for $\mathfrak{j} \geqslant 2$.
That's all Karp-Luby needs to estimate $\operatorname{Pr}\left[\nu_{i}^{\prime} \xrightarrow{\mathrm{G}_{v_{i}^{\prime}}(\mathrm{p})} \mathrm{t}\right]$ !

## Computing the marginal

How do we estimate $\operatorname{Pr}\left[\nu_{i} \xrightarrow{\mathrm{G}_{v_{i}}(\mathrm{p})} \mathrm{t} \mid \mathrm{e} \in \mathrm{G}_{v_{i}}(\mathrm{p})\right]$ ?


Contract $v_{i}$ and $u_{1}$. Call it $v_{i}^{\prime}$. Note that we would have already computed ( $\left.R_{u_{j}}, S_{u_{j}}\right)$ for $\mathfrak{j} \geqslant 2$.
That's all Karp-Luby needs to estimate $\operatorname{Pr}\left[\nu_{i}^{\prime} \xrightarrow{\mathrm{G}_{v_{i}^{\prime}}(\mathrm{p})} \mathrm{t}\right]$ !
The case for $\operatorname{Pr}\left[v_{i} \xrightarrow{\mathrm{G}_{v_{i}}(\mathrm{p})} \mathrm{t} \mid \mathrm{e} \notin \mathrm{G}_{v_{i}}(\mathrm{p})\right]$ is similar. Then move on to the next edge and repeat.

## Computing the marginal

How do we estimate $\operatorname{Pr}\left[\nu_{i} \xrightarrow{\mathrm{G}_{v_{i}}(\mathrm{p})} \mathrm{t} \mid \mathrm{e} \in \mathrm{G}_{v_{i}}(\mathrm{p})\right]$ ?


Contract $v_{i}$ and $u_{1}$. Call it $v_{i}^{\prime}$. Note that we would have already computed ( $\left.R_{u_{j}}, S_{u_{j}}\right)$ for $\mathfrak{j} \geqslant 2$.
That's all Karp-Luby needs to estimate $\operatorname{Pr}\left[\nu_{i}^{\prime} \xrightarrow{\mathrm{G}_{v_{i}^{\prime}}(\mathrm{p})} \mathrm{t}\right]$ !
The case for $\operatorname{Pr}\left[v_{i} \xrightarrow{\mathrm{G}_{v_{i}}(\mathrm{p})} \mathrm{t} \mid \mathrm{e} \notin \mathrm{G}_{v_{i}}(\mathrm{p})\right]$ is similar. Then move on to the next edge and repeat.

## Computing the marginal

How do we estimate $\operatorname{Pr}\left[\nu_{i} \xrightarrow{\mathrm{G}_{v_{i}}(\mathrm{p})} \mathrm{t} \mid \mathrm{e} \in \mathrm{G}_{v_{i}}(\mathrm{p})\right]$ ?


- Blue edges: those chosen using the marginal by Karp-Luby, forming a tree.
- Green edges: conditioned on the blue edges, they have no affect on reaching $t$, and thus have marginal $1-p$.
- Red edges: the current frontier, whose tails have been processed (namely $\left(R_{u}, S_{u}\right)$ has been computed already).


## Efficiency of the algorithm

The self-reduction to generate one sample uses, say, $k$ samples for each Karp-Luby step. If we generate fresh samples each time, in total at least $\mathrm{k}^{\mathrm{n}}$ samples are required. The key to be efficient here is that samples can be reused!

## Reusing samples introduces subtle correlation among $R_{v_{i}}$ 's and the samples. However:

- accurate marginals

accurate samples;
- accurate samples $\square$ accurate marginals

Accurate marginals come from accurate estimation of the $R_{v_{i}}$ 's. The samples are subtly correlated, but they can be coupled with independent fresh samples with small error

## Efficiency of the algorithm

The self-reduction to generate one sample uses, say, $k$ samples for each Karp-Luby step. If we generate fresh samples each time, in total at least $k^{n}$ samples are required. The key to be efficient here is that samples can be reused!

Reusing samples introduces subtle correlation among $R_{v_{i}}$ 's and the samples. However:

- accurate marginals $\xrightarrow{\text { union bound over all edges }}$ accurate samples;
- accurate samples $\xrightarrow{\text { union bound over all possibilities }}$ accurate marginals.

Accurate marginals come from accurate estimation of the $R_{v_{i}}$ 's. The samples are subtly correlated, but they can be coupled with independent fresh samples with small error.

## The overall running time

We need union bounds over all edges at various points. Also they accumulate during the DP. Thus set the error $\delta:=n^{-1} \min \left\{\mathrm{~m}^{-1}, \varepsilon\right\}$ in Karp-Luby, and we need $\mathrm{O}\left(\mathrm{n} \delta^{-2}\right)$ samples each time. (In fact here we use the self-adjusting algorithm of Karp, Luby, and Madras, 1989.) To compute the estimator, we need $\mathrm{O}(\mathrm{m})$ time, so one Karp-Luby step takes $\mathrm{O}\left(\mathrm{mn} \delta^{-2}\right)$ time.

One run of Karp-Luby succeeds with constant probability. To show that reusing samples is fine, we need a union bound over all possible scenarios in the sampling algorithm, which are exponentially many. Thus, we repeat Karp-luby $\mathrm{O}(\mathrm{m})$ times to achieve $\exp (-\mathrm{O}(\mathrm{m}))$ failure probability.

How many samples do we need for each $S_{i}$ ?
On average, we need $0:=\frac{\mathrm{O}\left(\mathrm{mn}^{-2}\right)}{n}=\mathrm{O}\left(\mathrm{m}^{-2}\right)$ samples. This in fact suffices.
At each step of the DP, we do approximate counting once and sampling $\ell$ times, and each sampling step, in turn, calls approximate counting $\mathrm{O}(\mathrm{m})$ times.

Thus, the overall running time is

## The OVERALL RUNNING TIME

We need union bounds over all edges at various points. Also they accumulate during the DP. Thus set the error $\delta:=n^{-1} \min \left\{m^{-1}, \varepsilon\right\}$ in Karp-Luby, and we need $O\left(n \delta^{-2}\right)$ samples each time. (In fact here we use the self-adjusting algorithm of Karp, Luby, and Madras, 1989.) To compute the estimator, we need $\mathrm{O}(\mathrm{m})$ time, so one Karp-Luby step takes $\mathrm{O}\left(\mathrm{mn} \delta^{-2}\right)$ time.

One run of Karp-Luby succeeds with constant probability. To show that reusing samples is fine, we need a union bound over all possible scenarios in the sampling algorithm, which are exponentially many. Thus, we repeat Karp-luby $\mathrm{O}(\mathrm{m})$ times to achieve $\exp (-\mathrm{O}(\mathrm{m}))$ failure probability.

How many samples do we need for each $S_{i}$ ?
On average, we need $\ell:=\frac{\mathrm{O}\left(\mathrm{mn} \delta^{-2}\right)}{n}=\mathrm{O}\left(\mathrm{m}^{-2}\right)$ samples. This in fact suffices.

At each step of the DP, we do approximate counting once and sampling $\ell$ times, and each sampling step, in turn, calls approximate counting $\mathrm{O}(\mathrm{m})$ times.

## The overall running time

We need union bounds over all edges at various points. Also they accumulate during the DP. Thus set the error $\delta:=n^{-1} \min \left\{m^{-1}, \varepsilon\right\}$ in Karp-Luby, and we need $O\left(n \delta^{-2}\right)$ samples each time. (In fact here we use the self-adjusting algorithm of Karp, Luby, and Madras, 1989.) To compute the estimator, we need $\mathrm{O}(\mathrm{m})$ time, so one Karp-Luby step takes $\mathrm{O}\left(\mathrm{mn} \delta^{-2}\right)$ time.

One run of Karp-Luby succeeds with constant probability. To show that reusing samples is fine, we need a union bound over all possible scenarios in the sampling algorithm, which are exponentially many. Thus, we repeat Karp-luby $\mathrm{O}(\mathrm{m})$ times to achieve $\exp (-\mathrm{O}(\mathrm{m}))$ failure probability.

How many samples do we need for each $S_{i}$ ?
On average, we need $\ell:=\frac{\mathrm{O}\left(\mathrm{mn} \delta^{-2}\right)}{n}=\mathrm{O}\left(\mathrm{m} \delta^{-2}\right)$ samples. This in fact suffices.
At each step of the DP, we do approximate counting once and sampling $\ell$ times, and each sampling step, in turn, calls approximate counting $\mathrm{O}(\mathrm{m})$ times.

## The overall running time

We need union bounds over all edges at various points. Also they accumulate during the DP. Thus set the error $\delta:=n^{-1} \min \left\{m^{-1}, \varepsilon\right\}$ in Karp-Luby, and we need $O\left(n \delta^{-2}\right)$ samples each time. (In fact here we use the self-adjusting algorithm of Karp, Luby, and Madras, 1989.) To compute the estimator, we need $\mathrm{O}(\mathrm{m})$ time, so one Karp-Luby step takes $\mathrm{O}\left(\mathrm{mn} \delta^{-2}\right)$ time.

One run of Karp-Luby succeeds with constant probability. To show that reusing samples is fine, we need a union bound over all possible scenarios in the sampling algorithm, which are exponentially many. Thus, we repeat Karp-luby $\mathrm{O}(\mathrm{m})$ times to achieve $\exp (-\mathrm{O}(\mathrm{m}))$ failure probability.

How many samples do we need for each $S_{i}$ ?
On average, we need $\ell:=\frac{\mathrm{O}\left(\mathrm{mn} \delta^{-2}\right)}{n}=\mathrm{O}\left(\mathrm{m} \delta^{-2}\right)$ samples. This in fact suffices.
At each step of the DP, we do approximate counting once and sampling $\ell$ times, and each sampling step, in turn, calls approximate counting $\mathrm{O}(\mathrm{m})$ times.

[^0]
## The overall running time

We need union bounds over all edges at various points. Also they accumulate during the DP. Thus set the error $\delta:=n^{-1} \min \left\{m^{-1}, \varepsilon\right\}$ in Karp-Luby, and we need $O\left(n \delta^{-2}\right)$ samples each time. (In fact here we use the self-adjusting algorithm of Karp, Luby, and Madras, 1989.) To compute the estimator, we need $\mathrm{O}(\mathrm{m})$ time, so one Karp-Luby step takes $\mathrm{O}\left(\mathrm{mn} \delta^{-2}\right)$ time.

One run of Karp-Luby succeeds with constant probability. To show that reusing samples is fine, we need a union bound over all possible scenarios in the sampling algorithm, which are exponentially many. Thus, we repeat Karp-luby $\mathrm{O}(\mathrm{m})$ times to achieve $\exp (-\mathrm{O}(\mathrm{m}))$ failure probability.

How many samples do we need for each $S_{i}$ ?
On average, we need $\ell:=\frac{\mathrm{O}\left(\mathrm{mn} \delta^{-2}\right)}{n}=\mathrm{O}\left(\mathrm{m} \delta^{-2}\right)$ samples. This in fact suffices.
At each step of the DP, we do approximate counting once and sampling $\ell$ times, and each sampling step, in turn, calls approximate counting $\mathrm{O}(\mathrm{m})$ times.

Thus, the overall running time is

$$
n \times m \ell \times m \times O\left(m n \delta^{-2}\right)=O\left(m^{4} n^{2} \delta^{-4}\right)=O\left(m^{4} n^{6} \max \left\{m^{4}, \varepsilon^{-4}\right\}\right)
$$

Hardness results

## \#BIS-HARDNESS FOR TWO-TERMINAL VERTEX UNRELIABILITY IN DAGS

\#BIS
$\leqslant$


## \#BIS-HARDNESS FOR TWO-TERMINAL VERTEX UNRELIABILITY IN DAGS

\#BIS
$\leqslant$


## \#BIS-HARDNESS FOR TWO-TERMINAL VERTEX UNRELIABILITY IN DAGS

\#BIS
$\leqslant$
Two-terminal vertex unreliability


## \#BIS-HARDNESS FOR TWO-TERMINAL VERTEX UNRELIABILITY IN DAGS

\#BIS
$\leqslant$
Two-terminal vertex unreliability


## \#BIS-HARDNESS FOR Two-terminal vertex unreliability in DAGs

\#BIS
$\leqslant$
Two-terminal vertex unreliability

\#BIS
$\leqslant$
Two-terminal vertex unreliability

\#BIS
$\leqslant$
Two-terminal vertex unreliability


Vertex unreliability


## From vertex to edge unreliability in DAGs

## Vertex unreliability



## Edge unreliability



## From vertex to edge unreliability in DAGs

## Vertex unreliability



## Edge unreliability



## From vertex to edge unreliability in DAGs

## Vertex unreliability



## Edge unreliability



## Vertex unreliability



## Edge unreliability



Black edges have failure probability 0 . Colored edges have failure probability $1 / 2$.

We reduce from Minimum directed Steiner tree in DAGs with unit weights, which is NP-hard by reducing from Set Covers.

Note that minimum Steiner trees are the extremal configurations for $S-t$ reliability. Thus, if the failure probability of edges is $1-\exp (-\mathrm{O}(\mathrm{n}))$, then, drawing from the corresponding distribution, we would only see the minimizers.

Exponentially small success probability can be simulated by gadgets, such as replacing a normal edge by a path (sometimes called a stretching).

Concluding remarks

## Open problems

- Faster than $\mathrm{O}\left(m^{4} n^{6} \max \left\{m^{4}, \varepsilon^{-4}\right\}\right)$ ?
- Deterministic algorithms? This would require derandomise the Karp-Luby algorithm, which has been open for decades.
- Approximation complexity for Two-terminal reliability in directed and undirected graphs?
- Improve the quasi-polynomial time algorithm for \#CFG by Gore, Jerrum, Kannan, Sweedyk, and Mahaney (1997).


## Thank you!

arXiv:2310.00938


[^0]:    Thus, the overall running time is

