

Approximation Algorithms for D -optimal Design

Mohit Singh

School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332, mohitsinghr@gmail.com.

Weijun Xie

Department of Industrial and Systems Engineering, Virginia Tech, Blacksburg, VA 24061, wxie@vt.edu.

Experimental design is a classical statistics problem and its aim is to estimate an unknown m -dimensional vector β from linear measurements where a Gaussian noise is introduced in each measurement. For the combinatorial experimental design problem, the goal is to pick k out of the given n experiments so as to make the most accurate estimate of the unknown parameters, denoted as $\hat{\beta}$. In this paper, we will study one of the most robust measures of error estimation - D -optimality criterion, which corresponds to minimizing the volume of the confidence ellipsoid for the estimation error $\beta - \hat{\beta}$. The problem gives rise to two natural variants depending on whether repetitions of experiments are allowed or not. We first propose an approximation algorithm with a $\frac{1}{\epsilon}$ -approximation for the D -optimal design problem with and without repetitions, giving the first constant factor approximation for the problem. We then analyze another sampling approximation algorithm and prove that it is $(1 - \epsilon)$ -approximation if $k \geq \frac{4m}{\epsilon} + \frac{12}{\epsilon^2} \log(\frac{1}{\epsilon})$ for any $\epsilon \in (0, 1)$. Finally, for D -optimal design with repetitions, we study a different algorithm proposed by literature and show that it can improve this asymptotic approximation ratio.

Key words: D -optimal Design; approximation algorithm; determinant; derandomization.

History:

1. Introduction Experimental design is a classical problem in statistics [5, 17, 18, 23, 31] and recently has also been applied to machine learning [2, 39]. In an experimental design problem, its aim is to estimate an m -dimensional vector $\beta \in \mathbb{R}^m$ from n linear measurements of the form $y_i = \mathbf{a}_i^T \beta + \tilde{\epsilon}_i$ for each $i \in [n] := \{1, 2, \dots, n\}$, where vector $\mathbf{a}_i \in \mathbb{R}^m$ characterizes the i^{th} experiment and $\{\tilde{\epsilon}_i\}_{i \in [n]}$ are i.i.d. independent Gaussian random variables with zero mean and variance σ^2 (i.e., $\tilde{\epsilon}_i \sim \mathcal{N}(0, \sigma^2)$ for all $i \in [n]$). Due to limited resources, it might be quite expensive to conduct all the n experiments. Therefore, as a compromise, in the combinatorial experimental design problem, we are given an integer $k \in [m, n]$, and our goal is to pick k out of the n experiments so as to make the *most accurate estimate* of the parameters, denoted as $\hat{\beta}$. Suppose that a size k subset of experiments $S \subseteq [n]$ is chosen, then the most likelihood estimation of β (cf., [21]) is given by

$$\hat{\beta} = \left(\sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^T \right)^{-1} \sum_{i \in S} y_i \mathbf{a}_i.$$

There are many criteria on how to choose the best estimation among all the possible size k subsets of n experiments (see [5] for a review). One of the most robust measures of error estimation is

known as D -optimality criterion, where the goal is to choose the best size k subset S to maximize $[\det(\sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top)]^{\frac{1}{m}}$, i.e., the following combinatorial optimization problem:

$$\max_S \left\{ f(S) := \left[\det \left(\sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top \right) \right]^{\frac{1}{m}} : \text{Supp}(S) \subseteq [n], |S| = k \right\}, \quad (1)$$

where $\text{Supp}(S)$ denotes the support of set S and $|S|$ denotes the cardinality of set S . Note that optimization model (1) corresponds to minimizing the volume of the confidence ellipsoid for the estimation error $\beta - \hat{\beta}$. Equivalently, the objective function of (1) can be chosen as determinant itself (i.e., $\det(\sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top)$), or log-determinant $\log \det(\sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top)$ [21]. However, both objective functions are problematic with the following reasons: (1) the determinant function is nonconvex and has numerical issue especially when k, m, n are large; and (2) albeit log-determinant function is concave, it can also be numerically unstable, in particular when the determinant is close to zero. Therefore, in this paper, we follow the work in [32] and consider m th root of determinant function, which is concave and numerically stable.

In the problem description, the same experiment may or may not be allowed to choose multiple times. We refer the problem as *D -optimal design with repetitions* if we are allowed to pick an experiment more than once and *D -optimal design without repetitions* otherwise. Correspondingly, for D -optimal design with repetitions and without repetitions, in (1), the subset S denotes a *multi-set*, where elements of S can be duplicated, and a *conventional set*, where elements of S must be different from each other, respectively. The former problem has been studied very extensively in statistics [23, 31, 32]. The latter problem has also been studied as the sensor selection problem [21], where the goal is to find the best subset of sensor locations to obtain the most accurate estimate of unknown parameters. It is easy to see that D -optimal design with repetitions is a special case of the D -optimal design problem without repetitions. To do so, for the D -optimal design with repetitions, we can create k copies of each vector, which reduces to the D -optimal design without repetitions with nk vectors.

The remainder of the paper is organized as follows. Section 2 details the problem setting, reviews related literature, and summarizes our contributions. Section 3 develops and analyzes a randomized algorithm, its approximation results and deterministic implementation. Section 4 proposes another randomized algorithm and its deterministic implementation as well as asymptotic behavior. Section 5 analyzes a randomized algorithm for D -optimal design with repetitions proposed by literature, and investigates its deterministic implementation as well as approximation ratios. Finally, Section 6 concludes the paper.

Notations: The following notation is used throughout the paper. We use bold-letters (e.g., \mathbf{x} , \mathbf{A}) to denote vectors or matrices, and use corresponding non-bold letters to denote their components. We let \mathbf{e} be the all-ones vector. We let \mathbb{R}_+ , \mathbb{Q}_+ , \mathbb{Z}_+ denote the sets of positive real numbers, rational numbers and integers, respectively. Given a positive integer N and a set Q , we let $[N] := \{1, \dots, N\}$, $N! = \prod_{i \in [N]} i$, $|Q|$ denote its cardinality, $\text{Supp}(Q)$ denotes the support of Q and $\binom{Q}{N}$ denote all the possible subsets of Q whose cardinality equals to N . Given a multi-set Q , for each $i \in \text{Supp}(Q) \subseteq [n]$, we let function $M_Q(i)$ denote the number of occurrences of i in Q , and for any $i \in [n]$, $Q(i)$ denotes its i th component. Given a matrix \mathbf{A} and two sets R, T , we let $\det(\mathbf{A})$ denote its determinant if \mathbf{A} is a square matrix, let $\mathbf{A}_{R,T}$ denote a submatrix of \mathbf{A} with rows and columns from sets R, T , let \mathbf{A}_i denote i th column of matrix \mathbf{A} and let \mathbf{A}_R denote submatrix of \mathbf{A} with columns from set R . For any positive integer r , we let \mathbf{I}_r denote $r \times r$ identity matrix. We use \tilde{S} to denote a random set. For notational convenience, give a set $S \in [n]$ and vector $\mathbf{x} \in \mathbb{R}_+^n$, we define $f(S) = [\det(\sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top)]^{\frac{1}{m}}$ and $f(\mathbf{x}) = [\det(\sum_{i \in [n]} x_i \mathbf{a}_i \mathbf{a}_i^\top)]^{\frac{1}{m}}$. Additional notation will be introduced as needed.

2. Model Formulation, Related Literature and Contributions

2.1. Model Formulation To formulate D -optimal design problem (1) as an equivalent mathematical program, we first let set B denote the set of all nonnegative numbers (i.e., $B = \mathbb{R}_+$) if repetitions are allowed, otherwise, $B = [0, 1]$. Next, we introduce integer decision variable $x_i \in B \cap \mathbb{Z}_+$ to represent how many times i th experiment will be chosen for each $i \in [n]$. With the notation introduced above, the D -optimal design problem can be formulated as a mixed integer convex program below:

$$w^* := \max_{\mathbf{x}, w} \left\{ w : w \leq f(\mathbf{x}) = \left[\det \left(\sum_{i \in [n]} x_i \mathbf{a}_i \mathbf{a}_i^\top \right) \right]^{\frac{1}{m}}, \sum_{i \in [n]} x_i = k, \mathbf{x} \in B^n \cap \mathbb{Z}_+^n \right\}, \quad (2)$$

where for notational convenience, we let $f(\mathbf{x})$ to denote the objective function. Note that if $B = \mathbb{R}_+$, (2) is equivalent to D -optimal design with repetitions and if $B = [0, 1]$, then (2) corresponds to D -optimal design without repetitions.

It can be easily shown that $f(\mathbf{x}) = [\det(\sum_{i \in [n]} x_i \mathbf{a}_i \mathbf{a}_i^\top)]^{\frac{1}{m}}$ is concave in \mathbf{x} (cf., [8]). Therefore, a straightforward convex relaxation of (2) is to relax the binary vector \mathbf{x} to continuous, i.e., $\mathbf{x} \in [0, 1]^n$, which is formulated as below:

$$\hat{w} := \max_{\mathbf{x}, w} \left\{ w : w \leq \left[\det \left(\sum_{i \in [n]} x_i \mathbf{a}_i \mathbf{a}_i^\top \right) \right]^{\frac{1}{m}}, \sum_{i \in [n]} x_i = k, \mathbf{x} \in B^n \right\}. \quad (3)$$

Note that (3) is a tractable convex program (cf., [21]), thus is efficiently solvable. Recently, in [32], the authors proposed an alternative second order conic program (SOCP) formulation for (3), which can be solved by a more effective interior point method or even off-the-shelf solvers (e.g., CPLEX, Gurobi, MOSEK). We remark that according to [8], the time complexity of solving (3) is $O(n^5)$.

2.2. Related Literature As remarked earlier, experimental design is a classical area in Statistics. We refer the reader to Pukelsheim [31], Chapter 9 on details about D -optimality criterion as well as other related (A, E, T) -criteria. The combinatorial version, where the number of each experiment needs to be chosen is an integer as in (2), is also called exact experimental design. It turns out that the D -optimality criterion is proven to be NP-hard [40]. Thus, there has been plenty of works on heuristic methods, including local search and its variants, to obtain good solutions [18, 21].

From an approximation algorithm viewpoint, D -optimal design has received a lot of attention recently. For example, Bouhou et al. [9] gave a $\left(\frac{n}{k}\right)^{\frac{1}{m}}$ -approximation algorithm, and Wang et al. [39] building on [6] gave a $(1 + \epsilon)$ -approximation if $k \geq \frac{m^2}{\epsilon}$. Recently, Allen-Zhu et al [2] realized the connection between this problem and matrix sparsification [7, 36] and used regret minimization methods [3] to obtain $O(1)$ -approximation algorithm if $k \geq 2m$ and $(1 + \epsilon)$ -approximation when $k \geq O\left(\frac{m}{\epsilon^2}\right)$. We also remark that their results are general and also are applicable to other optimality criteria.

Another closely related problem is the largest j -simplex problem, whose problem description is as follows:

(largest j -simplex problem) Given a set of n vectors $\mathbf{a}_1, \dots, \mathbf{a}_n \in \mathbb{R}^m$ and integer $k \leq m$, pick a set of S of k vectors to maximize the k^{th} -root of the pseudo-determinant of $\mathbf{X} = \sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top$, i.e., the geometric mean of the non-zero eigenvalues of X .

The problem has also received much attention recently [22, 27, 38] and Nikolov[27] gave a $\frac{1}{\epsilon}$ -approximation algorithm. Observe that the special case of $k = m$ of this problem coincides with the special case of $k = m$ for the D -optimal design problem. Indeed, Nikolov's algorithm, while applicable, results in a $e^{-\frac{k}{m}}$ -approximation for the D -optimal design problem. Recently, matroid constrained versions of the largest j -simplex problem have also been studied [4, 28, 37].

The D -optimality criterion is also closely related to constrained submodular maximization [26], a classical combinatorial problem, for which there has been much progress recently [14, 24]. Indeed, the set function $m \log f(S) := \log \det \left(\sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top \right)$ is known to be submodular [33]. Unfortunately, this submodular function is not necessarily non-negative, a prerequisite for all the results on constrained submodular maximization and thus these results are not directly applicable. We

also remark that for a multiplicative guarantee for the det objective, we would aim for an additive guarantee for log det objective.

2.3. Contributions In this paper, we make several contributions to the approximations of D -optimal design problems both with and without repetitions. Our approximation algorithms are randomized, where we sample k experiments out of n with given marginals. This type of sampling procedure has been studied intensively in the approximation algorithms and many different schemes have been proposed [11], where most of them exhibit a negative correlation of various degrees [10]. In this work, we will study sampling schemes which exhibit approximate positive correlation and prove the approximation ratios of these algorithms.

All the proposed approximations come with approximation guarantees. Given an *approximation ratio* $\gamma \in (0, 1]$, a γ -approximation algorithm for D -optimal design returns a solution $\bar{x} \in B^n \cap \mathbb{Z}_+^n$, such that $\sum_{i \in [n]} \bar{x}_i = k$ and $f(\bar{x}) \geq \gamma w^*$, i.e., the solution is feasible and has an objective value at least γ times the optimal value. The approximation ratios of our randomized algorithms only hold in the sense of expectation. To improve them, we further propose polynomial-time deterministic implementations for all the randomized algorithms with iterated conditional expectation method, which also achieve the same approximation guarantees. Following is a summary of our contributions.

1. We develop a $\frac{1}{\epsilon}$ -approximation algorithm and its polynomial-time deterministic implementation, giving the first constant factor approximation for D -optimal design problem for both with and without repetitions. Previously, constant factor approximations were known only for a restricted range of parameters [2, 6, 39] (see related work for details).
2. We study a different sampling algorithm and its polynomial-time deterministic implementation, showing that its solution is $(1 - \epsilon)$ optimal if $k \geq \frac{4m}{\epsilon} + \frac{12}{\epsilon^2} \log(\frac{1}{\epsilon})$ for any given $\epsilon \in (0, 1)$. This results substantially improve the previous work [2, 39].
3. For D -optimal design with repetitions, we investigate a simple randomized algorithm similar to that in Nikolov [27], study its polynomial-time deterministic implementation and provide a significant different analysis of the approximation guarantee. We show that the proposed algorithm yields $(1 - \epsilon)$ -approximation for the D -optimal design problem with repetitions when $k \geq \frac{m-1}{\epsilon}$.

Note that the preliminary version of the paper has appeared in ACM-SIAM Symposium on Discrete Algorithms (SODA18) [34]. Compared to [34], the current paper has the following major improvement: (1) for the constant-factor approximation algorithm presented in Section 3, we simplify its polynomial-time deterministic implementation and improve its analysis, (2) for the asymptotically optimal algorithm in Section 4, we simplify its sampling procedure and derive

its polynomial-time deterministic implementation, and (3) for the approximation algorithm of D -optimal design with repetitions, we improve its approximation ratio analysis and propose its polynomial-time deterministic implementation.

3. Approximation Algorithm for D -optimal Design Problem In this section, we will propose a sampling procedure and prove its approximation ratio. We also develop an efficient way to implement this algorithm and finally we will show its polynomial-time deterministic implementation with the same performance guarantees.

First of all, we note that for D -optimal design problem with repetitions, i.e., $B = \mathbb{R}_+$ in (2), it can be equivalently reformulated as D -optimal design problem without repetitions by creating k copies of vectors $\{\mathbf{a}_i\}_{i \in [n]}$. Therefore, the approximation for D -optimal design problem without repetitions directly apply to that with repetitions. Hence, in this and next sections, we will only focus on D -optimal design problem without repetitions, i.e., in (2) and (3), we only consider $B = [0, 1]$.

3.1. Sampling Algorithm and Its Efficient Implementation In this subsection, we will introduce a sampling procedure and explain its efficient implementation.

In this sampling algorithm, we first suppose $(\hat{\mathbf{x}}, \hat{w})$ to be an optimal solution to the convex relaxation (3), where $\hat{\mathbf{x}} \in [0, 1]^n$ with $\sum_{i \in [n]} \hat{x}_i = k$ and $\hat{w} = f(\hat{\mathbf{x}})$. Then we randomly choose a size- k subset \tilde{S} according to the following probability:

$$\mathbb{P}[\tilde{S} = S] = \frac{\prod_{j \in S} \hat{x}_j}{\sum_{\tilde{S} \in \binom{[n]}{k}} \prod_{i \in \tilde{S}} \hat{x}_i} \quad (4)$$

for every $S \in \binom{[n]}{k}$, where $\binom{[n]}{k}$ denotes all the possible size- k subsets of $[n]$.

Note that the sampling procedure in (4) is not an efficient implementable description since there are $\binom{n}{k}$ candidate subsets to be sampled from. Therefore, we propose an efficient way (i.e., Algorithm 1) to obtain a size- k random subset \tilde{S} with probability distribution in (4). We first observe a useful way to compute the probabilities in Algorithm 1 efficiently.

Observation 1 *Suppose $\mathbf{x} \in \mathbb{R}^t$ and integer $0 \leq r \leq t$, then $\sum_{S \in \binom{[t]}{r}} \prod_{i \in S} x_i$ is the coefficient of y^r of the polynomial $\prod_{i \in [t]} (1 + x_i y)$.*

In fact, it has been shown that the product of two polynomials with degree at most t can be done in $O(t \log t)$ amount of time by the Fast Fourier Transform (FFT) [16]. Thus, by divide-and-conquer approach, it takes $O(t \log^2 t)$ time to expand the polynomial $\prod_{i \in [t]} (1 + x_i y)$, i.e., it takes $O(t \log^2 t)$ time to compute the coefficient of y^r of the polynomial $\prod_{i \in [t]} (1 + x_i y)$.

Note that we need to compute the probability $\mathbb{P}[\tilde{S} = S]$ in (4) efficiently. The main idea of the efficient implementation is to sample elements one by one based on conditioning on the chosen elements and unchosen elements and update the conditional probability distribution sequentially. Indeed, suppose that we are given a subset of chosen elements S with $|S| < k$ and a subset of unchosen elements T with $|T| < n - k$ which can be empty, then probability that the j th experiment with $j \in [n] \setminus (S \cup T)$ will be chosen is equal to

$$\mathbb{P}[j \text{ will be chosen} | S, T] = \frac{\hat{x}_j \left(\sum_{\tilde{S} \in \binom{[n] \setminus (S \cup T)}{k-1-|S|}} \prod_{\tau \in \tilde{S}} \hat{x}_\tau \right)}{\left(\sum_{\tilde{S} \in \binom{[n] \setminus (S \cup T)}{k-|S|}} \prod_{\tau \in \tilde{S}} \hat{x}_\tau \right)}.$$

In the above formula, the denominator and numerator can be computed efficiently based on Observation 1. Thus, we flip a coin with success rate equal to the above probability, which clearly has the following two outcomes: if j is chosen, then update $S := S \cup \{j\}$; otherwise, update $T := T \cup \{j\}$. Then, go to next iteration and repeat this procedure until $|S| = k$. By applying iterated conditional probability, the probability that S is chosen equal to $\frac{\prod_{j \in S} \hat{x}_j}{\sum_{\tilde{S} \in \binom{[n]}{k}} \prod_{j \in \tilde{S}} \hat{x}_j}$, i.e., (4) holds. The detailed implementation is shown in Algorithm 1. Note that the time complexity of Algorithm 1 is $O(n^2)$.

3.2. m -wise α -positively Correlated Probability Distributions In this subsection, we will introduce the main proof idea of the approximation guarantees, which is to analyze the probability distribution (4) and show that it is approximately positively correlated (see, e.g., [13]). The formal derivation will be in the next subsection.

Recall that a set of random variables X_1, \dots, X_n are pairwise positively correlated if for each $i, j \in [n]$, we have $\mathbb{E}[X_i X_j] \geq \mathbb{E}[X_i] \cdot \mathbb{E}[X_j]$, which for $\{0, 1\}$ -valued random variables translates to $\mathbb{P}[X_i = 1, X_j = 1] \geq \mathbb{P}[X_i = 1] \cdot \mathbb{P}[X_j = 1]$. This definition aims to capture settings where random variables are more likely to take similar values than independent random variables with the same marginals. More generally, given an integer m , $\{0, 1\}$ -valued random variables X_1, \dots, X_n are called m -wise positively correlated random variables if $\mathbb{P}[X_i = 1 \forall i \in T] \geq \prod_{i \in T} \mathbb{P}[X_i = 1]$ for all $T \subseteq [n]$ where $|T| = m$. Below we provide an generalized definition of positive correlation that is crucial to our analysis.

Definition 1 Given $x \in [0, 1]^n$ such that $\sum_{i \in [n]} x_i = k$ for integer $k \geq 1$, let μ be a probability distribution on subsets of $[n]$ of size k . Let X_1, \dots, X_n denote the indicator random variables, thus $X_i = 1$ if $i \in \tilde{S}$ and 0 otherwise for each $i \in [n]$ where random set \tilde{S} of size- k is sampled from μ . Then X_1, \dots, X_n are m -wise α -positively correlated for some $\alpha \in [0, 1]$ if for each $T \subseteq [n]$ such that $|T| = m$, we have

$$\mathbb{P}[X_i = 1 \forall i \in T] = \mathbb{P}[T \subseteq \tilde{S}] \geq \alpha^m \prod_{i \in T} x_i.$$

Algorithm 1 Efficient Implementation of Sampling Procedure (4) with Constant Factor Approximation

- 1: Suppose (\hat{x}, \hat{w}) is an optimal solution to the convex relaxation (3) with $B = [0, 1]$, where $\hat{x} \in [0, 1]^n$ with $\sum_{i \in [n]} \hat{x}_i = k$ and $\hat{w} = f(\hat{x})$
 - 2: Initialize chosen set $\tilde{S} = \emptyset$ and unchosen set $T = \emptyset$
 - 3: Two factors: $A_1 = \sum_{\tilde{S} \in \binom{[n]}{k}} \prod_{i \in \tilde{S}} \hat{x}_i, A_2 = 0$
 - 4: **for** $j = 1, \dots, n$ **do**
 - 5: **if** $|\tilde{S}| == k$ **then**
 - 6: **break**
 - 7: **else if** $|T| = n - k$ **then**
 - 8: $\tilde{S} = [n] \setminus T$
 - 9: **break**
 - 10: **end if**
 - 11: Let $A_2 = \left(\sum_{\tilde{S} \in \binom{[n] \setminus (\tilde{S} \cup T)}{k-1-|\tilde{S}|}} \prod_{\tau \in \tilde{S}} \hat{x}_\tau \right)$
 - 12: Sample a $(0, 1)$ uniform random variable U
 - 13: **if** $\hat{x}_j A_2 / A_1 \geq U$ **then**
 - 14: Add j to set \tilde{S}
 - 15: $A_1 = A_2$
 - 16: **else**
 - 17: Add j to set T
 - 18: $A_1 = A_1 - \hat{x}_j A_2$
 - 19: **end if**
 - 20: **end for**
 - 21: Output \tilde{S}
-

With a slight abuse of notation, we call the distribution μ to be m -wise α -positively correlated with respect to x if the above condition is satisfied. Observe that if $\alpha = 1$, then the above definition implies that the random variables X_1, \dots, X_n are m -wise positively correlated.

The following lemma shows the crucial role played by m -wise approximate positively correlated distributions in the design of algorithms for D -optimal design.

Lemma 1 Suppose (\hat{x}, \hat{w}) is an optimal solution to the convex relaxation (3). Then for any $\alpha \in (0, 1]$, if there exists an efficiently computable distribution that is m -wise α -positively correlated with respect to

$\hat{\mathbf{x}}$, then the D -optimal design problem has a randomized α -approximation algorithm, i.e.,

$$\left\{ \mathbb{E} \left[\det \left(\sum_{i \in \tilde{S}} \mathbf{a}_i \mathbf{a}_i^\top \right) \right] \right\}^{\frac{1}{m}} \geq \alpha w^*$$

where random set \tilde{S} with size k is the output of the approximation algorithm.

The proof of Lemma 1 relies on the polynomial formulation of matrix determinant and convex relaxation of the D -optimal design problem. We show that a m -wise α -positively correlated distribution leads to a randomized algorithm for the D -optimal design problem that approximates each of the coefficients in the polynomial formulation. Note that the approximation ratio α in Lemma 1 only holds in the sense of expectation. Therefore, one might need to derandomize the algorithm to achieve the approximation ratio.

Before proving Lemma 1, we would like to introduce some useful results below. The following lemmas follow from Cauchy-Binet formula [12] and use the fact that a matrix's determinant is polynomial in entries of the matrix. Interested readers can find the proofs in the appendix.

Lemma 2 Suppose $\mathbf{a}_i \in \mathbb{R}^m$ for $i \in T$ with $|T| \geq m$, then

$$\det \left(\sum_{i \in T} \mathbf{a}_i \mathbf{a}_i^\top \right) = \sum_{S \in \binom{T}{m}} \det \left(\sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top \right). \quad (5)$$

Proof: See Appendix A.1. □

Lemma 3 For any $\mathbf{x} \in [0, 1]^n$, then

$$\det \left(\sum_{i \in [n]} x_i \mathbf{a}_i \mathbf{a}_i^\top \right) = \sum_{S \in \binom{[n]}{m}} \prod_{i \in S} x_i \det \left(\sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top \right). \quad (6)$$

Proof: See Appendix A.2. □

Now we are ready to prove the main Lemma 1.

Proof: (Proof of Lemma 1) Note that $(\hat{\mathbf{x}}, \hat{w})$ is an optimal solution to (3), and the distribution μ given by Lemma 1 for this $\hat{\mathbf{x}}$ which satisfies the conditions of Definition 1. We now consider the randomized algorithm that samples a random set \tilde{S} from this distribution μ . We show this randomized algorithm satisfies the guarantee claimed in the Lemma. All expectations and probabilities of events are under the probability measure μ and for simplicity, we drop it from the notation.

Since $\left[\det \left(\sum_{i \in [n]} \hat{x}_i \mathbf{a}_i \mathbf{a}_i^\top \right) \right]^{\frac{1}{m}}$ is at least as large as the optimal value to D -optimal design problem (3) with $B = [0, 1]$, we only need to show that

$$\left\{ \mathbb{E} \left[\det \left(\sum_{i \in \tilde{S}} \mathbf{a}_i \mathbf{a}_i^\top \right) \right] \right\}^{\frac{1}{m}} \geq \alpha \left[\det \left(\sum_{i \in [n]} \hat{x}_i \mathbf{a}_i \mathbf{a}_i^\top \right) \right]^{\frac{1}{m}},$$

or equivalently

$$\mathbb{E} \left[\det \left(\sum_{i \in \tilde{S}} \mathbf{a}_i \mathbf{a}_i^\top \right) \right] \geq \alpha^m \det \left(\sum_{i \in [n]} \hat{x}_i \mathbf{a}_i \mathbf{a}_i^\top \right). \quad (7)$$

This indeed holds since

$$\begin{aligned} \mathbb{E} \left[\det \left(\sum_{i \in \tilde{S}} \mathbf{a}_i \mathbf{a}_i^\top \right) \right] &= \sum_{S \in \binom{[n]}{k}} \mathbb{P}[\tilde{S} = S] \det \left(\sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top \right) = \sum_{S \in \binom{[n]}{k}} \mathbb{P}[\tilde{S} = S] \sum_{T \in \binom{S}{m}} \det \left(\sum_{i \in T} \mathbf{a}_i \mathbf{a}_i^\top \right) \\ &= \sum_{T \in \binom{[n]}{m}} \mathbb{P}[T \subseteq \tilde{S}] \det \left(\sum_{i \in T} \mathbf{a}_i \mathbf{a}_i^\top \right) \geq \alpha^m \sum_{T \in \binom{[n]}{m}} \prod_{i \in T} \hat{x}_i \det \left(\sum_{i \in T} \mathbf{a}_i \mathbf{a}_i^\top \right) = \alpha^m \det \left(\sum_{i \in [n]} \hat{x}_i \mathbf{a}_i \mathbf{a}_i^\top \right) \end{aligned}$$

where the first equality is because of definition of probability measure μ , the second equality is due to Lemma 3, the third equality is due to the interchange of summation, the first inequality is due to Definition 1 and the fourth equality is because of Lemma 3. \square

3.3. Analysis of Sampling Scheme In this subsection, we will analyze the proposed sampling procedure, i.e., deriving the approximation ratio α of sampling procedure (4) in Lemma 1. The key idea is to derive lower bound for the ratio

$$\frac{\mathbb{P}[T \subseteq \tilde{S}]}{\prod_{i \in T} \hat{x}_i}$$

for any $T \in \binom{[n]}{m}$, where $(\hat{\mathbf{x}}, \hat{w})$ is an optimal solution to (3) with $B = [0, 1]$.

By the definition of random set \tilde{S} in (4), the probability $\mathbb{P}[T \subseteq \tilde{S}]$ is equal to

$$\mathbb{P}[T \subseteq \tilde{S}] = \frac{\sum_{S \in \binom{[n]}{k}: T \subseteq S} \prod_{j \in S} \hat{x}_j}{\sum_{\tilde{S} \in \binom{[n]}{k}} \prod_{i \in \tilde{S}} \hat{x}_i}. \quad (8)$$

Observe that the denominator in (8) is a degree k polynomial that is invariant under any permutation of $[n]$. Moreover, the numerator is also invariant under any permutation of T as well as any permutation of $[n] \setminus T$. These observations allow us to use inequalities on symmetric polynomials and reduce the worst-case ratio of $\mathbb{P}[T \subseteq \tilde{S}]$ with $\prod_{i \in T} \hat{x}_i$ to a single variable optimization problem as shown in the following proposition. We then analyze the single variable optimization problem to prove the desired bound.

Proposition 1 Let $\tilde{\mathcal{S}}$ be the random set defined in (4). Then for any $T \subseteq [n]$ such that $|T| = m$, we have

$$\mathbb{P}[T \subseteq \tilde{\mathcal{S}}] \geq \frac{1}{g(m, n, k)} \prod_{i \in T} \hat{x}_i := \alpha^m \prod_{i \in T} \hat{x}_i,$$

where

$$g(m, n, k) = \max_y \left\{ \sum_{\tau=0}^m \frac{\binom{n-m}{k-\tau}}{(n-m)^{m-\tau}} \frac{\binom{m}{\tau}}{m^\tau} (k-y)^{m-\tau} (y)^\tau : \frac{mk}{n} \leq y \leq m \right\}. \quad (9)$$

Proof: See Section A.3. □

Next, we derive the upper bound of $g(m, n, k)$ in (9), which is a single-variable optimization problem. To derive the desired results, we first observe that for any given (m, k) with $m \leq k$, $g(m, n, k)$ is monotone non-decreasing in n . This motivates us to find an upper bound on $\lim_{n \rightarrow \infty} g(m, n, k)$, which leads to the proposition below.

Proposition 2 For any $n \geq k \geq m$, we have

$$\alpha^{-1} = [g(m, n, k)]^{\frac{1}{m}} \leq \lim_{\tau \rightarrow \infty} [g(m, \tau, k)]^{\frac{1}{m}} \leq \min \left\{ e, 1 + \frac{k}{k-m+1} \right\}. \quad (10)$$

Proof: See Appendix A.4. □

Finally, we present our first approximation result below.

Theorem 1 For any positive integers $m \leq k \leq n$, Algorithm 1 yields $\frac{1}{e}$ -approximation for the D -optimal design problem.

Proof: The result directly follows from Lemma 1 and Proposition 2 given that $n \geq k \geq m$. □

We also note that when k is large enough, Algorithm 1 is a near 0.5-approximation.

Corollary 1 Given $\epsilon \in (0, 1)$ and positive integers $m \leq k \leq n$, if $k \geq \frac{m-1}{2\epsilon}$, then Algorithm 1 yields $(0.5 - \epsilon)$ -approximation for the D -optimal design problem.

Proof: For any $\epsilon \in (0, 1)$, from Proposition 2, let the lower bound approximation ratio $\alpha \geq [\min\{e, 1 + \frac{k}{k-m+1}\}]^{-1} \geq 0.5 - \epsilon$, or equivalently, let

$$1 + \frac{k}{k-m+1} \leq \frac{1}{0.5 - \epsilon}.$$

Then the conclusion follows. □

3.4. Deterministic Implementation The approximation ratios presented in the previous subsection only hold in the sense of expectation. In this subsection, we will overcome this issue and present a deterministic Algorithm 2 with the same approximation guarantees. The key idea is to derandomize Algorithm 2 using the method of conditional expectation (cf., [35]), and the main challenge is how to compute the conditional expectation efficiently. We next will show that it can be done by evaluating a determinant of a $n \times n$ matrix whose entries are linear polynomials in three indeterminates.

In this deterministic Algorithm 2, suppose we are given a subset $S, P \subseteq [n]$, where S is a chosen subset and P is a unchosen set such that $|S| = s \leq k$. Then the expectation of m th power of objective function given S and P is

$$\begin{aligned}
H(S, P) &:= \mathbb{E} \left[\det \left(\sum_{i \in \tilde{S}} \mathbf{a}_i \mathbf{a}_i^\top \right) \middle| S \subseteq \tilde{S}, P \cap \tilde{S} = \emptyset \right] \\
&= \sum_{U \in \binom{[n] \setminus (S \cup P)}{k-s}} \frac{\prod_{j \in U} \hat{x}_j}{\sum_{\tilde{U} \in \binom{[n] \setminus (S \cup P)}{k-s}} \prod_{i \in \tilde{U}} \hat{x}_i} \det \left(\sum_{i \in U} \mathbf{a}_i \mathbf{a}_i^\top + \sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top \right) \\
&= \left(\sum_{\tilde{U} \in \binom{[n] \setminus (S \cup P)}{k-s}} \prod_{i \in \tilde{U}} \hat{x}_i \right)^{-1} \sum_{U \in \binom{[n] \setminus (S \cup P)}{k-s}} \prod_{j \in U} \hat{x}_j \sum_{R \in \binom{U \cup S}{m}} \det \left(\sum_{i \in R} \mathbf{a}_i \mathbf{a}_i^\top \right) \\
&= \frac{\sum_{R \in \binom{[n] \setminus P}{m}, r: |R \setminus S| \leq k-s} \prod_{j \in R \setminus S} \hat{x}_j \det \left(\sum_{i \in R} \mathbf{a}_i \mathbf{a}_i^\top \right) \sum_{W \in \binom{[n] \setminus (S \cup P \cup R)}{k-s-r}} \prod_{j \in W} \hat{x}_j}{\sum_{\tilde{U} \in \binom{[n] \setminus (S \cup P)}{k-s}} \prod_{i \in \tilde{U}} \hat{x}_i}, \tag{11}
\end{aligned}$$

where the second equality is a direct computing of the conditional probability, the third equality is due to Lemma 2 and the last one is because of interchange of summation.

Note that the denominator in (11) can be computed efficiently according to Observation 1. Next, we show that the numerator in (11) can be also computed efficiently.

Proposition 3 Let matrix $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n]$. Consider the following function

$$F(t_1, t_2, t_3) = \det \left(\mathbf{I}_n + t_1 \text{diag}(\mathbf{y})^{\frac{1}{2}} \mathbf{A}^\top \mathbf{A} \text{diag}(\mathbf{y})^{\frac{1}{2}} + \text{diag}(\mathbf{y}) \right), \tag{12}$$

where $t_1, t_2, t_3 \in \mathbb{R}$, $\mathbf{y} \in \mathbb{R}^n$ are indeterminate and

$$y_i = \begin{cases} t_3, & \text{if } i \in S \\ 0, & \text{if } i \in P \\ \hat{x}_i t_2, & \text{otherwise} \end{cases}.$$

Then, the coefficient of $t_1^m t_2^{k-s} t_3^s$ in $F(t_1, t_2, t_3)$ equals to

$$\sum_{R \in \binom{[n] \setminus P}{m}, r: |R \setminus S| \leq k-s} \prod_{j \in R \setminus S} \hat{x}_j \det \left(\sum_{i \in R} \mathbf{a}_i \mathbf{a}_i^\top \right) \sum_{W \in \binom{[n] \setminus (S \cup P \cup R)}{k-s-r}} \prod_{j \in W} \hat{x}_j. \tag{13}$$

Algorithm 2 Derandomization of Algorithm 1

-
- 1: Suppose $(\hat{\mathbf{x}}, \hat{w})$ is an optimal solution to the convex relaxation (3) with $B = [0, 1]$, where $\hat{\mathbf{x}} \in [0, 1]^n$ with $\sum_{i \in [n]} \hat{x}_i = k$ and $\hat{w} = f(\hat{\mathbf{x}})$
 - 2: Initialize chosen set $S = \emptyset$, unchosen set $P = \emptyset$, and $j = 1$
 - 3: **do**
 - 4: Compute $H(S \cup j, P), H(S, P \cup j)$ as defined in (11), and their denominators and numerators can be computed by Observation 1 and Proposition 3, respectively
 - 5: **if** $H(S \cup j, P) > H(S, P \cup j)$ **then**
 - 6: Add j to set S
 - 7: **else**
 - 8: Add j to set P
 - 9: **end if**
 - 10: $j = j + 1$
 - 11: **while** $|S| < k$ and $j \leq n$
 - 12: Output S
-

Proof: First of all, we can rewrite $F(t_1, t_2, t_3)$ as

$$\begin{aligned} F(t_1, t_2, t_3) &= \det(\mathbf{I}_n + \text{diag}(\mathbf{y})) \det\left(\mathbf{I}_n + t_1 \text{diag}(\mathbf{e} + \mathbf{y})^{-\frac{1}{2}} \text{diag}(\mathbf{y})^{\frac{1}{2}} \mathbf{A}^\top \mathbf{A} \text{diag}(\mathbf{y})^{\frac{1}{2}} \text{diag}(\mathbf{e} + \mathbf{y})^{-\frac{1}{2}}\right) \\ &= \prod_{i \in S} (1 + t_3) \prod_{i \in [n] \setminus (S \cup P)} (1 + \hat{x}_i t_2) \det(\mathbf{I}_n + t_1 \mathbf{W}^\top \mathbf{W}) \end{aligned}$$

where the i th column of matrix \mathbf{W} is

$$\mathbf{W}_i = \begin{cases} \sqrt{\frac{t_3}{1+t_3}} \mathbf{a}_i, & \text{if } i \in S \\ \mathbf{0}, & \text{if } i \in P \\ \sqrt{\frac{\hat{x}_i t_2}{1+\hat{x}_i t_2}} \mathbf{a}_i, & \text{otherwise} \end{cases}.$$

Note that the coefficient of t_1^m in $\det(\mathbf{I}_n + t_1 \mathbf{W}_{[n] \setminus P}^\top \mathbf{W}_{[n] \setminus P})$ is equal to the one of $\prod_{i \in [n] \setminus P} (1 + t_1 \Lambda_i)$, where $\{\Lambda_i\}_{i \in [n] \setminus P}$ are the eigenvalues of $\mathbf{W}^\top \mathbf{W}$. Thus, the coefficient of t_1^m is

$$\sum_{R \in \binom{[n] \setminus P}{m}} \prod_{i \in R} \Lambda_i = \sum_{R \in \binom{[n] \setminus P}{m}} \det((\mathbf{W}^\top \mathbf{W})_{R,R}) = \sum_{R \in \binom{[n] \setminus P}{m}} \det\left(\sum_{i \in R} \mathbf{w}_i \mathbf{w}_i^\top\right) = \sum_{R \in \binom{[n] \setminus P}{m}} \det\left(\sum_{i \in R} \mathbf{w}_i \mathbf{w}_i^\top\right)$$

where \mathbf{P}_{R_1, R_2} denotes a submatrix of \mathbf{P} with rows and columns from sets R_1, R_2 , the first equality is due to the property of the eigenvalues (Theorem 1.2.12 in [19]), and the second inequality is because the length of each column of \mathbf{W} is m , and the third equality is because the determinant of singular matrix is 0.

Therefore, the coefficient of $t_1^m t_2^{k-s} t_3^s$ in $F(t_1, t_2, t_3)$ is equivalent to the one of

$$\prod_{i \in S} (1 + t_3) \prod_{i \in [n] \setminus (S \cup P)} (1 + \hat{x}_i t_2) \sum_{R \in \binom{[n] \setminus P}{m}} \det \left(\sum_{i \in R} \mathbf{w}_i \mathbf{w}_i^\top \right).$$

By Lemma 3 with $n = m$ and the definition of matrix \mathbf{W} , the coefficient of $t_1^m t_2^{k-s} t_3^s$ in $F(t_1, t_2, t_3)$ is further equivalent to the one of

$$\begin{aligned} & \prod_{i \in S} (1 + t_3) \prod_{i \in [n] \setminus (S \cup P)} (1 + \hat{x}_i t_2) \sum_{R \in \binom{[n] \setminus P}{m}} t_1^m \prod_{j \in R \setminus S} \frac{\hat{x}_j}{1 + t_2 \hat{x}_j} \prod_{j \in R \cap S} \frac{t_3}{1 + t_3} \det \left(\sum_{i \in R} \mathbf{a}_i \mathbf{a}_i^\top \right) \\ &= t_1^m \sum_{R \in \binom{[n] \setminus P}{m}} t_2^{|R \setminus S|} t_3^{|R \cap S|} (1 + t_3)^{|S \setminus R|} \prod_{i \in [n] \setminus (S \cup R)} (1 + \hat{x}_i t_2) \prod_{j \in R \setminus S} \hat{x}_j \det \left(\sum_{i \in R} \mathbf{a}_i \mathbf{a}_i^\top \right) \end{aligned}$$

which is equal to (13) by collecting coefficients of $t_1^m t_2^{k-s} t_3^s$. \square

Note that the characteristic function of a matrix can be computed efficiently by using Faddeev-LeVerrier algorithm [20]. Thus, the polynomial $F(t_1, t_2, t_3)$ is efficiently computable with time complexity of $O(n^4)$.

Algorithm 2 proceeds as follows. We start with an empty subset S of chosen elements and an empty subset P of unchosen elements, and for each $j \in [n] \setminus (S \cup P)$, we compute two expected m th power of objective functions that j will be chosen $H(S \cup j, P)$ or j will be not chosen $H(S, P \cup j)$. We add j to S if $H(S \cup j, P) > H(S, P \cup j)$. Otherwise, we add j to P . Then go to next iteration. This procedure will terminate if $|S| = k$ or $S \cup P = [n]$. Note that Algorithm 2 requires $O(n)$ evaluations of function $H(S, P)$, thus its the time complexity is $O(n^5)$. Hence, in practice, we recommend Algorithm 1 due to its shorter running time.

The approximation results for Algorithm 2 are identical to Theorem 1 and Corollary 1, which are summarized as follows:

Theorem 2 For any positive integers $m \leq k \leq n$,

- (i) deterministic Algorithm 2 is efficiently computable and yield $\frac{1}{e}$ -approximation for the D -optimal design problem; and
- (ii) given $\epsilon \in (0, 1)$, if $k \geq \frac{m-1}{2\epsilon}$, then deterministic Algorithm 2 yields $(0.5 - \epsilon)$ -approximation for the D -optimal design problem.

4. Improving Approximation Bound in Asymptotic Regime In this section, we propose another sampling Algorithm 3 which achieves asymptotic optimality, i.e. the output of Algorithm 3 is close to be optimal when $k/m \rightarrow \infty$. We also show the derandomization of Algorithm 3. Similar to previous section, in this section, we consider D -optimal design problem without repetitions, i.e., in (2) and (3), we let $B = [0, 1]$.

In Algorithm 3, suppose that $(\hat{\mathbf{x}}, \hat{w})$ is an optimal solution to the convex relaxation (3) with $B = [0, 1]$, $\epsilon \in (0, 1)$ is a positive threshold and \mathcal{N} is a random permutation of $[n]$. Then for each $j \in \mathcal{N}$, we select j with probability $\frac{x_j}{1+\epsilon}$, and let $\tilde{\mathcal{S}}$ be the set of selected elements. If $|\tilde{\mathcal{S}}| < k$, then we can add $k - |\tilde{\mathcal{S}}|$ more elements from $[n] \setminus \tilde{\mathcal{S}}$. On the other hand, if $|\tilde{\mathcal{S}}| > k$, then we repeat the sampling procedure. Algorithm 3 has time complexity $O(n)$. In addition, note that the difference between Algorithm 1 and Algorithm 3 is that in Algorithm 3, we inflate the probability of choosing j th experiment by $\frac{1}{1+\epsilon}$. This condition guarantees that when $k \gg m$, according to concentration inequality, the probability of size- m subset T to be chosen will be nearly equal to $\prod_{j \in T} \frac{x_j}{1+\epsilon}$.

Algorithm 3 Asymptotic Sampling Algorithm

- 1: Suppose $(\hat{\mathbf{x}}, \hat{w})$ is an optimal solution to the convex relaxation (3) with $B = [0, 1]$, where $\hat{\mathbf{x}} \in [0, 1]^n$ with $\sum_{i \in [n]} \hat{x}_i = k$ and $\hat{w} = f(\hat{\mathbf{x}})$
 - 2: Initialize $\tilde{\mathcal{S}} = \emptyset$ and a positive number $\epsilon > 0$
 - 3: **do**
 - 4: Let set \mathcal{N} be a random permutation set of $\{1, \dots, n\}$
 - 5: **for** $j \in \mathcal{N}$ **do**
 - 6: Sample a $(0, 1)$ uniform random variable U
 - 7: **if** $U \leq \frac{\hat{x}_j}{1+\epsilon}$ **then**
 - 8: Add j to set $\tilde{\mathcal{S}}$
 - 9: **end if**
 - 10: **end for**
 - 11: **while** $|\tilde{\mathcal{S}}| > k$
 - 12: **if** $|\tilde{\mathcal{S}}| < k$ **then** ▷ Greedy step to enforce $|\tilde{\mathcal{S}}| = k$
 - 13: Let $j^* \in \arg \max_{j \in [n] \setminus \tilde{\mathcal{S}}} \left[\det \left(\sum_{i \in \tilde{\mathcal{S}}} \mathbf{a}_i \mathbf{a}_i^\top + \mathbf{a}_j \mathbf{a}_j^\top \right) \right]^{\frac{1}{m}}$
 - 14: Add j^* to set $\tilde{\mathcal{S}}$
 - 15: **end if**
 - 16: Output $\tilde{\mathcal{S}}$
-

4.1. Analysis of Sampling Algorithm 3 To analyze sampling Algorithm 3, we first show the following probability bound. The key idea is to prove the lower bound $\frac{1}{\prod_{i \in T} \hat{x}_i} \mathbb{P} \left\{ T \subseteq \tilde{\mathcal{S}} \mid |\tilde{\mathcal{S}}| \leq k \right\}$ by using Chernoff Bound [15].

Lemma 4 Let $\epsilon > 0$ and $\tilde{\mathcal{S}} \subseteq [n]$ be a random set output from Algorithm 3. Given $T \subseteq [n]$ with $|T| = m \leq n$, then we have

$$\alpha^m := \frac{1}{\prod_{i \in T} \hat{x}_i} \mathbb{P} \left\{ T \subseteq \tilde{\mathcal{S}} \mid |\tilde{\mathcal{S}}| \leq k \right\} \geq (1 + \epsilon)^{-m} \left(1 - e^{-\frac{(\epsilon k - (1+\epsilon)m)^2}{k(2+\epsilon)(1+\epsilon)}} \right), \quad (14)$$

where α is in Definition 1. In addition, when $k \geq \frac{4m}{\epsilon} + \frac{12}{\epsilon^2} \log(\frac{1}{\epsilon})$, then

$$\alpha^m \geq (1 - \epsilon)^m. \quad (15)$$

Proof: We note that $\tilde{\mathcal{S}} \subseteq [n]$ is a random set, where each $i \in [n]$ is independently sampled according to Bernoulli random variable X_i with the probability of success $\frac{\hat{x}_i}{1+\epsilon}$. According to Definition 1 and by ignoring the greedy procedure Algorithm 3, it is sufficient to derive the lower bound of $\frac{1}{\prod_{i \in T} \hat{x}_i} \mathbb{P} \left\{ T \subseteq \tilde{\mathcal{S}} \mid |\tilde{\mathcal{S}}| \leq k \right\}$, i.e.,

$$\frac{\mathbb{P} \left\{ T \subseteq \tilde{\mathcal{S}} \mid |\tilde{\mathcal{S}}| \leq k \right\}}{\prod_{i \in T} \hat{x}_i} = \frac{\mathbb{P} \left\{ T \subseteq \tilde{\mathcal{S}}, |\tilde{\mathcal{S}}| \leq k \right\}}{\prod_{i \in T} \hat{x}_i \mathbb{P} \left\{ |\tilde{\mathcal{S}}| \leq k \right\}} \geq (1 + \epsilon)^{-m} \mathbb{P} \left\{ \sum_{i \in [n] \setminus T} X_i \leq k - m \right\}$$

where the first inequality is due to $\mathbb{P}\{|\tilde{\mathcal{S}}| \leq k\} \leq 1$.

Therefore, it is sufficient to bound the following probability

$$\mathbb{P} \left\{ \sum_{i \in [n] \setminus T} X_i \leq k - m \right\}.$$

Since $X_i \in \{0, 1\}$ for each $i \in [n]$, and $\mathbb{E}[\sum_{i \in [n] \setminus T} X_i] = \frac{1}{1+\epsilon} \sum_{i \in [n] \setminus T} \hat{x}_i$. According to Chernoff bound [15], we have

$$\mathbb{P} \left\{ \sum_{i \in [n] \setminus T} X_i > (1 + \bar{\epsilon}) \mathbb{E} \left[\sum_{i \in [n] \setminus T} X_i \right] \right\} \leq e^{-\frac{\bar{\epsilon}^2}{2 + \bar{\epsilon}} \mathbb{E}[\sum_{i \in [n] \setminus T} X_i]},$$

where $\bar{\epsilon}$ is a positive constant. Therefore, by choosing $\bar{\epsilon} = \frac{(1+\epsilon)(k-m)}{\sum_{i \in [n] \setminus T} \hat{x}_i} - 1$, we have

$$(1 + \epsilon)^{-m} \mathbb{P} \left\{ \sum_{i \in [n] \setminus T} X_i \leq k - m \right\} \geq (1 + \epsilon)^{-m} \left(1 - e^{-\frac{\bar{\epsilon}^2 \sum_{i \in [n] \setminus T} \hat{x}_i}{(2 + \bar{\epsilon})(1 + \epsilon)}} \right). \quad (16)$$

Note that $k - m \leq \sum_{i \in [n] \setminus T} X_i \leq k$, $\epsilon - (1 + \epsilon) \frac{m}{k} \leq \bar{\epsilon} \leq \epsilon$ and $\epsilon k - (1 + \epsilon)m \leq \bar{\epsilon} \sum_{i \in [n] \setminus T} \hat{x}_i \leq \epsilon(k - m)$. Suppose that $k \geq \frac{1+\epsilon}{\epsilon} m$, then the left-hand side of (16) can be further lower bounded as

$$(1 + \epsilon)^{-m} \left(1 - e^{-\frac{\bar{\epsilon}^2 \sum_{i \in [n] \setminus T} \hat{x}_i}{(2 + \bar{\epsilon})(1 + \epsilon)}} \right) \geq (1 + \epsilon)^{-m} \left(1 - e^{-\frac{(\epsilon k - (1 + \epsilon)m)^2}{k(2 + \epsilon)(1 + \epsilon)}} \right).$$

To prove(15), it remains to show

$$1 - e^{-\frac{(\epsilon k - (1 + \epsilon)m)^2}{k(2 + \epsilon)(1 + \epsilon)}} \geq ((1 - \epsilon)(1 + \epsilon))^m,$$

or equivalently,

$$\log [1 - (1 - \epsilon^2)^m] \geq -\frac{(\epsilon k - (1 + \epsilon)m)^2}{k(2 + \epsilon)(1 + \epsilon)}, \quad (17)$$

which holds if

$$k \geq \left(1 + \frac{1}{\epsilon}\right) \left(2m - \left(1 + \frac{2}{\epsilon}\right) \log [1 - (1 - \epsilon^2)^m]\right).$$

We note that $-\log [1 - (1 - \epsilon^2)^m]$ is non-increasing over $m \geq 1$, therefore is upper bounded by $2 \log(\frac{1}{\epsilon})$. Hence, (17) holds if $k \geq \frac{4m}{\epsilon} + \frac{12}{\epsilon^2} \log(\frac{1}{\epsilon})$. \square

Finally, we state our main approximation result as below.

Theorem 3 For any integers $m \leq k \leq n$ and $\epsilon \in (0, 1)$, if $k \geq \frac{4m}{\epsilon} + \frac{12}{\epsilon^2} \log(\frac{1}{\epsilon})$, then Algorithm 3 is a $(1 - \epsilon)$ -approximation for the D -optimal design problem.

Proof: The result directly follows from Lemmas 1 and 4. \square

4.2. Deterministic Implementation Similar to Section 3.4, the approximation ratios presented in the previous subsection only hold in the sense of expectation. In this subsection, we will overcome this issue and present a deterministic Algorithm 3 with the same approximation guarantees.

In this deterministic Algorithm 4, S denotes a subset such that $|S| = s \leq k$ and P denotes the unchosen set. Then the expectation of m th power of objective function given S and P is

$$\begin{aligned} H(S, P) &:= \mathbb{E} \left[\det \left(\sum_{i \in \tilde{S}} \mathbf{a}_i \mathbf{a}_i^\top \right) \middle| S \subseteq \tilde{S}, |\tilde{S}| \leq k, \tilde{S} \cap P = \emptyset \right] \\ &= \frac{\sum_{\kappa=s}^k \sum_{U \in \binom{[n] \setminus (S \cup P)}{\kappa-s}} \prod_{j \in U} \frac{\hat{x}_j}{1+\epsilon} \prod_{j \in [n] \setminus (S \cup P \cup U)} \left(1 - \frac{\hat{x}_j}{1+\epsilon}\right) \det \left(\sum_{i \in U} \mathbf{a}_i \mathbf{a}_i^\top + \sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top \right)}{\sum_{\kappa=1}^k \sum_{\bar{U} \in \binom{[n] \setminus (S \cup P)}{\kappa-s}} \prod_{i \in \bar{U}} \frac{\hat{x}_i}{1+\epsilon} \prod_{j \in [n] \setminus (S \cup P \cup \bar{U})} \left(1 - \frac{\hat{x}_j}{1+\epsilon}\right)} \\ &= \left(\sum_{\kappa=s}^k \sum_{\bar{U} \in \binom{[n] \setminus (S \cup P)}{\kappa-s}} \prod_{i \in \bar{U}} \frac{\hat{x}_i}{1+\epsilon - \hat{x}_i} \right)^{-1} \sum_{\kappa=s}^k \sum_{U \in \binom{[n] \setminus (S \cup P)}{\kappa-s}} \prod_{j \in U} \frac{\hat{x}_j}{1+\epsilon - \hat{x}_j} \sum_{R \in \binom{U \cup S}{m}} \det \left(\sum_{i \in R} \mathbf{a}_i \mathbf{a}_i^\top \right) \\ &= \frac{\sum_{\kappa=s}^k \sum_{R \in \binom{[n] \setminus P}{m}, \tau := |R \setminus S| \leq \kappa-s} \prod_{j \in R \setminus S} \frac{\hat{x}_j}{1+\epsilon - \hat{x}_j} \det \left(\sum_{i \in R} \mathbf{a}_i \mathbf{a}_i^\top \right) \sum_{W \in \binom{[n] \setminus (S \cup P \cup R)}{\kappa-s-\tau}} \prod_{j \in W} \frac{\hat{x}_j}{1+\epsilon - \hat{x}_j}}{\sum_{\kappa=s}^k \sum_{\bar{U} \in \binom{[n] \setminus (S \cup P)}{\kappa-s}} \prod_{i \in \bar{U}} \frac{\hat{x}_i}{1+\epsilon - \hat{x}_i}}, \end{aligned} \quad (18)$$

where the second equality is a direct computing of the conditional probability, the third equality is due to Lemma 2, dividing both denominator and numerator by $\prod_{j \in [n] \setminus (S \cup P)} \left(1 - \frac{\hat{x}_j}{1+\epsilon}\right)$ and the convention, $\text{set} \binom{S}{\tau} = \emptyset$ if $\tau > |S|$ or $\tau < 0$ and the fourth one is because of interchange of summation.

Note that the κ th entry with $\kappa \in \{s, \dots, k\}$ of the denominator in (18) can be computed efficiently according to Observation 1 by letting $x_i := \frac{\hat{x}_i}{1+\epsilon - \hat{x}_i}$. Meanwhile, κ th entry with $\kappa \in \{s, \dots, k\}$ of

the numerator in (11) can be also computed efficiently by Proposition 3 by letting $\hat{x}_i := \frac{\hat{x}_i}{1+\epsilon-\hat{x}_i}$. Therefore, the conditional expectation in (11) is efficiently computable

In summary, Algorithm 4 proceeds as follows. We start with an empty subset S of chosen elements and an empty subset P of unchosen elements, and for each $j \in [n] \setminus (S \cup P)$, we compute two expected m th power of objective functions that j will be chosen $H(S \cup j, P)$ or j will be not chosen $H(S, P \cup j)$. We add j to S if $H(S \cup j, P) > H(S, P \cup j)$. Otherwise, we add j to P . Then go to next iteration. This procedure will terminate if $|S| = k$ or $S \cup P = [n]$. Similar to Algorithm 2, the time complexity of Algorithm 4 is $O(n^5 k)$. Hence, in practice, we recommend the more efficient Algorithm 3.

The approximation result for Algorithm 4 is identical to Theorem 3, which is summarized as follows:

Theorem 4 For any positive integers $m \leq k \leq n$ and $\epsilon \in (0, 1)$, deterministic Algorithm 4 is efficiently computable and yield $1 - \epsilon$ -approximation for the D -optimal design problem if $k \geq \frac{4m}{\epsilon} + \frac{12}{\epsilon^2} \log(\frac{1}{\epsilon})$.

Algorithm 4 Derandomization of Algorithm 3

- 1: Suppose $(\hat{\mathbf{x}}, \hat{w})$ is an optimal solution to the convex relaxation (3) with $B = [0, 1]$, where $\hat{\mathbf{x}} \in [0, 1]^n$ with $\sum_{i \in [n]} \hat{x}_i = k$ and $\hat{w} = f(\hat{\mathbf{x}})$
 - 2: Initialize chosen set $S = \emptyset$, unchosen set $P = \emptyset$, and $j = 1$
 - 3: **do**
 - 4: Compute $H(S \cup j, P)$, $H(S, P \cup j)$ as defined in (18), and their denominators and numerators can be computed by Observation 1 and Proposition 3, respectively
 - 5: **if** $H(S \cup j, P) > H(S, P \cup j)$ **then**
 - 6: Add j to set S
 - 7: **else**
 - 8: Add j to set P
 - 9: **end if**
 - 10: $j = j + 1$
 - 11: **while** $|S| < k$ and $j \leq n$
 - 12: Output S
-

5. Approximation Algorithm for D -optimal Design Problem with Repetitions In this section, we consider the D -optimal design problem with repetitions, i.e., we let $B = \mathbb{R}_+$ in (2) and (3). We will propose a new analysis of the algorithm proposed by [27], derive its approximation ratio

and show its deterministic implementation. Again, in this section, we also let $(\hat{\mathbf{x}}, \hat{w})$ be an optimal solution to the convex relaxation (3), where $\hat{\mathbf{x}} \in \mathbb{R}_+^n$ with $\sum_{i \in [n]} \hat{x}_i = k$ and $\hat{w} = f(\hat{\mathbf{x}})$. Since the set of all nonnegative rational vectors is dense in the set of all nonnegative real vectors, thus without loss of generality, we assume that $\hat{\mathbf{x}}$ is a nonnegative rational vector (i.e., $\hat{\mathbf{x}} \in \mathbb{Q}_+^n$).

In [27], the author suggested to obtain k -sample set $\tilde{\mathcal{S}}$ with replacement, i.e. $\tilde{\mathcal{S}}$ can be a multi-set. The sampling procedure can be separated into k steps. At each step, a sample s is selected with probability $\mathbb{P}\{s = i\} = \frac{\hat{x}_i}{k}$ (note that $\hat{\mathbf{x}} \in \mathbb{R}_+^n$ with $\sum_{i \in [n]} \hat{x}_i = k$). The detailed description is in Algorithm 5. This sampling procedure can be interpreted as follows: let $\{X_i\}_{i \in [n]}$ be independent Poisson random variables where X_i has arrival rate \hat{x}_i . We note that conditioning on total number of arrivals equal to k (i.e., $\sum_{i \in [n]} X_i = k$), the distribution of $\{X_i\}_{i \in [n]}$ is multinomial (cf., [1]), where there are k trials and the probability of i th entry to be chosen is $\frac{\hat{x}_i}{k}$. We terminate this sampling procedure if the total number of arrivals equals to k . Note that the time complexity of Algorithm 5 is $O(n)$.

Algorithm 5 Sampling Algorithm for D -optimal Design with Repetitions

- 1: Suppose $(\hat{\mathbf{x}}, \hat{w})$ is an optimal solution to the convex relaxation (3) with $B = \mathbb{R}_+$, where $\hat{\mathbf{x}} \in \mathbb{Q}_+^n$ with $\sum_{i \in [n]} \hat{x}_i = k$ and $\hat{w} = f(\hat{\mathbf{x}})$
 - 2: Initialize chosen multi-set $\tilde{\mathcal{S}} = \emptyset$
 - 3: **for** $j = 1, \dots, k$ **do**
 - 4: Sample s from $[n]$ with probability $\mathbb{P}\{s = i\} = \frac{\hat{x}_i}{k}$
 - 5: Let $\tilde{\mathcal{S}} = \tilde{\mathcal{S}} \cup \{s\}$
 - 6: **end for**
 - 7: Output $\tilde{\mathcal{S}}$
-

To analyze Algorithm 5, let us consider another Algorithm 6, which turns out to be arbitrarily close to Algorithm 5. As $\hat{\mathbf{x}}$ is a nonnegative rational vector (i.e., $\hat{\mathbf{x}} \in \mathbb{Q}_+^n$), we let q be a common multiple of the denominators of rational numbers $\hat{x}_1, \dots, \hat{x}_n$, i.e. $q\hat{x}_1, \dots, q\hat{x}_n \in \mathbb{Z}_+$. Next, we create a multi-set \mathcal{A}_q , which contains $q\hat{x}_i$ copies of index i for each $i \in [n]$, i.e. $|\mathcal{A}_q| = qk$. Finally, we sample a subset $\tilde{\mathcal{S}}_q$ of k items from set \mathcal{A}_q uniformly, i.e. with probability $\binom{qk}{k}^{-1}$. The detailed description is in Algorithm 6. In this case, the sampling procedure has the following interpretation. Since sum of i.i.d. Bernoulli random variables is Binomial, hence we let $\{X'_i\}_{i \in [n]}$ be independent binomial random variables where X'_i has number of trials $q\hat{x}_i$ and probability of success $\frac{1}{q}$ for each $i \in [n]$. We terminate the sampling procedure if the total number of succeeded trials equals to k .

The following lemma shows that the probability distributions of outputs of Algorithms 5 and 6 can be arbitrarily close.

Algorithm 6 Approximation of Algorithm 5

- 1: Suppose $(\hat{\mathbf{x}}, \hat{w})$ is an optimal solution to the convex relaxation (3) with $B = \mathbb{R}^+$, where $\hat{\mathbf{x}} \in \mathbb{Q}_+^n$ with $\sum_{i \in [n]} \hat{x}_i = k$ and $\hat{w} = f(\hat{\mathbf{x}})$
- 2: Let q be a common multiple of the denominators of rational numbers $\hat{x}_1, \dots, \hat{x}_n$, i.e. $q\hat{x}_1, \dots, q\hat{x}_n \in \mathbb{Z}_+$
- 3: Duplicate $q\hat{x}_i$ copies of index i for each $i \in [n]$ as set \mathcal{A}_q , i.e. $|\mathcal{A}_q| = qk$
- 4: Sample a subset $\tilde{\mathcal{S}}_q$ of k items from set \mathcal{A}_q with probability $\binom{qk}{k}^{-1}$
- 5: Output $\tilde{\mathcal{S}}_q$

Lemma 5 Let $\tilde{\mathcal{S}}$ and $\tilde{\mathcal{S}}_q$ be outputs of Algorithms 5 and 6, respectively. Then $\tilde{\mathcal{S}}_q \xrightarrow{\mu} \tilde{\mathcal{S}}$, i.e. the probability distribution of $\tilde{\mathcal{S}}_q$ converges to $\tilde{\mathcal{S}}$ as $q \rightarrow \infty$.

Proof: Consider two classes of independent random variables $\{X_i\}_{i \in [n]}$, $\{X'_i\}_{i \in [n]}$, where X_i is Poisson random variable with arrival rate \hat{x}_i for each $i \in [n]$ and X'_i is binomial random variable with number of trials $q\hat{x}_i$ and probability of success $\frac{1}{q}$ for each $i \in [n]$, respectively.

Given a size- k multi-set \mathcal{R} with support $\text{Supp } \mathcal{R} \subseteq [n]$ and $M_{\mathcal{R}}(i)$ denoting the number of occurrences of i in \mathcal{R} , according to the description of Algorithm 6, we have

$$\begin{aligned} \mathbb{P}\{\tilde{\mathcal{S}} = \mathcal{R}\} &= \mathbb{P}\left\{X_i = M_{\mathcal{R}}(i), \forall i \in [n] \mid \sum_{i \in [n]} X_i = k\right\} = \frac{\mathbb{P}\left\{X_i = M_{\mathcal{R}}(i), \forall i \in [n], \sum_{i \in [n]} X_i = k\right\}}{\mathbb{P}\left\{\sum_{i \in [n]} X_i = k\right\}} \\ &= \mathbb{I}(|\mathcal{R}| = k) \frac{\prod_{i \in [n]} \mathbb{P}\{X_i = M_{\mathcal{R}}(i)\}}{\mathbb{P}\left\{\sum_{i \in [n]} X_i = k\right\}}, \end{aligned}$$

where the first equality is from the description of Algorithm 5, the second equality is by the definition of conditional probability, the third equality is because $\{X_i\}_{i \in [n]}$ are independent from each other and $\mathbb{I}(\cdot)$ denotes indicator function. Similarly, we also have

$$\mathbb{P}\{\tilde{\mathcal{S}}_q = \mathcal{R}\} = \mathbb{I}(|\mathcal{R}| = k) \frac{\prod_{i \in [n]} \mathbb{P}\{X'_i = M_{\mathcal{R}}(i)\}}{\mathbb{P}\left\{\sum_{i \in [n]} X'_i = k\right\}}.$$

Followed by the well-known Poisson limit theorem (cf. [30]), X_i and X'_i have the same distribution as $q \rightarrow \infty$ for any $i \in [n]$. Therefore,

$$\mathbb{P}\{\tilde{\mathcal{S}}_q = \mathcal{R}\} \rightarrow \mathbb{P}\{\tilde{\mathcal{S}} = \mathcal{R}\},$$

when $q \rightarrow \infty$, i.e., the outputs of Algorithm 5 and 6 have the same distribution when $q \rightarrow \infty$. \square

Now we are ready to present our approximation results for Algorithm 5. The proof idea is based on Lemma 5, i.e., we first analyze Algorithm 6 and apply its result to Algorithm 5 by letting $q \rightarrow \infty$.

Proposition 4 Let \tilde{S} and \tilde{S}_q be outputs of Algorithms 5 and 6, respectively. Then

$$\left(\mathbb{E}[(f(\tilde{S}_q))^m]\right)^{\frac{1}{m}} \geq \left(\mathbb{E}[(f(\tilde{S}))^m]\right)^{\frac{1}{m}} \geq g(m, k)^{-1} w^*,$$

where

$$g(m, k) = \left[\frac{(k-m)!k^m}{k!} \right]^{\frac{1}{m}} \leq \min \left\{ e, \frac{k}{k-m+1} \right\}. \quad (19)$$

Proof: We will first show the approximation ratio of Algorithm 6 and then apply it to Algorithm 5 by Lemma 5 when $q \rightarrow \infty$.

(i) Let (\bar{x}'_q, \bar{w}'_q) be output of Algorithm 6, and . Similar to the proof of Theorem 1, we have

$$\begin{aligned} \mathbb{E}[(\bar{w}'_q)^m] &= \sum_{S \in \binom{\mathcal{A}_q}{k}} \frac{1}{\binom{qk}{k}} \det \left(\sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top \right) = \frac{q^m}{\binom{qk}{k}} \sum_{S \in \binom{\mathcal{A}_q}{k}} \frac{1}{q^m} \det \left(\sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top \right) \\ &= \frac{q^m}{\binom{qk}{k}} \sum_{S \in \binom{\mathcal{A}_q}{k}} \frac{1}{q^m} \sum_{T \in \binom{S}{m}} \det \left(\sum_{i \in T} \mathbf{a}_i \mathbf{a}_i^\top \right) = \frac{q^m \binom{qk-m}{k-m}}{\binom{qk}{k}} \sum_{T \in \binom{\mathcal{A}_q}{m}} \frac{1}{q^m} \det \left(\sum_{i \in T} \mathbf{a}_i \mathbf{a}_i^\top \right) \\ &= \frac{q^m \binom{qk-m}{k-m}}{\binom{qk}{k}} \det \left(\sum_{i \in [n]} \hat{x}_i \mathbf{a}_i \mathbf{a}_i^\top \right) \geq \frac{k!}{(k-m)!k^m} [f(\hat{\mathbf{x}})]^m \geq \frac{k!}{(k-m)!k^m} (w^*)^m \end{aligned}$$

where the first and second equalities are due to Algorithm 6, the third equality is because of Lemma 2 and $k \geq m$, the fourth equality is due to interchange of summation, the fifth equality is because of the identity $\sum_{i \in [n]} \hat{x}_i \mathbf{a}_i \mathbf{a}_i^\top = \sum_{i \in \mathcal{A}_q} \frac{1}{q} \mathbf{a}_i \mathbf{a}_i^\top$, the first inequality holds because $\frac{\binom{qk-m}{k-m}}{\binom{qk}{k}} \geq 1$, and the last inequality is because $\hat{\mathbf{x}}$ is an optimal solution of the continuous relaxation.

(ii) From Lemma 5, we know that the output of Algorithm 6 has the same probability distribution as the output of Algorithm 5 when $q \rightarrow \infty$. Thus, we have

$$\begin{aligned} \mathbb{E}[(\bar{w})^m] &= \lim_{q \rightarrow \infty} \mathbb{E}[(\bar{w}'_q)^m] = \lim_{q \rightarrow \infty} \frac{q^m \binom{qk-m}{k-m}}{\binom{qk}{k}} \det \left(\sum_{i \in [n]} \hat{x}_i \mathbf{a}_i \mathbf{a}_i^\top \right) \\ &= \frac{k!}{(k-m)!k^m} \det \left(\sum_{i \in [n]} \hat{x}_i \mathbf{a}_i \mathbf{a}_i^\top \right) = \frac{k!}{(k-m)!k^m} [f(\hat{\mathbf{x}})]^m. \end{aligned}$$

(iii) Next, let

$$g(m, k) = \left[\frac{(k-m)!k^m}{k!} \right]^{\frac{1}{m}},$$

and we would like to investigate its bound.

First note that

$$\log \left(\frac{g(m, k+1)}{g(m, k)} \right) = m \log \left(1 + \frac{1}{k} \right) + \log \left(1 - \frac{m}{k+1} \right)$$

which is nondecreasing over $k \in [m, \infty)$. Thus,

$$\log \left(\frac{g(m, k+1)}{g(m, k)} \right) \leq \lim_{k' \rightarrow \infty} \log \left(\frac{g(m, k'+1)}{g(m, k')} \right) = 0,$$

i.e., $g(m, k) \leq g(m, m) = \left[\frac{m^m}{m!} \right]^{\frac{1}{m}} \leq e$.

On the other hand, since $\frac{(k-m)!}{k!} \leq \frac{1}{(k-m+1)^m}$, thus

$$g(m, k) \leq \left[\frac{k^m}{(k-m+1)^m} \right]^{\frac{1}{m}} = \frac{k}{k-m+1}.$$

Hence, $g(m, k) \leq \min \left\{ e, \frac{k}{k-m+1} \right\}$.

□

From Proposition 4, we note that when k is large enough, the output of Algorithm 5 is almost optimal.

Finally, we present our approximation results below.

Theorem 5 For any positive integers $m \leq k \leq n$,

- (i) both Algorithm 5 and Algorithm 6 yield $\frac{1}{e}$ -approximation for the D -optimal design problem with repetitions; and
- (ii) given any $\epsilon \in (0, 1)$, if $k \geq \frac{m-1}{\epsilon}$, then both Algorithm 5 and Algorithm 6 yield $(1 - \epsilon)$ -approximation for the D -optimal design problem with repetitions.

Proof: The first result directly follows from Proposition 4. For the second one, given $\epsilon \in (0, 1)$, by Proposition 4, let

$$\frac{k-m+1}{k} \geq 1 - \epsilon.$$

Then the conclusion follows by letting $k \geq \frac{m-1}{\epsilon}$. □

To conclude this part, we remark that although the results from previous sections hold for D -optimal design with repetitions as well, Algorithm 5 has tighter approximation ratios. Therefore, investigating the convex relaxation solution and approximation algorithms of D -optimal design with repetitions alone does help us improve the approximation bounds.

5.1. Deterministic Implementation Similar to Section 3.4, the approximation ratios presented in the previous subsection hold in the sense of expectation. Recall that (\hat{x}, \hat{w}) is an optimal solution of (3) with $\hat{x} \in \mathbb{Q}_+^n$, q is a common multiple of the denominators $\hat{x}_1, \dots, \hat{x}_n$ and multi-set \mathcal{A}_q of size qk contains $q\hat{x}_i$ copies of index i for each $i \in [n]$. In this subsection, we will show that the deterministic Algorithm 7 applies to Algorithm 5, which achieves the same approximation ratios.

In this deterministic Algorithm 7, let S be a subset such that $|S| = s \leq k$. Let $\tilde{\mathcal{S}}_q, \tilde{\mathcal{S}}$ be outputs of Algorithm 5, Algorithm 6, respectively. We know that from Lemma 5, $\tilde{\mathcal{S}}_q \xrightarrow{\mu} \tilde{\mathcal{S}}$ when $q \rightarrow \infty$. Thus, the expectation of m th power of objective function given S is

$$\begin{aligned}
H(S) &:= \mathbb{E} \left[\det \left(\sum_{i \in \tilde{\mathcal{S}}} \mathbf{a}_i \mathbf{a}_i^\top \right) \middle| S \subseteq \tilde{\mathcal{S}} \right] = \lim_{q \rightarrow \infty} \mathbb{E} \left[\det \left(\sum_{i \in \tilde{\mathcal{S}}_q} \mathbf{a}_i \mathbf{a}_i^\top \right) \middle| S \subseteq \tilde{\mathcal{S}}_q \right] \\
&= \lim_{q \rightarrow \infty} \sum_{U \in \binom{\mathcal{A}_q \setminus S}{k-s}} \frac{\frac{1}{q^k}}{\sum_{\tilde{U} \in \binom{\mathcal{A}_q \setminus S}{k-s}} \frac{1}{q^k}} \det \left(\sum_{i \in U} \mathbf{a}_i \mathbf{a}_i^\top + \sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top \right) \\
&= \lim_{q \rightarrow \infty} \binom{qk-s}{k-s}^{-1} \sum_{U \in \binom{\mathcal{A}_q \setminus S}{k-s}} \sum_{R \in \binom{U \cup S}{m}} \det \left(\sum_{i \in R} \mathbf{a}_i \mathbf{a}_i^\top \right) \\
&= \lim_{q \rightarrow \infty} \sum_{r=1}^{\min\{k-s, m\}} \frac{q^r \binom{qk-s-r}{k-s-r}}{\binom{qk-s}{k-s}} \sum_{R \in \binom{\mathcal{A}_q}{m}, |R \setminus S|=r} q^{-r} \det \left(\sum_{i \in R} \mathbf{a}_i \mathbf{a}_i^\top \right) \\
&= \sum_{r=1}^{\min\{k-s, m\}} \lim_{q \rightarrow \infty} \frac{q^r \binom{qk-s-r}{k-s-r}}{\binom{qk-s}{k-s}} \lim_{q \rightarrow \infty} \sum_{R \in \binom{\mathcal{A}_q}{m}, |R \setminus S|=r} q^{-r} \det \left(\sum_{i \in R} \mathbf{a}_i \mathbf{a}_i^\top \right), \\
&= \sum_{r=1}^{\min\{k-s, m\}} \frac{(k-s)!}{k^r (k-s-r)!} \lim_{q \rightarrow \infty} \sum_{R \in \binom{\mathcal{A}_q}{m}, |R \setminus S|=r} q^{-r} \det \left(\sum_{i \in R} \mathbf{a}_i \mathbf{a}_i^\top \right) \tag{21}
\end{aligned}$$

where the second equality is due to $\tilde{\mathcal{S}}_q \xrightarrow{\mu} \tilde{\mathcal{S}}$ when $q \rightarrow \infty$, third equality is a direct computing of the conditional probability, the fourth equality is due to Lemma 2, the fifth and last equalities is because both $\lim_{q \rightarrow \infty} \frac{q^r \binom{qk-s-r}{k-s-r}}{\binom{qk-s}{k-s}}$ and $\lim_{q \rightarrow \infty} \sum_{R \in \binom{\mathcal{A}_q}{m}, |R \setminus S|=r} q^{-r} \det(\sum_{i \in R} \mathbf{a}_i \mathbf{a}_i^\top)$ exist and are finite for each $r \in \{1, \dots, \min\{k-s, m\}\}$.

We note that from Lemma 3, for each $r = 1, \dots, \min\{k-s, m\}$, the limit in (21) can be computed efficiently according to the following lemma.

Lemma 6 For each $r = 1, \dots, \min\{k-s, m\}$,

- (i) the term $\sum_{R \in \binom{\mathcal{A}_q}{m}, |R \setminus S|=r} q^{-r} \det(\sum_{i \in R} \mathbf{a}_i \mathbf{a}_i^\top)$ is equal to the coefficient t^r of the following determinant function

$$\det \left(\frac{t}{q} \sum_{i \in \mathcal{A}_q \setminus S} \mathbf{a}_i \mathbf{a}_i^\top + \sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top \right);$$

Algorithm 7 Derandomization of Algorithm 6

-
- 1: Suppose $(\widehat{\mathbf{x}}, \widehat{w})$ is an optimal solution to the convex relaxation (3) with $B = [0, 1]$, where $\widehat{\mathbf{x}} \in [0, 1]^n$ with $\sum_{i \in [n]} \widehat{x}_i = k$ and $\widehat{w} = f(\widehat{\mathbf{x}})$
 - 2: Initialize chosen set $S = \emptyset$
 - 3: **do**
 - 4: Let $j^* \in \arg \max_{j \in [n] \setminus S} H(S \cup j)$, where $H(S \cup j)$ is define (21), and the limit can be computed by Lemma 6
 - 5: Add j^* to set S
 - 6: **while** $|S| < k$
 - 7: Output S
-

(ii) the term $\lim_{q \rightarrow \infty} \sum_{R \in \binom{[n]}{m}, |R \setminus S| = r} q^{-r} \det \left(\sum_{i \in R} \mathbf{a}_i \mathbf{a}_i^\top \right)$ is equal to the coefficient t^r of the following determinant function

$$\det \left(t \sum_{i \in [n]} \widehat{x}_i \mathbf{a}_i \mathbf{a}_i^\top + \sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top \right).$$

Proof:

(i) The results follow directly by Lemma 3.

(ii) By part (i), the term $\lim_{q \rightarrow \infty} \sum_{R \in \binom{[n]}{m}, |R \setminus S| = r} q^{-r} \det \left(\sum_{i \in R} \mathbf{a}_i \mathbf{a}_i^\top \right)$ is equal to the coefficient t^r of the following determinant function

$$\lim_{q \rightarrow \infty} \det \left(\frac{t}{q} \sum_{i \in \mathcal{A}_q \setminus S} \mathbf{a}_i \mathbf{a}_i^\top + \sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top \right).$$

Since $\lim_{q \rightarrow \infty} \frac{1}{q} \sum_{i \in \mathcal{A}_q \setminus S} \mathbf{a}_i \mathbf{a}_i^\top = \sum_{i \in [n]} \widehat{x}_i \mathbf{a}_i \mathbf{a}_i^\top$ and $\det(\cdot)$ is a continuous function, thus we arrive at the conclusion. □

Algorithm 7 proceeds as follows. We start with an empty subset S of chosen elements, and for each $j \notin S$, we compute the expected m th power of objective function that j will be chosen $H(S \cup j)$. We update $S := S \cup \{j^*\}$, where $j^* \in \arg \max_{j \in [n] \setminus S} H(S \cup j)$. Then go to next iteration. This procedure will terminate if $|S| = k$. Similar to Algorithm 2 and Algorithm 4, the time complexity of Algorithm 7 is $O(m^4 n k^2)$. Thus, in practice, we recommend Algorithm 5 for D -optimal design problem with repetitions.

The approximation results for Algorithm 7 are identical to those in Theorem 5, which is

Theorem 6 For any positive integers $m \leq k \leq n$ and $\epsilon \in (0, 1)$,

- (i) deterministic Algorithm 7 is efficiently computable and yields $\frac{1}{\epsilon}$ -approximation for the D -optimal design problem with repetitions; and

- (ii) given $\epsilon \in (0, 1)$, if $k \geq \frac{m-1}{\epsilon}$, then deterministic Algorithm 7 yields $(1 - \epsilon)$ -approximation for the D -optimal design problem with repetitions.

6. Closing Remarks and Conclusion In this section, we make our final remarks about the proposed algorithms and present a conclusion of this paper.

Closing Remarks: We first remark that the proposed methods work also for A-optimality design, which has been studied in [29]. In their paper, the authors also showed that the proposed methods might not work for other criteria. For D -optimal Design Problem without repetitions, if $k \approx m$, then we recommend sampling Algorithm 1 due to its efficiency and accuracy; if $k \gg m$, then we recommend sampling Algorithm 3 because of its efficiency and asymptotic optimality. For D -optimal Design Problem with repetitions, we recommend sampling Algorithm 5 since it is much more efficient than its deterministic counterpart.

Conclusion: In this paper, we show that D -optimal design problem admits $\frac{1}{\epsilon}$ -approximation guarantee. That is, we propose a sampling algorithm and its deterministic implementation, whose solution is at most $\frac{1}{\epsilon}$ of the true optimal objective value, giving the first constant approximation ratio for this problem. We also analyze a different sampling algorithm, which achieves the asymptotic optimality, i.e., the output of the algorithm is $(1 - \epsilon)$ -approximation if $k \geq \frac{4m}{\epsilon} + \frac{12}{\epsilon^2} \log(\frac{1}{\epsilon})$ for any $\epsilon \in (0, 1)$. For D -optimal design problem with repetitions, i.e., each experiment can be picked multiple times, our sampling algorithm and its derandomization improves asymptotic approximation ratio, i.e., the output of the algorithm is $(1 - \epsilon)$ -approximation if $k \geq \frac{m-1}{\epsilon}$ for any $\epsilon \in (0, 1)$. For future research, we would like to investigate if more sophisticated relaxation schemes can be used to improve the approximation analyses. Another direction is to prove the tightness of the approximation bounds. In particular, we conjecture that for D -optimal design problem with or without repetitions, to achieve $(1 - \epsilon)$ -approximation, one must have $k = \Omega(m/\epsilon)$.

Acknowledgment The first author has been supported by National Science Foundation grant CCF-1717947. Valuable comments from the editors and two anonymous reviewers are gratefully acknowledged. The authors are also grateful to Jiaqi Wang for identifying a mistake in the derandomization algorithms.

References

- [1] Isabelle Albert and Jean-Baptiste Denis. Dirichlet and multinomial distributions: properties and uses in jags. *Analysis*, 31:1141–1155, 2011.
- [2] Zeyuan Allen-Zhu, Yuanzhi Li, Aarti Singh, and Yining Wang. Near-optimal design of experiments via regret minimization. In *International Conference on Machine Learning*, pages 126–135, 2017.

- [3] Zeyuan Allen-Zhu, Zhenyu Liao, and Lorenzo Orecchia. Spectral sparsification and regret minimization beyond matrix multiplicative updates. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 237–245. ACM, 2015.
- [4] Nima Anari and Shayan Oveis Gharan. A generalization of permanent inequalities and applications in counting and optimization. *arXiv preprint arXiv:1702.02937*, 2017.
- [5] Anthony Atkinson, Alexander Donev, and Randall Tobias. *Optimum experimental designs, with SAS*, volume 34. Oxford University Press, 2007.
- [6] Haim Avron, Christos Boutsidis, Sivan Toledo, and Anastasios Zouzias. Efficient dimensionality reduction for canonical correlation analysis. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 347–355, 2013.
- [7] Joshua Batson, Daniel A Spielman, and Nikhil Srivastava. Twice-ramanujan sparsifiers. *SIAM Journal on Computing*, 41(6):1704–1721, 2012.
- [8] Ahron Ben-Tal and Arkadi Nemirovski. *Lectures on modern convex optimization: analysis, algorithms, and engineering applications*, volume 2. SIAM, 2001.
- [9] Mustapha Bouhtou, Stephane Gaubert, and Guillaume Sagnol. Submodularity and randomized rounding techniques for optimal experimental design. *Electronic Notes in Discrete Mathematics*, 36:679–686, 2010.
- [10] Petter Brändén and Johan Jonasson. Negative dependence in sampling. *Scandinavian Journal of Statistics*, 39(4):830–838, 2012.
- [11] Ken RW Brewer and Muhammad Hanif. *Sampling with unequal probabilities*, volume 15. Springer Science & Business Media, 2013.
- [12] Joel G Broida and Stanley Gill Williamson. *A comprehensive introduction to linear algebra*. Addison-Wesley Reading, Mass., 1989.
- [13] Jaroslaw Byrka, Thomas Pensyl, Bartosz Rybicki, Aravind Srinivasan, and Khoa Trinh. An improved approximation for k-median, and positive correlation in budgeted optimization. In *Proceedings of the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms*, pages 737–756. SIAM, 2014.
- [14] Gruia Calinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing*, 40(6):1740–1766, 2011.
- [15] Herman Chernoff. A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. *The Annals of Mathematical Statistics*, pages 493–507, 1952.
- [16] James W Cooley and John W Tukey. An algorithm for the machine calculation of complex fourier series. *Mathematics of computation*, 19(90):297–301, 1965.
- [17] Walter Theodore Federer et al. *Experimental design*. Macmillan Co., New York and London, 1955.
- [18] Valerii Vadimovich Fedorov. *Theory of optimal experiments*. Elsevier, 1972.

-
- [19] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 1985.
- [20] Shui-Hung Hou. Classroom note: A simple proof of the leverrier–faddeev characteristic polynomial algorithm. *SIAM review*, 40(3):706–709, 1998.
- [21] Siddharth Joshi and Stephen Boyd. Sensor selection via convex optimization. *IEEE Transactions on Signal Processing*, 57(2):451–462, 2009.
- [22] Leonid G Khachiyan. Rounding of polytopes in the real number model of computation. *Mathematics of Operations Research*, 21(2):307–320, 1996.
- [23] Roger E Kirk. *Experimental design*. Wiley Online Library, 1982.
- [24] Andreas Krause and Daniel Golovin. Submodular function maximization. Available at http://www.cs.cmu.edu/afs/.cs.cmu.edu/Web/People/dgolovin/papers/submodular_survey12.pdf, 2014.
- [25] Mi Lin and Neil Sydney Trudinger. *On Some Inequalities for Elementary Symmetric Function*. Australian National University, School of Mathematical Sciences, Centre for Mathematics and its Applications, 1993.
- [26] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming*, 14(1):265–294, 1978.
- [27] Aleksandar Nikolov. Randomized rounding for the largest simplex problem. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing*, pages 861–870. ACM, 2015.
- [28] Aleksandar Nikolov and Mohit Singh. Maximizing determinants under partition constraints. In *STOC*, pages 192–201, 2016.
- [29] Aleksandar Nikolov, Mohit Singh, and Uthaipon Tao Tantipongpipat. Proportional volume sampling and approximation algorithms for a-optimal design. *arXiv preprint arXiv:1802.08318*, 2018.
- [30] Athanasios Papoulis and S Unnikrishna Pillai. *Probability, random variables, and stochastic processes*. Tata McGraw-Hill Education, 2002.
- [31] Friedrich Pukelsheim. *Optimal design of experiments*. SIAM, 2006.
- [32] Guillaume Sagnol and Radoslav Harman. Computing exact d -optimal designs by mixed integer second-order cone programming. *The Annals of Statistics*, 43(5):2198–2224, 2015.
- [33] Manohar Shamaiah, Siddhartha Banerjee, and Haris Vikalo. Greedy sensor selection: Leveraging submodularity. In *Decision and Control (CDC), 2010 49th IEEE Conference on*, pages 2572–2577. IEEE, 2010.
- [34] Mohit Singh and Weijun Xie. Approximate positively correlated distributions and approximation algorithms for d -optimal design. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2240–2255. Society for Industrial and Applied Mathematics, 2018.
- [35] Joel H Spencer and Joel Harold Spencer. *Ten lectures on the probabilistic method*, volume 52. Society for Industrial and Applied Mathematics Philadelphia, PA, 1987.

- [36] Daniel A Spielman and Nikhil Srivastava. Graph sparsification by effective resistances. *SIAM Journal on Computing*, 40(6):1913–1926, 2011.
- [37] Damian Straszak and Nisheeth K Vishnoi. Real stable polynomials and matroids: optimization and counting. *arXiv preprint arXiv:1611.04548*, 2016.
- [38] 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*, pages 315–323. Society for Industrial and Applied Mathematics, 2015.
- [39] Yining Wang, Adams Wei Yu, and Aarti Singh. On computationally tractable selection of experiments in regression models. *arXiv preprint arXiv:1601.02068*, 2016.
- [40] William J Welch. Algorithmic complexity: three np-hard problems in computational statistics. *Journal of Statistical Computation and Simulation*, 15(1):17–25, 1982.
- [41] Jianhong Xu. Generalized newton-like inequalities. *J. Inequal. Pure and Appl. Math*, 9(3), 2008.

Appendix A. Proofs

A.1 Proof of Lemma 2

Lemma 2 Suppose $\mathbf{a}_i \in \mathbb{R}^m$ for $i \in T$ with $|T| \geq m$, then

$$\det \left(\sum_{i \in T} \mathbf{a}_i \mathbf{a}_i^\top \right) = \sum_{S \in \binom{T}{m}} \det \left(\sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top \right). \quad (5)$$

Proof: Suppose that $T = \{i_1, \dots, i_{|T|}\}$. Let matrix $A = [\mathbf{a}_{i_1}, \dots, \mathbf{a}_{i_{|T|}}]$, then

$$\det \left(\sum_{i \in T} \mathbf{a}_i \mathbf{a}_i^\top \right) = \det (AA^\top). \quad (22)$$

Next the right-hand side of (22) is equivalent to

$$\det (AA^\top) = \sum_{S \in \binom{T}{m}} \det (A_S)^2 = \sum_{S \in \binom{T}{m}} \det (A_S A_S^\top) = \sum_{S \in \binom{T}{m}} \det \left(\sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top \right),$$

where A_S is the submatrix of A with columns from subset S , the first equality is due to Cauchy-Binet Formula [12], the second equality is because A_S is a square matrix, and the last inequality is the definition of $A_S A_S^\top$. \square

A.2 Proof of Lemma 3

Lemma 3 For any $\mathbf{x} \in [0, 1]^n$, then

$$\det \left(\sum_{i \in [n]} x_i \mathbf{a}_i \mathbf{a}_i^\top \right) = \sum_{S \in \binom{[n]}{m}} \prod_{i \in S} x_i \det \left(\sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top \right). \quad (6)$$

Proof: [Proof of Lemma 3] Let $P = \text{diag}(\mathbf{x}) \in \mathbb{R}^{n \times n}$ be the diagonal matrix with diagonal vector equal to \mathbf{x} and matrix $A = [\mathbf{a}_1, \dots, \mathbf{a}_n]$. By Lemma 2, we have

$$\det \left(\sum_{i \in [n]} x_i \mathbf{a}_i \mathbf{a}_i^\top \right) = \det \left(\sum_{i \in [n]} (\sqrt{x_i} \mathbf{a}_i) (\sqrt{x_i} \mathbf{a}_i)^\top \right) = \sum_{S \in \binom{[n]}{m}} \det \left(\sum_{i \in S} x_i \mathbf{a}_i \mathbf{a}_i^\top \right). \quad (23a)$$

Note that $\sum_{i \in S} x_i \mathbf{a}_i \mathbf{a}_i^\top = A_S P_S A_S^\top$, where A_S is the submatrix of A with columns from subset \tilde{S} , and P_S is the square submatrix of P with rows and columns from S . Thus, (23a) further yields

$$\begin{aligned} \det \left(\sum_{i \in [n]} x_i \mathbf{a}_i \mathbf{a}_i^\top \right) &= \sum_{S \in \binom{[n]}{m}} \det \left(\sum_{i \in S} x_i \mathbf{a}_i \mathbf{a}_i^\top \right) = \sum_{S \in \binom{[n]}{m}} \det (A_S P_S A_S^\top) = \sum_{S \in \binom{[n]}{m}} \det (A_S)^2 \det (P_S) \\ &= \sum_{S \in \binom{[n]}{m}} \prod_{i \in S} x_i \det \left(\sum_{i \in S} \mathbf{a}_i \mathbf{a}_i^\top \right) \end{aligned} \quad (23b)$$

where the third and fourth equalities are because the determinant of products of square matrices is equal to the products of individual determinants. \square

A.3 Proof of Proposition 1 Before proving Proposition 1, we first introduce two well-known results for sum of homogeneous and symmetric polynomials.

Lemma 7 (Maclaurin's inequality [25]) Given a set S , an integer $s \in \{0, 1, \dots, |S|\}$ and nonnegative vector $x \in \mathbb{R}_+^{|S|}$, we must have

$$\frac{1}{|S|} \left(\sum_{i \in S} x_i \right) \geq \sqrt[s]{\frac{1}{\binom{|S|}{s}} \left(\sum_{Q \in \binom{S}{s}} \prod_{i \in Q} x_i \right)}.$$

And

Lemma 8 (Generalized Newton's inequality [41]) Given a set S , two nonnegative positive integers $s, \tau \in \mathbb{Z}_+$ such that $s, \tau \leq |S|$ and nonnegative vector $x \in \mathbb{R}_+^{|S|}$, then we have

$$\frac{\left(\sum_{R \in \binom{S}{s}} \prod_{j \in R} x_j \right)}{\binom{|S|}{s}} \frac{\left(\sum_{R \in \binom{S}{\tau}} \prod_{i \in R} x_i \right)}{\binom{|S|}{\tau}} \geq \frac{\sum_{Q \in \binom{S}{s+\tau}} \prod_{i \in Q} x_i}{\binom{|S|}{s+\tau}}$$

Now we are ready to prove the main proposition.

Proposition 1 Let \tilde{S} be the random set defined in (4). Then for any $T \subseteq [n]$ such that $|T| = m$, we have

$$\mathbb{P}[T \subseteq \tilde{S}] \geq \frac{1}{g(m, n, k)} \prod_{i \in T} \hat{x}_i := \alpha^m \prod_{i \in T} \hat{x}_i,$$

where

$$g(m, n, k) = \max_y \left\{ \sum_{\tau=0}^m \frac{\binom{n-m}{k-\tau}}{(n-m)^{m-\tau} \binom{n-m}{k-m}} \frac{\binom{m}{\tau}}{m^\tau} (k-y)^{m-\tau} (y)^\tau : \frac{mk}{n} \leq y \leq m \right\}. \quad (9)$$

Proof: According to Lemma 1 and sampling procedure in (4), we have

$$\mathbb{P}[T \subseteq \tilde{S}] = \prod_{j \in T} \hat{x}_j \frac{\sum_{R \in \binom{[n] \setminus T}{k-m}} \prod_{j \in R} \hat{x}_j}{\sum_{\tilde{S} \in \binom{[n]}{k}} \prod_{i \in \tilde{S}} \hat{x}_i} = \prod_{j \in T} \hat{x}_j \frac{\sum_{R \in \binom{[n] \setminus T}{k-m}} \prod_{j \in R} \hat{x}_j}{\sum_{\tau=0}^m \sum_{W \in \binom{T}{\tau}} \prod_{i \in W} \hat{x}_i \left(\sum_{Q \in \binom{[n] \setminus T}{k-\tau}} \prod_{i \in Q} \hat{x}_i \right)}$$

where the second equality uses the following identity

$$\binom{[n]}{k} = \bigcup_{\tau=0}^m \left\{ W \cup Q : W \in \binom{T}{\tau}, Q \in \binom{[n] \setminus T}{k-\tau} \right\}.$$

We now let

$$A_T(x) = \frac{\sum_{\tau=0}^m \sum_{W \in \binom{T}{\tau}} \prod_{i \in W} \hat{x}_i \left(\sum_{Q \in \binom{[n] \setminus T}{k-\tau}} \prod_{i \in Q} \hat{x}_i \right)}{\sum_{R \in \binom{[n] \setminus T}{k-m}} \prod_{j \in R} \hat{x}_j}. \quad (24a)$$

According to Definition 1, it is sufficient to find a lower bound to $\frac{1}{\prod_{i \in T} \hat{x}_i} \mathbb{P} [T \subseteq \tilde{\mathcal{S}}]$, i.e.,

$$\frac{1}{g(m, n, k)} \leq \min_x \left\{ \frac{1}{\prod_{i \in T} \hat{x}_i} \mathbb{P} [T \subseteq \tilde{\mathcal{S}}] = \frac{1}{A_T(x)} : \sum_{i \in [n]} \hat{x}_i = k, x \in [0, 1]^n \right\}.$$

Or equivalently, we would like to find an upper bound of $A_T(x)$ for any x which satisfies $\sum_{i \in [n]} \hat{x}_i = k, x \in [0, 1]^n$, i.e., show that

$$g(m, n, k) \geq \max_x \left\{ A_T(x) : \sum_{i \in [n]} \hat{x}_i = k, x \in [0, 1]^n \right\}. \quad (24b)$$

In the following steps, we first observe that in (24a), the components of $\{x_i\}_{i \in T}$ and $\{x_i\}_{i \in [n] \setminus T}$ are both symmetric in the expression of $A_T(x)$. We will show that for the worst case, $\{x_i\}_{i \in T}$ are all equal and $\{x_i\}_{i \in [n] \setminus T}$ are also equal. We also show that $\hat{x}_j \leq \hat{x}_i$ for each $i \in T$ and $j \in [n] \setminus T$. This allows us to reduce the optimization problem in R.H.S. of (24b) to a single variable optimization problem, i.e., (9). The proof is now separated into following three claims.

(i) First, we claim that

Claim 1 *The optimal solution to (24b) must satisfy the following condition - for each $i \in T$ and $j \in [n] \setminus T$, $\hat{x}_j \leq \hat{x}_i$.*

Proof: We prove it by contradiction. Suppose that there exists $i' \in T$ and $j' \in [n] \setminus T$, where $\hat{x}_{i'} < \hat{x}_{j'}$. By collecting the coefficients of $1, \hat{x}_{i'}, \hat{x}_{j'}, \hat{x}_{i'}\hat{x}_{j'}$, we have

$$A_T(x) = \frac{b_1 + b_2\hat{x}_{i'} + b_2\hat{x}_{j'} + b_3\hat{x}_{i'}\hat{x}_{j'}}{c_1 + c_2\hat{x}_{j'}}$$

where b_1, b_2, b_3, c_1, c_2 are all non-negative numbers with

$$\begin{aligned} b_1 &= \sum_{\bar{S} \in \binom{[n] \setminus \{i', j'\}}{k}} \prod_{i \in \bar{S}} \hat{x}_i, b_2 = \sum_{\bar{S} \in \binom{[n] \setminus \{i', j'\}}{k-1}} \prod_{i \in \bar{S}} \hat{x}_i, b_3 = \sum_{\bar{S} \in \binom{[n] \setminus \{i', j'\}}{k-2}} \prod_{i \in \bar{S}} \hat{x}_i \\ c_1 &= \sum_{R \in \binom{[n] \setminus (T \cup \{j'\})}{k-m}} \prod_{j \in R} \hat{x}_j, c_2 = \sum_{R \in \binom{[n] \setminus (T \cup \{j'\})}{k-m-1}} \prod_{j \in R} \hat{x}_j. \end{aligned}$$

Note that

$$\hat{x}_{i'}\hat{x}_{j'} \leq \frac{1}{4} (\hat{x}_{i'} + \hat{x}_{j'})^2.$$

Therefore, $A_T(x)$ has a larger value if we replace $\hat{x}_{i'}, \hat{x}_{j'}$ by their average, i.e. $\hat{x}_{i'} := \frac{1}{2}(\hat{x}_{i'} + \hat{x}_{j'})$, $\hat{x}_{j'} := \frac{1}{2}(\hat{x}_{i'} + \hat{x}_{j'})$. \diamond

(ii) Next, we claim that

Claim 2 for any feasible x to (24b), and for each $S \subseteq [n]$ and $s \in \{0, 1, \dots, |S|\}$, we must have

$$\sum_{Q \in \binom{[n]}{s}} \prod_{i \in Q} \hat{x}_i \leq \frac{\binom{|S|}{s}}{|S|^s} \left(\sum_{i \in S} \hat{x}_i \right)^s.$$

Proof: This directly follows from Lemma 7. \diamond

And also

Claim 3 for each $T \subseteq [n]$ with $|T| = k$ and $\tau \in \{0, 1, \dots, m\}$,

$$\sum_{Q \in \binom{[n] \setminus T}{k-\tau}} \prod_{i \in Q} \hat{x}_i \leq \frac{\binom{n-m}{k-\tau}}{(n-m)^{m-\tau} \binom{n-m}{k-m}} \left(\sum_{R \in \binom{[n] \setminus T}{k-m}} \prod_{j \in R} \hat{x}_j \right) \left(\sum_{i \in [n] \setminus T} \hat{x}_i \right)^{m-\tau}.$$

Proof: This can be shown by Claim 2 and Lemma 8

$$\begin{aligned} & \frac{\binom{n-m}{k-\tau}}{(n-m)^{m-\tau} \binom{n-m}{k-m}} \left(\sum_{R \in \binom{[n] \setminus T}{k-m}} \prod_{j \in R} \hat{x}_j \right) \left(\sum_{i \in [n] \setminus T} \hat{x}_i \right)^{m-\tau} \\ & \geq \frac{\binom{n-m}{k-\tau}}{\binom{n-m}{m-\tau} \binom{n-m}{k-m}} \left(\sum_{R \in \binom{[n] \setminus T}{k-m}} \prod_{j \in R} \hat{x}_j \right) \left(\sum_{S \in \binom{[n] \setminus T}{m-\tau}} \prod_{i \in S} \hat{x}_i \right) \geq \sum_{Q \in \binom{[n] \setminus T}{k-\tau}} \prod_{i \in Q} \hat{x}_i \end{aligned}$$

where the first inequality is due to Claim 2, and the last inequality is because of Lemma 8. \diamond

(iii) Thus, by Claim 3, for any feasible x to (24b), $A_T(x)$ in (24a) can be upper bounded as

$$\begin{aligned} A_T(x) & \leq \sum_{\tau=0}^m \frac{\binom{n-m}{k-\tau}}{(n-m)^{m-\tau} \binom{n-m}{k-m}} \left(\sum_{i \in [n] \setminus T} \hat{x}_i \right)^{m-\tau} \sum_{W \in \binom{T}{\tau}} \prod_{i \in W} \hat{x}_i \\ & \leq \sum_{\tau=0}^m \frac{\binom{n-m}{k-\tau}}{(n-m)^{m-\tau} \binom{n-m}{k-m}} \frac{\binom{m}{\tau}}{m^\tau} \left(\sum_{i \in [n] \setminus T} \hat{x}_i \right)^{m-\tau} \left(\sum_{i \in T} \hat{x}_i \right)^\tau \\ & \leq \max_y \left\{ \sum_{\tau=0}^m \frac{\binom{n-m}{k-\tau}}{(n-m)^{m-\tau} \binom{n-m}{k-m}} \frac{\binom{m}{\tau}}{m^\tau} (k-y)^{m-\tau} (y)^\tau : \frac{mk}{n} \leq y \leq m \right\} := g(m, n, k) \quad (24c) \end{aligned}$$

where the second inequality is due to Claim 2, and the last inequality is because we let $y = \sum_{i \in T} \hat{x}_i$ which is no larger than m , maximize over it and Claim 1 yields that $y/m \geq (k-y)/(n-m)$, i.e. $\frac{mk}{n} \leq y \leq m$. This completes the proof. \square

A.4 Proof of Proposition 2

Proposition 2 For any $n \geq k \geq m$, we have

$$\alpha^{-1} = [g(m, n, k)]^{\frac{1}{m}} \leq \lim_{\tau \rightarrow \infty} [g(m, \tau, k)]^{\frac{1}{m}} \leq \min \left\{ e, 1 + \frac{k}{k-m+1} \right\}. \quad (10)$$

Proof:

(i) First of all, we prove the following claim.

Claim 4 For any $m \leq k \leq n$, we have

$$g(m, n, k) \leq g(m, n+1, k).$$

Proof: Let y^* be the maximizer to (9) for any given $m \leq k \leq n$, i.e.,

$$g(m, n, k) = \sum_{\tau=0}^m \frac{\binom{n-m}{k-\tau}}{(n-m)^{m-\tau} \binom{n-m}{k-m}} \frac{\binom{m}{\tau}}{m^\tau} (k-y^*)^{m-\tau} (y^*)^\tau.$$

Clearly, y^* is feasible to (9) with pair $(m, n+1, k)$. We only need to show that

$$g(m, n, k) \leq \sum_{\tau=0}^m \frac{\binom{n+1-m}{k-\tau}}{(n+1-m)^{m-\tau} \binom{n+1-m}{k-m}} \frac{\binom{m}{\tau}}{m^\tau} (k-y^*)^{m-\tau} (y^*)^\tau.$$

In other words, it is sufficient to show for any $0 \leq \tau \leq m$,

$$\frac{\binom{n-m}{k-\tau}}{(n-m)^{m-\tau} \binom{n-m}{k-m}} \leq \frac{\binom{n+1-m}{k-\tau}}{(n+1-m)^{m-\tau} \binom{n+1-m}{k-m}},$$

which is equivalent to prove

$$\frac{n-k}{n-m} \cdot \frac{n-k-1}{n-m} \cdots \frac{n-k-m+\tau+1}{n-m} \leq \frac{n+1-k}{n+1-m} \cdot \frac{n+1-k-1}{n+1-m} \cdots \frac{n+1-k-m+\tau+1}{n+1-m}.$$

The above inequality holds since for any positive integers p, q with $p < q$, we must have

$$\frac{p}{q} \leq \frac{p+1}{q+1}. \quad \diamond$$

(ii) By Claim 4, it is sufficient to investigate the bound $\lim_{n' \rightarrow \infty} g(m, n', k)$, which provides an upper bound to $g(m, n, k)$ for any integers $n \geq k \geq m$. Therefore, from now on, we only consider the case when $n \rightarrow \infty$ for any fixed $k \geq m$.

Note that for any given y , $\sum_{\tau=0}^m \frac{\binom{n-m}{k-\tau}}{(n-m)^{m-\tau} \binom{n-m}{k-m}} \frac{\binom{m}{\tau}}{m^\tau} (k-y)^{m-\tau} y^\tau$ is the coefficient of t^k in the following polynomial:

$$R_1(t) := \frac{(n-m)^{k-m}}{(k-y)^{k-m} \binom{n-m}{k-m}} \left(1 + \frac{k-y}{n-m} t\right)^{n-m} \left(1 + \frac{y}{m} t\right)^m$$

which is upper bounded by

$$\begin{aligned} R_2(t) &:= \frac{(n-m)^{k-m}}{(k-y)^{k-m} \binom{n-m}{k-m}} \left(1 + \frac{k-y}{n-m} t + \frac{1}{2!} \left(\frac{k-y}{n-m} t\right)^2 + \dots\right)^{n-m} \left(1 + \frac{y}{m} t + \frac{1}{2!} \left(\frac{y}{m} t\right)^2 + \dots\right)^m \\ &= \frac{(n-m)^{k-m}}{(k-y)^{k-m} \binom{n-m}{k-m}} \left(e^{\frac{k-y}{n-m} t}\right)^{n-m} \left(e^{\frac{y}{m} t}\right)^m \end{aligned}$$

because of the inequality $e^r = 1 + r + \frac{1}{2}r^2 + \dots$ for any r and $t \geq 0$. Therefore, we also have

$$\begin{aligned}
& \lim_{n \rightarrow \infty} \frac{1}{k!} \left. \frac{d^k R_1(t)}{dt^k} \right|_{t=0} = \lim_{n \rightarrow \infty} \sum_{\tau=0}^m \frac{\binom{n-m}{k-\tau}}{(n-m)^{m-\tau} \binom{n-m}{k-m}} \frac{\binom{m}{\tau}}{m^\tau} (k-y)^{m-\tau} y^\tau \\
& \leq \lim_{n \rightarrow \infty} \sum_{\tau=0}^m \frac{\binom{n-m}{k-\tau}}{(n-m)^{m-\tau} \binom{n-m}{k-m}} \frac{\binom{m}{\tau}}{m^\tau} (k-y)^{m-\tau} y^\tau \\
& + \sum_{\tau=0}^m \sum_{\substack{i_j \in \mathbb{Z}_+, \forall j \in [n] \\ \sum_{j \in [n-m]} i_j = k-\tau \\ \sum_{j \in [n] \setminus [n-m]} i_j = \tau \\ \max_{j \in [n]} i_j \geq 2}} \frac{1}{\prod_{j \in [n]} i_j!} \frac{1}{(n-m)^{m-\tau} \binom{n-m}{k-m}} \frac{1}{m^\tau} (k-y)^{m-\tau} y^\tau := \lim_{n \rightarrow \infty} \frac{1}{k!} \left. \frac{d^k R_2(t)}{dt^k} \right|_{t=0} \\
& = \lim_{n \rightarrow \infty} \frac{k^k}{k!} \frac{(n-m)^{k-m}}{(k-y)^{k-m} \binom{n-m}{k-m}} \leq \lim_{n \rightarrow \infty} \frac{k^k}{k!} \frac{(n-m)^{k-m}}{(k-m)^{k-m} \binom{n-m}{k-m}} \\
& = \frac{k^k}{k!} \frac{(k-m)!}{(k-m)^{k-m}} := R_3(m, k)
\end{aligned}$$

where the first inequality is due to the non-negativity of the second term of $\frac{1}{k!} \left. \frac{d^k R_2(t)}{dt^k} \right|_{t=0}$, the second and third equalities are because of two equivalent definitions of $R_2(t)$, the last inequality is due to $y \leq m$ and the fourth equality holds because of $n \rightarrow \infty$.

Note that $R_3(m, k)$ is nondecreasing over $k \in [m, \infty)$. Indeed, for any given m ,

$$\log \frac{R_3(m, k+1)}{R_3(m, k)} = k \log \left(1 + \frac{1}{k} \right) - (k-m) \log \left(1 + \frac{1}{(k-m)} \right),$$

whose first derivative over k is equal to

$$\log \left(1 + \frac{1}{k} \right) - \frac{1}{k+1} - \log \left(1 + \frac{1}{(k-m)} \right) + \frac{1}{k-m+1} \leq 0,$$

i.e., $\log \frac{R_3(m, k+1)}{R_3(m, k)}$ is nonincreasing over k . Therefore,

$$\log \frac{R_3(m, k+1)}{R_3(m, k)} = k \log \left(1 + \frac{1}{k} \right) - (k-m) \log \left(1 + \frac{1}{(k-m)} \right) \geq \lim_{k \rightarrow \infty} \log \frac{R_3(m, k+1)}{R_3(m, k)} = 0$$

Thus, $R_3(m, k)$ is upper bounded when $k \rightarrow \infty$, i.e.,

$$R_3(m, k) \leq \lim_{k' \rightarrow \infty} R_3(m, k') = \lim_{k' \rightarrow \infty} \left[\left(1 - \frac{m}{k'} \right)^{-\frac{k'}{m}} \right]^m \frac{(k'-m)^m}{k'(k'-1) \cdots (k'-m+1)} = e^m,$$

where the last equality is due to the fact that $\lim_{k' \rightarrow \infty} \left(1 - \frac{m}{k'} \right)^{-\frac{k'}{m}} = e$ and $\lim_{k' \rightarrow \infty} \frac{(k'-m)^m}{k'(k'-1) \cdots (k'-m+1)} = 1$. Therefore,

$$\begin{aligned}
\lim_{n \rightarrow \infty} [g(m, n, k)]^{\frac{1}{m}} &= \lim_{n \rightarrow \infty} \left[\max_y \left\{ \sum_{\tau=0}^m \frac{\binom{n-m}{k-\tau}}{(n-m)^{m-\tau} \binom{n-m}{k-m}} \frac{\binom{m}{\tau}}{m^\tau} (k-y)^{m-\tau} y^\tau : \frac{mk}{n} \leq y \leq m \right\} \right]^{\frac{1}{m}} \\
&\leq [R_3(m, k)]^{\frac{1}{m}} \leq e
\end{aligned}$$

(iii) We now compute another bound $1 + \frac{k}{k-m+1}$ for $[g(m, n, k)]^{\frac{1}{m}}$, which can be smaller than e when k is large. By Claim 4, we have

$$g(m, n, k) \leq \lim_{n' \rightarrow \infty} g(m, n', k) = \max_y \left\{ \sum_{\tau=0}^m \frac{(k-m)!}{(k-\tau)!} \frac{\binom{m}{\tau}}{m^\tau} (k-y)^{m-\tau} y^\tau : 0 \leq y \leq m \right\}.$$

Note that $0 \leq y \leq m$, thus $k-y \leq k$. Therefore, we have

$$\lim_{n \rightarrow \infty} g(m, n, k) \leq \sum_{\tau=0}^m \frac{(k-m)!}{(k-\tau)!} \binom{m}{\tau} k^{m-\tau} \leq \sum_{\tau=0}^m \left(\frac{k}{k-m+1} \right)^{m-\tau} = \left(1 + \frac{k}{k-m+1} \right)^m,$$

where the last inequality is due to $\frac{(k-m)!}{(k-\tau)!} = \frac{1}{(k-\tau) \cdots (k-m+1)} \leq \left(\frac{1}{k-m+1} \right)^{m-\tau}$.

Therefore, we have

$$\begin{aligned} [g(m, n, k)]^{\frac{1}{m}} &= \left[\max_y \left\{ \sum_{\tau=0}^m \frac{\binom{n-m}{k-\tau}}{(n-m)^{m-\tau}} \frac{\binom{m}{\tau}}{\binom{n-m}{k-m}} \frac{1}{m^\tau} (k-y)^{m-\tau} y^\tau : \frac{mk}{n} \leq y \leq m \right\} \right]^{\frac{1}{m}} \\ &\leq 1 + \frac{k}{k-m+1} \end{aligned}$$

for any $m \leq k \leq n$.

□