Sampling Spanning Trees in Dense Graphs in Sublinear Time

Joon Kim and Evan Ellis

December 2025

1 Motivation and Background

We consider the problem of sampling spanning trees proportional to the product of their edge weights, $p(T) \propto \prod_e w(e)$. This is a fundamental problem that serves as a baseline procedure in many graph algorithms, implying that speeding up sampling spanning trees can improve scaling for other important algorithms as well. For regimes of $|E| = \tilde{O}(|V|)$[1], there exist an optimal algorithm that runs in $\tilde{O}(|E|) = \tilde{O}(|V|)$ due to [ALG+21] . Thus, we are interested in dense graphs where $|V| << |E|$. [ALV22] shows that by appropriately preprocessing the graph in $\tilde{O}(|E|)$ time once, we can then sample a spanning tree in $\tilde{O}(|V|)$ time by sparsifying the graph to $O(|V|)$ edges and calling the $\tilde{O}(|E|)$ algorithm only $\tilde{O}(1)$ times. This is **sublinear** w.r.t. $|E|$, and in fact optimal up to polylog factors. We will study this algorithm in detail.

There is a good reason to be surprised that a sublinear algorithm exists at all for dense graphs. Intuitively, we should be tempted to at least look at all of the elements that can be sampled, which is already $O(|E|)$. This is generally true, since a naive algorithm would have no notion of which edges are "important," either due to their weights or the graph structure. However, if we assume that all edges have roughly the same probability of being included in a randomly sampled spanning tree (the marginals), then this problem disappears. This is reminiscent of assuming a convex body is "well-rounded" and "isotropic", such that an MCMC can more aggressively explore the body. In fact, [ALV22] also uses the term "isotropic" to describe such a configuration. Another parallel to volume estimation would be that preprocessing is slightly more costly than sampling.

The mathematical workhorse behind bounding the number of calls to the baseline algorithm to only $\tilde{O}(1)$ is **Entropic Independence**. The main idea is that once we preprocess the edges to be (approximately) isotropic, the entropy contracts fast (on average) for a Down-Up sampling MCMC over a complement set. The two handwavy terms, approximately and on average, are rigorously justified by concentration bounds such that they would not affect the overall runtime. The structure of the note is as follows: first, we define the bare minimum terms required to understand the algorithm. Next, a concrete algorithm is presented. Finally, an in-depth analysis using entropic independence justifies the choice of parameters in the algorithm.

2 Preliminaries

Let $\mu : \binom{[n]}{k} \to \mathbb{R}_{\geq 0}$ be a distribution over k-subsets of n elements. In the context of spanning trees, $[n] = E$ and $k = (|V| - 1)$.[2] All spanning trees live in this space. Then, it might be natural to ask the question, "for a given edge, what is the probability that it will be included in a randomly sampled spanning tree?" This is the concept of marginals. It is notable that the sum of all marginals, $\sum_{i \in [n]} p(\mu)_i = k$.

Definition 2.1 *$p(\mu)_i := \Pr_{S \sim \mu}[i \in S]$ is a **marginal** of element $i \in supp(\mu)$. $p(\mu)^{max} := \max_i \{p(\mu)_i\}$.*

A motivation for studying this quantity is that if we had access to an oracle that output exact marginal probabilities, sampling would be trivial via the chain rule (iteratively including or excluding edges). But

[1] We suppress polylog factors via $\tilde{O}$ notation throughout this note.

[2] This is **not** the standard notation of $n = |V|, m = |E|$. However, this is for the sake of being consistent with the paper.

while counting spanning trees is in P via the Matrix Tree Theorem, it is computationally expensive, violating sublinearity. Hence, [ALV22] turns to a tractable alternative of calculating marginal **overestimates**.

Definition 2.2 $(p_1, \ldots, p_n) \in \mathbb{R}^n$ *are* ***marginal overestimates*** *of* μ *if* $\forall i \in [n]$, $p(\mu)_i \leq p_i$.

Proposition 2.3 *There exist marginal overestimates* $(p_1, \ldots, p_n)$ *s.t.* $\sum_{i\in[n]} p_i = K = O(k)$, *and there exists an* $\tilde{O}(n)$ *algorithm to calculate it.*

[ALV22] develops their own divide-and-conquer method that runs in $\tilde{O}(n)$ time in their Section 5 . Notably, this is also possible with the previously mentioned exact solver via the Matrix Tree Theorem, since the appeal of the algorithm is that we only need to get the marginal overestimate once before producing spanning trees. However, this still does not solve the issue that certain marginals may be magnitudes higher than others, which leads to a bottleneck in sampling. The answer to this is to perform an **isotropic transformation** on μ such that it has low marginals everywhere while not blowing up the number of elements. If the distribution is isotropic, no single edge is 'too important,' so we can add or delete edges randomly without changing the structure too much.

Definition 2.4 *For* μ *and marginal overestimates* $(p_1, \ldots, p_n)$ *s.t.* $\sum p_i \leq K$, *let* $t_i := \lceil \frac{n}{K} p_i \rceil$. *Create* t_i *copies of element* i *and let the collection of all these copies be* U. *Then, the* ***isotropic transformation*** $\mu' : \binom{U}{k} \to \mathbb{R}_{\geq 0}$ *of* μ *is defined to be* $\mu'(\{i_1^{j_1}, \ldots, i_k^{j_k}\}) := \frac{\mu(i_1,\ldots,i_k)}{t_1,\ldots,t_k}$.

An easy way of thinking about the transformed distribution is to imagine sampling from μ and then for each element i, choose one of t_i copies of it u.a.r. If we ignore copies, the support of μ' is the same as μ.

Proposition 2.5 *For some* μ *and its isotropic transformation* μ',

1. *Near isotropy:* $\forall i^{(j)} \in U$, *marginal* $p(\mu')_{i^{(j)}} = \Pr_{S\sim\mu'}[i^{(j)} \in S] \leq \frac{K}{n}$.
2. *Linear ground set size:* $|U| \leq 2n$.[3]

Combining the two statements, we can deduce that the sum of marginals of μ' is still $O(K) = O(k)$. Also, since we assumed that $k << n$, each marginal will be bounded by some small constant such as $1/100$. This is conceptually useful for analysis. Finally, we introduce one more useful definition.

Definition 2.6 *A* ***restricted distribution*** *for* $S \subseteq [n]$ *is* $\mu_S :=$ *distibution of* $F \sim \mu$ *s.t.* $F \subseteq S$.

Intuitively, a restricted distribution for S only considers spanning trees such that all its edges are in S and renormalizes their probabilities to 1. This is useful when we observe that Spanning Trees are *negatively correlated* (strongly Rayleigh, or SR). Formally, if we restrict the distribution to a subset of edges T (conditioning on $F \subseteq T$), we effectively discard all edges outside T. This removal of "competitors" causes the marginal probability of the remaining edges $e \in T$ to increase. Formally, this is described as $P_{S\sim\mu}[i \in S] \leq P_{S\sim\mu_T}[i \in S]$ for $i \in T$.[4]

For simplicity, we claim that all relevant distributions throughout this text remain SR and approximately isotropic (w.v.h.p.) across transformations and such, as some theorems only work with them as assumptions. Justification of this claim is rigorously analyzed in Section 4 of [ALV22]. Through a martingale argument, they prove that marginals do not drift far from their initial values throughout the sampling procedure. This is the high-level reasoning why one preprocessing step suffices to produce arbitrarily many spanning trees in the future.

[3] A quick sanity check is to observe that $\sum t_i \leq \sum_{i=1}^{n} (\frac{n}{K} p_i + 1) \leq n + n = 2n$.

[4] The paper likely has a typo for this definition. They used "restricted" as "discarding" in this explanation, which directly overrides Definition 2.6 that they just defined.

3 The Algorithm

Algorithm 1 Sampling Spanning Trees on Dense Graphs

Require: Graph $G = (V, E)$ where $|E| >> |V|$, Weight Function $w : e \to \mathbb{R}_{>0}$
Ensure: M spanning trees sampled from distribution $\mu(T) \propto \prod w(e)$

1: Calculate marginal overestimates $p_1, \ldots, p_n$. ▷ $\sum_{i\in[n]} p_i = K = O(k)$
2: Perform isotropic transformation $\mu \to \mu'$. ▷ $p(\mu')^{max} = O(\frac{K}{n})$
3: $S_0 \leftarrow$ Initial Spanning Tree ▷ $|S_0| = k = (|V| - 1)$
4: **repeat** M **times**
5: **repeat** $O(\kappa^{-1} \cdot \log n)$ **times** ▷ $\kappa^{-1} = O(\log^2 n)$ for choice of t below
6: Sample u.a.r. $T \in \binom{[n]-S_0}{t-k}$ ▷ $t = \Theta(n \cdot p(\mu')^{max}) = O(K) = O(k)$
7: Downsample $S_1 \sim \mu'_{S_0 \cup T}$ ▷ $|S_0 \cup T| = O(k)$
8: Update $S_0 \leftarrow S_1$
9: **end repeat**
10: Output S_0
11: **end repeat**

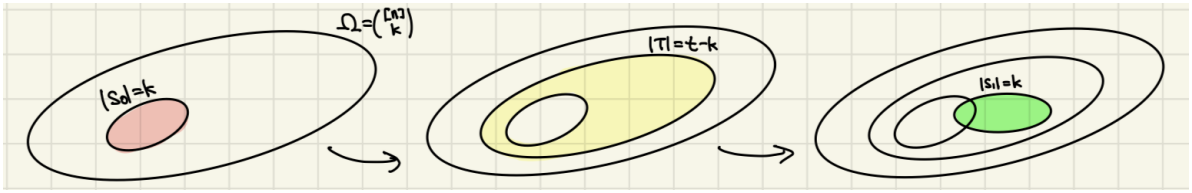

Figure 1: One step of MCMC

Now we are equipped to understand Algorithm 1. The algorithm yields M spanning trees sampled from graph $G = (V, E)$ and corresponding weight function $w(e)$ in $\tilde{O}(|E|+N|V|)$ time. Lines 1-3 are preprocessing steps that take $\tilde{O}(|E|)$ time, but once we pay that cost, every new spanning tree only takes $\tilde{O}(|V|)$ to generate via an MCMC in lines 5-10.

Performing preprocessing guarantees that we have access to μ' such that all marginals are approximately isotropic ($p(\mu')^{max} = O(\frac{K}{n})$). Initializing S_0 is easy since it is independent of edge weights.

We now generate M spanning trees. For each run, we repeat the procedure of randomly adding edges (upsampling) up to some t edges and then downsampling back to k edges according to the appropriate restricted distribution $\mu'_{S_0 \cup T}$. If we choose $t = \Theta(n \cdot p(\mu')^{max})$, then it suffices to repeat only $O(\kappa^{-1} \cdot \log n) = O(\log^3 n) = \tilde{O}(1)$ times for mixing such that $d_{TV}(\nu, \pi) = n^{-O(1)}$.[5] Thus, if we show that one loop takes $\tilde{O}(k)$ time, we are done.

The mixing time and runtime of each step is dictated by t. The choice of t is quite mysterious at first glance. Indeed, the result that $\kappa^{-1} = O(\log^2 n)$ a deeply nontrivial analysis involving entropic independence. For now, we can think of the parameter t somehow striking a perfect balance to allow a $\tilde{O}(1)$ mixing time for each MCMC round while keeping the size of $S_0 \cup T = O(k)$. Then, the downsampling is handled by the baseline sampling algorithm from [ALG+21], which only takes $\tilde{O}(k) = \tilde{O}(|V|)$ time. The upsampling step is also efficient since we can enumerate all edges with $O(\log n)$ bits and repeatedly toss coins until the edge set is saturated with $t = O(k)$ edges. Effectively, we have reduced the problem of sampling from $O(|E|)$ elements to only $O(|V|)$ elements at the cost of repeating only $\tilde{O}(1)$ times. This is the "sparsifying" aspect of this algorithm.

It is educational to think about the effect of setting a different value of t. If we take a large t, the mixing time will be small because the MCMC aggressively explores the states, but each call to the baseline algorithm

[5] This is the byproduct of MLSI and Pinsker's inequality, but it's also useful as it fools any $poly(n)$ sample algorithm.

will no longer be $O(|V|)$. On the other hand, a small t may improve the baseline algorithm's runtime but slows down the convergence of the MCMC, possibly breaking the sublinearity of the algorithm.[6]

One might be reminiscent of the Down-Up sampling that was covered in class for sampling matroid bases. In fact, one inner loop of upsampling-downsampling is exactly a Down-Up sampling on the **complement set**, $\bar{\mu}(S) := \mu([n] - S)$. Formally, this is captured by the transition $\bar{\mu}D_{(n-k)\to(n-t)}U_{(n-t)\to(n-k)}$.[7] This is moderately confusing, but it is helpful to recall that the Down step removes samples u.a.r., whereas the the Up step must incorporate information about the importance of sampling each marginal. This is an important part of the analysis.

Now, we are left with justifying the choice of t. This is the main knowledge contribution of this paper, which extensively uses the notion of entropic independence to claim fast entropic contraction for near-isotropic distributions.

4 Fast Entropy Contraction

To choose t, we will show how it influences the entropy contraction of the $D_{(n-k)\to(n-t)}$ operator. We do not analyze the contraction of $U_{(n-t)\to(n-k)}$, simply noting that it cannot increase divergence by the data processing inequality [CT06]. We will first show that the $D_{(n-k)\to 1}$ down operator contracts entropy more strongly if $\mu \in \mathbb{R}^{\binom{[n]}{k}}$ has nearly uniform marginals. Then, we use an average-case local-to-global argument to bound the contraction for an arbitrary t. By the preprocessing step which ensures μ is near-isotropic we have fast mixing of the down-up walk.

Theorem 4.1 *[AJK+21]. Any strongly Rayleigh distribution $\mu : \binom{[n]}{k} \to \mathbb{R}_{\geq 0}$ is 1-entropically independent:*

$$D_{KL}(vD_{k\to 1} \mid \mu D_{k\to 1}) \leq \frac{1}{k} D_{KL}(v||\mu). \tag{1}$$

Theorem 4.2 *Let $\mu \in \mathbb{R}^{\binom{[n]}{k}}$ be a 1-entropically independent distribution per Theorem 4.1 with $p(\mu)^{\max} \leq \frac{1}{100}$. Then for any distribution $\bar{v} \subseteq \mathbb{R}^{\binom{[n]}{n-k}}$,*

$$D_{KL}(\bar{v}D_{(n-k)\to 1}||\bar{\mu}D_{(n-k)\to 1}) \leq \frac{1}{(n-k)\log\left((ep(\mu)^{\max})^{-1}\right)} D_{KL}(\bar{v}||\bar{\mu}). \tag{2}$$

By Theorem 4.1, if μ is strongly Rayleigh then it is also 1-entropically independent, so this result applies to all near-isotropic distributions that we care about. Near isotropy is important for another reason: if every set in the support of $\bar{\mu}$ shares an edge e_i then $p(\bar{\mu})_i = (\bar{\mu}D_{(n-k)\to 1})_i = 0$, so the KL would be undefined. Intuitively, we can also view this bound as exploiting the fact that the uniform distribution, such as $\bar{\mu}D_{(n-k)\to 1}$, minimizes the worst-case KL divergence with another distribution. The distributions in question have intuitive descriptions: $\bar{v}D_{(n-k)\to 1}$ is the normalized distribution over the edges not included in sets sampled from v. $\bar{\mu}D_{(n-k)\to 1}$ is the same for μ.

To prove Theorem 4.2, we will first relate the divergence between the distributions of edges that are not included with the divergence of the distributions of edges that are included.

Lemma 4.3 *If $p(\mu)^{\max} \leq \frac{1}{100}$ and $\frac{k}{n} \leq \frac{1}{100}$, then for any distribution $\bar{v} \subseteq \mathbb{R}^{\binom{[n]}{n-k}}$*

$$D_{KL}\left(\bar{v}D_{(n-k)\to 1} \,||\, \bar{\mu}D_{(n-k)\to 1}\right) \leq C \cdot D_{KL}\left(vD_{k\to 1}||\mu D_{k\to 1}\right)$$

for $C = \frac{k}{(n-k)\log\left((ep(\mu)^{\max})^{-1}\right)}$.

[6] The paper proves that a one-step Down-Up MCMC ($t = k+1$) requires $\tilde{O}(K)$ calls to the baseline instead of $\tilde{O}(1)$ calls.
[7] We drop the notation μ' in favor of μ and assume it is approximately isotropic from now on.

This relation is useful because it allows us to use the 1-entropic independence of μ to show that $D_{(n-k)\to 1}$ contracts entropy faster when we know that $p(\mu)^{\max} \leq \frac{1}{100}$.

Proof of Theorem 4.3. We first note without proof that $\forall v \in \mathbb{R}^{\binom{[n]}{k}}$, $\forall i \in n$, if $p(\mu)^{\max} \leq 1/100$,

$$\frac{p(v_i)}{\log\left((ep(\mu)_i)^{-1}\right)} \log\left(\frac{p(v)_i}{p(\mu)_i}\right) - p(\bar{v})_i \log\left(\frac{p(\bar{v})_i}{p(\bar{\mu})_i}\right) \geq \left(1 + \frac{k}{\log\left((ep(\mu)_i)^{-1}\right)}\right)(p(v)_i - p(\mu)_i). \tag{3}$$

Now we can sum Equation (3). First, note that the right-hand side's dependence on $\log p(\mu)_i$ can be lower-bounded with $\log p(\mu)^{\max}$. Similarly, the left-hand side's dependence on $\log p(\mu)_i$ can be upper-bounded with $\log p(\mu)^{\max}$. We can now sum the inequality over all i and cancel terms:

$$\sum_i \frac{p(v_i)}{\log\left((ep(\mu)^{\max})^{-1}\right)} \log\left(\frac{p(v)_i}{p(\mu)_i}\right) - \sum_i p(\bar{v})_i \log\left(\frac{p(\bar{v})_i}{p(\bar{\mu})_i}\right) \geq \sum_i \left(1 + \frac{k}{\log\left((ep(\mu)^{\max})^{-1}\right)}\right)(p(v)_i - p(\mu)_i)$$

$$\frac{1}{\log\left((ep(\mu)^{\max})^{-1}\right)} \sum_i p(v_i) \log\left(\frac{p(v)_i}{p(\mu)_i}\right) - \sum_i p(\bar{v})_i \log\left(\frac{p(\bar{v})_i}{p(\bar{\mu})_i}\right) \geq 0$$

$$\frac{1}{\log\left((ep(\mu)^{\max})^{-1}\right)} \sum_i k\left(vD_{k\to 1}\right)_i \log\left(\frac{(vD_{k\to 1})_i}{(\mu D_{k\to 1})_i}\right) \geq \sum_i (n-k)(\bar{v}D_{(n-k)\to 1})_i \log\left(\frac{(\bar{v}D_{(n-k)\to 1})_i}{(\bar{\mu}D_{(n-k)\to 1})_i}\right)$$

$$\frac{k}{(n-k)\log\left((ep(\mu)^{\max})^{-1}\right)} D_{\mathrm{KL}}(vD_{k\to 1}||\mu D_{k\to 1}) \geq D_{\mathrm{KL}}(\bar{v}D_{(n-k)\to 1}||\bar{\mu}D_{(n-k)\to 1}).$$

This proves Theorem 4.3. We did not explicitly show Equation (3), but it may be proved with calculus. The requirement that $p(\mu)^{\max} < \frac{1}{100}$ is used to show that the second derivative is positive, because it relies on $1 - p(\mu)^{\max}(1 + \log(1/(ep(\mu)^{\max}))) = 1 + p(\mu)^{\max} \log p(\mu)^{\max} \geq 0$ when $p(\mu)^{\max} \leq \frac{1}{100}$.

Proof of Theorem 4.2. First, note that $D_{\mathrm{KL}}(\bar{v}||\bar{\mu}) = D_{\mathrm{KL}}(v||\mu)$. Now we can directly combine Theorem 4.3 with 1-entropic independence (REF) of μ to show Theorem 4.2:

$$\begin{aligned} D_{\mathrm{KL}}(\bar{v}D_{(n-k)\to 1}||\bar{\mu}D_{(n-k)\to 1}) &\leq C \cdot D_{\mathrm{KL}}(vD_{k\to 1}||\mu D_{k\to 1}) \\ &\leq \frac{C}{k} D_{\mathrm{KL}}(\bar{v}||\bar{\mu}) \\ &= \frac{1}{(n-k)\log\left((ep(\mu)^{\max})^{-1}\right)} D_{\mathrm{KL}}(\bar{v}||\bar{\mu}). \end{aligned}$$

This is a direct result of our isotropic construction, which enables Theorem 4.3. Without isotropy, we would still have a contraction, but a smaller one.

Now we show that this result can be extended to bound the contraction of all choices of $D_{(n-k)\to(n-t)}$, not just $D_{(n-k)\to 1}$. This is an **average-case** argument, arguing that with high-likelihood the entropy will be contracted by this amount.

Theorem 4.4 *If $p(\mu)^{\max} \leq 1/500$, then $\forall t > k$,*

$$D_{KL}(\bar{v}D_{(n-k)\to(n-t)}||\bar{\mu}D_{(n-k)\to(n-t)}) \leq \left(1 - \frac{t-k}{2C\left(np(\mu)^{\max} + t + 1\right)\log^2 n}\right) D_{KL}(\bar{v}||\bar{\mu})$$

where C is a constant that ensures the distribution remains approximately isotropic.

We will refer to the rate of contraction as $(1-\kappa)$. Examining this formula, if we choose t to be large $(O(n))$ then we have a faster contraction. However, the cost of the down and up operators is $\tilde{O}(t)$, so we spend more compute per step. The optimal choice is $t = \Theta(np(\mu)^{\max})$, which gives us $\kappa^{-1} = O(\log^2 n)$ and $O(np(\mu)^{\max}) = O(K) = O(k)$ operations per step.

Finally, we plug in κ into the discrete modified log-Sobolev inequality [BT06] to bound the mixing time.

Theorem 4.5 *For the approximately isotropic down-up walk on the spanning tree with* $t = \Theta(np(\mu)^{\max})$*, we achieve* $n^{-O(1)}$ *total variation distance from* μ *in* $O(\log^2 n)$ *calls to* $O(k)$*. The total time complexity is* $O(k \log^2 n)$*.*

References

[AJK+21] Nima Anari, Vishesh Jain, Frederic Koehler, Huy Tuan Pham, and Thuy-Duong Vuong. Entropic independence in high-dimensional expanders: Modified log-sobolev inequalities for fractionally log-concave polynomials and the ising model. *CoRR*, abs/2106.04105, 2021.

[ALG+21] Nima Anari, Kuikui Liu, Shayan Oveis Gharan, Cynthia Vinzant, and Thuy-Duong Vuong. Log-concave polynomials IV: approximate exchange, tight mixing times, and near-optimal sampling of forests. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2021, pages 408–420, New York, NY, USA, 2021. Association for Computing Machinery.

[ALV22] Nima Anari, Yang P. Liu, and Thuy-Duong Vuong. Optimal Sublinear Sampling of Spanning Trees and Determinantal Point Processes via Average-Case Entropic Independence. pages 123–134, Denver, CO, USA, October 2022. IEEE.

[BT06] Sergey G. Bobkov and Prasad Tetali. Modified logarithmic sobolev inequalities in discrete settings. *Journal of Theoretical Probability*, 19(2):289–336, Jun 2006.

[CT06] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory 2nd Edition (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, July 2006.