

# Log-Concave Polynomials IV: Approximate Exchange, Tight Mixing Times, and Near-Optimal Sampling of Forests

Nima Anari<sup>1</sup>, Kuikui Liu<sup>2</sup>, Shayan Oveis Gharan<sup>2</sup>, Cynthia Vinzant<sup>3</sup>, and Thuy-Duong Vuong<sup>1</sup>

<sup>1</sup>Stanford University, [anari@cs.stanford.edu](mailto:anari@cs.stanford.edu), [tdvuong@stanford.edu](mailto:tdvuong@stanford.edu)

<sup>2</sup>University of Washington, [liukui17@cs.washington.edu](mailto:liukui17@cs.washington.edu), [shayan@cs.washington.edu](mailto:shayan@cs.washington.edu)

<sup>3</sup>North Carolina State University, [cvinzan@ncsu.edu](mailto:cvinzan@ncsu.edu)

## Abstract

We prove tight mixing time bounds for natural random walks on bases of matroids, determinantal distributions, and more generally distributions associated with log-concave polynomials. For a matroid of rank  $k$  on a ground set of  $n$  elements, or more generally distributions associated with log-concave polynomials of homogeneous degree  $k$  on  $n$  variables, we show that the down-up random walk, started from an arbitrary point in the support, mixes in time  $O(k \log k)$ . Our bound has no dependence on  $n$  or the starting point, unlike the previous analyses of Anari et al. (STOC 2019), Cryan et al. (FOCS 2019), and is tight up to constant factors. The main new ingredient is a property we call approximate exchange, a generalization of well-studied exchange properties for matroids and valuated matroids, which may be of independent interest. In particular, given a distribution  $\mu$  over size- $k$  subsets of  $[n]$ , our approximate exchange property implies that a simple local search algorithm gives a  $k^{O(k)}$ -approximation of  $\max_S \mu(S)$  when  $\mu$  is generated by a log-concave polynomial, and that greedy gives the same approximation ratio when  $\mu$  is strongly Rayleigh.

As an application, we show how to leverage down-up random walks to approximately sample random forests or random spanning trees in a graph with  $n$  edges in time  $O(n \log^2 n)$ . The best known result for sampling random forest was a FPAUS with high polynomial runtime recently found by Anari et al. (STOC 2019), Cryan et al. (FOCS 2019). For spanning tree, we improve on the almost-linear time algorithm by Schild (STOC 2018). Our analysis works on weighted graphs too, and is the first to achieve nearly-linear running time for these problems. Our algorithms can be naturally extended to support approximately sampling from random forests of size between  $k_1$  and  $k_2$  in time  $O(n \log^2 n)$ , for fixed parameters  $k_1, k_2$ .

## 1 Introduction

Let  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$  be a density function on size  $k$  subsets of  $[n] = \{1, \dots, n\}$ , defining a distribution  $\mathbb{P}[S] \propto \mu(S)$ . The generating polynomial of  $\mu$  is the multivariate  $k$ -homogeneous polynomial defined as follows:

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

We say that  $g_\mu$  is log-concave if  $\log(g_\mu)$  is a concave function over  $\mathbb{R}_{\geq 0}^n$ . The study of log-concave polynomials has recently enabled breakthroughs on old conjectures about matroids, including the resolution of a conjecture of Mihail and Vazirani [MV89] on the expansion of the bases-exchange graphs [Ana+19], and Mason's ultra-log-concavity conjecture [Ana+18; BH18]. These results rely on the log-concavity of the generating polynomial for various distributions associated with matroids, most importantly the uniform distribution on the set of bases [AOV18].

Besides distributions associated with matroids, several other classes of distributions possess a log-concave generating polynomial. An important subclass consists of strongly Rayleigh distributions [BBL09] which includes determinantal point processes, distributions that have found numerous applications in machine learning [see KT12, for a survey]. A well-studied example belonging to all classes mentioned so far consists

of the uniform distribution over spanning trees of a graph  $G = (V, E)$ . Here  $n$  is the number of edges  $|E|$  in the graph and  $k$  is the number of edges in a spanning tree, i.e.,  $|V| - 1$ . Spanning trees of a graph form bases of a matroid called the graphic matroid [see, e.g., [Ox106](#)] and they can also be viewed as a determinantal point process because of the matrix-tree theorem [see, e.g., [BBL09](#)], and are consequently strongly Rayleigh.

The motivation behind the conjecture of Mihail and Vazirani [[MV89](#)] was to solve the problem of approximately sampling from bases of a matroid. After this conjecture was made, efficient sampling algorithms were developed for various special classes of matroids [[FM92](#); [Gam99](#); [JS02](#); [Jer+04](#); [Jer06](#); [Clo10](#); [CTY15](#); [GJ18](#)] until Anari, Liu, Oveis Gharan, and Vinzant [[Ana+19](#)] showed an efficient approximate sampling algorithm for all matroids. This algorithm used a variant of random walks on the so-called “bases-exchange” graphs of matroids, that is known as the “down-up” random walk studied in the context of high-dimensional expanders [[KM16](#); [DK17](#); [KO20](#)]. For a distribution defined by  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$ , the down-up random walk  $P$  starts from a set  $S_0 \in \binom{[n]}{k}$  and produces the Markovian sequence  $S_0, S_1, S_2, \dots$  as follows:

---

**Algorithm 1:** Down-up Walk

---

**for**  $t = 0, 1, 2, \dots$  **do**

  Let  $T_t \in \binom{S_t}{k-1}$  be a subset of  $S_t$  obtained by dropping one element of  $S_t$  uniformly at random.

  Let  $S_{t+1} = T_t \cup \{e\}$ , where the element  $e$  is chosen with probability  $\propto \mu(T_t \cup \{e\})$ .

---

The random walk  $P$  has  $\mu$  as its stationary distribution and can be efficiently implemented by probing  $\mu$  on at most  $n$  different sets each time. Thus assuming oracle access to  $\mu$ , or in the case of matroids, an independence oracle for the matroid, each step of  $P$  takes  $O(n)$  time. The challenging part has been establishing the mixing time of  $P$ , i.e., bounds on the time  $t$  such that the distribution of  $S_t$  is  $\epsilon$ -close in total variation distance to the one defined by  $\mu$ :

$$t_{\text{mix}}(P, S_0, \epsilon) := \min \{t \mid \|P^t(S_0, \cdot) - \mu(\cdot)\|_{\text{TV}} \leq \epsilon\}.$$

Anari, Liu, Oveis Gharan, and Vinzant [[Ana+19](#)] proved that when  $\mu$  has a log-concave generating polynomial, the spectral gap of the random walk  $P$  is at least  $1/k$ . This implied that

$$t_{\text{mix}}(P, S_0, \epsilon) \leq O \left( k \cdot \left( \log \frac{1}{\mathbb{P}_\mu[S_0]} + \log \frac{1}{\epsilon} \right) \right).$$

Later, Cryan, Guo, and Mousa [[CGM19](#)] proved a Modified Log-Sobolev Inequality (MLSI) for the same random walk which resulted in a tighter mixing time:

$$t_{\text{mix}}(P, S_0, \epsilon) \leq O \left( k \cdot \left( \log \log \frac{1}{\mathbb{P}_\mu[S_0]} + \log \frac{1}{\epsilon} \right) \right).$$

These results lead to efficient algorithms assuming that the mass of the starting set,  $\mathbb{P}_\mu[S_0]$ , is not terribly small; this can often be achieved in practice. For example, for matroids, any starting basis  $S_0$  will satisfy  $\mathbb{P}_\mu[S_0] \geq 1/\binom{n}{k} \geq n^{-k}$ , because the number of bases is at most  $\binom{n}{k}$ . Consequently the above bounds turn into  $t_{\text{mix}}(P, S_0, \epsilon) \leq O(k(k \log(n) + \log(1/\epsilon)))$  and  $t_{\text{mix}}(P, S_0, \epsilon) \leq O(k(\log k + \log \log n + \log(1/\epsilon)))$  respectively. However, for other distributions  $\mu$  with a log-concave generating polynomial, even in the very special case of determinantal point processes, there is no control on  $\min \{\mathbb{P}_\mu[S_0] \mid S_0 \in \text{supp}(\mu)\}$ , so one has to rely on clever tricks to find a good starting set  $S_0$ ; even then, the best hope is to find a set  $S_0$  with  $\mathbb{P}_\mu[S_0] \gtrsim 1/\binom{n}{k}$ , which results in a mixing time mildly depending on  $n$ .

Historically, earlier works on a subclass of matroids, called balanced matroids, followed a similar development, where initially a spectral gap result was proved, resulting in a running time<sup>1</sup> of  $O(nk(k \log n +$

<sup>1</sup>Note that the running time is  $n$  times the mixing time for the down-up walk.

$\log(1/\epsilon))$ ) followed by MLSI which resulted in a mixing time of  $O(k(\log k + \log \log n + \log(1/\epsilon)))$  [see [MT06](#), for a survey]. Noting that the term  $\log \log n$  seems unnecessary, Montenegro and Tetali [[MT06](#)] raised the question of proving a better inequality that would result in a running time of  $O(nk \log(k/\epsilon))$ . They specifically hoped for the possibility of proving a Nash inequality, an advanced type of functional inequality used to derive very tight mixing times for some Markov chains [[MT06](#)]. We believe there are barriers to using functional inequalities in general to prove  $O(k \log(k/\epsilon))$  mixing time for the down-up random walk; we defer an explanation of this to a future version of this paper. However, without proving new functional inequalities, we manage to sidestep this barrier and improve the running time to the conjectured  $O(nk \log(k/\epsilon))$  for not just balanced matroids, but the class of all matroids.

Our first result is a tight analysis of the mixing time, entirely removing the dependence on  $\mathbb{P}_\mu[S_0]$  and  $n$ .

**Theorem 1.** *For any distribution defined by  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$  with a log-concave generating polynomial  $g_\mu$ , the mixing time of the down-up random walk  $P$ , starting from any  $S_0$  in the support of  $\mu$  is*

$$t_{\text{mix}}(P, S_0, \epsilon) \leq O(k \log(k/\epsilon)).$$

Note that generally we cannot hope for a better mixing time than  $k \log k$ ; each step of the random walk  $P$  replaces one element of the current set, and by a coupon collector argument, at least  $\simeq k \log k$  steps are needed to replace every element of the starting set  $S_0$ . As long as  $k$  is not too close to  $n$ , say  $k < 0.99n$ , replacing every starting element is needed for sufficient mixing, even for the simple distribution  $\mu$  which is uniform over  $\binom{[n]}{k}$ .

Our mixing time bound is an asymptotic improvement over prior work for  $k = O(1)$ , or more generally when  $k$  is smaller than  $\log(n)^\epsilon$  for all  $\epsilon > 0$ . Another consequence of the new mixing time bound is that it enables the analysis of the down-up random walk when  $n$  is infinitely large; for example, this is the case for continuous determinantal point processes [see, e.g., [OR18](#)].<sup>2</sup> To avoid complicating the notation, we do not consider infinitely large ground sets in this paper, but note that the results do generalize to such cases.

Our next result is the first quasi-linear time algorithm to sample from the uniform distribution over forests of a graph  $G = G(V, E)$ . This improves upon the recent result by Anari, Liu, Oveis Gharan, and Vinzant [[Ana+19](#)] which gives a polynomial time algorithm to sample random forest, but the run-time of this algorithm is far from being linear in the number of edges. Their algorithm samples and counts forests of fixed-size  $k$  for each  $k \leq |V| - 1$ , thus takes at least  $\Omega(|V| |E|)$  time. Moreover, they employ the approximate sampling to approximate counting reduction [[JV86](#); [AD20](#)], which introduces large polynomial blow-up in run-time. For applications of sampling random forests, see e.g. [[Goe+](#)].

In addition, we show a similar algorithm that also runs in quasi-linear time and samples from the uniform distribution over spanning trees of  $G$ . Much attention has been paid to the problem of sampling random spanning trees over the years, starting from the seminal works of Aldous [[Ald90](#)] and Broder [[Bro89](#)] who proposed a simple routine to extract a random spanning tree from the trace of a random walk on  $G$  itself. Subsequent works introduced improved algorithms [[Wil96](#); [CMN96](#); [KM09](#); [MST14](#); [Dur+17a](#); [Dur+17b](#)] until finally Schild [[Sch18](#)] managed to obtain an almost-linear time algorithm running in time  $n^{1+o(1)}$  on graphs with  $n$  edges. This algorithm and that of several prior works were all based on the original work of Aldous [[Ald90](#)] and Broder [[Bro89](#)]; they achieved an improved running time by employing several clever, but complicated, tricks to shortcut the trace of a random walk over  $G$ . Our algorithms to sample a random spanning tree or random forest is wholly different, based on the down-up random walk, that achieves a nearly-linear running time of  $n \log^2(n)$ , while being arguably much simpler to describe and implement. Our algorithms can be naturally extended to sample from *weighted* distributions over forests or spanning trees as well.

**Theorem 2.** *There is an algorithm that takes a weighted graph  $G = G(V, E)$  on  $n$  edges with weight function  $w : E \rightarrow \mathbb{R}_{\geq 0}$ , parameters  $q \geq 0$  and  $\epsilon > 0$  as input and outputs a forest  $F \subseteq E$  in time  $O(n \log(n) \log(n/\epsilon))$ ; the*

<sup>2</sup>We note however that one still needs to be able to implement each step of the random walk efficiently when  $n$  is infinitely large. For examples where this is possible see [[OR18](#)].

distribution of  $F$  is guaranteed to be  $\epsilon$ -close in total variation distance to the distribution  $\mu$  over forests of  $G$  defined by

$$\mu(F) \propto q^{k-|F|} w^F$$

where  $k$  is the rank of the graphic matroid of  $G$ , and  $|F|$  denotes the number of edges in  $F$ .

In particular, when  $w(e) = 1$  for all  $e$ ,  $\mu$  is the uniform distribution on forests of  $G$  if  $q = 1$ , and is the uniform distribution on spanning trees of  $G$  if  $G$  is connected and  $q = 0$ .

In fact, we can extend [Theorem 2](#) to allow sampling from the uniform distribution over forests of size between  $k_1$  and  $k_2$ , for any parameters  $k_1, k_2$ , in quasi-linear time.

**Theorem 3.** *There is an algorithm that takes a weighted graph  $G = G(V, E)$  on  $n$  edges with weight function  $w : E \rightarrow \mathbb{R}_{\geq 0}$ , parameters  $q \geq 0$ ,  $k_1, k_2 \in \mathbb{N}$  and  $\epsilon > 0$  as input and outputs a forest  $F \subseteq E$  in time  $O(n \log(n) \log(n/\epsilon))$ ; the distribution of  $F$  is guaranteed to be  $\epsilon$ -close in total variation distance to the distribution  $\mu^{(k_1, k_2)}$  over forests of  $G$  defined by*

$$\mu^{(k_1, k_2)}(F) \propto q^{k_2-|F|} w^F$$

restricted to forests of size  $|F| \in [k_1, k_2]$ .

Since our algorithms are based on the MCMC method, they can only approximately sample from the forest or spanning tree distribution. In contrast, some of the prior works, including [[Sch18](#)], can sample exactly from the spanning tree distribution. This is mostly an inconsequential difference in practice, as no polynomial-time user of the algorithm can sense a difference between exact sampling and approximate sampling; one simply needs to set  $\epsilon$  to be inverse-polynomially small.

We remark that our technique also leads to algorithm(s) that perform the more general task of approximately sampling from the uniform distribution over the family of independent sets of an arbitrary matroid, given access to suitable oracles. Specifically, for a matroid  $\mathcal{M} = ([n], \mathcal{I})$  of rank  $k$ , an algorithm similar to the one from [Theorem 2](#) samples from a distribution that is  $\epsilon$ -close to the uniform distribution over the independent set  $\mathcal{I}$  using  $O(n \log \frac{n}{\epsilon})$  calls to a data structure  $\mathcal{O}'$  that maintains a set  $S \subseteq [n]$ , guaranteed to contain at most one circuit and allows for

- Addition, and removal of an element from  $S$ , provided we maintain the property that  $S$  contains at most one circuit.
- Outputting a uniformly random element from the unique circuit in  $S$  if, such a circuit exists.

For graphic matroids, we can implement  $\mathcal{O}'$  with amortized quasi-constant query time using link-cut trees. In general, since the input  $S$  is guaranteed to have size at most  $k + 1$ , we can implement each call to  $\mathcal{O}'$  using  $O(k)$  calls to the more familiar *independence oracle*  $\mathcal{O}_I$  for  $\mathcal{M}$ , resulting in a  $O(kn \log \frac{n}{\epsilon})$ -time algorithm.

## 1.1 Techniques

In order to prove [Theorem 1](#), our strategy is to combine a new analysis of the initial steps of the down-up random walk with the previously known Modified Log-Sobolev Inequality [[CGM19](#)]. Specifically we show that conditioned on having replaced every element of the starting set  $S_0$  at least once by time  $t$ , the set at time  $t$  can be used as a *warm start* for the rest of the steps. Specifically, we show that the density of the set at time  $t$  w.r.t.  $\mu$ , conditioned on this event, is upper-bounded by only a function of  $k$ .

In order to prove this, we introduce a new property of functions  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$  that we call  $\alpha$ -approximate exchange. This property says that for every  $S, T \in \binom{[n]}{k}$ , and  $i \in S$ , there exists  $j \in T$  such that

$$\mu(S)\mu(T) \leq \alpha \cdot \mu(S - i + j)\mu(T + i - j).$$

Note that when  $\mu$  takes values in  $\{0, 1\}$  and  $\alpha \geq 1$ , this property becomes equivalent to the famous strong basis exchange axiom of matroids [[Oxl06](#)]; if  $\mathcal{B} = \mu^{-1}(1)$  is the family of sets indicated by  $\mu$ , this property says that for every  $S, T \in \mathcal{B}$  and  $i \in S$ , there exists  $j \in T$  such that  $S - i + j \in \mathcal{B}$  and  $T + i - j \in \mathcal{B}$ . This

property can be seen as a quantitative variant of strong basis exchange. Alternatively, it can be viewed as an approximate and multiplicative form of  $M^\natural$ -concavity, a cornerstone of discrete convex analysis [MS99]. We prove that every  $\mu$  with a log-concave generating polynomial satisfies  $2^{O(k)}$ -approximate exchange. Crucially, our  $\alpha$  does not depend on  $n$ . We remark that Brändén and Huh [BH18] showed a result that can be thought of as a partial converse to this. They proved that  $M^\natural$ -concavity of  $\log \mu$ , equivalent to 1-approximate exchange property, implies that the generating polynomial of  $\mu$  is log-concave.

We show that a similar approximate exchange property implies that a simple local search algorithm gives a  $k^{2k}$ -approximation on the problem of maximizing  $\mu(S)$  for  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$  generated by a log-concave polynomial. If the generating polynomial of  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$  is moreover strongly Rayleigh, then  $\mu$  satisfies a slightly stronger exchange property (see [Lemma 24](#)) that in turn implies greedy gives a  $k^{2k}$ -approximation of  $\max_S \mu(S)$  (see [Lemma 25](#)). This is a generalization of Khachiyan [Kha95]’s classical result that greedy produces a  $k^{O(k)}$ -approximation of the (sub)determinant maximization problem [Kha95; Di +14; Nik15], as well as [Pac04; CM10]’s more recent result that greedy gives  $k^{O(k)}$ -approximation for the largest  $j$ -dimensional simplex problem. The best result on the largest  $j$ -dimensional simplex problem is a  $2^{O(k)}$ -approximation by Nikolov [Nik15], matching the lower bound given by [Di +14; CM10].

We discuss the high-level ideas for proving [Theorem 2](#). For the sake of simplicity, we consider the unweighted case, i.e.,  $w(e) = 1$  for all  $e \in E$ . It would be illuminating to first discuss the special case where  $q = 0$ ,  $G$  is connected, and  $\mu$  is uniformly distributed over spanning trees of  $G$ . We would like to use the down-up random walk from [Theorem 1](#) to sample from  $\mu$ . Though the down-up walk on the support of  $\mu$  mixes in nearly-linear time, we do not see a way to implement each step of it in polylogarithmic time. Fortunately, the down-up random walk on an equivalent family of sets, the *dual* of the graphic matroid of  $G$ , which consists of the *complements* of spanning trees, also mixes fast, and we can implement each step in amortized  $O(\log n)$ -time using the link-cut tree data structure [ST83; RTF18].

For  $q \neq 0$ , the distribution  $\mu$  over forests of  $G$  is not *homogeneous*, i.e., the support of  $\mu$  contains different-size subsets of  $E$ , so we cannot immediately apply [Theorem 1](#). Let  $\bar{\mu}$  be the *complement* distribution of  $\mu$  i.e.  $\bar{\mu}(E \setminus F) = \mu(F)$  if  $F$  is a forest, then sampling from  $\mu$  and from  $\bar{\mu}$  are equivalent. We add auxiliary elements to each  $\bar{F} \in \text{supp}(\bar{\mu})$  to obtain a homogeneous distribution. More precisely, we design a homogeneous distribution  $\mu^\uparrow : \binom{E \cup Y}{n} \rightarrow \mathbb{R}_{\geq 0}$  whose projection to  $E$  is  $\bar{\mu}$  i.e.  $\mathbb{P}_{T \sim \mu^\uparrow}[T \cap E] = \bar{\mu}(T \cap E)$  where  $Y$  is the set of auxiliary elements, such that the generating polynomial of  $\mu^\uparrow$  is log-concave. Specifically, in [Lemma 26](#), we prove, using results of Brändén and Huh [BH19], that for any matroid  $\mathcal{M}$  of rank  $r$  over ground set  $[n]$ , the polynomial  $f_{\mathcal{M}}(z_0, z_1, \dots, z_n) = \sum_{S \in I(\mathcal{M})} z_0^{|S|} z^{[n] \setminus S}$  is strongly log-concave, and then use *polarization* (see [Proposition 16](#)) to transform  $f_{\mathcal{M}}$  into a multi-affine homogeneous log-concave polynomial

$$f_{\mathcal{M}}^\uparrow(y_1, \dots, y_r, z_1, \dots, z_n) = \sum_{S \in I(\mathcal{M}), T \in \binom{[r]}{|S|}} \frac{1}{\binom{r}{|S|}} y^T z^{[n] \setminus S}$$

The fact that  $f_{\mathcal{M}}$  is strongly log-concave could be of independent interest.

The distribution  $\mu^\uparrow$  is generated by  $f_{\mathcal{M}}^\uparrow$ . Our algorithm runs the down-up random walk on  $\mu^\uparrow$ , which mixes rapidly by [Theorem 1](#), and then outputs  $E \setminus (T_t \cap E)$  where  $T_t \in \text{supp}(\mu^\uparrow)$  is the random set we obtained after  $t = O(n \log \frac{n}{\epsilon})$  down-up steps. Each step of the walk, even in the weighted case, can again be implemented in amortized  $O(\log n)$ -time using link-cut trees [ST83; RTF18].

If we only consider the effect of the down-up walk on  $\bar{T}_{t,E} := E \setminus (T_t \cap E)$ , then each step of the down-up walk can be viewed as follows:

- With probability  $1 - \frac{|\bar{T}_{t,E}|}{n}$ , sample an edge  $e \notin \bar{T}_{t,E}$  uniformly at random and add  $e$  to  $\bar{T}_{t,E}$ .
- If there is a cycle formed in  $\bar{T}_{t,E}$  by the previous operation, remove an edge uniformly at random from the cycle. Else, with probability  $\frac{q}{1+q}$ , remove an edge uniformly at random from  $\bar{T}_{t,E}$ . Note that this has no effects if  $\bar{T}_{t,E}$  is already empty.

Observe that if  $q = 0$ , we never remove an edge from  $\overline{T}_{t,E}$  unless  $\overline{T}_{t,E}$  contains a cycle, thus if  $\overline{T}_{0,E}$  is a spanning tree then so is  $\overline{T}_{t,E}$  for all  $t$ . For  $q = 0$ , our algorithm (to sample random spanning trees) is the same as the one proposed by Russo, Teixeira, and Francisco [RTF18]. Despite not having the tight mixing time analysis, they empirically observed fast mixing times for the proposed algorithm, and additionally showed how link-cut trees can be used to implement each step.

To prove [Theorem 3](#), we only need to show that the following polynomial is strongly log-concave:

$$f_{\mathcal{M}}^{k_1}(z_0, z_1, \dots, z_n) = \sum_{S \in \mathcal{I}(\mathcal{M}), |S| \geq k_1} z_0^{|S|} z^{[n] \setminus S}.$$

Then, we apply this for the matroid  $\mathcal{M}$  whose bases are the size- $k_2$  forests of graph  $G$ . Next, we employ the *polarization* trick then running down-up walk framework which we use to prove [Theorem 2](#).

## 1.2 Structure of the Paper

In [Section 2](#) we provide some background on Markov chains and geometry of polynomials. In [Section 3](#) we prove [Theorem 1](#). In [Section 4](#) we prove certain approximate exchange properties, and their algorithmic implications. In [Section 5](#) we prove [Theorems 2](#) and [3](#), as well as other results on sampling independent sets of an arbitrary matroid. The results in [Section 5](#) can be read independently of [Sections 3](#) and [4](#).

## 2 Preliminaries

We use  $[n]$  to denote the set  $\{1, \dots, n\}$  and  $\binom{[n]}{k}$  to denote the family of size  $k$  subsets of  $[n]$ . When  $n$  is clear from context, we use  $\mathbb{1}_S \in \mathbb{R}^n$  to denote the indicator vector of the set  $S \subseteq [n]$ , having a coordinate of 0 everywhere except for elements of  $S$ , where the coordinate is 1. We use  $\text{conv}$  to denote the operator that maps a set of points to their convex hull.

We use  $z_S$  as shorthand for  $\{z_i \mid i \in S\}$  and  $z^S$  as shorthand for  $\prod_{i \in S} z^i$ . For polynomial  $f = \sum_S c_S z^S \in \mathbb{C}[z_0, \dots, z_n]$ , we let the support of  $f$  be  $\text{supp}(f) := \{S : c_S \neq 0\}$ , and write  $\partial_i f$  as shorthand for  $\frac{\partial f}{\partial z_i}$ .

We use  $e_k(z_1, \dots, z_n)$  to denote the  $k$ -th symmetric polynomial in  $z_1, \dots, z_n$ . We sometimes abuse notation and write  $e_k(u, z_S)$  to denote the  $k$ -th symmetric polynomial in variables  $z_S \cup \{u\}$ .

### 2.1 Matroids

In this paper we use one of the many cryptomorphic definitions of a matroid in terms of the polytope of its bases. For equivalence to other prominent definitions of a matroid, and more generally references to facts stated here see [Oxl06].

**Definition 4.** We say that a family  $\mathcal{B} \subseteq \binom{[n]}{k}$  is the family of bases of a matroid if the polytope  $\text{conv} \{ \mathbb{1}_B \mid B \in \mathcal{B} \}$  has only edges of the minimum possible length, namely  $\sqrt{2}$ . We call  $k$  the rank of the matroid, and  $[n]$  the ground set of the matroid.

We let the family of independent sets of the matroid be  $\mathcal{I} = \left\{ I \in 2^{[n]} \mid \exists B \in \mathcal{B} : I \subseteq B \right\}$

A well-known fact about matroids, that can be easily derived from [Definition 4](#), is that the dual of a matroid, defined below, is another matroid.

**Proposition 5.** If  $\mathcal{B} \subseteq \binom{[n]}{k}$  is the family of bases of a matroid, then the following is also the family of bases of another matroid, called the dual matroid:

$$\mathcal{B}^* := \{ [n] - B \mid B \in \mathcal{B} \}.$$

In this paper we will use a famous class of matroids constructed from graphs, called graphic matroids.

**Proposition 6.** Let  $G = (V, E)$  be a graph. Then the following is the family of bases of a matroid, called the graphic matroid of  $G$ :

$$\{T \subseteq E \mid T \text{ forms a spanning forest}\}.$$

Note that the rank of the graphic matroid is  $\leq |V| - 1$  and the ground set is  $E$ . If  $G$  is connected, then the bases are spanning trees of  $G$ , and the rank is exactly  $|V| - 1$ .

## 2.2 M-Convex Sets

**Definition 7** (M-convex sets). We define a subset  $J \subseteq \mathbb{N}^n$  to be M-convex if it satisfies any one of the following equivalent conditions:

- For any  $\alpha, \beta \in J$  and any index  $i$  satisfying  $\alpha_i > \beta_i$ , there is an index  $j$  satisfying  $\alpha_j < \beta_j$  and  $\alpha - e_i + e_j \in J$ .
- For any  $\alpha, \beta \in J$  and any index  $i$  satisfying  $\alpha_i > \beta_i$ , there is an index  $j$  satisfying  $\alpha_j < \beta_j$  and  $\alpha - e_i + e_j \in J$  and  $\beta - e_j + e_i \in J$ .

We note that any M-convex set  $J$  must be a subset of

$$\Delta_n^d := \{\alpha \in \mathbb{N}^n \mid |\alpha|_1 = d\}$$

for some fixed  $d$ . Conversely, for  $d = 1$ , any  $J \subseteq \Delta_n^1$  is M-convex.

## 2.3 Stable Polynomials

**Definition 8** (Half-plane stable). Consider an open half-plane

$$H_\theta = \left\{ e^{-i\theta} z \mid \operatorname{Im}(z) > 0 \right\} \subseteq \mathbb{C}.$$

We say a polynomial  $g(z_1, \dots, z_n) \in \mathbb{C}[z_1, \dots, z_n]$   $H_\theta$ -stable if  $g$  does not have root in  $H_\theta^n$ . In particular, the zero polynomial is  $H_\theta$ -stable.

We call  $H_0$  and  $H_{\pi/2}$  the upper-half and right-half plane respectively. We say  $g$  is Hurwitz stable if it is  $H_{\pi/2}$ -stable. We say  $g$  is real stable if it is  $H_0$ -stable and has real coefficients.

We observe that for homogeneous polynomials, the definition of  $H_\theta$ -stable is equivalent for all angles  $\theta$ .

A distribution  $\mu : 2^{[n]} \rightarrow \mathbb{R}_{\geq 0}$  is strongly Rayleigh if and only if its generating polynomial is real stable [see BBL09]. Real stability is preserved under differentiation, and identification:

**Theorem 9** ([Wag09, Lemma 2.4]). If  $g \in \mathbb{R}[z_1, \dots, z_n]$  is real stable, then the following are also real stable

- $g|_{z_i=a}$  for  $a \in \mathbb{R}$
- $\partial_i g$  for all  $i \in [n]$ .

We will need the following classical fact [see e.g. BBL09, Proposition 3.1 for a proof].

**Theorem 10.** For  $k \leq n$ , the  $k$ -th symmetric polynomial in  $n$  variables  $e_k(z_1, \dots, z_n)$  is real stable.

## 2.4 Log-Concave Polynomials

For a distribution or density function  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$  we denote by  $g_\mu$  the generating polynomial of  $\mu$  defined as

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

We call a polynomial  $g \in \mathbb{R}[z_1, \dots, z_n]$  with nonnegative coefficients log-concave when viewed as a function, it is log-concave over the positive orthant, i.e., for  $x, y \in \mathbb{R}_{\geq 0}^n$  and  $\lambda \in (0, 1)$

$$g(\lambda x + (1 - \lambda)y) \geq g(x)^\lambda g(y)^{1-\lambda}.$$

For a multiaffine polynomial  $g$ , its derivatives can be obtained as

$$\partial_1 g = \lim_{c \rightarrow \infty} \frac{g(c, z_2, \dots, z_n)}{c}.$$

This shows that the derivatives of a multiaffine log-concave polynomial are limits of log-concave polynomials, which themselves are log-concave. It follows that a multiaffine homogeneous log-concave polynomial satisfies the seemingly stronger notions of *strong log-concavity* [Gur09] and *complete log-concavity* [AOV18; Ana+18; BH19]. The latter means that such polynomials are also closed under directional derivatives

**Lemma 11.** *Let  $g \in \mathbb{R}[z_1, \dots, z_n]$  be a multiaffine homogeneous polynomial with nonnegative coefficients. If  $g$  is log-concave, then it is completely log-concave as well, which means that for any  $k \in \mathbb{Z}_{\geq 0}$  and directions  $v_1, \dots, v_k \in \mathbb{R}_{\geq 0}^n$ , the following polynomial is log-concave:*

$$\partial_{v_1} \cdots \partial_{v_k} g.$$

We will use an alternative characterization of strongly/completely log-concave homogeneous polynomial [AOV18], given under the name of Lorentzian polynomials by Brändén and Huh [BH19, Definition 2.6].

**Definition 12** (Strongly Log-Concave/Lorentzian polynomials). Let  $g \in \mathbb{R}[z_1, \dots, z_n]$  be a homogeneous polynomial of degree  $d$  with nonnegative coefficients. We say  $g$  is Lorentzian, or equivalently strongly log-concave, if either  $d \leq 1$ , or  $d \geq 2$ ,  $\text{supp}(g)$  is  $M$ -convex, and  $\partial^\alpha g$  is real-stable for all  $\alpha$  satisfying  $|\alpha|_1 = d - 2$ .

In particular, if  $g$  is strongly log-concave then its support  $\text{supp}(g)$  is  $M$ -convex, and all its (directional) derivatives are strongly log-concave as well. For the equivalence of strongly log-concave, completely log-concave, and Lorentzian homogeneous polynomials, see [BH19].

An important class of polynomials are those associated with uniform distributions over bases of a matroid.

**Theorem 13** ([AOV18] based on [AHK18]). *If  $\mathcal{B} \subseteq \binom{[n]}{k}$  is the family of bases of a matroid, then the following polynomial is log-concave:*

$$g(z_1, \dots, z_n) := \sum_{B \in \mathcal{B}} \prod_{i \in B} z_i.$$

**Theorem 14** ([BH19; Ana+18]). *For any matroid  $\mathcal{M}$  with family of independent sets  $\mathcal{I}$ , the polynomial*

$$g_{\mathcal{M}}(y, z_1, \dots, z_n) = \sum_{I \in \mathcal{I}} y^{n-|I|} \prod_{i \in I} z_i$$

*in  $\mathbb{R}[y, z_1, \dots, z_n]$  is completely log-concave.*

We will use the following simple fact about log-concave polynomials:

**Proposition 15** ([see Ana+18; BH18]). *If  $g$  is a log-concave polynomial with nonnegative coefficients, then  $\nabla^2 g$  evaluated at any point in the positive orthant has at most one positive eigenvalue.*

One of the basic operations preserving (complete) log-concavity is composition with a linear map. That is if  $T : \mathbb{R}^m \rightarrow \mathbb{R}^n$  is an affine linear map for which  $T(\mathbb{R}_{\geq 0}^m) \subseteq \mathbb{R}_{\geq 0}^n$ , then  $g \circ T$  is (completely) log-concave as well. We state other operations that preserve strong log-concavity/Lorentzian property.

**Proposition 16** ([BH19, Polarization]). *For an element  $\kappa$  of  $\mathbb{N}^n$  let*

$$\begin{aligned} \mathbb{R}_\kappa[z_i]_{1 \leq i \leq n} \\ = \{ \text{polynomials in } \mathbb{R}[z_i]_{1 \leq i \leq n} \text{ of degree at most } \kappa_i \text{ in } z_i \forall i \} \end{aligned}$$

$$\mathbb{R}_\kappa^a[z_{ij}] = \{ \text{multi-affine polynomials in } \mathbb{R}[z_{ij}]_{1 \leq i \leq n, 1 \leq j \leq \kappa_i} \}$$

The polarization map  $\Pi_\kappa^\uparrow$  is a linear map that sends monomial  $z^\alpha = \prod_{i=1}^n z_i^{\alpha_i}$  to the product

$$\frac{1}{\binom{\kappa}{\alpha}} \prod_{i=1}^n e_{\alpha_i}(\{z_{ij}\}_{1 \leq j \leq \kappa_i})$$

where  $e_{\alpha_i}(\{z_{ij}\}_{1 \leq j \leq \kappa_i})$  is the elementary symmetric polynomial of degree  $\alpha_i$  in the variables  $\{z_{ij}\}_{1 \leq j \leq \kappa_i}$  and  $\binom{\kappa}{\alpha} = \prod_{i=1}^n \binom{\kappa_i}{\alpha_i}$ . If  $g \in \mathbb{R}_\kappa[z_i]_{1 \leq i \leq n}$  is strongly log-concave then  $\Pi_\kappa^\uparrow(g)$  is also strongly log-concave.

**Proposition 17.** *The product of two strongly log-concave polynomials is also strongly log-concave.*

## 2.5 Down-Up Random Walk

For two distributions  $\nu, \mu$  we define the Kullback-Leibler divergence, KL-divergence for short,  $\mathcal{D}_{\text{KL}}(\nu \parallel \mu)$  as

$$\mathcal{D}_{\text{KL}}(\nu \parallel \mu) := \mathbb{E}_{S \sim \mu} \left[ \frac{\nu(S)}{\mu(S)} \log \frac{\nu(S)}{\mu(S)} \right] = \mathbb{E}_{S \sim \nu} \left[ \log \frac{\nu(S)}{\mu(S)} \right],$$

and the total variation distance between  $\nu$  and  $\mu$  as

$$\|\nu - \mu\|_{\text{TV}} := \frac{1}{2} \sum_S |\nu(S) - \mu(S)|.$$

The two are related by Pinsker's inequality:

**Proposition 18** ([see, e.g., CT12]). *KL-divergence and the total variation distance are related by the following inequality*

$$\|\nu - \mu\|_{\text{TV}} \leq \sqrt{\frac{1}{2} \mathcal{D}_{\text{KL}}(\nu \parallel \mu)}$$

Cryan, Guo, and Mousa [CGM19] proved shrinkage of the KL-divergence under the down-up random walk. Coupled with Pinsker's inequality, this resulted in a mixing time bound.

**Lemma 19.** [CGM19] *If  $\nu, \mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$  are distributions where  $\mu$  has a log-concave generating polynomial, and  $P$  is the down-up random walk operator whose stationary distribution is  $\mu$ , then*

$$\mathcal{D}_{\text{KL}}(\nu P \parallel \mu P) = \mathcal{D}_{\text{KL}}(\nu P \parallel \mu) \leq (1 - 1/k) \mathcal{D}_{\text{KL}}(\nu \parallel \mu).$$

## 3 Mixing Time Analysis

In this section we prove [Theorem 1](#) by analyzing the down-up random walk for distributions  $\mu$  that have a log-concave generating polynomial. As a reminder, in each step, the down-up random walk transitions from a set  $S \in \binom{[n]}{k}$  to  $S' \in \binom{[n]}{k}$  as follows:

- From  $S$  choose a subset  $T \subseteq S$  of size  $k - 1$  uniformly at random.
- From all supersets  $S' \supseteq T$ , choose one with probability  $\propto \mu(S')$ .

Notice that the first step above simply drops a uniformly random element, and the second step replaces it with a new one (potentially the same element). Our high-level strategy is to prove that in  $O(k \log k)$  steps, every element of the initial set is replaced at least once, and when this happens the distribution becomes a *warm start* and converges to  $\mu$  in an additional  $O(k \log k)$  steps.

Let  $\tau$  be the first time such that every element in our initial set has been replaced at least once. In other words think of initial elements as unmarked, and every time we replace an element we mark the new element brought in. Then  $\tau$  is the first time that every element is marked.

We will prove the following:

**Lemma 20.** Let  $S_t$  be the set at time  $t$  in the down-up random walk. Then for any  $X \in \binom{[n]}{k}$  and any time  $t$ ,

$$\mathbb{P}[S_t = X \mid \tau \leq t] \leq 2^{O(k^2)} \mathbb{P}_\mu[X].$$

Note that without  $2^{O(k^2)}$ , the r.h.s. is simply the stationary distribution. So this statement can be understood to say that as long as we have replaced each element at least once, we cannot be too far off from the stationary distribution.

Before proving Lemma 20, let us finish the proof of Theorem 1 assuming it.

*Proof of Theorem 1 assuming Lemma 20.* Note that for any fixed time  $t$ , we can simply bound  $\mathbb{P}[\tau > t]$  by  $k(1 - 1/k)^t \leq ke^{-t/k}$ . In particular this probability rapidly converges to 0 after about  $k \log k$  steps.

Now let  $t_1 < t_2$  be two time indices. Let  $\nu_t$  denote the distribution of the state of random walk, i.e.,  $S_t$ , at time  $t$ . Our goal is to bound  $\|\nu_t - \mu\|_{\text{TV}}$ , where for simplicity of notation, we assume  $\mu$  is properly normalized to be a probability distribution. Let  $\nu'_{t_1}$  be the distribution of  $S_{t_1}$  conditioned on  $\tau \leq t_1$ , and let  $\nu''_{t_1}$  be the distribution of  $S_{t_1}$  conditioned on  $\tau \geq t_1$ . Then we can write

$$\nu_{t_1} = \mathbb{P}[\tau \leq t_1] \cdot \nu'_{t_1} + \mathbb{P}[\tau > t_1] \cdot \nu''_{t_1}.$$

If  $P$  denotes the random walk operator, then note that  $\nu_{t_2} = \nu_{t_1} P^{t_2 - t_1}$ . So we get

$$\nu_{t_2} = \mathbb{P}[\tau \leq t_1] \nu'_{t_1} P^{t_2 - t_1} + \mathbb{P}[\tau > t_1] \nu''_{t_1} P^{t_2 - t_1}.$$

Using the triangle inequality we can bound

$$\|\nu_{t_2} - \mu\|_{\text{TV}} \leq \|\nu'_{t_1} P^{t_2 - t_1} - \mu\|_{\text{TV}} + \mathbb{P}[\tau > t_1].$$

Here we used the fact that  $\mathbb{P}[\tau \leq t_1] \leq 1$ , and  $\|\nu''_{t_1} P^{t_2 - t_1} - \mu\|_{\text{TV}} \leq 1$ ; the latter inequality is because  $\|\cdot\|_{\text{TV}}$  is always upper bounded by 1.

We can bound the second term in the above inequality by  $ke^{-t_1/k}$  as stated before. For the first term, note that the KL-divergence between  $\nu'_{t_1}$  and  $\mu$  is at most  $O(k^2)$  by Lemma 20. This is because

$$\mathcal{D}_{\text{KL}}(\nu'_{t_1} \parallel \mu) = \mathbb{E}_{S \sim \nu'_{t_1}} \left[ \log \frac{\nu'_{t_1}(S)}{\mu(S)} \right] \leq \log(2^{O(k^2)}) = O(k^2).$$

So by Lemma 19 in  $t_2 - t_1$  steps this KL-divergence decreases to  $(1 - 1/k)^{t_2 - t_1} O(k^2) = O(k^2 e^{-(t_2 - t_1)/k})$ . By Pinsker's inequality, Proposition 18, we get that

$$\|\nu'_{t_1} P^{t_2 - t_1} - \mu\|_{\text{TV}} \leq O(ke^{-(t_2 - t_1)/2k}).$$

So in the end we get the following bound

$$\|\nu_{t_2} - \mu\|_{\text{TV}} \leq O(ke^{-(t_2 - t_1)/2k} + ke^{-t_1/k}).$$

In order for this to be at most  $\epsilon$ , it is enough to make sure that  $\min \{t_1, t_2 - t_1\} = \Omega(k \log k + k \log \frac{1}{\epsilon})$ . So we can simply let  $t_1 = t_2/2$ , and then make sure that  $t_2 = \Omega(k \log(k/\epsilon))$ .  $\square$

As the main tool we use to prove Lemma 20, we introduce a new inequality for log-concave polynomials, that we call approximate exchange. We state the inequality below and defer its proof to Section 4.

**Lemma 21.** Any  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$  with a log-concave generating polynomial satisfies a  $2^{O(k)}$ -exchange property. That is, for every  $S, T \in \binom{[n]}{k}$  and  $i \in S$  there exists  $j \in T$  such that

$$\mu(S)\mu(T) \leq 2^{O(k)} \mu(S - i + j)\mu(T + i - j).$$

Armed with Lemma 21, let us prove Lemma 20.

*Proof of Lemma 20.* Let's look at the down-up walk process *with orders*. This means that we start with some elements  $e_1, \dots, e_k$  that together form the starting set. In each time step we replace one of the  $e_i$ 's. But we keep track of the ordering and do not convert these to sets. So we can talk about  $e_i^t$  as the  $i$ -th element at time  $t$ . In particular  $S_t$  is simply the *unordered* collection  $\{e_1^t, \dots, e_k^t\}$ . Let's say that  $X = \{f_1, \dots, f_k\}$ . Then to have  $S_t = X$ , there must be some permutation of  $f_1, \dots, f_k$  that equals  $e_1^t, \dots, e_k^t$ . We will show that for any such permutation the promised bound in Lemma 20 holds. Since there are  $k! = 2^{O(k \log k)}$  many permutations, this extra factor of  $k!$  can be absorbed into the factor of  $2^{O(k^2)}$  without any loss. So we fix an arbitrary permutation, w.l.o.g. the identity permutation, and try to bound the following

$$\mathbb{P}[e_1^t = f_1, \dots, e_k^t = f_k \mid \tau \leq t].$$

Since we are conditioning on  $\tau \leq t$ , note that there must be some time  $\tau_i \leq t$ , which is the last time before  $t$  where the  $i$ -th element gets replaced by the down-up random walk. We will bound the above probability, even conditioned on  $\tau_1, \dots, \tau_k$  having any set of fixed values up to  $t$ . Note that the index of the element that gets replaced in every step is uniformly random and independent of everything else that happens in the random walk, in particular the identity of the elements that come in as replacements. In the rest of the proof, we condition on the indices of the elements that get replaced at every step up to time  $t$ ; note that this also uniquely determines  $\tau_1, \dots, \tau_k$ , so we assume  $\tau_1, \dots, \tau_k$  are some fixed time indices. W.l.o.g. assume that  $\tau_1 < \tau_2 < \dots < \tau_k$ . We will use induction to prove the following statement for  $i = 0, \dots, k$ :

$$\begin{aligned} \mathbb{P}[e_1^{\tau_1} = f_1, e_2^{\tau_2} = f_2, \dots, e_i^{\tau_i} = f_i \mid \text{replacement indices}] \\ \leq 2^{O(ik)} \mathbb{P}_{U \sim \mu}[f_1, \dots, f_i \in U]. \end{aligned}$$

Notice that for  $i = 0$ , both sides are trivially equal to 1, and for  $i = k$ , this inequality is the main statement we want to prove.

It remains to show the inductive step. We will show that going from  $i - 1$  to  $i$ , the l.h.s. gets multiplied by a smaller quantity compared to the r.h.s. If we have below inequality in hand, then it is not hard to see that we can complete the induction, since the factors that get multiplied on each side are the two sides of this inequality.

$$\begin{aligned} \mathbb{P}[e_i^{\tau_i} = f_i \mid e_1^{\tau_1} = f_1, \dots, e_{i-1}^{\tau_{i-1}} = f_{i-1} \\ \text{and replacement indices}] \\ \leq 2^{O(k)} \mathbb{P}_{U \sim \mu}[f_i \in U \mid f_1, \dots, f_{i-1} \in U]. \quad (1) \end{aligned}$$

Instead of conditioning only on  $f_1, \dots, f_{i-1}$  being chosen at the appropriate times on the l.h.s., we will refine the conditioning and condition on the *history of the random walk* up to time  $\tau_{i-1}$ . This means we can in particular assume that the elements  $e_{i+1}^{\tau_i}, \dots, e_k^{\tau_i}$  are fixed, that  $e_1^{\tau_i} = f_1, \dots, e_{i-1}^{\tau_i} = f_{i-1}$ , and the only uncertain thing is what the  $i$ -th element is being replaced by at time  $\tau_i$ .

Let  $S = \{f_1, \dots, f_i, e_{i+1}^{\tau_i}, \dots, e_k^{\tau_i}\}$ . Then the conditional probability of choosing  $f_i$  at time  $\tau_i$  is:

$$\begin{aligned} \mathbb{P}[e_i^{\tau_i} = f_i \mid e_1^{\tau_1} = f_1, \dots, e_{i-1}^{\tau_{i-1}} = f_{i-1} \\ \text{and replacement indices}] = \frac{\mu(S)}{\sum_{V \supset S - f_i} \mu(V)}. \quad (2) \end{aligned}$$

On the other hand

$$\mathbb{P}_{U \sim \mu}[f_i \in U \mid f_1, \dots, f_{i-1} \in U] = \frac{\sum_{U \ni f_1, \dots, f_i} \mu(U)}{\sum_{T \ni f_1, \dots, f_{i-1}} \mu(T)}$$

So we have to show the following:

$$\begin{aligned} \mu(S) \left( \sum_{T \ni f_1, \dots, f_{i-1}} \mu(T) \right) \\ \leq 2^{O(k)} \left( \sum_{V \supset S - f_i} \mu(V) \right) \left( \sum_{U \ni f_1, \dots, f_i} \mu(U) \right). \quad (3) \end{aligned}$$

We will give an injection from the terms on the l.h.s. to the terms in the expanded form of the r.h.s. Choose some set  $T \ni f_1, \dots, f_{i-1}$ . Apply [Lemma 21](#) to  $S$  and  $T$  with the element  $f_i \in S$ . We get that there must be some element  $e \in T$  such that

$$\mu(S)\mu(T) \leq 2^{O(k)} \mu(S - f_i + e)\mu(T + f_i - e).$$

Note that  $V := S - f_i + e$  contains  $S - f_i$ , and  $U := T + f_i - e$  contains  $\{f_1, \dots, f_i\}$ . So  $\mu(U)\mu(V)$  appears on the r.h.s. of the desired inequality. So for each  $T$  appearing on the l.h.s. of the desired inequality we produced a pair of  $U$  and  $V$ . Note that this mapping from  $T$  to  $(X, Y)$  is injective. This is because given  $(X, Y)$ , we can recover  $T$  as the xor/symmetric difference of the other three sets, that is  $T = S \Delta U \Delta V$ .  $\square$

## 4 Approximate Exchange Property

In this section we prove [Lemma 21](#). In addition, we show other variant(s) of approximate exchange property ([Lemma 24](#)) with implication for the approximation guarantee of local search and greedy on the problem of  $\max_S \mu(S)$  given a log-concave/strongly-Rayleigh distribution  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$ .

**Definition 22.** We say that  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$  has an  $\alpha$ -approximate exchange property, or  $\alpha$ -exchange for short, if for every  $S, T \in \binom{[n]}{k}$  and every  $i \in S$ , there exists  $j \in T$  such that

$$\alpha \cdot \mu(S - i + j)\mu(T + i - j) \geq \mu(S)\mu(T).$$

Note that if  $\mu$  is the indicator of bases of a matroid, then it has a 1-exchange property, also known as the strong basis exchange property [[Oxl06](#)].

Although we do not directly need it, we give another example where approximate exchange can be proven by elementary means. This is the class of  $k$ -determinantal point processes [[BBL09](#); [KT12](#)].

**Proposition 23.** Suppose that  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$  is defined as

$$\mu(S) = \det([v_i]_{i \in S})^2,$$

for some vectors  $v_1, \dots, v_n \in \mathbb{R}^k$ . Then  $\mu$  has a  $k^2$ -exchange property.

*Proof.* It is enough to consider the case where  $S$  and  $T$  are disjoint; otherwise, the problem can be reduced to lower values of  $k$  by taking out the intersection, and projecting all vectors on the orthogonal complement of the space spanned by the intersection.

Define the number  $\beta_j$  as  $\sqrt{\mu(S - i + j)\mu(T + i - j)}$  and let  $\alpha$  be  $\sqrt{\mu(S)\mu(T)}$ . The Plücker relations for the Grassmannian [see, e.g., [Abe80](#)] say that a signed sum of  $\alpha$  and  $\beta_j$  is zero:

$$\alpha + \sum_{j \in T} \pm \beta_j = 0.$$

This means that there is at least one  $j$  such that  $|\beta_j| \geq \frac{1}{k}\alpha$ , and this concludes the proof.  $\square$

Next we take steps to prove [Lemma 21](#), namely that if  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$  has a log-concave generating polynomial  $g_\mu$ , then  $\mu$  has a  $2^{O(k)}$ -exchange property. We conjecture that a  $k^{O(1)}$ -exchange property should hold, but even if true, this will not improve the mixing time results in this paper beyond constants hidden in the  $O(\cdot)$  notation.

Our strategy is to prove the case of  $k = 2$  of [Lemma 21](#) by using log-concavity of  $g_\mu$  (note that  $k = 1$  is trivial). We will then use an induction to prove the general case. We remark that this type of induction is a standard procedure used in many other places, such as in the context of proving Plücker relations and  $M^4$ -concavity [[MS18](#)].

Before delving into the proof, note that we can always assume  $S \cap T = \emptyset$ . This is because we can always condition the distribution  $\mu$  on having any set of elements, and then throwing out those elements; this operation corresponds to taking partial derivatives of  $g_\mu$  which results in a log-concave polynomial by [Lemma 11](#). In particular, we can condition  $\mu$  on having  $S \cap T$ , and then throwing out  $S \cap T$  from the ground set.

*Proof of Lemma 21 for  $k = 2$ .* When  $k = 2$ , we might as well assume that  $n = 4$ , because no element outside of  $S \cup T$  is important, and we can condition the distribution  $\mu$  on not having those elements. This corresponds to substituting 0 for variables outside  $S \cup T$  in  $g_\mu$  which preserves log-concavity.

So our goal now is to show that for a log-concave quadratic polynomial in four variables

$$g_\mu = \sum_{\{i,j\} \in \binom{[4]}{2}} \mu(\{i,j\}) z_i z_j,$$

we have an  $O(1)$ -exchange property. W.l.o.g. assume that  $S = \{1, 2\}$  and  $T = \{3, 4\}$ .

Let us consider  $\nabla^2 g_\mu$ . This is a constant matrix, which has at most one positive eigenvalue by [Proposition 15](#). On the other hand it is a matrix with nonnegative entries, so it must have at least one nonnegative eigenvalue as well. Analyzing the possible signs of the eigenvalues, we see that their product, i.e., the determinant is nonpositive:

$$\det(\nabla^2 g_\mu) \leq 0.$$

This determinant can be written in a special way. Let us define:

$$\begin{aligned} A &:= \mu(\{1, 2\}) \mu(\{3, 4\}), \\ B &:= \mu(\{1, 3\}) \mu(\{2, 4\}), \\ C &:= \mu(\{1, 4\}) \mu(\{2, 3\}). \end{aligned}$$

Notice that approximate exchange for  $S, T$  any any  $i \in S$  is equivalent to saying that  $A \leq O(1) \cdot \max \{B, C\}$ . We can write  $\det(\nabla^2 g_\mu) = A^2 + B^2 + C^2 - 2(AB + AC + BC)$ . So we get the inequality

$$A^2 + B^2 + C^2 \leq 2(AB + AC + BC).$$

This is the same as

$$(A - B - C)^2 \leq 4BC.$$

Taking square-roots we get

$$A - B - C \leq 2\sqrt{BC},$$

which is the same as saying

$$A \leq (\sqrt{B} + \sqrt{C})^2.$$

Taking square-roots again we get

$$\sqrt{A} \leq \sqrt{B} + \sqrt{C}.$$

In particular one of  $\sqrt{B}$  and  $\sqrt{C}$  must be at least  $\frac{1}{2}\sqrt{A}$ . This proves that  $\mu$  satisfies a  $2^2 = 4$ -approximate exchange property for  $S = \{1, 2\}$  and  $T = \{3, 4\}$ .  $\square$

We now complete the proof by inducting on  $k$ .

*Proof of Lemma 21 for the general case.* We can assume that for any  $S, T$  such that  $|S \cap T| \geq 1$ , we have a  $2^{O(k-|S \cap T|)}$ -approximate exchange property. This is because by the arguments we had, such nonempty intersections can be reduced to smaller values of  $k$  by conditioning and throwing out  $S \cap T$ .

Now let  $S \cap T = \emptyset$  and let  $i \in S$  be given. Our goal is to find  $j$  such that

$$\mu(S)\mu(T) \leq 2^{O(k)}\mu(S-i+j)\mu(T+i-j).$$

Let  $i' \neq i$  be another, arbitrary, element of  $S$ . We will exchange  $i'$  with an element  $j' \in T$  and use induction on  $S - i' + j'$  and  $T$ . We need to be careful how we choose  $j'$  though. Let us choose  $j'$  to be the element of  $T$  that maximizes the expression  $\mu(T+i-j')\mu(S-i'+j')$ . The reason for this choice will become apparent in the rest of the proof.

Then the sets  $S - i' + j'$  and  $T$  have an intersection of one element, so by induction we know an approximate exchange property for them. Therefore, there must be a  $j \in T$  such that

$$\mu(S-i'+j')\mu(T) \leq 2^{O(k-1)}\mu(S-i-i'+j+j')\mu(T+i-j). \quad (4)$$

We will apply approximate exchange a second time. The sets  $S$  and  $S - i - i' + j + j'$  have a very large intersection. In particular their exchange property reduces to the case of  $k = 2$  of Lemma 21, which we have already proven. By this exchange property, we have

$$\begin{aligned} \mu(S)\mu(S-i-i'+j+j') &\leq \\ &2^{O(1)} \max\{\mu(S-i+j)\mu(S-i'+j'), \\ &\mu(S-i+j')\mu(S-i'+j)\}. \end{aligned} \quad (5)$$

If the first term in Eq. (5) achieves the maximum, then we are done, because multiplying Eqs. (4) and (5) yields

$$\begin{aligned} \mu(S-i'+j')\mu(T)\mu(S)\mu(S-i-i'+j+j') &\leq \\ &2^{O(k)}\mu(S-i-i'+j+j')\mu(T+i-j)\mu(S-i+j)\mu(S-i'+j'), \end{aligned}$$

which simplifies to

$$\mu(S)\mu(T) \leq 2^{O(k)}\mu(S-i+j)\mu(T+i-j),$$

showing that  $i$  can be exchanged for  $j$ .

So assume that the second term in Eq. (5) achieves the maximum. We will show that in this case  $i$  can be exchanged for  $j'$ . Multiplying Eqs. (4) and (5) yields

$$\begin{aligned} \mu(S-i'+j')\mu(T)\mu(S)\mu(S-i-i'+j+j') &\leq \\ &2^{O(k)}\mu(S-i-i'+j+j')\mu(T+i-j)\mu(S-i+j)\mu(S-i'+j), \end{aligned}$$

which simplifies to

$$\begin{aligned} \mu(S)\mu(T) &\leq \\ &2^{O(k)} \cdot \mu(S-i+j')\mu(T+i-j') \frac{\mu(T+i-j)\mu(S-i'+j)}{\mu(T+i-j')\mu(S-i'+j')} \end{aligned}$$

Notice that by our choice of  $j'$ , the fraction appearing on the r.h.s. is  $\leq 1$ . So we can conclude that

$$\mu(S)\mu(T) \leq 2^{O(k)}\mu(S-i+j')\mu(T+i-j').$$

□

If we require the stricter assumption that  $\mu$  is generated by a real-stable polynomial, then we obtain an exponentially improved  $k^2$ -exchange. This is a generalization of [Proposition 23](#).

**Lemma 24.** Consider  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$  that is generated by a real-stable polynomial. For every  $S, T \in \binom{[n]}{k}$  and  $i \in S \setminus T$

$$\sqrt{\mu(S)\mu(T)} \leq \sum_{j \in T \setminus S} \sqrt{\mu(S - i + j)\mu(T + i - j)} \quad (6)$$

Consequently, there exists  $j \in T \setminus S$  such that

$$\mu(S)\mu(T) \leq k^2\mu(S - i + j)\mu(T + i - j) \quad (7)$$

Thus  $\mu$  satisfies a  $k^2$ -exchange property. Moreover, for  $S \in \binom{[n]}{k}$  and  $j \notin S$ ,

$$\mu(S)\mu(j) \leq k \sum_{e \in S} \mu(S + j - e)\mu(e) \quad (8)$$

where  $\mu(t) = \sum_{T \in \binom{[n]}{k}: t \in T} \mu(T)$  for  $t \in \{j, e\}$ .

A consequence of [Lemma 24](#) is that natural local search and greedy algorithms for finding the maximum of  $\mu : \binom{[n]}{k} \rightarrow \mathbb{R}_{\geq 0}$  (i.e., MAP inference) give a  $(k!)^2 \simeq k^{O(k)}$ -approximation of  $\max \{\mu(S)\}$ , assuming  $\mu$ 's generating polynomial  $g_\mu(z_1, \dots, z_n) = \sum_S \mu(S)z^S$  is real-stable/strongly Rayleigh. This generalizes similar results for determinantal  $\mu$  to the class of strongly Rayleigh distributions [see, e.g., [KD16](#)], giving further evidence for the efficacy of local search and greedy methods [see also [Fed13](#)].

For subset  $T$  of  $[n]$  of size  $\leq k$ , let  $\mu(T) = \sum_{S \in \binom{[n]}{k}: S \supseteq T} \mu(S)$ .

---

**Algorithm 2:** Greedy

---

```

Initialize  $S \leftarrow \emptyset$ 
while  $|S| < k$  do
| Pick  $i \notin S$  that maximizes  $\mu(S \cup \{i\})$ , and update  $S \leftarrow S \cup \{i\}$ 
end

```

---

**Lemma 25.** If  $\mu$  is strongly Rayleigh, then the output  $S \in \binom{[n]}{k}$  of Algorithm 2 is a  $(k!)^2$ -approximation of  $OPT := \max_{T \in \binom{[n]}{k}} \mu(T)$ .

We leave the proof of [Lemmas 24](#) and [25](#) to the full version. We remark that similar guarantees can be obtained for a closely related local search algorithm, which moves between sets of size  $k$ , each time replacing one element by another. Note that our improved exchange property for strongly Rayleigh distributions is crucial in obtaining  $k^{O(k)}$ -approximation. For arbitrary log-concave distributions, we can show the approximate exchange property in [Lemma 24](#) with approximation factor  $2^{O(k)}$  instead of  $k^2$ , thus proving a  $2^{O(k^2)}$ -approximation guarantee for greedy. Furthermore, we show that local search yields  $k^{O(k)}$ -approximation. We leave the proof of these statements to the full version.

## 5 Sampling Forests/Spanning Trees

In this section we prove [Theorems 2](#) and [3](#).<sup>3</sup>

Let  $\mu$  be the distribution over forests of  $G$  defined in [Theorem 2](#). In [Lemma 26](#), we show a homogeneous multiaffine log-concave polynomial  $f_{\mathcal{M}, q, w}^\uparrow$  that generates a homogeneous distribution  $\mu^\uparrow : \binom{E \cup Y}{n} \rightarrow \mathbb{R}_{\geq 0}$  whose projection to  $E$  is the complement distribution of  $\mu$ , i.e.,  $\mathbb{P}_{T \sim \mu^\uparrow}[T \cap E] = \mu(E \setminus (T \cap E))$ .

---

<sup>3</sup>We remark that reliance on [Theorem 1](#) in this section is not mandatory and the results of this section would have been possible even without [Theorem 1](#).

We then run the down-up random walk on the distribution  $\mu^\uparrow$  for some  $t$  steps, and obtain a random set  $T_t \in \text{supp}(\mu^\uparrow)$ . We argue that the distribution of  $E \setminus (T_t \cap E)$  is  $\epsilon$ -close to  $\mu$  for some  $t = O\left(|E| \log \frac{|E|}{\epsilon}\right)$  using the mixing time bound proved in [Theorem 1](#). For completeness, we briefly discuss how to implement each step of the random walk in  $O(\log |E|)$  time.

**Lemma 26.** *Let  $\mathcal{M}$  be a matroid of rank  $r$  over ground set  $[n]$ , and  $I(\mathcal{M})$  be its family of independent sets. The following polynomials are completely log-concave.*

(a)

$$f_{\mathcal{M}}(z_0, z_1, \dots, z_n) = \sum_{S \in I(\mathcal{M})} z_0^{|S|} z^{[n] \setminus S}$$

(b) For

$$f_{\mathcal{M},q,w}(z_0, z_1, \dots, z_n) = \sum_{S \in I(\mathcal{M})} q^{r-|S|} w^S z_0^{|S|} z^{[n] \setminus S}$$

where  $q \geq 0$  and  $w_1, \dots, w_n > 0$

(c)

$$\begin{aligned} f_{\mathcal{M},q,w}^\uparrow(y_1, \dots, y_r, z_1, \dots, z_n) \\ = \sum_{S \in I(\mathcal{M}), T \in \binom{[r]}{|S|}} \frac{1}{\binom{r}{|S|}} q^{r-|S|} w^S y^T z^{[n] \setminus S} \end{aligned} \quad (9)$$

where  $q \geq 0$  and  $w_1, \dots, w_n > 0$

*Proof.* Since

$$f_{\mathcal{M},q,w}^\uparrow(y_1, \dots, y_r, z_1, \dots, z_n) = \prod_{\kappa}^\uparrow (f_{\mathcal{M},q,w})$$

with  $\kappa_0 = r$  and  $\kappa_i = 1 \forall i \in [n]$ , [Item \(b\)](#) implies [Item \(c\)](#) by [Proposition 16](#).

We show that [Item \(a\)](#) implies [Item \(b\)](#).

If  $q > 0$  then

$$f_{\mathcal{M},q,w}(z_0, z_1, \dots, z_n) \propto f_{\mathcal{M}}\left(\frac{z_0}{q}, \frac{z_1}{w_1}, \dots, \frac{z_n}{w_n}\right)$$

is completely log-concave since composing with a linear map preserves complete log-concavity.

If  $q = 0$  then

$$\begin{aligned} f_{\mathcal{M},q,w}(z_0, z_1, \dots, z_n) &= \sum_{S \in B(\mathcal{M})} w^S z^{[n] \setminus S} \\ &\propto \frac{\partial^r f_{\mathcal{M}}}{\partial z_0^r}\left(\frac{z_1}{w_1}, \dots, \frac{z_n}{w_n}\right) \end{aligned} \quad (10)$$

is completely log-concave since taking derivative preserves complete log-concavity.

Now we show [Item \(a\)](#). Let  $f := f_{\mathcal{M}}$ . We first show that  $\text{supp}(f)$  is  $M$ -convex (see [Definition 7](#)).

First, the support of  $g(y, z_1, \dots, z_n) = \sum_{S \in I(\mathcal{M})} y^{n-|S|} w^S$  is  $M$ -convex. Note that

$$\text{supp}(f) = \{\vec{v} - \tilde{w} \mid \tilde{w} \in \text{supp}(g)\}$$

where  $v_0 = n$  and  $v_i = 1 \forall i \in [n]$ . Thus,  $\text{supp}(f)$  is also  $M$ -convex.

Indeed, consider any  $\alpha, \beta \in \text{supp}(f)$  and  $i \in \{0, \dots, n\}$  s.t.  $\alpha_i < \beta_i$ . Then  $\vec{v} - \alpha, \vec{v} - \beta \in \text{supp}(g)$  and  $(\vec{v} - \alpha)_i > (\vec{v} - \beta)_i$  so there exists  $j$  s.t.  $(\vec{v} - \alpha)_j < (\vec{v} - \beta)_j$ , and  $(\vec{v} - \alpha) - e_i + e_j$  and  $(\vec{v} - \beta) - e_j + e_i$  are in  $\text{supp}(g)$ . This implies  $\alpha_j > \beta_j$  and  $\alpha - e_j + e_i$  and  $\beta - e_i + e_j$  are in  $\text{supp}(f)$ .

We proceed using induction on  $n$ . Obviously, for  $n = 1$ ,  $f$  is a linear function in  $w_0, w_1$  with positive coefficient, which is trivially completely log-concave. Suppose the statement is true for matroids  $\mathcal{M}'$  on ground set  $[n-1]$  with  $n \geq 2$ .

We only need to verify  $\partial^\alpha f$  is log-concave for all  $\alpha$  with  $|\alpha| = n-2$ . Note that for  $i \in [n]$ ,

$$\partial_i f = \sum_{S \in I(\mathcal{M}): i \notin S} z_0^{|S|} z^{([n] \setminus i) \setminus S} = f_{\mathcal{M} \setminus i}$$

is completely log-concave by applying induction hypothesis to  $\mathcal{M} \setminus i$ . We only need to show  $\partial_0^{n-2} f$  is log-concave. Note that  $\partial_0^{n-2} f \neq 0$  only if  $r = \text{rank}(\mathcal{M}) \geq n-2$ . Also, for  $n=2$ ,  $\partial_0^{n-2} f$  is exactly  $f$ .

For  $r = n-2$ ,

$$\partial_0^{n-2} f = (n-2)! \sum_{S \in \mathcal{B}(\mathcal{M})} z^{[n] \setminus S} = (n-2)! \sum_{S' \in \mathcal{B}(\mathcal{M}^D)} z^{S'}$$

is strongly log-concave (and of degree 2, thus real-stable), since it is the sum over the bases of dual matroid  $\mathcal{M}^*$  of  $\mathcal{M}$  ([Theorem 13](#)).

For  $r = n$ , then  $\mathcal{B}(\mathcal{M}) = \{[n]\}$  and  $f = \prod_{i \in [n]} (z_0 + z_i)$  is real-stable as product of real stable polynomials  $z_0 + z_i$  for  $i \in [n]$ , thus so is  $\partial_0^{n-2} f$ .

For  $r = n-1$ ,

$$\begin{aligned} \partial_0^{n-2} f &= (n-1)! z_0 e_1(z_T) \\ &\quad + (n-2)! (e_2(z_T) + e_1(z_T) e_1(z_{[n] \setminus T})) \end{aligned} \quad (11)$$

where  $T \subseteq [n]$  is such that  $\mathcal{B}(\mathcal{M}) = \{[n] \setminus i \mid i \in T\}$  are the bases of  $\mathcal{M}$ ,  $e_k$  is elementary symmetric polynomial of degree  $k$ , and  $z_S$  is shorthand for  $\{z_i \mid i \in S\}$ . This is because  $\{S \in \mathcal{I}(\mathcal{M}) \mid |S| = n-2\}$  is exactly

$$\{[n] - t - t' \mid t, t' \in T\} \cup \{[n] - t - \bar{t} \mid t \in T, \bar{t} \notin T\}.$$

Set  $u := (n-1)z_0 + e_1(z_{[n] \setminus T})$  then

$$\partial_0^{n-2} f = (n-2)! (e_1(z_T)u + e_2(z_T)) = (n-2)! e_2(u, z_T)$$

Note that  $e_2$  is real-stable ([Theorem 10](#)), and that  $u \in \mathcal{H}$  whenever  $(z_0, z_{[n] \setminus T}) \in \mathcal{H}^{n+1-|T|}$ , where  $\mathcal{H}$  is the upper half plane. Thus  $\partial_0^{n-2} f$  is nonzero for any  $(z_0, z_{[n]}) \in \mathcal{H}^{n+1}$  i.e.  $\partial_0^{n-2} f$  is real stable.  $\square$

*Remark 27.* Observe that

$$f_{\mathcal{M}}(z_0, z_1, \dots, z_n) = \sum_{S \in I(\mathcal{M})} z_0^{|S|} z^{[n] \setminus S}$$

is the *dual* or *complement* of the strongly log-concave polynomial

$$g_{\mathcal{M}}(z_0, z_1, \dots, z_n) = \sum_{S \in I(\mathcal{M})} z_0^{r-|S|} z^S$$

(see [Theorem 14](#)), in the sense that

$$f_{\mathcal{M}}(z_0, z_1, \dots, z_n) = z_0^r z^{[n]} g_{\mathcal{M}}(z_0^{-1}, z_1^{-1}, \dots, z_n^{-1})$$

We remark that while the dual of a real-stable polynomial is real-stable, the dual of a strongly log-concave polynomials is not in general necessarily strongly log-concave.

We are ready to prove [Theorem 2](#).

*Proof of Theorem 2.* Let  $\mathcal{M}$  be the graphic matroid on graph  $G = G(V, E)$  on  $n$  edges. Let  $k := \text{rank}(\mathcal{M})$ . W.l.o.g., we can label the edges by  $1, 2, \dots, n$  and assume  $E = [n]$ . Let  $\mu$  be the distribution over independent sets of  $\mathcal{M}$  (i.e. forest of  $G$ ) where  $\mu(F) \propto q^{k-|F|} w^F$  for  $F \in \mathcal{I}(\mathcal{M})$ . Note that we can remove all edge of weight 0 from  $E$  without changing  $\mu$ . W.l.o.g., we assume this is already done; thus  $w(e) > 0$  for all  $e \in E$ .

Let  $Y := \{y_1, \dots, y_k\}, Z := \{z_1, \dots, z_n\}$ . We identify the variable  $z_i$  with the edge labeled by  $i$ . Let  $f := f_{\mathcal{M}, q, w}^\uparrow$  then  $f$  is multi-affine, homogeneous, and completely log-concave by [Lemma 26](#). Observe that  $f$  is the generating polynomial for distribution  $\mu^\uparrow : \binom{Y \cup Z}{n} \rightarrow \mathbb{R}_{\geq 0}$  defined by  $\mu^\uparrow(T) \propto \frac{q^{k-|T \cap Y|}}{\binom{k}{|T \cap Y|}} w^{Z \setminus T}$  if  $Z \setminus T \in \mathcal{I}(\mathcal{M})$ , and 0 otherwise.

We run the down-up walk starting from  $T_0 \in \text{supp}(\mu^\uparrow)$  (for e.g.  $T_0 = (Z \setminus \mathcal{F}) \cup Y$  for some spanning forest  $\mathcal{F}$ ). Let  $\nu^\uparrow$  be the distribution of the set  $T \in \text{sup}(\mu^\uparrow)$  we obtained after  $O(n \log(n/\epsilon))$  steps; [Theorem 1](#) implies  $\|\mu^\uparrow - \nu^\uparrow\|_{\text{TV}} \leq \epsilon$ . We then remove all  $y_j$  from  $S$  i.e. collapse  $T \subseteq Y \cup Z$  to  $T_Z := Z \setminus (T \cap Z)$ . Let  $\nu$  be the distribution of  $T_Z$ . Clearly,  $\text{supp}(\nu) = \mathcal{I}(\mathcal{M})$ , and if  $\nu^\uparrow$  is the same distribution as  $\mu^\uparrow$ , then  $\nu$  is the same as  $\mu$ . By the data processing inequality, the total variation between  $\nu$  and  $\mu$  is at most  $\epsilon$  since  $\|\mu - \nu\| \leq \|\mu^\uparrow - \nu^\uparrow\|_{\text{TV}} \leq \epsilon$ .

We show each step of the random walk can be implemented in  $O(\log n)$  time. In the down step, we keep track of whether the level  $(n-1)$  set  $S$  still satisfies  $S_Z := Z \setminus S$  is a forest. Note that if we dropped an  $y_j$  in the down step to arrive at  $S$ , then  $S_Z$  is always a forest; if, instead, we dropped a  $z_i$  (equivalently, added  $z_i$  to  $S_Z$ ), then we can check whether  $S_Z$  stays a forest in  $O(\log n)$  amortized time using link-cut tree [[ST83](#); [RTF18](#)]. If  $S_Z$  is not a forest, then let  $\text{cycle}_S$  be the unique cycle in  $S_Z$  which contains the edge  $z_i$  that was added to  $S_Z$ . When we perform an up-step from  $S$ , if  $S_Z$  is not a forest, select  $z_f$  among the edges in  $\text{cycle}_S$  with probability  $\propto 1/w_f$  and add it to  $S$  (equivalently, remove  $z_f$  from  $S_Z$ ). This can be done in  $O(\log n)$ -amortized time (see [[RTF18](#)]). If  $S_Z$  is empty, then we can only add  $y_j$  which is not already in  $S_Y := S \cap Y$  with uniform probability. If  $S_Z$  is a nonempty forest, then we can add any variable  $y_j \notin S_Y$  or  $z_i \in S_Z$ . In this case, the probability of adding variable can be explicitly computed i.e. uniform among  $y_j \notin S_Y$ , and  $\mathbb{P}[z_i] / \mathbb{P}[y_j] = \frac{q^{k-(\ell-1)} w_i^{-1} / \binom{k}{\ell-1}}{q^{k-\ell} / \binom{k}{\ell}}$  where  $\ell := |S_Z| = |S_Y| + 1$ . We can perform these operations in  $O(\log n)$  time by

- With probability  $1/(1+\tau)$  where

$$\tau := \frac{q^{k-(\ell-1)} \sum_{i \in S_Z} w_i^{-1} / \binom{k}{\ell-1}}{q^{k-\ell} (k - |S_Y|) / \binom{k}{\ell}} = \frac{q \sum_{i \in S_Z} w_i^{-1}}{\ell},$$

sample  $y_j$  uniformly at random from  $Y \setminus S_Y$  and add  $y_j$  to  $S_Y$ . Note that this action will always be performed if  $q = 0$ .

- Maintain an array of cumulative sums  $s_t := \sum_{h=1}^t w_{i_h}^{-1}$  for  $t \in [\ell]$  where  $w_{i_1}, w_{i_2}, \dots, w_{i_\ell}$  are the weights corresponding to the edges in  $S_Z$ ; this data structure supports amortized  $O(\log n)$ -time insertion and deletion from  $S_Z$  and binary search in the sorted array  $[s_t]_{t=1}^\ell$ . This data structure can be implemented using a splay tree where each node stores the sum of all leaves in its rooted subtree. With probability  $\tau/(1+\tau)$ , sample  $z_f$  from  $S_Z$  with probability  $\propto 1/w_f$  by: sample uniformly random  $p \in [0, s_\ell]$ , find minimum  $t \in [\ell]$  where  $p \leq s_t$ , and remove  $z_{i_t}$  from  $S_Z$ . This removal will split a tree in the forest  $S_Z$ , and we update the link-cut tree representation of  $S_Z$  accordingly in  $O(\log n)$  time.

See [Fig. 1](#) for a visualization of how one up-then-down step may change the set  $T_Z = Z \setminus T$ .

For completeness, we briefly summarize how to handle sampling and removing an edge from  $\text{cycle}_S$ , which was described in [[RTF18](#)]. We represent  $S_Z$  as a forest of link-cut trees. When we add an edge  $e = (u, v)$  that forms a cycle, splay  $u$  to be the root of its tree  $\mathcal{T}_u$ , then access  $v$  (which is also in  $T_u$ ) so that the entire path  $\mathcal{P}_{u,v}$  from  $u$  to  $v$  in  $T_u$  is stored in one auxiliary tree. This auxiliary splay tree can be augmented to support

(weighted) sampling an edge  $f$  from  $\mathcal{P}_{u,v}$  as described above with  $S_Z$ . Remove  $f$  (which disconnects  $\mathcal{T}_u$  into two trees) then add  $e$ . Link-cut trees support these operations in amortized  $O(\log n)$  time, and the augmentation increases the run-time by only a constant factor.  $\square$

We briefly discuss how to sample from the family of independent sets of a matroid  $\mathcal{M} = ([n], \mathcal{I})$  of rank  $k$  using the framework developed here. If we are given access to a data structure  $\mathcal{O}'$  that maintains a set  $S \subseteq [n]$ , guaranteed to contain at most one circuit and allows for the operations:

- Addition, and removal of an element from  $S$ , provided we maintain the property that  $S$  contains at most one circuit.
- Outputting a uniformly random element from the unique circuit in  $S$  if, such a circuit exists.

then each step of the down-up walk described in [Theorem 2](#) can be implemented with  $O(1)$  calls to  $\mathcal{O}'$ , resulting in a  $O(n \log n)$ -time algorithm to sample uniformly from the family of independent sets of  $\mathcal{M}$ .

We remark that [Theorem 14](#) and the polarization trick employed in Proof of [Theorem 2](#) already give a  $O(kn \log \frac{k}{\epsilon})$ -time algorithm, given access to the independent set oracle  $\mathcal{O}_I$  for  $\mathcal{M}$ . Indeed, the down-up walk on the distribution defined by the polarization of the strongly log-concave polynomial  $g_{\mathcal{M}}(z_1, \dots, z_n) = \sum_{S \in I(\mathcal{M})} z_0^{k-|S|} z^S$  (see [Theorem 14](#)) mixes in  $O(k \log \frac{k}{\epsilon})$  steps, and each step can be implemented using  $O(n)$  calls to  $\mathcal{O}_I$ .

We leave the proof of [Theorem 3](#) to the full version.

## References

- [Abe80] Silvana Abeasis. “On the Plücker relations for the Grassmann varieties”. In: *Advances in Mathematics* 36.3 (1980), pp. 277–282.
- [AD20] Nima Anari and Michal Derezhinski. “Isotropy and Log-Concave Polynomials: Accelerated Sampling and High-Precision Counting of Matroid Bases”. In: *ArXiv* abs/2004.09079 (2020).
- [AHK18] Karim Adiprasito, June Huh, and Eric Katz. “Hodge theory for combinatorial geometries”. In: *Annals of Mathematics* 188.2 (2018), pp. 381–452.
- [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.
- [Ana+18] Nima Anari, Kuikui Liu, Shayan Oveis Gharan, and Cynthia Vinzant. “Log-Concave Polynomials III: Mason’s Ultra-Log-Concavity Conjecture for Independent Sets of Matroids”. In: *CoRR* abs/1811.01600 (2018).
- [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*. ACM, June 2019.
- [AOV18] Nima Anari, Shayan Oveis Gharan, and Cynthia Vinzant. “Log-Concave Polynomials I: Entropy and a Deterministic Approximation Algorithm for Counting Bases of Matroids”. In: *Proceedings of the 59th IEEE Annual Symposium on Foundations of Computer Science*. IEEE Computer Society, Oct. 2018. DOI: 10.1109/focs.2018.00013.
- [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.
- [BH18] Petter Brändén and June Huh. “Hodge-Riemann relations for Potts model partition functions”. In: *arXiv preprint arXiv:1811.01696* (2018).
- [BH19] Petter Brändén and June Huh. “Lorentzian polynomials”. In: *arXiv preprint arXiv:1902.03719* (2019).
- [Bro89] Andrei Z Broder. “Generating random spanning trees”. In: *FOCS*. Vol. 89. Citeseer. 1989, pp. 442–447.
- [CGM19] Mary Cryan, Heng Guo, and Giorgos Mousa. “Modified log-Sobolev inequalities for strongly log-concave distributions”. In: *arXiv preprint arXiv:1903.06081* (2019).

[Clo10] Brian Cloteaux. "Approximating the number of bases for almost all matroids". In: *Congressus Numerantium* 202 (2010), pp. 149–153.

[ÇM10] Ali Çivril and Malik Magdon-Ismail. "Exponential Inapproximability of Selecting a Maximum Volume Sub-matrix". In: *CoRR* abs/1006.4349 (2010). arXiv: 1006.4349.

[CMN96] Charles J Colbourn, Wendy J Myrvold, and Eugene Neufeld. "Two algorithms for unranking arborescences". In: *Journal of Algorithms* 20.2 (1996), pp. 268–281.

[CT12] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[CTY15] Emma Cohen, Prasad Tetali, and Damir Yeliussizov. "Lattice path matroids: negative correlation and fast mixing". In: *arXiv preprint arXiv:1505.06710* (2015).

[Di +14] Marco Di Summa, Friedrich Eisenbrand, Yuri Faenza, and Carsten Moldenhauer. "On largest volume simplices and sub-determinants". In: *Proceedings of the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms*. SIAM, 2014, pp. 315–323.

[DK17] Irit Dinur and Tali Kaufman. "High dimensional expanders imply agreement expanders". In: *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 2017, pp. 974–985.

[Dur+17a] David Durfee, Rasmus Kyng, John Peebles, Anup B Rao, and Sushant Sachdeva. "Sampling random spanning trees faster than matrix multiplication". In: *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*. 2017, pp. 730–742.

[Dur+17b] David Durfee, John Peebles, Richard Peng, and Anup B Rao. "Determinant-preserving sparsification of SDDM matrices with applications to counting and sampling spanning trees". In: *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 2017, pp. 926–937.

[Fed13] Valerii Vadimovich Fedorov. *Theory of optimal experiments*. Elsevier, 2013.

[FM92] Tomás Feder and Milena Mihail. "Balanced matroids". In: *Proceedings of the twenty-fourth annual ACM symposium on Theory of computing*. 1992, pp. 26–38.

[Gam99] Anna Gambin. "On approximating the number of bases of exchange preserving matroids". In: *International Symposium on Mathematical Foundations of Computer Science*. Springer, 1999, pp. 332–342.

[GJ18] Heng Guo and Mark Jerrum. "Approximately counting bases of bicircular matroids". In: *arXiv preprint arXiv:1808.09548* (2018).

[Goe+] Ashish Goel, Sanjeev Khanna, Sharath Raghvendra, and Hongyang Zhang. *Connectivity in Random Forests and Credit Networks*.

[Gur09] Leonid Gurvits. "On multivariate Newton-like inequalities". In: *Advances in combinatorial mathematics*. Springer, 2009, pp. 61–78.

[Jer+04] Mark Jerrum, Jung-Bae Son, Prasad Tetali, and Eric Vigoda. "Elementary bounds on Poincaré and log-Sobolev constants for decomposable Markov chains". In: *The Annals of Applied Probability* 14.4 (2004), pp. 1741–1765.

[Jer06] Mark Jerrum. "Two remarks concerning balanced matroids". In: *Combinatorica* 26.6 (2006), pp. 733–742.

[JS02] Mark Jerrum and Jung-Bae Son. "Spectral gap and log-Sobolev constant for balanced matroids". In: *The 43rd Annual IEEE Symposium on Foundations of Computer Science, 2002. Proceedings*. IEEE, 2002, pp. 721–729.

[JV86] M R Jerrum, L G Valiant, and V V Vazirani. "Random Generation of Combinatorial Structures from a Uniform". In: *Theor. Comput. Sci.* 43.23 (July 1986), pp. 169–188. ISSN: 0304-3975.

[KD16] Tarun Kathuria and Amit Deshpande. "On sampling and greedy map inference of constrained determinantal point processes". In: *arXiv preprint arXiv:1607.01551* (2016).

[Kha95] Leonid Khachiyan. "On the complexity of approximating extremal determinants in matrices". In: *Journal of Complexity* 11.1 (1995), pp. 138–153.

[KM09] Jonathan A Kelner and Aleksander Madry. "Faster generation of random spanning trees". In: *2009 50th Annual IEEE Symposium on Foundations of Computer Science*. IEEE, 2009, pp. 13–21.

[KM16] Tali Kaufman and David Mass. "High dimensional random walks and colorful expansion". In: *arXiv preprint arXiv:1604.02947* (2016).

[KO20] Tali Kaufman and Izhar Oppenheim. "High order random walks: Beyond spectral gap". In: *Combinatorica* (2020), pp. 1–37.

[KT12] Alex Kulesza and Ben Taskar. "Determinantal point processes for machine learning". In: *Foundations and Trends in Machine Learning* 5.2–3 (2012), pp. 123–286.

[MS18] Kazuo Murota and Akiyoshi Shioura. "Simpler exchange axioms for M-concave functions on generalized polymatroids". In: *Japan Journal of Industrial and Applied Mathematics* 35.1 (2018), pp. 235–259.

[MS99] Kazuo Murota and Akiyoshi Shioura. "M-convex function on generalized polymatroid". In: *Mathematics of operations research* 24.1 (1999), pp. 95–105.

[MST14] Aleksander Madry, Damian Straszak, and Jakub Tarnawski. "Fast generation of random spanning trees and the effective resistance metric". In: *Proceedings of the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms*. SIAM. 2014, pp. 2019–2036.

[MT06] Ravi Montenegro and Prasad Tetali. "Mathematical aspects of mixing times in Markov chains". In: *Foundations and Trends in Theoretical Computer Science* 1.3 (2006), pp. 237–354.

[MV89] M. Mihail and U. Vazirani. "On the expansion of 0/1 polytopes". In: *Journal of Combinatorial Theory. B* (1989).

[Nik15] Aleksandar Nikolov. "Randomized rounding for the largest simplex problem". In: *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*. 2015, pp. 861–870.

[OR18] Shayan Oveis Gharan and Alireza Rezaei. "A Polynomial Time MCMC Method for Sampling from Continuous DPPs". In: *arXiv preprint arXiv:1810.08867* (2018).

[Oxl06] James G Oxley. *Matroid theory*. Vol. 3. Oxford University Press, USA, 2006.

[Pac04] Asa Packer. "Polynomial-time approximation of largest simplices in V-polytopes". In: *Discret. Appl. Math.* 134 (2004), pp. 213–237.

[RTF18] Luis Russo, Andreia Sofia Teixeira, and Alexandre P Francisco. "Linking and Cutting Spanning Trees". In: *Algorithms* 11.4 (2018), p. 53.

[Sch18] Aaron Schild. "An almost-linear time algorithm for uniform random spanning tree generation". In: *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*. ACM. 2018, pp. 214–227.

[ST83] Daniel D Sleator and Robert Endre Tarjan. "A data structure for dynamic trees". In: *Journal of computer and system sciences* 26.3 (1983), pp. 362–391.

[Wag09] David G. Wagner. *Multivariate stable polynomials: theory and applications*. 2009. arXiv: 0911.3569 [math.CV].

[Wil96] David Bruce Wilson. "Generating random spanning trees more quickly than the cover time". In: *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*. 1996, pp. 296–303.

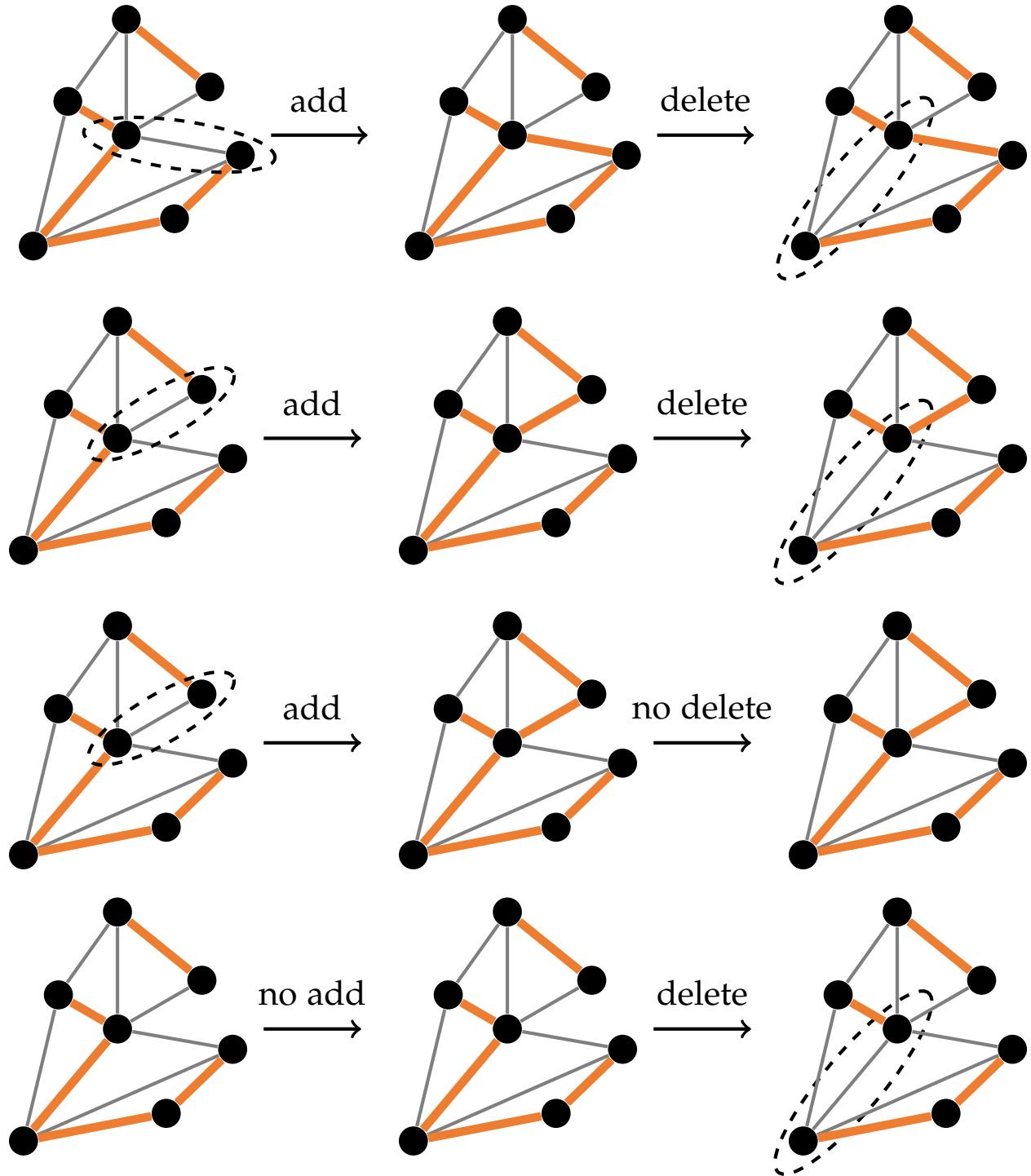


Figure 1: The effect of one step in the down-up random walk on  $\mu^\uparrow$  on  $T_Z = Z \setminus T$ . Four possible ways for  $T_Z$  to change.

Figure 2: Down-up random walk.