

# Improved Sampling-to-Counting Reductions in High-Dimensional Expanders and Faster Parallel Determinantal Sampling

Nima Anari<sup>1</sup>, Callum Burgess<sup>1</sup>, Kevin Tian<sup>1</sup>, and Thuy-Duong Vuong<sup>1</sup>

<sup>1</sup>Stanford University, {anari, callumb, kjtian, tdvuong}@stanford.edu

## Abstract

We study parallel sampling algorithms for classes of distributions defined via determinants: symmetric, nonsymmetric, and partition-constrained determinantal point processes. For these distributions, counting, a.k.a. computing the partition function, can be reduced to a simple determinant computation which is highly parallelizable; Csanky proved it is in NC. However, parallel counting does not automatically translate to parallel sampling, as the classic reductions between sampling and counting are inherently sequential. Despite this, we show that for all the aforementioned determinant-based distributions, a roughly quadratic parallel speedup over sequential sampling can be achieved. 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  $c > 0$ . In the special case of symmetric determinantal point processes, our bound improves to  $\tilde{O}(\sqrt{k})$  and we show how to sample exactly in this case.

We obtain our results via a generic sampling-to-counting reduction that uses approximate rejection sampling. As our main technical contribution, we show that whenever a distribution satisfies a certain form of high-dimensional expansion called entropic independence, approximate rejection sampling can achieve a roughly quadratic speedup in sampling via counting. Various forms of high-dimensional expansion, including the notion of entropic independence we use in this work, have been the source of major breakthroughs in sampling algorithms in recent years; thus we expect our framework to prove useful in the future for distributions beyond those defined by determinants.

# 1 Introduction

Sampling and counting are intimately connected computational problems. For many classes of distributions defined by weight functions  $\mu : X \rightarrow \mathbb{R}_{\geq 0}$ , where typically the space  $X$  is exponential-sized, the problems of 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 appears to break down for complexity classes below P. For example, there is no known polylogarithmic-overhead parallel reduction between approximate counting and approximate sampling.

Motivated by the mysterious relationship between sampling and counting in the parallel algorithms world, Anari, Hu, Saberi, and Schild [Ana+20], based on the earlier work of Teng [Ten95], raised the question of designing fast parallel *sampling* algorithms for several classes of distributions where counting, even exactly, was possible in polylogarithmic time and polynomial work, i.e., in the class NC. The distributions in this challenge set all enjoy fast parallel counting algorithms because their partition functions 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, the algorithms in these two works are highly tailored to the random spanning tree and random arborescence distributions, and they do not provide any general recipe for parallel sampling from 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 a recent trend 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, namely entropic independence [Ana+21b], the sampling-to-counting reduction can be made roughly quadratically faster using parallelization. We formally define entropic independence in Section 3, see Definition 20.

In our setup, we consider combinatorial distributions defined on size  $k$  sets of a ground set of elements  $[n]$ , which we denote by an unnormalized density

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

We remark that the choice of  $\binom{[n]}{k}$  is standard in the high-dimensional expanders literature, and many other domains such as the popular product spaces, can be naturally transformed into  $\binom{[n]}{k}$  [ALO20]. Our access to this distribution is through an oracle that can answer counting queries. Given *any*<sup>1</sup> 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 classical reduction from sampling to counting 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 in the ground set are computed and a new element

---

<sup>1</sup>Note that by querying sets  $T$  of size exactly  $k$ , a counting query can also return the value of  $\mu$  on any desired set.

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 address is:

For which  $\mu$  is there a faster parallel reduction from sampling to counting?

Our main result establishes that for distributions  $\mu$  which are good high-dimensional expanders, measured in terms of the notion of entropic independence [Ana+21b], the sampling-to-counting reduction can be sped up roughly quadratically. Throughout (e.g. in Theorem 1), 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, informal, see Theorem 33). *Let  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$  be  $O(1)$ -entropically independent, and assume that we have access to a counting oracle for  $\mu$ . For any constant  $c > 0$  and any  $\epsilon \in (0, 1)$ , there exists an algorithm that can sample from a distribution within total variation distance  $\epsilon$  of  $\mu$  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 of computation.*

We remark that various notions of high-dimensional expansion and in particular entropic independence have proven useful in the analysis of Markov chains and sequential sampling algorithms [Ana+21b; Ana+21c], but this is the first work to relate these notions to parallel algorithms. We also note that entropic independence is not a binary property of a distribution, but rather every distribution  $\mu$  has some parameter of entropic independence  $\in [1, k]$  and the above result applies for all distributions whose entropic independence parameter is  $O(1)$ . See Theorem 33 in Section 6 for details. This is also the exact condition that implies fast mixing of Markov chains and a polynomial runtime for sequential sampling algorithms [Ana+21b].

*Remark 2* (Beyond determinantal distributions). In this work, we explore the applications of Theorem 1 to distributions defined via determinants defined in the next section; this is due to the fact that partition functions of such distributions can be computed in NC, which gives us a fast parallel counting oracle. However, we believe Theorem 1 may find 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 have the potential to be parallelized, as they involve enumerating a small number of combinatorial structures, unlike the more involved and inherently sequential Markov Chain Monte Carlo methods. Recent works [Ali+21; CLV21] have shown that absence of roots in the complex plane implies forms of high-dimensional expansion, including entropic independence, paving the way for the application of Theorem 1 to such distributions.

## 1.1 Determinantal distributions

In this work, we consider applications of Theorem 1 to various distributions  $\mu$  that are defined based on determinants. Prior progress on designing parallel sampling algorithms for problems that enjoy determinant-based counting has been very 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 algorithms for sampling spanning trees of a graph. Anari, Hu, Saberi, and Schild [Ana+20] extended this to sampling arborescences, a.k.a. directed

spanning trees, of 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:  $\mathbb{P}[Y] \propto \det(L_{Y,Y})$ .

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

**Definition 3** (Symmetric DPP). Given a symmetric  $n \times n$  matrix  $L \succeq 0$ , 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 the (symmetric) determinantal point processes defined above, our work provides sampling algorithms for a variety of discrete distributions related to determinants, which serve different roles in modeling applications. In the remainder of the section, we will outline each family of distributions.

Recently, [Bru18; Gar+19; Gar+20] initiated the study of non-symmetric DPPs in applications and argued for their use because of their increased modeling power. Non-symmetric DPPs are characterized by a non-symmetric positive-definite matrix  $L$ , i.e., a matrix  $L$  where  $L + L^\top \succeq 0$ . Symmetric DPPs necessarily exhibit strong forms of negative dependence [BBL09], which are unrealistic in some applications; non-symmetric DPPs on the other hand, can have positive correlations. As an example application, a good recommender system for online shopping should model complementary items, such as tablets and tablet pens, as having positive correlation; non-symmetric DPPs can model such positive interactions.

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

**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 on DPPs are DPPs under 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. More concretely, suppose the ground set  $[n]$  is partitioned into disjoint sets  $[n] = V_1 \cup V_2 \cup \dots \cup V_r$ , 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 is  $O(1)$  and that counting is #P-hard when the number of constraints is unbounded – it includes as a special case the problem of

computing mixed discriminants. In this paper, we will only study Partition-DPPs when the ensemble matrix  $L$  is symmetric PSD and the number of constraints is  $O(1)$ . Alimohammadi, Anari, Shiragur, and Vuong [Ali+21] showed that local Markov chains can be used to sample from these Partition-DPPs.

**Definition 7** (Partition-DPP). Given a symmetric  $n \times n$  matrix  $L \succeq 0$  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 of the 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 determinantal distributions discussed in this work was established by Alimohammadi, Anari, Shiragur, and Vuong [Ali+21] and Anari, Jain, Koehler, Pham, and Vuong [Ana+21b].

**Theorem 8** (Sampling from non-symmetric DPPs). *Let  $L$  be a  $n \times n$  non-symmetric PSD matrix,  $\epsilon \in (0, 1)$ , 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  $\epsilon$  total variation distance of  $\mu_k$  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  $\epsilon$  total variation distance of  $\mu$  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  $\epsilon$  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 are able to improve [Theorem 1](#) to obtain a parallel runtime of  $\tilde{O}(\sqrt{k})$ . Our algorithms below have a small chance  $\delta$  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 sample exactly from the desired 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  $\mu_k$  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  $\mu$  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 5](#), where we prove [Theorem 28](#).

## 1.2 Techniques and algorithms

Throughout, we heavily use the fact that for all distributions  $\mu$  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  $\mu$  are self-reducible — by conditioning on element inclusion, we obtain another distribution in the same family of DPP variants. These two properties alone are the basis of the most classical (inherently sequential) algorithm for sampling from DPPs, which we describe below.

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

Compute the marginals of  $\mu$  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**  $\{x_1, \dots, x_k\}$ .

We show that rejection sampling can be used to speed up this algorithm. Roughly speaking, we compute marginals of  $\mu$ , and sample a batch of elements  $x_1, \dots, x_\ell$  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  $\ell$ -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**  $T := \bigcup_i T_i$ .

---

Clearly, 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 DPP weight of 0. Hence, we must set the batch size  $\ell \lesssim \sqrt{k}$  to have a good acceptance probability.

A significant difficulty that arises for DPP variants beyond symmetric DPPs is the lack of negative dependence. This not only poses an analysis challenge, but presents an algorithmic difficulty as well. 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  $\simeq 2^\ell$ ; otherwise we would sometimes have to accept with probability  $> 1$ . 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 all of our models using the recently established notion of entropic independence [Ana+21b]. Intuitively, we prove that correlations in our model must be limited to small groups of elements and a batch of  $\simeq k^{\frac{1}{2}-\epsilon}$  elements will, with high probability not contain more than one element from the same group of highly correlated elements. We formalize this approach to prove Theorems 8 and 9 in Section 6. Finally, we give an example showing that this sub-polynomial overhead in the parallel depth is likely to be necessary for our batched rejection sampling approach in Section 7.

### 1.3 Further related work

The prior works of [Ten95; Ana+20] study the problems of sampling spanning trees and arborescences from graphs in parallel. Spanning trees are a special case of DPPs, but there are specialized algorithms for sampling from spanning trees and arborescences [Ald90; Bro89] that were parallelized in prior works; as far as we know the random-walk-based algorithms of Aldous [Ald90] and Broder [Bro89] have no counterpart for general DPPs.

Beyond determinant-based distributions, Feng, Hayes, and Yin [FHY21], and more recently Liu and Yin [LY21], showed how to efficiently parallelize a popular class of Metropolis Markov chains and obtain nearly optimal parallelism for several graphical models such as the hardcore and Ising models, as well as proper colorings. While their results are stated more generally for arbitrary Metropolis chains satisfying certain Lipschitz conditions on log-densities, they do not directly apply to our models. While there are efficient Markov chains for sampling from DPPs [AOR16; HS19; Ali+21], the Metropolis versions of these Markov chains (where a rejection filter is applied) do not have a nearly-linear mixing time; in fact, even for the simplest case of symmetric DPPs, they have at least a quadratic mixing time,  $O(nk)$  in the case of symmetric DPPs (this is not to be confused with non-Metropolis versions of these chains which do have linear mixing time). The results of Feng, Hayes, and Yin [FHY21] can only achieve a linear speedup, i.e., a reduction by a factor of  $n$ , for Metropolis chains, which is moot by the existence of  $\tilde{O}(k)$  parallel time sampling algorithms for  $k$ -DPPs.

### 1.4 Acknowledgements

Nima Anari and Thuy-Duong Vuong are supported by NSF CAREER Award CCF-2045354, a Sloan Research Fellowship, and a Google Faculty Research Award. Callum Burgess was supported by a Stanford CURIS Fellowship. Kevin Tian was supported by a Google Ph.D. Fellowship

and a Simons-Berkeley VMware Research Fellowship.

## 2 Overview of approach

In this section, we give a technical outline of how we prove our main results. We also provide a roadmap of the rest of the paper. All preliminaries can be found in [Section 3](#).

**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 applications do not. To prove [Theorem 10](#), we directly bound the acceptance probability of [Algorithm 1](#) for batch size  $\ell \simeq \sqrt{k}$ . Specifically, we show that for such a batch size, directly applying negative dependence bounds the acceptance probability by  $\exp(-\frac{\ell^2}{k})$ . By a simple recursion argument to bound the overall parallel depth of our sampler by  $O(\sqrt{k})$ , we obtain [Theorem 10](#).

**Bounded symmetric DPPs.** In [Section 5](#), we give improved sampling guarantees in terms of parallel depth for DPPs with kernel matrices exhibiting bounded spectral structure by proving [Theorem 28](#). The two types of spectral bounds we consider are parameterized by the kernel matrix trace and largest eigenvalue. To obtain the first result, we apply a concentration bound from [\[PP13\]](#) to show that the size of the sample concentrates tightly around its mean, which by linearity of expectation is the trace. This implies by a strategy outlined in [Remark 12](#) that with high probability, the parallel depth is bounded by a function of the trace, via our algorithm in [Theorem 10](#).

For our other result, we appeal to an alternative analysis of our rejection sampling, which uses properties of negatively-correlated distributions. We use [Algorithm 4](#), a modification of [Algorithm 1](#) based on directly manipulating the kernel matrix, as our base method. We show that if the kernel matrix is scaled down by a factor  $\alpha$  so its largest eigenvalue is bounded by  $\approx \frac{1}{\sqrt{n}}$ , we can achieve polylogarithmic parallel depth for each run of [Algorithm 4](#). We also prove a characterization of this scaling down procedure as randomly dropping elements from a sample from the original DPP, so that each element is kept with probability  $\alpha$ . By setting  $\alpha \approx \frac{1}{\lambda_{\max}(K)\sqrt{n}}$ , standard binomial concentration bounds the number of calls to this “scaled-down sampler” and yields the other half of [Theorem 28](#).

**Entropic independence.** In [Section 6](#), we provide a meta-result ([Theorem 33](#), a formal restatement of [Theorem 1](#)) used to derive [Theorems 8](#) and [9](#) as corollaries. [Theorem 33](#) shows that for *any* constant-entropically independent distribution supported on subsets of size  $k$ , we can reduce sampling to marginal computations at a parallel depth overhead of  $\tilde{O}(k^{\frac{1}{2}+c})$  with high probability. Here,  $c$  is any constant, which parameterizes the (polynomial) number of machines used.

To demonstrate [Theorem 33](#), we use the assumed entropic independence property to derive concentration bounds on the acceptance probability in [Algorithm 3](#) applied to our distribution. 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 KL divergence between our target distribution (the  $\ell$ -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 argue 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 without loss of generality that our distribution has roughly even 1-marginals, by using a subdivision process 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-acceptance probability is a submartingale, which yields concentration via Markov’s inequality.

**Hard instance.** It is natural to ask: can we improve the subpolynomial overhead in [Theorem 33](#) (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. In particular, our hard instance pairs indices in  $[n]$ , randomly chooses  $\frac{k}{2}$  of these pairs, and then includes both elements in each of the  $\frac{k}{2}$  selected pairs. This distribution is a  $k$ -nonsymmetric DPP defined by a block-diagonal matrix  $L$  composed of  $\frac{n}{2} \cdot 2 \times 2$  blocks, all equal to  $\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$ . To demonstrate hardness, we first argue that to succeed with good probability on a polynomially-bounded number of parallel machines, our rejection sampler must have the property that it is likely only a constant number of “duplicates” are encountered when drawing samples from the product distribution. We then show using a Birthday Paradox-like analysis that under this restriction, our batch size  $\ell$  must be polynomially smaller than  $\sqrt{k}$ , yielding our claim.

### 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  $\mu(\cdot \mid T)$  to be the distribution on  $2^{[n] \setminus T}$  defined by  $\mu(F \mid T) \propto \mu(F \cup T)$ . We will sometimes use the shorthand  $\mu|_T$ .

For density function  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$ , the generating polynomial of  $\mu$  is the multivariate  $k$ -homogeneous polynomial defined as follows:

$$g_\mu(z_1, \dots, z_n) = \sum_{S \in \binom{[n]}{k}} \mu(S) \prod_{i \in S} z_i.$$

#### 3.1 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  $K \preceq I$  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:  $\det(L_S) \geq 0$ . Matrices that satisfy this property are called  $P_0$ -matrices [Fan89, Definition 1]. Any nonsymmetric (or symmetric) PSD matrix is automatically a  $P_0$ -matrix [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

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

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  $c'_i = c_i - |V_i \cap Y|$ .

**Proposition 11.** *Suppose  $\mu$  is one of the following distributions.*

1.  $k$ -DPP:  $\mu(S) \propto \mathbf{1}[|Y| = k] \det(L_S)$ .
2. DPP:  $\mu(S) \propto \det(L_S)$ .
3. Partition-DPP:  $\mu_{V,c}(S) \propto \mathbf{1}[\forall i \in [r] : |S \cap V_i| = c_i] \det(L_S)$  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 computes the conditional probabilities  $\mu(T \mid S)$ .
2. Given  $S \subseteq [n]$  and integer  $t \in [n]$ , exactly computes  $\mathbb{P}_{T \sim \mu}[|T| = t]$ .

*Proof of Proposition 11.* First, note that DPPs and  $k$ -DPPs correspond to Partition-DPPs with 0 and 1 partition constraints respectively. Thus, we only need to show the claim for  $\mu$  being a Partition-DPP with  $O(1)$  constraints. Computing the marginals is equivalent to computing the partition functions of  $\mu$  and of  $\mu$  conditioned on subsets. As shown in [Cel+17, Theorem 1.1], computing the partition function is equivalent to computing the coefficients of a certain univariate polynomial which can be evaluated efficiently given access to  $g_\mu$ . Evaluating  $g_\mu$  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].  $\square$

**Remark 12.** 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  $\mu_k$  using the results of this paper.

### 3.2 Real-stable polynomials and negative correlation

In this section, we define various polynomial properties and their relationships.

**Definition 13.** Consider an open half-plane  $H_\theta = \{e^{-i\theta}z \mid \text{Im}(z) > 0\} \subseteq \mathbb{C}$ . We say a polynomial  $g(z_1, \dots, z_n) \in \mathbb{C}[z_1, \dots, z_n]$  real-stable if  $g$  has real coefficients and does not have any root in  $H_0^n$ . In particular, the zero polynomial is real-stable.

We say that distribution  $\mu : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$  is strongly Rayleigh if and only if its generating polynomial is real stable [see BBL09]. If  $\mu$  is strongly Rayleigh then for any set  $F \subseteq [n]$ , the conditional distribution  $\mu(\cdot \mid F)$  is also strongly Rayleigh. A useful property of strongly Rayleigh polynomials is that they satisfy negative correlation, defined in the following.

**Lemma 14** (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 15.** *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 16.** *Let  $K \in \mathbb{R}^{n \times n}$  satisfy  $0 \preceq K \preceq I$ . For any set  $T \subseteq [n]$ ,*

$$\det(K_T) \leq \prod_{i \in T} K_{i,i}$$

*Proof.* Consider the DPP  $\mu$  with kernel matrix  $K$ . Apply Lemma 14 and note that  $\det(K_T) = \mathbb{P}_{S \sim \mu}[T \subseteq S]$  and  $K_{i,i} = \mathbb{P}_{S \sim \mu}[i \in S]$ . □

**Lemma 17.** *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

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

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

By applying [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  $a + \bar{f} \leq 2\bar{f} \log \frac{1}{\epsilon}$ . □

### 3.3 Fractional log-concavity and entropic independence

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

**Definition 18** ([Ali+21]). A probability distribution  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$  is  $\alpha$ -fractionally-log-concave if  $g_\mu(z_1^\alpha, \dots, z_n^\alpha)$  is log-concave for  $z_1, \dots, z_n \in \mathbb{R}_{\geq 0}^n$ . If  $\alpha = 1$ , we say  $\mu$  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  $\ell$ .

**Definition 19** (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

$$D_{k \rightarrow \ell}(S, T) = \begin{cases} 0 & \text{if } T \not\subseteq S \\ \frac{1}{\binom{k}{\ell}} & \text{otherwise.} \end{cases}$$

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

**Definition 20.** A probability distribution  $\mu$  on  $\binom{[n]}{k}$  is said to be  $\frac{1}{\alpha}$ -entropically independent, for  $\alpha \in (0, 1]$ , if for all probability distributions  $\nu$  on  $\binom{[n]}{k}$ ,

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

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

**Lemma 22** ([Ali+21]). The following distributions are  $\alpha$ -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  $\mu, \nu$  over the same domain, and parameter  $C$  such that  $\max_{x \in \text{supp}(\nu)} \frac{\mu(x)}{\nu(x)} \leq C$ . Assuming sample access to  $\nu$ , we can also sample from  $\mu$  via rejection sampling as follows.

---

**Algorithm 2:** Rejection sampling

---

**Input:** parameter  $C > 0$  such that  $\max_{x \in \text{supp}(\nu)} \frac{\mu(x)}{\nu(x)} \leq C$ .

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  $\mu$ , since

$$\mathbb{P}[\text{Algorithm 2 outputs } x] \propto \frac{\mu(x)}{C\nu(x)} \nu(x) \propto \mu(x).$$

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 23.** *There is an algorithm that with probability  $1 - \delta$ , outputs a sample from  $\mu$  in the same asymptotic parallel time as required to sample from  $\nu$ , 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  $\Omega \subseteq \text{supp}(\nu) : \max_{x \in \Omega} \frac{\mu(x)}{\nu(x)} \leq C$  where  $\sum_{x \in \Omega} \mu(x) \geq 1 - \epsilon$  for  $\epsilon \in [0, 1)$ .

---

**Algorithm 3:** Modified rejection sampling

---

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

Sample  $x \sim \nu$ .

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

---

The guarantees of [Algorithm 3](#) follow immediately from [Proposition 24](#) and the fact that the restriction of  $\mu$  to  $\Omega$  has total variation at most  $\epsilon$  from  $\mu$ . In particular, we will use [Proposition 24](#) with  $\delta = \epsilon$  and output an arbitrary sample when it fails to accept a sample.

**Proposition 24.** *There is an algorithm that outputs a sample from  $\tilde{\mu}$  in the same asymptotic parallel time as required to sample from  $\nu$ , using  $O(C \text{poly}(n) \log \frac{1}{\epsilon})$  machines, where  $d_{\text{TV}}(\tilde{\mu}, \mu) = O(\epsilon)$ .*

### 3.5 Divergences

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

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

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

We remark that our definition of  $\mathcal{D}_\lambda$  is (up to a constant scalar multiplication) the exponential of the standard Renyi divergence of order  $\lambda$ . These divergences exhibit the following useful bound.

**Lemma 25.** *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  $S = [n]$*

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

More generally,

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

*Proof.* 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

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

□

## 4 Symmetric DPP

Here, we prove our basic result, [Theorem 10](#). We provide a strengthening for DPPs satisfying nontrivial spectral bounds, [Theorem 28](#), in [Section 5](#). We first state helper bounds used in the proof.

**Lemma 26.** Suppose  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$  is negatively correlated. Let  $\mu_t = \mu D_{k \rightarrow t}$  and  $p_i = \mathbb{P}_\mu[i \in S]$ . Then

$$\frac{\mu_t(T)}{t! \prod_{i \in T} \frac{p_i}{k}} \leq \exp\left(\frac{t^2}{k}\right).$$

*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,

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

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 27.** *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 15](#), when  $\mu$  is a  $k$ -DPP defined by symmetric PSD ensemble matrix  $L$ ,  $\mu$  and the conditionals of  $\mu$  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 26](#) 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  $\nu$  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  $\mu_t$  with the distribution  $\mu_t^*$  over  $[n]^t$  where  $\mu_t^*(\{i_1, \dots, i_t\}) = \frac{\mu(\{i_1, \dots, i_t\})}{t!}$ . Let  $\delta' = \frac{\delta}{2\sqrt{k}}$ , where [Algorithm 1](#) takes at most  $2\sqrt{k}$  iterations by [Proposition 27](#). We run the rejection sampling algorithm in [Proposition 23](#), which succeeds with probability  $1 - \delta'$ , to sample from  $\mu_t^*$  given samples from  $\nu$ , with  $C \leq \exp(\frac{t^2}{k}) = O(1)$  since  $t = \lceil \sqrt{k} \rceil$ . Clearly obtaining a sample from  $\mu_t^*$  yields a sample from  $\mu_t$  by forgetting the ordering on the elements.

Hence, each iteration  $i$  of [Algorithm 1](#) takes  $\tilde{O}(1)$ -parallel time. By [Proposition 27](#), the algorithm takes  $O(\sqrt{k})$ -parallel time. By a union bound, the success probability is at least  $1 - 2\sqrt{k}\delta' = 1 - \delta$ . The number of machines used is the same as in [Proposition 23](#), that is,  $O(\text{poly}(n) \log \frac{k}{\delta})$ .

The result for DPPs immediately follows from [Remark 12](#).  $\square$

## 5 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 28.** 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  $K = L(I + L)^{-1} \preceq I$  be the kernel of  $L$ . There exists an algorithm to approximately sample from within  $\epsilon$  total variation distance of  $\mu$  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 will use the following “filtered” variant of [Algorithm 1](#).

---

**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  $T_i \sim \text{DPP}$  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 3.1](#))

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

Output  $S_R$

---

We prove [Theorem 28](#) 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  $\epsilon$  of the target distribution  $\mu$ .

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

**Proposition 29.** Let  $\mu$  be a DPP with kernel  $K$ . Let  $\mu'$  be the DPP with kernel  $K' := \alpha K$ . Let  $\nu$  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,  $\mu'$  and  $\nu$  are identical.

*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

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

□

**Proposition 30.** 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 29](#). 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 have

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

where the third equality uses [Proposition 29](#) and the definition of  $L_1 = ((1 - \alpha)L)^{S_0}$  to derive

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

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

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

and hence,

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

□

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

**Lemma 31.** *Let  $\mu : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$  be a DPP with marginal kernel  $K$ . If  $\lambda_{\max}(K) \leq \frac{1}{\sqrt{n}}$  then we can sample from a distribution  $\epsilon$ -away in total variation distance from  $\mu$  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 17](#). Let  $\Omega := \{S \subseteq [n] \mid |S| \leq s\}$ . Let  $p_i := K_{i,i} = \mathbb{P}_{S \sim \mu}[i \in S]$ . Let  $\nu$  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 17](#),  $\sum_{S \in \Omega} \mu(S) \geq 1 - \epsilon'$ . Moreover, for fixed  $T \in \Omega$ , we have

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

where we use  $I + L = (I - K)^{-1}$ . By applying [Corollary 16](#) 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 24](#). 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 16](#). Thus by [Proposition 24](#), we can sample from  $\tilde{\mu}$  that is  $\epsilon$ -away from  $\mu$  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  $\epsilon$  in this proof by a constant.  $\square$

**Proposition 32.** *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  $\epsilon$ .*

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

Let  $\Lambda$  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 31](#) as the eigenvalue bound is satisfied.  $\square$

Finally, we are ready to prove [Theorem 28](#).

*Proof of Theorem 28.* The bound involving  $\text{Tr}(K)$  follows from a similar argument as in [Remark 12](#). Note that  $\text{Tr}(K) = \mathbb{E}_{S \sim \mu}[|S|]$ , and that by [Lemma 17](#), 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  $\mu_k$ , else we output an arbitrary subset. By the triangle inequality, the output's distribution is within  $\frac{\epsilon}{2} + \frac{\epsilon}{2} = \epsilon$  of  $\mu$ . 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 31](#) 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 30](#) 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  $i^{\text{th}}$  step of the for loop. By [Proposition 32](#), we can modify step (2) to sample from  $\hat{\nu}^{(i)}$  that is  $\epsilon'$ -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 30](#) yields the conclusion.  $\square$

## 6 Entropic independence

In this section, we prove the following main result. We will use [Theorem 33](#) to derive our samplers for various entropically independent distributions, namely [Theorems 8](#) and [9](#), which immediately follow from combining [Remark 12](#), [Lemma 21](#), [Lemma 22](#), and [Theorem 33](#).

**Theorem 33.** *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  $\mathbb{P}_{\mu}[i \mid S]$  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 that uses  $\mathcal{O}$  to sample from a distribution within total variation distance  $\epsilon$  of  $\mu$  in*

$$\tilde{O} \left( \sqrt{k} \cdot \left( \frac{k}{\epsilon} \right)^c \right)$$

*parallel time using  $(\frac{n}{\epsilon})^{O(c-1)}$  machines.*

### 6.1 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 34.** 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  $\mu$  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  $\mu^{\text{iso}} : \binom{U}{k} \rightarrow \mathbb{R}_{\geq 0}$ :

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

We call  $\mu^{\text{iso}}$  the *isotropic transformation* of  $\mu$ .



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

**Proposition 35** ([Ana+21a, Proposition 19]). *If  $\mu$  is  $\frac{1}{\alpha}$ -entropically-independent, then so is  $\mu^{\text{iso}}$ .*

The following generalizes [Ana+21a, Proposition 24], which summarizes useful properties of  $\mu^{\text{iso}}$ .

**Proposition 36.** *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 34 for some  $\beta$ . Let  $C = 1 + \sqrt{\beta}$ . The following hold for  $\mu^{\text{iso}}$ .*

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  $\ell \leq k$*

$$\sum_{S \in \binom{R}{\ell}} \mu^{\text{iso}}_S(S) \geq 1 - \sqrt{\beta}\ell.$$

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

*Proof.* First, we check the cardinality of the new ground set  $U$ :

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

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 the sampling from  $\mu^{\text{iso}}$  as first sampling from  $\mu$ , and then choosing a copy  $j \in [t_i]$  for each element. This yields

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

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

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

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

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

□

The following is a simple consequence of the data processing inequality.

*Remark 37.* For any  $\ell \in [k]$ , suppose algorithm  $\mathcal{A}$  can sample from within total variation distance  $\epsilon$  of  $\mu_\ell^{\text{iso}}$ . Then  $\mathcal{A}$  can also be used to sample from within total variation distance  $\epsilon$  of  $\mu_\ell$  using the same amount of (parallel) time and machines.

## 6.2 KL divergence bound

Throughout this section and [Section 6.3](#), let  $\ell \in [k]$ . We begin by proving a bound on the KL divergence between conditional marginals from an observed set.

**Lemma 38.** *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  $\mu_{t+1|S} := \mu(\cdot \mid S) D_{(k-t) \rightarrow 1}$ . Then,*

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

*Proof.* 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

$$\begin{aligned} A_S &= \sum_{i \notin S} \frac{q_i}{k-t} \log \left( \frac{q_i}{p_i} \cdot \frac{k}{k-t} \right) \\ &= \frac{k}{k-t} \sum_{i \notin S} \frac{q_i}{k} \log \left( \frac{q_i}{p_i} \right) + \log \frac{k}{k-t} \\ &\leq \frac{k}{k-t} \sum_{i \notin S} \frac{q_i}{k} \log \left( \frac{q_i}{p_i} \right) + \frac{2t}{k}. \end{aligned} \quad (4)$$

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

$$A_S \leq 2 \sum_{i \notin S} \frac{q_i}{k} \log \left( \frac{q_i}{p_i} \right) + \frac{2t}{k} \leq 2B_S + \frac{2t}{k}, \quad (5)$$

since  $\log \frac{1}{p_i} \geq 0$  for all  $i \in [n]$ . We next give an interpretation of the quantity  $B_S$ . Let  $\mu_S$  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  $\mathcal{D}_{KL}(\mu_S D_{k \rightarrow 1} \| \mu D_{k \rightarrow 1})$  (since  $\mu D_{k \rightarrow 1} = \frac{1}{k}p$ ), which we can control by entropic independence of  $\mu$ . In particular,

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

Combining the above display with (5) completes the proof.  $\square$

By averaging [Lemma 38](#) over  $\mu_S$  (the conditional distribution of  $T \sim \mu$  on  $S \subset T$ ), we immediately obtain the following corollary.

**Corollary 39.** *Let  $t \leq \frac{1}{2}k$ . Then following the notation of [Lemma 38](#),*

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

*Proof.* It suffices to apply [Lemma 38](#), 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 39](#) to derive a KL divergence bound between the distributions  $\mu_\ell$  and  $\mu'_\ell$ , respectively the target and proposal distributions encountered in our rejection sampling scheme.

**Lemma 40.** *Let  $\mu'_j$  be the distribution of the set formed by  $j$  independent draws from  $\frac{1}{k}p$ . Let  $\ell \leq \frac{1}{2}k$ . Then,*

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

*Proof.* For any  $j \in [\ell]$ , following the notation of [Lemma 38](#),

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

In the first line, we used the chain rule of KL divergence, and the second line used [Corollary 39](#). Finally, the conclusion follows by telescoping the above display for  $1 \leq j \leq \ell$ .  $\square$

[Lemma 40](#) bounds the KL divergence between  $\mu_\ell$  and  $\mu'_\ell$ , which can be thought of as an average log-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  $\ell$ -marginals; however, it is only an average bound. We make this intuition rigorous in [Section 6.3](#), where we use the tools from this section to give concentration bounds on the acceptance probability of rejection sampling.

### 6.3 Concentration of acceptance probability

In this section, assume that we have already performed the transformation in [Proposition 36](#) parameterized by some  $\beta$ , 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 36](#). Let  $\nu := \nu^{\text{iso}}$  and let  $\nu'$  be defined analogously to [Section 6.2](#). Our goal is to sample from within  $\epsilon$  total variation of  $\nu_\ell$  for a suitably chosen  $\ell$ , which also implies that we can sample from within  $\epsilon$  of  $\mu_\ell$  (see [Remark 37](#)). Our algorithm will be the modified rejection sampler of [Algorithm 3](#).

To use [Algorithm 3](#) with  $P = \nu D_{k \rightarrow \ell}$  and  $Q$  the  $\ell$ -wise product distribution drawing from  $\frac{1}{k}p$ , we first define a relevant high-probability set  $\Omega$  on our state space  $\mathcal{X} := \binom{U}{k}$ . Our set  $\Omega$  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 41.** For any  $0 \leq \epsilon \leq \frac{1}{2^{|U|/\ell}}$ ,

$$\sum_{S \notin \tilde{\Omega}_\epsilon} \nu_\ell(S) \leq 2|U|\ell\epsilon.$$

*Proof.* Let  $\mathcal{C} := \bigcup_{t=1}^\ell \{T \in \binom{U}{t} \mid \nu_t(T) \leq \epsilon^t\}$ . For any  $S \notin \tilde{\Omega}_\epsilon$ , we say  $T \in \mathcal{C}$  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  $\mathcal{M} : \tilde{\Omega}_\epsilon^c \rightarrow \mathcal{C}$ . Moreover, for some  $T \in \mathcal{C}$ , let  $\tilde{\Omega}_\epsilon^c(T)$  be the set of all  $S \in \tilde{\Omega}_\epsilon^c$  such that  $\mathcal{M}(S) = T$ . Then since

$$\nu_t(T) = \sum_{S \supseteq T} \frac{1}{\binom{\ell}{t}} \nu_\ell(S) \implies \sum_{S \in \tilde{\Omega}_\epsilon^c(T)} \nu_\ell(S) \leq \binom{\ell}{t} \nu_t(T),$$

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

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

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  $\tilde{\Omega}_\varepsilon$  captures at least a  $1 - \frac{\varepsilon}{16}$  fraction of the mass of  $\binom{U}{\ell}$  according to  $\nu_\ell$ . We will also drop  $\varepsilon$  from  $\tilde{\Omega}_\varepsilon$  for simplicity.

Our next goal is to show that almost all of the sets in  $\tilde{\Omega}$  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  $t^{\text{th}}$  draw from  $\frac{1}{k}p$  in this process. For parameters  $\tau, \gamma \geq 0$  to be defined later, iteratively define the random variables:

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

with  $C = 1 + \sqrt{\beta}$  and  $R$  as defined in [Proposition 36](#). Also by [Proposition 36](#),  $p_i \leq \frac{Ck}{|U|}$  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 42.** *Let  $S_t = T$  have  $\nu_t(T) \geq \varepsilon^t$ . Assume  $\sqrt{\beta} \leq \min \left\{ \frac{1}{3\gamma}, \frac{t}{\alpha k} \log \frac{1}{\varepsilon} \right\}$ . Then,*

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

*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

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

The second-to-last inequality used [Lemma 25](#) with  $C = 1 + \sqrt{\beta}$ , and the last inequality used [Lemma 38](#), which shows that since  $\nu_t(T) \geq \varepsilon^t$ ,

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

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

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

□

Now, applying [Lemma 42](#) with the definition of  $\tilde{\Omega}' := \tilde{\Omega} \cap \left\{ S \in \binom{[k]}{\ell} \right\}$  allows us to obtain a high-probability bound on the acceptance probability of our rejection sampling scheme.

**Lemma 43.** *Let  $B \geq 1$ . For sufficiently small  $\varepsilon \in (0, 1)$ , and  $\ell \in [k]$  satisfying*

$$\frac{12\ell^2 \left( \frac{16}{\varepsilon} \right)^{\frac{3}{B}} \log \frac{1}{\varepsilon}}{\alpha k} \leq 1,$$



and supposing 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

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

We first observe that our parameter choices indeed satisfy the condition on  $\tau$  used in [Lemma 42](#):

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

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  $S_\ell$  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  $S_\ell$ , cancelling a factor of  $\ell!$  yields

$$L(S_\ell) := \frac{\nu_\ell(S_\ell)}{\nu'_\ell(S_\ell)} = \frac{\hat{\mu}_\ell(\{i_1, i_2, \dots, i_\ell\})}{\hat{\mu}'_\ell(\{i_1, i_2, \dots, i_\ell\})} = \prod_{j \in \ell} \frac{\nu_j|_{S_{j-1}=\{i_1, i_2, \dots, i_{j-1}\}}(\{i_j\})}{\frac{1}{k} p_{i_j}} \quad (6)$$

where  $\nu'_\ell$  is the distribution of the unordered set corresponding to  $\ell$  draws from  $\frac{1}{k}p$ . Next, we apply [Lemma 42](#) which yields a submartingale property on  $Y_\ell$ . 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

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

In the last two expressions,  $S_\ell$  denotes the unordered set  $\{i_1, i_2, \dots, i_\ell\}$ . The first inequality used the fact that  $Y_\ell$  is always nonnegative, and whenever  $S_\ell \in \tilde{\Omega}'$  we can apply [Lemma 42](#) 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  $\Delta_{t+1}$ , and then we can apply (6). Hence, for any  $B \geq 0$ ,

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

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

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

□

Finally, we combine [Lemma 41](#) and [Lemma 43](#) to prove our main result.

**Lemma 44.** *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 42](#) for sufficiently large  $n$ . Set  $\epsilon = \frac{\epsilon}{32|U|\ell}$ . Combining [Proposition 36](#) and [Lemma 41](#) and using a union bound, we have

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

By [Proposition 36](#),  $|U| \leq 2n\beta^{-1} = O(nk^2\epsilon^{-2})$  and

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

Thus, this setting of  $\ell$  and  $\beta$  satisfies the assumption of [Lemma 43](#). Hence, the subset  $\Omega \subset \tilde{\Omega}'$  which satisfies the conclusion of [Lemma 43](#) has measure at least  $1 - \frac{\epsilon}{4}$  according to  $\nu_\ell$ . Using [Algorithm 3](#) and [Remark 37](#), we can sample from within total variation  $\frac{\epsilon}{2}$  from  $\nu D_{k \rightarrow \ell}$  and from  $\mu D_{k \rightarrow \ell}$  in  $\tilde{O}(1)$ -time using  $O(|U|^B \log \frac{1}{\epsilon}) = O((nk^2\epsilon^{-2})^B \log \frac{1}{\epsilon})$  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.  $\square$

## 6.4 Proof of [Theorem 33](#)

In this section, we combine the isotropic transformation of [Section 6.1](#), the parallel sampler of [Lemma 44](#), and the recursive strategy of [Proposition 27](#) to prove [Theorem 33](#).

*Proof of [Theorem 33](#).* 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 27](#), it suffices to repeatedly sample from  $\mu D_{k \rightarrow \ell}$  for some choice of  $\ell$  respecting the bound in [Lemma 44](#), within total variation  $\epsilon'$ . We then condition on this set, and then repeat. By the coupling characterization of total variation, the resulting distribution will be at total variation  $\epsilon$  from  $\mu$ , since this process will clearly terminate within  $k$  rounds. It is straightforward to see this will terminate in

$$O\left(\sqrt{\frac{k}{\frac{\alpha}{\log \frac{n}{\epsilon'}} \cdot \epsilon'^{\frac{3}{B}}}}\right) \text{ iterations}$$

by a variation of the proof of [Proposition 27](#) and the maximum allowable  $\ell$  in [Lemma 44](#). Setting  $B = \frac{3}{c}$  gives the desired bound on the number of iterations.

□

## 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 33](#) may be inherent to our rejection sampling strategy. In particular, it is natural to hope that we can improve [Theorem 33](#) 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 which 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  $\mu$  is uniformly supported on sets of the form

$$S := \bigcup_{i \in S'} S_i, \text{ where } S' \in \left( \left[ \frac{n}{2} \right] \right). \quad (7)$$

In other words,  $\mu$  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  $\mu$  is  $\Omega(1)$ -FLC (see [\[Ana+21a\]](#)). 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 24](#), if we are willing to tolerate a total variation distance of  $\delta$  from  $\mu_\ell$ , we need to show that with probability at least  $1 - \delta$ ,  $S \sim \mu_\ell$  satisfies

$$\frac{\mu_\ell(S)}{\mu'_\ell(S)} \leq n^B. \quad (8)$$

Here and throughout the following discussion,  $\mu'_\ell(S) = \frac{\ell!}{n^\ell}$  is the probability  $S$  is formed by  $\ell$  independent draws from the uniform distribution on  $[n]$ . In particular, clearly the 1-marginal distribution of  $\mu$  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 (8), it cannot contain more than  $t = O(B)$  duplicates. We then show that this limitation, along with attaining a failure probability  $\delta$  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  $\mu_\ell$  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 (8), it suffices to fix a permutation  $T_\ell$  of  $S_\ell = S$  and bound the ratios of the probabilities  $T_\ell$  is observed according to each of the above processes. Clearly it is observed with probability  $n^{-\ell}$  according to the first process above, so satisfying (8) means the probability  $T_\ell$  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  $\Theta\left(\frac{1}{k}\right)$ . On the other hand, the probability of observing each singleton in its round is  $\Theta\left(\frac{1}{n}\right)$ . For  $k = o(n)$ , this shows that to meet (8) 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  $\mu$  is uniform over all sets of the form (7), 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 (7), and bound the probability that this downsampling process results in a subset with  $t$  duplicates. By symmetry of  $\mu$ , 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  $\ell$  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  $\mu_\ell$  has exactly  $t$  duplicates for constant  $t$  scales as

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

In other words, to guarantee that a draw from  $\mu_\ell$  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  $\delta$  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.

## 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).
- [AOR16] Nima Anari, Shayan Oveis Gharan, and Alireza Rezaei. “Monte Carlo Markov chain algorithms for sampling strongly Rayleigh distributions and determinantal point processes”. In: *Conference on Learning Theory*. PMLR. 2016, pp. 103–115.
- [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.
- [Fan89] Li Fang. “On the spectra of P- and P0-matrices”. In: *Linear Algebra and its Applications* 119 (1989), pp. 1–25. ISSN: 0024-3795. DOI: [https://doi.org/10.1016/0024-3795\(89\)90065-7](https://doi.org/10.1016/0024-3795(89)90065-7).
- [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].
- [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.
- [HS19] Jonathan Hermon and Justin Salez. “Modified log-Sobolev inequalities for strong-Rayleigh measures”. In: *arXiv preprint arXiv:1902.02775* (2019).
- [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.
- [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.