

Quadratic Speedups in Parallel Sampling from Determinantal Distributions

Nima Anari¹, Callum Burgess¹, Kevin Tian², and Thuy-Duong Vuong¹

¹Stanford University, {anari, callumb, tdvuong}@stanford.edu

²Microsoft Research, tiankevin@microsoft.com

Abstract

We study the problem of parallelizing sampling from distributions related to determinants: symmetric, nonsymmetric, and partition-constrained determinantal point processes, as well as planar perfect matchings. For these distributions, the partition function, a.k.a. the count, can be obtained via matrix determinants, a highly parallelizable computation; Csanky proved it is in NC. However, parallel counting does not automatically translate to parallel sampling, as classic reductions between the two are inherently sequential. We show that a nearly quadratic parallel speedup over sequential sampling can be achieved for all the aforementioned distributions. If the distribution is supported on subsets of size k of a ground set, we show how to approximately produce a sample in $\tilde{O}(k^{\frac{1}{2}+c})$ time with polynomially many processors for any constant $c > 0$. In the two special cases of symmetric determinantal point processes and planar perfect matchings, our bound improves to $\tilde{O}(\sqrt{k})$ and we show how to sample exactly in these cases.

As our main technical contribution, we fully characterize the limits of batching for the steps of sampling-to-counting reductions. We observe that only $O(1)$ steps can be batched together if we strive for exact sampling, even in the case of nonsymmetric determinantal point processes. However, we show that for approximate sampling, $\tilde{\Omega}(k^{\frac{1}{2}-c})$ steps can be batched together, for any entropically independent distribution, which includes all mentioned classes of determinantal point processes. Entropic independence and related notions have been the source of breakthroughs in Markov chain analysis in recent years, so we expect our framework to prove useful for distributions beyond those studied in this work.

1 Introduction

Sampling and counting are intimately connected problems. For many classes of measures μ defined on a, most often exponentially large, space X , approximately sampling $x \in X$ with $\mathbb{P}[x] \propto \mu(x)$ and approximately computing the partition function $\sum_{x \in X} \mu(x)$ are polynomial-time reducible to each other [JV86]. However, this equivalence breaks down for complexity classes below P. For example, there is no known polylogarithmic-time parallel reduction between approximate counting and approximate sampling.

Motivated by the underexplored relationship between sampling and counting in the world of parallel algorithms, Anari, Hu, Saberi, and Schild [Ana+20], based on the earlier work of Teng [Ten95], raised the question of designing fast parallel *samplers* for several distributions where counting, even exactly, was possible in polylogarithmic time and polynomial work, i.e., via NC algorithms. The distributions in this challenge set all enjoy fast parallel counting algorithms because their partition functions, $\sum_x \mu(x)$, can be written as determinants, and determinants are computable in NC [Csa75]. Anari, Hu, Saberi, and Schild [Ana+20] solved one of these challenges and showed how to sample random arborescences in RNC, completing the earlier work of Teng [Ten95] on random spanning trees. However, these works are tailored to the random spanning tree and arborescence distributions, as they parallelize the sequential sampling algorithm of Aldous [Ald90] and Broder [Bro89], and there is no known generalization of this algorithm to other distributions.

In this work, we study a general framework to improve the parallel efficiency of sampling-to-counting reductions. We build on the success of recently introduced techniques in the analysis of random walks and sampling algorithms, where combinatorial distributions are analyzed through the lens of high-dimensional expanders [Ana+19; AL20; ALO20]. We show that under one notion of high-dimensional expansion named entropic independence [Ana+21b], see Definition 22 for definition, sampling-to-counting reductions can be made sped up nearly quadratically on a PRAM.

To set up notation, we consider combinatorial distributions defined on size k subsets of a ground set of elements $\{1, \dots, n\}$, which we denote by an (unnormalized) measure $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$. We note that the choice of $\binom{[n]}{k}$ is a standard canonical form for the domain, and many other domains such as product spaces, can be transformed into $\binom{[n]}{k}$ [ALO20]. We access the measure μ through a counting oracle. Given *any*¹ set $T \subseteq [n]$, the oracle returns

$$\sum \left\{ \mu(S) \mid S \in \binom{[n]}{k}, T \subseteq S \right\}.$$

Our goal is to use the oracle and output a random set S that approximately follows $\mathbb{P}[S] \propto \mu(S)$.

The classic reduction from sampling to counting [JV86] proceeds by picking the k elements of S , one at a time. In each step, conditioned on all previously chosen elements, marginals $\mathbb{P}_{S \sim \mu}[i \in S \mid \text{previous choices}]$ of all remaining elements i are computed and a new element is picked randomly with probability proportional to the conditional marginals. In each step, marginals can be computed via parallel calls to the counting oracle. However, this procedure is inherently sequential as the choice of each element affects the conditional marginals in future iterations. A parallel implementation of this reduction takes time $\Omega(k)$. The main question we study is:

For which μ is there a faster parallel reduction from sampling to counting?

¹By querying a T of size exactly k , a counting query can also return $\mu(T)$.

Our main result shows that for distributions μ which are entropically independent [Ana+21b], sampling-to-counting reductions can be sped up roughly quadratically. Throughout, we use $\tilde{O}(\cdot)$ to hide logarithmic factors in n and failure probabilities; these factors primarily come from the parallel complexity of linear algebra (e.g., evaluating determinants and partition functions).

Theorem 1 (Main, see Theorem 29 for formal version). *Let $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$ be entropically independent and assume that we have access to a counting oracle for μ . For any constant $c > 0$ and any $\epsilon > 0$, there exists an algorithm that can sample from a distribution within total variation distance ϵ of μ in $\tilde{O}\left(\sqrt{k} \cdot \left(\frac{k}{\epsilon}\right)^c\right)$ parallel time using $(n/\epsilon)^{O(1/c)}$ machines in the PRAM model.*

As a corollary, we get improved parallel sampling for various classes of determinantal point processes: symmetric, non-symmetric, and partition-constrained. All of these distributions are known to be entropically independent [Ali+21; Ana+21b]. As an additional result, we show that by using different graph-separator-based techniques, a similar quadratic parallel speedup can be obtained for sampling planar perfect matchings, the only distribution in the challenge set of [Ana+20] not covered by Theorem 1; see Section 6 for details.

Remark 2 (Beyond determinantal distributions). Various notions of high-dimensional expansion, and in particular, entropic independence, have recently resulted in breakthroughs in the analysis of Markov chains and sequential sampling algorithms [Ana+21b; Ana+21c], but as far as we know, this is the first work to relate these notions to parallel sampling. Here we explore the applications of Theorem 1 to distributions related to determinants because the counting oracle for them can be implemented in RNC. However, we suspect Theorem 1 can have applications beyond determinantal distributions in the future. As an example, for distributions whose partition functions do not have roots in certain regions of the complex plane, Barvinok [Bar18] devised efficient deterministic approximate counting algorithms, which have been refined by subsequent works [PR17]. These counting algorithms can be parallelized in many settings, as they involve enumerating a polynomial/quasi-polynomial number of small, logarithmic-sized, combinatorial substructures. Recent works [Ali+21; CLV21] have shown that the absence of roots in the complex plane implies certain forms of high-dimensional expansion, including entropic independence [Ana+21b].

1.1 Determinantal distributions

This work considers applications of Theorem 1 to various distributions μ defined based on determinants. Prior progress on designing parallel samplers for problems that enjoy determinant-based counting has been limited. Teng [Ten95] showed how to simulate random walks on a graph in parallel, which combined with the classic algorithm of Aldous [Ald90] and Broder [Bro89] yielded RNC samplers for uniformly random spanning trees in a graph. Anari, Hu, Saberi, and Schild [Ana+20] extended this to sampling arborescences (directed spanning trees) in directed graphs. In this work, we tackle a much larger class of problems that enjoy determinant-based counting, namely variants of determinantal point processes.

Determinantal point processes (DPPs) have found many applications, such as data summarization [Gon+14; LB12], recommender systems [GPK16; Wil+18], neural network compression [MS15], kernel approximation [LJS16], multi-modal output generation [Elf+19], and randomized numerical linear algebra [DM21]. Formally, a DPP on a set of items $[n] = \{1, \dots, n\}$ is a probability distribution over subsets $Y \subseteq [n]$ defined via an $n \times n$ matrix L where probabilities are given (proportionally) by principal minors: $\mathbb{P}[Y] \propto \det(L_{Y,Y})$. The partition function of such a distribution is simply $\det(L + I)$,

which can be efficiently computed in parallel.

Note that for the distribution to be well-defined, all principal minors of L have to be ≥ 0 . For symmetric L , that is $L = L^\top$, this is equivalent to L being positive semi-definite (PSD). Symmetric DPPs, where $L = L^\top \geq 0$ have received the most attention in the literature.

Definition 3 (Symmetric DPP). Given a symmetric $n \times n$ matrix $L \geq 0$, the symmetric DPP defined by L is the probability distribution over subsets $Y \subseteq [n]$, where $\mathbb{P}[Y] \propto \det(L_{Y,Y})$.

Beyond (symmetric) determinantal point processes, our work provides sampling algorithms for a variety of other discrete distributions related to determinants: non-symmetric and partition-constrained DPPs, as well as planar perfect matchings. Next, we will outline each family of such distributions.

Recently, [Bru18; Gar+19; Gar+20] initiated the study of non-symmetric DPPs in applications and argued for their use because of increased modeling power. Symmetric DPPs necessarily exhibit strong forms of negative dependence [BBL09], which are unrealistic in some applications; on the other hand, non-symmetric DPPs can have positive correlations. Non-symmetric DPPs are characterized by a non-symmetric positive-definite matrix L , i.e., a matrix L where $L + L^\top \geq 0$.

Definition 4. A matrix $L \in \mathbb{R}^{n \times n}$ is *non-symmetric positive semidefinite* (nPSD) if $L + L^\top \geq 0$.

Definition 5 (Non-symmetric DPP). Given an nPSD $n \times n$ matrix L , the non-symmetric DPP defined by it is the probability distribution over subsets $Y \subseteq [n]$ given by $\mathbb{P}[Y] \propto \det(L_{Y,Y})$.

A related and more commonly used model related to DPPs is a k -DPP, where we constrain the cardinality of the sampled set Y to be exactly k . In many applications, restricting to sets of a predetermined size is more desirable [KT12b].

Definition 6 (k -DPP). Given a PSD or nPSD matrix L , the k -DPP defined by it is the distribution of the corresponding determinantal point process restricted to only k -sized sets.

A natural generalization of simple cardinality constraints is partition constraints [Cel+16]. Partition constraints arise naturally when there is an inherent labeling or grouping of the ground set items that is not captured by the DPP kernel itself. Concretely, suppose that the ground set $[n]$ is partitioned into disjoint sets $V_1 \cup V_2 \cup \dots \cup V_r$, and we want to produce a subset S with c_1 items from V_1 , c_2 items from V_2 and so on. We define Partition-DPP as the corresponding conditioning of the DPP under these constraints on S . Celis, Deshpande, Kathuria, Straszak, and Vishnoi [Cel+16] established that efficiently sampling and counting from Partition-DPPs is possible when the number of constraints r is $O(1)$ and that counting is #P-hard without such restrictions on r – it includes, as a special case, the problem of computing mixed discriminants. Here, we will only study Partition-DPPs where the ensemble matrix L is symmetric PSD and the number of constraints is $O(1)$. Alimohammadi, Anari, Shiragur, and Vuong [Ali+21] showed that these distributions are entropically independent and that local Markov chains can be used to sample (sequentially) from these Partition-DPPs.

Definition 7 (Partition-DPP). Given a symmetric $n \times n$ matrix $L \geq 0$ and a partitioning of $[n] = V_1 \cup V_2 \cup \dots \cup V_r$ into $r = O(1)$ partitions together with $c_1, \dots, c_r \in \mathbb{Z}_{\geq 0}$, the Partition-DPP is the distribution of the DPP defined by L restricted to sets S that have $|S \cap V_i| = c_i$ for all i .

In this work, we establish as corollaries of [Theorem 1](#), a roughly quadratic parallel speedup in sampling from all aforementioned distributions. A crucial part of our algorithm relies on the existence of highly parallel counting oracles for these models. For example, for unconstrained DPPs, the partition function can be written as $\sum_S \det(L_{S,S}) = \det(L + I)$, and this can be computed in NC

[Csa75]. For k -DPPs and Partition-DPPs, the partition function can be computed via polynomial interpolation [Cel+16], which is again highly parallelizable (by, e.g., solving linear systems of equations involving Vandermonde matrices). The entropic independence of all the aforementioned determinantal distributions discussed was established by prior works [Ali+21; Ana+21b].

Theorem 8 (Sampling from non-symmetric DPPs). *Let L be a $n \times n$ non-symmetric PSD matrix, $\epsilon > 0$, and $k \in [n]$.*

1. *Let $\mu_k : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$ be the k -DPP defined by L . For any constant $c > 0$, there exists an algorithm to approximately sample from within ϵ total variation distance of μ_k in $\tilde{O}\left(\sqrt{k}\left(\frac{k}{\epsilon}\right)^c\right)$ parallel time using $(n/\epsilon)^{O(1/c)}$ machines.*
2. *Let $\mu : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$ be the DPP defined by L . For any constant $c > 0$, there exists an algorithm to approximately sample from within ϵ total variation distance of μ in $\tilde{O}\left(\sqrt{n}\left(\frac{n}{\epsilon}\right)^c\right)$ parallel time using $(n/\epsilon)^{O(1/c)}$ machines.*

Theorem 9 (Sampling from Partition-DPPs). *Let L be a $n \times n$ symmetric PSD matrix. Let $r = O(1)$, and let $V_1 \cup \dots \cup V_r = [n]$ be a partition of $[n]$ together with integers t_1, \dots, t_r . Let $k = \sum_{i \in [r]} t_i$. Let $\mu_{L;V,t} : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$ be the DPP with partition constraints defined by*

$$\mu_{L;V,t}(S) \propto \det(L_{S,S}) \cdot \prod_{i=1}^r \mathbb{1}[|S \cap V_i| = t_i].$$

For any constant $c > 0$, there exists an algorithm to approximately sample from within ϵ total variation distance of $\mu_{L;V,t}$ in $\tilde{O}\left(\sqrt{k}\left(\frac{k}{\epsilon}\right)^c\right)$ parallel time using $(n/\epsilon)^{O(1/c)}$ machines.

In the case of symmetric DPPs and symmetric k -DPPs, we improve [Theorem 1](#) to obtain a parallel runtime of $\tilde{O}(\sqrt{k})$. Our algorithms have a small chance δ of failure but, conditioned on success, they sample exactly from the desired distribution; this is desirable, as we can repeat the algorithm in the case of failure, to eventually obtain an exact sample from the distribution. [Theorem 10](#) is proven in [Section 4](#).

Theorem 10 (Sampling from symmetric DPPs). *Let L be a $n \times n$ symmetric PSD matrix, $k \in [n]$, and $\delta \in (0, 1)$.*

1. *Let $\mu_k : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$ be the k -DPP defined by L . There exists an algorithm that with probability $\geq 1 - \delta$, exactly samples from μ_k in $\tilde{O}(\sqrt{k})$ parallel time using $\text{poly}(n) \cdot \log \frac{k}{\delta}$ machines.*
2. *Let $\mu : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$ be the DPP defined by L . There exists an algorithm, that with probability $\geq 1 - \delta$, exactly samples from μ in $\tilde{O}(\sqrt{n})$ parallel time using $\text{poly}(n) \cdot \log \frac{n}{\delta}$ machines.*

We are also able to refine our results about DPPs so that the runtime is expressed in terms of typical sizes of the sets S in the support, as measured by eigenvalues or traces of the matrix L . We leave the details to [Section 8](#), where we prove [Theorem 41](#).

Finally, another distribution whose partition function can be computed via determinants is uniform perfect matchings in planar graphs. These distributions, alongside determinantal point processes, were raised as challenge distributions for parallel sampling [Ana+20]. Unfortunately, planar perfect matchings are not entropically independent, and we cannot apply [Theorem 1](#) to them. Nevertheless, using alternative techniques based on graph separators, we manage to obtain a similar quadratic speedup in sampling from them.

Theorem 11. Let graph $G = (V, E)$ be planar and μ be the uniform distribution over perfect matchings of G . There exists an algorithm that samples from μ in $\tilde{O}(\sqrt{n})$ parallel time using $\text{poly}(n)$ machines.

1.2 Techniques and algorithms

Throughout, we heavily use the fact that for all distributions μ that we study in this paper, the marginals $\mathbb{P}_{S \sim \mu}[i \in S]$ can be computed in NC, and that the distributions μ are self-reducible—by conditioning on element inclusion, we obtain another distribution in the same family of DPP variants or planar perfect matchings on a smaller graph. These two properties alone form the basis of the most classic, inherently sequential, algorithm for sampling described below.

for $i = 1, \dots, k$ **do**

Compute the marginals of μ conditioned on elements x_1, \dots, x_{i-1} .
 Sample an element outside x_1, \dots, x_{i-1} with probability proportional to the computed marginals. Call the sampled element x_i .

return $\{x_1, \dots, x_k\}$.

Our main idea is to use rejection sampling to batch the iterations of this algorithm. Roughly speaking, we compute marginals of μ and sample a batch of elements x_1, \dots, x_ℓ i.i.d. from these marginals. We then use rejection sampling to accept or reject the batch to make sure any set $\{x_1, \dots, x_\ell\}$ is selected with probability given by the ℓ -order marginals $\propto \mathbb{P}_{S \sim \mu}[\{x_1, \dots, x_\ell\} \subseteq S]$. Once we have a batch of elements successfully accepted, we continue sampling the next batch from the distribution conditioned on including this batch. A high-level description of this algorithm can be seen in [Algorithm 1](#). Our innovation is to implement the batch sampling step highlighted via (*) by i.i.d. sampling from marginals and performing a correction based on rejection sampling.

Algorithm 1: Batched sampling

Input: $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$

$k_0 \leftarrow k$

$\mu^{(0)} \leftarrow \mu$

for $i = 0, 1, \dots, 2\sqrt{k}$ **do**

(*) Sample $T_i \sim \mu^{(i)}$ with $|T_i| = \lceil \sqrt{k_i} \rceil$
 Update $\mu^{(i+1)} \leftarrow \mu^{(i)}(\cdot \mid T_i)$
 Update $k_{i+1} = k_i - |T_i|$

return $T := \bigcup_i T_i$.

The sizes of the batches we sample dictate the parallel runtime of this algorithm. Even for symmetric DPPs, there is a natural barrier at batch size $\ell \simeq \sqrt{k}$. Consider L to be the Gram matrix of vectors $\{e_1, e_1, \dots, e_k, e_k\}$ (where every standard basis vector $e_i \in \mathbb{R}^n$ is repeated twice, so $k = \frac{n}{2}$). The marginals of this DPP are uniform, but because of the Birthday Paradox, any sample of $\gg \sqrt{k}$ elements contains a pair of identical vectors with high probability, resulting in a joint marginal of 0, i.e., an overwhelming probability of rejection. Hence, we must set the batch size $\ell \lesssim \sqrt{k}$ to have a non-negligible acceptance probability.

A significant difficulty that arises for DPP variants beyond symmetric DPPs is the lack of negative dependence. Roughly speaking, due to the lack of negative dependence, the acceptance probabilities used in symmetric DPPs for rejection sampling have to be scaled down in other cases by a factor

of $\approx 2^\ell$; otherwise, we would sometimes have to accept with probability > 1 . See [Section 7](#) for a detailed example of where this phenomenon can be observed. This scaling of 2^ℓ means the overall acceptance probability can be at most $2^{-\ell}$, which forces the number of machines we use to sample, in parallel, possible batches for one iteration to scale up by a factor of 2^ℓ . This limits ℓ to be only logarithmic, and the parallel runtime would not be improved beyond logarithmic factors.

We overcome this challenge by replacing rejection sampling with approximate rejection sampling, where we allow the acceptance probabilities to go above 1 on a small subset of the event space, and if we see any batch of this kind we declare the algorithm has failed. Our main insight is that such bad batches of elements must consist of large groups of highly correlated elements. On the other hand, we quantify limits on correlations in our distributions of interest by obtaining novel consequences of entropic independence [[Ana+21b](#)]. Intuitively, we prove that correlations in any entropically independent distribution must be limited to small groups of elements, and a batch of $\approx k^{\frac{1}{2}-c}$ elements will likely contain no more than one element from each correlated group. Formalizing this intuition, we prove [Theorems 8 and 9](#) in [Section 5](#). We give an example showing this sub-polynomial overhead in the parallel depth may be necessary for our batched rejection sampling approach in [Section 7](#).

Finally, for planar perfect matchings, we use a completely different approach. We use planar separators to break the graph into smaller pieces; the pieces can then be processed in parallel, once we fix the part of the perfect matching touching the separator.

1.3 Further related work

The works [[Ten95](#); [Ana+20](#)] study the problems of parallel sampling spanning trees and arborescences from graphs. While spanning trees are special cases of DPPs, these works parallelize an algorithm of Aldous [[Ald90](#)] and Broder [[Bro89](#)], which is very specific to spanning trees, and does not have a known counterpart for general DPPs or other distributions studied in this work.

There have been several approaches to parallel sampling in the literature. Notably, Feng, Hayes, and Yin [[FHY21](#)], and more recently Liu and Yin [[LY21](#)], showed how to efficiently parallelize a popular class of sampling algorithms based on Markov chains and obtain nearly optimal parallelism for several graphical models such as the hardcore, Ising, and proper coloring models in certain regimes. While these results manage to parallelize certain types of Markov chains, they do not apply to distributions studied in this work. A prerequisite for these parallelization techniques to work is the existence of a Markov chain with single-site updates, that is changing one coordinate in each move assuming the distribution is supported on a product space, that mixes in nearly-linear time. No such Markov chain is known for our application distributions.

Finally, subsequent to our results, it was recently communicated to us that [[Ana+23](#)] have obtained RNC sampling algorithms for distributions that satisfy a stronger condition than entropic independence and using a stronger oracle than the simple counting oracle: an oracle that can compute for any given $\lambda \in \mathbb{R}_{>0}^n$, the value $\sum_S \mu(S) \prod_{i \in S} \lambda_i$. While their general result is not comparable to our main result, [Theorem 1](#), it does as a corollary give an RNC sampling algorithm for Partition-DPPs. For nonsymmetric DPPs and planar perfect matchings, as far as we know, RNC sampling is still open.

Acknowledgment

This work was supported by an NSF CAREER Award CCF-2045354 and a Sloan Research Fellowship. Tuy-Duong Vuong was supported by a Microsoft Research Ph.D. Fellowship. Callum Burgess was supported by the Stanford CURIS program.

2 Overview

In this section, we give a technical outline of our proofs and a roadmap for the rest of the paper.

Symmetric DPPs. In [Section 4](#), we prove our most basic result, [Theorem 10](#) (sampling symmetric DPPs), as an introduction to our techniques and to demonstrate how we apply [Algorithm 1](#). Conveniently, symmetric DPPs exhibit strong negative dependence properties which the rest of our distributions do not. To prove [Theorem 10](#), we directly bound the acceptance probability of [Algorithm 1](#) for batch size $\ell \simeq \sqrt{k}$. We show that directly applying negative dependence bounds the acceptance probability by $\exp(-\frac{\ell^2}{k})$. In [Section 8](#) we obtain refined results for DPPs whose size is not constrained (i.e., not k -DPPs), but whose kernel matrix satisfies various forms of spectral boundedness, all related to the typical size of sampled sets.

Entropic independence. In [Section 5](#), we provide a meta-result ([Theorem 29](#), a formal restatement of [Theorem 1](#)) used to derive [Theorems 8](#) and [9](#) as corollaries. [Theorem 29](#) shows that for *any* entropically independent distribution over subsets of size k , we can reduce sampling to marginal computations with $\tilde{O}(k^{\frac{1}{2}+c})$ parallel depth overhead, with a high probability of success. Here, c is any constant, parameterizing the (polynomial) number of machines used.

To prove [Theorem 29](#), we use the entropic independence property to derive concentration bounds on the acceptance probability in [Algorithm 3](#). As a first step towards this goal, we use entropic independence to demonstrate that up to parallel depth $\ell \approx k^{\frac{1}{2}-c}$, the Kullback-Leibler (KL) divergence between our target distribution (the order- ℓ -marginals) and our proposal distribution (the product distribution on 1-marginals) is bounded. This KL divergence bound does not suffice for our overall scheme; intuitively, it provides an “average case” bound on the log-acceptance probability of [Algorithm 3](#), whereas we would like to have a high probability bound since we need to union bound over at least \sqrt{k} stages of rejection sampling. To simplify our concentration argument, we begin by assuming w.l.o.g. that our distribution has roughly uniform 1-marginals, by using a subdivision process, similar to the ones used in [[AD20](#); [Ana+21a](#)]. We then use comparison inequalities between KL divergences and (exponentiated) Renyi divergences for nearly-uniform distributions to bound moments associated with our rejection sampler’s acceptance probabilities. Finally, we use these moment bounds to show that over a high-probability set of outcomes (in the sense of [Algorithm 3](#)), the log of acceptance probability is a submartingale, yielding concentration via Markov’s inequality.

Hard instance. It is natural to ask: Can we improve the small additional overhead of $(k/\epsilon)^c$ in [Theorem 29](#) (and hence, [Theorems 8](#) and [9](#)) to a smaller overhead, e.g. polylogarithmic? In [Section 7](#), we give a hard example showing that the subpolynomial overhead may be inherent to rejection sampling strategies, at least in the full generality of entropically independent distributions.

Planar graph perfect matchings. In [Section 6](#), we consider the problem of parallel sampling of perfect matchings from a planar graph (whose counts can be written as a determinant [[Kas67](#)]). Sampling planar perfect matchings was raised as a challenge in [[Ana+20](#)]; while our approach in [Section 6](#) departs from [Theorem 29](#), it still results in a quadratic speedup over naïve sequential sampling. We attain this speedup, [Theorem 11](#), by leveraging parallel implementations of the planar separator

theorem. By sequentially sampling the portion of the matching adjacent to the vertices in a planar separator of size $O(\sqrt{n})$, and recursing on the (geometrically smaller) disconnected components, we obtain a roughly-quadratic speedup for this sampling problem as well.

3 Preliminaries

In this section, we provide preliminaries for the rest of the paper.

We use $[n]$ to denote the set $\{1, \dots, n\}$. For a set S , $\binom{S}{k}$ denotes the family of subsets of size k . For a distribution $\mu : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$ and $T \subseteq [n]$, define $\mu(\cdot | T)$ to be the distribution on $2^{[n] \setminus T}$ defined by $\mu(F | T) \propto \mu(F \cup T)$. We will sometimes use the shorthand $\mu|_T$.

For a measure or density function $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$, the generating polynomial of μ is the multivariate homogeneous polynomial defined as $g_\mu(z_1, \dots, z_n) = \sum_{S \in \binom{[n]}{k}} \mu(S) \prod_{i \in S} z_i$.

3.1 Divergences

Let q, p be distributions over the same finite ground set $[n]$. We define the KL divergence and, for $\lambda \geq 1$, the λ -divergence between q and p as follows:

$$\mathcal{D}_{\text{KL}}(q \parallel p) := \mathbb{E}_q \left[\log \frac{q}{p} \right] = \sum_{i \in [n]} q_i \log \left(\frac{q_i}{p_i} \right),$$

$$\mathcal{D}_\lambda(q \parallel p) := \mathbb{E}_p \left[\left(\frac{q}{p} \right)^\lambda \right] = \sum_{i \in [n]} q_i^\lambda p_i^{1-\lambda}.$$

We remark that our definition of \mathcal{D}_λ is (up to a constant scalar multiplication) the exponential of the standard Renyi divergence of order λ . The KL and Renyi divergences exhibit the following useful bound whose proof we include for completeness in [Section 9](#).

Lemma 12. *Let q, p be distributions over $[n]$. Suppose for some $C \geq 1$ and $S \subseteq [n]$: $p_i \leq \frac{C}{n}$ for all $i \in [n]$ and $p_i \geq \frac{1}{Cn}$ for all $i \in S$. Then, for any $\lambda \geq 1$, if $S = [n]$*

$$\mathcal{D}_\lambda(q \parallel p) \leq C^{\lambda-1} \left(1 + n^{\lambda-1} \lambda (\lambda - 1) (\mathcal{D}_{\text{KL}}(q \parallel p) + \log C) \right).$$

More generally,

$$\sum_{i \in S} q_i \left(\frac{q_i}{p_i} \right)^{\lambda-1} \leq C^{\lambda-1} \left(1 + n^{\lambda-1} \lambda (\lambda - 1) (\mathcal{D}_{\text{KL}}(q \parallel p) + \log C) \right).$$

3.2 Determinantal point processes

A DPP on n items defines a probability distribution over subsets $Y \subseteq [n]$. It is parameterized by a matrix $L \in \mathbb{R}^{n \times n}$: $\mathbb{P}_L[Y] \propto \det(L_Y)$, where L_Y is the principal submatrix whose columns and rows are indexed by Y . We call L the ensemble matrix. We define the marginal kernel K of \mathbb{P}_L by

$$K = L(I + L)^{-1} = I - (I + L)^{-1} = (L^{-1} + I)^{-1}. \quad (1)$$

Then, $\det(K_A) = \mathbb{P}_L[A \subseteq Y]$ (a proof can be found in [KT12a]). This also implies $K \leq I$ for symmetric K . Conversely,

$$L = K(I - K)^{-1} = (I - K)^{-1} - I = (K^{-1} - I)^{-1}. \quad (2)$$

Given a cardinality constraint k , the k -DPP parameterized by L is a distribution over subsets Y of size k , defined by $\mathbb{P}_L^k[Y] = \frac{\det(L_Y)}{\sum_{|Y'|=k} \det(L_{Y'})}$. To ensure that \mathbb{P}_L defines a probability distribution, all principal minors of L must be non-negative: $\det(L_S) \geq 0$. Any nonsymmetric (or symmetric) PSD matrix automatically has nonnegative principal minors [Gar+19, Lemma 1].

Consider a matrix $L \in \mathbb{R}^{n \times n}$, partition $V_1 \cup \dots \cup V_r = [n]$ of $[n]$, and tuple $\{c_i\}_{i=1}^r$ of integers. The DPP with partition constraint (Partition-DPP) $\mu_{L;V,c} : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$ is defined by

$$\mu_{L;V,c}(S) \propto \mathbb{1}[\forall i : |S \cap V_i| = c_i] \det(L_{S,S})$$

For any $Y \subseteq [n]$, if we condition the distribution \mathbb{P}_L (\mathbb{P}_L^k resp.) on the event that items in Y are included in the sample, we still get a DPP ($(k - |Y|)$ -DPP resp.); the new ensemble matrix is given by the Schur complement $L^Y = L_{\tilde{Y}} - L_{\tilde{Y},Y} L_{Y,Y}^{-1} L_{Y,\tilde{Y}}$ where $\tilde{Y} = [n] \setminus Y$.

For Partition-DPPs, a similar statement holds. Conditioning $\mu_{L;V,c}$ on Y being included in the set results in a Partition-DPP $\mu_{L^Y;V',c'}$ with ensemble matrix L^Y and partition $V'_1 \cup \dots \cup V'_r = [n] \setminus Y$ with $V'_i = V_i \setminus Y$, and $c'_i = c_i - |V_i \cap Y|$.

We defer the proof of the following computational facts about DPPs to [Section 9](#).

Proposition 13. *Suppose μ is one of the following distributions.*

1. k -DPP: $\mu(S) \propto \mathbb{1}[|Y| = k] \det(L_S)$.
2. DPP: $\mu(S) \propto \det(L_S)$.
3. Partition-DPP: $\mu_{V,c}(S) \propto \mathbb{1}[\forall i \in [r] : |S \cap V_i| = c_i] \det(L_S)$ with $r = O(1)$.

There are algorithms that perform the following tasks in $\tilde{O}(1)$ -parallel time using $\text{poly}(n)$ machines.

1. Given $S \subseteq T \subseteq [n]$, exactly compute $\mathbb{P}_{X \sim \mu}[T \subseteq X \mid S \subseteq X]$.
2. Given $S \subseteq [n]$ and $t \in [n]$, exactly compute $\mathbb{P}_{T \sim \mu}[|T| = t]$.

Well-known concentration inequalities on DPPs imply that the size of a typical set is tightly concentrated around its mean. We defer the proof of the following to [Section 9](#).

Lemma 14. *Let $\mu : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$ be a strongly Rayleigh distribution. Suppose $\mathbb{E}_{S \sim \mu}[|S|] \leq \sqrt{n}$. For $\epsilon \in (0, \frac{1}{4})$, there exists an absolute constant $c > 0$ such that*

$$\mathbb{P}_{S \sim \mu} \left[|S| \geq c \sqrt{n \log \frac{1}{\epsilon}} \right] \leq \epsilon$$

and

$$\mathbb{P}_{S \sim \mu} \left[|S| \geq c \mathbb{E}_{S \sim \mu}[|S|] \log \frac{1}{\epsilon} \right] \leq \epsilon$$

Remark 15. To sample from a DPP $\mu : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$, we can first in constant parallel-time compute the distribution \mathcal{H} on $[n]$ defined by $\mathbb{P}_{\mathcal{H}}[k] := \mathbb{P}_{S \sim \mu}[|S| = k]$ and sample the cardinality of the set k from \mathcal{H} , and then sample S from μ_k using the results of this paper.

Determinantal point processes (the symmetric kind) and their conditionings belong to a class of probability distributions called strongly Rayleigh [see [BBL09](#), for definition]. Strongly Rayleigh distributions satisfy a useful property called negative correlation.

Lemma 16 (Negative correlation). *Suppose $\mu : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$ is strongly Rayleigh. For any set T , $\mathbb{P}_{S \sim \mu}[T \subseteq S] \leq \prod_{i \in T} \mathbb{P}_{S \sim \mu}[i \in S]$.*

Lemma 17. *Let L be a symmetric PSD matrix. The following distributions are strongly Rayleigh.*

1. $\mu_k : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$, the k -DPP defined by L .
2. $\mu : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$, the DPP defined by L .

Corollary 18. *Let $K \in \mathbb{R}^{n \times n}$ satisfy $0 \leq K \leq I$. For any set $T \subseteq [n]$, $\det(K_T) \leq \prod_{i \in T} K_{i,i}$.*

Proof. Consider the DPP μ with kernel matrix K . Apply [Lemma 16](#) and note that $\det(K_T) = \mathbb{P}_{S \sim \mu}[T \subseteq S]$ and $K_{i,i} = \mathbb{P}_{S \sim \mu}[i \in S]$. \square

3.3 Entropic independence

We recall the notions of fractional log-concavity [[Ali+21](#)] and entropic independence [[Ana+21b](#)].

Definition 19 ([[Ali+21](#)]). A probability distribution $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$ is α -fractionally-log-concave if $\log g_\mu(z_1^\alpha, \dots, z_n^\alpha)$ is concave over $\mathbb{R}_{>0}^n$. For $\alpha = 1$, we say μ is log-concave.

To define entropic independence we need the definition of the “down” operator. In brief, $D_{k \rightarrow \ell}$ transitions from a set S of size k to a uniformly random subset of size ℓ .

Definition 20 (Down operator). For $\ell \leq k$ define the row-stochastic matrix $D_{k \rightarrow \ell} \in \mathbb{R}_{\geq 0}^{\binom{[n]}{k} \times \binom{[n]}{\ell}}$ by

$$D_{k \rightarrow \ell}(S, T) = \mathbb{1}[T \subseteq S] \cdot \frac{1}{\binom{k}{\ell}}.$$

Note that for a distribution μ on size- k sets, $\mu D_{k \rightarrow \ell}$ will be a distribution on size- ℓ sets. In particular, $\mu D_{k \rightarrow 1}$ will be the vector of normalized marginals of μ : $\{\frac{1}{k} \mathbb{P}[i \in S]\}_{i \in [n]}$. We will use the following shorthand:

Definition 21. For $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$, we use μ_ℓ to denote $\mu D_{k \rightarrow \ell}$.

Definition 22. A probability distribution μ on $\binom{[n]}{k}$ is said to be $\frac{1}{\alpha}$ -entropically independent, for $\alpha \in (0, 1]$, if for all probability distributions ν on $\binom{[n]}{k}$,

$$\mathcal{D}_{\text{KL}}(\nu_1 \parallel \mu_1) = \mathcal{D}_{\text{KL}}(\nu D_{k \rightarrow 1} \parallel \mu D_{k \rightarrow 1}) \leq \frac{1}{\alpha k} \mathcal{D}_{\text{KL}}(\nu \parallel \mu).$$

Lemma 23 ([[Ana+21b](#)], Theorem 4). *If μ is α -FLC then μ and all conditional distributions of μ , i.e. $\mu(\cdot \mid S)$ for any $S \subseteq [n]$, are $\frac{1}{\alpha}$ -entropically independent.*

Lemma 24 ([[Ali+21](#)]). *The following distributions are α -FLC for $\alpha = \Omega(1)$ and $k \in [n]$.*

1. k -DPP and DPP defined by nonsymmetric PSD $L \in \mathbb{R}^{n \times n}$.
2. Partition-DPP defined by symmetric PSD $L \in \mathbb{R}^{n \times n}$ and a partition $\{V_i\}_{i=1}^r$ with $r = O(1)$.

3.4 Rejection sampling

Consider distributions μ, ν over the same domain, and parameter C such that $\max\left\{\frac{\mu(x)}{\nu(x)}\right\} \leq C$. Assuming sample access to ν , we can also sample from μ via rejection sampling as in [Algorithm 2](#).

Algorithm 2: Rejection sampling

Input: parameter $C > 0$ such that $\max\left\{\frac{\mu(x)}{\nu(x)}\right\} \leq C$

Sample $x \sim \nu$.

Accept and output x with probability $\frac{\mu(x)}{C\nu(x)}$.

When [Algorithm 2](#) succeeds, its output distribution is exactly μ . [Algorithm 2](#) succeeds with probability

$$\mathbb{P}[\text{accept}] = \sum_x \frac{\mu(x)}{C\nu(x)}\nu(x) = \frac{1}{C}.$$

For any $\delta \in (0, 1)$, by running $C \log \delta^{-1}$ copies of the algorithm in parallel and taking the first accepted copy, we can boost the acceptance rate to $1 - \delta$, stated formally in the following.

Proposition 25. *There is an algorithm that with probability $1 - \delta$, outputs a sample from μ in the same asymptotic parallel time as required to sample from ν , using $O(C \text{poly}(n) \log \frac{1}{\delta})$ machines.*

We consider the following modification of [Algorithm 2](#) when we have a weaker assumption that for some $\Omega \subseteq \text{supp}(\nu)$ we have $\max\left\{\frac{\mu(x)}{\nu(x)} \mid x \in \Omega\right\} \leq C$ and $\sum_{x \in \Omega} \mu(x) \geq 1 - \epsilon$ for some $\epsilon \in [0, 1)$.

Algorithm 3: Modified rejection sampling

Input: $\Omega \subseteq \text{supp}(\nu)$, parameter $C > 0$ such that $\max\left\{\frac{\mu(x)}{\nu(x)} \mid x \in \Omega\right\} \leq C$

Sample $x \sim \nu$.

If $x \in \Omega$, accept and output x with probability $\frac{\mu(x)}{C\nu(x)}$.

The guarantees of [Algorithm 3](#) follow immediately from [Proposition 26](#) and that the restriction of μ to Ω has total variation distance at most ϵ from μ . In particular, we will use [Proposition 26](#) with $\delta = \epsilon$ and output an arbitrary sample when it fails to accept a sample.

Proposition 26. *There is an algorithm that outputs a sample from some $\tilde{\mu}$ in the same asymptotic parallel time as required to sample from ν , using $O(C \text{poly}(n) \log \frac{1}{\epsilon})$ machines, where $d_{\text{TV}}(\tilde{\mu}, \mu) = O(\epsilon)$.*

4 Symmetric DPP

Here, we prove our basic result, [Theorem 10](#). We provide a strengthening for DPPs satisfying nontrivial spectral bounds, [Theorem 41](#), in [Section 8](#). We first state helper bounds used in the proof.

Lemma 27. *Suppose μ on $\binom{[n]}{k}$ is negatively correlated. Let $\mu_t = \mu D_{k \rightarrow t}$ and $p_i = \mathbb{P}_\mu[i \in S]$. Then $\mu_t(T) / (t! \prod_{i \in T} \frac{p_i}{k}) \leq \exp\left(\frac{t^2}{k}\right)$.*

Proof. Note that $\mu_t(T) =$

$$\binom{k}{t}^{-1} \mathbb{P}_{S \sim \mu}[T \subseteq S] = \frac{t!}{k(k-1) \cdots (k-t+1)} \mathbb{P}_{S \sim \mu}[T \subseteq S].$$

Thus, by negative correlation,

$$\begin{aligned} \frac{\mu_t(T)}{t! \prod_{i \in T} \frac{p_i}{k}} &= \frac{k^t}{k(k-1) \cdots (k-t+1)} \frac{\mathbb{P}_{S \sim \mu}[T \subseteq S]}{\prod_{i \in T} \mathbb{P}_{S \sim \mu}[i \in S]} \\ &\leq \left(\prod_{i=1}^{t-1} \left(1 - \frac{i}{k}\right) \right)^{-1} \leq \exp\left(\frac{t^2}{k}\right) \end{aligned}$$

where we used the facts that $1 - x \geq e^{-2x}$ and $\prod_{i=1}^{t-1} \exp\left(\frac{2i}{k}\right) = \exp\left(\frac{t^2-t}{k}\right)$. \square

Proposition 28. *If step (*) takes $O(\tau)$ -parallel time, Algorithm 1 takes $O(\sqrt{k} \cdot \tau)$ -parallel time.*

Proof. Note that

$$k_{i+1} = k_i - \lceil \sqrt{k_i} \rceil \leq k_i - \sqrt{k_i} \leq \left(\sqrt{k_i} - \frac{1}{2}\right)^2.$$

and thus $\sqrt{k_{i+1}} \leq \sqrt{k_i} - \frac{1}{2}$. Hence, since $t \geq 2\sqrt{k}$ implies $\sqrt{k_t} \leq \sqrt{k_0} - \frac{t}{2} \leq 0$, the algorithm terminates in $O(\sqrt{k})$ iterations, and takes $O(\sqrt{k})$ parallel time. \square

Proof of Theorem 10. We first consider the case of sampling k -DPPs. By Lemma 17, when μ is a k -DPP defined by a symmetric PSD ensemble matrix L , μ and the conditionals of μ are real-stable. In particular, all $\mu^{(i)}$ as defined in Algorithm 1 are real-stable and hence strongly Rayleigh.

Consider some loop i , and let $\mu \equiv \mu^{(i)}$. We use Lemma 27 to implement step (*). In particular, we first compute the marginals $p_i = \mathbb{P}_\mu[i \in S]$ in $\tilde{O}(1)$ parallel time. Next, let ν be the distribution over ordered tuples $(i_1, \dots, i_t) \in [n]^t$ with

$$\nu(\{i_1, \dots, i_t\}) = \prod_{r=1}^t \frac{p_{i_r}}{k}.$$

We can identify μ_t with the distribution μ_t^* over $[n]^t$ where we denote $\mu_t^*(\{i_1, \dots, i_t\}) = \frac{\mu(\{i_1, \dots, i_t\})}{t!}$. Let $\delta' = \frac{\delta}{2\sqrt{k}}$, where Algorithm 1 takes at most $2\sqrt{k}$ iterations by Proposition 28. We run the rejection sampling algorithm in Proposition 25, which succeeds with probability $1 - \delta'$, to sample from μ_t^* given samples from ν , with $C \leq \exp\left(\frac{t^2}{k}\right) = O(1)$ since $t = \lceil \sqrt{k} \rceil$. Clearly obtaining a sample from μ_t^* yields a sample from μ_t by forgetting the ordering on the elements.

Hence, each iteration i of Algorithm 1 takes $\tilde{O}(1)$ parallel time. By Proposition 28, the algorithm takes $O(\sqrt{k})$ -parallel time. By a union bound, the success probability is $\geq 1 - 2\sqrt{k}\delta' = 1 - \delta$. The number of machines used is as in Proposition 25, $O(\text{poly}(n) \log \frac{k}{\delta})$.

The result for DPPs immediately follows by first sampling the size $|S|$ of $S \sim \mu$ and then sampling from the appropriate k -DPP. \square

5 Entropically independent distributions

In this section, we prove the main result. We will use [Theorem 29](#) to derive our samplers for various entropically independent distributions, namely [Theorems 8](#) and [9](#), which immediately follow from combining [Lemmas 23](#) and [24](#) and [Theorem 29](#).

Theorem 29. *Let $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$ be such that all its conditional distributions are $\frac{1}{\alpha}$ -entropically independent with $\alpha = \Omega(1)$. Suppose we can compute marginals $\mathbb{P}_\mu[i \in S]$ for $S \subseteq [n]$ and $i \notin S$ in $\tilde{O}(1)$ parallel time. For any constant $c > 0$ and any $\epsilon \in (0, 1)$, there exists an algorithm to sample from a distribution within total variation distance ϵ of μ in $\tilde{O}\left(\sqrt{k} \cdot \left(\frac{k}{\epsilon}\right)^c\right)$ parallel time using $\left(\frac{n}{\epsilon}\right)^{O(c-1)}$ machines.*

5.1 Isotropic transformation

We first reduce to the case of near-isotropic distributions. Similarly to [\[AD20; Ana+21a\]](#), we say a distribution $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$ is isotropic if for all $i \in [n]$, the marginal $\mathbb{P}_{S \sim \mu}[i \in S]$ is $\frac{k}{n}$. Prior work [\[AD20\]](#) introduced the following subdivision process transforming an arbitrary $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$ to a nearly-isotropic $\mu' : \binom{[U]}{k} \rightarrow \mathbb{R}_{\geq 0}$, while preserving entropic independence.

Definition 30. Let $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$ be an arbitrary probability distribution and assume that we have access to the marginals p_1, \dots, p_n of the distribution with $p_1 + \dots + p_n = k$ and $p_i = \mathbb{P}_{S \sim \mu}[i \in S]$ for all i . For a parameter $\beta \in (0, 1)$, let $t_i := \lceil \frac{n}{\beta k} p_i \rceil$. We create a new distribution out of μ as follows: for each $i \in [n]$, create t_i copies of element i and let the collection of all copies be the new ground set: $U := \bigcup_{i=1}^n \{i^{(j)}\}_{j \in [t_i]}$. Define the following distribution $\mu^{\text{iso}} : \binom{[U]}{k} \rightarrow \mathbb{R}_{\geq 0}$:

$$\mu^{\text{iso}}\left(\left\{i_1^{(j_1)}, \dots, i_k^{(j_k)}\right\}\right) := \frac{\mu(\{i_1, \dots, i_k\})}{t_1 \cdots t_k}.$$

We call μ^{iso} the *isotropic transformation* of μ .

Another way we can think of μ^{iso} is that to produce a sample from it, we can first generate a sample $\{i_1, \dots, i_k\}$ from μ , and then choose a copy $i_m^{(j_m)}$ for each element i_m in the sample, uniformly at random. Subdivision preserves entropic independence.

Proposition 31 ([\[Ana+21a, Proposition 19\]](#)). *If the distribution μ is $\frac{1}{\alpha}$ -entropically-independent, then so is μ^{iso} .*

The following useful properties of μ^{iso} generalize [\[Ana+21a, Proposition 24\]](#). We defer the proof to [Section 9](#).

Proposition 32. *Let $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$, and let $\mu^{\text{iso}} : \binom{[U]}{k} \rightarrow \mathbb{R}_{\geq 0}$ be the subdivided distribution from [Definition 30](#) for some β . Let $C = 1 + \sqrt{\beta}$. The following hold for μ^{iso} .*

1. *Marginal upper bound: For all $i^{(j)} \in U$, the marginal $\mathbb{P}_{S \sim \mu^{\text{iso}}}[i^{(j)} \in S] \leq C \frac{k}{|U|}$.*
2. *Marginal lower bound: If $p_i := \mathbb{P}_{S \sim \mu}[i \in S] \geq \frac{\sqrt{\beta k}}{n}$, then for all $j \in [t_i]$, $\mathbb{P}_{S \sim \mu^{\text{iso}}}[i^{(j)} \in S] \geq \frac{k}{C|U|}$.*
Furthermore, letting $R := \left\{i^{(j)} \mid p_i \geq \frac{\sqrt{\beta k}}{n}, j \in [t_i]\right\}$ then for any $\ell \leq k$

$$\sum_{S \in \binom{[U]}{\ell}} \mu_S^{\text{iso}} \geq 1 - \sqrt{\beta} \ell.$$

3. Bounded ground set size: $n\beta^{-1} \leq |U| \leq n(1 + \beta^{-1})$.

Remark 33. For any $\ell \in [k]$, suppose algorithm \mathcal{A} can sample from within total variation distance ϵ of μ_ℓ^{iso} . Then \mathcal{A} can also be used to sample from within total variation distance ϵ of μ_ℓ using the same amount of (parallel) time and machines.

5.2 KL divergence bound

Throughout this section and [Section 5.3](#), let $\ell \in [k]$. We begin by proving a bound on the KL divergence between conditional marginals from an observed set. We denote by p the vector of marginals, that is $p_i = \mathbb{P}_{S \sim \mu}[i \in S]$.

Lemma 34. *Let $S \in \binom{[n]}{t}$ for $t \leq \frac{1}{2}k$. Let $\mu_{t+1|S} : [n] \setminus S \rightarrow \mathbb{R}_{\geq 0}$ be the marginal distribution of elements in $S' \sim \mu_{t+1}$ conditioned on $S \subset S'$, namely $\mu_{t+1|S} := \mu(\cdot | S)D_{(k-t) \rightarrow 1}$. Then,*

$$\mathcal{D}_{\text{KL}}\left(\mu_{t+1|S} \parallel \frac{1}{k}p\right) \leq \frac{2}{\alpha k} \log\left(\frac{1}{\mathbb{P}_{T \sim \mu}[S \subset T]}\right) + \frac{2t}{k}. \quad (3)$$

This bound follows from the entropic independence of μ . We defer the proof to [Section 9](#).

By averaging [Lemma 34](#) over μ_S (the conditional distribution of $T \sim \mu$ on $S \subset T$), we immediately obtain the following corollary.

Corollary 35. *Let $t \leq \frac{1}{2}k$. Then following the notation of [Lemma 34](#),*

$$\sum_{S \in \binom{[n]}{t}} \mu D_{k \rightarrow t}(S) \mathcal{D}_{\text{KL}}\left(\mu_{t+1|S} \parallel \frac{1}{k}p\right) \leq \frac{2t}{k} \left(\frac{1}{\alpha} \log\left(\frac{2n}{k}\right) + 1\right).$$

Proof. It suffices to apply [Lemma 34](#), and the calculation

$$\begin{aligned} & \sum_{S \in \binom{[n]}{t}} \mu D_{k \rightarrow t}(S) \log\left(\frac{1}{\mathbb{P}_{T \sim \mu}[S \subset T]}\right) \\ &= \sum_{S \in \binom{[n]}{t}} \mu D_{k \rightarrow t}(S) \log\left(\frac{1}{\mu D_{k \rightarrow t}(S) \binom{k}{t}}\right) \\ &= \sum_{S \in \binom{[n]}{t}} \mu D_{k \rightarrow t}(S) \log\left(\frac{1}{\mu D_{k \rightarrow t}(S)}\right) + \log \frac{1}{\binom{k}{t}} \\ & \leq \log \frac{\binom{n}{t}}{\binom{k}{t}} \leq t \log\left(\frac{2n}{k}\right). \end{aligned}$$

The first equality used $\mathbb{P}_{T \sim \mu}[S \subset T] = \mu D_{k \rightarrow t}(S) \binom{k}{t}$, and the last line used that the negative entropy of a distribution supported on N elements is bounded by $\log N$. \square

Finally, we use [Corollary 35](#) to derive a KL divergence bound between the distributions μ_ℓ and μ'_ℓ , respectively the target and proposal distributions encountered in our rejection sampling scheme.

Lemma 36. Let μ'_j be the distribution of the set formed by j independent draws from $\frac{1}{k}p$. Let $\ell \leq \frac{1}{2}k$. Then,

$$\mathcal{D}_{\text{KL}}(\mu_\ell \parallel \mu'_\ell) \leq \frac{\ell^2}{k} \left(\frac{1}{\alpha} \log\left(\frac{2n}{k}\right) + 1 \right).$$

Proof. For any $j \in [\ell]$, following the notation of [Lemma 34](#),

$$\begin{aligned} \mathcal{D}_{\text{KL}}(\mu_j \parallel \mu'_j) - \mathcal{D}_{\text{KL}}(\mu_{j-1} \parallel \mu'_{j-1}) &= \\ \sum_{S \in \binom{[n]}{j-1}} \mu D_{k \rightarrow (j-1)}(S) \mathcal{D}_{\text{KL}}\left(\mu_{j|S} \parallel \frac{1}{k}p\right) &\leq \\ \frac{2(j-1)}{k} \left(\frac{1}{\alpha} \log\left(\frac{2n}{k}\right) + 1 \right). \end{aligned}$$

In the first line, we used the chain rule of KL divergence, and in the second line used [Corollary 35](#). Finally, the conclusion follows by telescoping the above display for $1 \leq j \leq \ell$. \square

[Lemma 36](#) bounds the KL divergence between μ_ℓ and μ'_ℓ , which can be thought of as an average log of acceptance probability for our rejection sampling scheme. For constant $\frac{1}{\alpha}$, this bound suggests that we can take $\ell \approx \sqrt{k}$ and obtain an efficient sampler for ℓ -marginals; however, it is only an average bound, whereas we need a high-probability bound; this is because we have multiple steps of rejection sampling and we cannot afford to fail in one step with constant probability. We make this rigorous in [Section 5.3](#), where we use the tools from this section to give concentration bounds on the acceptance probability of rejection sampling.

5.3 Concentration of acceptance probability

In this section, assume that we have already performed the transformation in [Proposition 32](#) parameterized by some β , and obtained a distribution $\nu^{\text{iso}} : \binom{U}{k} \rightarrow \mathbb{R}_{\geq 0}$ and a set $R \subseteq U$ of elements with lower bounds on marginals as given by [Proposition 32](#). Let $\nu := \nu^{\text{iso}}$ and let ν' be defined analogously to [Section 5.2](#). Our goal is to sample from within ϵ total variation of ν_ℓ for a suitably chosen ℓ , which also implies that we can sample from within ϵ of μ_ℓ (see [Remark 33](#)). Our algorithm will be the modified rejection sampler ([Algorithm 3](#)).

To use [Algorithm 3](#) with $P = \nu D_{k \rightarrow \ell}$ and Q the ℓ -wise product distribution drawing from $\frac{1}{k}p$, we first define a relevant high-probability set Ω on our state space $\mathcal{X} := \binom{U}{k}$. Our set Ω will be a subset of the following set, for some $\epsilon > 0$ we will choose later:

$$\tilde{\Omega}_\epsilon := \left\{ S \in \binom{U}{\ell} \mid \nu_{|T|}(T) \geq \epsilon^{|T|}, \forall T \subseteq S \right\}.$$

In other words, $\tilde{\Omega}_\epsilon$ contains all sets S such that all subsets $T \subset S$ are relatively well-represented according to $\nu_{|T|}(T)$. We begin with an observation lower bounding the measure of $\tilde{\Omega}_\epsilon$.

Lemma 37. For any $0 \leq \epsilon \leq \frac{1}{2|U|^\ell}$, we have $\sum_{S \in \tilde{\Omega}_\epsilon} \nu_\ell(S) \leq 2|U|\ell\epsilon$.

Proof. Let $C := \bigcup_{t=1}^\ell \{T \in \binom{U}{t} \mid \nu_t(T) \leq \epsilon^t\}$. For any $S \notin \tilde{\Omega}_\epsilon$, we say $T \in C$ is a ‘‘certificate’’ of S if $T \subset S$; every $S \in \tilde{\Omega}_\epsilon^c$ has at least one certificate, so there is a map $\mathcal{M} : \tilde{\Omega}_\epsilon^c \rightarrow C$. Moreover, for some

$T \in \mathcal{C}$, let $\widetilde{\Omega}_\varepsilon^c(T)$ be the set of all $S \in \widetilde{\Omega}_\varepsilon^c$ such that $\mathcal{M}(S) = T$. Then since

$$v_t(T) = \sum_{S \supseteq T} \frac{1}{\binom{\ell}{t}} v_\ell(S) \implies \sum_{S \in \widetilde{\Omega}_\varepsilon^c(T)} v_\ell(S) \leq \binom{\ell}{t} v_t(T),$$

summing over all $T \in \mathcal{C}$ yields

$$\begin{aligned} \sum_{S \in \widetilde{\Omega}_\varepsilon^c} v_\ell(S) &\leq \sum_{T \in \mathcal{C}} \binom{\ell}{t} v_{|T|}(T) = \sum_{1 \leq t \leq \ell} \sum_{\substack{T \in \binom{U}{t} \\ v_t(T) \leq \varepsilon^t}} \binom{\ell}{t} v_t(T) \\ &\leq \sum_{1 \leq t \leq \ell} (|U| \ell \varepsilon)^t \leq \frac{|U| \ell \varepsilon}{1 - |U| \ell \varepsilon} \leq 2|U| \ell \varepsilon. \end{aligned}$$

The last line used the approximations $\binom{\ell}{t} \leq \ell^t$, $\binom{|U|}{t} \leq |U|^t$. \square

For the remainder of the section we will specifically use $\varepsilon = \frac{\varepsilon}{32|U|\ell}$, such that $\widetilde{\Omega}_\varepsilon$ captures at least a $1 - \frac{\varepsilon}{16}$ fraction of the mass of $\binom{U}{\ell}$ according to v_ℓ . We will also drop ε from $\widetilde{\Omega}_\varepsilon$ for simplicity.

Our next goal is to show that almost all of the sets in $\widetilde{\Omega}$ have a polynomially bounded acceptance probability when the proposal is given by independent draws from $\frac{1}{k}p$. Consider iteratively building a set S_t for all $1 \leq t \leq \ell$, where S_t is a random variable formed by $S_{t-1} \cup \{i_t\}$ for $i_t \sim \frac{1}{k}p$. In particular, we use i_t to denote the t^{th} draw from $\frac{1}{k}p$ in this process. For parameters $\tau, \gamma \geq 0$ to be defined later, iteratively define the random variables:

$$\begin{aligned} Y_{t+1} &:= Y_t \exp(\Delta_{t+1}), \\ \Delta_{t+1} &:= \begin{cases} \gamma \log\left(\frac{v_{t+1|S_t}(i_{t+1})}{\frac{1}{k}p_{i_{t+1}}}\right) - \tau & v_t(S_t) \geq \varepsilon^t \text{ and } i_{t+1} \in R \\ -\infty & \text{otherwise} \end{cases} \end{aligned}$$

with $C = 1 + \sqrt{\beta}$ and R as defined in [Proposition 32](#). Also by [Proposition 32](#), $p_i \leq \frac{Ck}{|U|}$ for all $i \in U$. We use the convention $\exp(-\infty) = 0$.

We next prove that Y_{t+1} is a submartingale for appropriate parameter choices.

Lemma 38. *Let $S_t = T$ have $v_t(T) \geq \varepsilon^t$. Assume that $\sqrt{\beta} \leq \min\left\{\frac{1}{3\gamma}, \frac{t}{\alpha k} \log \frac{1}{\varepsilon}\right\}$. Then,*

$$\tau \geq |U|^\gamma \gamma (1 + \gamma) \cdot \left(\frac{12t}{\alpha k} \log \frac{1}{\varepsilon}\right) \implies \mathbb{E}_{i \sim v_{t+1|S_t}} [Y_{t+1} \mid S_t = T] \leq Y_t.$$

Proof. If $Y_t = 0$ then $Y_{t+1} = 0$ by definition. In the following, assume $Y_t > 0$. By the definition of

$Y_{t+1} = Y_t \exp(\Delta_{t+1})$, and since $\nu_t(T) \geq \varepsilon^t$, we have

$$\begin{aligned}
& \mathbb{E}_{i \sim \nu_{t+1|T}} \left[\frac{Y_{t+1}}{Y_t} \mid S_t = T \right] \\
&= \exp(-\tau) \mathbb{E}_{i \sim \nu_{t+1|T}} \left[\mathbb{1}_{i \in R} \left(\frac{\nu_{t+1|T}(i)}{\frac{1}{k} p_i} \right)^\gamma \mid S_t = T \right] \\
&= \exp(-\tau) \sum_{i \in R} \nu_{t+1|T}(i) \left(\frac{\nu_{t+1|T}(i)}{\frac{1}{k} p_i} \right)^\gamma \\
&\leq \exp(-\tau) C^\gamma \left(1 + |U|^\gamma \gamma (1 + \gamma) \left(\mathcal{D}_{\text{KL}} \left(\nu_{t+1|T} \parallel \frac{1}{k} p \right) + \log C \right) \right) \\
&\leq \exp(-\tau) (1 + 2\sqrt{\beta}\gamma) \left(1 + |U|^\gamma \gamma (1 + \gamma) \cdot \left(\frac{4t}{\alpha k} \log \frac{1}{\varepsilon} + \sqrt{\beta} \right) \right).
\end{aligned}$$

The second-to-last inequality uses [Lemma 12](#) with $C = 1 + \sqrt{\beta}$, and the last inequality used [Lemma 34](#), which shows that since $\nu_t(T) \geq \varepsilon^t$,

$$\begin{aligned}
\mathcal{D}_{\text{KL}} \left(\nu_{t+1|T} \parallel \frac{1}{k} p \right) &\leq \frac{2}{\alpha k} \log \left(\frac{1}{\mathbb{P}_{S \sim \nu} [T \subset S]} \right) + \frac{2t}{k} \\
&= \frac{2}{\alpha k} \log \left(\frac{1}{\nu_t(T) \binom{k}{t}} \right) + \frac{2t}{k} \leq \frac{4t}{\alpha k} \log \frac{1}{\varepsilon},
\end{aligned}$$

as well as $\log(1 + \sqrt{\beta}) \leq \sqrt{\beta}$ and

$$(1 + x)^\gamma \leq e^{x\gamma} \leq 1 + 2x\gamma$$

for $x\gamma := \sqrt{\beta}\gamma \leq \frac{1}{3}$. The conclusion follows from

$$\begin{aligned}
& (1 + 2\sqrt{\beta}\gamma) \left(1 + |U|^\gamma \gamma (1 + \gamma) \cdot \left(\frac{4t}{\alpha k} \log \frac{1}{\varepsilon} + \sqrt{\beta} \right) \right) \\
&\leq 1 + 2\sqrt{\beta}\gamma + |U|^\gamma \gamma (1 + \gamma) \cdot \left(\frac{5t}{\alpha k} \log \frac{1}{\varepsilon} \right) (1 + 2\sqrt{\beta}\gamma) \\
&\leq 1 + \gamma \left(\frac{2t}{\alpha k} \log \frac{1}{\varepsilon} \right) + |U|^\gamma \gamma (1 + \gamma) \cdot \left(\frac{5t}{\alpha k} \log \frac{1}{\varepsilon} \right) \left(1 + \frac{2}{3} \right) \\
&\leq 1 + |U|^\gamma \gamma (1 + \gamma) \cdot \left(\frac{12t}{\alpha k} \log \frac{1}{\varepsilon} \right) \leq 1 + \tau \leq \exp(\tau). \quad \square
\end{aligned}$$

Now, applying [Lemma 38](#) with the definition of $\tilde{\Omega}' := \tilde{\Omega} \cap \{S \in \binom{[k]}{\ell}\}$ allows us to obtain a high-probability bound on the acceptance probability of our rejection sampling scheme.

Lemma 39. *Let $B \geq 1$. For sufficiently small $\varepsilon \in (0, 1)$, and $\ell \in [k]$ satisfying $12\ell^2 \left(\frac{16}{\varepsilon}\right)^{\frac{3}{B}} \log \frac{1}{\varepsilon} \leq \alpha k$ and assuming our choices of parameters satisfy $\sqrt{\beta} \leq \min\left\{\frac{1}{3\gamma}, \frac{1}{\alpha k} \log \frac{1}{\varepsilon}\right\}$, we have*

$$\mathbb{P}_{S_\ell \sim \nu_\ell} \left[\frac{\nu_\ell(S_\ell)}{\nu'_\ell(S_\ell)} \geq |U|^B \mid S_\ell \in \tilde{\Omega}' \right] \leq \frac{\varepsilon}{8}.$$

Proof. Throughout this proof, we will assume

$$\gamma = \frac{2 \log \frac{16}{\epsilon}}{B \log |U|}, \quad \tau = \frac{\log \frac{16}{\epsilon}}{\ell}.$$

We first observe that our parameter choices indeed satisfy the condition on τ used in [Lemma 38](#):

$$\frac{\gamma(1+\gamma)|U|^\gamma}{\log(\frac{16}{\epsilon})} \leq \left(\frac{16}{\epsilon}\right)^{\frac{3}{B}} \implies \gamma(1+\gamma)|U|^\gamma \cdot \left(\frac{12\ell}{\alpha k} \log \frac{1}{\epsilon}\right) \cdot \frac{\ell}{\log \frac{16}{\epsilon}} \leq 1.$$

In the following we denote $\hat{\mu}_j$ to be the joint distribution of $\{i_1, i_2, \dots, i_j\}$ where $i_1 \sim \nu_1, i_2 \sim \nu_2|_{S_1=\{i_1\}}$, and so on. In other words, if S_ℓ is the unordered set of $\{i_1, i_2, \dots, i_\ell\}$, we have $\nu_\ell(S_\ell) = \ell! \cdot \hat{\mu}(\{i_1, i_2, \dots, i_\ell\})$. We similarly define $\hat{\mu}'_j$ so that $\nu'_\ell(S_\ell) = \ell! \cdot \hat{\mu}'_\ell(\{i_1, i_2, \dots, i_\ell\})$. For $S_\ell \in \binom{U}{\ell}$, and some realization $\{i_1, i_2, \dots, i_\ell\}$ whose unordered set is S_ℓ , cancelling a factor of $\ell!$ yields

$$L(S_\ell) := \frac{\nu_\ell(S_\ell)}{\nu'_\ell(S_\ell)} = \frac{\hat{\mu}_\ell(\{i_1, i_2, \dots, i_\ell\})}{\hat{\mu}'_\ell(\{i_1, i_2, \dots, i_\ell\})} = \prod_{j \in \ell} \frac{\nu_j|_{S_{j-1}=\{i_1, i_2, \dots, i_{j-1}\}}(\{i_j\})}{\frac{1}{k} p_{i_j}} \quad (4)$$

where ν'_ℓ is the distribution of the unordered set corresponding to ℓ draws from $\frac{1}{k}p$. Next, we apply [Lemma 38](#) which yields a submartingale property on Y_ℓ . Letting $\mathbb{1}_{S_\ell \in \tilde{\Omega}'}$ be the 0-1 valued indicator function of the event $S_\ell \in \tilde{\Omega}'$, we compute

$$\begin{aligned} 1 = Y_0 &\geq \mathbb{E}_{\{i_1, i_2, \dots, i_\ell\} \sim \hat{\mu}_\ell} \left[Y_\ell \cdot \mathbb{1}_{S_\ell \in \tilde{\Omega}'} \right] \\ &= \mathbb{P}_{S_\ell \sim \nu_\ell} \left[S_\ell \in \tilde{\Omega}' \right] \times \\ &\quad \mathbb{E}_{\{i_1, i_2, \dots, i_\ell\} \sim \hat{\mu}_\ell} \left[\exp(-\ell\tau + \gamma \log L(S_\ell)) \mid S_\ell \in \tilde{\Omega}' \right]. \end{aligned}$$

In the last two expressions, S_ℓ denotes the set $\{i_1, i_2, \dots, i_\ell\}$. The first inequality used the fact that Y_ℓ is always nonnegative, and whenever $S_\ell \in \tilde{\Omega}'$ we can apply [Lemma 38](#) to all subsets in the stages of its construction. The second line follows since whenever $S_\ell \in \tilde{\Omega}'$, we are always in the first case in the definition of Δ_{t+1} , and then we can apply (4). Hence, for any $B \geq 0$,

$$\begin{aligned} \mathbb{P}_{S_\ell \sim \nu_\ell} \left[S_\ell \in \tilde{\Omega}' \right] \cdot \mathbb{E}_{\{i_1, i_2, \dots, i_\ell\} \sim \hat{\mu}_\ell} \left[\exp(\gamma \log L(S_\ell)) \mid S_\ell \in \tilde{\Omega}' \right] \\ \leq \exp(\ell\tau) \end{aligned}$$

which implies

$$\begin{aligned} \mathbb{P}_{\{i_1, i_2, \dots, i_\ell\} \sim \hat{\mu}_\ell} \left[\log L(S_\ell) \geq B \log |U| \mid S_\ell \in \tilde{\Omega}' \right] \\ \leq 2 \exp(\ell\tau - \gamma B \log |U|), \end{aligned}$$

where we used Markov's inequality, and that $\tilde{\Omega}'$ captures at least half the mass of ν_ℓ . However, every permutation giving rise to the unordered set S_ℓ is equally likely under $\hat{\mu}_\ell$, so by aggregating permutations, this can be rewritten as the desired

$$\mathbb{P}_{S_\ell \sim \nu_\ell} \left[L(S_\ell) \geq |U|^B \mid S_\ell \in \tilde{\Omega}' \right] \leq 2 \exp(\ell\tau - \gamma B \log |U|) = \frac{\epsilon}{8}. \quad \square$$

Finally, we combine [Lemmas 37](#) and [39](#) to prove our main result.

Lemma 40. *Let $B \geq 1$, and for a sufficiently small constant below, suppose $\ell^2 = O\left(\frac{\alpha k}{\log \frac{n}{\epsilon}} \cdot \epsilon^{\frac{3}{B}}\right)$. There is a parallel algorithm using $O((nk^2\epsilon^{-2})^B \log \frac{1}{\epsilon})$ machines which runs in $O(1)$ time and returns a draw from a distribution within total variation distance $\frac{\epsilon}{2}$ of $\mu D_{k \rightarrow \ell}$.*

Proof. Without loss of generality, we can assume n is at least a sufficiently large constant, else the standard sequential sampler has parallel depth $\tilde{O}(1)$. We set

$$\sqrt{\beta} := \frac{\epsilon}{32k} \leq \min\left\{\frac{1}{\alpha k} \log \frac{1}{\epsilon}, \frac{1}{3\gamma}\right\} = \min\left\{\frac{1}{\alpha k} \log \frac{1}{\epsilon}, \frac{B \log n}{6 \log \frac{16}{\epsilon}}\right\},$$

which clearly satisfies the assumption of [Lemma 38](#) for sufficiently large n . Set $\epsilon = \frac{\epsilon}{32|U|\ell}$. Combining [Proposition 32](#) and [Lemma 37](#) and using a union bound, we have

$$\begin{aligned} v_\ell(\tilde{\Omega}') &\geq 1 - (1 - v_\ell(\tilde{\Omega})) - \left(1 - v_\ell\left(\left\{S \in \binom{R}{\ell}\right\}\right)\right) \\ &\geq 1 - 2\sqrt{\beta}k - 2|U|\ell\epsilon \geq 1 - \frac{\epsilon}{8}. \end{aligned}$$

By [Proposition 32](#), $|U| \leq 2n\beta^{-1} = O(nk^2\epsilon^{-2})$ and

$$\log \frac{1}{\epsilon} = O\left(\log \frac{|U|\ell}{\epsilon}\right) = O\left(\log \frac{n}{\epsilon}\right)$$

Thus, this setting of ℓ and β satisfies the assumption of [Lemma 39](#). Hence, the subset $\Omega \subset \tilde{\Omega}'$ which satisfies the conclusion of [Lemma 39](#) has measure at least $1 - \frac{\epsilon}{4}$ according to v_ℓ . Using [Algorithm 3](#) and [Remark 33](#), we can sample from within total variation $\frac{\epsilon}{2}$ from $\nu D_{k \rightarrow \ell}$ and $\mu D_{k \rightarrow \ell}$ in $\tilde{O}(1)$ -time using $O(|U|^B \log \frac{1}{\epsilon})$, which equals $O((nk^2\epsilon^{-2})^B \log \frac{1}{\epsilon})$ machines. We note that to implement our modified rejection sampling, it suffices to check that the likelihood ratio is bounded, which will certainly be the case for all elements in $\tilde{\Omega}'$, and if there are other sets with bounded likelihood ratio this only improves the total variation distance guarantee. \square

5.4 Proof of [Theorem 29](#)

Now we combine everything to prove [Theorem 29](#).

Proof of [Theorem 29](#). Since we can always sample in $\tilde{O}(k)$ parallel time, the statement is nontrivial only for $c \leq \frac{1}{2}$. Set $\epsilon' \leftarrow \frac{\epsilon}{k}$. As in [Proposition 28](#), it suffices to repeatedly sample from $\mu D_{k \rightarrow \ell}$ for some choice of ℓ respecting the bound in [Lemma 40](#), within total variation ϵ' . We then condition on this set and then repeat. By the coupling characterization of the total variation distance, the resulting distribution will be at total variation ϵ from μ , since this process will terminate within k rounds. It is straightforward to see this will terminate in $O\left(\sqrt{k / \left(\frac{\alpha}{\log \frac{n}{\epsilon'}} \cdot \epsilon'^{\frac{3}{B}}\right)}\right)$ iterations by a variation of the proof of [Proposition 28](#) and the maximum allowable ℓ in [Lemma 40](#). Setting $B = \frac{3}{c}$ gives the desired bound on the number of iterations. \square

6 Perfect matchings in planar graphs

In this section, we give our parallel algorithm for sampling perfect matchings from a planar graph $G = (V, E)$. Our algorithm departs somewhat from the rejection sampling-based framework used to prove [Theorem 29](#) and its specializations, but we include it as it highlights a different strategy for attaining quadratic parallel speedups by using oracles, catered to the structure of planar graphs. At a high level, our algorithm recursively finds a *planar separator*, which is a small set of vertices $S \subseteq V$ whose removal decomposes V into two disconnected components V_1 and V_2 containing roughly the same number of vertices. We then use our counting oracle to match the vertices in S sequentially, and then recursively solve the sampling problem in the subgraphs on V_1 and V_2 .

Proof of [Theorem 11](#). We find a planar separator $S \subseteq V$ of size $O(\sqrt{n})$ in $\tilde{O}(1)$ parallel time using $\text{poly}(n)$ machines, where removing S from G results in disconnected components V_1, V_2 containing at most $\frac{2n}{3}$ vertices each [[GM87](#)]. To sample a perfect matching recursively, we first sample matching endpoints of vertices in S as follows. Label the vertices in S by $v_1, \dots, v_{|S|}$. Let $v = v_1$, and compute in parallel the probability p_u that a random matching contains the edge (u, v) .² We then sample a matched edge for v from this distribution, set v to the next unmatched vertex in S , and continue conditioned on prior matched edges. Sequentially proceeding over S allows us to sample over partial matchings containing all of S in parallel time $\tilde{O}(\sqrt{n})$. We then remove S (and vertices matched to S in this process) from V_1, V_2 (which can only decrease their sizes), and return the union of the matching of S with the results of recursive calls to V_1 and V_2 .

The recursion depth is $\tilde{O}(1)$, and each round of recursion takes parallel time $O(|S|) = O(\sqrt{n})$, proving the overall parallel depth. For the work bound, let $Q(n)$ be a bound on the number of machines used in the top level of recursion (excluding the recursive calls to V_1, V_2 after matching S), and let $P(n)$ be the total bound on the number of machines used; assume without loss $Q(n) = \Omega(n^2)$. Then,

$$P(n) = 2P(2n/3) + Q(n) \implies P(n) = O(Q(n)). \quad \square$$

7 Hard instance for rejection sampling

In this section, we give a simple hard instance of a fractionally log-concave distribution, which demonstrates that the dependence on k in [Theorem 29](#) may be inherent to our rejection sampling strategy. In particular, it is natural to hope that we can improve [Theorem 29](#) to obtain a parallel depth of $\sqrt{k} \cdot \text{polylog}(k)$, as opposed to $k^{\frac{1}{2}+c}$ for a constant c . Here, we give an example that suggests that new algorithmic techniques may be necessary to obtain this improvement.

Our hard distribution $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$ will be defined as follows. Let n and k be even, and consider a partition of the ground set $[n]$ into pairs $S_i := (2i - 1, 2i)$ for all $i \in [\frac{n}{2}]$. Then, the distribution μ is uniformly supported on sets of the form

$$S := \bigcup_{i \in S'} S_i, \text{ where } S' \in \binom{[\frac{n}{2}]}{\frac{k}{2}}. \quad (5)$$

In other words, μ randomly chooses $\frac{k}{2}$ indices between 1 and $\frac{n}{2}$ and takes the k elements formed by including the pairs corresponding to those indices. It is known that μ is $\Omega(1)$ -FLC (see [[Ana+21a](#)]).

²To see that we can implement this oracle, [[Kas67](#)] shows counting perfect matchings on a planar graph is reducible to determinant computation, which is parallelizable [[Csa75](#)].

To simplify notation, we will assume that $k = o(n)$ and $\ell = o(k)$. We will also assume there is a constant B such that we have access to n^B parallel machines. Following the guarantees of [Algorithm 3](#) in [Proposition 26](#), if we are willing to tolerate a total variation distance of δ from μ_ℓ , we need to show that with probability at least $1 - \delta$, $S \sim \mu_\ell$ satisfies

$$\frac{\mu_\ell(S)}{\mu'_\ell(S)} \leq n^B. \quad (6)$$

Here and throughout the following discussion, $\mu'_\ell(S) = \frac{\ell!}{n^\ell}$ is the probability S is formed by ℓ independent draws from the uniform distribution on $[n]$. In particular, the 1-marginal distribution of μ is uniform, so this is the proposal distribution used by rejection sampling.

Our argument on the tightness of our rejection sampling proceeds as follows. Say that a set $S \in \binom{[n]}{\ell}$ has t “duplicates” if, amongst the elements of S , there are exactly t pairs of elements belonging to the same S_i . For example, for $\ell = 4$ we say the set $\{1, 2, 3, 5\}$ contains 1 duplicate, the pair $(1, 2)$. We first show that for a set S to satisfy (6), it cannot contain more than $t = O(B)$ duplicates. We then show that this limitation, along with attaining a failure probability δ inverse-polynomial in k , forces us to choose $\ell = k^{\frac{1}{2}-c}$ for a constant $c > 0$ which may depend on B .

How many duplicates can we afford? Suppose $S \in \binom{[n]}{\ell}$ contains t duplicates. Each permutation of S is equally likely to be observed by either of the following processes starting from $T_0 = \emptyset$ (we use T_i to denote an ordered set, and S_i to denote its unordered counterpart, for all $i \in [\ell]$).

1. For $i \in [\ell]$, draw $j \in [n]$ uniformly at random and add it to T_{i-1} to form T_i .
2. For $i \in [\ell]$, draw $j \in [n]$ according to the marginal distribution of μ_ℓ conditioned on including T_{i-1} and add it to T_{i-1} to form T_i .

Hence, to bound $\frac{\mu_\ell(S)}{\mu'_\ell(S)}$ as needed by (6), it suffices to fix a permutation T_ℓ of $S_\ell = S$ and bound the ratios of the probabilities T_ℓ is observed according to each of the above processes. It is observed with probability $n^{-\ell}$ according to the first process above, so satisfying (6) means the probability T_ℓ is observed by the second must be at most $n^{B-\ell}$.

It is straightforward to see that the probability we observe each second element in a duplicate pair in the relevant round $i \in [\ell]$ is $\Theta(\frac{1}{k})$. On the other hand, the probability of observing each singleton in its round is $\Theta(\frac{1}{n})$. For $k = o(n)$, this shows that to meet (6) we must have

$$\left(\Theta\left(\frac{1}{n}\right)\right)^{\ell-t} \left(\Theta\left(\frac{1}{k}\right)\right)^t \leq n^{B-\ell}.$$

This shows that we must have t at most a constant (depending on B).

Probability of t duplicates. Let t be a constant. Recall that the distribution μ is uniform over all sets of the form (5), and a sample from $\mu_\ell = D_{k \rightarrow \ell} \mu$ is formed by sampling a set $S \sim \mu$ and then randomly selecting one of the $\binom{k}{\ell}$ subsets of S . Hence, it suffices to fix some S of the form (5) and bound the probability that this downsampling process results in a subset with t duplicates. By symmetry of μ , we lose no generality by only considering the set $S = \cup_{i \in [\frac{k}{2}]} S_i$.

Now, for a constant t , the number of subsets S of size ℓ with exactly t duplicates is

$$\binom{\frac{k}{2}}{t} \cdot \binom{\frac{k}{2} - t}{\ell - 2t} \cdot 2^{\ell - 2t}.$$

The first term corresponds to choosing which t sets S_i will be fully included, the second corresponds to choosing which sets the remaining $\ell - 2t$ elements come from, and the third is because for each of the non-duplicated sets we have two options. Hence, the probability a draw from μ_ℓ has exactly t duplicates for constant t scales as

$$\begin{aligned} \frac{\binom{k}{t} \cdot \binom{k-2t}{\ell-2t} \cdot 2^{\ell-2t}}{\binom{k}{\ell}} &= \left(\Theta\left(\frac{k}{\ell}\right)\right)^{-\ell} \cdot (\Theta(k))^t \cdot \left(\Theta\left(\frac{k}{\ell}\right)\right)^{\ell-2t} \cdot 2^{\ell-2t} \\ &= \left(\Theta\left(\frac{\ell}{k}\right)\right)^{2t} \cdot (\Theta(k))^t = \left(\Theta\left(\frac{\ell^2}{k}\right)\right)^t. \end{aligned}$$

In other words, to guarantee that a draw from μ_ℓ contains less than t duplicates with probability at least $1 - \delta$, we need to ensure that

$$\left(\Theta\left(\frac{\ell^2}{k}\right)\right)^t \leq \delta \implies \ell = O\left(\sqrt{k\delta^{\frac{1}{2t}}}\right).$$

For δ scaling inverse-polynomially in k (which is necessary to perform a union bound over the $\text{poly}(k)$ iterations of rejection sampling), this shows we must take $\ell \leq k^{\frac{1}{2}-c}$ for some constant c which depends on our budget constant B from the earlier discussion.

8 Refined guarantees for bounded symmetric DPPs

For symmetric PSD ensemble matrices L with non-trivial eigenvalue or trace bounds, we give the following refined result improving upon [Theorem 10](#) in various interesting parameter regimes.

Theorem 41. *Let L be a $n \times n$ symmetric PSD matrix and $\epsilon \in (0, 1)$. Let $\mu : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$ be the DPP defined by L . Let $K = L(I + L)^{-1} \leq I$ be the kernel of L . There exists an algorithm to approximately sample from within ϵ total variation distance of μ in*

$$\tilde{O}\left(\min\left\{\sqrt{\text{tr}(K)}, \lambda_{\max}(K)\sqrt{n}\right\}\right)$$

parallel time using $\text{poly}(n)(\frac{1}{\epsilon})^{o(1)}$ machines.

We use [Algorithm 4](#), a “filtered” variant of [Algorithm 1](#).

We prove [Theorem 41](#) in this section. Our first step is to show that for $R = \Theta(\alpha^{-1} \log \frac{n}{\epsilon})$, the output distribution of [Algorithm 4](#) is within ϵ of the target distribution μ .

We require the following helper claims. The first shows that randomly independently dropping elements of a sample from a DPP μ is equivalent to scaling the kernel matrix.

Proposition 42. *Let μ be a DPP with kernel K . Let μ' be the DPP with kernel $K' := \alpha K$. Let ν be the distribution obtained by first sampling $U \sim \mu$, then outputting $S \subseteq U$ with probability $\alpha^{|S|}(1 - \alpha)^{|U|-|S|}$, i.e.*

$$\nu(S) = \sum_{U \supseteq S} \mu(U) \alpha^{|S|} (1 - \alpha)^{|U|-|S|}.$$

Then, μ' and ν are identical.

Algorithm 4: Filtering

Input: DPP $\mu : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$ with kernel K , $\lambda_{\max}(K) \leq \lambda$

$\alpha \leftarrow (\lambda\sqrt{n})^{-1}$

if $\alpha > 1$ **then**

└ (1): Sample $S \sim \mu$ and return S .

$S_{-1}, K^{(0)}, L^{(0)} \leftarrow \emptyset, K, L$

for $i = 0, 1, \dots, R$ **do**

└ (2): Sample $T_i \sim$ DPP with kernel $\tilde{K}^{(i)} := \alpha K^{(i)}$

Update $S_i \leftarrow S_{i-1} \cup T_i$

Update $L^{(i+1)} \leftarrow ((1 - \alpha)L^{(i)})^{T_i}$ (where $(L)^T$ is the ensemble matrix corresponding to the DPP with ensemble matrix L , conditioned on including T ; see [Section 3.2](#))

Update $K^{(i+1)} \leftarrow I - (I + L^{(i+1)})^{-1}$

Output S_R

Proof. Given a set A , we have $\mathbb{P}_{S \sim \mu}[A \subseteq S] = \det((\alpha K)_A) = \alpha^{|A|} \det(K_A) = \alpha^{|A|} \sum_{U \supseteq A} \mu(U)$. On the other hand, we have

$$\begin{aligned} \mathbb{P}_{S \sim \nu}[A \subseteq S] &= \sum_{S \supseteq A} \nu(S) = \sum_{U \supseteq S \supseteq A} \mu(U) \alpha^{|S|} (1 - \alpha)^{|U| - |S|} \\ &= \alpha^{|A|} \sum_{U \supseteq S \supseteq A} \mu(U) \alpha^{|S| - |A|} (1 - \alpha)^{|U| - |S|} \\ &= \alpha^{|A|} \sum_{U \supseteq A} \mu(U) \sum_{S' \subseteq U \setminus A} \alpha^{|S'|} (1 - \alpha)^{|U \setminus A| - |S'|} \\ &= \alpha^{|A|} \sum_{U \supseteq A} \mu(U). \quad \square \end{aligned}$$

Proposition 43. Consider the setup of [Algorithm 4](#). Suppose $\alpha \leq 1$. Let \mathbb{P}_i denote the distribution of S_i . Fix $\epsilon > 0$. For $i = \Omega(\alpha^{-1} \log \frac{n}{\epsilon})$ for a sufficiently large constant, $d_{\text{TV}}(\mathbb{P}_i, \mu) \leq \epsilon$.

Proof. Let $\mu^{(i)}$ be the DPP with ensemble matrix $L^{(i)}$ (and kernel matrix $K^{(i)}$), and let $\nu^{(i)}$ be the DPP with kernel matrix $\alpha K^{(i)}$. We will prove by induction that for all i ,

$$\mathbb{P}_i[S_i] = \sum_{U \supseteq S_i} \mu^{(0)}(U) (1 - \alpha)^{(i+1)(|U| - |S_i|)} (1 - (1 - \alpha)^{i+1})^{|S_i|}.$$

The base case $i = 0$ follows from [Proposition 42](#). Now, supposing the induction hypothesis holds for some $i - 1$, we show that it also holds for i . In the following, let S_0 be the set sampled in the first iteration of [Algorithm 4](#), and let $\mathbb{P}_i[S_i | S_0]$ denote the probability we observe S_i conditioned on the value of S_0 . The induction hypothesis then yields the probability we observe $S_i \setminus S_0$ in the next $i - 1$ iterations, with the starting matrix $L^{(1)} \leftarrow ((1 - \alpha)L)^{S_0}$ as follows:

$$\mathbb{P}_i[S_i | S_0] = \sum_{U \supseteq S_i} \mu^{(1)}(U \setminus S_0) (1 - \alpha)^{i(|U \setminus S_0| - |S_i \setminus S_0|)} (1 - (1 - \alpha)^i)^{|S_i \setminus S_0|}.$$

Hence, we compute

$$\begin{aligned}
\mathbb{P}_i[S_i] &= \sum_{S_0 \subseteq S_i} \mathbb{P}_i[S_i | S_0] \mathbb{P}_0[S_0] \\
&= \sum_{U \supseteq S_i \supseteq S_0} \mu^{(1)}(U \setminus S_0) \mathbb{P}_0[S_0] \\
&\quad \cdot (1 - \alpha)^{i(|U| - |S_i|)} (1 - (1 - \alpha)^i)^{|S_i \setminus S_0|} \\
&= \sum_{U \supseteq S_i \supseteq S_0} \mu^{(0)}(U) (1 - \alpha)^{|U| - |S_0|} \alpha^{|S_0|} \\
&\quad \cdot (1 - \alpha)^{i(|U| - |S_i|)} (1 - (1 - \alpha)^i)^{|S_i \setminus S_0|} \\
&= \sum_{U \supseteq S_i} \mu^{(0)}(U) (1 - \alpha)^{(i+1)(|U| - |S_i|)} \\
&\quad \cdot \sum_{S_0 \subseteq S_i} (1 - \alpha)^{|S_i| - |S_0|} (1 - (1 - \alpha)^i)^{|S_i \setminus S_0|} \alpha^{|S_0|} \\
&= \sum_{U \supseteq S_i} \mu^{(0)}(U) (1 - \alpha)^{(i+1)(|U| - |S_i|)} \\
&\quad \cdot \left(\alpha + (1 - \alpha)(1 - (1 - \alpha)^i) \right)^{|S_i|} \\
&= \sum_{U \supseteq S_i} \mu^{(0)}(U) (1 - \alpha)^{(i+1)(|U| - |S_i|)} (1 - (1 - \alpha)^{i+1})^{|S_i|}
\end{aligned}$$

where the third equality uses [Proposition 42](#) and the definition of $L_1 = ((1 - \alpha)L)^{S_0}$ to derive

$$\begin{aligned}
\mu^{(1)}(U \setminus S_0) \mathbb{P}_0[S_0] &= \mu^{(1)}(U \setminus S_0) \sum_{V \supseteq S_0} (1 - \alpha)^{|V \setminus S_0|} \alpha^{|S_0|} \mu^{(0)}(V) \\
&= \frac{(1 - \alpha)^{|U \setminus S_0|} \mu^{(0)}(U)}{\sum_{V \supseteq S_0} (1 - \alpha)^{|V \setminus S_0|} \mu^{(0)}(V)} \\
&\quad \cdot \sum_{V \supseteq S_0} (1 - \alpha)^{|V \setminus S_0|} \alpha^{|S_0|} \mu^{(0)}(V) \\
&= (1 - \alpha)^{|U \setminus S_0|} \alpha^{|S_0|} \mu^{(0)}(U).
\end{aligned}$$

Thus the induction hypothesis holds for all i . By taking $i = \Omega\left(\frac{\log \frac{n}{\epsilon}}{\alpha}\right)$, and only considering the summand corresponding to $U = S_i$,

$$\mathbb{P}_i[S_i] \geq \mu(S_i) (1 - (1 - \alpha)^{i+1})^{|S_i|} \geq \mu(S_i) \left(1 - O\left(\frac{\epsilon}{n}\right)\right)^n \geq \mu(S_i) (1 - \epsilon),$$

and hence,

$$\begin{aligned}
d_{\text{TV}}(\mathbb{P}_i[\cdot], \mu) &= \sum_{S_i: \mathbb{P}_i[S_i] \leq \mu(S_i)} (\mu(S_i) - \mathbb{P}_i[S_i]) \\
&\leq \sum_{S_i: \mathbb{P}_i[S_i] \leq \mu(S_i)} \epsilon \mu(S_i) \leq \epsilon.
\end{aligned}$$

□

Next, we show that each step of the for loop can be implemented in constant parallel time.

Lemma 44. Let $\mu : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$ be a DPP with marginal kernel K . If $\lambda_{\max}(K) \leq \frac{1}{\sqrt{n}}$, we can sample from a distribution ϵ -away in total variation from μ in $\tilde{O}(1)$ time using $O(\text{poly}(n)(\frac{1}{\epsilon})^{o(1)})$ machines.

Proof. Let $s = c\sqrt{n \log \frac{1}{\epsilon}}$ for c as in Lemma 14. Let $\Omega := \{S \subseteq [n] \mid |S| \leq s\}$. Let $p_i := K_{i,i} = \mathbb{P}_{S \sim \mu}[i \in S]$. Let ν be the distribution obtained by independently sampling independent $b_i \sim \text{Ber}(p_i)$ for all $i \in [n]$, and outputting $T = \{i \mid b_i = 1\}$. By Lemma 14, $\sum_{S \in \Omega} \mu(S) \geq 1 - \epsilon'$. Moreover, for fixed $T \in \Omega$, we have

$$\begin{aligned} \frac{\mu(T)}{\nu(T)} &= \frac{\det L_T}{\det(I + L)} \left(\prod_{i \in T} p_i \prod_{i \notin T} (1 - p_i) \right)^{-1} \\ &= \det(L_T) \det(I - K) \left(\prod_{i \in T} K_{i,i} \prod_{i \notin T} (1 - K_{i,i}) \right)^{-1}, \end{aligned}$$

where we use $I + L = (I - K)^{-1}$. By applying Corollary 18 to $I - K$, we have

$$\det(I - K) \leq \prod_{i \in [n]} (1 - K_{i,i}) \leq \prod_{i \notin T} (1 - K_{i,i}),$$

so it suffices to show

$$\det(L_T) \leq \left(\frac{1}{\epsilon}\right)^{o(1)} \prod_{i \in T} K_{i,i} \implies \frac{\mu(T)}{\nu(T)} \leq \left(\frac{1}{\epsilon}\right)^{o(1)},$$

at which point we can apply Proposition 26. Let $K = UDU^\top$ where $U \in \mathbb{R}^{n \times n}$ is an orthonormal basis of eigenvectors of K , and $D = \text{diag}(\{\lambda_i\}_{i \in [n]})$, where $\lambda_1 \geq \dots \geq \lambda_n$ are the eigenvalues of K . By (2), we can write

$$L = U \text{diag} \left(\left\{ \frac{\lambda_i}{1 - \lambda_i} \right\}_{i \in [n]} \right) U^\top.$$

Thus, by applying the Cauchy-Binet formula twice,

$$\begin{aligned} \det(L_T) &= \det \left(U_{T,[n]} \text{diag} \left(\left\{ \frac{\lambda_i}{1 - \lambda_i} \right\}_{i \in [n]} \right) U_{[n],T}^\top \right) \\ &= \sum_{S \subseteq [n], |S|=|T|} \det(\Lambda_{T,S}) \left(\prod_{i \in S} \frac{\lambda_i}{1 - \lambda_i} \right) \det(\Lambda_{S,T}^\top) \\ &\leq \exp \left(c \sqrt{\log \frac{1}{\epsilon}} \right) \sum_{S \subseteq [n], |S|=|T|} \det(\Lambda_{T,S}) \left(\prod_{i \in S} \lambda_i \right) \det(\Lambda_{S,T}^\top) \\ &= \exp \left(c \sqrt{\log \frac{1}{\epsilon}} \right) \det(K_T) \\ &\leq \exp \left(c \sqrt{\log \frac{1}{\epsilon}} \right) \prod_{i \in T} K_{i,i} \end{aligned}$$

where in the first inequality, we used

$$\prod_{i \in S} (1 - \lambda_i) \geq \left(1 - \frac{1}{\sqrt{n}}\right)^{|S|} \geq \left(1 - \frac{1}{\sqrt{n}}\right)^s \geq \exp\left(-c\sqrt{\log \frac{1}{\epsilon}}\right)$$

and in the last inequality, we used [Corollary 18](#). Thus by [Proposition 26](#), we can sample from $\tilde{\mu}$ that is ϵ -away from μ in $O(1)$ parallel time using $O((\frac{1}{\epsilon})^{o(1)} \text{poly}(n))$ machines, by setting $\delta \leftarrow \frac{\epsilon}{2}$ and adjusting the definition of ϵ in this proof by a constant. \square

Proposition 45. *Consider the setup of [Algorithm 4](#). Suppose $\alpha \leq 1$. We have $\lambda_{\max}(K^{(i)}) \leq \lambda$ for all i , so that $\lambda_{\max}(\tilde{K}^{(i)}) \leq \frac{1}{\sqrt{n}}$. Consequently, each iteration of the for loop can be implemented in $\tilde{O}(1)$ parallel time using $O(\text{poly}(n)(\frac{1}{\epsilon})^{o(1)})$ machines, up to total variation distance ϵ .*

Proof. We inductively show that $\lambda_{\max}(K^{(i)}) \leq \lambda$ for all i . The base case $i = 0$ directly follows from the input assumption. Now let us assume that $\lambda_{\max}(K^{(i)}) \leq \lambda$ for some $i \geq 0$. We will show that $\lambda_{\max}(K^{(i+1)}) \leq \lambda$ follows. Let S be the index set of L^i and $\tilde{S} = S \setminus T_i$ where T_i was sampled. Then,

$$L^{(i+1)} = ((1 - \alpha)L^{(i)})^{T_i} = (1 - \alpha)L_{\tilde{S}} - (1 - \alpha)L_{\tilde{S}, T_i} L_{T_i, T_i}^{-1} L_{T_i, \tilde{S}}.$$

Since $L^{(i)}$ is PSD, $L_{\tilde{S}, T_i}^{(i)} \left(L_{T_i, T_i}^{(i)}\right)^{-1} L_{T_i, \tilde{S}}^{(i)} \geq 0$. Thus $L^{(i+1)} \preceq (1 - \alpha)L_{\tilde{S}}^{(i)} \preceq L_{\tilde{S}}^{(i)}$.

Let Λ be the set of the eigenvalues of $K^{(i)}$. Due to (2), the eigenvalues of $L^{(i)}$ are given by $\{\frac{\lambda}{1-\lambda} \mid \lambda \in \Lambda\}$. As $\frac{\lambda}{1-\lambda}$ is strictly increasing in the range $[0, 1]$, and the largest eigenvalue of $L^{(i+1)}$ is dominated by the largest eigenvalue of $L^{(i)}$ (since restrictions to index sets can only decrease quadratic forms), we have the first desired conclusion. The second conclusion follows from [Lemma 44](#) as the eigenvalue bound is satisfied. \square

Finally, we are ready to prove [Theorem 41](#).

Proof of [Theorem 41](#). The bound involving $\text{tr}K$ follows from a similar argument as in [Remark 15](#). Note that $\text{tr}K = \mathbb{E}_{S \sim \mu}[|S|]$, and that by [Lemma 14](#), the set $\Omega := \{S \subseteq [n] \mid |S| \leq \text{tr}K \log \frac{2}{\epsilon}\}$ has $\mu(\Omega) \geq 1 - \frac{\epsilon}{2}$. When drawing k from \mathcal{H} (the distribution on cardinality values), if $k \leq \text{tr}K \log \frac{2}{\epsilon}$, we use [Theorem 10](#) to approximately sample from within $\frac{\epsilon}{2}$ of μ_k , else we output an arbitrary subset. By the triangle inequality, the output's distribution is within $\frac{\epsilon}{2} + \frac{\epsilon}{2} = \epsilon$ of μ . The algorithm runs in the stated parallel time depending on $\text{tr}K$ using the number of machines as [Theorem 10](#).

Now we focus on the bound involving $\lambda_{\max}(K)$. If $\alpha > 1$ then the conclusion follows from [Lemma 44](#) applied to step (1). Else, suppose $\alpha \leq 1$. We run [Algorithm 4](#) with $R = O(\lambda\sqrt{n} \log \frac{n}{\epsilon})$ such that [Proposition 43](#) guarantees that if we can run the algorithm correctly, the output has total variation $\frac{\epsilon}{2}$. Let $\epsilon' = \frac{\epsilon}{R}$. Let $\nu^{(i)}$ be the target distribution of T_i in i^{th} step of the for loop. By [Proposition 45](#), we can modify step (2) to sample from $\hat{\nu}^{(i)}$ that is ϵ' -away from $\nu^{(i)}$ in TV-distance in $\tilde{O}(1)$ time using $O(\text{poly}(n)(\frac{1}{\epsilon})^{o(1)})$ machines. Hence, by the triangle inequality, the output of the algorithm is $\frac{\epsilon}{2}$ away from the output if we were given exact sample access to each $\nu^{(i)}$. Combining with the approximation error of [Proposition 43](#) yields the conclusion. \square

9 Deferred proofs

Proof of Lemma 12. We first prove the inequality for the case $S = [n]$ and $C = 1$. Clearly,

$$f'(r) = n^{\lambda-1} \lambda \left(r^{\lambda-1} - (\lambda-1)(1 + \log r + \log n) \right)$$

and

$$f''(r) = n^{\lambda-1} \lambda (\lambda-1) \left(r^{\lambda-2} - \frac{1}{r} \right) \leq 0.$$

Thus f is concave and

$$\frac{1}{n} \sum_{i=1}^n f(q_i) \leq f\left(\frac{1}{n} \sum_{i=1}^n q_i\right) = f\left(\frac{1}{n}\right) = \frac{1}{n}$$

which is equivalent to

$$\sum_{i=1}^n q_i^\lambda n^{\lambda-1} \leq 1 + n^{\lambda-1} \lambda (\lambda-1) \sum_{i=1}^n q_i \log(nq_i).$$

The case $C > 1$ then follows from

$$\begin{aligned} \mathcal{D}_\lambda(q \parallel p) &= \sum_{i=1}^n q_i^\lambda p_i^{1-\lambda} \\ &\leq C^{\lambda-1} \sum_{i=1}^n q_i^\lambda n^{\lambda-1} \\ &\leq C^{\lambda-1} \left(1 + n^{\lambda-1} \lambda (\lambda-1) \sum_{i=1}^n q_i \log(nq_i) \right) \\ &\leq C^{\lambda-1} \left(1 + n^{\lambda-1} \lambda (\lambda-1) \left(\sum_{i=1}^n q_i \log \frac{q_i}{p_i} + \log C \right) \right). \end{aligned}$$

The case $S \neq [n]$ follows by noticing

$$\sum_{i \in S} q_i \left(\frac{q_i}{p_i} \right)^{\lambda-1} \leq \sum_{i=1}^n q_i^\lambda p_i^{1-\lambda}. \quad \square$$

Proof of Proposition 13. First, note that DPPs and k -DPPs correspond to Partition-DPPs with 0 and 1 partition constraints respectively. We thus show the claim for μ a Partition-DPP with $O(1)$ constraints. Computing the marginals is equivalent to computing the partition functions of μ and μ conditioned on subsets. As shown in [Cel+17, Theorem 1.1], computing the partition function is equivalent to computing the coefficients of a polynomial in $O(1)$ variables, obtained from the generating polynomial by plugging the same variable for all elements in a partition. This polynomial can be evaluated efficiently given access to g_μ . Evaluating g_μ at (z_1, \dots, z_n) is equivalent to computing $\det(L + \text{diag}(z_i)_{i=1}^n)$, which can be done in $\tilde{O}(1)$ -parallel time [Ber84]. From evaluations we can obtain the coefficients by polynomial interpolation; this can be solved, e.g., by solving linear equations, doable in NC [Csa75]. \square

Proof of Lemma 14. Let $f(S) = |S|$ correspond to the 1-Lipschitz (with respect to the Hamming metric) function of the size of a sample S from μ , and let \bar{f} indicate the value $\mathbb{E}_{S \sim \mu}[f]$.

By applying known concentration inequalities for strongly Rayleigh distributions [PP13, Theorem 3.2], it follows that

$$\mathbb{P}_{S \sim \mu}[f - \bar{f} > a] \leq 3 \exp\left(\frac{-a^2}{16(a + 2\bar{f})}\right).$$

For the first statement, let $a = 10\sqrt{n \log \frac{1}{\epsilon}}$, and note $\exp(-\frac{a^2}{16(a+2\bar{f})}) \leq \frac{\epsilon}{3}$ and $a + \bar{f} \leq 11\sqrt{n \log \frac{1}{\epsilon}}$. For the second, let $a = \bar{f} \log \frac{1}{\epsilon}$ and note $\exp(-\frac{a^2}{16(a+2\bar{f})}) \leq \frac{\epsilon}{3}$ and $a + \bar{f} \leq 2\bar{f} \log \frac{1}{\epsilon}$. \square

Proof of Proposition 32. First, we check the cardinality of the new ground set U :

$$\begin{aligned} n\beta^{-1} &= \sum_{i=1}^n \frac{n}{k\beta} p_i \leq |U| \\ &= \sum_{i=1}^n t_i \leq \sum_{i=1}^n \left(1 + \frac{n}{k\beta} p_i\right) = n + \frac{n}{k\beta} \sum_{i=1}^n p_i = n(1 + \beta^{-1}). \end{aligned}$$

Next, we check that for any $i^{(j)}$, the marginal probabilities $\mathbb{P}_{S \sim \mu^{\text{iso}}}[i^{(j)} \in S]$ are at most $\frac{Ck}{|U|}$. In the following calculation, we interpret sampling from μ^{iso} as first sampling from μ and then choosing a copy $j \in [t_i]$ for each element. This yields

$$\begin{aligned} \mathbb{P}_{S \sim \mu^{\text{iso}}}[i^{(j)} \in S] &= \sum_{S \ni i} \mathbb{P}[\text{we chose copy } j \mid \text{we sampled } S \text{ from } \mu] \\ &\quad \cdot \mathbb{P}[\text{we sampled } S \text{ from } \mu] \\ &= \sum_{S \ni i} \frac{1}{t_i} \cdot \mu(S) = \frac{1}{t_i} \sum_{S \ni i} \mu(S) = \frac{1}{t_i} \cdot \mathbb{P}_{S \sim \mu}[i \in S]. \end{aligned}$$

Since $1 + \frac{n}{\beta k} p_i \geq t_i \geq \frac{n}{\beta k} p_i$, we obtain

$$\begin{aligned} \frac{k}{kp_i^{-1} + n\beta^{-1}} &= \frac{p_i}{1 + \frac{n}{\beta k} p_i} \leq \mathbb{P}_{S \sim \mu^{\text{iso}}}[i^{(j)} \in S] \\ &\leq \frac{\beta k}{n} = \frac{k(\beta + 1)}{n(1 + \beta^{-1})} \leq \frac{k(\beta + 1)}{|U|} \leq \frac{Ck}{|U|}. \end{aligned}$$

The latter inequality shows the marginal upper bound. Next, to show the marginal lower bound, suppose $\mathbb{P}_{\mu}[i \in S] = p_i \geq \frac{\sqrt{\beta k}}{n}$. Then for all $j \in [t_i]$,

$$\mathbb{P}_{S \sim \mu^{\text{iso}}}[i^{(j)} \in S] \geq \frac{k}{kp_i^{-1} + n\beta^{-1}} \geq \frac{k}{n\beta^{-1}(1 + \sqrt{\beta})} \geq \frac{k}{C|U|}.$$

Finally, letting $\bar{R} := \left\{ i \mid p_i \geq \frac{\sqrt{\beta k}}{n} \right\} \subseteq [n]$,

$$\begin{aligned}
\sum_{S \in \binom{[R]}{\ell}} \mu_\ell^{\text{iso}}(S) &= \sum_{\bar{S} \in \binom{\bar{R}}{\ell}} \mu_\ell(\bar{S}) \\
&= 1 - \sum_{\bar{S} \subseteq \binom{[n]}{\ell}, \bar{S} \not\subseteq \bar{R}} \mu_\ell(\bar{S}) \\
&\geq 1 - \sum_{i \notin \bar{R}} \sum_{\bar{S} \subseteq \binom{[n]}{\ell}, i \in \bar{S}} \mu_\ell(\bar{S}) \\
&= 1 - \sum_{i \notin \bar{R}} \frac{\ell p_i}{k} \geq 1 - \sqrt{\beta \ell}. \quad \square
\end{aligned}$$

Proof of Lemma 34. Throughout this proof, fix the set $S \in \binom{[n]}{t}$, and let A_S denote the left-hand side of (3). Let $q_i := \mathbb{P}_{T \sim \mu}[i \in T \mid S \subseteq T]$, and note that $q_i = 1$ for all $i \in S$. Moreover, we have

$$\begin{aligned}
A_S &= \sum_{i \notin S} \frac{q_i}{k-t} \log \left(\frac{q_i}{p_i} \cdot \frac{k}{k-t} \right) \\
&= \frac{k}{k-t} \sum_{i \notin S} \frac{q_i}{k} \log \left(\frac{q_i}{p_i} \right) + \log \frac{k}{k-t} \\
&\leq \frac{k}{k-t} \sum_{i \notin S} \frac{q_i}{k} \log \left(\frac{q_i}{p_i} \right) + \frac{2t}{k}.
\end{aligned} \tag{7}$$

The first equation used that by definition, $\mu_{t+1|S} = \frac{q_S^c}{k-t}$ where q_S^c restricts q to $S^c := [n] \setminus S$; the only inequality used $\log(1+c) \leq c$ for all $c \geq 0$ and $t \leq \frac{1}{2}k$. We note that for

$$B_S := \sum_{i \notin S} \frac{q_i}{k} \log \left(\frac{q_i}{p_i} \right) + \sum_{i \in S} \frac{1}{k} \log \frac{1}{p_i},$$

we have by $t \leq \frac{1}{2}k$ that

$$A_S \leq 2 \sum_{i \notin S} \frac{q_i}{k} \log \left(\frac{q_i}{p_i} \right) + \frac{2t}{k} \leq 2B_S + \frac{2t}{k}, \tag{8}$$

since $\log \frac{1}{p_i} \geq 0$ for all $i \in [n]$. We next give an interpretation of the quantity B_S . Let μ_S be the distribution of $T \sim \mu$ conditioned on $S \subset T$, so that

$$\mu_S D_{k \rightarrow 1} = \begin{cases} \frac{1}{k} & i \in S \\ \frac{q_i}{k} & i \notin S \end{cases}.$$

Notice that B_S is defined to be $\mathcal{D}_{\text{KL}}(\mu_S D_{k \rightarrow 1} \parallel \mu D_{k \rightarrow 1})$ (since $\mu D_{k \rightarrow 1} = \frac{1}{k}p$), which we can control

by entropic independence of μ . In particular,

$$\begin{aligned}
B_S &= \mathcal{D}_{\text{KL}}(\mu_S D_{k \rightarrow 1} \parallel \mu D_{k \rightarrow 1}) \leq \frac{1}{\alpha k} \mathcal{D}_{\text{KL}}(\mu_S \parallel \mu) \\
&= \frac{1}{\alpha k} \sum_{\substack{T \in \binom{[n]}{k} \\ S \subset T}} \mu_S(T) \log \frac{\mu_S(T)}{\mu(T)} \\
&= \frac{1}{\alpha k} \sum_{\substack{T \in \binom{[n]}{k} \\ S \subset T}} \mu_S(T) \\
&\quad \cdot \log \left(\frac{\mu(T)}{\mathbb{P}_{T \sim \mu}[S \subset T]} \cdot \frac{1}{\mu(T)} \right) \\
&= \frac{1}{\alpha k} \log \left(\frac{1}{\mathbb{P}_{T \sim \mu}[S \subset T]} \right).
\end{aligned}$$

Combining the above display with (8) completes the proof. \square

References

- [AD20] Nima Anari and Michał Dereziński. “Isotropy and Log-Concave Polynomials: Accelerated Sampling and High-Precision Counting of Matroid Bases”. In: *Proceedings of the 61st Annual Symposium on Foundations of Computer Science*. 2020.
- [AL20] Vedat Levi Alev and Lap Chi Lau. “Improved analysis of higher order random walks and applications”. In: *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*. 2020, pp. 1198–1211.
- [Ald90] David J Aldous. “The random walk construction of uniform spanning trees and uniform labelled trees”. In: *SIAM Journal on Discrete Mathematics* 3.4 (1990), pp. 450–465.
- [Ali+21] Yeganeh Alimohammadi, Nima Anari, Kirankumar Shiragur, and Thuy-Duong Vuong. “Fractionally Log-Concave and Sector-Stable Polynomials: Counting Planar Matchings and More”. In: *arXiv preprint arXiv:2102.02708* (2021).
- [ALO20] Nima Anari, Kuikui Liu, and Shayan Oveis Gharan. “Spectral Independence in High-Dimensional Expanders and Applications to the Hardcore Model”. In: *Proceedings of the 61st IEEE Annual Symposium on Foundations of Computer Science*. IEEE Computer Society, 2020.
- [Ana+19] Nima Anari, Kuikui Liu, Shayan Oveis Gharan, and Cynthia Vinzant. “Log-concave polynomials II: high-dimensional walks and an FPRAS for counting bases of a matroid”. In: *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*. 2019, pp. 1–12.
- [Ana+20] Nima Anari, Nathan Hu, Amin Saberi, and Aaron Schild. “Sampling Arborescences in Parallel”. In: *arXiv preprint arXiv:2012.09502* (2020).
- [Ana+21a] Nima Anari, Michał Dereziński, Thuy-Duong Vuong, and Elizabeth Yang. “Domain Sparsification of Discrete Distributions using Entropic Independence”. In: *CoRR abs/2109.06442* (2021).
- [Ana+21b] Nima Anari, Vishesh Jain, Frederic Koehler, Huy Tuan Pham, and Thuy-Duong Vuong. “Entropic Independence I: Modified Log-Sobolev Inequalities for Fractionally Log-Concave Distributions and High-Temperature Ising Models”. In: *CoRR abs/2106.04105* (2021). arXiv: 2106.04105.

- [Ana+21c] Nima Anari, Vishesh Jain, Frederic Koehler, Huy Tuan Pham, and Thuy-Duong Vuong. “Entropic Independence II: Optimal Sampling and Concentration via Restricted Modified Log-Sobolev Inequalities”. In: *arXiv preprint arXiv:2111.03247* (2021).
- [Ana+23] Nima Anari, Yizhi Huang, Thuy-Duong Vuong, Brian Xu, and Katherine Yu. “Parallel Discrete Sampling via Continuous Walks”. 2023.
- [Bar18] Alexander Barvinok. “Approximating real-rooted and stable polynomials, with combinatorial applications”. In: *arXiv preprint arXiv:1806.07404* (2018).
- [BBL09] Julius Borcea, Petter Brändén, and Thomas Liggett. “Negative dependence and the geometry of polynomials”. In: *Journal of the American Mathematical Society* 22.2 (2009), pp. 521–567.
- [Ber84] Stuart J. Berkowitz. “On computing the determinant in small parallel time using a small number of processors”. In: *Information Processing Letters* 18.3 (1984), pp. 147–150.
- [Bro89] Andrei Z Broder. “Generating random spanning trees”. In: *FOCS*. Vol. 89. Citeseer, 1989, pp. 442–447.
- [Bru18] Victor-Emmanuel Brunel. “Learning Signed Determinantal Point Processes through the Principal Minor Assignment Problem”. In: *Advances in Neural Information Processing Systems*. Ed. by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett. Vol. 31. Curran Associates, Inc., 2018, pp. 7365–7374.
- [Cel+16] L Elisa Celis, Amit Deshpande, Tarun Kathuria, Damian Straszak, and Nisheeth K Vishnoi. “On the complexity of constrained determinantal point processes”. In: *arXiv preprint arXiv:1608.00554* (2016).
- [Cel+17] L. Elisa Celis, Amit Deshpande, Tarun Kathuria, Damian Straszak, and Nisheeth K. Vishnoi. “On the Complexity of Constrained Determinantal Point Processes”. In: *APPROX-RANDOM*. 2017.
- [CLV21] Zongchen Chen, Kuikui Liu, and Eric Vigoda. “Spectral independence via stability and applications to holant-type problems”. In: *arXiv preprint arXiv:2106.03366* (2021).
- [Csa75] Laszlo Csanky. “Fast parallel matrix inversion algorithms”. In: *16th Annual Symposium on Foundations of Computer Science (sfcs 1975)*. IEEE, 1975, pp. 11–12.
- [DM21] Michał Dereziński and Michael W Mahoney. “Determinantal point processes in randomized numerical linear algebra”. In: *Notices of the American Mathematical Society* 68.1 (2021), pp. 34–45.
- [Elf+19] Mohamed Elfeki, Camille Couprie, Morgane Riviere, and Mohamed Elhoseiny. “GDPP: Learning diverse generations using determinantal point processes”. In: *International Conference on Machine Learning*. PMLR, 2019, pp. 1774–1783.
- [FHY21] Weiming Feng, Thomas P Hayes, and Yitong Yin. “Distributed metropolis sampler with optimal parallelism”. In: *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*. SIAM, 2021, pp. 2121–2140.
- [Gar+19] Mike Gartrell, Victor-Emmanuel Brunel, Elvis Dohmatob, and Syrine Krichene. “Learning Nonsymmetric Determinantal Point Processes”. In: *ArXiv abs/1905.12962* (2019).
- [Gar+20] Mike Gartrell, Insu Han, Elvis Dohmatob, Jennifer Gillenwater, and Victor-Emmanuel Brunel. *Scalable Learning and MAP Inference for Nonsymmetric Determinantal Point Processes*. 2020. arXiv: 2006.09862 [cs.LG].
- [GM87] Hillel Gazit and Gary L. Miller. “A parallel algorithm for finding a separator in planar graphs”. In: *28th Annual Symposium on Foundations of Computer Science (sfcs 1987)*. 1987, pp. 238–248. DOI: 10.1109/SFCS.1987.3.
- [Gon+14] Boqing Gong, Wei-lun Chao, Kristen Grauman, and Fei Sha. *Large-Margin Determinantal Point Processes*. 2014. arXiv: 1411.1537 [stat.ML].

- [GPK16] Mike Gartrell, Ulrich Paquet, and Noam Koenigstein. “Bayesian Low-Rank Determinantal Point Processes”. In: *Proceedings of the 10th ACM Conference on Recommender Systems*. RecSys ’16. Boston, Massachusetts, USA: Association for Computing Machinery, 2016, pp. 349–356. ISBN: 9781450340359. DOI: 10.1145/2959100.2959178.
- [JVV86] Mark R Jerrum, Leslie G Valiant, and Vijay V Vazirani. “Random generation of combinatorial structures from a uniform distribution”. In: *Theoretical computer science* 43 (1986), pp. 169–188.
- [Kas67] Pieter Kasteleyn. “Graph theory and crystal physics”. In: *Graph theory and theoretical physics* (1967), pp. 43–110.
- [KT12a] Alex Kulesza and Ben Taskar. “Determinantal Point Processes for Machine Learning”. In: *Found. Trends Mach. Learn.* 5.2-3 (2012), pp. 123–286.
- [KT12b] Alex Kulesza and Ben Taskar. *k-DPPs: Fixed-Size Determinantal Point Processes*. 2012.
- [LB12] Hui Lin and Jeff Bilmes. “Learning Mixtures of Submodular Shells with Application to Document Summarization”. In: *Uncertainty in Artificial Intelligence - Proceedings of the 28th Conference, UAI 2012* (Oct. 2012).
- [LJS16] Chengtao Li, Stefanie Jegelka, and Suvrit Sra. “Fast DPP Sampling for Nyström with Application to Kernel Methods”. In: *CoRR* abs/1603.06052 (2016). arXiv: 1603.06052.
- [LY21] Hongyang Liu and Yitong Yin. “Simple Parallel Algorithms for Single-Site Dynamics”. In: *arXiv preprint arXiv:2111.04044* (2021).
- [MS15] Zelda Mariet and Suvrit Sra. “Fixed-point algorithms for determinantal point processes”. In: *CoRR*, abs/1508.00792 (2015).
- [PP13] Robin Pemantle and Yuval Peres. *Concentration of Lipschitz functionals of determinantal and other strong Rayleigh measures*. 2013. arXiv: 1108.0687 [math.PR].
- [PR17] Viresh Patel and Guus Regts. “Deterministic polynomial-time approximation algorithms for partition functions and graph polynomials”. In: *SIAM Journal on Computing* 46.6 (2017), pp. 1893–1919.
- [Ten95] Shang-Hua Teng. “Independent sets versus perfect matchings”. In: *Theoretical Computer Science* 145.1-2 (1995), pp. 381–390.
- [Wil+18] Mark Wilhelm, Ajith Ramanathan, Alexander Bonomo, Sagar Jain, Ed H. Chi, and Jennifer Gillenwater. “Practical Diversified Recommendations on YouTube with Determinantal Point Processes”. In: *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. CIKM ’18. Torino, Italy: Association for Computing Machinery, 2018, pp. 2165–2173. ISBN: 9781450360142. DOI: 10.1145/3269206.3272018.