Best Principal Submatrix Selection for the Maximum Entropy Sampling Problem: Scalable Algorithms and Performance Guarantees

Yongchun Li
Department of Industrial & Systems Engineering, Virginia Tech, Blacksburg, VA 24061, liyc@vt.edu

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

This paper studies a classic maximum entropy sampling problem (MESP), which aims to select the most informative principal submatrix of a prespecified size from a covariance matrix. MESP has been widely applied to many areas, including healthcare, power system, manufacturing and data science. By investigating its Lagrangian dual and primal characterization, we derive a novel convex integer program for MESP and show that its continuous relaxation yields a near-optimal solution. The results motivate us to study an efficient sampling algorithm and develop its approximation bound for MESP, which improves the best-known bound in literature. We then provide an efficient deterministic implementation of the sampling algorithm with the same approximation bound. By developing new mathematical tools for the singular matrices and analyzing the Lagrangian dual of the proposed convex integer program, we investigate the widely-used local search algorithm and prove its first-known approximation bound for MESP. The proof techniques further inspire us with an efficient implementation of the local search algorithm. Our numerical experiments demonstrate that these approximation algorithms can efficiently solve medium-sized and large-scale instances to near-optimality. Our proposed algorithms are coded and released as open-source software. Finally, we extend the analyses to the A-Optimal MESP (A-MESP), where the objective is to minimize the trace of the inverse of the selected principal submatrix.

Key words: Maximum Entropy Sampling Problem; Convex Integer Program; Sampling Algorithm; Local Search Algorithm; A-Optimality.

1. Introduction

The maximum entropy sampling problem (MESP) is a classic problem in statistics and information theory (Gilmore 1996, Jaynes 1957, Shewry and Wynn 1987), which aims to select a small number of random observations from a possibly large set of candidates to maximize the information obtained. The MESP has been widely applied to healthcare (Alarifi et al. 2019), power system (Li et al. 2012), manufacturing (Wang et al. 2019), data science (Charikar et al. 2000, Song and Liò 2010, Zilly et al. 2017), etc. Specifically, suppose that the n random variables follow a Gaussian distribution and their covariance matrix $\boldsymbol{C} \in \mathbb{R}^{n\times n}$ has a rank $d \leq n$. Then the goal of MESP is to seek a size-s ($s \leq d$) principal submatrix of $\boldsymbol{C}$ with the largest determinant, i.e., MESP can be formulated as

$$\text{(MESP)} \quad z^* := \max_S \left\{\log\det(\boldsymbol{C}_{S,S}) : S \subseteq [n], |S| = s\right\}, \tag{1}$$

where $\boldsymbol{C}_{S,S}$ denotes an $s \times s$ principal submatrix of $\boldsymbol{C}$ with rows and columns from set S and $[n] = \{1, \cdots, n\}$. Note that (a) MESP (1) can be generalized to the case that the observations follow multivariate elliptical distributions (see, e.g., Arellano-Valle et al. 2013); and (b) if we only know the mean and the covariance of the random observations, then MESP (1) is equivalent to the distributionally optimistic counterpart of maximum entropy sampling problem; namely, the joint Gaussian distribution achieves the largest entropy among all the probability distributions with the same mean and covariance (Cover and Thomas 2012). Thus, MESP (1) is indeed a very general model and covers many interesting cases.

1.1. Relevant Literature

We separate the relevant literature into three main parts: applications, relaxation bounds of MESP, and exact and approximation algorithms.

Applications: MESP dates back to Shewry and Wynn (1987) and has been applied to many different areas. One typical application of MESP is the sensor placement (Christodoulou 2015, Bueso et al. 1998). Due to a limited budget, it is desirable to place a small number of sensors to effectively monitor spatial and temporal phenomena, including temperature, humidity, air pollution, etc. Recently, it has been applied to water quality monitoring (O'Flynn et al. 2010). MESP has also played an important role in machine learning and data science, such as feature selection (Charikar et al. 2000, Song and Liò 2010), compressive sensing (Hoch et al. 2014, Schmieder et al. 1993), and image sampling (Rigau et al. 2003, Zilly et al. 2017).

Relaxation Bounds of MESP: It has been proven in Ko et al. (1995) that solving MESP in general is NP-hard. Hence, many efforts have been made to explore its strong relaxation bounds (see, e.g., Anstreicher et al. 1996, 1999, Anstreicher and Lee 2004, Anstreicher 2018a,b, Ko et al.

1995, Hoffman et al. 2001, Lee 1998, Lee and Williams 2003, Anstreicher 2018b). For example, Ko et al. (1995) used the eigenvalue interlacing property of symmetric matrices to derive an upper bound for MESP. Recent progress (Anstreicher 2018a) proposed a new upper bound, referred to as linx bound, and numerically showed that it dominated other bounds studied in the literature. In this paper, we derive a Lagrangian dual bound for MESP and also numerically demonstrate that this new upper bound can be stronger than linx bound among the majority of the testing instances.

Exact and Approximation Algorithms: Besides providing stronger upper bounds, researchers have also attempted to propose exact or approximation algorithms to solve MESP (1). Ko et al. (1995) was one of the first works to develop a branch and bound (B&B) algorithm for solving MESP to optimality. Similar works can be found in Anstreicher et al. (1999), Anstreicher (2018a,b), Burer and Lee (2007) by integrating stronger upper bounds with the B&B algorithm. In this paper, we provide an equivalent convex integer program for MESP, which is suitable for a branch and cut (B&C) algorithm.

However, exact algorithms might not be able to solve very large-scale instances. It has been shown in Anstreicher (2018a) that solving MESP (1) on the instance of $n=90$ to optimality can take as long as several days. As alternative ones, approximation algorithms have also attracted much attention. Many approximation algorithms such as greedy and exchange (i.e., local search) heuristics have been used to generate high-quality solutions for MESP in literature (Ko et al. 1995, Sharma et al. 2015). However, theoretical performance guarantees of these approximation algorithms are rarely known. Although the objective function of MESP (1) is submodular (Kelmans and Kimelfeld 1983), it is neither monotonic nor always nonnegative. Thus, existing results on maximizing the nondecreasing and nonnegative submodular function over a cardinality constraint might not apply and thus require additional assumptions (Charikar et al. 2000, Sharma et al. 2015). Recently, Nikolov (2015) studied a sampling algorithm for the maximum s-subdeterminant problem, which can be reduced to MESP (1), and developed its approximation guarantee. The inapproximability of MESP (1) can be found in Civril and Magdon-Ismail (2013), Summa et al. (2014), which shows that unless P=NP, it is impossible to approximate MESP within an additive error $s\log(c)$ for some constant $c>1$. This paper proposes a different sampling algorithm from the one in Nikolov (2015) and improves its approximation bound. We also analyze the well-known local search algorithm and derive its approximation guarantee. Table 1 summarizes the best-known approximation bounds in literature and our proposed bounds for MESP (1).

Table 1 Summary of Approximation Algorithms for MESP

	Approximation Algorithm	Approximation Bound[1]
Literature	Greedy (Çivril and Magdon-Ismail 2009)	$2^{-1}\log(2)s(s-1)+2^{-1}s\log(n)$
	Samping (Nikolov 2015)	$s\log(s)-\log(s!)$
This paper	Sampling Algorithm 2	$s\log(s)+\log(\binom{n}{s})-s\log(n)$
	Local Search Algorithm 4	$s\min\{\log(s),\log(n-s+2-n/s)\}$

[1] Approximation Bound is defined as the difference between the optimal value and the output value from the algorithm

1.2. Summary of Contributions

The objective of this paper is to develop a new convex integer programming formulation for MESP (1) and based on this formulation, analyze efficient approximation algorithms, and develop their implementations. Below is a summary of our main contributions:

(i) Through the Lagrangian dual of MESP (1) and its primal characterization, we derive a convex integer program for MESP (1) and show that its continuous relaxation solution is near-optimal. In addition, we apply the efficient Frank-Wolfe algorithm to solving the continuous relaxation and derive its rate of convergence.

(ii) The continuous relaxation of the proposed convex integer program motivates us an efficient sampling algorithm and develop its approximation bound for MESP (1), which improves the best-known bound in literature. We then provide an efficient deterministic implementation of the proposed sampling algorithm with the same approximation bound.

(iii) Using the weak duality between the proposed convex integer program and its Lagrangian dual, we investigate the widely-used local search algorithm and prove its first-known approximation bound for MESP (1). The proof techniques further inspire us with an efficient implementation of the local search algorithm.

(iv) Our numerical experiments demonstrate that these approximation algorithms can efficiently solve medium-sized and large-scale instances to near-optimality.

(v) Finally, we extend the analyses to the A-Optimal MESP (A-MESP), where its objective is to minimize the trace of the inverse of the selected principal submatrix. We propose a new convex integer program for A-MESP and study volume sampling and local search algorithms, and prove their approximation ratios.

Organization: The remainder of the paper is organized as follows. Section 2 derives an equivalent convex integer program for MESP. Section 3 develops the sampling algorithm and its deterministic implementation and also explores their approximation guarantees for MESP. Section 4 investigates the local search algorithm and proves its approximation guarantee for MESP. Section 5 conducts a numerical study to demonstrate the efficiency and the solution quality of our proposed approximation algorithms. Section 6 extends the analyses to the A-Optimal MESP (A-MESP). Finally, Section 7 concludes the paper.

Notation: The following notation is used throughout the paper. We use bold lower-case letters (e.g., $\boldsymbol{x}$) and bold upper-case letters (e.g., $\boldsymbol{X}$) to denote vectors and matrices, respectively, and use corresponding non-bold letters (e.g., x_i) to denote their components. We use $\mathbf{0}$ to denote the zero vector and $\mathbf{1}$ to denote the all-ones vector. We let $\mathbb{R}_+^n$ denote the set of all the n dimensional nonnegative vectors and let $\mathbb{R}_{++}^n$ denote the set of all the n dimensional positive vectors. Given an integer n, we let $[n]=\{1,2,\cdots,n\}$, let $[s,n]:=\{s,s+1,\cdots,n\}$. We let $\boldsymbol{I}_n$ denote the $n\times n$ identity matrix and let $\boldsymbol{e}_i$ denote its i-th column. Given a set S and an integer k, we let $|S|$ denote its cardinality and $\binom{S}{k}$ denote the collection of all the size-k subsets out of S. Given an $m\times n$ matrix $\boldsymbol{A}$ and two sets $S\in[m]$, $T\in[n]$, we let $\boldsymbol{A}_{S,T}$ denote a submatrix of $\boldsymbol{A}$ with rows and columns indexed by sets S,T, respectively, let $\boldsymbol{A}_S$ denote a submatrix of $\boldsymbol{A}$ with columns from the set S and let $\mathrm{col}(\boldsymbol{A})$ denote its column space. Given a vector $\boldsymbol{x}\in\mathbb{R}^n$, we let $\mathrm{Diag}(\boldsymbol{x})$ denote the diagonal matrix with diagonal elements $x_1,\cdots,x_n$, and let $\mathrm{supp}(\boldsymbol{x})$ denote the support of $\boldsymbol{x}$. Given a square symmetric matrix $\boldsymbol{A}$, let $\mathrm{diag}(\boldsymbol{A})$ denote the vector of diagonal entries of $\boldsymbol{A}$, let $\boldsymbol{A}^\dagger$ denote its pseudo inverse, let $\det(\boldsymbol{A})$ denote its determinant, let $\mathrm{tr}(\boldsymbol{A})$ denote its trace, and let $\lambda_{\min}(\boldsymbol{A}),\lambda_{\max}(\boldsymbol{A})$ denote the smallest and largest eigenvalues of $\boldsymbol{A}$, respectively. Given a convex set $\mathbb{D}$, we use $\mathrm{relint}(\mathbb{D})$ to denote its relative interior. Additional notation will be introduced later as needed.

2. Convex Integer Programming Formulation

In this section, we will derive the Lagrangian dual (LD) of MESP (1) and its primal characterization (PC), where the latter inspires us a new convex integer programming formulation of MESP (1) by enforcing its variables to be binary.

2.1. Mixed Integer Nonlinear Program of MESP

To begin with, we first observe that MESP (1) has an equivalent mixed integer nonlinear programming formulation using the Cholesky factorization. To do so, for $\boldsymbol{C}\succeq 0$, let $\boldsymbol{C}=\boldsymbol{V}^\top\boldsymbol{V}$ denote its Cholesky factorization, where $\boldsymbol{V}\in\mathbb{R}^{d\times n}$ and let $\boldsymbol{v}_i\in\mathbb{R}^d$ denote the i-th column vector of matrix $\boldsymbol{V}$ for each $i\in[n]$. Also, let us define the following two new functions, which correspond to the objective function of an alternative reformulation of MESP (1), and the objective function of its Lagrangian dual.

Definition 1 *For a $d \times d$ matrix $\boldsymbol{X} \succeq 0$ of its eigenvalues $\lambda_1 \geq \cdots \geq \lambda_d \geq 0$, denote*

(i) $\overset{s}{\det}(\boldsymbol{X}) = \prod_{i \in [s]} \lambda_i$,

(ii) $\underset{s}{\det}(\boldsymbol{X}) = \prod_{i \in [d-s+1,d]} \lambda_i$.

Note that for any matrix $\boldsymbol{X}$, $\overset{s}{\det}(\boldsymbol{X})$ denotes the product of the s largest eigenvalues and $\underset{s}{\det}(\boldsymbol{X})$ denotes the product of the s smallest eigenvalues. In fact, the following observation shows that the objective function of MESP (1) can be represented by the function $\overset{s}{\det}(\cdot)$

Observation 1 $\det\left(\boldsymbol{C}_{S,S}\right) = \overset{s}{\det}\left(\sum_{i \in S} \boldsymbol{v}_i \boldsymbol{v}_i^\top\right)$.

Proof. Note that $\boldsymbol{C}_{S,S} = \boldsymbol{V}_S^\top \boldsymbol{V}_S$. Suppose matrix $\boldsymbol{V}_S^\top \boldsymbol{V}_S$ has eigenvalues $\lambda_1 \geq \cdots \geq \lambda_s \geq 0$, which correspond to the s largest eigenvalues of $\boldsymbol{V}_S \boldsymbol{V}_S^\top$. Therefore, we must have

$$\det\left(\boldsymbol{C}_{S,S}\right) = \det\left(\boldsymbol{V}_S^\top \boldsymbol{V}_S\right) = \prod_{i \in [s]} \lambda_i = \overset{s}{\det}\left(\boldsymbol{V}_S \boldsymbol{V}_S^\top\right) = \overset{s}{\det}\left(\sum_{i \in S} \boldsymbol{v}_i \boldsymbol{v}_i^\top\right).$$

□

Let us introduce the binary variables $\boldsymbol{x} \in \{0,1\}^n$ where for each $i \in [n]$, $x_i = 1$ if the i-th column vector $\boldsymbol{v}_i$ is chosen, and 0 otherwise. Then according to Observation 1, MESP (1) can be reformulated as

$$\text{(MESP)} \quad z^* := \max_{\boldsymbol{x}} \left\{ \log \overset{s}{\det}\left(\sum_{i \in [n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top\right) : \sum_{i \in [n]} x_i = s, \boldsymbol{x} \in \{0,1\}^n \right\}. \tag{2}$$

Note that in this paper, we assume $s \leq d \leq n$. However, it is worth mentioning that when $d \leq s \leq n$, MESP becomes the well-known D-Optimal design problem, a classic problem in statistics (de Aguiar et al. 1995, Pukelsheim 2006).

The following proposition summarizes the properties of the objective function in MESP (2).

Proposition 1 *The objective function of MESP* (2) *is (i) discrete-submodular, (ii) non-monotonic, (iii) neither concave nor convex, and (iv) not always nonnegative.*

Proof. See Appendix A.1. □

The non-monotonicity and possible-negativity of the objective function (2) imply that the existing approximation results for maximizing monotonic or nonnegative submodular problems (Charikar et al. 2000, Sharma et al. 2015) are not directly applicable to MESP. The non-concavity motivates us to explore a new equivalent convex integer program of MESP.

2.2. Lagrangian Dual (LD) of MESP

In this subsection, we will develop the Lagrangian dual (LD) of MESP (2). First, let us introduce an auxiliary matrix $\boldsymbol{X} \in \mathbb{R}^{d\times d}$ and reformulate MESP (2) as

$$\text{(MESP)} \quad z^* := \max_{\boldsymbol{x},\boldsymbol{X}\succeq 0} \left\{ \log \overset{s}{\det}(\boldsymbol{X}) : \sum_{i\in[n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top \succeq \boldsymbol{X}, \sum_{i\in[n]} x_i = s, \boldsymbol{x}\in\{0,1\}^n \right\}. \tag{3}$$

By dualizing the constraint $\sum_{i\in[n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top \succeq \boldsymbol{X}$, we can obtain the LD of MESP (3). Before developing the LD formulation, we would like to establish the convex conjugate of the objective function in MESP (3).

Lemma 1 *For a $d\times d$ matrix $\boldsymbol{\Lambda}\succ 0$, we have*

$$\max_{\boldsymbol{X}\succeq 0} \left\{ \log \overset{s}{\det}(\boldsymbol{X}) - \operatorname{tr}(\boldsymbol{X}\boldsymbol{\Lambda}) \right\} = -\log \det_s(\boldsymbol{\Lambda}) - s, \tag{4}$$

where function $\det_s(\cdot)$ is defined in Definition 1.

Proof. See Appendix A.2. □

Using the result in Lemma 1, we are able to show the Lagrangian dual formulation of MESP.

Theorem 1 *The optimization problem below is the Lagrangian dual of MESP (3)*

$$\textit{(LD)} \quad z^{LD} := \min_{\boldsymbol{\Lambda}\succeq 0,\nu,\boldsymbol{\mu}\in\mathbb{R}^n_+} \left\{ -\log \det_s(\boldsymbol{\Lambda}) + s\nu + \sum_{i\in[n]} \mu_i - s : \nu + \mu_i \geq \boldsymbol{v}_i^\top \boldsymbol{\Lambda} \boldsymbol{v}_i, \forall i\in[n] \right\}, \tag{5}$$

and its optimal value provides an upper bound of MESP, i.e., $z^{LD} \geq z^*$.

Proof. We let $\boldsymbol{\Lambda}\succ 0$ denote the Lagrange multiplier associated with the constraint $\sum_{i\in[n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top \succeq \boldsymbol{X}$ in MESP (3). Thus, the resulting dual problem is

$$z^{LD} := \min_{\boldsymbol{\Lambda}\succ 0} \left\{ \max_{\boldsymbol{x},\boldsymbol{X}\succeq 0} \left\{ \log \overset{s}{\det}(\boldsymbol{X}) - \operatorname{tr}(\boldsymbol{X}\boldsymbol{\Lambda}) + \sum_{i\in[n]} x_i \boldsymbol{v}_i^\top \boldsymbol{\Lambda} \boldsymbol{v}_i : \sum_{i\in[n]} x_i = s, \boldsymbol{x}\in\{0,1\}^n \right\} \right\}. \tag{6}$$

Note that the inner maximization problem above can be separated into two parts: (i) maximization over $\boldsymbol{X}$ and (ii) maximization over $\boldsymbol{x}$.

(i) For the maximization over $\boldsymbol{X}$, applying the identity in Lemma 1, we have

$$\max_{\boldsymbol{X}\succeq 0} \left\{ \log \overset{s}{\det}(\boldsymbol{X}) - \operatorname{tr}(\boldsymbol{X}\boldsymbol{\Lambda}) \right\} = -\log \det_s(\boldsymbol{\Lambda}) - s.$$

(ii) For the maximization over $\boldsymbol{x}$, it is known that optimizing a linear function over a cardinality constraint is equivalent to its continuous relaxation, which leads to that

$$\max_{\boldsymbol{x}} \left\{ \sum_{i\in[n]} x_i \boldsymbol{v}_i^\top \boldsymbol{\Lambda} \boldsymbol{v}_i : \sum_{i\in[n]} x_i = s, \boldsymbol{x}\in\{0,1\}^n \right\} = \min_{\nu,\boldsymbol{\mu}\in\mathbb{R}^n_+} \left\{ s\nu + \sum_{i\in[n]} \mu_i : \nu + \mu_i \geq \boldsymbol{v}_i^\top \boldsymbol{\Lambda} \boldsymbol{v}_i, \forall i\in[n] \right\},$$

where the right-hand side is the dual of the continuous relaxation of the left-hand side.

Plugging the above results (i.e., Parts (i) and (ii)) into the dual problem (6) and combining the minimization problems over $(\boldsymbol{\Lambda}, \nu, \boldsymbol{\mu})$ together, we arrive at (5).

Further, the inequality $z^* \leq z^{LD}$ holds due to the weak duality. □

2.3. Primal Characterization (PC) of LD and Convex Integer Program of MESP

In this subsection, we will show the primal characterization (PC) of LD (5), which inspires us an equivalent convex integer program of MESP (2).

According to the standard result (see, e.g., Bertsekas 1982, Lemaréchal and Renaud 2001) on a primal characterization of the Lagrangian dual, we have

$$\text{(PC)} \quad z^{LD} := \max_{w,\boldsymbol{x},\boldsymbol{X}\succ 0} \Bigg\{ w : \sum_{i\in[n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top \succeq \boldsymbol{X},$$
$$(w,\boldsymbol{x},\boldsymbol{X}) \in \operatorname{conv}\Bigg\{ (w,\boldsymbol{x},\boldsymbol{X}) : w \leq \log \overset{s}{\det}(\boldsymbol{X}), \sum_{i\in[n]} x_i = s, \boldsymbol{x} \in \{0,1\}^n \Bigg\} \Bigg\}.$$

Usually, the convex hull is difficult to obtain, and thus alternatively, we will derive the primal characterization through the dual formulation of LD (5).

The primal characterization relies on the following results. First, for any given $\boldsymbol{\lambda} \in \mathbb{R}^d$, let us define a unique integer k based on its sorted elements as below.

Lemma 2 (lemma 14, Nikolov 2015) *Given a vector $\boldsymbol{\lambda} \in \mathbb{R}^d$ with its elements sorted by $\lambda_1 \geq \cdots \geq \lambda_d$ and an integer $s \in [d]$, there exists a unique integer $0 \leq k < s$ such that $\lambda_k > \frac{1}{s-k}\sum_{i\in[k+1,d]} \lambda_i \geq \lambda_{k+1}$, where by convention $\lambda_0 = \infty$.*

Throughout this paper, we use k to denote the unique integer in Lemma 2. Next, we define the objective function of the primal characterization below, which can be also found in Nikolov (2015).

Definition 2 *For a $d \times d$ matrix $\boldsymbol{X} \succeq 0$ of its eigenvalues $\lambda_1 \geq \cdots \geq \lambda_d \geq 0$, let us denote*

$$\Gamma_s(\boldsymbol{X}) = \log\left(\prod_{i\in[k]} \lambda_i\right) + (s-k)\log\left(\frac{1}{s-k}\sum_{i\in[k+1,d]} \lambda_i\right),$$

where the unique integer k is defined in Lemma 2.

We are now ready to derive the convex conjugate of the objective function in LD (5).

Lemma 3 *Given a $d \times d$ matrix $\boldsymbol{X} \succeq 0$ with rank $r \in [s,d]$, suppose that the eigenvalues of $\boldsymbol{X}$ are $\lambda_1 \geq \cdots \geq \lambda_r > \lambda_{r+1} = \cdots = \lambda_d = 0$ and $\boldsymbol{X} = \boldsymbol{Q}\operatorname{Diag}(\boldsymbol{\lambda})\boldsymbol{Q}^\top$ with an orthonormal matrix $\boldsymbol{Q}$. Then*

(i)

$$\min_{\boldsymbol{\Lambda}\succ 0} \Big\{ -\log \underset{s}{\det}(\boldsymbol{\Lambda}) + \operatorname{tr}(\boldsymbol{X}\boldsymbol{\Lambda}) \Big\} = \min_{\substack{\boldsymbol{\beta}\in\mathbb{R}^d_+,\\ 0<\beta_1\leq\cdots\leq\beta_d}} \Bigg\{ -\sum_{i\in[s]} \log(\beta_i) + \sum_{i\in[d]} \lambda_i \beta_i \Bigg\}, \tag{7}$$

(ii)

$$\min_{\substack{\boldsymbol{\beta}\in\mathbb{R}_+^d,\\ 0<\beta_1\leq\cdots\leq\beta_d}} \left\{-\sum_{i\in[s]}\log(\beta_i)+\sum_{i\in[d]}\lambda_i\beta_i\right\}=\Gamma_s(\boldsymbol{X})+s. \tag{8}$$

Proof. See Appendix A.3. □

With the convex conjugate of the objective function in LD (5), using the Lagrangian dual method, we are able to derive its dual problem and also show the primal characterization below.

Theorem 2 *LD* (5) *has the following primal characterization, i.e.,*

$$\text{(PC)}\quad z^{LD}:=\max_{\boldsymbol{x}}\left\{\Gamma_s\left(\sum_{i\in[n]}x_i\boldsymbol{v}_i\boldsymbol{v}_i^\top\right):\sum_{i\in[n]}x_i=s,\boldsymbol{x}\in[0,1]^n\right\}, \tag{9}$$

where function $\Gamma_s(\cdot)$ *can be found in Definition 2.*

Proof. In LD (5), let us introduce Lagrangian multiplies $\boldsymbol{x}$ associated with the constraints. Since $z^{LD}\geq z^*$ and the constraint system of LD (5) satisfies the relaxed Slater condition, according to theorem 3.2.2 in Ben-Tal and Nemirovski (2012), the strong duality holds, i.e.,

$$z^{LD}:=\max_{\boldsymbol{x}\in\mathbb{R}_+^n}\left\{\min_{\boldsymbol{\Lambda}\succ 0,\nu,\boldsymbol{\mu}\in\mathbb{R}_+^n}\left\{-\log\det_s(\boldsymbol{\Lambda})+s\nu+\sum_{i\in[n]}\mu_i-s+\sum_{i\in[n]}x_i(\boldsymbol{v}_i^\top\boldsymbol{\Lambda}\boldsymbol{v}_i-\nu-\mu_i)\right\}\right\}.$$

The inner minimization above can be separated into two parts: (i) minimization over $\boldsymbol{\Lambda}$ and (ii) minimization over $(\nu,\boldsymbol{\mu})$, which are discussed below.

(i) Let $\boldsymbol{X}=\sum_{i\in[n]}x_i\boldsymbol{v}_i\boldsymbol{v}_i^\top$. For the minimization over $\boldsymbol{\Lambda}$, applying the identities (7) and (8) in Lemma 3 and using the fact that $\sum_{i\in[n]}x_i\boldsymbol{v}_i^\top\boldsymbol{\Lambda}\boldsymbol{v}_i=\operatorname{tr}(\boldsymbol{X}\boldsymbol{\Lambda})$, we have

$$\min_{\boldsymbol{\Lambda}\succ 0}\left\{-\log\det_s(\boldsymbol{\Lambda})+\operatorname{tr}(\boldsymbol{X}\boldsymbol{\Lambda})\right\}-s=\Gamma_s(\boldsymbol{X}).$$

(ii) For the minimization over $(\nu,\boldsymbol{\mu})$, we have

$$\min_{\nu,\boldsymbol{\mu}\in\mathbb{R}_+^n}\left\{s\nu+\sum_{i\in[n]}\mu_i+\sum_{i\in[n]}x_i(-\nu-\mu_i)\right\}=\begin{cases}0, & \text{if } \sum_{i\in[n]}x_i=s, x_i\leq 1,\forall i\in[n],\\ -\infty, & \text{otherwise.}\end{cases}$$

Putting the above two pieces together, we arrive at (9). □

We remark that PC (9) has the same objective function as another convex relaxation proposed by Nikolov (2015), but we distinguish our formulation from Nikolov (2015)'s in the following three aspects: (i) We derive the primal characterization from a Lagrangian dual perspective, which is also applicable to the A-Optimality (see Section 6) and enables us to derive supdifferentials of the objective function; (ii) Our PC (9) can be stronger than the one in Nikolov (2015) due to the extra constraints $x_i \le 1$ for each $i \in [n]$; and (iii) LD (5) and PC (9) together are critical to the analysis of the local search algorithm in Section 4.

The PC (9) is a concave maximization problem and is efficiently solvable. In the next subsection, we will introduce the Frank-Wolfe algorithm to solve it. However, according to Definition 2, the objective function $\Gamma_s(\cdot)$ might not be differentiable. Fortunately, the following result shows how to derive its supdifferentials.

Proposition 2 *Given a $d \times d$ matrix $\boldsymbol{X} \succeq 0$ with rank $r \in [s, d]$, suppose that its eigenvalues are $\lambda_1 \ge \cdots \ge \lambda_r > \lambda_{r+1} = \cdots = \lambda_d = 0$ and $\boldsymbol{X} = \boldsymbol{Q}\operatorname{Diag}(\boldsymbol{\lambda})\boldsymbol{Q}^\top$ with an orthonormal matrix $\boldsymbol{Q}$. Then the supdifferential of the function $\Gamma_s(\cdot)$ at $\boldsymbol{X}$ that is denoted by $\partial\Gamma_s(\boldsymbol{X})$ is*

$$\partial\Gamma_s(\boldsymbol{X}) = \Bigg\{\boldsymbol{Q}\operatorname{Diag}(\boldsymbol{\beta})\boldsymbol{Q}^\top : \boldsymbol{X} = \boldsymbol{Q}\operatorname{Diag}(\boldsymbol{\lambda})\boldsymbol{Q}^\top, \boldsymbol{Q} \text{ is orthonormal}, \lambda_1 \ge \cdots \ge \lambda_d,$$
$$\boldsymbol{\beta} \in \operatorname{conv}\left\{\boldsymbol{\beta} : \beta_i = \frac{1}{\lambda_i}, \forall i \in [k], \beta_i = \frac{s-k}{\sum_{i\in[k+1,d]}\lambda_i}, \forall i \in [k+1, r], \beta_i \ge \beta_r, \forall i \in [r+1, d]\right\}\Bigg\},$$

where the unique integer k follows from Lemma 2. Note that the function $\Gamma_s(\cdot)$ is differentiable whenever $\boldsymbol{X}$ is a positive-definite matrix and the unique supgradient becomes the gradient.

Proof. First, let us define function $\gamma_s(\cdot)$ as below

$$\gamma_s(\boldsymbol{\lambda}) := \min_{\substack{\boldsymbol{\beta}\in\mathbb{R}^d_+,\\ 0<\beta_1\le\cdots\le\beta_d}} \left\{-\sum_{i\in[s]}\log(\beta_i) + \sum_{i\in[d]}\lambda_i\beta_i\right\} = \Gamma_s(\boldsymbol{X}) + s, \tag{10}$$

where the equation stems from the identity (8) in Lemma 3.

Since function $\Gamma_s(\boldsymbol{X})$ is invariant under all the permutations of its eigenvalues, according to Corollary 2.5 in Lewis (1995), we have

$$\partial\Gamma_s(\boldsymbol{X}) = \left\{\boldsymbol{Q}\operatorname{Diag}(\boldsymbol{\beta})\boldsymbol{Q}^\top : \boldsymbol{X} = \boldsymbol{Q}\operatorname{Diag}(\boldsymbol{\lambda})\boldsymbol{Q}^\top, \boldsymbol{Q} \text{ is orthonormal}, \boldsymbol{\beta} \in \partial\gamma_s(\boldsymbol{\lambda})\right\}.$$

Further, by Corollary 23.5.3 in Rockafellar (1970), the supdifferential of the concave function $\gamma_s(\boldsymbol{\lambda})$ is the convex hull of all the optimal solutions $\boldsymbol{\beta}^*$ of the minimization problem in (10). From the proof of Lemma 3, any optimal solution $\boldsymbol{\beta}^*$ satisfies

$$\beta_i^* = \frac{1}{\lambda_i}, \forall i \in [k], \beta_i^* = \frac{s-k}{\sum_{i\in[k+1,d]}\lambda_i}, \forall i \in [k+1, r], \beta_i^* \ge \beta_r^*, \forall i \in [r+1, d].$$

Hence, the supdifferential of function $\gamma_s(\boldsymbol{\lambda})$ at $\boldsymbol{\lambda}$ is

$$\partial\gamma_s(\boldsymbol{\lambda}) = \operatorname{conv}\left\{\boldsymbol{\beta} : \beta_i = \frac{1}{\lambda_i}, \forall i \in [k], \beta_i = \frac{s-k}{\sum_{i\in[k+1,d]}\lambda_i}, \forall i \in [k+1, r], \beta_i \ge \beta_r, \forall i \in [r+1, d]\right\}.$$

This completes the proof. □

As a side product of PC (9), we observe that if we enforce its variables $\boldsymbol{x}$ to be binary, we can arrive at an equivalent convex integer program for MESP.

Theorem 3 *MESP can be formulated as the following convex integer program*

$$\text{(MESP)} \quad z^* := \max_{\boldsymbol{x}} \left\{ \Gamma_s \left(\sum_{i\in[n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top \right) : \sum_{i\in[n]} x_i = s, \boldsymbol{x} \in \{0,1\}^n \right\}. \tag{11}$$

Proof. See Appendix A.4. □

We close this subsection by showing that under three special cases, the optimal value of PC (9) is equal to that of MESP, i.e., $z^{LD} = z^*$.

Proposition 3 *The optimal value of PC* (9) *is equal to* z^**, i.e.,* $z^{LD} = z^*$ *provided the following three special cases: (i)* $\boldsymbol{C}$ *is diagonal; (ii)* $s = 1$*; and (iii)* $s = n$*.*

Proof. See Appendix A.5. □

The results above demonstrate that the optimal value of the proposed PC (9) can be close to that of MESP. We will further numerically illustrate this property of PC (9) in Section 5.

3. Frank-Wolfe Algorithm, Sampling Algorithm, and its Deterministic Implementation

In this section, we apply the Frank-Wolfe algorithm to solving PC (9) and derive its convergence rate. We then study a randomized sampling algorithm for MESP and prove its approximation bound. We also show the deterministic implementation of the sampling algorithm with the same performance guarantee.

3.1. Solving PC (9) using Frank-Wolfe Algorithm

In this subsection, we will investigate the Frank-Wolfe algorithm for solving PC (9). We define a feasible solution $\widehat{\boldsymbol{x}}$ to be an α-optimal soluton to PC (9) if the inequality $\Gamma_s(\sum_{i\in[n]} \widehat{x}_i \boldsymbol{v}_i \boldsymbol{v}_i^\top) \geq z^{LD} - \alpha$ with $\alpha \in (0,\infty)$. Given a target accuracy α, our proposed Frank-Wolfe algorithm will return an α-optimal solution to PC (9).

The proposed Frank-Wolfe algorithm proceeds as follows. We denote PC (9) to be the primal problem and LD (5) to be the dual problem. At each iteration t, we set the step size $\epsilon_t := \frac{2}{t+2}$. For the current feasible primal solution $\boldsymbol{x}^t$, we let $\boldsymbol{X}^t = \sum_{i\in[n]} x_i^t \boldsymbol{v}_i \boldsymbol{v}_i^\top$ and then compute the eigendecomposition of matrix $\boldsymbol{X}^t$ with eigenvalues $\lambda_1 \geq \cdots \geq \lambda_d$ and an orthonormal matrix $\boldsymbol{Q}$ such that $\boldsymbol{X}^t = \boldsymbol{Q}\,\mathrm{Diag}(\boldsymbol{\lambda})\boldsymbol{Q}^\top$. Next, we compute the integer k according to Lemma 2 and construct a new vector $\boldsymbol{\beta}^t \in \mathbb{R}_+^d$ as

$$\beta_i^t = \frac{1}{\lambda_i}, \forall i \in [k], \beta_i^t = \frac{s-k}{\sum_{i\in[k+1,d]} \lambda_i}, \forall i \in [k+1,d].$$

Thus, let us denote the dual variable $\boldsymbol{\Lambda}^t = \boldsymbol{Q}\,\mathrm{Diag}(\boldsymbol{\beta}^t)\boldsymbol{Q}^\top$, which is also a supgradient of function $\Gamma_s(\cdot)$ at $\boldsymbol{X}^t$ in view of Proposition 2. Then we obtain the other two dual variables $(\nu^t, \boldsymbol{\mu}^t)$ of LD (5) by solving the following minimization problem with a closed-form optimal solution:

$$(\nu^t, \boldsymbol{\mu}^t) := \operatorname*{arg\,min}_{\nu, \boldsymbol{\mu} \in \mathbb{R}^n_+} \left\{ s\nu + \sum_{i\in[n]} \mu_i - s : \nu + \mu_i \geq \boldsymbol{v}_i^\top \boldsymbol{\Lambda}^t \boldsymbol{v}_i, \forall i \in [n] \right\},$$

i.e., suppose that $\boldsymbol{\sigma}$ is a permutation of $[n]$ such that $\boldsymbol{v}_{\sigma(1)}^\top \boldsymbol{\Lambda}^t \boldsymbol{v}_{\sigma(1)} \geq \cdots \geq \boldsymbol{v}_{\sigma(n)}^\top \boldsymbol{\Lambda}^t \boldsymbol{v}_{\sigma(n)}$, then

$$\nu^t = \boldsymbol{v}_{\sigma(s)}^\top \boldsymbol{\Lambda}^t \boldsymbol{v}_{\sigma(s)}, \mu^t_{\sigma(i)} = \begin{cases} \boldsymbol{v}_{\sigma(i)}^\top \boldsymbol{\Lambda}^t \boldsymbol{v}_{\sigma(i)} - \boldsymbol{v}_{\sigma(s)}^\top \boldsymbol{\Lambda}^t \boldsymbol{v}_{\sigma(s)}, & \forall i \in [s] \\ 0, & \forall i \in [s+1, n] \end{cases}.$$

According to Lemma 3, the construction of $\boldsymbol{\Lambda}^t$ implies that $\Gamma_s(\boldsymbol{X}^t) = -\log \det_s(\boldsymbol{\Lambda}^t)$. Thus, the duality gap at current iteration only relies on $s\nu^t + \sum_{i\in[n]} \mu_i^t - s$. We check if the smallest duality gap is less than the threshold α or not. If "Yes", then we terminate the algorithm. Otherwise, we keep on running the algorithm by: (i) deriving the supgradient of PC (9) at the current solution $\boldsymbol{x}^t$, which is $\boldsymbol{g}^t := (\boldsymbol{v}_1^\top \boldsymbol{\Lambda}^t \boldsymbol{v}_1, \cdots, \boldsymbol{v}_n^\top \boldsymbol{\Lambda}^t \boldsymbol{v}_n)^\top$; (ii) computing the incumbent solution $\widehat{\boldsymbol{x}}^t := \arg\max_{\boldsymbol{x}} \{(\boldsymbol{g}^t)^\top \boldsymbol{x} : \sum_{i\in[n]} x_i = s, \boldsymbol{x} \in [0,1]^n\}$, i.e.,

$$\widehat{x}^t_{\sigma(i)} = \begin{cases} 1, & \forall i \in [s] \\ 0, & \forall i \in [s+1, n] \end{cases};$$

and (iii) updating the solution $\boldsymbol{x}^{t+1} := \epsilon_t \widehat{\boldsymbol{x}}^t + (1-\epsilon_t)\boldsymbol{x}^t$. The detailed implementation can be found in Algorithm 1.

Compared to the other first-order methods, the Frank-Wolfe Algorithm 1 is known to deliver a sparse incumbent solution at each iteration (Freund and Grigas 2016), enabling us to study the size of the support of its output. To begin with, let us introduce the following key lemma.

Lemma 4 *Suppose that for any size-s subset $S \subseteq [n]$, the columns $\{\boldsymbol{v}_i\}_{i\in S}$ are linearly independent. Let $\mathbb{D} := \{\boldsymbol{x} \in \mathbb{R}^n : \sum_{i\in[n]} x_i = s, \boldsymbol{x} \in [0,1]^n\}$. Then for any $\boldsymbol{x} \in$ relint$(\mathbb{D})$, we have*

$$\nabla^2 \Gamma_s \left(\sum_{i\in[n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top \right) \succeq -\frac{\lambda^2_{\max}(\boldsymbol{C})}{\delta^2} \boldsymbol{I}_n, \tag{12}$$

where the constant $\delta := \min_{S\subseteq[n], |S|=s} \lambda_{\min}(\boldsymbol{C}_{S,S})$.

Proof. See Appendix A.6. □

The inequality in Lemma 4 implies that the Hessian of the objective function $\Gamma_s(\cdot)$ of PC (9) is lower bounded. Based upon this result, we are able to derive the rate of convergence of the proposed Frank-Wolfe Algorithm 1.

Theorem 4 *Let $\widehat{\boldsymbol{x}}$ denote the output of Frank-Wolfe Algorithm 1. Suppose that (a) for any subset $S \subseteq [n]$ with $|S| = s$, the columns $\{\boldsymbol{v}_i\}_{i\in S}$ are linearly independent, and (b) $\widehat{\boldsymbol{x}}$ is an α-optimal solution of PC (9) for some $\alpha \in (0, \infty)$. Then*

Algorithm 1 Frank-Wolfe Algorithm

Input: $n \times n$ matrix $\boldsymbol{C} \succeq 0$ of rank d, integer $s \in [d]$ and target accuracy $\alpha \in (0, \infty)$
Let $\boldsymbol{C} = \boldsymbol{V}^\top \boldsymbol{V}$ denote its Cholesky factorization where $\boldsymbol{V} \in \mathbb{R}^{d \times n}$
Let $\boldsymbol{v}_i \in \mathbb{R}^d$ denote the i-th column vector of $\boldsymbol{V}$ for each $i \in [n]$
Initialize a feasible solution $\boldsymbol{x}^0$ of PC (9), the number of steps $t = 0$ and the duality gap $\Delta = \infty$
do
    Let $\epsilon_t := \frac{2}{t+2}$
    Let $\boldsymbol{X}^t = \sum_{i \in [n]} x_i^t \boldsymbol{v}_i \boldsymbol{v}_i^\top$ with eigenvalues $\lambda_1 \geq \cdots \geq \lambda_d$ and compute $\boldsymbol{X}^t = \boldsymbol{Q} \operatorname{Diag}(\boldsymbol{\lambda}^t) \boldsymbol{Q}^\top$
    Compute k according to Lemma 2
    Compute the new vector $\boldsymbol{\beta}$: $\beta_i^t = \frac{1}{\lambda_i}$ for each $i \in [k]$ and $\frac{s-k}{\sum_{i \in [k+1,d]} \lambda_i}$, otherwise
    Let $\boldsymbol{\Lambda}^t = \boldsymbol{Q} \operatorname{Diag}(\boldsymbol{\beta}) \boldsymbol{Q}^\top$
    Let $\boldsymbol{\sigma}$ be a permutation of $[n]$ such that $\boldsymbol{v}_{\sigma(1)}^\top \boldsymbol{\Lambda}^t \boldsymbol{v}_{\sigma(1)} \geq \cdots \geq \boldsymbol{v}_{\sigma(n)}^\top \boldsymbol{\Lambda}^t \boldsymbol{v}_{\sigma(n)}$
    Let $\nu^t = \boldsymbol{v}_{\sigma(s)}^\top \boldsymbol{\Lambda}^t \boldsymbol{v}_{\sigma(s)}, \mu_{\sigma(i)}^t = \boldsymbol{v}_{\sigma(i)}^\top \boldsymbol{\Lambda}^t \boldsymbol{v}_{\sigma(i)} - \nu^t$ for each $i \in [s]$ and 0, otherwise
    Let $\widehat{x}_{\sigma(i)}^t = 1$ for all $i \in [s]$ and 0, otherwise
    Update $\boldsymbol{x}^{t+1} := \epsilon_t \widehat{\boldsymbol{x}}^t + (1 - \epsilon_t) \boldsymbol{x}^t$, $\Delta := \min\{\Delta, s\nu^t + \sum_{i \in [n]} \mu_i^t - s\}$ and $t := t + 1$
while $\Delta \geq \alpha$
Output: $\boldsymbol{x}^t$

(i) The number of iterations is bounded by $t \leq 4\alpha^{-1} L \min\{s, n-s\}$, *where* $L := \delta^{-2} \lambda_{\max}^2(\boldsymbol{C})$,
(ii) The size of support of $\widehat{\boldsymbol{x}}$ *satisfies* $|\operatorname{supp}(\widehat{\boldsymbol{x}})| \leq 4\alpha^{-1} L s \min\{s, n-s\}$.

Proof. **Part (i).** Let $\mathbb{D} := \{\boldsymbol{x} : \sum_{i \in [n]} x_i = s, \boldsymbol{x} \in [0,1]^n\}$. Since $\Gamma_s(\cdot)$ is continuous in $\mathbb{D}$, thus

$$z^{LD} := \max_{\boldsymbol{x} \in \mathbb{D}} \left\{ \Gamma_s \left(\sum_{i \in [n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top \right) \right\} := - \inf_{\boldsymbol{x} \in \operatorname{relint}(\mathbb{D})} \left\{ -\Gamma_s \left(\sum_{i \in [n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top \right) \right\}.$$

Thus, it is equivalent to analyze the Frank-Wolfe Algorithm 1 on solving the right-hand side problem. The inequality (12) in Lemma 4 indicates that for any $\boldsymbol{x} \in \operatorname{relint}(\mathbb{D})$, the largest eigenvalue of the Hessian of the convex function $-\Gamma_s(\sum_{i \in [n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top)$ is bounded by L. Therefore, the L-smoothness coefficient of $-\Gamma_s(\sum_{i \in [n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top)$ in $\operatorname{relint}(\mathbb{D})$ is at most L. Given the L-smoothness, for Frank-Wolfe Algorithm 1, after iteration t, theorem 2 in Pedregosa et al. (2018) showed that the duality gap is bounded by

$$\frac{2 \sup_{\boldsymbol{x}, \boldsymbol{y} \in \operatorname{relint}(\mathbb{D})} \|\boldsymbol{x} - \boldsymbol{y}\|_2^2 L}{t+1} = \frac{4L \min\{s, n-s\}}{t+1}.$$

Given the target of the duality gap to be α, it follows that

$$t \leq 4\alpha^{-1} L \min\{s, n-s\}.$$

Part (ii). Since each iteration of Algorithm 1 increases at most s nonzero entries for the current solution, thus the size of the support of the output solution $\widehat{\boldsymbol{x}}$ is bounded by

$$|\operatorname{supp}(\widehat{\boldsymbol{x}})| \leq st \leq 4\alpha^{-1} Ls \min\{s, n-s\}.$$

□

3.2. Sampling Algorithm

In this subsection, we will introduce and analyze a randomized sampling algorithm for MESP. Given an α-optimal solution $\widehat{\boldsymbol{x}}$ of PC (9) with $\alpha \in (0, \infty)$, our proposed sampling algorithm is to sample a size-s subset $S \subseteq [n]$ with probability

$$\mathbb{P}[\tilde{S} = S] = \frac{\prod_{i \in S} \widehat{x}_i}{\sum_{\bar{S} \in \binom{[n]}{s}} \prod_{i \in \bar{S}} \widehat{x}_i}. \tag{13}$$

Algorithm 2 Efficient Implementation of Sampling Procedure (13)

Input: $n \times n$ matrix $\boldsymbol{C} \succeq 0$ of rank d and integer $s \in [d]$
Let $\widehat{\boldsymbol{x}}$ be an α-optimal solution of PC (9) with $\alpha \in (0, \infty)$
Initialize chosen set $\tilde{S} = \emptyset$ and unchosen set $T = \emptyset$
Two factors: $A_1 = \sum_{S \in \binom{[n]}{s}} \prod_{i \in S} \widehat{x}_i, A_2 = 0$
for $j = 1, \cdots, n$ **do**
    Let $A_2 = \sum_{S \in \binom{[n] \setminus (\tilde{S} \cup T)}{s-1-|\tilde{S}|}} \prod_{\tau \in S} \widehat{x}_\tau$
    Sample a $(0,1)$ uniform random variable U
    if $\widehat{x}_j A_2 / A_1 \geq U$ **then**
        Add j to set $\tilde{S}$
        $A_1 = A_2$
    else
        Add j to set T
        $A_1 = A_1 - \widehat{x}_j A_2$
    end if
end for
Output $\tilde{S}$

The detailed implementation can be found in Algorithm 2. This sampling procedure is similar to algorithm 1 in Singh and Xie (2018a), which has been proved to be computationally efficient with running time complexity $O(n \log n)$. The following result helps us to establish a relationship between the expected objective value using our sampling procedure and the optimal value of PC (9).

Lemma 5 *Given an $n\times n$ matrix $\boldsymbol{X}\succeq 0$ of rank d such that $\boldsymbol{X}=\boldsymbol{V}^\top\boldsymbol{V}$ with $\boldsymbol{V}\in\mathbb{R}^{d\times n}$ and a vector $\widehat{\boldsymbol{x}}\in\mathbb{R}^n_+$, then we have*

$$\sum_{S\in\binom{[n]}{s}}\prod_{i\in\bar{S}}\widehat{x}_i \overset{s}{\det}(\boldsymbol{V}_S\boldsymbol{V}_S^\top)\geq \exp\left[\Gamma_s\left(\sum_{i\in[n]}\widehat{x}_i\boldsymbol{v}_i\boldsymbol{v}_i^\top\right)\right].$$

Proof. The proof follows from theorem 18 in Nikolov (2015) and is thus omitted here. □

Now we are ready to show the approximation bound of the proposed sampling Algorithm 2.

Theorem 5 *Given an α-optimal solution $\widehat{\boldsymbol{x}}$ of PC (9) with $\alpha\in(0,\infty)$, the random set generated by the sampling Algorithm 2 returns a $(s\log(s)+\log(\binom{n}{s})-s\log(n)+\alpha)$-approximation bound for MESP (2), i.e., suppose the output of Algorithm 2 is the random set $\tilde{S}$, then*

$$\log\mathbb{E}\left[\overset{s}{\det}\left(\sum_{i\in\tilde{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top\right)\right]\geq z^*-s\log(s)-\log\left(\binom{n}{s}\right)+s\log(n)-\alpha.$$

Proof. Given the random set $\tilde{S}$ and its sampling probability (13), the expected exponential of the objective value of MESP (2) is equal to

$$\begin{aligned}\mathbb{E}\left[\overset{s}{\det}\left(\sum_{i\in\tilde{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top\right)\right]&=\sum_{S\in\binom{[n]}{s}}\mathbb{P}[\tilde{S}=S]\overset{s}{\det}(\boldsymbol{V}_S\boldsymbol{V}_S^\top)=\sum_{S\in\binom{[n]}{s}}\frac{\prod_{i\in S}\widehat{x}_i}{\sum_{\bar{S}\in\binom{[n]}{s}}\prod_{i\in\bar{S}}\widehat{x}_i}\overset{s}{\det}(\boldsymbol{V}_S\boldsymbol{V}_S^\top)\\ &\geq\frac{\exp\left[\Gamma_s\left(\sum_{i\in[n]}\widehat{x}_i\boldsymbol{v}_i\boldsymbol{v}_i^\top\right)\right]}{\sum_{\bar{S}\in\binom{[n]}{s}}\prod_{i\in\bar{S}}\widehat{x}_i}\geq\left(\left(\frac{s}{n}\right)^s\binom{n}{s}\right)^{-1}\exp\left[\Gamma_s\left(\sum_{i\in[n]}\widehat{x}_i\boldsymbol{v}_i\boldsymbol{v}_i^\top\right)\right]\\ &\geq\left(\left(\frac{s}{n}\right)^s\binom{n}{s}\right)^{-1}\exp\left(z^*-\alpha\right),\end{aligned}$$

where the first inequality is due to Lemma 5, the second one is from Maclaurin's inequality (Lin and Trudinger 1994), and the last one is due to the α-optimality of the solution $\widehat{\boldsymbol{x}}$ and the weak duality $z^{LD}\geq z^*$. The conclusion follows by taking logarithm on both sides of the above inequalities. □

We make the following remarks about the result in Theorem 5.

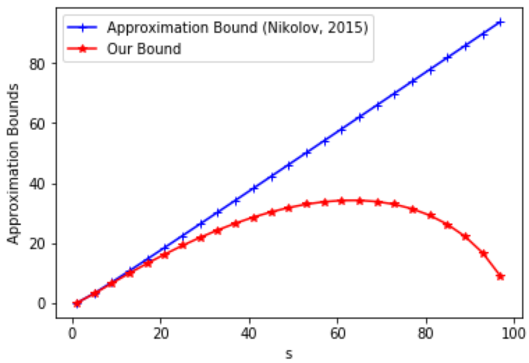
(i) This approximation bound of sampling Algorithm 2 improves the one studied in Nikolov (2015) using a different sampling scheme, where the existing approximation bound is $\log(s^s/s!)+\alpha$ (see Figure 1 for illustrations). To show this fact, it suffices to prove that

$$\left(\left(\frac{s}{n}\right)^s\binom{n}{s}\right)^{-1}\geq\frac{s!}{s^s},$$

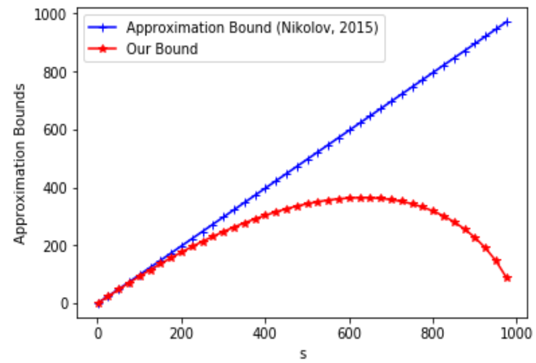
i.e.,

$$\left(\left(\frac{s}{n}\right)^s\binom{n}{s}\right)^{-1}\frac{s^s}{s!}=\frac{n^s}{n\cdots(n-s+1)}\geq 1,$$

where the inequality relies on the fact that $n\geq n-j+1$ for each $j\in[s]$.



(a) n=100



(b) n=1000

Figure 1 Approximation bounds comparison of our sampling Algorithm 2 and Nikolov (2015) with $\alpha = 0$.

(ii) The approximation bound attains zero when $s = 1$ and $s = n$.

(iii) The proof in Theorem 5 inspires us that the approximation bound depends on the sparsity of the α-optimal solution $\widehat{\boldsymbol{x}}$ to PC (9). Indeed, if we we consider the sampling probability as

$$\mathbb{P}[\tilde{S} = S] = \frac{\prod_{i\in S} \widehat{x}_i}{\sum_{\bar{S}\in\binom{\mathrm{supp}(\widehat{\boldsymbol{x}})}{s}} \prod_{i\in \bar{S}} \widehat{x}_i},$$

for any size-s subset $S \subseteq \mathrm{supp}(\widehat{\boldsymbol{x}})$. Then the approximation bound can be further improved as $(s\log(s) + \log(\binom{\widehat{n}}{s})) - s\log(\widehat{n}) + \alpha)$, where $\widehat{n} = |\mathrm{supp}(\widehat{\boldsymbol{x}})|$. This bound can be much smaller than the one in Theorem 5 if $\widehat{n} \ll n$.

Another observation is that the optimal value of the continuous relaxation of MESP (11) (i.e., PC (9)) is not too faraway from the optimal value z^*.

Corollary 1 *The optimal value of PC* (9) *is bounded by* $z^* + s\log(s) + \log(\binom{n}{s})) - s\log(n)$, *i.e.*

$$z^* \le z^{LD} \le z^* + s\log(s) + \log\left(\binom{n}{s}\right) - s\log(n).$$

Proof. The proof follows from that in Theorem 5 by observing that $z^* \ge \log \mathbb{E}[\overset{s}{\det}(\sum_{i\in\tilde{S}} \boldsymbol{v}_i\boldsymbol{v}_i^\top)]$ and α can be arbitrarily positive. □

The following instance illustrates the tightness of our analysis for the sampling Algorithm 2.

Proposition 4 *Given the sampling probability in* (13), *there exists an instance such that*

$$\log \mathbb{E}\left[\overset{s}{\det}\left(\sum_{i\in\tilde{S}} \boldsymbol{v}_i\boldsymbol{v}_i^\top\right)\right] = z^* - s\log(s) - \log\left(\binom{n}{s}\right) + s\log(n).$$

Proof. Let us consider the following example.

Example 1 *Suppose that* $d=s, n=\ell s$ *with some positive integer* ℓ, *and* $\boldsymbol{v}_{s\times(t-1)+i}=\boldsymbol{e}_i$ *for all* $(i,t)\in[s]\times[\ell]$.

Clearly, in Example 1, we have $z^*=z^{LD}=0$, and one optimal solution to PC (9) is $\widehat{x}_i=\frac{s}{n}=\frac{1}{\ell}$ for all $i\in[n]$. If we use $\widehat{\boldsymbol{x}}$ as the input of the sampling Algorithm 2, then the expected exponential of the output objective value is

$$\mathbb{E}\left[\overset{s}{\det}\left(\sum_{i\in\tilde{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top\right)\right]=\sum_{S\in\binom{[n]}{s}}\frac{\prod_{i\in S}\widehat{x}_i}{\sum_{\bar{S}\in\binom{[n]}{s}}\prod_{i\in\bar{S}}\widehat{x}_i}\overset{s}{\det}\left(\sum_{i\in S}\boldsymbol{v}_i\boldsymbol{v}_i^\top\right)=\left(\left(\frac{s}{n}\right)^s\binom{n}{s}\right)^{-1}\exp(z^*).$$

□

3.3. Deterministic Implementation

To overcome the issue of randomness from the sampling algorithms, it is common to derive their corresponding polynomial-time deterministic implementation (Nikolov 2015, Singh and Xie 2018b, Nikolov et al. 2019). In this subsection, we also develop the deterministic implementation of our proposed sampling Algorithm 2 with the same approximation bound, which is presented in Algorithm 3. The key idea of derandomization is to apply the method of conditional expectation (Alon and Spencer 2016), which requires to compute an auxiliary function regarding the conditional expected value of the function $\overset{s}{\det}(\cdot)$.

First of all, for the sake of notational convenience, let us introduce the elementary symmetric polynomials.

Definition 3 *For any vector* $\boldsymbol{x}\in\mathbb{R}^n$ *and a positive integer* ℓ, *we define the elementary symmetric polynomial of degree* ℓ *as*

$$E_\ell(\boldsymbol{x})=\sum_{S\in\binom{[n]}{\ell}}\prod_{i\in S}x_i.$$

In the deterministic Algorithm 3, given an α-optimal solution to PC (9) and a subset $T\subseteq[n]$ such that $|T|=t\le s$, according to the sampling probability (13), the conditional expected exponential of the objective value of MESP is equal to

$$\begin{aligned}\mathcal{H}(T)=\mathbb{E}\left[\overset{s}{\det}\left(\sum_{i\in\tilde{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top\right)|T\subseteq\tilde{S}\right]&=\sum_{\substack{S\in\binom{[n]}{s}\\T\subseteq S}}P(S|T\subseteq S)\overset{s}{\det}\left(\sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top\right)\\&=\sum_{\substack{S\in\binom{[n]}{s}\\T\subseteq S}}\frac{\prod_{i\in S\setminus T}\widehat{x}_i}{\sum_{\substack{\bar{S}\in\binom{[n]}{s}\\T\subseteq\bar{S}}}\prod_{i\in\bar{S}\setminus T}\widehat{x}_i}\det(\boldsymbol{C}_{S,S})=\frac{E_{s-|T|}(\boldsymbol{\lambda}(T))}{\sum_{\substack{\bar{S}\in\binom{[n]}{s}\\T\subseteq\bar{S}}}\prod_{i\in\bar{S}\setminus T}\widehat{x}_i}\det(\boldsymbol{C}_{T,T}),\end{aligned}\tag{14}$$

where $\boldsymbol{\lambda}(T)$ denotes the vector of eigenvalues of $\left(\boldsymbol{C}^{1/2}\boldsymbol{V}^\top(\mathbb{I}_d-(\boldsymbol{V}_T\boldsymbol{V}_T^\top)^\dagger\boldsymbol{V}_T\boldsymbol{V}_T^\top)\boldsymbol{V}\boldsymbol{C}^{1/2}\right)_{[n]\setminus T,[n]\setminus T}$, and the last equality is according to theorem 19 in Nikolov (2015). Note that the denominator in (14) can be computed efficiently according to Observation 1 in Singh and Xie (2018b) with running time complexity $O(n\log n)$. The numerator can be also computed efficiently according to the remark after theorem 19 in Nikolov (2015), which requires to compute the characteristic function of a matrix (e.g., Faddeev-LeVerrier algorithm in Hou 1998) with time complexity $O(n^4)$.

Algorithm 3 proceeds as follows. We start with an empty subset S, then for each $j\notin S$, we compute the the conditional expected exponential of the objective value of MESP provided that the j-th column $\boldsymbol{v}_j$ will be chosen, i.e., $\mathcal{H}(S\cup\{j\})$. We add j^* to S, where $j^*\in\arg\max_{j\in[n]\setminus S}\mathcal{H}(S\cup\{j\})$ and then go to next iteration. This procedure will terminate if $|S|=s$. Additionally, Algorithm 3 requires $O(ns)$ evaluations of function $\mathcal{H}(\cdot)$, thus the corresponding time complexity is $O(n^5 s)$. Hence, in practice, we recommend Algorithm 2 due to its simplicity and shorter running time.

The performance guarantee for Algorithm 3 is identical to Theorem 5, which is summarized below.

Algorithm 3 Deterministic Implementation

Input: $n\times n$ matrix $\boldsymbol{C}\succeq 0$ of rank d and integer $s\in[d]$

Let $\boldsymbol{C}=\boldsymbol{V}^\top\boldsymbol{V}$ denote its Cholesky factorization where $\boldsymbol{V}\in\mathbb{R}^{d\times n}$

Let $\boldsymbol{v}_i\in\mathbb{R}^d$ denote the i-th column vector of matrix $\boldsymbol{V}$ for each $i\in[n]$

Let $\widehat{\boldsymbol{x}}$ is an α-optimal solution $\widehat{\boldsymbol{x}}$ of PC (9) with $\alpha\in(0,\infty)$

Let set $S:=\emptyset$ denote the chosen set

for $i=1,\cdots,s$ **do**

  Let $j^*\in\arg\max_{j\in[n]\setminus S}\mathcal{H}(S\cup\{j\})$

  Add j^* to the set $\widehat{S}$

end for

Output: $\widehat{S}$

Theorem 6 *The deterministic Algorithm 3 yields the same approximation bound for MESP as the sampling Algorithm 2 , i.e, suppose that the output of Algorithm 3 outputs is $\widehat{S}$, then*

$$\log\overset{s}{\det}\left(\sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top\right)\geq z^*-s\log(s)-\log\left(\binom{n}{s}\right)+s\log(n)-\alpha.$$

4. Local Search Algorithm and its Approximation Guarantees

In this section, we investigate the widely-used local search algorithm (see, e.g., Hazimeh and Mazumder 2018, Madan et al. 2019) on solving MESP and prove its performance guarantee. The local search algorithm runs as follows: (i) First, we initialize a size-s subset $\widehat{S}\subseteq[n]$; (ii) Next, we swap one element from the set $\widehat{S}$ with one from the unchosen set $[n]\setminus\widehat{S}$, and we update the chosen set if such a movement strictly increases the objective value; and (iii) The algorithm terminates until no improvement can be found. The detailed implementation can be found in Algorithm 4.

Algorithm 4 Local Search Algorithm

Input: $n\times n$ matrix $\boldsymbol{C}\succeq 0$ of rank d and integer $s\in[d]$
Let $\boldsymbol{C}=\boldsymbol{V}^\top\boldsymbol{V}$ denote its Cholesky factorization where $\boldsymbol{V}\in\mathbb{R}^{d\times n}$
Let $\boldsymbol{v}_i\in\mathbb{R}^d$ denote the i-th column vector of matrix $\boldsymbol{V}$ for each $i\in[n]$
Initial subset $\widehat{S}\subseteq[n]$ of size s such that $\{\boldsymbol{v}_i\}_{i\in\widehat{S}}$ are linearly independent
do
    for each pair $(i,j)\in\widehat{S}\times([n]\setminus\widehat{S})$ **do**
        if $\log\overset{s}{\det}\left(\sum_{\ell\in\widehat{S}\cup\{j\}\setminus\{i\}}\boldsymbol{v}_\ell\boldsymbol{v}_\ell^\top\right)>\log\overset{s}{\det}\left(\sum_{\ell\in\widehat{S}}\boldsymbol{v}_\ell\boldsymbol{v}_\ell^\top\right)$ **then**
            Update $\widehat{S}:=\widehat{S}\cup\{j\}\setminus\{i\}$
        end if
    end for
while there is still an improvement
Output: $\widehat{S}$

Let us first derive the following technical results on the rank-one update of singular matrices, which are essential to the analysis of the local search Algorithm 4.

Lemma 6 *Consider a size-τ subset $\widehat{S}\subseteq[n]$ with $\tau\in[d]$ such that $\{\boldsymbol{v}_i\}_{i\in\widehat{S}}$ are linearly independent. Let $\boldsymbol{X}=\sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top$, and for each $i\in\widehat{S}$, let $\boldsymbol{X}_{-i}=\boldsymbol{X}-\boldsymbol{v}_i\boldsymbol{v}_i^\top$. Then for each $(i,j)\in\widehat{S}\times([n]\setminus\widehat{S})$, we have the followings*

(i) $\overset{\tau}{\det}(\boldsymbol{X})=\overset{\tau-1}{\det}(\boldsymbol{X}_{-i})\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i$,

(ii) $\begin{cases}\overset{\tau}{\det}(\boldsymbol{X}_{-i}+\boldsymbol{v}_j\boldsymbol{v}_j^\top)=\overset{\tau-1}{\det}(\boldsymbol{X}_{-i})\boldsymbol{v}_j^\top(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_j, & \text{if } \boldsymbol{v}_j\notin\operatorname{col}(\boldsymbol{X}_{-i}),\\ \overset{\tau-1}{\det}(\boldsymbol{X}_{-i}+\boldsymbol{v}_j\boldsymbol{v}_j^\top)=\overset{\tau-1}{\det}(\boldsymbol{X}_{-i})(1+\boldsymbol{v}_j^\top\boldsymbol{X}_{-i}^\dagger\boldsymbol{v}_j), & \text{otherwise},\end{cases}$

(iii) $\boldsymbol{X}^\dagger=\boldsymbol{X}_{-i}^\dagger-\frac{\boldsymbol{X}_{-i}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})}{||(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i||_2^2}-\frac{(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i\boldsymbol{v}_i^\top\boldsymbol{X}_{-i}^\dagger}{||(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i||_2^2}+\frac{(1+\boldsymbol{v}_i^\top\boldsymbol{X}_{-i}^\dagger\boldsymbol{v}_i)(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})}{||(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i||_2^4}$,

(iv) $\boldsymbol{X}_{-i}^\dagger=\boldsymbol{X}^\dagger-\frac{\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{X}^\dagger}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^2}-\frac{\boldsymbol{X}^\dagger\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^2}+\frac{\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^3\boldsymbol{v}_i\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^4}$,

(v) $\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{v}_i=1$,

(vi) $\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}^\dagger\boldsymbol{X})=\boldsymbol{0}$,

(vii) $\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i=\frac{1}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^2}$,

(viii) $\boldsymbol{v}_j^\top(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_j=\begin{cases}\boldsymbol{v}_j^\top(\boldsymbol{I}_d-\boldsymbol{X}^\dagger\boldsymbol{X})\boldsymbol{v}_j+\frac{(\boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_i)^2}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^2}, & \text{if } \boldsymbol{v}_j\notin\operatorname{col}(\boldsymbol{X}_{-i}),\\ 0, & \text{otherwise}.\end{cases}$

Proof. See Appendix A.7. □

Lemma 6 helps establish the local optimality condition (i.e., stopping criterion) of the local search Algorithm 4. That is, we first rewrite the local optimality condition as

$$\log \overset{s}{\det}\left(\sum_{\ell\in\widehat{S}\cup\{j\}\setminus\{i\}} \boldsymbol{v}_\ell\boldsymbol{v}_\ell^\top\right) - \log \overset{s-1}{\det}\left(\sum_{\ell\in\widehat{S}\setminus\{i\}} \boldsymbol{v}_\ell\boldsymbol{v}_\ell^\top\right) \le \log \overset{s}{\det}\left(\sum_{\ell\in\widehat{S}} \boldsymbol{v}_\ell\boldsymbol{v}_\ell^\top\right) - \log \overset{s-1}{\det}\left(\sum_{\ell\in\widehat{S}\setminus\{i\}} \boldsymbol{v}_\ell\boldsymbol{v}_\ell^\top\right),$$

for all $i\in\widehat{S}$ and $j\in[n]\setminus\widehat{S}$, and then use the results in Lemma 6 to simplify the both differences.

Lemma 7 *Let $\widehat{S}$ denote the output of the local search Algorithm 4 and let $\boldsymbol{X}=\sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top$. Then for each pair $(i,j)\in\widehat{S}\times([n]\setminus\widehat{S})$, the following inequality holds*

$$1\ge \left(\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{X}^\dagger\boldsymbol{v}_i\right)\boldsymbol{v}_j^\top(\boldsymbol{I}_d-\boldsymbol{X}^\dagger\boldsymbol{X})\boldsymbol{v}_j+\boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{v}_j.$$

Proof. See Appendix A.8. □

4.1. Analysis of Local Search Algorithm 4

Now we are ready to analyze the local search Algorithm 4. The main proof idea is two-fold: (i) Using the output of the local search Algorithm 4 and its local optimality condition in Lemma 7, we construct a dual feasible solution to LD (5), and (ii) we then show that the objective value of this dual feasible solution can be bounded by z^* with some extra constant.

Theorem 7 *Let $\widehat{S}$ denote the output of the local search Algorithm 4, then the set $\widehat{S}$ yields a $s\min\{\log(s),\log(n-s-n/s+2)\}$-approximation bound for MESP (2), i.e.,*

$$\log \overset{s}{\det}\left(\sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top\right)\ge z^*-s\min\left\{\log(s),\log\left(n-s-\frac{n}{s}+2\right)\right\}.$$

Proof. See Appendix A.9. □

We make the following remarks about Theorem 7.

(i) To the best of our knowledge, it is the first-known approximation bound of the local search algorithm for MESP.

(ii) The approximation bound attains the maximum when $s=\frac{n}{2}$ and it is equal to zero when $s=1$ or $s=n$.

(iii) The approximation bound is weaker than that of the sampling Algorithm 2 in Theorem 5 if the continuous relaxation can be solved to optimality or very close to optimality. That is, if $\alpha\to 0$, then we have

$$s\log(s)+\log\left(\binom{n}{s}\right)-s\log(n)\le s\min\left\{\log(s),\log\left(n-s-\frac{n}{s}+2\right)\right\}.$$

However, as we can see from the numerical study, the local search algorithm in practice is more capable to find good-quality solutions than the sampling algorithm.

(iv) The proof also relies on the sparsity of the optimal solution to PC (9). In fact, if there exists a sparse optimal solution $\boldsymbol{x}^*$ to PC (9) (i.e., $|\operatorname{supp}(\boldsymbol{x}^*)| \ll n$), then according to KKT conditions, we can drop the redundant dual constraints $\boldsymbol{v}_i^\top \boldsymbol{\Lambda} \boldsymbol{v}_i \leq \nu + \mu_i$ for each $i \in [n] \setminus \operatorname{supp}(\boldsymbol{x}^*)$ in LD (5). Therefore, following the same proof in Theorem 7, the approximation bound can be further improved as $s \min\{\log(s), \log(\widehat{n} - s - \widehat{n}/s + 2)\}$ where $\widehat{n} = |\operatorname{supp}(\boldsymbol{x}^*)|$.

The following instance shows that the proof of Theorem 7 is tight. That is, the approximation bound can not be improved if we construct a feasible $\boldsymbol{\Lambda}$ to LD (5) as

$$\boldsymbol{\Lambda} = \frac{1}{t}\left[\operatorname{tr}(\boldsymbol{X}^\dagger)(\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X}) + \boldsymbol{X}^\dagger\right], \tag{15}$$

where for the output $\widehat{S}$ of the local search Algorithm 4, we let $\boldsymbol{X} = \sum_{i \in \widehat{S}} \boldsymbol{v}_i \boldsymbol{v}_i^\top$ and $t > 0$ be a positive scaling factor.

Proposition 5 *If one follows the consruction of a feasible* $\boldsymbol{\Lambda}$ *in* (15) *to LD* (5)*, then even with the best choice of* $(\nu, \boldsymbol{\mu})$*, there exists an instance such that*

$$-\log \det_s(\boldsymbol{\Lambda}) + s\nu + \sum_{i \in [n]} \mu_i - s = z^* + s \min\left\{\log(s), \log(n - s - n/s + 2)\right\}.$$

Proof. See Appendix A.10. □

The above proposition shows the tightness of the analysis of Theorem 7. Thus, to improve the analysis of the local search Algorithm 4, one might need different ways to construct dual feasible solutions to LD (5). In fact, we show that under a certain assumption, the approximation bound of the local search algorithm can be improved.

Proposition 6 *Let* $\widehat{S}$ *denote the output of the local search Algorithm 4. Suppose that* $\boldsymbol{v}_i^\top \boldsymbol{v}_j = 0$ *for each pair* $(i,j) \in \widehat{S} \times ([n] \setminus \widehat{S})$*, then we have*

$$\log \overset{s}{\det}\left(\sum_{i \in \widehat{S}} \boldsymbol{v}_i \boldsymbol{v}_i^\top\right) \geq z^* - s \min\left\{\log\left(\frac{\lambda_{\max}(\boldsymbol{C})}{\delta}\right), \log\left(\frac{\lambda_{\max}(\boldsymbol{C})}{s\delta}(n-s) - \frac{n}{s} + 2\right)\right\},$$

where the constant δ *is defined in Lemma 4.*

Proof. See Appendix A.11. □

Compared with the bound $O(s \log s)$ in Theorem 7, the approximation bound in Proposition 6 is $O(s)$, which matches the order of the bound derived for the sampling Algorithm 2.

4.2. Efficient Implementation of the Local Search Algorithm

In this subsection, we will discuss how to efficiently implement the local search Algorithm 4 using the results in Lemma 6, and develop its corresponding time complexity.

Similar to many improving heuristics, the performance of the local search Algorithm 4 highly depends on the choice of the initial subset. In practice, we employ the greedy approach to find an initial solution. The greedy approach begins with an empty set $\widehat{S}=\emptyset$, then at each iteration, we select one element from the unchosen set $[n]\setminus S$ that maximizes the marginal increment of the objective value until $|\widehat{S}|=s$. That is, at current iteration $\ell\in[s]$, suppose that $\boldsymbol{X}=\sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top$ and $|\widehat{S}|=\ell<s$. Then by Part (ii) in Lemma 6, the next element that will be chosen is computed by

$$j^*\in\arg\max_{j\in[n]\setminus\widehat{S}}\left(\log\overset{\ell+1}{\det}(\boldsymbol{X}+\boldsymbol{v}_j\boldsymbol{v}_j^\top)-\log\overset{\ell}{\det}(\boldsymbol{X})\right)=\arg\max_{j\in[n]\setminus\widehat{S}}\boldsymbol{v}_j^\top(\boldsymbol{I}_d-\boldsymbol{X}\boldsymbol{X}^\dagger)\boldsymbol{v}_j.$$

The detailed implementation of the greedy approach can be found in Algorithm 5 at Steps 4 -10. Using the equation above and Part (iii) in Lemma 6, the greedy approach has a running time complexity of $O(s(n-s)d^2)$. Further, we show that the rank-one update techniques for the singular matrices in Lemma 6 can also improve the implementation of the local search algorithm.

One key component of the local search Algorithm 4 is the swapping procedure (i.e., Steps 6-9), which might cause the running time to be exponential in the size of the input. To avoid this, we can restrict the number of swapping iterations by simply introducing a small positive constant $\theta>0$ and replace the condition at Step 8 of Algorithm 4 by

$$\overset{s}{\det}\left(\sum_{\ell\in\widehat{S}\cup\{j\}\setminus\{i\}}\boldsymbol{v}_\ell\boldsymbol{v}_\ell^\top\right)>(1+\theta)\overset{s}{\det}\left(\sum_{\ell\in\widehat{S}}\boldsymbol{v}_\ell\boldsymbol{v}_\ell^\top\right).$$

In this way, following from the similar arguments in Madan et al. (2019), the number of swapping iterations is at most $O(Ld^3\theta^{-1}\log(s))$ where L is the encoding length of the matrix $\boldsymbol{V}$. Note that by doing so, the approximation bound in Theorem 7 now becomes $s\min\{\log(s(1+\theta)),\log((n-s)(1+\theta)-n/s+2)\}$.

On the other hand, we can use Parts (ii) and (iv) in Lemma 6 to complete the swapping and use Part (iii) in Lemma 6 to update matrix $\boldsymbol{X}^\dagger$. Hence, it takes $O(s(n-s)d^2)$ for each swapping. Thus, the local search Algorithm 5 has a polynomial-time complexity of $O(Ld^3\theta^{-1}\log(s)s(n-s)d^2)$. These results are summarized below.

Corollary 2 *The running time complexity of the local search Algorithm 5 is $O(Ld^3\theta^{-1}\log(s)s(n-s)d^2)$, where L denotes the encoding length of the matrix $\boldsymbol{V}$. In addition, the local search Algorithm 5 yields a $s\min\{\log(s(1+\theta)),\log((n-s)(1+\theta)-n/s+2)\}$-approximation bound for MESP.*

Algorithm 5 Efficient Implementation of Local Search Algorithm 4 Initialized by Greedy Solution

Input: $n \times n$ matrix $\boldsymbol{C} \succeq 0$ of rank d and integer $s \in [d]$
Let $\boldsymbol{C} = \boldsymbol{V}^\top \boldsymbol{V}$ denote its Cholesky factorization where $\boldsymbol{V} \in \mathbb{R}^{d \times n}$
Let $\boldsymbol{v}_i \in \mathbb{R}^d$ denote the i-th column vector of matrix $\boldsymbol{V}$ for each $i \in [n]$

(a) Greedy Selection

Let set $\widehat{S} := \emptyset$ denote the chosen set, $\boldsymbol{X} := \emptyset$ and $\boldsymbol{X}^\dagger := \emptyset$
for $\ell = 1, \cdots, s$ **do**
    Let $j^* \in \arg\max_{j \in [n] \setminus \widehat{S}} \boldsymbol{v}_j^\top (\boldsymbol{I}_d - \boldsymbol{X}\boldsymbol{X}^\dagger)\boldsymbol{v}_j$
    Add j^* to the set $\widehat{S}$
    Update $\boldsymbol{X}^\dagger := \boldsymbol{X}^\dagger - \frac{\boldsymbol{X}^\dagger \boldsymbol{v}_{j^*}\boldsymbol{v}_{j^*}^\top (\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X})}{\|(\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X})\boldsymbol{v}_{j^*}\|_2^2} - \frac{(\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X})\boldsymbol{v}_{j^*}\boldsymbol{v}_{j^*}^\top \boldsymbol{X}^\dagger}{\|(\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X})\boldsymbol{v}_{j^*}\|_2^2} + \frac{(1+\boldsymbol{v}_{j^*}^\top \boldsymbol{X}^\dagger \boldsymbol{v}_i)(\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X})\boldsymbol{v}_{j^*}\boldsymbol{v}_{j^*}^\top (\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X})}{\|(\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X})\boldsymbol{v}_{j^*}\|_2^4}$
    Update $\boldsymbol{X} := \boldsymbol{X} + \boldsymbol{v}_{j^*}\boldsymbol{v}_{j^*}^\top$
end for

(b) Swapping Procedure

Let θ denote a positive constant
do
    for each $i \in \widehat{S}$ **do**
        Compute $\boldsymbol{X}_{-i} = \boldsymbol{X} - \boldsymbol{v}_i \boldsymbol{v}_i^\top$, $\boldsymbol{X}_{-i}^\dagger = \boldsymbol{X}^\dagger - \frac{\boldsymbol{X}^\dagger \boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{X}^\dagger \boldsymbol{X}^\dagger}{\|\boldsymbol{X}^\dagger \boldsymbol{v}_i\|_2^2} - \frac{\boldsymbol{X}^\dagger \boldsymbol{X}^\dagger \boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{X}^\dagger}{\|\boldsymbol{X}^\dagger \boldsymbol{v}_i\|_2^2} + \frac{\boldsymbol{v}_i^\top (\boldsymbol{X}^\dagger)^3 \boldsymbol{v}_i \boldsymbol{X}^\dagger \boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{X}^\dagger}{\|\boldsymbol{X}^\dagger \boldsymbol{v}_i\|_2^4}$
        Let $j^* \in \arg\max_{j \in [n] \setminus \widehat{S}} \boldsymbol{v}_j^\top (\boldsymbol{I}_d - \boldsymbol{X}_{-i}\boldsymbol{X}_{-i}^\dagger)\boldsymbol{v}_j$
        if $\boldsymbol{v}_{j^*}^\top (\boldsymbol{I}_d - \boldsymbol{X}_{-i}\boldsymbol{X}_{-i}^\dagger)\boldsymbol{v}_{j^*} > (1+\theta)\boldsymbol{v}_i^\top (\boldsymbol{I}_d - \boldsymbol{X}_{-i}\boldsymbol{X}_{-i}^\dagger)\boldsymbol{v}_i$ **then**
            Update $\widehat{S} := \widehat{S} \cup \{j\} \setminus \{i\}$, $\boldsymbol{X} := \boldsymbol{X}_{-i} + \boldsymbol{v}_{j^*}\boldsymbol{v}_{j^*}^\top$ and $\boldsymbol{X}^\dagger := \boldsymbol{X}_{-i}^\dagger - \frac{\boldsymbol{X}_{-i}^\dagger \boldsymbol{v}_{j^*}\boldsymbol{v}_{j^*}^\top (\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})}{\|(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_{j^*}\|_2^2} - \frac{(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_{j^*}\boldsymbol{v}_{j^*}^\top \boldsymbol{X}_{-i}^\dagger}{\|(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_{j^*}\|_2^2} + \frac{(1+\boldsymbol{v}_{j^*}^\top \boldsymbol{X}_{-i}^\dagger \boldsymbol{v}_{j^*})(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_{j^*}\boldsymbol{v}_{j^*}^\top (\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})}{\|(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_{j^*}\|_2^4}$;
        end if
    end for
while there is still an update
Output: $\widehat{S}$

5. Numerical Illustrations

In this section, we present numerical experiments on two medium-sized instances in Hoffman et al. (2001) and Anstreicher (2018a), which were provided by Prof. Anstreicher, and one large-scale instance in Dey et al. (2018) to demonstrate the solution quality and computational efficiency of our proposed Frank-Wolfe Algorithm 1, sampling Algorithm 2 and local search Algorithm 4 for solving MESP. All the algorithms in this section are coded in Python 3.6 with calls to Gurobi 7.5 on a personal PC with 2.3 GHz Intel Core i5 processor and 8G of memory. The codes for these three algorithms are available at `https://github.com/yongchunli-13/Approximation-Algorithms-for-MESP`.

5.1. Numerical Experiments on Two Medium-sized Instances

In this subsection, we test the proposed algorithms on two commonly-used benchmark instances of MESP in literature and investigate their computational performance. In particular, the first instance has a covariance matrix of size 90×90 built on a temperature monitoring problem introduced in Anstreicher (2018a), denoted by $n = 90$ instance, and the second one is based on a covariance matrix of size 124×124 introduced by Hoffman et al. (2001), denoted by $n = 124$ instance. Please note that these two covariance matrices are non-singular, i.e., $n = d$. For the $n = 90$ instance, we test 8 cases with $s \in \{10, 20, \ldots, 80\}$, while for the $n = 124$ instance, we test 9 cases with $s \in \{20, \ldots, 100\}$. The computational results are displayed in Table 2 and Table 3, where we let **B&B**, **Frank-Wolfe**, **Sampling**, **Local Search** denote the Branch and Bound algorithm used in Anstreicher (2018a), the Frank-Wolfe Algorithm 1, the sampling Algorithm 2, the local search Algorithm 4, respectively. We also use **S-FW** to denote the size of the support of the continuous relaxation solution from the Frank-Wolfe Algorithm 1, use **LB-S** to denote the best lower bound from the sampling Algorithm 2, use **LB-L** to denote the lower bound from the local search Algorithm 4, use **time** to denote the total time of an algorithm spent on a case, and use **gap(%)** to denote the computational optimality gaps according to

$$100 \times \frac{z^{LD} - z^*}{z^*}, 100 \times \frac{z^* - \text{LB-S}}{z^*}, 100 \times \frac{z^* - \text{LB-L}}{z^*},$$

for Frank-Wolfe Algorithm 1, sampling Algorithm 2, local search Algorithm 4, respectively. Note that due to the randomness, we repeat the sampling Algorithm 2 one thousand times for each case and choose the best output, and its running time includes the time spent on the repetitions as well as that on running the Frank-Wolfe Algorithm 1.

Table 2 and 3 presents the numerical results. From Table 2 and 3, we can see that it can take more than two days to solve some cases to optimality using the B&B algorithm, indicating that the optimal value of MESP is in general difficult to obtain. Note that in the $n = 124$ instance, the optimal value z^* decreases when s increases from 80 to100, which demonstrates that the objective of MESP may not be monotonic with s. For both instances, the local search Algorithm 4 works quite well, where its optimality gap is always within 0.06%, and its running time is less than a second. The sampling Algorithm 2 is often worse than the local search Algorithm 4 in terms of optimality gap and computational time. It is seen that the Frank-Wolfe Algorithm 1 is quite effective, and its output can be indeed very sparse, especially when s is small.

Next, we compare two solution algorithms with the heuristic used in Anstreicher (2018a) and the results are illustrated in Figure 2(a) and 2(b). Clearly, the proposed local search Algorithm 4 performs the best among these methods. Finally, Figure 3 compares our Lagrangian dual bound z^{LD} with the best linx bound found in Anstreicher (2018a), where the latter has been shown to be superior to the other existing upper bounds of MESP on these two instances. In general, these two bounds are not comparable. However, we see that our dual bound outperforms the linx bound in some cases, especially when s is small.

Table 2 Computational results of MESP on the $n=90$ instance

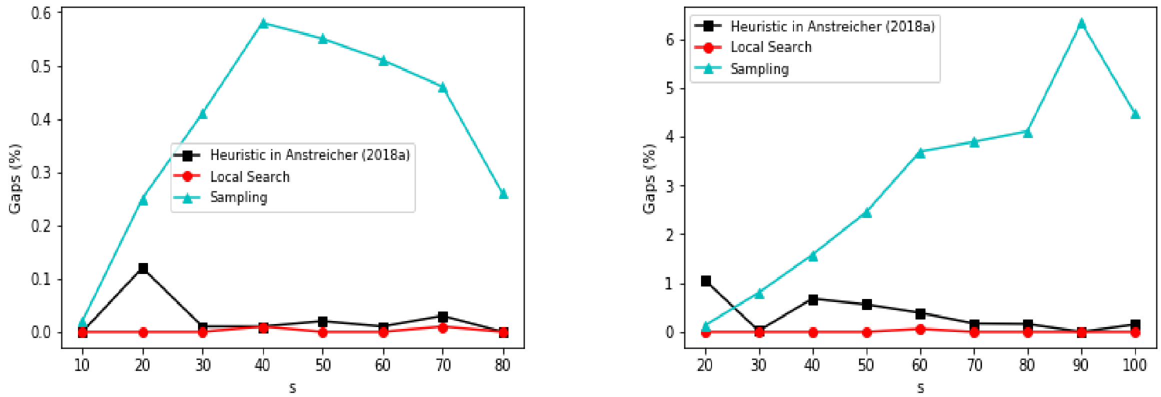
n=90	B&B[1]		Frank-Wolfe				Sampling			Local Search		
s	z^*	time[2]	z^{LD}	gap(%)	S-FW	time	LB-S	gap(%)	time	LB-L	gap(%)	time
10	58.532	2088	58.914	0.65	23	<1	58.521	0.02	18	58.532	0.00	<1[3]
20	111.482	95976	112.127	0.58	42	<1	111.207	0.25	20	111.482	0.00	<1
30	161.539	167796	162.392	0.53	60	<1	160.884	0.41	20	161.539	0.00	<1
40	209.969	187344	210.930	0.46	80	<1	208.757	0.58	19	209.958	0.01	<1
50	257.160	87912	258.115	0.37	84	<1	255.736	0.55	19	257.154	0.00	<1
60	303.019	12420	303.912	0.29	87	<1	301.474	0.51	19	303.008	0.00	<1
70	347.471	1044	348.192	0.21	89	<1	345.861	0.46	19	347.453	0.01	<1
80	389.997	36	390.382	0.10	89	<1	389.002	0.26	19	389.997	0.00	<1

[1] The optimal value and the running time of B&B algorithm are from Anstreicher (2018a)
[2] Time is in seconds
[3] The running time is less than a second

Table 3 Computational results of MESP on the $n=124$ instance

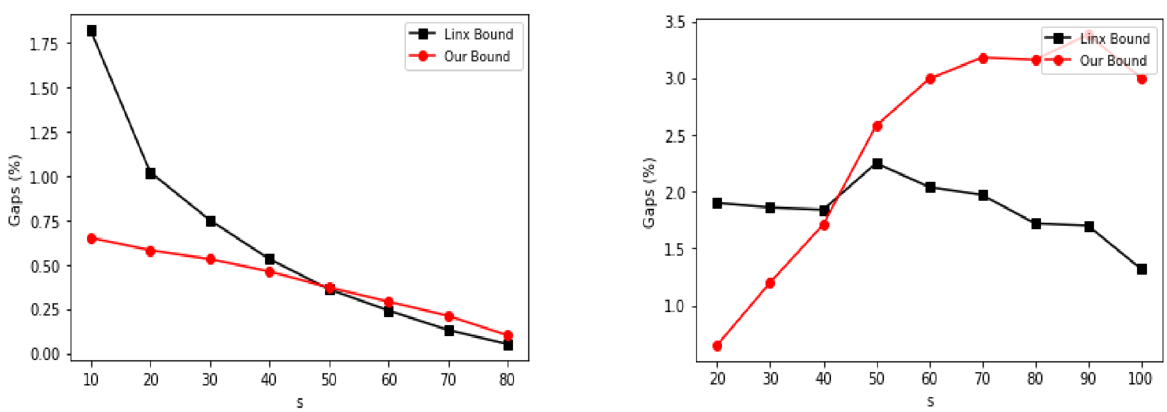
n=124	B&B[1]		Frank-Wolfe				Sampling			Local Search		
s	z^*	time[2]	z^{LD}	gap(%)	S-FW	time	LB-S	gap(%)	time	LB-L	gap(%)	time
20	77.827	756	78.337	0.65	40	1	77.726	0.13	35	77.826	0.00	<1[3]
30	106.700	1692	107.985	1.20	60	2	105.843	0.80	37	106.700	0.00	<1
40	131.055	8712	133.301	1.71	80	3	128.988	1.58	39	131.055	0.00	<1
50	149.498	186516	153.355	2.58	98	5	145.831	2.45	44	149.498	0.00	<1
60	164.012	241236	168.922	2.99	106	6	157.955	3.69	41	163.916	0.06	<1
70	172.528	136548	178.021	3.18	115	5	165.816	3.89	41	172.528	0.00	<1
80	175.091	45756	180.620	3.16	122	4	167.898	4.11	40	175.091	0.00	<1
90	171.262	17352	177.052	3.38	124	3	160.425	6.33	43	171.262	0.00	<1
100	162.865	4140	167.756	3.00	124	3	155.592	4.47	39	162.865	0.00	<1

[1] The optimal value and the running time of B&B algorithm are from Anstreicher (2018a)
[2] Time is in seconds
[3] The running time is less than a second



(a) n=90 (b) n=124

Figure 2 Optimality gap comparison of the sampling Algorithm 2, the local search Algorithm 4, and the best heuristic in Anstreicher (2018a).



(a) n=90 (b) n=124

Figure 3 Optimality gap comparison of z^{LD} and the linx bound in Anstreicher (2018a).

5.2. Numerical Experiments on a Large-scale Instance

In this subsection, we test the proposed algorithms on a large-scale instance with a 2000×2000 covariance matrix $\boldsymbol{C}$ based upon Reddit data from Dey et al. (2018). Note that for this instance, the matrix $\boldsymbol{C}$ is singular and its rank is equal to 949, i.e., $d = 949 < n = 2000$. The computational results are displayed in Table 3, where we use **B&C** to denote the branch and cut algorithm, use **UB** to denote the best upper bound output from B&C algorithm, and use UB to compute the optimality gaps for the sampling Algorithm 2 and the local search Algorithm 4. The lower bound of the B&C algorithm is always inferior to the one found by the local search algorithm and is thus not reported.

We make the following remarks of the implementation of B&C: (i) we use the warm start, i.e., we solve the continuous relaxation of MESP (11) using the cutting-plane method (i.e., at each iteration, we add a supgradient inequality) and add all the cuts into the root node, (ii) if we encounter a solution $\widehat{\boldsymbol{x}}$ with support $\widehat{S}$ such that its corresponding columns $\{\boldsymbol{v}_i\}_{i\in\widehat{S}}$ are not linearly independent, then the supgradient according to Proposition 2 is not well-defined, and thus we add no-good cut to cut it off, which is in the form of $1 \leq \sum_{i\in\widehat{S}}(1-x_i) + \sum_{i\in[n]\setminus\widehat{S}} x_i$, and (iii) we set the time limit to be $3,600$ seconds.

In Table 4, it is expected that the B&C algorithm will have difficulty in solving MESP to optimality; however, it produces a better upper bound than z^{LD}. Note that in the sampling algorithm, we only sample from the support of the output solution from the Frank-Wolfe algorithm for the sake of computational efficiency. Since we use UB to compute the optimality gaps of the sampling Algorithm 2 and the local search Algorithm 4, their true optimality gaps can be even smaller. We also observe that the solution output from the Frank-Wolfe Algorithm 1 is very sparse. The computational time of the Frank-Wolfe Algorithm 1 is longer because at each iteration, one has to compute the eigendecomposition in order to obtain the supgradient, which can be time-consuming. Again, we see that the local search Algorithm 4 outperforms the sampling Algorithm 2 both in time and solution quality. Thus, we recommend using this algorithm to solve practical problems.

Table 4 Computational results of MESP on the $n=2000$ instance

n=2000	B&C		Frank-Wolfe			Sampling			Local Search		
s	UB	time[1]	z^{LD}	S-FW	time	LB-S	gap(%)	time	LB-L	gap(%)	time
20	102.939	3600	103.007	30	119	102.608	0.32	232	102.902	0.04	21
40	185.327	3600	185.332	61	257	184.412	0.49	359	185.094	0.13	23
60	256.584	3600	256.589	93	321	254.169	0.94	463	256.281	0.12	33
80	320.812	3600	320.817	160	833	316.428	1.37	950	320.200	0.19	41
100	380.298	3600	380.307	214	1466	370.728	2.52	1333	379.081	0.32	52
120	436.336	3600	436.350	268	1935	417.858	4.23	1973	434.486	0.42	72

[1] Time is in seconds

6. Extension to the A-Optimal MESP (A-MESP)

In the section, we extend the analyses to the A-Optimal MESP (A-MESP), which instead, minimizes the trace of the inverse of $\boldsymbol{C}_{S,S}$. The A-Optimality, as an alternative measurement of information, has been widely used in the fields of experimental design (Madan et al. 2019, Nikolov et al. 2019), subdata selection (Yao and Wang 2019), and sensor placement (Moreno-Salinas et al. 2013, Xu and Doğançay 2017). Formally, A-MESP is formulated as

$$\text{(A-MESP)} \quad z_A^* := \min_S \left\{ \operatorname{tr}\left(\boldsymbol{C}_{S,S}^{-1}\right) : \quad S \subseteq [n], |S| = s \right\}. \tag{16}$$

By default, if $\boldsymbol{C}_{S,S}$ is singular, then $\operatorname{tr}\left(\boldsymbol{C}_{S,S}^{-1}\right)=\infty$.

6.1. Convex Integer Programming Formulation

Similar to Section 2, we derive an equivalent convex integer program for A-MESP (16).

First of all, we introduce the following three functions, corresponding to the objective function of another exact formulation for A-MESP (16), the objective function of the Lagrangian dual, and the objective function of the primal characterization, respectively.

Definition 4 *For a $d\times d$ matrix $\boldsymbol{X}\succeq 0$ of its eigenvalues $\lambda_1\geq\cdots\geq\lambda_d\geq 0$, let us denote*

(i) $\overset{s}{\operatorname{tr}}(\boldsymbol{X}^\dagger)=\sum_{i\in[s]}\frac{1}{\lambda_i}$,

(ii) $\underset{s}{\operatorname{tr}}(\boldsymbol{X})=\sum_{i\in[d-s+1,d]}\lambda_i$,

(iii) $\Phi_s(\boldsymbol{X})=\sum_{i\in[k]}\frac{1}{\lambda_i}+(s-k)\frac{s-k}{\sum_{i\in[k+1,d]}\lambda_i}$, *where the unique integer k is defined in Lemma 2.*

Similar to Observation 1, it is straightforward to show that

$$\operatorname{tr}\left(\boldsymbol{C}_{S,S}^{-1}\right)=\overset{s}{\operatorname{tr}}\left[\left(\sum_{i\in S}\boldsymbol{v}_i\boldsymbol{v}_i^\top\right)^\dagger\right].$$

Thus, A-MESP (16) can be reformulated as

$$\text{(A-MESP)}\quad z_A^*:=\min_{\boldsymbol{x}}\left\{\overset{s}{\operatorname{tr}}\left[\left(\sum_{i\in[n]}x_i\boldsymbol{v}_i\boldsymbol{v}_i^\top\right)^\dagger\right]:\sum_{i\in[n]}x_i=s,\boldsymbol{x}\in\{0,1\}^n\right\},\tag{17}$$

which reduces to the conventional A-Optimal design problem (Madan et al. 2019, Nikolov et al. 2019) if $d\leq s\leq n$. The following proposition summarizes the properties of the objective function of A-MESP (17).

Proposition 7 *The objective function of A-MESP* (17) *is (i) monotonic non-decreasing, (ii) neither discrete-supermodular nor discrete-submodular, and (iii) neither convex nor concave.*

Proof. See Appendix A.12. □

To derive an equivalent convex integer program, we introduce a matrix variable $\boldsymbol{X}\in\mathbb{R}^{d\times d}$ and reformulate A-MESP (17) as

$$\text{(A-MESP)}\quad z_A^*:=\min_{\boldsymbol{x},\boldsymbol{X}\succeq 0}\left\{\overset{s}{\operatorname{tr}}(\boldsymbol{X}^\dagger):\sum_{i\in[n]}x_i\boldsymbol{v}_i\boldsymbol{v}_i^\top\succeq\boldsymbol{X},\sum_{i\in[n]}x_i=s,\boldsymbol{x}\in\{0,1\}^n\right\}.\tag{18}$$

The key idea of deriving the convex integer program is summarized as follows: (i) Obtain Lagrangian dual of A-MESP (18) by dualizing the constraint $\sum_{i\in[n]}x_i\boldsymbol{v}_i\boldsymbol{v}_i^\top\succeq\boldsymbol{X}$; (ii) Characterize the primal formulation of the Lagrangian dual; and (iii) Enforce the continuous variables in the primal characterization to be binary. To begin with, we will introduce the following lemma, which is essential to derive the Lagrangian dual of A-MESP.

Lemma 8 *For a $d \times d$ matrix $\mathbf{\Lambda} \succeq 0$, we have*

$$\min_{\boldsymbol{X}\succeq 0}\left\{\overset{s}{\operatorname{tr}}(\boldsymbol{X}^\dagger)+\operatorname{tr}(\boldsymbol{X}\mathbf{\Lambda})\right\}=2\underset{s}{\operatorname{tr}}\left(\mathbf{\Lambda}^{\frac{1}{2}}\right). \tag{19}$$

Proof. See Appendix A.13. ☐

Next, we are going to show the Lagrangian dual of A-MESP (18), denoted by A-LD.

Theorem 8 *The Lagrangian dual of A-MESP* (17) *is*

$$\text{(A-LD)}\quad z_A^{LD}:=\max_{\mathbf{\Lambda}\succeq 0,\nu,\boldsymbol{\mu}\in\mathbb{R}_+^n}\left\{2\underset{s}{\operatorname{tr}}\left(\mathbf{\Lambda}^{\frac{1}{2}}\right)-s\nu-\sum_{i\in[n]}\mu_i:\nu+\mu_i\geq \boldsymbol{v}_i^\top\mathbf{\Lambda}\boldsymbol{v}_i, i\in[n]\right\}, \tag{20}$$

and its optimal value is a lower bound of A-MESP, i.e., $z_A^{LD}\leq z_A^*$.

Proof. By dualizing the inequality constraint of A-MESP (18), we are able to formulate the dual problem as

$$z_A^{LD}:=\max_{\mathbf{\Lambda}\succeq 0}\left\{\min_{\boldsymbol{x},\boldsymbol{X}\succeq 0}\left\{\overset{s}{\operatorname{tr}}\left(\boldsymbol{X}^\dagger\right)+\operatorname{tr}(\boldsymbol{X}\mathbf{\Lambda})-\sum_{i\in[n]}x_i\boldsymbol{v}_i^\top\mathbf{\Lambda}\boldsymbol{v}_i:\sum_{i\in[n]}x_i=s,\boldsymbol{x}\in\{0,1\}^n\right\}\right\}.$$

Applying Lemma 8 to the inner minimization problem over $\boldsymbol{X}$, the dual problem becomes

$$z_A^{LD}:=\max_{\mathbf{\Lambda}\succeq 0}\left\{\min_{\boldsymbol{x}}\left\{2\underset{s}{\operatorname{tr}}\left(\mathbf{\Lambda}^{\frac{1}{2}}\right)-\sum_{i\in[n]}x_i\boldsymbol{v}_i^\top\mathbf{\Lambda}\boldsymbol{v}_i:\sum_{i\in[n]}x_i=s,\boldsymbol{x}\in\{0,1\}^n\right\}\right\}.$$

Similarly, we derive the dual of minimization problem over $\boldsymbol{x}$ and combine the dual with the maximization over $\mathbf{\Lambda}$, which obtains A-LD problem. Apparently, $z_A^{LD}\leq z_A^*$ by weak duality. ☐

In addition, A-LD (20) has an equivalent primal characterization.

Theorem 9 *The primal characterization of A-LD* (20)*, referred to as (A-PC), is*

$$\text{(A-PC)}\quad z_A^{LD}:=\min_{\boldsymbol{x}}\left\{\Phi_s\left(\sum_{i\in[n]}x_i\boldsymbol{v}_i\boldsymbol{v}_i^\top\right):\sum_{i\in[n]}x_i=s,\boldsymbol{x}\in[0,1]^n\right\}. \tag{21}$$

Proof. See Appendix A.14. ☐

As a side product of Theorem 9, we can obtain the subdifferentials of the convex but non-smooth objective function $\Phi_s(\cdot)$ for A-PC (21).

Proposition 8 *Given a $d\times d$ matrix $\boldsymbol{X}\succeq 0$ with rank $r\geq s$, suppose the vector of eigenvalues of $\boldsymbol{X}$ is $\boldsymbol{\lambda}$ such that $\lambda_1\geq\cdots\geq\lambda_r>\lambda_{r+1}=\cdots=\lambda_d=0$ and $\boldsymbol{X}=\boldsymbol{Q}\operatorname{Diag}(\boldsymbol{\lambda})\boldsymbol{Q}^\top$ with an orthonormal matrix $\boldsymbol{Q}$. Then the subdifferential of function $\Phi_s(\cdot)$ at $\boldsymbol{X}$ that is denoted by $\partial\Phi_s(\boldsymbol{X})$ is*

$$\partial\Phi_s(\boldsymbol{X})=\Bigg\{\boldsymbol{Q}\operatorname{Diag}(\boldsymbol{\beta})\boldsymbol{Q}^\top:\boldsymbol{X}=\boldsymbol{Q}\operatorname{Diag}(\boldsymbol{\lambda})\boldsymbol{Q}^\top,\boldsymbol{Q}\text{ is orthonormal},$$

$$\boldsymbol{\beta} \in \operatorname{conv}\left\{\boldsymbol{\beta}: \beta_i = \frac{1}{\lambda_i}, \forall i \in [k], \beta_i = \frac{s-k}{\sum_{i\in[k+1,d]}\lambda_i}, \forall i \in [k+1,r], \beta_i \geq \beta_r, \forall i \in [r+1,d]\right\}\Bigg\}.$$

Note that the subdifferential of $\Phi_s(\cdot)$ above is unique and becomes the gradient when $\boldsymbol{X} \succ 0$ is non-singular.

Proof. The proof is similar to that of Proposition 2 and is thus omitted here. □

Another side product is that we obtain an equivalent convex integer program of A-MESP by enforcing the variables $\boldsymbol{x}$ in A-PC (21) to be binary.

Theorem 10 *The A-MESP is equivalent to the following convex integer program*

$$\text{(A-MESP)} \quad z_A^* := \min_{\boldsymbol{x}} \left\{ \Phi_s\left(\sum_{i\in[n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top\right) : \sum_{i\in[n]} x_i = s, \boldsymbol{x} \in \{0,1\}^n \right\}. \tag{22}$$

Proof. The proof is similar to that of Theorem 3 and is thus omitted. □

6.2. Volume Sampling Algorithm

In this subsection, we present a polynomial-time volume sampling algorithm for A-MESP, which has applied successfully to the generalized A-Optimal design (Derezinski and Warmuth 2017, Nikolov et al. 2019). A size-s subset $S \subseteq [n]$ is sampled with the probability as

$$\mathbb{P}[\tilde{S} = S] = \frac{\prod_{i\in S} \widehat{x}_i \overset{s}{\det}(\sum_{i\in S} \boldsymbol{v}_i \boldsymbol{v}_i^\top)}{\sum_{\bar{S}\in\binom{[n]}{s}} \prod_{i\in\bar{S}} \widehat{x}_i \overset{s}{\det}(\sum_{i\in\bar{S}} \boldsymbol{v}_i \boldsymbol{v}_i^\top)}.$$

Different from the sampling Algorithm 2, this probability formula, known as volume sampling (Derezinski and Warmuth 2017, Nikolov et al. 2019), delivers the proportional volume spanned by the selected vectors. Algorithm 6 describes an efficient implementation of this volume sampling algorithm, with running time complexity $O(n^5)$.

Next, we analyze the approximation ratio of the volume sampling Algorithm 6. We start with the following observation.

Lemma 9 *For any feasible solution $\boldsymbol{x}$ to A-PC (21), let $\boldsymbol{\lambda} \in \mathbb{R}_+^d$ denote the vector of eigenvalues of matrix $\sum_{i\in[n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top$, then we have*

$$\Phi_s\left(\sum_{i\in[n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top\right) \geq \frac{E_{s-1}(\boldsymbol{\lambda})}{E_s(\boldsymbol{\lambda})}, \tag{23}$$

where function $E_s(\cdot)$ is introduced in Definition 3.

Proof. See Appendix A.15. □

Algorithm 6 Efficient Implementation of Volume Sampling Procedure

Input: $n \times n$ matrix $\boldsymbol{C} \succeq 0$ of rank d and integer $s \in [d]$
Let $\widehat{\boldsymbol{x}}$ is an optimal solution of A-PC
Initialize chosen set $\tilde{S} = \emptyset$ and unchosen set $T = \emptyset$
Two factors: $A_1 = \sum_{S \in \binom{[n]}{s}} \left(\prod_{i \in S} \widehat{x}_i\right) \det\left(\boldsymbol{V}_S^\top \boldsymbol{V}_S\right), A_2 = 0$
for $j = 1, \cdots, n$ **do**
    Let $A_2 = \sum_{S \in \binom{[n]}{s}, \tilde{S} \subseteq S, T \cap S = \emptyset} \left(\prod_{i \in S} \widehat{x}_i\right) \det\left(\boldsymbol{V}_S^\top \boldsymbol{V}_S\right)$
    Sample a $(0,1)$ uniform random variable U
    if $A_2/A_1 \geq U$ **then**
        Add j to set $\tilde{S}$
        $A_1 = A_2$
    else
        Add j to set T
        $A_1 = A_1 - A_2$
    end if
end for
Output $\tilde{S}$

Observe that the right-hand side of the inequality (23) is equivalent to the relaxation bound of A-MESP proposed by Nikolov et al. (2019). Hence, Lemma 9 also indicates that our proposed bound is stronger than the existing one. The following theorem shows that we further improve the approximation ratio of the volume sampling Algorithm 6.

Theorem 11 *Given an optimal solution $\widehat{\boldsymbol{x}}$ to A-PC, the volume sampling Algorithm 6 yields a $\min(s, n-s+1)$-approximation ratio of A-MESP, i.e.,*

$$\mathbb{E}\left[\overset{s}{\operatorname{tr}}\left[\left(\sum_{i \in \tilde{S}} \boldsymbol{v}_i \boldsymbol{v}_i^\top\right)^\dagger\right]\right] \leq \min(s, n-s+1) z_A^*.$$

Proof. See Appendix A.16. □

Note that this approximation ratio improves the one stated in theorem A.3 (Nikolov et al. 2019), in particular, if $s \geq \frac{n+1}{2}$, then our approximation ratio is strictly better. Since we use the same volume sampling procedure, its deterministic implementation follows exactly from Appendix B in Nikolov et al. (2019) and is thus omitted here.

6.3. Local Search Algorithm for A-MESP

This subsection analyzes the local search algorithm to solve A-MESP, which is presented in Algorithm 7. The efficient implementation straightforwardly follows from the local search Algorithm 4 in Section 4 and is thus omitted. Therefore, we mainly focus on deriving the approximation ratio of the local search Algorithm 7.

Algorithm 7 Local Search Algorithm

```
Input: n × n matrix C ⪰ 0 of rank d and integer s ∈ [d]
Let C = V^⊤ V denote its Cholesky factorization where V ∈ ℝ^{d×n}
Let v_i ∈ ℝ^d denote the i-th column vector of matrix V for each i ∈ [n]
Initial subset Ŝ ⊆ [n] of size s such that {v_i}_{i∈Ŝ} are linearly independent
do
    for each pair (i, j) ∈ Ŝ × ([n] \ Ŝ) do
        if tr^s( Σ_{i∈Ŝ∪{j}\{i}} v_i v_i^⊤ ) < tr^s( Σ_{i∈Ŝ} v_i v_i^⊤ ) then
            Update Ŝ := Ŝ ∪ {j} \ {i}
        end if
    end for
while there is still an improvement
Output: Ŝ
```

Let us begin with the following local optimality condition for the Algorithm 7.

Lemma 10 *Suppose that $\widehat{S}$ is the output of the local search Algorithm 7 and $\boldsymbol{X} = \sum_{i\in\widehat{S}} \boldsymbol{v}_i\boldsymbol{v}_i^\top$, for each pair $(i,j) \in \widehat{S} \times ([n] \setminus \widehat{S})$, the following inequality always holds*

$$\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^3\boldsymbol{v}_i\boldsymbol{v}_j^\top(\boldsymbol{I}_n - \boldsymbol{X}^\dagger\boldsymbol{X})\boldsymbol{v}_j \le \boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_i + \boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_i\boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_j - 2\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_j\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{v}_j.$$

Proof. See Appendix A.17. □

The local optimality condition inspires us a construction of a feasible solution to A-LD (20) and thus allows the weak duality to bound the output value from the local search Algorithm 7 by the optimal value of A-MESP.

Theorem 12 *The local search Algorithm 7 yields a $s/2+\delta^{-1}\min\{\lambda_{\max}(\boldsymbol{C}), n\delta+(n-s)\lambda_{\max}(\boldsymbol{C})\}$-approximation ratio for A-MESP, i.e,*

$$\overset{s}{\operatorname{tr}}\left(\sum_{i\in\widehat{S}} \boldsymbol{v}_i\boldsymbol{v}_i^\top\right) \le \min\left\{\frac{s}{2}\left(1+\frac{\lambda_{\max}(\boldsymbol{C})}{\delta}\right), \frac{1}{2}\left(n+s+\frac{(n-s)\lambda_{\max}(\boldsymbol{C})}{\delta}\right)\right\} z_A^*,$$

where $\widehat{S}$ is the set produced by Algorithm 7, and δ is defined in Lemma 4.

Proof. See Appendix A.18. □

We remark that the result in Theorem 12 is the first-known approximation ratio of the local search algorithm for A-MESP.

Finally, Table 5 summarizes the existing and our developed approximation ratios for A-MESP.

Table 5 Summary of Approximation Algorithms for A-MESP

	Algorithm	Approximation Ratio[1]
Literature	Volume Sampling (Nikolov et al. 2019)	s
This paper	Volume Sampling Algorithm 6	$\min\{s, n-s+1\}$
	Local Search Algorithm 7	$s/2+\delta^{-1}\min\{\lambda_{\max}(\boldsymbol{C}), n\delta+(n-s)\lambda_{\max}(\boldsymbol{C})\}$

[1] Approximation Ratio denotes the ratio of the output value of the algorithm and the optimal value

7. Conclusion

This paper studies the maximum entropy sampling problem (MESP) and develops and analyzes two approximation algorithms with provable performance guarantees. Observing that the objective function of MESP is neither convex nor concave, we derive a new convex integer program for MESP through the Lagrangian dual relaxation and its primal characterization. Using the optimal solution of the primal characterization, we develop an efficient sampling algorithm and prove its approximation bound, which improves the best-known bound in literature. By developing new mathematical tools for the singular matrices and analyzing the Lagrangian dual of the proposed convex integer program, we further analyze the local search algorithm and prove its first-known approximation bound for MESP. The proof techniques that we developed inspire us an efficient implementation of the local search algorithm. Our numerical study shows that both algorithms work very well, and the local search algorithm performs the best and consistently yields near-optimal solutions. Finally, we extend all analyses to the A-Optimal MESP (A-MESP), develop a new convex integer program and study the volume sampling and local search algorithms with their approximation ratios. Our proposed algorithms are coded and released as open-source software. One possible future direction is to study MESP with general distributions.

References

Abdi H (2007) The eigen-decomposition: Eigenvalues and eigenvectors. *Encyclopedia of Measurement and Statistics* 304–308.

Alarifi A, AlZubi AA, Al-Maitah M, Al-Kasasbeh B (2019) An optimal sensor placement algorithm (o-spa) for improving tracking precision of human activity in real-world healthcare systems. *Computer Communications* 148:9–16.

Alon N, Spencer JH (2016) *The probabilistic method* (John Wiley & Sons).

Anstreicher KM (2018a) Efficient solution of maximum-entropy sampling problems .

Anstreicher KM (2018b) Maximum-entropy sampling and the boolean quadric polytope. *Journal of Global Optimization* 72(4):603–618.

Anstreicher KM, Fampa M, Lee J, Williams J (1996) Continuous relaxations for constrained maximum-entropy sampling. *International Conference on Integer Programming and Combinatorial Optimization*, 234–248 (Springer).

Anstreicher KM, Fampa M, Lee J, Williams J, et al. (1999) Using continuous nonlinear relaxations to solve. constrained maximum-entropy sampling problems. *Mathematical Programming* 85(2):221–240.

Anstreicher KM, Lee J (2004) A masked spectral bound for maximum-entropy sampling. *mODa 7—Advances in Model-Oriented Design and Analysis*, 1–12 (Springer).

Arellano-Valle RB, Contreras-Reyes JE, Genton MG (2013) Shannon entropy and mutual information for multivariate skew-elliptical distributions. *Scandinavian Journal of Statistics* 40(1):42–62.

Bellman R (1997) *Introduction to matrix analysis*, volume 19 (Siam).

Ben-Tal A, Nemirovski A (2001) *Lectures on modern convex optimization: analysis, algorithms, and engineering applications*, volume 2 (Siam).

Ben-Tal A, Nemirovski A (2012) Optimization iii: Convex analysis, nonlinear programming theory, nonlinear programming algorithms. *Lecture Notes* 34.

Bertsekas DP (1982) *Constrained optimization and Lagrange multiplier methods* (Academic press).

Broida JG, Williamson SG (1989) *A comprehensive introduction to linear algebra* (Addison-Wesley Redwood City, CA).

Bueso M, Angulo J, Alonso F (1998) A state-space model approach to optimum spatial sampling design based on entropy. *Environmental and Ecological Statistics* 5(1):29–44.

Burer S, Lee J (2007) Solving maximum-entropy sampling problems using factored masks. *Mathematical Programming* 109(2-3):263–281.

Charikar M, Guruswami V, Kumar R, Rajagopalan S, Sahai A (2000) Combinatorial feature selection problems. *Proceedings 41st Annual Symposium on Foundations of Computer Science*, 631–640 (IEEE).

Christodoulou S (2015) Smarting up water distribution networks with an entropy-based optimal sensor placement strategy. *Journal of Smart Cities* 1(1):47–58.

Çivril A, Magdon-Ismail M (2009) On selecting a maximum volume sub-matrix of a matrix and related problems. *Theoretical Computer Science* 410(47-49):4801–4811.

Civril A, Magdon-Ismail M (2013) Exponential inapproximability of selecting a maximum volume sub-matrix. *Algorithmica* 65(1):159–176.

Cover TM, Thomas JA (2012) *Elements of information theory* (John Wiley & Sons).

de Aguiar PF, Bourguignon B, Khots M, Massart D, Phan-Than-Luu R (1995) D-optimal designs. *Chemometrics and Intelligent Laboratory Systems* 30(2):199–210.

Derezinski M, Warmuth MK (2017) Unbiased estimates for linear regression via volume sampling. *Advances in Neural Information Processing Systems*, 3084–3093.

Dey SS, Mazumder R, Wang G (2018) A convex integer programming approach for optimal sparse pca. *arXiv preprint arXiv:1810.09062* .

Fan K (1949) On a theorem of weyl concerning eigenvalues of linear transformations i. *Proceedings of the National Academy of Sciences of the United States of America* 35(11):652.

Freund RM, Grigas P (2016) New analysis and results for the frank–wolfe method. *Mathematical Programming* 155(1-2):199–230.

Gilmore CJ (1996) Maximum entropy and bayesian statistics in crystallography: a review of practical applications. *Acta Crystallographica Section A: Foundations of Crystallography* 52(4):561–589.

Guruswami V, Sinop AK (2012) Optimal column-based low-rank matrix reconstruction. *Proceedings of the twenty-third annual ACM-SIAM symposium on Discrete Algorithms*, 1207–1214 (SIAM).

Hardy G, Littlewood J, Polya G (1952) Inequalities cambridge univ. *Press, Cambridge* (1988).

Harville DA (1998) Matrix algebra from a statistician's perspective.

Hazimeh H, Mazumder R (2018) Fast best subset selection: Coordinate descent and local combinatorial optimization algorithms. *arXiv preprint arXiv:1803.01454* .

Hoch JC, Maciejewski MW, Mobli M, Schuyler AD, Stern AS (2014) Nonuniform sampling and maximum entropy reconstruction in multidimensional nmr. *Accounts of Chemical Research* 47(2):708–717.

Hoffman A, Lee J, Williams J (2001) New upper bounds for maximum-entropy sampling. *mODa 6—Advances in Model-Oriented Design and Analysis*, 143–153 (Springer).

Horn A (1954) Doubly stochastic matrices and the diagonal of a rotation matrix. *American Journal of Mathematics* 76(3):620–630.

Hou SH (1998) Classroom note: A simple proof of the leverrier–faddeev characteristic polynomial algorithm. *SIAM Review* 40(3):706–709.

Hwang FK, Rothblum UG (1993) Majorization and schur convexity with respect to partial orders. *Mathematics of Operations Research* 18(4):928–944.

Jaynes ET (1957) Information theory and statistical mechanics. *Physical Review* 106(4):620.

Kelmans AK, Kimelfeld B (1983) Multiplicative submodularity of a matrix's principal minor as a function of the set of its rows and some combinatorial applications. *Discrete Mathematics* 44(1):113–116.

Ko CW, Lee J, Queyranne M (1995) An exact algorithm for maximum entropy sampling. *Operations Research* 43(4):684–691.

Lee J (1998) Constrained maximum-entropy sampling. *Operations Research* 46(5):655–664.

Lee J, Williams J (2003) A linear integer programming bound for maximum-entropy sampling. *Mathematical Programming* 94(2-3):247–256.

Lemaréchal C, Renaud A (2001) A geometric study of duality gaps, with applications. *Mathematical Programming* 90(3):399–427.

Lewis A (1995) The convex analysis of unitarily invariant matrix functions. *Journal of Convex Analysis* 2(1/2):173–183.

Li Q, Cui T, Weng Y, Negi R, Franchetti F, Ilic MD (2012) An information-theoretic approach to pmu placement in electric power systems. *IEEE Transactions on Smart Grid* 4(1):446–456.

Lin M, Trudinger NS (1994) On some inequalities for elementary symmetric functions. *Bulletin of the Australian Mathematical Society* 50(2):317–326.

Madan V, Singh M, Tantipongpipat U, Xie W (2019) Combinatorial algorithms for optimal design. *Conference on Learning Theory*, 2210–2258.

Magnus JR (1985) On differentiating eigenvalues and eigenvectors. *Econometric Theory* 1(2):179–191.

Meyer CD Jr (1973) Generalized inversion of modified matrices. *SIAM Journal on Applied Mathematics* 24(3):315–323.

Moreno-Salinas D, Pascoal A, Aranda J (2013) Sensor networks for optimal target localization with bearings-only measurements in constrained three-dimensional scenarios. *Sensors* 13(8):10386–10417.

Nikolov A (2015) Randomized rounding for the largest simplex problem. *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, 861–870 (ACM).

Nikolov A, Singh M, Tantipongpipat UT (2019) Proportional volume sampling and approximation algorithms for a-optimal design. *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, 1369–1386 (SIAM).

O'Flynn B, Regan F, Lawlor A, Wallace J, Torres J, O'Mathuna C (2010) Experiences and recommendations in deploying a real-time, water quality monitoring system. *Measurement Science and Technology* 21(12):124004.

Overton ML, Womersley RS (1995) Second derivatives for optimizing eigenvalues of symmetric matrices. *SIAM Journal on Matrix Analysis and Applications* 16(3):697–718.

Pedregosa F, Askari A, Negiar G, Jaggi M (2018) Step-size adaptivity in projection-free optimization. *arXiv preprint arXiv:1806.05123* .

Pukelsheim F (2006) *Optimal design of experiments* (SIAM).

Rigau J, Feixas M, Sbert M (2003) Entropy-based adaptive sampling. *Graphics Interface*, volume 2, 79–87.

Rockafellar RT (1970) *Convex analysis*, volume 28 (Princeton university press).

Sagnol G, Harman R, et al. (2015) Computing exact d-optimal designs by mixed integer second-order cone programming. *The Annals of Statistics* 43(5):2198–2224.

Schmieder P, Stern AS, Wagner G, Hoch JC (1993) Application of nonlinear sampling schemes to cosy-type spectra. *Journal of Biomolecular NMR* 3(5):569–576.

Sharma D, Kapoor A, Deshpande A (2015) On greedy maximization of entropy. *International Conference on Machine Learning*, 1330–1338.

Shewry MC, Wynn HP (1987) Maximum entropy sampling. *Journal of Applied Statistics* 14(2):165–170.

Singh M, Xie W (2018a) Approximate positive correlated distributions and approximation algorithms for d-optimal design. *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2240–2255 (Society for Industrial and Applied Mathematics).

Singh M, Xie W (2018b) Approximation algorithms for d-optimal design. Available at Optimization Online.

Song Y, Liò P (2010) A new approach for epileptic seizure detection: sample entropy based feature extraction and extreme learning machine. *Journal of Biomedical Science and Engineering* 3(06):556.

Summa MD, Eisenbrand F, Faenza Y, Moldenhauer C (2014) On largest volume simplices and subdeterminants. *Proceedings of the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms*, 315–323 (SIAM).

Thompson RC (1977) Singular values, diagonal elements, and convexity. *SIAM Journal on Applied Mathematics* 32(1):39–63.

Tsing NK, Fan MK, Verriest EI (1994) On analyticity of functions involving eigenvalues. *Linear Algebra and its Applications* 207:159–180.

Wang Y, Zhang L, Chen G (2019) Optimal sensor placement for obstacle detection of manipulator based on relative entropy. *2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 702–707 (IEEE).

Xu S, Doğançay K (2017) Optimal sensor placement for 3-d angle-of-arrival target localization. *IEEE Transactions on Aerospace and Electronic Systems* 53(3):1196–1211.

Yao Y, Wang H (2019) Optimal subsampling for softmax regression. *Statistical Papers* 60(2):235–249.

Zilly J, Buhmann JM, Mahapatra D (2017) Glaucoma detection using entropy sampling and ensemble learning for automatic optic cup and disc segmentation. *Computerized Medical Imaging and Graphics* 55:28–41.

Appendix A. Proofs

A.1 Proof of Proposition 1

Proposition 1 *The objective function of MESP* (2) *is (i) discrete-submodular, (ii) non-monotonic, (iii) neither concave nor convex, and (iv) not always nonnegative.*

Proof. **Part (i).** The discrete-submodularity has been proved by Kelmans and Kimelfeld (1983).

We will show the other three properties using the following example.

Example 2 *For MESP* (2)*, let* $n=d=2$, $\boldsymbol{v}_1=(\sqrt{a},0)^\top$ *and* $\boldsymbol{v}_2=(0,\sqrt{b})^\top$.

Part (ii) & Part (iv). In Example 2, when $a=2$ and $b=1/4$, we have

$$\log \overset{1}{\det}\left(\boldsymbol{v}_1\boldsymbol{v}_1^\top\right)=\log 2\geq \log \overset{2}{\det}\left(\boldsymbol{v}_1\boldsymbol{v}_1^\top+\boldsymbol{v}_2\boldsymbol{v}_2^\top\right)=\log\frac{1}{2}<0,$$

which proves that the objective function of MESP is not monotonic and is not always nonnegative.
Part (iii). In Example 2, let us consider two feasible solutions $\boldsymbol{x}^1=(1,0)^\top$ and $\boldsymbol{x}^2=(0,1)^\top$ with $s=1$. If $a=1$ and $b=1$, then we have

$$\frac{1}{2}\log \overset{1}{\det}\left(\boldsymbol{v}_1\boldsymbol{v}_1^\top\right)+\frac{1}{2}\log \overset{1}{\det}\left(\boldsymbol{v}_2\boldsymbol{v}_2^\top\right)=0\geq \log \overset{1}{\det}\left(\sum_{i\in[n]}\frac{x_i^1+x_i^2}{2}\boldsymbol{v}_i\boldsymbol{v}_i^\top\right)=\log\frac{1}{2},$$

which disproves the concavity.

If $a=16$ and $b=1$, then we have

$$\frac{1}{2}\log \overset{1}{\det}\left(\boldsymbol{v}_1\boldsymbol{v}_1^\top\right)+\frac{1}{2}\log \overset{1}{\det}\left(\boldsymbol{v}_2\boldsymbol{v}_2^\top\right)=\log 4\leq \log \overset{1}{\det}\left(\sum_{i\in[n]}\frac{x_i^1+x_i^2}{2}\boldsymbol{v}_i\boldsymbol{v}_i^\top\right)=\log 8,$$

which disproves the convexity. □

A.2 Proof of Lemma 1

Before proving Lemma 1, we will first show the following technical lemma.

Lemma 11 *Given* $\lambda_1\geq\cdots\geq\lambda_d\geq 0$ *and* $0\leq\beta_1\leq\cdots\leq\beta_d$, *we have*

(i)

$$\boldsymbol{\lambda}:=\underset{\substack{\boldsymbol{\theta}\in\mathbb{R}_+^d,\\ \theta_1\geq\cdots\geq\theta_d}}{\arg\min}\left\{\sum_{i\in[d]}\theta_i\beta_i:\sum_{i\in[t]}\theta_i\leq\sum_{i\in[t]}\lambda_i,\forall t\in[d-1],\sum_{i\in[d]}\theta_i=\sum_{i\in[d]}\lambda_i\right\},\tag{24}$$

(ii)

$$\boldsymbol{\beta}:=\underset{\substack{\boldsymbol{\theta}\in\mathbb{R}_+^d,\\ \theta_1\leq\cdots\leq\theta_d}}{\arg\min}\left\{\sum_{i\in[d]}\theta_i\lambda_i:\sum_{i\in[t+1,d]}\theta_i\leq\sum_{i\in[t+1,d]}\beta_i,\forall t\in[d-1],\sum_{i\in[d]}\theta_i=\sum_{i\in[d]}\beta_i\right\}.\tag{25}$$

Proof. To prove Part(i), it needs to show that the vector $\boldsymbol{\lambda} \in \mathbb{R}_+^d$ is an optimal solution to the minimization problem in the right-hand size of (24). We will use the induction to prove this result.

(a) When $d=1$, clearly, there is only one optimal solution, which is $\theta_1^* = \lambda_1$.

(b) Suppose that the result holds for any $d < \widehat{d}$ where $\widehat{d} \geq 1$. Now let us consider the case that $d = \widehat{d}$. Since the feasible region of the minimization problem in the right-hand size of (24) does not contain a ray, one of its optimal solutions must be an extreme point, which is denoted to be $\widehat{\theta}$. Then $\widehat{\theta}$, as an extreme point, must satisfy at least d binding constraints. There are two cases to be discussed:

- If there exists an integer $\widehat{t} \in [d-1]$ such that $\sum_{i\in[\widehat{t}]} \widehat{\theta}_i = \sum_{i\in[\widehat{t}]} \lambda_i$, then problem (24) can be lower bounded by the sum of the following two minimization problems:

$$\min_{\boldsymbol{\theta}} \left\{ \sum_{i\in[\widehat{t}]} \theta_i\beta_i : \sum_{i\in[t]} \theta_i \leq \sum_{i\in[t]} \lambda_i, \forall t \in [\widehat{t}-1], \sum_{i\in[\widehat{t}]} \theta_i = \sum_{i\in[\widehat{t}]} \lambda_i, \theta_1 \geq \cdots \geq \theta_{\widehat{t}} \right\},$$

$$\min_{\boldsymbol{\theta}} \left\{ \sum_{i\in[\widehat{t}+1,d]} \theta_i\beta_i : \sum_{i\in[\widehat{t}+1,t]} \theta_i \leq \sum_{i\in[\widehat{t}+1,t]} \lambda_i, \forall t \in [\widehat{t}+1,d], \sum_{i\in[\widehat{t}+1,d]} \theta_i = \sum_{i\in[\widehat{t}+1,d]} \lambda_i, \theta_{\widehat{t}+1} \geq \cdots \geq \theta_d \right\}.$$

According to the induction, there exists an optimal solution of each minimization problem such that $\theta_i^* = \lambda_i$ for any $i \in [d]$, which is feasible to the original problem (24) and thus is optimal.

- If there does not exist an integer $\widehat{t} \in [d-1]$ such that $\sum_{i\in[\widehat{t}]} \widehat{\theta}_i = \sum_{i\in[\widehat{t}]} \lambda_i$, then the extreme point $\widehat{\boldsymbol{\theta}}$ must satisfy $\widehat{\theta}_1 = \cdots = \widehat{\theta}_d = \frac{\sum_{i\in[d]}\lambda_i}{d}$. Given $0 \leq \beta_1 \leq \cdots \leq \beta_d$, obviously, we have

$$\sum_{i\in[d]} \lambda_i\beta_i \leq \frac{\sum_{i\in[d]}\lambda_i}{d} \sum_{i\in[d]} \beta_i.$$

Therefore, when $d = \widehat{d}$, $\boldsymbol{\theta}^* = \boldsymbol{\lambda}$ is also an optimal solution.

The proof of Part (ii) directly follows from the above if we consider $\boldsymbol{\beta} = (\lambda_d, \lambda_{d-1}, \cdots, \lambda_1)^\top$, $\boldsymbol{\lambda} = (\beta_d, \beta_{d-1}, \cdots, \beta_1)^\top$ and $\boldsymbol{\theta} = (\theta_d, \theta_{d-1}, \cdots, \theta_1)^\top$ in Part (i). $\square$

Now let us prove Lemma 1.

Lemma 1 *For a $d \times d$ matrix $\boldsymbol{\Lambda} \succ 0$, we have*

$$\max_{\boldsymbol{X}\succeq 0} \left\{ \log \overset{s}{\det}(\boldsymbol{X}) - \operatorname{tr}(\boldsymbol{X}\boldsymbol{\Lambda}) \right\} = -\log \underset{s}{\det}(\boldsymbol{\Lambda}) - s, \tag{4}$$

where function $\underset{s}{\det}(\cdot)$ is defined in Definition 1.

Proof. For any $d \times d$ matrix $\boldsymbol{X} \succeq 0$, suppose that $\boldsymbol{\lambda}$ is the vector of its eigenvalues satisfying $\lambda_1 \geq \cdots \geq \lambda_d \geq 0$, and according to the eigendecomposition (Abdi 2007), there exists an orthonormal matrix $\boldsymbol{Q}$ such that $\boldsymbol{X}=\boldsymbol{Q}\,\mathrm{Diag}(\boldsymbol{\lambda})\boldsymbol{Q}^\top$. Then the objective function in the left-hand side of (4) is equivalent to

$$\log \overset{s}{\det}(\boldsymbol{X})-\mathrm{tr}(\boldsymbol{X}\boldsymbol{\Lambda})=\log\left(\prod_{i\in[s]}\lambda_i\right)-\mathrm{tr}(\mathrm{Diag}(\boldsymbol{\lambda})\boldsymbol{Q}^\top\boldsymbol{\Lambda}\boldsymbol{Q})=\log\left(\prod_{i\in[s]}\lambda_i\right)-\sum_{i\in[d]}\theta_i\lambda_i,$$

where let $\boldsymbol{\theta}=\mathrm{diag}(\boldsymbol{Q}^\top\boldsymbol{\Lambda}\boldsymbol{Q})$. Thus, the left-hand side of (4) becomes

$$\max_{\substack{\boldsymbol{\lambda}\in\mathbb{R}_+^d,\\ \lambda_1\geq\cdots\geq\lambda_d\geq 0}}\left\{\log\left(\prod_{i\in[s]}\lambda_i\right)-\min_{\boldsymbol{Q},\boldsymbol{\theta}\in\mathbb{R}_+^d}\left\{\sum_{i\in[d]}\theta_i\lambda_i:\boldsymbol{\theta}=\mathrm{diag}(\boldsymbol{Q}^\top\boldsymbol{\Lambda}\boldsymbol{Q}),\boldsymbol{Q}\text{ is orthonormal}\right\}\right\}.$$

Since any permutation matrix is orthonormal, thus for any fixed $\lambda_1\geq\cdots\geq\lambda_d$, to maximize $-\sum_{i\in[d]}\theta_i\lambda_i$, we must have $\theta_1\leq\cdots\leq\theta_d$ based on the rearrangement inequality (Hardy et al. 1952). Thus, the left-hand side of (4) is further reduced to

$$\max_{\substack{\boldsymbol{\lambda}\in\mathbb{R}_+^d,\\ \lambda_1\geq\cdots\geq\lambda_d\geq 0}}\left\{\log\left(\prod_{i\in[s]}\lambda_i\right)-\min_{\substack{\boldsymbol{Q},\boldsymbol{\theta}\in\mathbb{R}_+^d\\ \theta_1\leq\cdots\leq\theta_d}}\left\{\sum_{i\in[d]}\theta_i\lambda_i:\boldsymbol{\theta}=\mathrm{diag}(\boldsymbol{Q}^\top\boldsymbol{\Lambda}\boldsymbol{Q}),\boldsymbol{Q}\text{ is orthonormal}\right\}\right\}. \tag{26}$$

Let $\boldsymbol{\beta}$ denote the vector of eigenvalues of $\boldsymbol{\Lambda}$ such that $\beta_1\leq\cdots\leq\beta_d$ and let $\boldsymbol{\Lambda}=\boldsymbol{P}\,\mathrm{Diag}(\boldsymbol{\beta})\boldsymbol{P}^\top$ with an orthonormal matrix $\boldsymbol{P}$. Since $\boldsymbol{Q}$ is orthonormal, the eigenvalues of $\boldsymbol{Q}^\top\boldsymbol{\Lambda}\boldsymbol{Q}$ are also equal to $\boldsymbol{\beta}$. According to the well-known majorization inequalities between eigenvalues $\boldsymbol{\beta}$ and diagonal entries $\boldsymbol{\theta}$ (see, e.g., Horn 1954, Thompson 1977), the inner minimization problem in (26) can be lower bounded by

$$\min_{\substack{\boldsymbol{\theta}\in\mathbb{R}_+^d,\\ \theta_1\leq\cdots\leq\theta_d}}\left\{\sum_{i\in[d]}\theta_i\lambda_i:\sum_{i\in[t+1,d]}\theta_i\leq\sum_{i\in[t+1,d]}\beta_i,\forall t\in[d-1],\sum_{i\in[d]}\theta_i=\sum_{i\in[d]}\beta_i\right\}$$

Applying Part (i) in Lemma 11, an optimal solution to the minimization problem is $\boldsymbol{\theta}^*=\boldsymbol{\beta}$. Thus, the optimal value of the relaxed minimization problem is $\sum_{i\in[d]}\lambda_i\beta_i$, which is achieved by letting $\boldsymbol{Q}^*=\boldsymbol{P}$ and $\boldsymbol{\theta}^*=\boldsymbol{\beta}$ for the inner optimization problem in (26) and is thus optimal.

Plugging this optimal solution into the inner maximization problem in (26), we can obtain

$$\max_{\substack{\boldsymbol{\lambda}\in\mathbb{R}_+^d,\\ \lambda_1\geq\cdots\geq\lambda_d\geq 0}}\left\{\log\left(\prod_{i\in[s]}\lambda_i\right)-\sum_{i\in[d]}\beta_i\lambda_i\right\}. \tag{27}$$

The above maximization problem can be solved by $\lambda_i^*=\frac{1}{\beta_i}$ for all $i\in[s]$ and 0 otherwise. Therefore, we have

$$\max_{\boldsymbol{X}\succeq 0}\left\{\log\overset{s}{\det}(\boldsymbol{X})-\mathrm{tr}(\boldsymbol{X}\boldsymbol{\Lambda})\right\}=-\log\det_s(\boldsymbol{\Lambda})-s.$$

This completes the proof. □

A.3 Proof of Lemma 3

Lemma 3 *Given a $d\times d$ matrix $\boldsymbol{X}\succeq 0$ with rank $r\in[s,d]$, suppose that the eigenvalues of $\boldsymbol{X}$ are $\lambda_1\geq\cdots\geq\lambda_r>\lambda_{r+1}=\cdots=\lambda_d=0$ and $\boldsymbol{X}=\boldsymbol{Q}\operatorname{Diag}(\boldsymbol{\lambda})\boldsymbol{Q}^\top$ with an orthonormal matrix $\boldsymbol{Q}$. Then*

(i)

$$\min_{\boldsymbol{\Lambda}\succ 0}\left\{-\log\operatorname*{det}_s(\boldsymbol{\Lambda})+\operatorname{tr}(\boldsymbol{X}\boldsymbol{\Lambda})\right\}=\min_{\substack{\boldsymbol{\beta}\in\mathbb{R}_+^d,\\ 0<\beta_1\leq\cdots\leq\beta_d}}\left\{-\sum_{i\in[s]}\log(\beta_i)+\sum_{i\in[d]}\lambda_i\beta_i\right\},\tag{7}$$

(ii)

$$\min_{\substack{\boldsymbol{\beta}\in\mathbb{R}_+^d,\\ 0<\beta_1\leq\cdots\leq\beta_d}}\left\{-\sum_{i\in[s]}\log(\beta_i)+\sum_{i\in[d]}\lambda_i\beta_i\right\}=\Gamma_s(\boldsymbol{X})+s.\tag{8}$$

Proof. **Part (i).** Suppose $\boldsymbol{\Lambda}$ has eigenvalues $0<\beta_1\leq\cdots\leq\beta_d$ and $\boldsymbol{\Lambda}=\boldsymbol{P}\operatorname{Diag}(\boldsymbol{\beta})\boldsymbol{P}^\top$ with an orthonormal matrix $\boldsymbol{P}$. Then the objective function in the left-hand side of (7) is equal to

$$-\log\operatorname*{det}_s(\boldsymbol{\Lambda})+\operatorname{tr}(\boldsymbol{X}\boldsymbol{\Lambda})=-\log\left(\prod_{i\in[s]}\beta_i\right)+\operatorname{tr}\left(\boldsymbol{P}^\top\boldsymbol{X}\boldsymbol{P}\operatorname{Diag}(\boldsymbol{\beta})\right)=-\log\left(\prod_{i\in[s]}\beta_i\right)+\sum_{i\in[d]}\theta_i\beta_i,$$

where $\boldsymbol{\theta}=\operatorname{diag}(\boldsymbol{P}^\top\boldsymbol{X}\boldsymbol{P})$.

For any fixed $\beta_1\leq\cdots\leq\beta_d$, according to the rearrangement inequality (Hardy et al. 1952), to minimize $\sum_{i\in[d]}\theta_i\beta_i$, we must have $\theta_1\geq\cdots\geq\theta_d$. Thus, the left-hand side of (7) becomes

$$\min_{\substack{\boldsymbol{\beta}\in\mathbb{R}_+^d,\\ 0<\beta_1\leq\cdots\leq\beta_d}}\left\{-\log\left(\prod_{i\in[s]}\beta_i\right)+\min_{\substack{\boldsymbol{P},\boldsymbol{\theta}\in\mathbb{R}_+^d\\ \theta_1\geq\cdots\geq\theta_d}}\left\{\sum_{i\in[d]}\theta_i\beta_i:\boldsymbol{\theta}=\operatorname{diag}(\boldsymbol{P}^\top\boldsymbol{X}\boldsymbol{P}),\boldsymbol{P}\text{ is orthonormal}\right\}\right\}.\tag{28}$$

As $\boldsymbol{P}$ is orthonormal, thus the eigenvalues of $\boldsymbol{P}^\top\boldsymbol{X}\boldsymbol{P}$ are also equal to $\boldsymbol{\lambda}$. Then the inner minimization problem in (28) can be lower bounded by

$$\min_{\boldsymbol{\theta}}\left\{\sum_{i\in[d]}\theta_i\beta_i:\sum_{i\in[t]}\theta_i\leq\sum_{i\in[t]}\lambda_i,\forall t\in[d-1],\sum_{i\in[d]}\theta_i=\sum_{i\in[d]}\lambda_i,\theta_1\geq\cdots\geq\theta_d\right\}.$$

According to Part (ii) in Lemma 11, the optimal value of the inner minimization problem in (28) is $\sum_{i\in[d]}\lambda_i\beta_i$, which is achieved by letting $\boldsymbol{P}^*=\boldsymbol{Q}$ and $\boldsymbol{\theta}^*=\boldsymbol{\lambda}$. This proves the identity (7).

Part (ii). Let us introduce an additional variable τ to differentiate the first s smallest $\boldsymbol{\beta}$ elements and simplify the order constraint in the left-hand problem (8) as

$$\min_{\boldsymbol{\beta}\in\mathbb{R}_+^d,\tau}\left\{-\sum_{i\in[s]}\log(\beta_i)+\sum_{i\in[d]}\lambda_i\beta_i:\beta_i\leq\tau,\forall i\in[s],\beta_i\geq\tau,\forall i\in[s+1,d]\right\}.\tag{29}$$

Let $\boldsymbol{\mu}\in\mathbb{R}^d$ denote the Lagrangian multipliers and the Lagrangian function is

$$L(\boldsymbol{\mu},\boldsymbol{\beta},\tau)=-\sum_{i\in[s]}\log(\beta_i)+\sum_{i\in[d]}\lambda_i\beta_i+\sum_{i\in[s]}\mu_i(\beta_i-\tau)+\sum_{i\in[s+1,d]}\mu_i(\tau-\beta_i).$$

Clearly, as the constraints in the convex program (29) are linear, the relaxed Slater condition holds. Let $(\boldsymbol{\mu}^*, \boldsymbol{\beta}^*, \tau^*)$ denote the pair of optimal primal and dual solutions. Then the KKT conditions of the convex program (29) are

$$\begin{aligned}
&\frac{\partial L}{\partial \beta_i}(\boldsymbol{\mu}^*, \boldsymbol{\beta}^*, \tau^*) = -\frac{1}{\beta_i^*} + \lambda_i + \mu_i^* = 0, \forall i \in [s], \frac{\partial L}{\partial \beta_i}(\boldsymbol{\mu}^*, \boldsymbol{\beta}^*, \tau^*) = \lambda_i - \mu_i^* = 0, \forall i \in [s+1, d],\\
&\frac{\partial L}{\partial \tau}(\boldsymbol{\mu}^*, \boldsymbol{\beta}^*, \tau^*) = \sum_{i\in[s]} \mu_i^* - \sum_{i\in[s+1,d]} \mu_i^* = 0, \mu_i^*(\beta_i^* - \tau^*) = 0, \forall i \in [s], \mu_i^*(\tau^* - \beta_i^*) = 0, \forall i \in [s+1, d],\\
&\beta_i^* \le \tau^*, \forall i \in [s], \beta_i^* \ge \tau^*, \forall i \in [s+1, d], \mu_i^* \ge 0, \forall i \in [d],
\end{aligned}$$

which are necessary and sufficient optimality conditions (see theorem 3.2.4 in Ben-Tal and Nemirovski 2012). Recall that matrix $\boldsymbol{X}$ has rank r and its eigenvalues are sorted such that $\lambda_1 \ge \cdots \ge \lambda_s \ge \cdots \ge \lambda_r > \lambda_{r+1} = \cdots = \lambda_d = 0$. Additionally, according to the KKT conditions, the optimal solution $\{\beta_i\}_{i\in[s]}$ must be sorted in an ascending order, i.e., $\beta_1 \le \cdots \le \beta_s$. Thus, let integer $k \in [0, s]$ denote the largest index such that $\beta_i^* < \tau^*$ (by convention, we let $\beta_0^* = 0, \lambda_0 = \infty$). Then the above KKT conditions can be simplified as

$$\begin{aligned}
&\beta_i^* = \frac{1}{\lambda_i}, \mu_i^* = 0, \forall i \in [k]; \beta_i^* = \tau^*, \mu_i^* = \frac{1}{\tau^*} - \lambda_i \ge 0, \forall i \in [k+1, s];\\
&\mu_i^* = \lambda_i > 0, \beta_i^* = \tau^*, \forall i \in [s+1, r]; \mu_i^* = \lambda_i = 0, \beta_i^* \ge \tau^*, \forall i \in [r+1, d];\\
&\sum_{i\in[s]} \mu_i^* - \sum_{i\in[s+1,d]} \mu_i^* = 0.
\end{aligned}$$

This implies that all pairs of the optimal primal and dual solutions are characterized by the following set

$$\Omega = \left\{ (\boldsymbol{\mu}, \boldsymbol{\beta}, \tau) : \tau = \frac{s-k}{\sum_{i\in[k+1,d]} \lambda_i}, \beta_i = \frac{1}{\lambda_i}, \forall i \in [k], \beta_i = \tau, \forall i = [k+1, r], \beta_i \ge \beta_r, \forall i \in [r+1, d], \right.$$
$$\left. \mu_i = 0, \forall i = [k], \mu_i = \frac{1}{\tau} - \lambda_i, \forall i = [k+1, r], \mu_i = 0, \forall i = [r+1, d] \right\}.$$

Consequently, any optimal solution for problem (29) satisfies

$$\beta_i^* = \frac{1}{\lambda_i}, \forall i \in [k], \beta_i^* = \frac{s-k}{\sum_{i\in[k+1,d]} \lambda_i}, \forall i \in [k+1, r], \beta_i^* \ge \frac{s-k}{\sum_{i\in[k+1,d]} \lambda_i}, \forall i \in [r+1, d],$$

which is feasible to the minimization problem in (8) and thus is optimal.

Then the optimal value of the minimization problem in (8) is equal to

$$-\sum_{i\in[s]} \log(\beta_i^*) + \sum_{i\in[d]} \lambda_i \beta_i^* = \sum_{i\in[k]} \log(\lambda_i) + (s-k) \log\left(\frac{\sum_{i\in[k+1,d]} \lambda_i}{s-k} \right) + s = \Gamma_s(\boldsymbol{X}) + s,$$

where the second equality is due to Definition 2 of $\Gamma_s(\boldsymbol{X})$. This completes the proof. □

A.4 Proof of Theorem 3

Theorem 3 *MESP can be formulated as the following convex integer program*

$$\text{(MESP)} \quad z^* := \max_{\boldsymbol{x}} \left\{ \Gamma_s \left(\sum_{i \in [n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top \right) : \sum_{i \in [n]} x_i = s, \boldsymbol{x} \in \{0,1\}^n \right\}. \tag{11}$$

Proof. It is sufficient to prove that for any feasible solution $\boldsymbol{x}$ to MESP (11), we must have

$$\log \overset{s}{\det} \left(\sum_{i \in [n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top \right) = \Gamma_s \left(\sum_{i \in [n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top \right).$$

Given a solution $\boldsymbol{x}$, we let $\boldsymbol{X} = \sum_{i \in [n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top$ with rank r and let $\boldsymbol{\lambda}$ denote its eigenvalues such that $\lambda_1 \geq \cdots \geq \lambda_d \geq 0$. Since the rank of matrix $\boldsymbol{X}$ satisfies $r \leq s$, there are two cases to be discussed regarding whether $r = s$ holds or not.

(i) If $r < s$, then clearly, we have $\log \overset{s}{\det} (\boldsymbol{X}) = -\infty$. On the other hand, by the choice of k in Lemma 2, it is evident that $k = r$ such that $\frac{1}{s-k} \sum_{i \in [k+1,d]} \lambda_i = 0$. It follows that $\Gamma_s(\boldsymbol{X}) = -\infty = \log \overset{s}{\det} (\boldsymbol{X})$.

(ii) If $r = s$, there must exist an integer ℓ such that $\lambda_1 \geq \cdots \geq \lambda_\ell > \lambda_{\ell+1} = \cdots = \lambda_s > \lambda_{s+1} = \cdots = \lambda_d = 0$. By the uniqueness of k, we must have $k = \ell$. Thus, from Definition 2, the objective value is equal to

$$\Gamma_s(\boldsymbol{X}) = \log \left(\prod_{i \in [k]} \lambda_i \right) + (s-k) \log \left(\frac{1}{s-k} \sum_{i \in [k+1,d]} \lambda_i \right) = \log \left(\prod_{i \in [s]} \lambda_i \right) = \log \overset{s}{\det} (\boldsymbol{X}).$$

□

A.5 Proof of Proposition 3

Proposition 3 *The optimal value of PC* (9) *is equal to* z^*, *i.e.,* $z^{LD} = z^*$ *provided the following three special cases: (i)* $\boldsymbol{C}$ *is diagonal; (ii)* $s = 1$*; and (iii)* $s = n$.

Proof. We will show the three special cases separately.

(i) Suppose that $\boldsymbol{C}$ is diagonal. Without loss of generality, assume that $\boldsymbol{C} = \text{Diag}(\boldsymbol{\lambda})$ with a nonnegative vector $\boldsymbol{\lambda}$ such that $\lambda_1 \geq \cdots \geq \lambda_d > \lambda_{d+1} = \cdots = \lambda_n = 0$, then we have $\boldsymbol{v}_i = \sqrt{\lambda_i} \boldsymbol{e}_i$ for each $i \in [n]$ and $\boldsymbol{C} = \boldsymbol{V}^\top \boldsymbol{V}$. Clearly, the optimal solution of MESP (2) is $x_i^* = 1$ for each $i \in [s]$ and 0 otherwise. Thus, $z^* = \log \overset{s}{\det} \left(\prod_{i \in [n]} x_i^* \boldsymbol{v}_i \boldsymbol{v}_i^\top \right) = \log \left(\prod_{i \in [s]} \lambda_i \right)$.

Let $\boldsymbol{X} = \sum_{i \in [n]} x_i^* \boldsymbol{v}_i \boldsymbol{v}_i^\top$, then we construct the feasible solution to LD (5) as

$$\boldsymbol{\Lambda}^* = \frac{1}{\lambda_s} (\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X}) + \boldsymbol{X}^\dagger, \nu^* = 1, \mu_i^* = 0, \forall i \in [n].$$

It is easy to see that $(\boldsymbol{\Lambda}^*, \nu^*, \boldsymbol{\mu}^*)$ is feasible to LD (5) with the objective value

$$z^{LD} \leq -\log \underset{s}{\det}(\boldsymbol{\Lambda}^*) + sv^* + \sum_{i \in [n]} \mu_i^* - s = \sum_{i \in [s]} \log(\lambda_i) = z^* \leq z^{LD},$$

where the first inequality is by feasibility of $(\boldsymbol{\Lambda}^*, \nu^*, \boldsymbol{\mu}^*)$ and the second one is from the weak duality.

(ii) Suppose that $s=1$. Given any feasible solution $\boldsymbol{x}$ to PC (9), assume that matrix $\boldsymbol{X} = \sum_{i\in[n]} x_i \boldsymbol{v}_i\boldsymbol{v}_i^\top$ has the eigenvalue vector $\boldsymbol{\lambda}$ such that $\lambda_1 \geq \cdots \geq \lambda_d$. By Lemma 2, as $k < s$, we must have $k=0$. Thus, the objective value of PC (9) becomes

$$\Gamma_s(\boldsymbol{X}) = (s-k)\log\left(\frac{1}{s-k}\sum_{i\in[k+1,d]}\lambda_i\right) = \log\left(\sum_{i\in[d]}\lambda_i\right) = \log\left(\sum_{i\in[n]} x_i \boldsymbol{v}_i^\top \boldsymbol{v}_i\right).$$

Therefore, in this case, we have

$$z^{LD} = \max_{\boldsymbol{x}}\left\{\log\left(\sum_{i\in[n]} x_i \boldsymbol{v}_i^\top \boldsymbol{v}_i\right) : \sum_{i\in[n]} x_i = 1, \boldsymbol{x}\in[0,1]^n\right\} = \max_{i\in[n]}\left\{\log(\boldsymbol{v}_i^\top \boldsymbol{v}_i)\right\} = z^*.$$

(iii) Suppose that $s=n$. In this case, the only feasible solution of PC (9) or MESP (11) is $x_i = 1$ for each $i\in[n]$ and clearly, PC (9) and MESP (11) are equivalent. ☐

A.6 Proof of Lemma 4

Lemma 4 *Suppose that for any size-s subset $S\subseteq[n]$, the columns $\{\boldsymbol{v}_i\}_{i\in S}$ are linearly independent. Let $\mathbb{D} := \{\boldsymbol{x}\in\mathbb{R}^n : \sum_{i\in[n]} x_i = s, \boldsymbol{x}\in[0,1]^n\}$. Then for any $\boldsymbol{x}\in$ relint$(\mathbb{D})$, we have*

$$\nabla^2\Gamma_s\left(\sum_{i\in[n]} x_i\boldsymbol{v}_i\boldsymbol{v}_i^\top\right) \succeq -\frac{\lambda_{\max}^2(\boldsymbol{C})}{\delta^2}\boldsymbol{I}_n, \tag{12}$$

where the constant $\delta := \min_{S\subseteq[n],|S|=s}\lambda_{\min}(\boldsymbol{C}_{S,S})$.

Proof. We split the proof into four steps.

Step (i)- An Equivalent Statement. For any $\boldsymbol{x},\boldsymbol{y}\in$ relint$(\mathbb{D})$, let $\boldsymbol{X} = \sum_{i\in[n]} x_i\boldsymbol{v}_i\boldsymbol{v}_i^\top$ and $\boldsymbol{Y} = \sum_{i\in[n]} y_i\boldsymbol{v}_i\boldsymbol{v}_i^\top$, clearly, matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$ are positive-definite and non-singular. Let us define a function $h(t) = \Gamma_s(\boldsymbol{X} + t(\boldsymbol{Y}-\boldsymbol{X}))$ with $t\in[0,\epsilon]$ for some sufficiently small positive number ϵ. Let $\boldsymbol{\lambda}\in\mathbb{R}^d_{++}$ denote the vector of eigenvalues of $\boldsymbol{X}$ and $\lambda_1 \geq \cdots \lambda_d > 0$. Since

$$\Gamma_s(\boldsymbol{X}) = F(\boldsymbol{\lambda}) := \log\left(\prod_{i\in[k]}\lambda_i\right) + (s-k)\log\left(\frac{1}{s-k}\sum_{i\in[k+1,d]}\lambda_i\right),$$

and $F(\boldsymbol{\lambda})$ is symmetric and analytic at $\mathbb{R}^d_{++}$, thus according to theorem 2.1 in Tsing et al. (1994), $\Gamma_s(\boldsymbol{X})$ is analytic and is thus continuous differentiable. Since the positive-definite matrices with distinct eigenvalues are dense in the space of all the positive-definite matrices, without loss of generality, we can assume that $\boldsymbol{X}$ has eigenvalues $\lambda_1 > \cdots > \lambda_d > 0$ and their corresponding eigenvectors are $\boldsymbol{q}_1,\cdots,\boldsymbol{q}_d$. Suppose that the eigenvalues and their corresponding eigenvectors of $\boldsymbol{X}+t(\boldsymbol{Y}-\boldsymbol{X})$ are $\lambda_1(t),\cdots,\lambda_d(t)$ and $\boldsymbol{q}_1(t),\cdots,\boldsymbol{q}_d(t)$. As ϵ is sufficiently small, thus, we still have $\lambda_1(t) > \cdots > \lambda_d(t)$ and according to Lemma 2, $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}(t)$ share the same integer k for all $t\in[0,\epsilon]$. Since all the eigenvalues are distinct, the eigenvalues $\{\lambda_i(t)\}_{i\in[d]}$ and eigenvectors $\{\boldsymbol{q}_i(t)\}_{i\in[d]}$ are continuous in the range of $[0,\epsilon]$ (see, e.g., Magnus 1985, Overton and Womersley 1995).

As stated in Proposition 2, function $\Gamma_s(\widehat{\boldsymbol{X}})$ is differentiable if matrix $\widehat{\boldsymbol{X}}$ is positive-definite. Thus, for any $t\in(0,\epsilon)$, we have

$$h'(t)=\frac{\mathrm{d}}{\mathrm{d}t}h(t)=\langle\nabla\Gamma_s(\boldsymbol{X}+t(\boldsymbol{Y}-\boldsymbol{X})),\boldsymbol{Y}-\boldsymbol{X}\rangle,$$

which implies that

$$h''(0)=\frac{\mathrm{d}^2}{\mathrm{d}t^2}h(t)\Big|_{t=0}=\left\langle\frac{\mathrm{d}}{\mathrm{d}t}\nabla\Gamma_s(\boldsymbol{X}+t(\boldsymbol{Y}-\boldsymbol{X}))\Big|_{t=0},\boldsymbol{Y}-\boldsymbol{X}\right\rangle.$$

Therefore, to prove the inequality (12), it is sufficient to show that

$$h''(0)\geq-\frac{\lambda_{\max}^2(\boldsymbol{C})}{\delta^2}\|\boldsymbol{x}-\boldsymbol{y}\|_2^2. \tag{30}$$

Step (ii)- A Representation of $h''(0)$.

By Proposition 2, we have

$$\nabla\Gamma_s(\boldsymbol{X}+t(\boldsymbol{Y}-\boldsymbol{X}))=\sum_{i\in[k]}\frac{1}{\lambda_i(t)}\boldsymbol{q}_i(t)\boldsymbol{q}_i(t)^\top+\sum_{i\in[k+1,d]}\frac{s-k}{\sum_{j\in[k+1,d]}\lambda_j(t)}\boldsymbol{q}_i(t)\boldsymbol{q}_i(t)^\top.$$

For the notational convenience, let us define a vector $\boldsymbol{\beta}\in\mathbb{R}_+^d$ such that

$$\beta_i=\lambda_i,\forall i\in[k],\beta_i=\frac{1}{s-k}\sum_{j\in[k+1,d]}\lambda_j,\forall i\in[k+1,d].$$

Taking the derivative of eigenvalues and eigenvectors over t separately, we obtain

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\nabla\Gamma_s(\boldsymbol{X}+t(\boldsymbol{Y}-\boldsymbol{X}))\Big|_{t=0}=&\underbrace{-\sum_{i\in[k]}\frac{1}{\beta_i^2}\frac{d\lambda_i(t)}{dt}\Big|_{t=0}\boldsymbol{q}_i\boldsymbol{q}_i^\top-\sum_{i\in[k+1,d]}\frac{1}{(s-k)\beta_i^2}\frac{d\lambda_i(t)}{dt}\Big|_{t=0}\boldsymbol{q}_i\boldsymbol{q}_i^\top}_{:=A}\\&+\underbrace{\sum_{i\in[d]}\frac{1}{\beta_i}\frac{d\boldsymbol{q}_i(t)}{dt}\Big|_{t=0}\boldsymbol{q}_i^\top+\sum_{i\in[d]}\frac{1}{\beta_i}\boldsymbol{q}_i\Big(\frac{d\boldsymbol{q}_i(t)}{dt}\Big|_{t=0}\Big)^\top}_{:=B}.\end{aligned}$$

It follows that

$$h''(0)=\big\langle A,\boldsymbol{Y}-\boldsymbol{X}\big\rangle+\big\langle B,\boldsymbol{Y}-\boldsymbol{X}\big\rangle. \tag{31}$$

Thus, to prove (30), we need to find lower bounds of $\langle A,\boldsymbol{Y}-\boldsymbol{X}\rangle$ and $\langle B,\boldsymbol{Y}-\boldsymbol{X}\rangle$ separately.

Step (iii)- Lower Bounds of $\langle A,\boldsymbol{Y}-\boldsymbol{X}\rangle$ and $\langle B,\boldsymbol{Y}-\boldsymbol{X}\rangle$.

Before we proceed, let us first prove the following claim.

Claim 1 *For any $\ell \in [s-1]$, we have*

$$\min_{\boldsymbol{u}\in\mathbb{D}}\left\{\frac{1}{s-\ell}\sum_{i\in[\ell+1,d]}\lambda_i(\boldsymbol{U}):\boldsymbol{U}=\sum_{i\in[n]}u_i\boldsymbol{v}_i\boldsymbol{v}_i^\top\right\}\geq\min_{S\in[n],|S|=s}\lambda_s\Big(\sum_{j\in S}\boldsymbol{v}_j\boldsymbol{v}_j^\top\Big):=\delta,$$

where for a symmetric matrix $\boldsymbol{X}$, we let $\lambda_i(\boldsymbol{X})$ denotes its i-th largest eigenvalue.

Proof. For a $d\times d$ positive-semidefinite matrix $\boldsymbol{U}$, the function $\sum_{i\in[\ell+1,d]}\lambda_i(\boldsymbol{U})$ is concave (Fan 1949). On the other hand, it is known that for the concave minimization problem, the optimum can be achieved by one of the extreme points of the feasible region. Thus,

$$\begin{aligned}\inf_{\boldsymbol{u}\in\mathbb{D}}\left\{\frac{1}{s-\ell}\sum_{i\in[\ell+1,d]}\lambda_i(\boldsymbol{U}):\boldsymbol{U}=\sum_{i\in[n]}u_i\boldsymbol{v}_i\boldsymbol{v}_i^\top\right\}&=\frac{1}{s-\ell}\min_{S\in[n],|S|=s}\sum_{i\in[\ell+1,d]}\lambda_i\Big(\sum_{j\in S}\boldsymbol{v}_j\boldsymbol{v}_j^\top\Big)\\&=\frac{1}{s-\ell}\min_{S\in[n],|S|=s}\sum_{i\in[\ell+1,s]}\lambda_i\Big(\sum_{j\in S}\boldsymbol{v}_j\boldsymbol{v}_j^\top\Big)\\&\geq\min_{S\in[n],|S|=s}\lambda_s\Big(\sum_{j\in S}\boldsymbol{v}_j\boldsymbol{v}_j^\top\Big),\end{aligned}$$

where the second equation is due to the fact that rank of $\sum_{j\in S}\boldsymbol{v}_j\boldsymbol{v}_j^\top$ is equal to s, and the first inequality is because $\lambda_s\Big(\sum_{j\in S}\boldsymbol{v}_j\boldsymbol{v}_j^\top\Big)$ is the smallest positive eigenvalues of matrix $\sum_{j\in S}\boldsymbol{v}_j\boldsymbol{v}_j^\top$. ⋄

Now we are ready to show the lower bounds of $\langle A,\boldsymbol{Y}-\boldsymbol{X}\rangle$ and $\langle B,\boldsymbol{Y}-\boldsymbol{X}\rangle$.

(a) According to Overton and Womersley (1995), we have

$$\frac{d\lambda_i(t)}{dt}\Big|_{t=0}=\boldsymbol{q}_i^\top\frac{d(\boldsymbol{X}+t(\boldsymbol{Y}-\boldsymbol{X}))}{dt}\Big|_{t=0}\boldsymbol{q}_i=\boldsymbol{q}_i^\top(\boldsymbol{Y}-\boldsymbol{X})\boldsymbol{q}_i.$$

Therefore, $\langle A,\boldsymbol{Y}-\boldsymbol{X}\rangle$ is equivalent to

$$\begin{aligned}\langle A,\boldsymbol{Y}-\boldsymbol{X}\rangle&=-\sum_{i\in[k]}\frac{1}{\beta_i^2}\left(\boldsymbol{q}_i^\top(\boldsymbol{Y}-\boldsymbol{X})\boldsymbol{q}_i\right)^2-\frac{1}{(s-k)\,\beta_i^2}\sum_{i\in[k+1,d]}\left(\boldsymbol{q}_i^\top(\boldsymbol{Y}-\boldsymbol{X})\boldsymbol{q}_i\right)^2\\&\geq-\frac{(s-k)^2}{(\sum_{j\in[k+1,d]}\lambda_j)^2}\sum_{i\in[k]}\left(\boldsymbol{q}_i^\top(\boldsymbol{Y}-\boldsymbol{X})\boldsymbol{q}_i\right)^2-\frac{s-k}{(\sum_{j\in[k+1,d]}\lambda_j)^2}\sum_{i\in[k+1,d]}\left(\boldsymbol{q}_i^\top(\boldsymbol{Y}-\boldsymbol{X})\boldsymbol{q}_i\right)^2\\&\geq-\frac{1}{\delta^2}\sum_{i\in[d]}\left(\boldsymbol{q}_i^\top(\boldsymbol{Y}-\boldsymbol{X})\boldsymbol{q}_i\right)^2,\end{aligned}\tag{32}$$

where the first inequality is due to the fact that $\lambda_1\geq\cdots\lambda_k>\frac{\sum_{j\in[k+1,d]}\lambda_j}{s-k}$, the second inequality is because of Claim 1, and $s-k\geq 1$.

(b) According to the result from Magnus (1985) that $\frac{d\boldsymbol{q}_i(t)}{dt}\Big|_{t=0}=(\lambda_i\boldsymbol{I}_d-\boldsymbol{X})^\dagger\frac{d(\boldsymbol{X}+t(\boldsymbol{Y}-\boldsymbol{X}))}{dt}\Big|_{t=0}\boldsymbol{q}_i=(\lambda_i\boldsymbol{I}_d-\boldsymbol{X})^\dagger(\boldsymbol{Y}-\boldsymbol{X})\boldsymbol{q}_i$, where

$$(\lambda_i\boldsymbol{I}_d-\boldsymbol{X})^\dagger=\sum_{j\in[d],j\neq i}\frac{1}{\lambda_i-\lambda_j}\boldsymbol{q}_j\boldsymbol{q}_j^\top.$$

Thus, $\langle B, \boldsymbol{Y}-\boldsymbol{X}\rangle$ is equivalent to

$$\begin{aligned}\langle B, \boldsymbol{Y}-\boldsymbol{X}\rangle &= \sum_{i\in[d]}\frac{1}{\beta_i}\sum_{j\in[d],j\neq i}\frac{1}{\lambda_i-\lambda_j}\left(\boldsymbol{q}_j^\top(\boldsymbol{Y}-\boldsymbol{X})\boldsymbol{q}_i\right)^2+\sum_{j\in[d]}\frac{1}{\beta_j}\sum_{i\in[d],i\neq j}\frac{1}{\lambda_j-\lambda_i}\left(\boldsymbol{q}_j^\top(\boldsymbol{Y}-\boldsymbol{X})\boldsymbol{q}_i\right)^2\\ &= \sum_{i\in[d]}\sum_{j\in[d],j\neq i}\left(\frac{1}{\beta_i}\frac{1}{\lambda_i-\lambda_j}+\frac{1}{\beta_j}\frac{1}{\lambda_j-\lambda_i}\right)\left(\boldsymbol{q}_j^\top(\boldsymbol{Y}-\boldsymbol{X})\boldsymbol{q}_i\right)^2. \quad (33)\end{aligned}$$

Above, we can split the summations in the right-hand side of (33) into four cases and also by plugging the values of $\boldsymbol{\beta}$, we can rewrite $\langle B, \boldsymbol{Y}-\boldsymbol{X}\rangle$ as

$$\begin{aligned}\langle B, \boldsymbol{Y}-\boldsymbol{X}\rangle =& \sum_{i\in[k]}\sum_{j\in[k],j\neq i}\frac{1}{\lambda_i\lambda_j}\left(\boldsymbol{q}_j^\top(\boldsymbol{Y}-\boldsymbol{X})\boldsymbol{q}_i\right)^2+\sum_{i\in[k+1,d]}\sum_{j\in[k+1,d],j\neq i}0\\ &+\sum_{i\in[k]}\sum_{j\in[k+1,d],j\neq i}\left(\frac{1}{\lambda_i}\frac{1}{\lambda_i-\lambda_j}+\frac{s-k}{\sum_{\ell\in[k+1,d]}\lambda_\ell}\frac{1}{\lambda_j-\lambda_i}\right)\left(\boldsymbol{q}_j^\top(\boldsymbol{Y}-\boldsymbol{X})\boldsymbol{q}_i\right)^2\\ &+\sum_{i\in[k+1,d]}\sum_{j\in[k],j\neq i}\left(\frac{s-k}{\sum_{\ell\in[k+1,d]}\lambda_\ell}\frac{1}{\lambda_i-\lambda_j}+\frac{1}{\lambda_j}\frac{1}{\lambda_j-\lambda_i}\right)\left(\boldsymbol{q}_j^\top(\boldsymbol{Y}-\boldsymbol{X})\boldsymbol{q}_i\right)^2\\ \geq& -\frac{1}{\delta^2}\sum_{i\in[d]}\sum_{j\in[d],j\neq i}\left(\boldsymbol{q}_j^\top(\boldsymbol{Y}-\boldsymbol{X})\boldsymbol{q}_i\right)^2, \quad (34)\end{aligned}$$

where the inequality is because $\lambda_i > \frac{\sum_{\ell\in[k+1,d]}\lambda_\ell}{s-k}\geq\lambda_j$ for each pair $(i,j)\in[k]\times[k+1,d]$, and $\frac{\sum_{\ell\in[k+1,d]}\lambda_\ell}{s-k}\geq\delta$ by Claim 1.

Step (iv)- Combining All the Pieces Together. According to the results (31), (32), and (34), we can derive that

$$\begin{aligned}h''(0) &\geq -\frac{1}{\delta^2}\sum_{i\in[d]}\sum_{j\in[d]}\left(\boldsymbol{q}_j^\top(\boldsymbol{Y}-\boldsymbol{X})\boldsymbol{q}_i\right)^2 = -\frac{1}{\delta^2}\operatorname{tr}(((\boldsymbol{Y}-\boldsymbol{X})\boldsymbol{Q})^2)\\ &\geq -\frac{1}{\delta^2}||\boldsymbol{Y}-\boldsymbol{X}||_2^2\\ &\geq -\frac{1}{\delta^2}\lambda_{\max}^2(\boldsymbol{C})||\boldsymbol{y}-\boldsymbol{x}||_2^2,\end{aligned}$$

where the second inequality is due to Cauchy-Schwartz inequality and that matrix $\boldsymbol{Q}$ is orthonormal, and the third inequality stems from the fact that $||\boldsymbol{Y}-\boldsymbol{X}||_2^2 = \|\boldsymbol{V}\operatorname{Diag}(\boldsymbol{y}-\boldsymbol{x})\boldsymbol{V}^\top\|_2^2 \leq \lambda_{\max}^2(\boldsymbol{C})||\boldsymbol{y}-\boldsymbol{x}||_2^2$. □

A.7 Proof of Lemma 6

Lemma 6 *Consider a size-τ subset $\widehat{S}\subseteq[n]$ with $\tau\in[d]$ such that $\{\boldsymbol{v}_i\}_{i\in\widehat{S}}$ are linearly independent. Let $\boldsymbol{X}=\sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top$, and for each $i\in\widehat{S}$, let $\boldsymbol{X}_{-i}=\boldsymbol{X}-\boldsymbol{v}_i\boldsymbol{v}_i^\top$. Then for each $(i,j)\in\widehat{S}\times([n]\setminus\widehat{S})$, we have the followings*

(i) $\overset{\tau}{\det}(\boldsymbol{X}) = \overset{\tau-1}{\det}(\boldsymbol{X}_{-i})\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i$,

(ii) $\begin{cases}\overset{\tau}{\det}(\boldsymbol{X}_{-i}+\boldsymbol{v}_j\boldsymbol{v}_j^\top) = \overset{\tau-1}{\det}(\boldsymbol{X}_{-i})\boldsymbol{v}_j^\top(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_j, & \textit{if } \boldsymbol{v}_j\notin\operatorname{col}(\boldsymbol{X}_{-i}),\\ \overset{\tau-1}{\det}(\boldsymbol{X}_{-i}+\boldsymbol{v}_j\boldsymbol{v}_j^\top) = \overset{\tau-1}{\det}(\boldsymbol{X}_{-i})(1+\boldsymbol{v}_j^\top\boldsymbol{X}_{-i}^\dagger\boldsymbol{v}_j), & \textit{otherwise},\end{cases}$

(iii) $\boldsymbol{X}^\dagger = \boldsymbol{X}_{-i}^\dagger - \frac{\boldsymbol{X}_{-i}^\dagger \boldsymbol{v}_i \boldsymbol{v}_i^\top (\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})}{||(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_i||_2^2} - \frac{(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{X}_{-i}^\dagger}{||(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_i||_2^2} + \frac{(1+\boldsymbol{v}_i^\top \boldsymbol{X}_{-i}^\dagger \boldsymbol{v}_i)(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_i \boldsymbol{v}_i^\top (\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})}{||(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_i||_2^4}$,

(iv) $\boldsymbol{X}_{-i}^\dagger = \boldsymbol{X}^\dagger - \frac{\boldsymbol{X}^\dagger \boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{X}^\dagger \boldsymbol{X}^\dagger}{||\boldsymbol{X}^\dagger \boldsymbol{v}_i||_2^2} - \frac{\boldsymbol{X}^\dagger \boldsymbol{X}^\dagger \boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{X}^\dagger}{||\boldsymbol{X}^\dagger \boldsymbol{v}_i||_2^2} + \frac{\boldsymbol{v}_i^\top (\boldsymbol{X}^\dagger)^3 \boldsymbol{v}_i \boldsymbol{X}^\dagger \boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{X}^\dagger}{||\boldsymbol{X}^\dagger \boldsymbol{v}_i||_2^4}$,

(v) $\boldsymbol{v}_i^\top \boldsymbol{X}^\dagger \boldsymbol{v}_i = 1$,

(vi) $\boldsymbol{v}_i^\top (\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X}) = \boldsymbol{0}$,

(vii) $\boldsymbol{v}_i^\top (\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_i = \frac{1}{||\boldsymbol{X}^\dagger \boldsymbol{v}_i||_2^2}$,

(viii) $\boldsymbol{v}_j^\top (\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_j = \begin{cases} \boldsymbol{v}_j^\top (\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X})\boldsymbol{v}_j + \frac{(\boldsymbol{v}_j^\top \boldsymbol{X}^\dagger \boldsymbol{v}_i)^2}{||\boldsymbol{X}^\dagger \boldsymbol{v}_i||_2^2}, & \text{if } \boldsymbol{v}_j \notin \operatorname{col}(\boldsymbol{X}_{-i}), \\ 0, & \text{otherwise.} \end{cases}$

Proof. **Part (i).** Let $\boldsymbol{X}_{-i} = \boldsymbol{Q}\operatorname{Diag}(\boldsymbol{\lambda})\boldsymbol{Q}^\top$ denote its eigendecomposition. Since the rank of $\boldsymbol{X}_{-i}$ is $\tau-1$, without loss of generality, we assume that its eigenvalues satisfy $\lambda_1 \geq \cdots \lambda_{\tau-1} > \lambda_\tau = \cdots = \lambda_d = 0$.

For any $\epsilon > 0$, we have

$$\begin{aligned}\det(\boldsymbol{X} + \epsilon \boldsymbol{I}_d) &= \det(\boldsymbol{X}_{-i} + \epsilon \boldsymbol{I}_d)\left(1 + \boldsymbol{v}_i^\top (\boldsymbol{X}_{-i} + \epsilon \boldsymbol{I}_d)^{-1}\boldsymbol{v}_i\right) \\ &= \epsilon^{n-\tau+1} \prod_{i\in[\tau-1]} (\lambda_i + \epsilon)\left(1 + \boldsymbol{v}_i^\top (\boldsymbol{X}_{-i} + \epsilon \boldsymbol{I}_d)^{-1}\boldsymbol{v}_i\right) \\ &= \epsilon^{n-\tau} \prod_{i\in[\tau-1]} (\lambda_i + \epsilon)\left(\epsilon + \boldsymbol{v}_i^\top \boldsymbol{Q}\operatorname{Diag}(\boldsymbol{\beta}(\epsilon))\boldsymbol{Q}^\top \boldsymbol{v}_i\right),\end{aligned}$$

where the first equality is from the Matrix Determinant lemma (Harville 1998) and in the third equality, we let $\boldsymbol{\beta}(\epsilon) = (\frac{\epsilon}{\lambda_1+\epsilon}, \cdots, \frac{\epsilon}{\lambda_{\tau-1}+\epsilon}, 1, \cdots, 1)^\top$ denote the eigenvalues of $\epsilon(\boldsymbol{X}_{-i} + \epsilon \boldsymbol{I}_d)^{-1}$. As $\overset{\tau}{\det}(\boldsymbol{X}) = \lim_{\epsilon\to 0} \epsilon^{-(n-\tau)} \det(\boldsymbol{X} + \epsilon \boldsymbol{I}_d)$, thus

$$\begin{aligned}\overset{\tau}{\det}(\boldsymbol{X}) &= \lim_{\epsilon\to 0} \frac{\det(\boldsymbol{X} + \epsilon \boldsymbol{I}_d)}{\epsilon^{n-\tau}} = \lim_{\epsilon\to 0} \prod_{i\in[\tau-1]} (\lambda_i + \epsilon)\left(\epsilon + \boldsymbol{v}_i^\top \boldsymbol{Q}\operatorname{Diag}(\boldsymbol{\beta}(\epsilon))\boldsymbol{Q}^\top \boldsymbol{v}_i\right) \\ &= \lim_{\epsilon\to 0} \prod_{i\in[\tau-1]} (\lambda_i + \epsilon) \lim_{\epsilon\to 0}\left(\epsilon + \boldsymbol{v}_i^\top \boldsymbol{Q}\operatorname{Diag}(\boldsymbol{\beta}(\epsilon))\boldsymbol{Q}^\top \boldsymbol{v}_i\right) \\ &= \overset{\tau-1}{\det}(\boldsymbol{X}_{-i})\left(\boldsymbol{v}_i^\top \boldsymbol{Q}\operatorname{Diag}(\boldsymbol{\beta}(0))\boldsymbol{Q}^\top \boldsymbol{v}_i\right) = \overset{\tau-1}{\det}(\boldsymbol{X}_{-i})\boldsymbol{v}_i^\top (\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_i,\end{aligned}$$

where the third equality is because both limits exist and the last equality is from the fact that the vector of eigenvalues of $(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})$ is equal to $\boldsymbol{\beta}(0)$ and the corresponding matrix consisting of the eigenvectors is $\boldsymbol{Q}$.

The proof of **Part (ii)** is similar to **Part (i)** and is thus omitted here.

Part (iii) and **Part (iv)** follow directly from theorem 1 and theorem 6 in Meyer (1973).

Part (v). By **Part (iii)** and the fact that $(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})$ is a projection matrix, we have

$$\boldsymbol{v}_i^\top \boldsymbol{X}^\dagger \boldsymbol{v}_i = \boldsymbol{v}_i^\top \boldsymbol{X}_{-i}^\dagger \boldsymbol{v}_i - \frac{\boldsymbol{v}_i^\top \boldsymbol{X}_{-i} \boldsymbol{v}_i \boldsymbol{v}_i^\top (\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_i}{||(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_i||_2^2} - \frac{\boldsymbol{v}_i^\top (\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{X}_{-i} \boldsymbol{v}_i}{||(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_i||_2^2}$$

$$
\begin{aligned}
&+\frac{(1+\boldsymbol{v}_i^\top\boldsymbol{X}_{-i}\boldsymbol{v}_i)\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i}{||(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i||_2^4}\\
=&\boldsymbol{v}_i^\top\boldsymbol{X}_{-i}^\dagger\boldsymbol{v}_i-\boldsymbol{v}_i^\top\boldsymbol{X}_{-i}^\dagger\boldsymbol{v}_i-\boldsymbol{v}_i^\top\boldsymbol{X}_{-i}^\dagger\boldsymbol{v}_i+1+\boldsymbol{v}_i^\top\boldsymbol{X}_{-i}^\dagger\boldsymbol{v}_i=1.
\end{aligned}
$$

Part (vi). Since $\boldsymbol{X}=\boldsymbol{X}_{-i}+\boldsymbol{v}_i\boldsymbol{v}_i^\top$, then we have

$$
\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}^\dagger\boldsymbol{X})=\boldsymbol{v}_i^\top-\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{X}_{-i}-\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top=-\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{X}_{-i},
$$

where the second equality is from the fact that $\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{v}_i=1$ in **Part (v)**.

To compute $\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{X}_{-i}$, using the result in **Part (iii)** and the facts that $(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{X}_{-i}^\dagger=0$ and $(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})$ is a projection matrix, we then obtain

$$
\begin{aligned}
\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{X}_{-i}=&\boldsymbol{v}_i^\top\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i}^\dagger-\frac{\boldsymbol{v}_i^\top\boldsymbol{X}_{-i}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{X}_{-i}^\dagger}{||(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i||_2^2}-\frac{\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i\boldsymbol{v}_i^\top\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i}^\dagger}{||(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i||_2^2}\\
&+\frac{(1+\boldsymbol{v}_i^\top\boldsymbol{X}_{-i}^\dagger\boldsymbol{v}_i)\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{X}_{-i}^\dagger}{||(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i||_2^4}\\
=&\boldsymbol{v}_i^\top\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i}^\dagger-\boldsymbol{v}_i^\top\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i}^\dagger=\boldsymbol{0}.
\end{aligned}
$$

Hence, $\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}^\dagger\boldsymbol{X})=-\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{X}_{-i}=\boldsymbol{0}$.

Part (vii). According to **Part (iv)**, we have

$$
\begin{aligned}
\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}=&\boldsymbol{X}^\dagger\boldsymbol{X}-\frac{\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^2}-\frac{\boldsymbol{X}^\dagger\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{X}}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^2}+\frac{\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^3\boldsymbol{v}_i\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{X}}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^4}, && (35)\\
\boldsymbol{X}_{-i}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top=&\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top-\frac{\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^2}-\frac{\boldsymbol{X}^\dagger\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^2}+\frac{\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^3\boldsymbol{v}_i\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^4}\\
=&-\frac{\boldsymbol{X}^\dagger\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^2}+\frac{\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^3\boldsymbol{v}_i\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^4}. && (36)
\end{aligned}
$$

where the third equality is due to $\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{v}_i=1$ from **Part (v)**.

Since $\boldsymbol{X}=\boldsymbol{X}_{-i}+\boldsymbol{v}_i\boldsymbol{v}_i^\top$, we can obtain

$$
\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i=\boldsymbol{v}_i^\top\boldsymbol{v}_i-\boldsymbol{v}_i^\top\boldsymbol{X}_{-i}^\dagger(\boldsymbol{X}-\boldsymbol{v}_i\boldsymbol{v}_i^\top)\boldsymbol{v}_i=\boldsymbol{v}_i^\top\boldsymbol{v}_i-\boldsymbol{v}_i^\top\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}\boldsymbol{v}_i+\boldsymbol{v}_i^\top\boldsymbol{X}_{-i}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top\boldsymbol{v}_i.
$$

Applying the identities in (35) and (36), we further have

$$
\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i=\frac{1}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^2}+\frac{\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^3\boldsymbol{v}_i\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}^\dagger\boldsymbol{X})\boldsymbol{v}_i}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^4}=\frac{1}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^2}.
$$

where the last equality is due to the fact that $\boldsymbol{v}_i^\top(\boldsymbol{I}_d-\boldsymbol{X}^\dagger\boldsymbol{X})=\boldsymbol{0}$ from **Part (vi)**.

Part (viii). There are two cases: whether $\boldsymbol{v}_j$ is in the column space of $\boldsymbol{X}_{-i}$ or not.

(a) If $\boldsymbol{v}_j \notin \operatorname{col}(\boldsymbol{X}_{-i})$, we will follow the proof of **Part (vii)**. Since $\boldsymbol{X} = \boldsymbol{X}_{-i} + \boldsymbol{v}_i\boldsymbol{v}_i^\top$, we can obtain

$$\begin{aligned}\boldsymbol{v}_j^\top(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_j =& \boldsymbol{v}_j^\top\boldsymbol{v}_j - \boldsymbol{v}_j^\top\boldsymbol{X}_{-i}^\dagger\boldsymbol{X}\boldsymbol{v}_j + \boldsymbol{v}_j^\top\boldsymbol{X}_{-i}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top\boldsymbol{v}_j = \boldsymbol{v}_j^\top(\boldsymbol{I}_d - \boldsymbol{X}^\dagger\boldsymbol{X})\boldsymbol{v}_j + \frac{(\boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_i)^2}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^2}\\ &- \boldsymbol{v}_i^\top(\boldsymbol{I}_d - \boldsymbol{X}^\dagger\boldsymbol{X})\boldsymbol{v}_j\frac{\boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{X}^\dagger\boldsymbol{v}_i}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^2} + \boldsymbol{v}_i^\top(\boldsymbol{I}_d - \boldsymbol{X}^\dagger\boldsymbol{X})\boldsymbol{v}_j\frac{\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^3\boldsymbol{v}_i\boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_i}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^4}\\ =& \boldsymbol{v}_j^\top(\boldsymbol{I}_d - \boldsymbol{X}^\dagger\boldsymbol{X})\boldsymbol{v}_j + \frac{(\boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_i)^2}{||\boldsymbol{X}^\dagger\boldsymbol{v}_i||_2^2},\end{aligned}$$

where the second equality is due to the identites in (35) and (36), and the last equality is because $\boldsymbol{v}_i^\top(\boldsymbol{I}_d - \boldsymbol{X}^\dagger\boldsymbol{X}) = \boldsymbol{0}$ from **Part (vi)**.

(b) Second, if $\boldsymbol{v}_j \in \operatorname{col}(\boldsymbol{X}_{-i})$, then we rewrite $\boldsymbol{v}_j = \sum_{\ell\in\widehat{S}\setminus\{i\}} a_\ell\boldsymbol{v}_l$, which stems from the fact that the vectors $\{\boldsymbol{v}_\ell, \ell\in\widehat{S}\setminus\{i\}\}$ span the column space of $\boldsymbol{X}_{-i}$. Then it follows that

$$\boldsymbol{v}_j^\top(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_j = \sum_{\ell\in S\setminus\{i\}} a_\ell\boldsymbol{v}_\ell^\top(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_j = 0,$$

where the second equality is because $\boldsymbol{v}_\ell^\top(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i}) = \boldsymbol{0}$ for all $\ell\in\widehat{S}\setminus\{i\}$ from **Part (vi)**.

□

A.8 Proof of Lemma 7

Lemma 7 *Let $\widehat{S}$ denote the output of the local search Algorithm 4 and let $\boldsymbol{X} = \sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top$. Then for each pair $(i,j)\in\widehat{S}\times([n]\setminus\widehat{S})$, the following inequality holds*

$$1 \geq \left(\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{X}^\dagger\boldsymbol{v}_i\right)\boldsymbol{v}_j^\top(\boldsymbol{I}_d - \boldsymbol{X}^\dagger\boldsymbol{X})\boldsymbol{v}_j + \boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{v}_j.$$

Proof. For each pair $(i,j)\in\widehat{S}\times([n]\setminus\widehat{S})$, the stopping criterion of Algorithm 4 implies that

$$\overset{s}{\det}(\boldsymbol{X}_{-i} + \boldsymbol{v}_i\boldsymbol{v}_i^\top) \geq \overset{s}{\det}(\boldsymbol{X}_{-i} + \boldsymbol{v}_j\boldsymbol{v}_j^\top), \tag{37}$$

and $\{\boldsymbol{v}_\ell\}_{\ell\in\widehat{S}}$ are linearly independent. There are two cases to be considered: whether $\boldsymbol{v}_j$ is in the column space of $\boldsymbol{X}_{-i}$ or not.

(i) If $\boldsymbol{v}_j \notin \operatorname{col}(\boldsymbol{X}_{-i})$, then by Parts (i) and (ii) in Lemma 6 and the fact that $\overset{s-1}{\det}(\boldsymbol{X}_{-i}) > 0$, the local optimality condition (37) is equivalent to

$$\boldsymbol{v}_i^\top(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_i \geq \boldsymbol{v}_j^\top(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger\boldsymbol{X}_{-i})\boldsymbol{v}_j. \tag{38}$$

Plugging the results of Parts (vii) and (viii) in Lemma 6, the above inequality is further reduced to

$$1 \geq \left(\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{X}^\dagger\boldsymbol{v}_i\right)\boldsymbol{v}_j^\top(\boldsymbol{I}_d - \boldsymbol{X}^\dagger\boldsymbol{X})\boldsymbol{v}_j + \boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{v}_j. \tag{39}$$

(ii) If $\boldsymbol{v}_j \in \operatorname{col}(\boldsymbol{X}_{-i})$, then we must have $\boldsymbol{v}_j \in \operatorname{col}(\boldsymbol{X})$. According to Part (vi) in Lemma 6, we have

$$\left(\boldsymbol{v}_i^\top \boldsymbol{X}^\dagger \boldsymbol{X}^\dagger \boldsymbol{v}_i\right) \boldsymbol{v}_j^\top (\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X})\boldsymbol{v}_j + \boldsymbol{v}_j^\top \boldsymbol{X}^\dagger \boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{X}^\dagger \boldsymbol{v}_j = (\boldsymbol{v}_i^\top \boldsymbol{X}^\dagger \boldsymbol{v}_j)^2.$$

Using Part (iii) in Lemma 6, we have

$$\begin{aligned}
\boldsymbol{v}_i^\top \boldsymbol{X}^\dagger \boldsymbol{v}_j =& \boldsymbol{v}_i^\top \boldsymbol{X}_{-i}^\dagger \boldsymbol{v}_j - \frac{\boldsymbol{v}_i^\top \boldsymbol{X}_{-i} \boldsymbol{v}_i \boldsymbol{v}_i^\top (\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_j}{||(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_i||_2^2} - \frac{\boldsymbol{v}_i^\top (\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{X}_{-i} \boldsymbol{v}_j}{||(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_i||_2^2} \\
&+ \frac{(1 + \boldsymbol{v}_i^\top \boldsymbol{X}_{-i} \boldsymbol{v}_i) \boldsymbol{v}_i^\top (\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i}) \boldsymbol{v}_i \boldsymbol{v}_i^\top (\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_j}{||(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_i||_2^4} \\
=& \boldsymbol{v}_i^\top \boldsymbol{X}_{-i}^\dagger \boldsymbol{v}_j - \frac{\boldsymbol{v}_i^\top (\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{X}_{-i} \boldsymbol{v}_j}{||(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})\boldsymbol{v}_i||_2^2} = 0,
\end{aligned}$$

where the first equality is due to Part (iii) in Lemma 6, the second equality is due to Part (vi) in Lemma 6 and $\boldsymbol{v}_j$ is a linear combination of $\{\boldsymbol{v}_\ell\}_{\ell \in \widehat{S} \setminus \{i\}}$, and the last equality is because $(\boldsymbol{I}_d - \boldsymbol{X}_{-i}^\dagger \boldsymbol{X}_{-i})$ is a projection matrix.

Thus, clearly, we arrive at

$$\left(\boldsymbol{v}_i^\top \boldsymbol{X}^\dagger \boldsymbol{X}^\dagger \boldsymbol{v}_i\right) \boldsymbol{v}_j^\top (\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X})\boldsymbol{v}_j + \boldsymbol{v}_j^\top \boldsymbol{X}^\dagger \boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{X}^\dagger \boldsymbol{v}_j = (\boldsymbol{v}_i^\top \boldsymbol{X}^\dagger \boldsymbol{v}_j)^2 = 0 \leq 1.$$

□

A.9 Proof of Theorem 7

Theorem 7 *Let $\widehat{S}$ denote the output of the local search Algorithm 4, then the set $\widehat{S}$ yields a $s\min\{\log(s), \log(n-s-n/s+2)\}$-approximation bound for MESP (2), i.e.,*

$$\log \overset{s}{\det}\left(\sum_{i\in\widehat{S}} \boldsymbol{v}_i \boldsymbol{v}_i^\top\right) \geq z^* - s\min\left\{\log(s), \log\left(n - s - \frac{n}{s} + 2\right)\right\}.$$

Proof. We split the proof into three steps.

Step 1. Constructing Solution of Dual Variable $\boldsymbol{\Lambda}$.

Given the output $\widehat{S}$ of the local search Algorithm 4, let us denote $\boldsymbol{X} = \sum_{i\in\widehat{S}} \boldsymbol{v}_i \boldsymbol{v}_i^\top$ and let $\boldsymbol{X}_{-i} = \boldsymbol{X} - \boldsymbol{v}_i \boldsymbol{v}_i^\top$ for each $i \in \widehat{S}$.

We first construct $\boldsymbol{\Lambda}$ of LD (5) as below

$$\boldsymbol{\Lambda} = \frac{1}{t}\left[\operatorname{tr}(\boldsymbol{X}^\dagger)(\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X}) + \boldsymbol{X}^\dagger\right], \tag{40}$$

where $t > 0$ is a scaling factor and will be specified later. Accordingly, the identity (40) leads to that $\log \underset{s}{\det}(\boldsymbol{\Lambda}) = \log \overset{s}{\det}(\boldsymbol{X}) + s\log t$.

Step 2. Constructing Solution of the Other Dual Variables $(\nu, \boldsymbol{\mu})$ with $\boldsymbol{\Lambda}$ in (40).

Next, to construct the solution of the other two dual variables $(\nu, \boldsymbol{\mu})$, we need to check the feasibility of constraints in LD (5), i.e.,

$$\boldsymbol{v}_i^\top \boldsymbol{\Lambda} \boldsymbol{v}_i \leq \nu + \mu_i, \forall i \in [n]. \tag{41}$$

We consider the following two cases: (i) for each $i \in \widehat{S}$ and (ii) for each $j \in [n] \setminus \widehat{S}$.

(i) For each $i \in \widehat{S}$, we have

$$\boldsymbol{v}_i^\top \boldsymbol{\Lambda} \boldsymbol{v}_i = \frac{1}{t}\left[\operatorname{tr}(\boldsymbol{X}^\dagger)\boldsymbol{v}_i^\top (\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X})\boldsymbol{v}_i + \boldsymbol{v}_i^\top \boldsymbol{X}^\dagger \boldsymbol{v}_i\right] = \frac{1}{t}, \tag{42}$$

where the second equality results from Parts (v) and (vi) in Lemma 6 with $\tau = s$.

(ii) For each $j \in [n] \setminus \widehat{S}$, according to Lemma 7, we have

$$1 \geq \left(\boldsymbol{v}_i^\top \boldsymbol{X}^\dagger \boldsymbol{X}^\dagger \boldsymbol{v}_i\right) \boldsymbol{v}_j^\top (\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X})\boldsymbol{v}_j + \boldsymbol{v}_j^\top \boldsymbol{X}^\dagger \boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{X}^\dagger \boldsymbol{v}_j, \forall i \in \widehat{S}.$$

Summing the above inequality over $i \in \widehat{S}$ and using the fact that $\boldsymbol{X} = \sum_{i \in \widehat{S}} \boldsymbol{v}_i \boldsymbol{v}_i^\top$, we have

$$s \geq \operatorname{tr}(\boldsymbol{X}^\dagger)\boldsymbol{v}_j^\top (\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X})\boldsymbol{v}_j + \boldsymbol{v}_j^\top \boldsymbol{X}^\dagger \boldsymbol{v}_j = t\boldsymbol{v}_j^\top \boldsymbol{\Lambda} \boldsymbol{v}_j. \tag{43}$$

By inequalities (42) and (43), to find the best $(\nu, \boldsymbol{\mu})$, it suffices to solve the optimization problem below:

$$z^{LD} \leq \min_{t>0} \min_{\nu, \boldsymbol{\mu} \in \mathbb{R}_+^n} \left\{ \log \overset{s}{\det}(\boldsymbol{X}) + s\log(t) + s\nu + \sum_{i \in [n]} \mu_i - s : \nu + \mu_i \geq \frac{1}{t}, \forall i \in \widehat{S}, \nu + \mu_i \geq \frac{s}{t}, \forall i \in [n] \setminus \widehat{S} \right\}.$$

Above, by checking the primal and dual of inner minimization problems, there are following two candidate optimal solutions

$$\nu^a = \frac{s}{t}, \mu_i^a = 0, \forall i \in [n],$$
$$\nu^b = \frac{1}{t}, \mu_i^b = 0, \forall i \in \widehat{S}, \mu_i^b = \frac{s-1}{t}, \forall i \in [n] \setminus \widehat{S}.$$

Step 3. Finding the Best Scaler t and Proving the Approximation Bound.

Plugging in these two candidate solutions of $(\nu, \boldsymbol{\mu})$, the right-hand side of the above minimization problem becomes

$$z^{LD} \leq \log \overset{s}{\det}(\boldsymbol{X}) + \min_{t>0} \min \left\{ s\log(t) + s\left(\frac{s}{t} - 1\right), s\log(t) + (n-s)\frac{s-1}{t} + \frac{s}{t} - s \right\}.$$

By swapping the two minimum operators and optimizing over t, the right-hand side of above inequality is further equivalent to

$$z^{LD} \leq \log \overset{s}{\det}(\boldsymbol{X}) + s\min\left\{\log(s), \log\left(n - s - \frac{n}{s} + 2\right)\right\}.$$

According to the weak duality between MESP (3) and LD (5) and the fact that $\widehat{S}$ is feasible to MESP (1), we have

$$\log \overset{s}{\det}\left(\sum_{i \in \widehat{S}} \boldsymbol{v}_i \boldsymbol{v}_i^\top\right) = \log \overset{s}{\det}(\boldsymbol{X}) \leq z^* \leq z^{LD} \leq \log \overset{s}{\det}(\boldsymbol{X}) + s\min\left\{\log(s), \log\left(n - s - \frac{n}{s} + 2\right)\right\},$$

which completes the proof. □

A.10 Proof of Proposition 5

Proposition 5 *If one follows the consruction of a feasible* $\mathbf{\Lambda}$ *in* (15) *to LD* (5)*, then even with the best choice of* $(\nu, \boldsymbol{\mu})$*, there exists an instance such that*

$$-\log \det_s(\mathbf{\Lambda}) + s\nu + \sum_{i\in[n]} \mu_i - s = z^* + s\min\{\log(s), \log(n-s-n/s+2)\}.$$

Proof. We construct the following instance.

Example 3 *Given* $s \le d \le n$*, suppose that for each* $i \in [n]$*,*

$$\boldsymbol{v}_i = \begin{cases} \boldsymbol{e}_i, & \text{if } i \in [s], \\ \sum_{j\in[s]} \boldsymbol{e}_j, & \text{otherwise.} \end{cases}$$

In the above example, one optimal solution to MESP (2) is $S^* = [s]$. Suppose in the local search Algorithm 4, we start with $\widehat{S} = S^*$, then it will terminate immediately. We follow (15) to construct a feasible $\mathbf{\Lambda}$ to LD, which is identical to the one (40) used in Theorem 7. According to the proof of Theorem 7, we only need to check if the inequalities (43) are tight, i.e.,

$$s = \mathrm{tr}(\boldsymbol{X}^\dagger)\boldsymbol{v}_j^\top(\boldsymbol{I}_d - \boldsymbol{X}^\dagger\boldsymbol{X})\boldsymbol{v}_j + \boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_j = t\boldsymbol{v}_j^\top\mathbf{\Lambda}\boldsymbol{v}_j, \forall j \in [s+1, n].$$

In fact,

$$\begin{aligned}\mathrm{tr}(\boldsymbol{X}^\dagger)\boldsymbol{v}_j^\top(\boldsymbol{I}_d - \boldsymbol{X}^\dagger\boldsymbol{X})\boldsymbol{v}_j + \boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_j &= \mathrm{tr}(\boldsymbol{X}^\dagger)\Big(\sum_{i\in[s]}\boldsymbol{e}_i\Big)^\top(\boldsymbol{I}_d - \boldsymbol{X}^\dagger\boldsymbol{X})\Big(\sum_{i\in[s]}\boldsymbol{e}_i\Big) + \boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_j \\ &= \sum_{i\in[s]}\boldsymbol{e}_i^\top\boldsymbol{X}^\dagger\boldsymbol{e}_i = s, \forall j \in [s+1, n],\end{aligned}$$

where the second equality is due to Part (vi) in Lemma 6 with $\tau = s$ and the third one is due to $\boldsymbol{X} = \sum_{i\in[s]}\boldsymbol{e}_i\boldsymbol{e}_i^\top$ and $\boldsymbol{e}_i^\top\boldsymbol{X}^\dagger\boldsymbol{e}_\ell = 0$ for all $i, \ell \in [s]$ and $i \neq \ell$. □

A.11 Proof of Proposition 6

Proposition 6 *Let* $\widehat{S}$ *denote the output of the local search Algorithm 4. Suppose that* $\boldsymbol{v}_i^\top\boldsymbol{v}_j = 0$ *for each pair* $(i,j) \in \widehat{S} \times ([n] \setminus \widehat{S})$*, then we have*

$$\log \det^s\Big(\sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top\Big) \ge z^* - s\min\left\{\log\left(\frac{\lambda_{\max}(\boldsymbol{C})}{\delta}\right), \log\left(\frac{\lambda_{\max}(\boldsymbol{C})}{s\delta}(n-s) - \frac{n}{s} + 2\right)\right\},$$

where the constant δ *is defined in Lemma 4.*

Proof. The proof follows directly from Theorem 7. Thus, we only sketch the proof for the sake of page limit.

Step 0. Given the output $\widehat{S}$ of the local search Algorithm 4, let us denote $\boldsymbol{X}=\sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top$. Let $\lambda_1\geq\cdots\geq\lambda_s>\lambda_{s+1}=\cdots=\lambda_d=0$ denote the eigenvalues of $\boldsymbol{X}$. Clearly, according to the definition of δ and Cauchy's Interlacing theorem (Bellman 1997), we have $\lambda_{\max}(\boldsymbol{C})\geq\lambda_1$ and $\lambda_s\geq\delta$.

Step 1. Construct $\boldsymbol{\Lambda}=(\lambda_s t)^{-1}(\boldsymbol{I}_d-\boldsymbol{X}^\dagger\boldsymbol{X})+t^{-1}\boldsymbol{X}^\dagger$ such that $\log\overset{s}{\det}(\boldsymbol{X})=-\log\underset{s}{\det}(\boldsymbol{\Lambda})+s\log t$.

Step 2. We can show that $\boldsymbol{v}_i^\top\boldsymbol{\Lambda}\boldsymbol{v}_i=1/t$ for all $i\in\widehat{S}$.

Since the vectors $\{\boldsymbol{v}_i\}_{i\in\widehat{S}}$ span the column space of $\boldsymbol{X}$, the assumption that $\boldsymbol{v}_i^\top\boldsymbol{v}_j=0$ for each pair $(i,j)\in\widehat{S}\times([n]\setminus\widehat{S})$ implies that $\boldsymbol{v}_j$ is orthogonal to the column space of $\boldsymbol{X}$. Thus, we have

$$\boldsymbol{v}_j^\top\boldsymbol{X}=\boldsymbol{0},\boldsymbol{v}_j^\top\boldsymbol{X}^\dagger=\boldsymbol{0},\boldsymbol{v}_j^\top\boldsymbol{\Lambda}\boldsymbol{v}_j=(\lambda_s t)^{-1}\boldsymbol{v}_j^\top\boldsymbol{v}_j,\forall j\in[n]\setminus\widehat{S}.$$

To obtain the upper bound of $\boldsymbol{v}_j^\top\boldsymbol{\Lambda}\boldsymbol{v}_j$, according to Lemma 7, we have

$$1\geq\left(\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{X}^\dagger\boldsymbol{v}_i\right)\boldsymbol{v}_j^\top(\boldsymbol{I}_d-\boldsymbol{X}^\dagger\boldsymbol{X})\boldsymbol{v}_j+\boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_i\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{v}_j=\left(\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{X}^\dagger\boldsymbol{v}_i\right)\boldsymbol{v}_j^\top\boldsymbol{v}_j,\forall i\in\widehat{S},$$

where the equality is due to $\boldsymbol{v}_j^\top\boldsymbol{X}^\dagger=\boldsymbol{0}$. Summing the above inequalities over $i\in\widehat{S}$, then for each $j\in[n]\setminus\widehat{S}$, we have

$$\boldsymbol{v}_j^\top\boldsymbol{\Lambda}\boldsymbol{v}_j=\frac{1}{\lambda_s t}\boldsymbol{v}_j^\top\boldsymbol{v}_j\leq\frac{1}{\lambda_s t}\frac{s}{\operatorname{tr}(\boldsymbol{X}^\dagger)}\leq\frac{\lambda_1}{\lambda_s t}\leq\frac{\lambda_{\max}(\boldsymbol{C})}{\delta t},$$

where the second inequality is due to $\lambda_1\operatorname{tr}(\boldsymbol{X}^\dagger)\geq s$, and the third inequality is from $\lambda_{\max}(\boldsymbol{C})\geq\lambda_1$ and $\delta\leq\lambda_s$.

Step 3. To choose $(\nu,\boldsymbol{\mu})$ such that $(\boldsymbol{\Lambda},\nu,\boldsymbol{\mu})$ is feasible to LD (5), let us consider the optimization problem below

$$\min_{t>0}\min_{\nu,\boldsymbol{\mu}\in\mathbb{R}_+^n}\left\{\log\overset{s}{\det}(\boldsymbol{X})+s\log t+s\nu+\sum_{i\in[n]}\mu_i-s:\nu+\mu_i\geq\frac{1}{t},\forall i\in\widehat{S},\nu+\mu_i\geq\frac{\lambda_{\max}(\boldsymbol{C})}{\delta t},\forall i\in[n]\setminus\widehat{S}\right\},$$

which provides an upper bound to z^{LD}. By optimizing the right-hand side, we obtain

$$z^{LD}\leq\log\overset{s}{\det}(\boldsymbol{X})+\min\left\{s\log\left(\frac{\lambda_{\max}(\boldsymbol{C})}{\delta}\right),s\log\left(\frac{\lambda_{\max}(\boldsymbol{C})}{s\delta}(n-s)+\frac{2s-n}{s}\right)\right\}.$$

Invoking the weak duality between MESP (1) and LD (5) and the fact that $\widehat{S}$ is feasible to MESP (1), we conclude that

$$\log\overset{s}{\det}(\boldsymbol{X})\leq z^*\leq z^{LD}\leq\log\overset{s}{\det}(\boldsymbol{X})+s\min\left\{\log\left(\frac{\lambda_{\max}(\boldsymbol{C})}{\delta}\right),\log\left(\frac{\lambda_{\max}(\boldsymbol{C})}{s\delta}(n-s)-n/s+2\right)\right\}.$$

□

A.12 Proof of Proposition 7

Proposition 7 *The objective function of A-MESP* (17) *is (i) monotonic non-decreasing, (ii) neither discrete-supermodular nor discrete-submodular, and (iii) neither convex nor concave.*

Proof. **Part (i).** For any size-s subset $S \subseteq [n]$ with $s \geq 1$, let $\boldsymbol{X} = \sum_{\ell \in S} \boldsymbol{v}_\ell \boldsymbol{v}_\ell^\top$, then for any $i \in [n] \setminus S$, we have

$$\overset{s+1}{\operatorname{tr}}\left[(\boldsymbol{X} + \boldsymbol{v}_i \boldsymbol{v}_i^\top)^\dagger\right] = \overset{s}{\operatorname{tr}}(\boldsymbol{X}^\dagger) + \frac{1 + \boldsymbol{v}_i^\top \boldsymbol{X}^\dagger \boldsymbol{v}_i}{\boldsymbol{v}_i^\top (\boldsymbol{I}_n - \boldsymbol{X}\boldsymbol{X}^\dagger)\boldsymbol{v}_i} \geq \overset{s}{\operatorname{tr}}(\boldsymbol{X}^\dagger),$$

where the equality is due to Part (iii) in Lemma 6 , and thus proves the monotonicity.

Part (ii). Consider an instance of $n = 3$, $\boldsymbol{v}_1 = 2\boldsymbol{e}_1 + \boldsymbol{e}_2$, $\boldsymbol{v}_2 = 2\boldsymbol{e}_1 - \boldsymbol{e}_2$ and $\boldsymbol{v}_3 \in \mathbb{R}^3$. Then we let $S_1 = \{1\}, S_2 = \{1, 2\}$ and $\boldsymbol{X}_1 = \sum_{i \in S_1} \boldsymbol{v}_i \boldsymbol{v}_i^\top$, $\boldsymbol{X}_2 = \sum_{i \in S_2} \boldsymbol{v}_i \boldsymbol{v}_i^\top$. In this way, we have

$$\boldsymbol{X}_1 = \begin{pmatrix} 4 & 2 & 0 \\ 2 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad \boldsymbol{X}_1^\dagger = \begin{pmatrix} 0.16 & 0.08 & 0 \\ 0.08 & 0.04 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad \boldsymbol{X}_2 = \begin{pmatrix} 8 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad \boldsymbol{X}_2^\dagger = \begin{pmatrix} 0.125 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

If $\boldsymbol{v}_3 = (40 \;\; 10 \;\; 20)^\top$, then

$$\overset{2}{\operatorname{tr}}\left[(\boldsymbol{X}_1 + \boldsymbol{v}_3 \boldsymbol{v}_3^\top)^\dagger\right] - \overset{1}{\operatorname{tr}}(\boldsymbol{X}_1^\dagger) = \frac{1 + 324}{480} \geq \frac{1 + 250}{400} = \overset{3}{\operatorname{tr}}\left[(\boldsymbol{X}_2 + \boldsymbol{v}_3 \boldsymbol{v}_3^\top)^\dagger\right] - \overset{2}{\operatorname{tr}}(\boldsymbol{X}_2^\dagger),$$

which disproves the discrete-supermodularity.

If $\boldsymbol{v}_3 = (10 \;\; 10 \;\; 20)^\top$, then

$$\overset{2}{\operatorname{tr}}\left[(\boldsymbol{X}_1 + \boldsymbol{v}_3 \boldsymbol{v}_3^\top)^\dagger\right] - \overset{1}{\operatorname{tr}}(\boldsymbol{X}_1^\dagger) = \frac{1 + 52}{420} \leq \frac{1 + 62.5}{400} = \overset{3}{\operatorname{tr}}\left[(\boldsymbol{X}_2 + \boldsymbol{v}_3 \boldsymbol{v}_3^\top)^\dagger\right] - \overset{2}{\operatorname{tr}}(\boldsymbol{X}_2^\dagger),$$

which disproves the discrete-submodularity.

Part (iii). Let us consider Example 2 in Proposition 1. In this example, we consider two feasible solutions $\boldsymbol{x}^1 = (1, 0)^\top$ and $\boldsymbol{x}^2 = (0, 1)^\top$ of A-MESP (17) with $s = 1$. The following two cases disprove the convexity and concavity:

Case 1. If $a = 1$ and $b = 1$, we have

$$\frac{1}{2}\overset{1}{\operatorname{tr}}\left[\left(\boldsymbol{v}_1 \boldsymbol{v}_1^\top\right)^\dagger\right] + \frac{1}{2}\overset{1}{\operatorname{tr}}\left[\left(\boldsymbol{v}_2 \boldsymbol{v}_2^\top\right)^\dagger\right] = 1 \leq \overset{1}{\operatorname{tr}}\left[\left(\sum_{i \in [n]} \frac{x_i^1 + x_i^2}{2} \boldsymbol{v}_i \boldsymbol{v}_i^\top\right)^\dagger\right] = 2,$$

which disproves the convexity.

Case 2. If $a = 4$ and $b = 1$, then we have

$$\frac{1}{2}\overset{1}{\operatorname{tr}}\left[\left(\boldsymbol{v}_1 \boldsymbol{v}_1^\top\right)^\dagger\right] + \frac{1}{2}\overset{1}{\operatorname{tr}}\left[\left(\boldsymbol{v}_2 \boldsymbol{v}_2^\top\right)^\dagger\right] = \frac{1}{8} + \frac{1}{2} \geq \overset{1}{\operatorname{tr}}\left[\left(\sum_{i \in [n]} \frac{x_i^1 + x_i^2}{2} \boldsymbol{v}_i \boldsymbol{v}_i^\top\right)^\dagger\right] = \frac{1}{2},$$

which disproves the concavity. □

A.13 Proof of Lemma 8

Lemma 8 *For a $d\times d$ matrix $\boldsymbol{\Lambda}\succeq 0$, we have*

$$\min_{\boldsymbol{X}\succeq 0}\left\{\overset{s}{\operatorname{tr}}(\boldsymbol{X}^{\dagger})+\operatorname{tr}(\boldsymbol{X}\boldsymbol{\Lambda})\right\}=2\operatorname*{tr}_{s}\left(\boldsymbol{\Lambda}^{\frac{1}{2}}\right). \tag{19}$$

Proof. Similar to the proof of Lemma 1, the left-hand side of (19) can be equivalently written as

$$\min_{\substack{\boldsymbol{\lambda}\in\mathbb{R}_+^d\\ \lambda_1\geq\cdots\geq\lambda_d\geq 0}}\left\{\sum_{i\in[s]}\frac{1}{\lambda_i}+\min_{\substack{\boldsymbol{Q},\boldsymbol{\theta}\in\mathbb{R}_+^d\\ \theta_1\leq\cdots\leq\theta_d}}\left\{\sum_{i\in[d]}\theta_i\lambda_i:\boldsymbol{\theta}=\operatorname{diag}(\boldsymbol{Q}^\top\boldsymbol{\Lambda}\boldsymbol{Q}),\boldsymbol{Q}\text{ is orthonormal}\right\}\right\}.$$

Following the proof in Lemma 1 to solve the inner minimization problem, the above optimization problem becomes

$$\min_{\substack{\boldsymbol{\lambda}\in\mathbb{R}_+^d\\ \lambda_1\geq\cdots\geq\lambda_d\geq 0}}\left\{\sum_{i\in[s]}\frac{1}{\lambda_i}+\sum_{i\in[d]}\beta_i\lambda_i\right\}.$$

Minimizing the inner problem over $\boldsymbol{\lambda}$ yields $\lambda_i=\frac{1}{\sqrt{\beta_i}}$ for any $i\in[s]$ and $\lambda_i=0$ otherwise. Thus,

$$\min_{\boldsymbol{X}\succeq 0}\left\{\overset{s}{\operatorname{tr}}(\boldsymbol{X}^{\dagger})+\operatorname{tr}(\boldsymbol{X}\boldsymbol{\Lambda})\right\}=2\sum_{i\in[s]}\sqrt{\beta_i}=2\operatorname*{tr}_{s}\left(\boldsymbol{\Lambda}^{\frac{1}{2}}\right).$$

□

A.14 Proof of Theorem 9

Theorem 9 *The primal characterization of A-LD* (20)*, referred to as (A-PC), is*

$$\text{(A-PC)}\quad z_A^{LD}:=\min_{\boldsymbol{x}}\left\{\Phi_s\left(\sum_{i\in[n]}x_i\boldsymbol{v}_i\boldsymbol{v}_i^\top\right):\sum_{i\in[n]}x_i=s,\boldsymbol{x}\in[0,1]^n\right\}. \tag{21}$$

Proof. For A-LD (20), let $\boldsymbol{x}\in\mathbb{R}_+^n$ denote the Lagrangian multipliers associated with $\nu+\mu_i\geq\boldsymbol{v}_i^\top\boldsymbol{\Lambda}\boldsymbol{v}_i$ for each $i\in[n]$ and thus its dual is equal to

$$z_A^{LD}:=\min_{\boldsymbol{x}\in\mathbb{R}_+^n}\max_{\boldsymbol{\Lambda}\succeq 0,\nu,\boldsymbol{\mu}\in\mathbb{R}_+^n}\left\{2\operatorname*{tr}_{s}\left(\boldsymbol{\Lambda}^{\frac{1}{2}}\right)-s\nu-\sum_{i\in[n]}\mu_i+\sum_{i\in[n]}x_i(\nu+\mu_i-\boldsymbol{v}_i^\top\boldsymbol{\Lambda}\boldsymbol{v}_i)\right\},$$

where according to theorem 3.2.2 in Ben-Tal and Nemirovski (2012), the strong duality holds since the constraint system satisfies the relaxed Slater condition.

Clearly, the inner maximization can be separated into two parts: maximization over $\boldsymbol{\Lambda}\succeq 0$ and maximization over $\nu,\boldsymbol{\mu}\in\mathbb{R}_+^n$.

(i) Let $\boldsymbol{X}=\sum_{i\in[n]}x_i\boldsymbol{v}_i\boldsymbol{v}_i^\top$ and then the inner maximization problem over $\boldsymbol{\Lambda}\succeq 0$ becomes

$$\max_{\boldsymbol{\Lambda}\succeq 0}\left\{2\operatorname*{tr}_{s}\left(\boldsymbol{\Lambda}^{\frac{1}{2}}\right)-\operatorname{tr}(\boldsymbol{\Lambda}\boldsymbol{X})\right\}.$$

Suppose $\boldsymbol{\Lambda}$ has eigenvalues $0\leq\beta_1\leq\cdots\leq\beta_d$ and $\boldsymbol{\Lambda}=\boldsymbol{P}\operatorname{Diag}(\boldsymbol{\beta})\boldsymbol{P}^\top$ with an orthonormal matrix $\boldsymbol{P}$. Let us denote $\boldsymbol{\theta}=\operatorname{diag}(\boldsymbol{P}^\top\boldsymbol{X}\boldsymbol{P})$ and for $\boldsymbol{X}$ with rank r, let $\boldsymbol{X}=\boldsymbol{Q}\operatorname{Diag}(\boldsymbol{\lambda})\boldsymbol{Q}^\top$

denote its eigendecomposition, where $\lambda_1 \geq \cdots \geq \lambda_r > \lambda_{r+1} = \cdots = \lambda_d = 0$ and $\boldsymbol{Q}$ is orthonormal. Following the similar proof of Lemma 3, we can reformulate the above maximization problem as

$$\max_{\substack{\boldsymbol{P},\boldsymbol{\theta}\in\mathbb{R}^d_+,\boldsymbol{\beta}\in\mathbb{R}^d_+,\\ 0\leq\beta_1\leq\cdots\leq\beta_d,\\ \theta_1\geq\cdots\geq\theta_d\geq 0}} \left\{2\sum_{i\in[s]}\sqrt{\beta_i} - \sum_{i\in[d]}\theta_i\beta_i : \boldsymbol{\theta} = \operatorname{diag}(\boldsymbol{P}^\top \boldsymbol{X}\boldsymbol{P}), \boldsymbol{P} \text{ is orthonormal}\right\} = \Phi_s(\boldsymbol{X}),$$

with an optimal solution

$$\boldsymbol{P}^* = \boldsymbol{Q}, \boldsymbol{\theta}^* = \boldsymbol{\lambda}, \beta_i^* = \frac{1}{\lambda_i^2}, \forall i \in [k], \beta_i^* = \frac{(s-k)^2}{(\sum_{i\in[k+1,d]}\lambda_i)^2}, \forall i \in [k+1,r], \beta_i^* \geq \beta_r^*, \forall i \in [r+1,d].$$

(ii) For the maximization with respect to $\nu, \boldsymbol{\mu} \in \mathbb{R}^n_+$, we have

$$\max_{\nu,\boldsymbol{\mu}\in\mathbb{R}^n_+} \left\{-s\nu - \sum_{i\in[n]}\mu_i + \sum_{i\in[n]}x_i(\nu+\mu_i)\right\} = \begin{cases} 0, & \text{if } \sum_{i\in[n]} x_i = s, x_i \leq 1,\\ \infty, & \text{otherwise.}\end{cases}$$

Combining Parts (i) and (ii), we arrive at (21). □

A.15 Proof of Lemma 9

Lemma 9 *For any feasible solution* $\boldsymbol{x}$ *to A-PC* (21)*, let* $\boldsymbol{\lambda} \in \mathbb{R}^d_+$ *denote the vector of eigenvalues of matrix* $\sum_{i\in[n]} x_i\boldsymbol{v}_i\boldsymbol{v}_i^\top$*, then we have*

$$\Phi_s\left(\sum_{i\in[n]} x_i\boldsymbol{v}_i\boldsymbol{v}_i^\top\right) \geq \frac{E_{s-1}(\boldsymbol{\lambda})}{E_s(\boldsymbol{\lambda})}, \tag{23}$$

where function $E_s(\cdot)$ *is introduced in Definition 3.*

Proof. Without loss of generality, suppose that the eigenvalues of matrix $\sum_{i\in[n]} x_i\boldsymbol{v}_i\boldsymbol{v}_i^\top$ are sorted in a descending order, i.e., $\lambda_1 \geq \cdots \geq \lambda_d \geq 0$. Let us construct a new vector $\boldsymbol{\beta}$ as

$$\beta_i = \lambda_i, \forall i \in [k], \beta_i = \frac{\sum_{i\in[k+1,d]}\lambda_i}{s-k}, \forall i \in [k+1,s], \beta_i = 0, \forall i \in [s+1,d].$$

For any two vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$, we say that $\boldsymbol{x}$ is majorized by $\boldsymbol{y}$ if

$$\sum_{i\in[t]} x_i \leq \sum_{i\in[t]} y_i, \forall t \in [d-1], \sum_{i\in[d]} x_i = \sum_{i\in[d]} y_i.$$

Further, a function f is Schur-convex if $f(\boldsymbol{x}) \leq f(\boldsymbol{y})$ holds for any $\boldsymbol{x}, \boldsymbol{y} \in dom(f)$ that $\boldsymbol{x}$ is majorized by $\boldsymbol{y}$ (see, e.g., Hwang and Rothblum 1993).

Clearly, $\boldsymbol{\lambda}$ is majorized by $\boldsymbol{\beta}$ and thus obtain

$$\Phi_s\left(\sum_{i\in[n]} x_i\boldsymbol{v}_i\boldsymbol{v}_i^\top\right) = \frac{E_{s-1}(\boldsymbol{\beta})}{E_s(\boldsymbol{\beta})} \geq \frac{E_{s-1}(\boldsymbol{\lambda})}{E_s(\boldsymbol{\lambda})},$$

where the inequality follows from the Schur-convexity of function $\frac{E_{s-1}(\cdot)}{E_s(\cdot)}$ (see theorem 3.1 in Guruswami and Sinop 2012 and the fact $1/(f(\boldsymbol{x}))$ is Schur-convex if $f(\boldsymbol{x})$ is Schur-concave). □

A.16 Proof of Theorem 11

Theorem 11 *Given an optimal solution $\widehat{\boldsymbol{x}}$ to A-PC, the volume sampling Algorithm 6 yields a* $\min(s, n-s+1)$*-approximation ratio of A-MESP, i.e.,*

$$\mathbb{E}\left[\overset{s}{\operatorname{tr}}\left[\left(\sum_{i\in\tilde{S}} \boldsymbol{v}_i\boldsymbol{v}_i^\top\right)^\dagger\right]\right] \le \min(s, n-s+1) z_A^*.$$

Proof. For any positive semidefinite matrix $\boldsymbol{X}\succeq 0$, let $\boldsymbol{\lambda}(\boldsymbol{X})$ denote the vector of its eigenvalues.

The expected objective value output from Algorithm 6 can be upper bounded by

$$\begin{aligned}
\mathbb{E}\left[\overset{s}{\operatorname{tr}}\left[\left(\sum_{i\in\tilde{S}} \boldsymbol{v}_i\boldsymbol{v}_i^\top\right)^\dagger\right]\right] &= \sum_{S\in\binom{[n]}{s}} \mathbb{P}[\tilde{S}=S]\overset{s}{\operatorname{tr}}\left[\left(\boldsymbol{V}_S\boldsymbol{V}_S^\top\right)^\dagger\right]\\
&= \sum_{S\in\binom{[n]}{s}} \frac{\prod_{i\in S}\widehat{x}_i \overset{s}{\det}(\boldsymbol{V}_S\boldsymbol{V}_S^\top)}{\sum_{\bar{S}\in\binom{[n]}{s}}\prod_{i\in\bar{S}}\widehat{x}_i\overset{s}{\det}(\boldsymbol{V}_{\bar{S}}\boldsymbol{V}_{\bar{S}}^\top)} \frac{E_{s-1}(\boldsymbol{\lambda}(\boldsymbol{V}_S\boldsymbol{V}_S^\top))}{\overset{s}{\det}(\boldsymbol{V}_S\boldsymbol{V}_S^\top)}\\
&= \frac{\sum_{S\in\binom{[n]}{s}}\prod_{i\in S}\widehat{x}_i\sum_{T\in\binom{S}{s-1}} E_{s-1}(\boldsymbol{\lambda}(\boldsymbol{V}_T\boldsymbol{V}_T^\top))}{\sum_{\bar{S}\in\binom{[n]}{s}}\prod_{i\in\bar{S}}\widehat{x}_i\overset{s}{\det}(\boldsymbol{V}_{\bar{S}}\boldsymbol{V}_{\bar{S}}^\top)}\\
&= \frac{\sum_{T\in\binom{[n]}{s-1}}\sum_{S\in\binom{[n]}{s},T\subseteq S}\prod_{i\in S}\widehat{x}_i E_{s-1}(\boldsymbol{\lambda}(\boldsymbol{V}_T\boldsymbol{V}_T^\top))}{\sum_{\bar{S}\in\binom{[n]}{s}}\prod_{i\in\bar{S}}\widehat{x}_i\overset{s}{\det}(\boldsymbol{V}_{\bar{S}}\boldsymbol{V}_{\bar{S}}^\top)}\\
&= \frac{\sum_{T\in\binom{[n]}{s-1}}(\sum_{i\in[n]\setminus T}\widehat{x}_i)\prod_{i\in T}\widehat{x}_i E_{s-1}(\boldsymbol{\lambda}(\boldsymbol{V}_T\boldsymbol{V}_T^\top))}{\sum_{\bar{S}\in\binom{[n]}{s}}\prod_{i\in\bar{S}}\widehat{x}_i\overset{s}{\det}(\boldsymbol{V}_{\bar{S}}\boldsymbol{V}_{\bar{S}}^\top)}\\
&\le \min(s,n-s+1)\frac{\sum_{T\in\binom{[n]}{s-1}}\prod_{i\in T}\widehat{x}_i E_{s-1}(\boldsymbol{\lambda}(\boldsymbol{V}_T\boldsymbol{V}_T^\top))}{\sum_{\bar{S}\in\binom{[n]}{s}}\prod_{i\in\bar{S}}\widehat{x}_i E_s(\boldsymbol{\lambda}(\boldsymbol{V}_{\bar{S}}\boldsymbol{V}_{\bar{S}}^\top))}\\
&= \min(s,n-s+1)\frac{E_{s-1}(\boldsymbol{\lambda}(\sum_{i\in[n]}\widehat{x}_i\boldsymbol{v}_i\boldsymbol{v}_i^\top))}{E_s(\boldsymbol{\lambda}(\sum_{i\in[n]}\widehat{x}_i\boldsymbol{v}_i\boldsymbol{v}_i^\top))}\\
&\le \min(s,n-s+1)\Phi_s\left(\sum_{i\in[n]}\widehat{x}_i\boldsymbol{v}_i\boldsymbol{v}_i^\top\right) \le \min(s,n-s+1)z_A^*
\end{aligned}$$

where the third equality is due to Cauchy-Binet formula (Broida and Williamson 1989), the fourth and fifth equalities are due to interchange of summations and collecting terms, the first inequality stems from the fact that $\sum_{i\in[n]\setminus T}\widehat{x}_i \le \min(s, n-s+1)$ for any size-$(s-1)$ subset T, the sixth equality is due to Cauchy-Binet formula (Broida and Williamson 1989), the second inequality is from Lemma 9 and the last inequality results from the weak duality. □

A.17 Proof of Lemma 10

Lemma 10 *Suppose that $\widehat{S}$ is the output of the local search Algorithm 7 and $\boldsymbol{X}=\sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top$, for each pair $(i,j)\in\widehat{S}\times([n]\setminus\widehat{S})$, the following inequality always holds*

$$\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^3\boldsymbol{v}_i\boldsymbol{v}_j^\top(\boldsymbol{I}_n-\boldsymbol{X}^\dagger\boldsymbol{X})\boldsymbol{v}_j \le \boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_i + \boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_i\boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_j - 2\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_j\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{v}_j.$$

Proof. Similar to the analysis of Lemma 7, for each pair (i,j), there are two cases to be considered, conditional on whether $\boldsymbol{v}_j \in \operatorname{col}(\boldsymbol{X}_{-i})$ or not. If the rank of $\boldsymbol{X}$ is s, then $\overset{s}{\operatorname{tr}}(\boldsymbol{X}^\dagger) = \operatorname{tr}(\boldsymbol{X}^\dagger)$, thus for notational convenience, we will use $\operatorname{tr}(\cdot)$ instead.

(i) If $\boldsymbol{v}_j \notin \operatorname{col}(\boldsymbol{X}_{-i})$, according to the local optimality condition, we have

$$\begin{aligned}\operatorname{tr}(\boldsymbol{X}^\dagger) &\le \operatorname{tr}[(\boldsymbol{X}_{-i} + \boldsymbol{v}_j\boldsymbol{v}_j^\top)^\dagger] = \operatorname{tr}(\boldsymbol{X}_{-i}^\dagger) + \frac{1+\boldsymbol{v}_j^\top \boldsymbol{X}_{-i}^\dagger \boldsymbol{v}_j}{\boldsymbol{v}_j^\top(\boldsymbol{I}_n - \boldsymbol{X}_{-i}\boldsymbol{X}_{-i}^\dagger)\boldsymbol{v}_j} \\ &= \operatorname{tr}(\boldsymbol{X}^\dagger) - \frac{\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^3\boldsymbol{v}_i}{\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_i} + \frac{1+\boldsymbol{v}_j^\top \boldsymbol{X}_{-i}^\dagger \boldsymbol{v}_j}{\boldsymbol{v}_j^\top(\boldsymbol{I}_n - \boldsymbol{X}_{-i}\boldsymbol{X}_{-i}^\dagger)\boldsymbol{v}_j} \\ &= \operatorname{tr}(\boldsymbol{X}^\dagger) - \frac{\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^3\boldsymbol{v}_i}{\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_i} + \frac{1+\boldsymbol{v}_j^\top \boldsymbol{X}_{-i}^\dagger \boldsymbol{v}_j}{\boldsymbol{v}_j^\top(\boldsymbol{I}_n - \boldsymbol{X}\boldsymbol{X}^\dagger)\boldsymbol{v}_j + (\boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_i)^2/\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_i}, \end{aligned} \tag{44}$$

where the equalities follow from Part (iii), Part (iv) and Part (viii) in Lemma 6, respectively.

Then by Part (iv) in Lemma 6, we further have

$$\boldsymbol{v}_j^\top \boldsymbol{X}_{-i}^\dagger \boldsymbol{v}_j = \boldsymbol{v}_j^\top \boldsymbol{X}^\dagger \boldsymbol{v}_j - 2\frac{\boldsymbol{v}_j^\top \boldsymbol{X}^\dagger \boldsymbol{v}_i \boldsymbol{v}_j^\top (\boldsymbol{X}^\dagger)^2 \boldsymbol{v}_i}{\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_i} + \frac{\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^3\boldsymbol{v}_i(\boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_i)^2}{(\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_i)^2}.$$

Plugging the equation above into the local optimality condition (44), we can simplify it as

$$\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^3\boldsymbol{v}_i\boldsymbol{v}_j^\top(\boldsymbol{I}_n - \boldsymbol{X}^\dagger\boldsymbol{X})\boldsymbol{v}_j \le \boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_i + \boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_i\boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_j - 2\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_j\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{v}_j.$$

(ii) If $\boldsymbol{v}_j \in \operatorname{col}(\boldsymbol{X}_{-i})$, we show that $\boldsymbol{v}_j^\top(\boldsymbol{I}_n - \boldsymbol{X}^\dagger\boldsymbol{X})\boldsymbol{v}_j = 0$ and $\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{v}_j = 0$ for each $i \in \widehat{S}$ in the proof of Lemma 7. Thus, it is clear that

$$\begin{aligned} 0 = \boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^3\boldsymbol{v}_i\boldsymbol{v}_j^\top(\boldsymbol{I}_n - \boldsymbol{X}^\dagger\boldsymbol{X})\boldsymbol{v}_j &\le \boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_i + \boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_i\boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_j \\ &= \boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_i + \boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_i\boldsymbol{v}_j^\top\boldsymbol{X}^\dagger\boldsymbol{v}_j - 2\boldsymbol{v}_i^\top(\boldsymbol{X}^\dagger)^2\boldsymbol{v}_j\boldsymbol{v}_i^\top\boldsymbol{X}^\dagger\boldsymbol{v}_j. \end{aligned}$$

□

A.18 Proof of Theorem 12

Theorem 12 *The local search Algorithm 7 yields a* $s/2+\delta^{-1}\min\{\lambda_{\max}(\boldsymbol{C}), n\delta+(n-s)\lambda_{\max}(\boldsymbol{C})\}$*-approximation ratio for A-MESP, i.e,*

$$\overset{s}{\operatorname{tr}}\left(\sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top\right) \le \min\left\{\frac{s}{2}\left(1+\frac{\lambda_{\max}(\boldsymbol{C})}{\delta}\right), \frac{1}{2}\left(n+s+\frac{(n-s)\lambda_{\max}(\boldsymbol{C})}{\delta}\right)\right\} z_A^*,$$

where $\widehat{S}$ *is the set produced by Algorithm 7, and* δ *is defined in Lemma 4.*

Proof. Let us denote $\boldsymbol{X} = \sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top$. Clearly, the rank of $\boldsymbol{X}$ is s and suppose that its eigenvalues satisfy $\lambda_1 \ge \cdots \ge \lambda_s > \lambda_{s+1} = \cdots = \lambda_d = 0$. Thus, $\overset{s}{\operatorname{tr}}(\boldsymbol{X}^\dagger) = \sum_{i\in[s]}\frac{1}{\lambda_i} = \operatorname{tr}(\boldsymbol{X}^\dagger)$. If the rank of an $n\times n$ positive semi-definite matrix $\boldsymbol{Y}$ is s, since $\overset{s}{\operatorname{tr}}(\boldsymbol{Y}) = \operatorname{tr}(\boldsymbol{Y})$, thus for notational convenience, we will use $\operatorname{tr}(\cdot)$ instead.

Similar to the proof in Theorem 7, our proof relies on the weak duality of A-LD (20). Consider a feasible variable $\boldsymbol{\Lambda}$ of A-LD (20) as

$$\boldsymbol{\Lambda} = 2t^2(\boldsymbol{X}^\dagger)^2 + 2t^2\lambda_s^{-2}(\boldsymbol{I}_n - \boldsymbol{X}^\dagger\boldsymbol{X}),$$

where $t > 0$ is a scaling factor and will be specified later. Next, to construct the solution of the other two dual variables $(\nu, \boldsymbol{\mu})$, we need to check the feasibility of constraints in A-LD (20), i.e.,

$$\boldsymbol{v}_i^\top \boldsymbol{\Lambda} \boldsymbol{v}_i \leq \nu + \mu_i, \forall i \in [n].$$

There are two cases to be considered: (i) for each $i \in \widehat{S}$ and (ii) for each $j \in [n] \setminus \widehat{S}$.

(i) For each $i \in \widehat{S}$, we have

$$\boldsymbol{v}_i^\top \boldsymbol{\Lambda} \boldsymbol{v}_i = 2t^2 \boldsymbol{v}_i^\top (\boldsymbol{X}^\dagger)^2 \boldsymbol{v}_i \leq 2t^2 \operatorname{tr}(\boldsymbol{X}^\dagger), \tag{45}$$

where the equation is due to Part (vi) in Lemma 6 and the inequality is from $\sum_{i\in\widehat{S}} \boldsymbol{v}_i^\top (\boldsymbol{X}^\dagger)^2 \boldsymbol{v}_i = \operatorname{tr}(\boldsymbol{X}^\dagger)$.

(ii) For each $j \in [n] \setminus \widehat{S}$, according to Lemma 10, for each $i \in \widehat{S}$, we have

$$\boldsymbol{v}_i^\top (\boldsymbol{X}^\dagger)^3 \boldsymbol{v}_i \boldsymbol{v}_j^\top (\boldsymbol{I}_n - \boldsymbol{X}^\dagger \boldsymbol{X}) \boldsymbol{v}_j \leq \boldsymbol{v}_i^\top (\boldsymbol{X}^\dagger)^2 \boldsymbol{v}_i + \boldsymbol{v}_i^\top (\boldsymbol{X}^\dagger)^2 \boldsymbol{v}_i \boldsymbol{v}_j^\top \boldsymbol{X}^\dagger \boldsymbol{v}_j - 2\boldsymbol{v}_i^\top (\boldsymbol{X}^\dagger)^2 \boldsymbol{v}_j \boldsymbol{v}_i^\top \boldsymbol{X}^\dagger \boldsymbol{v}_j.$$

Summing up the above inequality over all $i \in \widehat{S}$, we can obtain

$$\begin{aligned}\frac{1}{t^2}\boldsymbol{v}_j^\top \boldsymbol{\Lambda} \boldsymbol{v}_j &\leq \lambda_s^{-2} \boldsymbol{v}_j^\top (\boldsymbol{I}_d + \boldsymbol{X}^\dagger \boldsymbol{X}) \boldsymbol{v}_j + \operatorname{tr}[(\boldsymbol{X}^\dagger)^2] \boldsymbol{v}_j^\top (\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X}) \boldsymbol{v}_j + 2\boldsymbol{v}_j^\top (\boldsymbol{X}^\dagger)^2 \boldsymbol{v}_j \\ &\leq \operatorname{tr}(\boldsymbol{X}^\dagger) + \operatorname{tr}(\boldsymbol{X}^\dagger) \boldsymbol{v}_j^\top \boldsymbol{X}^\dagger \boldsymbol{v}_j + \lambda_s^{-2} \boldsymbol{v}_j^\top (\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X}) \boldsymbol{v}_j,\end{aligned}$$

where the first inequality is due to $\operatorname{tr}[(\boldsymbol{X}^\dagger)^2] \geq \lambda_s^{-2}$. Above, we can further bound the right-hand side as below

$$\begin{aligned}\frac{1}{t^2}\boldsymbol{v}_j^\top \boldsymbol{\Lambda} \boldsymbol{v}_j &\leq \operatorname{tr}(\boldsymbol{X}^\dagger) + \operatorname{tr}(\boldsymbol{X}^\dagger) \boldsymbol{v}_j^\top \boldsymbol{X}^\dagger \boldsymbol{v}_j + \lambda_s^{-2} \boldsymbol{v}_j^\top (\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X}) \boldsymbol{v}_j \\ &\leq \operatorname{tr}(\boldsymbol{X}^\dagger) + \lambda_s^{-1} \operatorname{tr}(\boldsymbol{X}^\dagger) \boldsymbol{v}_j^\top \boldsymbol{X}^\dagger \boldsymbol{X} \boldsymbol{v}_j + \lambda_s^{-1} \operatorname{tr}(\boldsymbol{X}^\dagger) \boldsymbol{v}_j^\top (\boldsymbol{I}_d - \boldsymbol{X}^\dagger \boldsymbol{X}) \boldsymbol{v}_j \\ &= \operatorname{tr}(\boldsymbol{X}^\dagger)(1 + \lambda_s^{-1} \boldsymbol{v}_j^\top \boldsymbol{v}_j) \leq \operatorname{tr}(\boldsymbol{X}^\dagger)(1 + \frac{\lambda_{\max}(\boldsymbol{C})}{\delta}),\end{aligned}$$

where the second inequality is because $\operatorname{tr}[(\boldsymbol{X}^\dagger)] \geq \lambda_s^{-1}$ and $\boldsymbol{X}^\dagger \succeq \lambda_s^{-1} \boldsymbol{X}^\dagger \boldsymbol{X}$, and the third inequality is due to the facts that $\boldsymbol{v}_\ell^\top \boldsymbol{v}_\ell \leq \lambda_{\max}(\boldsymbol{C})$ for any $\ell \in [n]$ and $\lambda_s \geq \delta$.

Thus, for each $j \in [n] \setminus \widehat{S}$, we must have

$$\boldsymbol{v}_j^\top \boldsymbol{\Lambda} \boldsymbol{v}_j \leq t^2 \operatorname{tr}(\boldsymbol{X}^\dagger)(1 + \frac{\lambda_{\max}(\boldsymbol{C})}{\delta}). \tag{46}$$

Using inequalities (45) and (46) to construct $(\nu, \boldsymbol{\mu})$, it suffices to solve the optimization problem

$$z_A^{LD} \geq \max_{t>0} \max_{\nu, \boldsymbol{\mu} \in \mathbb{R}_+^n} \left\{ t2\sqrt{2}\overset{s}{\mathrm{tr}}(\boldsymbol{X}^\dagger) - s\nu - \sum_{i\in[n]} \mu_i : \nu + \mu_i \geq 2t^2 \mathrm{tr}(\boldsymbol{X}^\dagger), \forall i \in \widehat{S}, \right.$$
$$\left. \nu + \mu_i \geq t^2 \mathrm{tr}(\boldsymbol{X}^\dagger)\left(1 + \frac{\lambda_{\max}(\boldsymbol{C})}{\delta}\right), \forall i \in [n] \setminus \widehat{S} \right\}.$$

Above, by checking the primal and dual of inner maximization problems, there are following two candidate optimal solutions:

$$\nu^a = t^2 \mathrm{tr}(\boldsymbol{X}^\dagger)\left(1 + \frac{\lambda_{\max}(\boldsymbol{C})}{\delta}\right), \mu_i^a = 0, \forall i \in [n],$$
$$\nu^b = 2t^2 \mathrm{tr}(\boldsymbol{X}^\dagger), \mu_i^b = 0, \forall i \in \widehat{S}, \mu_i^b = t^2 \mathrm{tr}(\boldsymbol{X}^\dagger)\left(\frac{\lambda_{\max}(\boldsymbol{C})}{\delta} - 1\right), \forall i \in [n] \setminus \widehat{S}.$$

Plugging in these two solutions, the above maximization problem becomes

$$z_A^{LD} \geq \mathrm{tr}(\boldsymbol{X}^\dagger) \max_{t>0} \max \left\{ 2\sqrt{2}t - s(1 + \frac{\lambda_{\max}(\boldsymbol{C})}{\delta})t^2, 2\sqrt{2}t - \left(2s + (n-s)\left(\frac{\lambda_{\max}(\boldsymbol{C})}{\delta} - 1\right)\right)t^2 \right\}.$$

By swapping the two maximization operators and optimizing over t, the right-hand side of above inequality is further equivalent to

$$z_A^{LD} \geq \mathrm{tr}(\boldsymbol{X}^\dagger) \max \left\{ \frac{2}{s(1 + \lambda_{\max}(\boldsymbol{C})/\delta)}, \frac{2}{n + s + (n-s)\lambda_{\max}(\boldsymbol{C})/\delta} \right\}.$$

Using the fact that $z_A^{LD} \leq z_A^*$, we obtain the desired approximation ratio. □

Appendix B. MISOCP Formulation of MESP

In this section, we develop a mixed integer second-order conic programming (MISOCP) formulation for MESP, which is equivalent to the nonlinear convex integer program studied by Anstreicher (2018a). The formulation from Anstreicher (2018a) has the following form

$$\text{(MESP)}\ z^* := \max_{\boldsymbol{x}} \left\{ \frac{1}{2} \log\det\left(\gamma \boldsymbol{C}\, \mathrm{Diag}(\boldsymbol{x}) \boldsymbol{C} + \boldsymbol{I}_n - \mathrm{Diag}(\boldsymbol{x})\right) - \frac{1}{2} s \log(\gamma) : \sum_{i\in[n]} x_i = s, \boldsymbol{x} \in [0,1]^n \right\}, \tag{47}$$

where γ is a positive scalar and can be arbitrary. In fact, a good choice of γ can improve the continuous relaxation of formulation (47). According to table 1 in Sagnol et al. (2015), we can show that the above formulation (47) is equivalent to

$$
\begin{aligned}
\text{(MESP)} \quad z^* := \max_{\boldsymbol{x},\boldsymbol{Z}_1,\cdots,\boldsymbol{Z}_{2n}\boldsymbol{t},\boldsymbol{J}} \quad & \frac{n}{2}\log\left(\prod_{j=1}^{n}(\boldsymbol{J}_{j,j})^{1/n}\right) - \frac{1}{2}s\log(\gamma) \\
\text{s.t.} \quad & \sum_{i\in[n]} \sqrt{\gamma}\boldsymbol{C}_i\boldsymbol{Z}_i + \sum_{k\in[n+1,2n]} \boldsymbol{e}_{k-n}\boldsymbol{Z}_k = \boldsymbol{J}, \\
& \boldsymbol{J} \text{ is lower trangular}, \\
& ||\boldsymbol{Z}_i\boldsymbol{e}_j||^2 \leq \boldsymbol{t}_{ij}x_i, \forall i\in[2n], \forall j\in[n], \\
& \sum_{i\in[2n]} \boldsymbol{t}_{ij} \leq \boldsymbol{J}_{j,j}, \forall j\in[n], \\
& \boldsymbol{t}_{i,j} \geq 0, \forall i\in[2n], \forall j\in[n], \\
& 1 - x_i = x_{n+i}, \forall i\in[n], \\
& \sum_{i\in[n]} x_i = s, \\
& \boldsymbol{x}\in\{0,1\}^n,
\end{aligned}
\tag{48}
$$

where $\boldsymbol{C}_i$ denotes the i-th column vector of matrix $\boldsymbol{C}$. Note that according to chapter 2.3 in Ben-Tal and Nemirovski (2001), we can equivalently represent the objective function in the formulation (48) as a second order conic program.