D-optimal Data Fusion:
Exact and Approximation Algorithms

Yongchun Li

School of ISyE, Georgia Institute of Technology, Atlanta, GA, USA, ycli@gatech.edu

Marcia Fampa

COPPE, Universidade Federal do Rio de Janeiro, Brasil, fampa@cos.ufrj.br

Jon Lee

IOE Department, University of Michigan, Ann Arbor, MI, USA, jonxlee@umich.edu

Feng Qiu

Energy Systems Division, Argonne National Laboratory, Lemont, IL, USA, fqiu@anl.gov

Weijun Xie

School of ISyE, Georgia Institute of Technology, Atlanta, GA, USA, wxie@gatech.edu

Rui Yao

Energy Systems Division, Argonne National Laboratory, Lemont, IL, USA, ruiyao@ieee.org

We study the D-optimal Data Fusion (DDF) problem, which aims to select new data points, given an existing Fisher information matrix, so as to maximize the logarithm of the determinant of the overall Fisher information matrix. We show that the DDF problem is NP-hard and has no constant-factor polynomial-time approximation algorithm unless P = NP. Therefore, to solve the DDF problem effectively, we propose two convex integer-programming formulations and investigate their corresponding complementary and Lagrangian-dual problems. Leveraging the concavity of the objective functions in the two proposed convex integer-programming formulations, we design an exact algorithm, aimed at solving the DDF problem to optimality. We further derive a family of submodular valid inequalities and optimality cuts, which can significantly enhance the algorithm performance. We also develop scalable randomized-sampling and local-search algorithms with provable performance guarantees. Finally, we test our algorithms using real-world data on the new phasor-measurement-units placement problem for modern power grids, considering the existing conventional sensors. Our numerical study demonstrates the efficiency of our exact algorithm and the scalability and high-quality outputs of our approximation algorithms.

Key words: Data fusion, Fisher information matrix, D-optimality, maximum-entropy sampling, exact algorithm, approximation algorithm, submodular inequality, optimality cut

1. Introduction

We study the following D-optimal Data Fusion problem

$$z^* := \max_{S \subseteq [n], |S| = s} \mathrm{ldet}\left(\boldsymbol{C} + \sum_{i \in S} \boldsymbol{a}_i \boldsymbol{a}_i^\top\right), \tag{DDF}$$

where ldet denotes the natural logarithm of the determinant, the positive-definite matrix $\boldsymbol{C} \in \mathbb{S}_{++}^d$ is an existing Fisher Information Matrix (FIM), the columns $\{\boldsymbol{a}_i \in \mathbb{R}^d\}_{i \in [n]}$ of $\boldsymbol{A} \in \mathbb{R}^{d \times n}$ represent n candidate d-dimensional data points to be selected, and the positive integer $s \in [n] := \{1, 2, \ldots, n\}$ denotes the number of data points to be selected. In DDF, the FIM comprises two parts: $\boldsymbol{C}$ and $\sum_{i \in S} \boldsymbol{a}_i \boldsymbol{a}_i^\top$, corresponding to the information obtained from existing data and new selected points, respectively. Therefore, the goal of DDF is to maximize the information gain by integrating new data points with conventional data from other sources. It is worth remarking that DDF differs

1

from the classic D-optimal design problem in the following aspects: (i) the conventional D-optimal design problem does not have the existing FIM C; (ii) the D-optimal design problem typically assumes that $s \geq d$, while in DDF, it is possible that $s < d$. Hence, the existing results for D-optimal design in [SX20, MSTX19, NST19, PFL22] do not directly apply to DDF.

Below, we describe some interesting examples for which the proposed DDF can be applicable.

- *Sensor Fusion*: Modern sensor networks often involve multi-type sensors working collectively for a specific monitoring task (see [Var12, Zha95]). When installing new (possibly high-end) sensors, in order to achieve economical and effective operation, it is desirable to maximize the overall information obtained by fusing a small number of new sensors with the existing ones. Using the D-optimality criterion, widely-used in sensor placement, the corresponding optimal sensor-fusion problem is equivalent to DDF. In particular, $C \in \mathbb{S}_{++}^n$ represents the FIM of the existing sensors installed at an n-node network. The additional FIM components introduced by new sensors can be shown to be equivalent to adding their corresponding rank-one matrices into the existing FIM (see, e.g., [LNI11, KG11, YKBB13]). In this problem, the dimension d of sensor measurements is equal to the number of sensor locations, i.e., $d = n$, and we must have $s \leq d = n$. We test a sensor fusion problem in power systems in our numerical study section.

- *Active Learning*: In the big-data era, there are much more unlabeled data than labeled ones, where the latter are often expensive to acquire. Recently, researchers have been proactively working on active learning to select a small subset of unlabeled data points to label, with the hope of achieving a desired learning goal (e.g., classification accuracy) using fewer training data and/or achieving smaller labeling costs. Active learning enables the learners to iteratively select the most informative data points by exploiting labeled ones and exploring the unlabeled ones, which is also known as sequential experimental design in statistics. Thus, given that the FIM of the existing labeled data points is positive-definite, the D-optimal new-data selection problem can be formulated in the form of DDF (see [He09, LC21, MAL19, YBT06]). At each active learning iteration, the number of selected unlabeled data points is often much smaller than the dimension, especially for high-dimensional data, and can even be set to one for the sake of computational efficiency and stable convergence (see [WP17]).

- *Regularized D-optimal Design*: The regularized experimental design that arises from the linear models with a regularization penalty has been recently studied in [Tan20, MAL19], which admits the same form as our DDF. For regularized D-optimal design, its FIM often involves an additional positive-definite matrix C (e.g., an identity matrix in ridge-regularized D-optimal design).

1.1. Maximum-Entropy Sampling Problem (MESP)

DDF is closely related to the maximum-entropy sampling problem

$$\max_{S \subseteq [n], |S| = s} \operatorname{ldet} \widehat{C}_{S,S}, \tag{MESP}$$

where $\widehat{C} \in \mathbb{S}_+^n$ has rank at least s, and $\widehat{C}_{S,S}$ denotes the principal submatrix of $\widehat{C}$ indexed by S.

Since [SW87], MESP has seen a variety of applications in statistics and information theory, as well as algorithmic advances (see [FL22a]). Below, we demonstrate that MESP is a special case of DDF with $n = d$ when the covariance matrix of MESP is positive-definite. This result implies that DDF is NP-hard (see [KLQ95]) and cannot be approximated by any polynomial-time algorithm within $\Omega(n \log(c))$ with some constant $c > 1$ unless $P = NP$ according to [CMI13]. Therefore, to efficiently solve DDF, this paper focuses on developing exact and near-optimal approximation algorithms.

In fact, as we will see below, DDF is also a special case of MESP.

THEOREM 1. *For a covariance matrix $\widehat{C} \in \mathbb{S}_{++}^n$, MESP can be reduced to a special DDF with $n = d$.*

Proof. Let $\boldsymbol{F}$ be the Cholesky factor of the positive semidefinite matrix $(\widehat{\boldsymbol{C}}/\lambda_{\min}(\widehat{\boldsymbol{C}}) - \boldsymbol{I}_n)$. Then, $\widehat{\boldsymbol{C}}/\lambda_{\min}(\widehat{\boldsymbol{C}}) = \boldsymbol{I}_n + \boldsymbol{F}^\top \boldsymbol{F}$, and for any $S \subseteq [n]$, $|S| = s$, we can verify that

$$
\begin{aligned}
\operatorname{ldet} \widehat{\boldsymbol{C}}_{S,S} &= \operatorname{ldet}\left((\widehat{\boldsymbol{C}}/\lambda_{\min}(\widehat{\boldsymbol{C}}))_{S,S}\right) - s\log(1/\lambda_{\min}(\widehat{\boldsymbol{C}})) \\
&= \operatorname{ldet}\left((\boldsymbol{I}_n + \boldsymbol{F}^\top \boldsymbol{F})_{S,S}\right) - s\log(1/\lambda_{\min}(\widehat{\boldsymbol{C}})) \\
&= \operatorname{ldet}\left(\boldsymbol{I}_n + \sum_{i=1}^n x_i \boldsymbol{f}_i \boldsymbol{f}_i^\top\right) - s\log(1/\lambda_{\min}(\widehat{\boldsymbol{C}})),
\end{aligned}
\tag{1}
$$

where $\boldsymbol{f}_i \in \mathbb{R}^n$ is the i-th column of $\boldsymbol{F}$ and $\boldsymbol{x}$ is the 0/1 characteristic vector of $S \subseteq [n]$, i.e., $x_j = 1$, if $j \in S$; $x_j = 0$ if $j \in [n] \setminus S$. The result follows because the first term in (1) is the objective function of MESP for the covariance matrix $\widehat{\boldsymbol{C}}$, and the last term is the objective function of a particular DDF added to a constant. □

THEOREM 2. *For the matrix $\boldsymbol{C} \in \mathbb{S}_{++}^d$ and data points $\{\boldsymbol{a}_i \in \mathbb{R}^d\}_{i \in [n]}$, DDF can be reduced to MESP with an $n \times n$ positive-definite covariance matrix.*

Proof. Let $\boldsymbol{C}^{\frac{1}{2}}$ be the square root of $\boldsymbol{C}$ and $\boldsymbol{B} := [\boldsymbol{b}_1, \cdots, \boldsymbol{b}_n] \in \mathbb{R}^{d \times n}$, where $\boldsymbol{b}_i := \boldsymbol{C}^{-\frac{1}{2}} \boldsymbol{a}_i$, for $i \in [n]$. Then, for any $S \subseteq [n]$, $|S| = s$, we can verify that

$$
\begin{aligned}
\operatorname{ldet}\left(\boldsymbol{C} + \sum_{i \in S} \boldsymbol{a}_i \boldsymbol{a}_i^\top\right) &= \operatorname{ldet} \boldsymbol{C} + \operatorname{ldet}(\boldsymbol{I}_d + \sum_{i \in S} \boldsymbol{C}^{-\frac{1}{2}} \boldsymbol{a}_i \boldsymbol{a}_i^\top \boldsymbol{C}^{-\frac{1}{2}}) \\
&= \operatorname{ldet} \boldsymbol{C} + \operatorname{ldet}(\boldsymbol{I}_d + \sum_{i \in S} \boldsymbol{b}_i \boldsymbol{b}_i^\top) \\
&= \operatorname{ldet} \boldsymbol{C} + \operatorname{ldet}(\boldsymbol{I}_n + \boldsymbol{B}^\top \boldsymbol{B})_{S,S}.
\end{aligned}
\tag{2}
$$

The result follows because the first term in (2) is the objective function of DDF for the existing FIM $\boldsymbol{C}$, set $\{\boldsymbol{a}_i \in \mathbb{R}^d\}_{i \in [n]}$ of n candidate data points, and s number of data points to be selected; and the last term in (2) is the objective function of a particular MESP added to a constant. □

1.2. Relevant literature

In this subsection, we survey the relevant literature on exact and approximation algorithms for solving DDF or its variants including the regularized D-optimal design and MESP.

Exact Algorithms: As a special case of DDF, [He09] and [MAL19] studied the regularized D-optimal design with number of selected data being $s = 1$ and derived a closed-form optimal solution by using the formula for the determinant of a rank-one change to a symmetric matrix. A series of research works aimed at solving MESP, another special case of DDF, to optimality by a branch-and-bound algorithm have been conducted in [KLQ95, Lee98, AFLW96, AFLW99, LW03, HLW01, AL04, BL07, Ans18, Ans20, CFLL21, CFL23]. [CHB08, CHB13] and also [FL22b] instead considered outer-approximation approaches, and [LX23] developed a branch-and-cut algorithm based on (sub)gradient inequalities. In contrast to previous approaches, we propose a new algorithm by deriving (sub)gradient inequalities and submodular inequalities from two equivalent convex integer-programming formulations of DDF, and exploring probing techniques to obtain optimality cuts. The submodular inequalities and the optimality cuts are numerically demonstrated to significantly strengthen the exact algorithm proposed, which is an enhancement of the LP/NLP branch-and-bound algorithm from [QG92].

Approximation Algorithms: Besides exact algorithms, more scalable yet effective approximation algorithms have also attracted attention, such as greedy, local-search, and randomized-sampling algorithms. Although the objective function of DDF is submodular and non-decreasing, it can be negative. Thus, the existing performance guarantees (e.g., $(1 - 1/e)$-approximation ratio) for non-negative non-decreasing submodular maximization cannot be directly applied [NWF78]. On the other hand, it has been shown that the randomized-sampling and local-search algorithms can be successfully applied to the D-optimal design and MESP to generate provably near-optimal solutions [LX23, MSTX19, SX20, Nik15, NST19]. A generalized MESP under matroid constraints has

been studied in [NS16]. Motivated by these recent breakthroughs, we tailor these two approximations algorithms to DDF, and we develop efficient implementations and theoretical guarantees. In particular, our approximation bounds of the sampling and local-search algorithms are invariant with respect to $s \leftrightarrow n - s$, outperforming the state-of-the-art ones [Nik15, LX23] for MESP when $s > n/2$. The detailed comparison can be found in Section 4. We also numerically compare the proposed sampling and local-search algorithms with the greedy algorithm [LNI11, LCW+12] adopted in a real sensor fusion problem. That greedy algorithm searches for a new data point to maximize the objective value at each iteration; it re-calculates objective values and is thus computationally expensive in contrast to our efficiently-implemented algorithms.

1.3. Summary of the organization and contributions

(i) We derive two convex integer-programming (CIP) formulations of DDF, termed R-DDF and M-DDF. We also study the Lagrangian-dual of their continuous relaxations and establish their optimality gaps.

(ii) We develop complementary formulations of R-DDF and M-DDF, based on the fact that DDF can be interpreted as excluding $n - s$ least informative data points.

(iii) Exploring the two CIPs, we succeed to derive closed-form (sub)gradient and submodular based valid inequalities, which are used on an enhancement of the LP/NLP branch-and-bound (B&B) algorithm proposed in [QG92], for solving DDF. In [MFR20] a numerical comparison is presented between the main algorithms in the literature for convex mixed-integer nonlinear-programming (MINLP), and the LP/NLP B&B presents an excellent performance in the comparison.

(iv) We investigate probing techniques to derive optimality cuts to strengthen DDF from both primal and dual perspectives, where our probing schemes are effective and easy-to-implement, based on tight Lagrangian dual bounds and near-optimal approximation algorithms. Our numerical study confirms the effectiveness of the optimality cuts.

(v) Exploring the two CIPs and the Lagrangian-dual of their continuous relaxations and their complements, we establish the theoretical performance guarantees of proposed local-search and randomize-sampling algorithms for solving DDF, as displayed in Table 1. We remark that each CIP provides us a different analysis of approximation bounds and some bounds in Table 1 are invariant with respect to s and $n - s$, due to the complementary formulations.

(vi) The numerical experiments present a real-world DDF application, optimal sensor placement in power systems, which confirms that our exact algorithm along with optimality cuts is effective and that our approximation algorithms are scalable and near-optimal.

Table 1   Approximation bounds for the local-search and sampling algorithms

	R-DDF	M-DDF
Local-Search Algorithm 1	$d \log \left(1 + \frac{\bar{s}\sigma_{\max}^2}{d(1+\sigma_{\max})} \right)$	$\bar{s} \log(\bar{s})$
Sampling Algorithm 2	$n \log(x_{\min}) - (n-s)\log(1+\delta)$	$\bar{s} \log \left(\frac{\bar{s}}{n} \right) + \log \left(\binom{n}{\bar{s}} \right)$

$\bar{s} := \min\{s, n-s\}; \sigma_{\max} := \max_{i \in [n]} \boldsymbol{a}_i^\top \boldsymbol{C}^{-1} \boldsymbol{a}_i; \delta = \lambda_{\max}(\boldsymbol{A}^\top \boldsymbol{C}^{-1} \boldsymbol{A})$

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 we use corresponding non-bold letters (e.g., x_i) to denote their components. We let $\mathbb{R}_+^n$ denote the set of all n-dimensional nonnegative vectors. Given positive integers $s < n$, we let $[n] := \{1, 2, \cdots, n\}$, $[s, n] := \{s, s+1, \cdots, n\}$, and $\bar{s} := \min\{s, n-s\}$. Further, we let $\mathcal{Z}_s$ denote the collection of feasible solutions satisfying the cardinality constraint, i.e., $\mathcal{Z}_s := \{\boldsymbol{x} \in \{0,1\}^n : \sum_{i \in [n]} x_i = s\}$. We let $\mathbb{S}_+^n$ (resp., $\mathbb{S}_{++}^n$) denote the cone of $n \times n$ symmetric positive semidefinite (resp., definite) matrices. We let $\boldsymbol{C}^{\frac{1}{2}}$ denote the square root of matrix $\boldsymbol{C}$, i.e., $\boldsymbol{C}^{\frac{1}{2}} \boldsymbol{C}^{\frac{1}{2}} = \boldsymbol{C}$. We let $\boldsymbol{C}^\dagger$ denote the Moore-Penrose

pseudo-inverse of $\boldsymbol{C}$. We let $\boldsymbol{I}_n$ denote the $n \times n$ identity matrix, and we let $\boldsymbol{e}_i \in \mathbb{R}^n$ denote the i-th standard-unit vector. We let $\mathbf{1}$ denote a vector with all entries being 1. For $\boldsymbol{x} \in \mathbb{R}^n$, we denote its support by $\text{supp}(\boldsymbol{x}) \subseteq [n]$. We let $|S|$ denote the cardinality of a finte set S. Overloading the notation $\binom{s}{k}$ for the number of k-subsets of an s-element set, given a set S, we let $\binom{S}{k}$ denote the collection of all the cardinality-k subsets of S. Given an $m \times n$ matrix $\boldsymbol{X}$ and two subsets $S \subseteq [m]$, $T \subseteq [n]$, we let $\boldsymbol{X}_{S,T}$ denote the submatrix of $\boldsymbol{X}$ with rows and columns indexed by sets S, T, respectively, and we let $\boldsymbol{X}_S$ denote the submatrix of $\boldsymbol{X}$ with columns indexed by S. Given a symmetric matrix $\boldsymbol{X}$, we let $\lambda_{\min}(\boldsymbol{X}), \lambda_{\max}(\boldsymbol{X})$ denote the least and greatest eigenvalues of $\boldsymbol{X}$, respectively. For $\boldsymbol{X}, \boldsymbol{Y} \in \mathbb{R}^{n \times n}$, we let $\boldsymbol{X} \circ \boldsymbol{Y}$ denote their Hadamard product, and $\text{tr}\,\boldsymbol{X}$ denote the trace of $\boldsymbol{X}$. Additional notation is introduced as needed.

REMARK 1. Throughout the paper, considering the existing FIM $\boldsymbol{C}$, the n candidate d-dimensional data points $\{\boldsymbol{a}_i \in \mathbb{R}^d\}_{i \in [n]}$ to be selected in DDF, and $\boldsymbol{A} := [\boldsymbol{a}_1, \cdots, \boldsymbol{a}_n] \in \mathbb{R}^{d \times n}$, we use the definitions: $\boldsymbol{b}_i := \boldsymbol{C}^{-\frac{1}{2}} \boldsymbol{a}_i$, for $i \in [n]$, $\boldsymbol{B} := [\boldsymbol{b}_1, \cdots, \boldsymbol{b}_n] \in \mathbb{R}^{d \times n}$, $\boldsymbol{q}_i := (\boldsymbol{C} + \boldsymbol{A}\boldsymbol{A}^\top)^{-\frac{1}{2}} \boldsymbol{a}_i$, for $i \in [n]$; we let $\boldsymbol{V}^\top \boldsymbol{V}$ be the Cholesky factorization of $\boldsymbol{I}_n + \boldsymbol{B}^\top \boldsymbol{B}$, and let $\boldsymbol{v}_i$ be the i-th column of $\boldsymbol{V}$, for $i \in [n]$.

Organization: The remainder of the paper is organized as follows. In Section 2, we develop two CIP formulations for DDF, and their corresponding complementary problems and Lagrangian duals. In Section 3, we present our exact algorithmic approach with the introduction of valid submodular inequalities and optimality cuts. In Section 4, we develop and analyze two approximation algorithms. In Section 5, we present numerical results on a real-world application in power systems. Finally, Section 6 contains brief conclusions.

2. Two convex integer-programming formulations

Next, we present two convex integer-programming (CIP) formulations for DDF, termed R-DDF and M-DDF, as well as their complementary problems and the Lagrangian-dual of their continuous relaxations.

2.1. First CIP formulation: R-DDF

To formulate DDF as a mathematical program, we introduce the binary variable $x_i = 1$, if the i-th data point is selected, and 0 otherwise, for each $i \in [n]$. Our first formulation of DDF is as follows.

PROPOSITION 1. *DDF is equivalent to*

$$z^* := \text{ldet}\,\boldsymbol{C} + \max_{\boldsymbol{x} \in \mathcal{Z}_s} \left\{ \text{ldet}\left(\boldsymbol{I}_d + \sum_{i \in [n]} x_i \boldsymbol{b}_i \boldsymbol{b}_i^\top\right) \right\}. \tag{R-DDF}$$

Proof. Let $\boldsymbol{x}$ be the 0/1 characteristic vector of $S \subseteq [n]$. Then, we have

$$\det\left(\boldsymbol{C} + \textstyle\sum_{i \in S} \boldsymbol{a}_i \boldsymbol{a}_i^\top\right) = \det\left(\boldsymbol{C} + \textstyle\sum_{i \in [n]} x_i \boldsymbol{a}_i \boldsymbol{a}_i^\top\right) = \det(\boldsymbol{C}) \det\left(\boldsymbol{I}_d + \textstyle\sum_{i \in [n]} x_i \boldsymbol{b}_i \boldsymbol{b}_i^\top\right).$$

The result follows from the definition of $\boldsymbol{b}_i$ (see Remark 1) and by taking the logarithm on both sides of the above equation. $\square$

We refer to the problem as R-DDF because the objective function is similar to that of the regularized D-optimal design problem. The following result presents the Lagrangian dual of the continuous relaxation of R-DDF.

PROPOSITION 2. *The Lagrangian dual of the continuous relaxation of R-DDF is*

$$\widehat{z}_R := \text{ldet}\,\boldsymbol{C} + \min_{\substack{\boldsymbol{\Lambda} \in \mathbb{S}_{++}^d, \\ \nu, \boldsymbol{\mu} \in \mathbb{R}_+^n}} \left\{ -\text{ldet}(\boldsymbol{\Lambda}) + \text{tr}\,\boldsymbol{\Lambda} + s\nu + \sum_{i \in [n]} \mu_i - d : \boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i \leq \nu + \mu_i, \ i \in [n] \right\}. \tag{3}$$

Proof. See e-companion EC.1.1. □

We further derive the gradient, Hessian, and Lipschitz constant of the objective function of R-DDF, which allow us to use first- or second-order methods (e.g., Frank-Wolfe algorithm) to compute $\widehat{z}_R$ with a proven convergence rate. We demonstrate the following result.

PROPOSITION 3. *For any $\boldsymbol{x} \in [0,1]^n$, the gradient $\boldsymbol{g} \in \mathbb{R}^n$ and the Hessian $\boldsymbol{H} \in \mathbb{R}^{n \times n}$ of the objective function in the continuous relaxation of R-DDF are*

$$\boldsymbol{g}(\boldsymbol{x}) := [\boldsymbol{b}_1^\top \boldsymbol{X}^{-1} \boldsymbol{b}_1, \cdots, \boldsymbol{b}_n^\top \boldsymbol{X}^{-1} \boldsymbol{b}_n], \quad \boldsymbol{H}(\boldsymbol{x}) := -(\boldsymbol{B}^\top \boldsymbol{X}^{-1} \boldsymbol{B}) \circ (\boldsymbol{B}^\top \boldsymbol{X}^{-1} \boldsymbol{B}), \text{ and } \boldsymbol{H}(\boldsymbol{x}) \succeq -\delta^2 \boldsymbol{I}_n,$$

where $\boldsymbol{X} := \boldsymbol{I}_d + \sum_{i \in [n]} x_i \boldsymbol{b}_i \boldsymbol{b}_i^\top$ and $\delta := \lambda_{\max}(\boldsymbol{B}^\top \boldsymbol{B})$.

Proof. See e-companion EC.1.2. □

We adopt the well-known Frank-Wolfe algorithm to compute $\widehat{z}_R$ in the numerical study. Due to Proposition 3 and [LX23, Theorem 4], the Frank-Wolfe algorithm admits a convergence rate of $O(\bar{s}^2 \lambda_{\max}^2(\boldsymbol{B}^\top \boldsymbol{B})/\kappa)$, where κ denotes the number of iterations.

An alternative interpretation of DDF is via excluding $n - s$ ineffective data points, which leads to the complementary formulation of R-DDF and the resulting Lagrangian dual problem below.

PROPOSITION 4. *R-DDF is equivalent to*

$$z^* = \operatorname{ldet}(\boldsymbol{C} + \boldsymbol{A}\boldsymbol{A}^\top) + \max_{\boldsymbol{x} \in \mathcal{Z}_{n-s}} \left\{ \operatorname{ldet}\left(\boldsymbol{I}_d - \sum_{i \in [n]} x_i \boldsymbol{q}_i \boldsymbol{q}_i^\top\right) \right\}, \qquad \text{(R-DDF-comp)}$$

and the Lagrangian dual of the continuous relaxation of R-DDF-comp is

$$\widehat{z}_R = \operatorname{ldet}(\boldsymbol{C} + \boldsymbol{A}\boldsymbol{A}^\top) + \min_{\substack{\boldsymbol{\Lambda} \in \mathbb{S}_{++}^d, \\ \nu, \boldsymbol{\mu} \in \mathbb{R}_+^n}} \left\{ -\operatorname{ldet}\boldsymbol{\Lambda} + \operatorname{tr}\boldsymbol{\Lambda} + (n-s)\nu + \sum_{i \in [n]} \mu_i - d : -\boldsymbol{q}_i^\top \boldsymbol{\Lambda} \boldsymbol{q}_i \leq \nu + \mu_i, \ i \in [n] \right\}. \quad (4)$$

Proof. For any $\boldsymbol{x} \in \mathcal{Z}_s$, the objective function of R-DDF can be written as

$$\operatorname{ldet}\left(\boldsymbol{C} + \sum_{i \in [n]} x_i \boldsymbol{a}_i \boldsymbol{a}_i^\top\right) = \operatorname{ldet}\left(\boldsymbol{C} + \boldsymbol{A}\boldsymbol{A}^\top - \sum_{i \in [n]}(1 - x_i)\boldsymbol{a}_i \boldsymbol{a}_i^\top\right)$$

$$= \operatorname{ldet}(\boldsymbol{C} + \boldsymbol{A}\boldsymbol{A}^\top) + \operatorname{ldet}\left(\boldsymbol{I}_d - \sum_{i \in [n]}(1 - x_i)(\boldsymbol{C} + \boldsymbol{A}\boldsymbol{A}^\top)^{-\frac{1}{2}} \boldsymbol{a}_i \boldsymbol{a}_i^\top (\boldsymbol{C} + \boldsymbol{A}\boldsymbol{A}^\top)^{-\frac{1}{2}}\right).$$

Then, replacing variable $\boldsymbol{x}$ by $\boldsymbol{1} - \boldsymbol{x}$, and considering the definition of $\boldsymbol{q}_i$ (see Remark 1), we arrive at the equivalent formulation of R-DDF given by R-DDF-comp.

Following the similar derivation as Proposition 2, we can also derive the Lagrangian dual of the continuous relaxation of R-DDF-comp. □

We remark that (i) the greatest eigenvalue of $\sum_{i \in [n]} \boldsymbol{q}_i \boldsymbol{q}_i^\top$ is strictly less than one, so the matrix $\boldsymbol{I}_d - \sum_{\in [n]} x_i \boldsymbol{q}_i \boldsymbol{q}_i^\top$ in R-DDF-comp is always positive-definite and the objective value is finite; (ii) the continuous relaxation value of R-DDF does not vary with the complementary transformation, but considering the two Lagrangian dual problems (3) and (4) together, we obtain an approximation bound that is better than by considering only one of the Lagrangian duals, as discussed in Section 4.

2.2. Second CIP formulation: M-DDF

LEMMA 1. *[[Nik15], Lemma 13] Let $\boldsymbol{\lambda} \in \mathbb{R}_+^n$ with $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$, and let $0 < s \leq n$. There exists a unique integer k, with $0 \leq k < s$, such that*

$$\lambda_k > \frac{1}{s-k} \sum_{i \in [k+1,n]} \lambda_i \geq \lambda_{k+1}, \qquad (5)$$

with the convention $\lambda_0 = +\infty$.

LEMMA 2. *Let $\boldsymbol{\lambda} \in \mathbb{R}^n_+$ with $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_t > \lambda_{t+1} = \cdots = \lambda_s > \lambda_{s+1} = \cdots = \lambda_n = 0$. Then, the k satisfying* (5) *is precisely t.*

Proof. The result is immediate because

$$\frac{1}{s-t} \sum_{i \in [t+1,n]} \lambda_i = \lambda_{t+1}.$$

$\square$

DEFINITION 1 ([NIK15]). *For a matrix $\boldsymbol{X} \in \mathbb{S}^n_+$ with eigenvalues $\lambda_1 \geq \cdots \geq \lambda_n \geq 0$, we let*

$$f(\boldsymbol{X}) = \prod_{i \in [k]} \lambda_i \times \left(\frac{1}{s-k} \sum_{i \in [k+1,n]} \lambda_i \right)^{s-k},$$

where $k < s$ is unique non-negative integer satisfying (5).

It has been shown in [Nik15] that $f(\cdot)$ is a concave function. With the notation above, we are ready to define M-DDF.

THEOREM 3. *DDF is equivalent to the following CIP formulation*

$$z^* = \operatorname{ldet} \boldsymbol{C} + \max_{\boldsymbol{x} \in \mathcal{Z}_s} \left\{ \log f \left(\sum_{i \in [n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top \right) \right\}. \tag{M-DDF}$$

Proof. It is sufficient to prove that the objective functions of M-DDF and R-DDF are equal for any $\boldsymbol{x} \in \mathcal{Z}_s$. So, let $S := \operatorname{supp}(\boldsymbol{x})$ for an $\boldsymbol{x} \in \mathcal{Z}_s$. Clearly $|S| = s$, and thus we have

$$\det \left(\boldsymbol{I}_d + \sum_{i \in [n]} x_i \boldsymbol{b}_i \boldsymbol{b}_i^\top \right) = \det(\boldsymbol{I}_d + \boldsymbol{B}_S \boldsymbol{B}_S^\top) = \det(\boldsymbol{I}_s + \boldsymbol{B}_S^\top \boldsymbol{B}_S) = \det(\boldsymbol{I}_n + \boldsymbol{B}^\top \boldsymbol{B})_{S,S}$$

$$= \det(\boldsymbol{V}_S^\top \boldsymbol{V}_S) = f \left(\sum_{i \in [n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top \right),$$

where the second equation is because $\boldsymbol{B}_S \boldsymbol{B}_S^\top$ and $\boldsymbol{B}_S^\top \boldsymbol{B}_S = (\boldsymbol{B}^\top \boldsymbol{B})_{S,S}$ have the same non-zero eigenvalues, and the last one follows from the definition of $\boldsymbol{v}_i$ (see Remark 1) and from Lemma 2. This completes the proof. $\square$

The matrix in the objective of M-DDF is of order n, while that of R-DDF is of order d. We may consider choosing between them in practice, based on the two parameters n and d and on the continuous-relaxation bounds.

PROPOSITION 5. *[[LX23]] The Lagrangian dual of the continuous relaxation of M-DDF is*

$$\widehat{z}_M := \operatorname{ldet} \boldsymbol{C} + \min_{\substack{\boldsymbol{\Lambda} \in \mathbb{S}^n_{++}, \\ \nu, \boldsymbol{\mu} \in \mathbb{R}^n_+}} \left\{ -\log \det_s \boldsymbol{\Lambda} + s\nu + \sum_{i \in [n]} \mu_i - s : \boldsymbol{v}_i^\top \boldsymbol{\Lambda} \boldsymbol{v}_i \leq \nu + \mu_i, \ i \in [n] \right\}, \tag{6}$$

where the function $\det_s(\cdot)$ denotes the product of s least eigenvalues.

We note that the continuous relaxation value $\widehat{z}_M$ of M-DDF can be computed efficiently via the Frank-Wolfe algorithm (see [LX23]). Similar to R-DDF, M-DDF also admits a complementary formulation, which has been widely-studied in the MESP literature (e.g., see [AFLW96, AFLW99, Ans20, CFLL21, CFL23, FL22a]).

According to the identity

$$\det(\boldsymbol{I}_n + \boldsymbol{B}^\top \boldsymbol{B})_{S,S} = \det(\boldsymbol{I}_n + \boldsymbol{B}^\top \boldsymbol{B}) \det((\boldsymbol{I}_n + \boldsymbol{B}^\top \boldsymbol{B})^{-1})_{[n] \setminus S, [n] \setminus S}$$

for any subset $S \subseteq [n]$, see [JH85, Section 0.8.4], we can derive the complementary formulation for M-DDF and then the Lagrangian dual of its continuous relaxation.

PROPOSITION 6. *M-DDF is equivalent to*

$$z^* = \mathrm{ldet}(\boldsymbol{C} + \boldsymbol{A}\boldsymbol{A}^\top) + \max_{\boldsymbol{x} \in \mathcal{Z}_{n-s}} \left\{ f\left(\sum_{i \in [n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top \right) \right\}, \tag{M-DDF-comp}$$

and the Lagrangian dual of the continuous relaxation of M-DDF-comp is

$$\widehat{z}_M^c := \mathrm{ldet}(\boldsymbol{C} + \boldsymbol{A}\boldsymbol{A}^\top) + \min_{\substack{\boldsymbol{\Lambda} \in \mathbb{S}_{++}^n, \\ \nu, \boldsymbol{\mu} \in \mathbb{R}_+^n}} \left\{ -\log \det_{n-s} \boldsymbol{\Lambda} + (n-s)\nu + \sum_{i \in [n]} \mu_i - (n-s) : \boldsymbol{v}_i^\top \boldsymbol{\Lambda} \boldsymbol{v}_i \le \nu + \mu_i, \ i \in [n] \right\}. \tag{7}$$

Contrary to Proposition 4, we will observe in our numerical study in Section 5, that the Lagrangian dual problem (6) is not equivalent to (7). It is worth mentioning that the Lagrangian duals in Proposition 4, Proposition 5, and Proposition 6 enable us to derive the optimality cuts for DDF in the next section. Furthermore, the complementary problem M-DDF-comp and the Lagrangian dual (7) motivate us to further improve the approximation bounds of the local-search and sampling algorithms in [LX23], as shown in Section 4.

The two alternative formulations R-DDF and M-DDF, together with their complementary problems, often provide us with distinct continuous-relaxation solutions, which can help improve the analyses of the approximation algorithms. Each formulation has its own advantage under different circumstances. For example, if existing data contain more accurate information, i.e., if the existing FIM $\boldsymbol{C}$ dominates the overall FIM matrix, then we recommend R-DDF because its continuous relaxation provides a tighter upper bound. On the other hand, if the information from new data points is more valuable, i.e., the effect of $\boldsymbol{C}$ is negligible, then M-DDF tends to yield a stronger continuous relaxation bound. Our theoretical analyses and numerical study will further confirm these phenomena.

3. Our exact algorithmic approach for DDF

To solve DDF to optimality, we propose an enhancement on the LP/NLP B&B algorithm proposed in [QG92]. We first formulate DDF as

$$z^* = \mathrm{ldet}\, \boldsymbol{C} + \max_{\substack{z \in \mathbb{R}, \\ \boldsymbol{x} \in \mathcal{Z}_s}} \left\{ z \ : \ z \le \mathrm{ldet}\left(\boldsymbol{I}_d + \sum_{i \in [n]} x_i \boldsymbol{b}_i \boldsymbol{b}_i^\top \right), z \le f\left(\sum_{i \in [n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top \right) \right\}, \tag{DDF-MINLP}$$

for which the optimal value of the continuous relaxation is the best bound for DDF given by the continuous relaxations of both formulations R-DDF and M-DDF.

Using the concavity of the objective functions of both formulations, LP/NLP B&B considers, for a given $S \subseteq [n]$, the following linear relaxations of the two nonlinear inequalities in DDF-MINLP:

$$\text{(Linearization)} \quad z \le \mathrm{ldet}\,[\boldsymbol{B}(S)] - \sum_{i \in S} \boldsymbol{b}_i^\top [\boldsymbol{B}(S)]^{-1} \boldsymbol{b}_i + \sum_{i \in [n]} \boldsymbol{b}_i^\top [\boldsymbol{B}(S)]^{-1} \boldsymbol{b}_i x_i,$$

$$z \le f\,[\boldsymbol{V}(S)] - \sum_{i \in S} \boldsymbol{v}_i^\top \boldsymbol{g}(S) \boldsymbol{v}_i + \sum_{i \in [n]} \boldsymbol{v}_i^\top \boldsymbol{g}(S) \boldsymbol{v}_i x_i, \tag{8}$$

where for any subset $T \subseteq [n]$, $\boldsymbol{g}(T) := \boldsymbol{V}^\dagger(T) + 1/\lambda_{\min}(\boldsymbol{V}(T))[\boldsymbol{I}_d - \boldsymbol{V}(T)\boldsymbol{V}^\dagger(T)]$ is a subgradient of $f(\cdot)$ at $\boldsymbol{V}(T) := \sum_{i \in T} \boldsymbol{v}_i \boldsymbol{v}_i^\top$, according to Proposition 2 in [LX23].

Then, at iteration ℓ of LP/NLP B&B, a mixed-integer linear-programming (MILP) problem M^ℓ is solved. The so-called master problem M^ℓ is a relaxation of DDF-MINLP obtained by replacing the nonlinear inequalities in z by the inequalities in (8), constructed for all S in a given set $\mathcal{S}^\ell$ of linearization points represented by elements of $\binom{[n]}{s}$. M^ℓ is solved by a branch-and-bound algorithm, and every time a feasible solution $\widehat{\boldsymbol{x}}$ of M^ℓ is obtained during the execution of the algorithm,

supp($\hat{\boldsymbol{x}}$) is included in $\mathcal{S}^\ell$. In [MFR20], the authors highlight that LP/NLP B&B can be efficiently implemented in advanced MILP packages (e.g., CPLEX, Gurobi) with the use of lazy constraints and callback functions. As the set of linearization points increases at each iteration of LP/NLP B&B, the solution values of M^ℓ form a non-increasing sequence of upper bounds for DDF-MINLP and the algorithm stops when the difference between this upper bound and the best known lower bound is small enough (see [QG92]).

3.1. Submodular cuts

Next, we propose a first enhancement on LP/NLP B&B for DDF-MINLP. We note that, because the objective functions of R-DDF and M-DDF are monotone (due to (2)) and submodular (via the Hadamard-Fischer inequalities: see, for example, [JH85, Section 7.8, Problem 14]), according to the results in [WN99, AA11], the following submodular linear inequalities are valid for DDF-MINLP, for any $S \subseteq [n]$:

$$\text{(Submodular)} \quad z \leq \text{ldet}\,[\boldsymbol{B}(S)] - \sum_{i \in S} \rho_i(N \setminus \{i\})(1 - x_i) + \sum_{i \in [n] \setminus S} \rho_i(S) x_i,$$
$$z \leq \text{ldet}\,[\boldsymbol{B}(S)] - \sum_{i \in S} \rho_i(S \setminus \{i\})(1 - x_i) + \sum_{i \in [n] \setminus S} \rho_i(\emptyset) x_i, \tag{9}$$

where for any $T \subseteq [n]$ and $i \in [n] \setminus T$, we define the difference function $\rho_i(T) := \text{ldet}\,[\boldsymbol{B}(T) + \boldsymbol{b}_i \boldsymbol{b}_i^\top] - \text{ldet}\,[\boldsymbol{B}(T)]$ with $\boldsymbol{B}(T) := \boldsymbol{I}_d + \sum_{i \in T} \boldsymbol{b}_i \boldsymbol{b}_i^\top$.

Moreover, by using the identity for rank-one update of the determinant of a symmetric matrix, we have the closed-form expression for the difference function $\rho_i(T) = \log[1 + \boldsymbol{b}_i^\top [\boldsymbol{A}(T)]^{-1} \boldsymbol{b}_i]$. Similarly, for any subset $T \subseteq [n]$ and $i \in T$, we have $\rho_i(T \setminus \{i\}) = -\log[1 - \boldsymbol{b}_i^\top [\boldsymbol{A}(T)]^{-1} \boldsymbol{b}_i]$. Hence, the constraint coefficients involved with the difference functions can be easily computed.

Then, to tighten the MILP relaxation of DDF-MINLP at LP/NLP B&B, besides including the standard linearization inequalities (8) in M^ℓ, we also include the submodular inequalities (9), for each set S added to $\mathcal{S}^\ell$.

3.2. Optimality cuts

We consider choosing one or multiple data points and fixing their corresponding binary variables in DDF-MINLP to either one or zero, and then probing the restricted DDF to derive effective optimality cuts on these binary variables, which can help significantly reduce the size of the feasible region of DDF while maintaining the optimal value. Specifically, suppose that sets $S^1, S^0 \subseteq [n]$ denote the index set of data points being selected (i.e., $x_i = 1$ for each $i \in S^1$) and being discarded (i.e., $x_i = 0$ for each $i \in S^0$), respectively. Then a restricted problem of DDF is defined as

$$\text{(Restricted DDF)} \quad z(S^1, S^0) := \max_{\substack{S \subseteq [n] \setminus (S^1 \cup S^0), \\ |S| = s - |S^1|}} \text{ldet}\left(\boldsymbol{C} + \sum_{i \in S^1} \boldsymbol{a}_i \boldsymbol{a}_i + \sum_{i \in S} \boldsymbol{a}_i \boldsymbol{a}_i\right), \tag{10}$$

where sets S^1, S^0 are disjoint and set S^1 is of cardinality no larger than s.

Clearly, if $S^1 = S^0 = \emptyset$, then $z(S^1, S^0) = z^*$ and the formulation is equivalent to DDF. Otherwise, if $z(S^1, S^0) < z^*$, then at least one constraint built on sets S^1, S^0 is violated by an optimal solution and we thus obtain an optimality cut for DDF-MINLP that cuts off a subset of sub-optimal solutions. This result is summarized below.

THEOREM 4. *For any two disjoint sets S^1, S^0 with $|S^1| \leq s$ and $S^1 \cup S^0 \subseteq [n]$, if $z(S^1, S^0) < z^*$, then at least one of the two inequalities below is an optimality cut of DDF-MINLP.*

$$\sum_{i \in S^1} x_i \leq |S^1| - 1, \quad \sum_{j \in S^0} x_j \geq 1, \quad \forall \boldsymbol{x} \in \mathcal{Z}_s. \tag{11}$$

Proof. Given an optimal solution $\boldsymbol{x}^*$ of DDF-MINLP, if $\boldsymbol{x}^*$ satisfies the constraints $\sum_{i \in S^1} x_i^* = |S^1|$ and $\sum_{j \in S^0} x_j^* = 0$, then supp$(\boldsymbol{x}^*)$ will be feasible to the restricted problem (10) with the same objective value z^*, which contradicts $z(S^1, S^0) < z^*$. Therefore, $\boldsymbol{x}^*$ must violate at least one of the equality constraints above and using the fact that $\boldsymbol{x}^*$ is binary, we complete the proof. $\square$

For the result of Theorem 4, we remark that (i) if either S^1 or S^0 is empty, then the inequality in (11) based on the non-empty set must be an optimality cut; (ii) if one of sets S^1 and S^0 is singleton and the other one is empty, the inequalities in (11) recover the well-known variable-fixing ones [FL10]; and (iii) if both sets S^1 and S^0 are non-empty, the optimality cuts in (11) can be enforced via disjunctive programming. The optimality cuts are effective at reducing the search space and significantly improve the LP/NLP B&B algorithm as shown in our numerical results.

Albeit being effective, a common criticism of the probing technique in mixed-integer programming is its computation expense [ABG+20], e.g., the optimal values $z(S^1, S^0)$ and z^* in Theorem 4 may not be easily computable. Motivated by our near-optimal approximation algorithms and strong Lagrangian dual bounds for DDF, a compromise is that if an upper bound for $z(S^1, S^0)$ is less than a lower bound for z^* (denoted by z^{lb}), then the conclusion in Theorem 4 holds. Besides, following the spirit of the two Lagrangian duals for the continuous relaxations of DDF in Section 2, the restricted problem (10) also admits two alternative upper bounds as follows, corresponding to Lagrangian dual problems (3) and (6), respectively.

$$
\begin{aligned}
\widehat{z}_R(S^1, S^0) := \text{ldet}\,\boldsymbol{C} + \min_{\boldsymbol{\Lambda} \in \mathbb{S}_{++}^d, \nu, \boldsymbol{\mu} \in \mathbb{R}_+^n} &\left\{ -\text{ldet}\,\boldsymbol{\Lambda} + \text{tr}\,\boldsymbol{\Lambda} + \nu + \sum_{i \in [n]} \mu_i - d \right. \\
&+ \left(\sum_{j \in S^1} (\boldsymbol{b}_j^\top \boldsymbol{\Lambda} \boldsymbol{b}_j - \nu - \mu_j) \right) + \left(-\sum_{l \in S^0} \mu_l \right) : \boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i \le \nu + \mu_i, \forall i \in [n] \setminus \{S^1 \cup S^0\} \bigg\}, \\
\widehat{z}_M(S^1, S^0) := \text{ldet}\,\boldsymbol{C} + \min_{\boldsymbol{\Lambda} \in \mathbb{S}_{++}^n, \nu, \boldsymbol{\mu} \in \mathbb{R}_+^n} &\left\{ -\log\det_s(\boldsymbol{\Lambda}) + s\nu + \sum_{i \in [n]} \mu_i - s \right. \\
&+ \left(\sum_{j \in S^1} (\boldsymbol{v}_j^\top \boldsymbol{\Lambda} \boldsymbol{v}_j - \nu - \mu_j) \right) + \left(-\sum_{l \in S^0} \mu_l \right) : \boldsymbol{v}_i^\top \boldsymbol{\Lambda} \boldsymbol{v}_i \le \nu + \mu_i, \forall i \in [n] \setminus \{S^1 \cup S^0\} \bigg\}.
\end{aligned}
\tag{12}
$$

We observe that for some appropriately selected sets S^1 and S^0, the Lagrangian dual bounds (12) can be smaller than the lower bound of DDF, i.e., $\widehat{z}_R(S^1, S^0) < z^{lb}$ or $\widehat{z}_M(S^1, S^0) < z^{lb}$. Our selection strategy of sets S^1 and S^0 is a unification of the primal and dual perspectives, with an aim of reducing values: $\widehat{z}_R(S^1, S^0)$ and $\widehat{z}_M(S^1, S^0)$, which is discussed below. Notably, it is desired that set S^1 contains data points being discarded at optimality, and set S^0 contains data points being selected at optimality, which will provide the optimality cuts for DDF as shown in Theorem 4.

(i) **Primal:** Given an optimal solution $\widehat{\boldsymbol{x}}$ to the continuous relaxation of R-DDF or M-DDF, we let $S^1 \subseteq \{i : \widehat{x}_i \le \xi^0, \forall i \in [n]\}$ with $\xi^0 \in [0,1]$ being a positive number close to 0 and $S^0 \subseteq \{i : \widehat{x}_i \ge \xi^1, \forall i \in [n]\}$ with $\xi^1 \in [0,1]$ being close to 1. In this case, we expect a big reduction on the restricted Lagrangian dual bounds in (12), when compared to the unrestricted bounds $\widehat{z}_R$ and $\widehat{z}_M$. Our numerical experiments suggest that this selection strategy performs very well in exploring appropriate subset S^1 to construct an optimality cut.

(ii) **Dual:** Using Lagrangian dual formulations (3) and (6), we can ensure that the Lagrangian dual bounds of restricted DDF problem (10) decrease by at least a given threshold. According Lagrangian dual formulations in (12), given an optimal dual solution $(\widehat{\boldsymbol{\Lambda}}, \widehat{\nu}, \widehat{\boldsymbol{\mu}})$ of (3) or (6), we see that the corresponding restricted Lagrangian dual bound achieves a reduction of at least $\sum_{j \in S^1} (\boldsymbol{b}_j^\top \widehat{\boldsymbol{\Lambda}} \boldsymbol{b}_j - \widehat{\nu} - \widehat{\mu}_j) - \sum_{l \in S^0} \widehat{\mu}_l$ or $\sum_{j \in S^1} (\boldsymbol{v}_j^\top \widehat{\boldsymbol{\Lambda}} \boldsymbol{v}_j - \widehat{\nu} - \widehat{\mu}_j) - \sum_{l \in S^0} \widehat{\mu}_l$, compared with the original optimal dual value (see Proposition 7 below). This inspires us to identify sets S^1 and S^0 satisfying $\sum_{j \in S^1} (\boldsymbol{b}_j^\top \widehat{\boldsymbol{\Lambda}} \boldsymbol{b}_j - \widehat{\nu} - \widehat{\mu}_j) < 0$ or $\sum_{j \in S^1} (\boldsymbol{v}_j^\top \widehat{\boldsymbol{\Lambda}} \boldsymbol{v}_j - \widehat{\nu} - \widehat{\mu}_j) < 0$, and $\sum_{l \in S^0} \widehat{\mu}_l > 0$, such that each restricted Lagrangian dual problems in (12) yields a smaller upper bound for restricted DDF (10) than the original Lagrangian dual value. It is worth mentioning that

we can warm-start the solution procedure of the restricted dual problems (12) by using the optimal solution $(\widehat{\boldsymbol{\Lambda}}, \widehat{\nu}, \widehat{\boldsymbol{\mu}})$ of the original Lagrangian dual problem (3) or (6). We also observe in the numerical study that the dual selection strategy is good at exploring an appropriate subset S^0 to construct an optimality cut.

PROPOSITION 7. *Let* $(\widehat{\boldsymbol{\Lambda}}, \widehat{\nu}, \widehat{\boldsymbol{\mu}})$ *be a feasible solution of Lagrangian dual problem* (3) *(resp.,* (6)*) with the objective value* z^{ub}. *Then,* $(\widehat{\boldsymbol{\Lambda}}, \widehat{\nu}, \widehat{\boldsymbol{\mu}})$ *is also feasible to its corresponding restricted Lagrangian dual problem in* (12) *and the resulting objective value is equal to*

$$z^{ub} + \sum_{j \in S^1} (\boldsymbol{b}_j^\top \widehat{\boldsymbol{\Lambda}} \boldsymbol{b}_j - \widehat{\nu} - \widehat{\mu}_j) - \sum_{l \in S^0} \widehat{\mu}_l \quad \left(resp., \ \ z^{ub} + \sum_{j \in S^1} (\boldsymbol{v}_j^\top \widehat{\boldsymbol{\Lambda}} \boldsymbol{v}_j - \widehat{\nu} - \widehat{\mu}_j) - \sum_{l \in S^0} \widehat{\mu}_l \right).$$

Proof. If $(\widehat{\boldsymbol{\Lambda}}, \widehat{\nu}, \widehat{\boldsymbol{\mu}})$ is a feasible solution of Lagrangian dual problem (3), it is easy to check that it is also feasible to the first optimization problem in (12) whose objective value is $z^{ub} + \sum_{j \in S^1} (\boldsymbol{b}_j^\top \widehat{\boldsymbol{\Lambda}} \boldsymbol{b}_j - \widehat{\nu} - \widehat{\mu}_j) - \sum_{l \in S^0} \widehat{\mu}_l$. Similarly, for the Lagrangian dual problem (6), the same result holds by replacing $\boldsymbol{b}_i$ by $\boldsymbol{v}_i$ for all $i \in [n]$. $\square$

We note that (i) our selection strategies are easy-to-implement because each continuous relaxation of DDF and its corresponding Lagrangian dual problem can be efficiently solved by the primal-dual Frank-Wolfe algorithm with a sublinear rate of convergence, (ii) the primal and dual selection strategies do not dominate each other and are complementary as shown in our numerical experiments, and (iii) all the analyses and selection strategies can be directly extended to complementary formulations of DDF.

4. Two approximation algorithms for DDF

Motivated by our two CIP formulations of DDF, we investigate simple and scalable approximation algorithms (i.e., local-search and randomized-sampling algorithms) for providing near-optimal selections of the new data points.

4.1. A local-search algorithm

In this subsection, we study a local-search algorithm for DDF, which has been successfully applied to many combinatorial optimization problems (see, for example, [SX20, LX23, LX20]). The algorithm runs as follows: (i) first, we start with a cardinality-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 the swapping strictly increases the objective value; and (iii) the algorithm terminates when there is no improvement. Motivated by R-DDF, we provide an efficient implementation of the local-search algorithm, as shown in Algorithm 1, with time complexity of $\mathcal{O}(nd^2 + s(n-s)d^2)$ at the for-loop (i.e., Steps 5-11). Specifically, at Step 6, the strict improvement $\det(\boldsymbol{I}_d + \boldsymbol{X} - \boldsymbol{b}_i \boldsymbol{b}_i^\top + \boldsymbol{b}_j \boldsymbol{b}_j^\top) > \det(\boldsymbol{I}_d + \boldsymbol{X})$, can be efficiently computed as

$$\begin{aligned}
&\det(\boldsymbol{I}_d + \boldsymbol{X} - \boldsymbol{b}_i \boldsymbol{b}_i^\top + \boldsymbol{b}_j \boldsymbol{b}_j^\top) > \det(\boldsymbol{I}_d + \boldsymbol{X}) \\
\Longleftrightarrow \ &\det(\boldsymbol{I}_d + \boldsymbol{X})(1 + \boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i) \left[1 + \boldsymbol{b}_j^\top (\boldsymbol{I}_d + \boldsymbol{X} - \boldsymbol{b}_i \boldsymbol{b}_i^\top)^{-1} \boldsymbol{b}_j \right] > \det(\boldsymbol{I}_d + \boldsymbol{X}) \\
\Longleftrightarrow \ &\boldsymbol{b}_j^\top \boldsymbol{\Lambda} \boldsymbol{b}_j - \boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i \boldsymbol{b}_j^\top \boldsymbol{\Lambda} \boldsymbol{b}_j + \boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_j \boldsymbol{b}_j^\top \boldsymbol{\Lambda} \boldsymbol{b}_i > \boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i,
\end{aligned} \tag{13}$$

which follows from the Sherman–Morrison formula. The update of matrix $\boldsymbol{\Lambda}$ at Step 8 also follows from Sherman–Morrison formula, to avoid calculations of inverses from scratch.

We use the proposed Lagrangian-dual of R-DDF, M-DDF, and their complements R-DDF-comp, M-DDF-comp to provide a valid bound for the output of Algorithm 1 by constructing feasible dual solutions:

Algorithm 1 Local-Search Algorithm

1: **Input:** vectors $\{\boldsymbol{b}_i \in \mathbb{R}^d\}_{i\in[n]}$ and a positive integer $s \in [n]$
2: Initialize a cardinality-s subset $\widehat{S} \subseteq [n]$, matrix $\boldsymbol{X} := \sum_{i\in\widehat{S}} \boldsymbol{b}_i\boldsymbol{b}_i^\top$, and matrix $\boldsymbol{\Lambda} := (\boldsymbol{I}_d + \boldsymbol{X})^{-1}$
3: Compute $\boldsymbol{b}_j^\top\boldsymbol{\Lambda}\boldsymbol{b}_j$ for each $j\in[n]$ and $\boldsymbol{b}_i^\top\boldsymbol{\Lambda}\boldsymbol{b}_j$ for each $i\in\widehat{S}$, $j\in[n]\setminus\widehat{S}$
4: **do**
5:     **for** each pair $(i,j) \in \widehat{S}\times([n]\setminus\widehat{S})$
6:        **if** $\boldsymbol{b}_j^\top\boldsymbol{\Lambda}\boldsymbol{b}_j - \boldsymbol{b}_i^\top\boldsymbol{\Lambda}\boldsymbol{b}_i\boldsymbol{b}_j^\top\boldsymbol{\Lambda}\boldsymbol{b}_j + (\boldsymbol{b}_i^\top\boldsymbol{\Lambda}\boldsymbol{b}_j)^2 > \boldsymbol{b}_i^\top\boldsymbol{\Lambda}\boldsymbol{b}_i$
7:           Update $\widehat{S} := \widehat{S}\cup\{j\}\setminus\{i\}$ and $\boldsymbol{X} := \boldsymbol{X} - \boldsymbol{b}_i\boldsymbol{b}_i^\top + \boldsymbol{b}_j\boldsymbol{b}_j^\top$
8:           Compute $\boldsymbol{\Lambda}_i := \boldsymbol{\Lambda} + \frac{\boldsymbol{\Lambda}\boldsymbol{b}_i\boldsymbol{b}_i^\top\boldsymbol{\Lambda}}{1-\boldsymbol{b}_i^\top\boldsymbol{\Lambda}\boldsymbol{b}_i}$, and update $\boldsymbol{\Lambda} := \boldsymbol{\Lambda}_i - \frac{\boldsymbol{\Lambda}_i\boldsymbol{b}_j\boldsymbol{b}_j^\top\boldsymbol{\Lambda}_i}{1+\boldsymbol{b}_j^\top\boldsymbol{\Lambda}_i\boldsymbol{b}_j}$
9:           Update $\boldsymbol{b}_l^\top\boldsymbol{\Lambda}\boldsymbol{b}_l$ for each $l\in[n]$, and update $\boldsymbol{b}_{i'}^\top\boldsymbol{\Lambda}\boldsymbol{b}_{j'}$ for each $i'\in\widehat{S}$, $j'\in[n]\setminus\widehat{S}$
10:       **end if**
11:     **end for**
12: **while** there is still an improvement (i.e., Step 6 is true for some pair (i,j))
13: **Output:** $\widehat{S}$

THEOREM 5. *Let $\widehat{S}$ denote the output of the local-search Algorithm 1, and let $\bar{s} := \min\{s, n-s\}$, then the set $\widehat{S}$ yields a $\min\{d\log(1+(\bar{s}/d)\sigma_{\max}^2/(1+\sigma_{\max})), \bar{s}\log(\bar{s})\}$-approximation bound for DDF, i.e.,*

$$\mathrm{ldet}\left(\boldsymbol{C} + \sum_{i\in\widehat{S}}\boldsymbol{a}_i\boldsymbol{a}_i^\top\right) \geq z^* - \min\left\{d\log\left(1 + \frac{\bar{s}\sigma_{\max}^2}{d(1+\sigma_{\max})}\right), \bar{s}\log(\bar{s})\right\},$$

where the constant $\sigma_{\max} := \max_{i\in[n]}\boldsymbol{a}_i^\top\boldsymbol{C}^{-1}\boldsymbol{a}_i$.

Proof. The approximation bound attains the minimum of $d\log(1+(\bar{s}/d)\sigma_{\max}^2/(1+\sigma_{\max}))$ and $\bar{s}\log(\bar{s})$, where they are derived based on R-DDF, M-DDF, and their complementary problems, respectively. Exploring the local optimality of the output solution $\widehat{S}$, we can show that

$$z^* \leq \widehat{z}_R \leq \mathrm{ldet}\,\boldsymbol{C} + \mathrm{ldet}\left(\boldsymbol{I}_d + \sum_{i\in\widehat{S}}\boldsymbol{b}_i\boldsymbol{b}_i^\top\right) + d\log\left(1 + \frac{\bar{s}\sigma_{\max}^2}{d(1+\sigma_{\max})}\right),$$

$$z^* \leq \widehat{z}_M \leq \log f\left(\sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top\right) + \mathrm{ldet}(\boldsymbol{C}) + s\min\left\{\log(s), \log\left(n-s-\tfrac{n}{s}+2\right)\right\},$$

$$z^* \leq \widehat{z}_M^c \leq \log f\left(\sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top\right) + \mathrm{ldet}(\boldsymbol{C}) + (n-s)\min\left\{\log(n-s), \log\left(s-\tfrac{n}{n-s}+2\right)\right\}.$$

The detailed proof can be found in e-companion EC.1.3. $\qquad\square$

We make the following remarks concerning Theorem 5:

(i) Either approximation bound with the 'min' in Theorem 5 is invariant with s and $n-s$ by leveraging the two CIPs and their complements;

(ii) The second approximation bound attains zero when $\bar{s}=1$, implying that the output solution of the local-search Algorithm 1 is optimal, and when $s=1$, we have $\widehat{z}_M = z^*$, and when $s=n-1$, we have $\widehat{z}_M^c = z^*$, as summarized in Corollary 1;

(iii) The second approximation bound (i.e., $\bar{s}\log(\bar{s})$) of the local-search Algorithm 1 improves on the one for MESP (i.e., $s\log[s - (s-1)/s(2s-n)_+]$) derived by [LX23], when the covariance matrix of MESP is positive-definite. Figure 1 illustrates the comparisons of two approximation bounds (i.e., our new bound $\bar{s}\log(\bar{s})$ versus the existing bound $s\log[s-(s-1)/s(2s-n)_+]$) with $n = 10, 50$, where the greatest improvement of our bound over that in [LX23] is indicated by a black dashed line. We see that when $s > n/2$, our new bound provides the local-search Algorithm 1 with a tighter performance guarantee;

(iv) The first approximation bound involving the constant $\sigma_{\max}$ is new and is discussed after Theorem 6, along with that of the sampling algorithm which is also derived based on R-DDF;

(v) Another side product of Theorem 5 is to provide DDF with the optimality gaps of three proposed Lagrangian dual bounds: $\widehat{z}_R$, $\widehat{z}_M$, and $\widehat{z}_M^c$, as presented in Corollary 2; and

(vi) Finally, we provide the maximum number of swaps and time complexity of each swap adopted in Algorithm 1 in Corollary 3.

COROLLARY 1. *When $s = 1$ and $s = n - 1$, we have $\widehat{z}_M = z^*$ and $\widehat{z}_M^c = z^*$, respectively. In both cases, the local-search Algorithm 1 returns an optimal solution.*
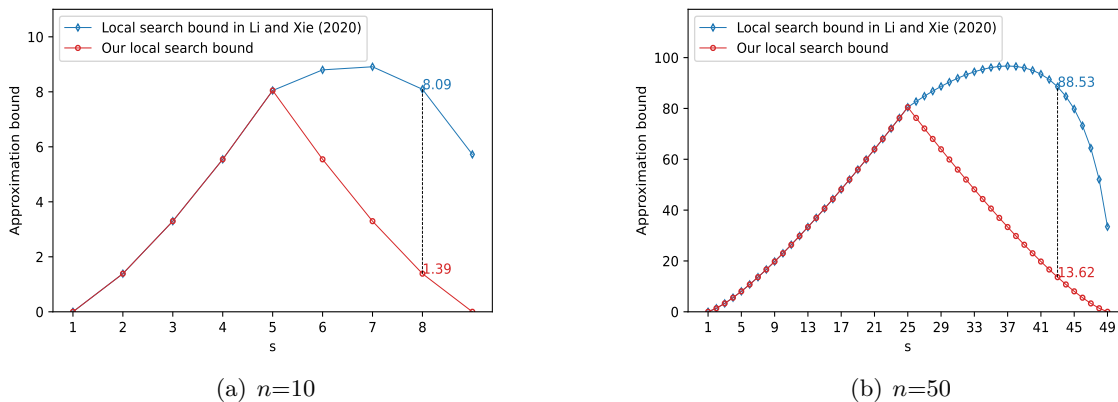

(a) $n=10$ (b) $n=50$

Figure 1   Comparison between our approximation bound from the local-search Algorithm 1 with the existing one

COROLLARY 2. *The continuous relaxation values $\widehat{z}_R$ of R-DDF, $\widehat{z}_M$ of M-DDF, and $\widehat{z}_M^c$ of the complementary M-DDF (M-DDF-comp) satisfy*

$$z^* \le \widehat{z}_R \le z^* + n \log\left(1 + \frac{\bar{s}\sigma_{\max}^2}{n(1+\sigma_{\max})}\right);$$
$$z^* \le \widehat{z}_M \le z^* + s \log\left(s - \frac{s-1}{s}(2s-n)_+\right);$$
$$z^* \le \widehat{z}_M^c \le z^* + (n-s) \log\left(n - s - \frac{n-s-1}{n-s}(n-2s)_+\right).$$

Proof. The proof follows from that of Theorem 5.   □

As mentioned before, R-DDF and its complement R-DDF-comp have the same continuous relaxation value, whereas M-DDF and its complement M-DDF-comp do not. This fact is captured by Corollary 2, where we demonstrate a symmetric optimality gap of $\widehat{z}_R$ and non-symmetric gaps of $\widehat{z}_M$ and $\widehat{z}_M^c$.

COROLLARY 3. *The local-search Algorithm 1 takes at most $\log_{1+\delta}(sL)$ swaps and each swap takes $\mathcal{O}(nd^2 + s(n-s)d^2)$ arithmetic operations, where $\delta > 0$ denotes the strict improvement factor, and L denotes the natural logarithm of the largest eigenvalue of matrix $\boldsymbol{I}_d + \boldsymbol{B}\boldsymbol{B}^\top$.*

Proof. For any cardinality-s subset $\widehat{S} \subseteq [n]$, suppose $\lambda_{\max}$ denotes the largest eigenvalue of matrix $\sum_{i \in \widehat{S}} \boldsymbol{b}_i \boldsymbol{b}_i^\top$, then we have

$$\text{ldet}\left(\boldsymbol{I}_d + \sum_{i \in \widehat{S}} \boldsymbol{b}_i \boldsymbol{b}_i^\top\right) \le s \log(1 + \lambda_{\max}) \le sL,$$

where the first inequality stems from the fact that matrix $\sum_{i \in \widehat{S}} \boldsymbol{b}_i \boldsymbol{b}_i^\top$ admits only s nonzero eigenvalues.

For Step 6 of the local-search Algorithm 1, we add a factor δ to the right-hand side of the strict inequality, where $\delta > 0$ denotes a strict improvement factor. Therefore, Algorithm 1 takes at most $\log_{1+\delta}(sL)$ swaps. Also, note that for each swapping pair (i, j) satisfying the condition at Step 6, all of the following arithmetic operations over $d \times d$ matrices are $\mathcal{O}(nd^2 + s(n-s)d^2)$. $\qquad\square$

4.2. A randomized-sampling algorithm

In this subsection, we study a randomized-sampling algorithm which relies on the optimal continuous-relaxation solutions of R-DDF, M-DDF, and M-DDF-comp. Given an optimal continuous-relaxation solution $\widehat{\boldsymbol{x}}$ of either problem, our Algorithm 2 samples a cardinality-s subset $S \subseteq [n]$ with appropriate probability.

Algorithm 2 Sampling Algorithm

1: **Input:** An optimal solution $\widehat{\boldsymbol{x}} \in [0,1]^n$ of the continuous relaxation of R-DDF, M-DDF, or M-DDF-comp and a positive integer $s \in [n]$
2: For a cardinality-s subset $S \in [n]$, its probability to be chosen is

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

3: **Output:** Random set $\tilde{S}$

The sampling procedure has $O(n \log n)$ complexity, and its detailed efficient implementation can be found in [SX20, Section 3.1]. The approximation bound of the output of Algorithm 2 depends on the choice of the relaxation, as presented in Theorem 6.

THEOREM 6. *Let $\widehat{\boldsymbol{x}}_D, \widehat{\boldsymbol{x}}_M, \widehat{\boldsymbol{x}}_M^c$ denote optimal continuous-relaxation solutions of R-DDF, M-DDF, and M-DDF-comp, respectively. Suppose that Algorithm 2 generates random sets $\tilde{S}_D, \tilde{S}_M, \tilde{S}_{M^c}$ with $\widehat{\boldsymbol{x}}_D, \widehat{\boldsymbol{x}}_M, \widehat{\boldsymbol{x}}_M^c$ as inputs, respectively, then we have*

(i) $\log \mathbb{E}\left[\det\left(\boldsymbol{C} + \sum_{i \in \tilde{S}_D} \boldsymbol{a}_i \boldsymbol{a}_i^\top\right)\right] \geq z^* + n \log(x_{\min}) - (n-s) \log(1+\delta)$, *where $x_{\min} > 0$ is the least non-zero entry in $\widehat{\boldsymbol{x}}_D$ and $\delta := \lambda_{\max}(\boldsymbol{A}^\top \boldsymbol{C}^{-1} \boldsymbol{A})$;*

(ii) $\log \mathbb{E}\left[\det\left(\boldsymbol{C} + \sum_{i \in \tilde{S}_M} \boldsymbol{a}_i \boldsymbol{a}_i^\top\right)\right] \geq z^* - s \log\left(\frac{s}{n}\right) - \log\left(\binom{n}{s}\right);$

(iii) $\log \mathbb{E}\left[\det\left(\boldsymbol{C} + \sum_{i \in \tilde{S}_M^c} \boldsymbol{a}_i \boldsymbol{a}_i^\top\right)\right] \geq z^* - (n-s) \log\left(\frac{n-s}{n}\right) - \log\left(\binom{n}{n-s}\right);$

(iv) Algorithm 2 can be derandomized as a polynomial-time algorithm with the same performance guarantees.

We establish the approximation bounds for Algorithm 2, using the solutions of R-DDF, M-DDF, and M-DDF-comp. The detailed proof can be found in e-companion EC.1.4. We further remark:

(i) The performance of Algorithm 2 depends on the quality of the continuous-relaxation solution $\widehat{\boldsymbol{x}}$, i.e., a tighter continuous relaxation bound yields better sampling results. It can be also seen that the running time of Algorithm 2 is dependent of the dimensionality of the data, i.e., d;

(ii) The first and third approximation bounds in Theorem 6 are new, and the second one follows the one for MESP [LX23].

(iii) Using the optimal continuous-relaxation solutions from M-DDF and M-DDF-comp, the output from Algorithm 2 can be at most $\bar{s} \log(\bar{s}/n) + \log(\binom{n}{\bar{s}})$ away from the optimal value (recall that $\bar{s} := \min\{s, n-s\}$), which improves the approximation bound $(s \log(\bar{s}/n) + \log(\binom{n}{s}))$ of Algorithm 2 for solving MESP in [LX23]. The comparison between these two bounds is displayed in Figure 2; and

(iv) Similar to the corollaries of Theorem 5, Theorem 6 also implies the optimality of Algorithm 2 for two special cases in Corollary 4 and alternative optimality gaps of the proposed continuous relaxation values in Corollary 5.
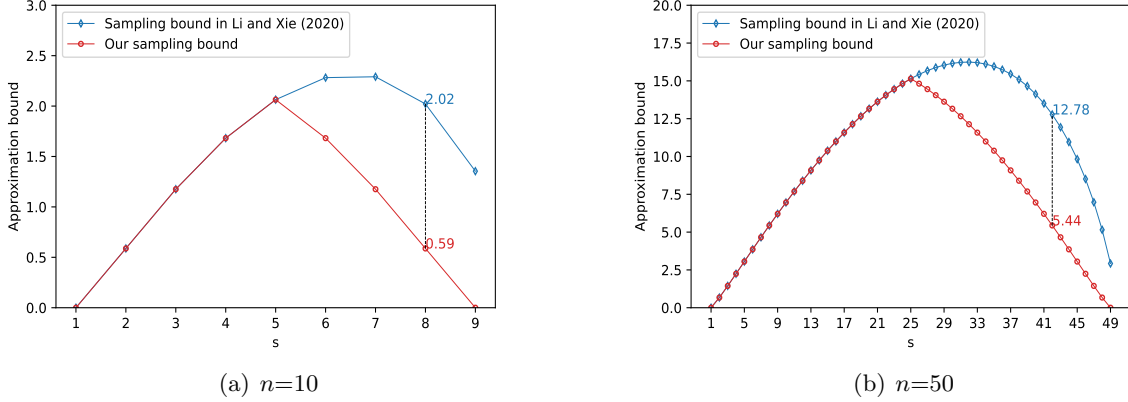

(a) n=10   (b) n=50

Figure 2   Comparison between our approximation bound from the sampling Algorithm 2 with the existing one

COROLLARY 4. *When $s = 1$ and $s = n - 1$, Algorithm 2 returns an optimal solution when using $\widehat{\boldsymbol{x}}_M$ and $\widehat{\boldsymbol{x}}_M^c$ as the input, respectively.*

Proof. According to Theorem 6, if $s = 1$ and $\widehat{\boldsymbol{x}}_M$ is used as the input, the approximation bound of Algorithm 2 is equal to 0. When $s = n - 1$, the same argument follows. $\square$

COROLLARY 5. *The continuous relaxation values $\widehat{z}_R$ of R-DDF, $\widehat{z}_M$ of M-DDF, and $\widehat{z}_M^c$ of M-DDF-comp satisfy*

$$z^* \leq \widehat{z}_R \leq z^* - n\log(x_{\min}) + (n - s)\log(1 + \delta);$$
$$z^* \leq \widehat{z}_M \leq z^* + s\log\left(\tfrac{s}{n}\right) + \log\left(\binom{n}{s}\right);$$
$$z^* \leq \widehat{z}_M^c \leq z^* + (n - s)\log\left(\tfrac{n-s}{n}\right) + \log\left(\binom{n}{n-s}\right).$$

Proof. The proof follows directly from that of Theorem 6. $\square$

Note that the theoretical optimality gaps of the three continuous relaxation values in Corollary 2 and Corollary 5 are not comparable; thus, taking the minimum of both values yields a better optimality gap. Specifically, for the continuous relaxation value $\widehat{z}_R$, the two optimality gaps from Corollary 2 and Corollary 5 depend on parameters $\sigma_{\max}$, δ, and $x_{\min}$ and thus are not comparable. The explicit comparison of alternative optimality gaps for $\widehat{z}_M$ and $\widehat{z}_M^c$ with $n = 10, 50$ can be found in Figure 3, where the blue diamond line and red circle line represent the results in Corollary 2 and Corollary 5, respectively. We observe that (i) for either continuous relaxation value, the optimality gap in Corollary 5 is tighter than that of Corollary 2, except one case; and (ii) if $s \leq n/2$, then $\widehat{z}_M$ outperforms $\widehat{z}_M^c$ and is a tighter upper bound for DDF.

Note that in Theorem 5 and Theorem 6, Corollary 2 and Corollary 5, the magnitudes of $\sigma_{\max}$ and δ depend on the contained information difference of new and existing data. If the existing data are very informative, i.e., $\sigma_{\max}$ and δ tend to be small, and the proposed approximation bounds based on R-DDF get tighter. Hence, for Algorithm 2, we recommend applying the continuous-relaxation solution of R-DDF if the existing data are more informative; otherwise, applying M-DDF and its complement.
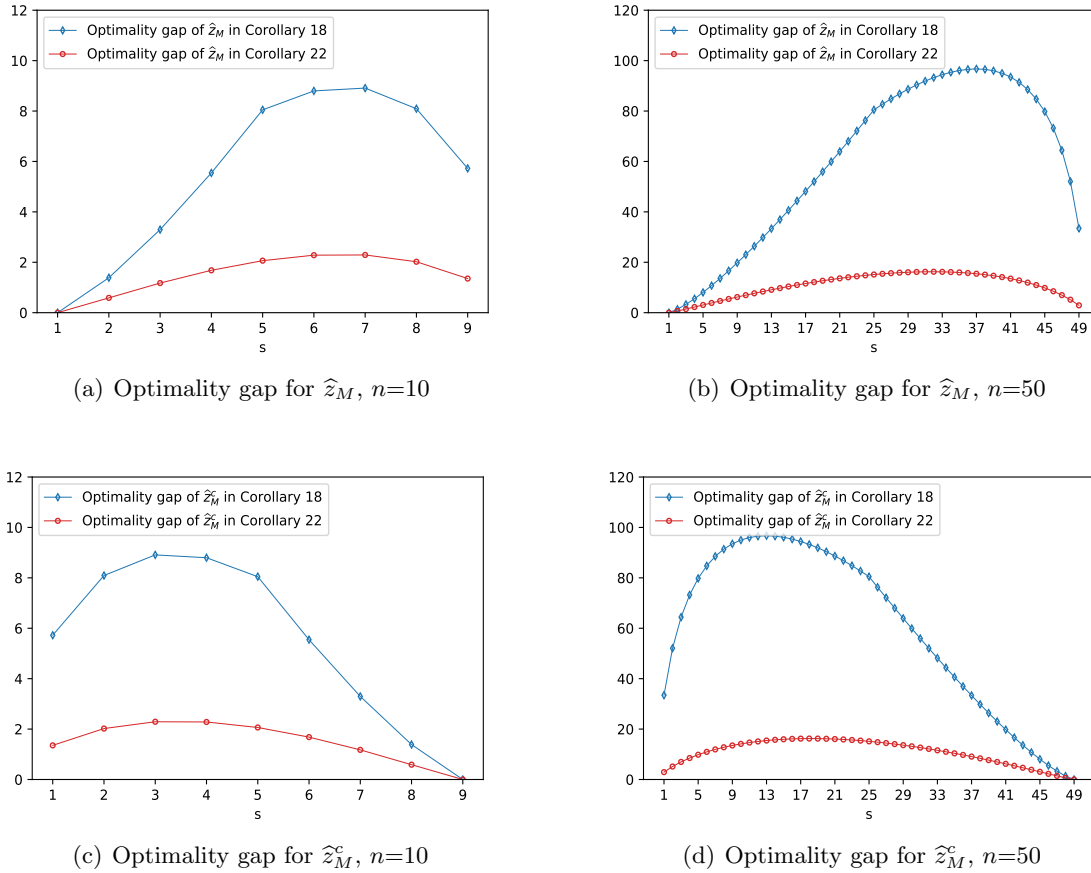
(a) Optimality gap for $\widehat{z}_M$, n=10

(b) Optimality gap for $\widehat{z}_M$, n=50

(c) Optimality gap for $\widehat{z}_M^c$, n=10

(d) Optimality gap for $\widehat{z}_M^c$, n=50

Figure 3   Comparison between optimality gaps for the two continuous relaxation values $\widehat{z}_M$ and $\widehat{z}_M^c$

5. Numerical study on sensor fusion in power systems

In this section, we present a real-world sensor-fusion problem in power systems, which can be formulated as DDF. We tested the proposed formulations and algorithms with varying-scale instances. All the experiments were conducted in Python 3.6 with calls to Gurobi 9.0 on a PC with 2.8 GHz Intel Core i5 processor and 8G of memory. All times reported are wall-clock times. The codes for the proposed algorithms and data are available in a Github repository at https://github.com/yongchunli-13/D-optimal-Data-Fusion.

In power systems, phasor measurement units (PMUs) are the most accurate and high-speed time-synchronized devices used to measure phasors of bus voltages and currents in an electric grid (see [Nuq01]). PMUs have broad applications, including state estimation, security assessment, system monitoring, and wide-area control (see [DLRCTP10, The17]). In particular, reliable state estimation is an essential component of managing a modern energy system, aiming at determining the true voltages at all buses, based on available measurements and information that consist of observed voltage angles, power flows, and injections (see [TVC+10]).

Before the advent of PMUs, the conventional sensors, including supervisory control and data acquisition (SCADA) meter readings, were widely used to perform state estimation in power systems (see [Mon00]). It is recognized that, in many cases, one barely equips a power grid with a sufficient number of PMUs to achieve the state estimation fully (see [ZCTP06]), due to the budget and resource constraints. Therefore, it is important to make an intelligent choice of fusing PMUs and SCADA when deciding the PMU locations out of non-reference buses, in order to collect maximum information to best improve the state estimation. According to the PMU and SCADA

measurement model in [LNI11], for an n-bus power system, the overall Fisher information matrix (FIM) is defined as

$$C + \sum_{i \in S} \frac{1}{\widehat{\sigma}_i^2} e_i e_i^\top, \tag{14}$$

where matrix $C \in \mathbb{S}_{++}^n$ denotes the FIM obtained from conventional sensors and is positive definite [Pal81], the set $S \subseteq [n]$ denotes the bus locations of installed PMUs, and for each $i \in [n]$, $\widehat{\sigma}_i$ denotes the standard deviation of PMU measurements at i-th bus. The matrix C in the FIM (14) is positive definite because power-system states are typically observable with conventional SCADAs installed (see, e.g., [LNI11]). The FIM (14) of the sensor fusion problem in power systems can reduce to the objective matrix in DDF by letting $a_i = 1/\widehat{\sigma}_i e_i$ for each $i \in [n]$ and $d = n$. When employing D-optimality as the information selection criterion of the sensor fusion problem, based on the FIM (14), it follows that DDF exactly formulates this sensor fusion problem (see [LNI11]). Please note that even when all standard deviations $\{\widehat{\sigma}_i\}_{i \in [n]}$ are equal, the DDF built on the special FIM (14) is still NP-hard as shown in e-companion EC.2. Nevertheless, in the next subsection, we randomly generate the PMU standard deviations to compare the proposed continuous relaxations across various settings. Additionally, all other numerical experiments are based on a pre-specified PMU standard error commonly used in the literature.

5.1. A comparison of continuous relaxations: IEEE 118- and 300-bus instances

From our theoretical analysis, we see that a tight continuous-relaxation bound for DDF has an important role in the performance of Algorithm 2 and the effectiveness of optimality cuts. Therefore, we first investigated the three continuous-relaxation bounds of R-DDF, M-DDF, and M-DDF-comp, i.e., $\widehat{z}_R$, $\widehat{z}_M$, and $\widehat{z}_M^c$, respectively, using two IEEE benchmark instances with 118 and 300 buses (see [AKFFS09]) of the PMU placement problem that provide the matrix C in DDF. For each instance, we set the first bus to be the reference bus and renumber non-reference buses from one, as seen in [LNI11]. Hence, it suffices to focus on the $n = 117$, $n = 299$ non-reference buses when solving the IEEE-118, 300 instance. To compare the three alternative continuous-relaxation values, we conducted a controlled experiment with respect to the PMU standard deviations $\{\widehat{\sigma}_i\}_{i \in [n]}$, where large and small PMU standard deviations separately represent the two cases where either the existing sensors or the new sensors are more accurate for state estimation.

We used the Frank-Wolfe algorithm to compute the three upper bounds on the optimal value of DDF. The computational results for the two instances are displayed in Figure 4 and Figure 5, where the optimality gap is equal to the difference between an upper bound and a lower bound for DDF returned by our local-search Algorithm 1. We note that the Frank-Wolfe algorithm and our local-search Algorithm 1 are very efficient, and their computational time is negligible (i.e., less than one minute), so we do not report them.

For the 118-bus instance (Figure 4), we consider cases where the number of installed PMUs $s \in \{10, 15, \cdots, 105\}$, for $n = 117$, in order to compare the upper bounds for a wide range of the parameter s. In Figure 4(a), we sample the PMU variances $\{\widehat{\sigma}_i^2\}_{i \in [n]}$ as n independent uniform random variables in the range $[0, 1000]$, and the new sensors contribute less to the state estimation than the existing ones. We see that the continuous-relaxation value $\widehat{z}_R$ is smaller than both $\widehat{z}_M$ and $\widehat{z}_M^c$ in most cases in Figure 4(a). By contrast, in Figure 4(b), we sample the PMU variances $\{\widehat{\sigma}_i^2\}_{i \in [n]}$ as n independent uniform random variables in the range $[0, 0.1]$, assuming new more accurate sensors. In this setting, we see that the continuous-relaxation values $\widehat{z}_M$ and $\widehat{z}_M^c$ are much smaller than $\widehat{z}_R$, so we use a pair of vertical axes to illustrate their performance. The comparison results parallel our theoretical findings in Section 2.

The comparison of the three upper bounds is also illustrated in Figure 5 for the IEEE 300-bus instance and the conclusions are the same. Specifically, given the gain matrix C of this instance, we generate the relatively large and small PMU variances $\{\widehat{\sigma}_i^2\}_{i \in [n]}$ from the uniform distribution in the

range $[0, 10]$ and $[0, 0.01]$, respectively. Thus, both theoretical analyses and numerical comparisons in Figure 4 and Figure 5 demonstrate that the continuous relaxation of M-DDF is more stable and tighter than that of R-DDF when the new sensors are subject to smaller measurement variances. In addition, we observe in both figures that $\widehat{z}_M$ tends to be stronger than $\widehat{z}_M^c$ if s is small; otherwise $\widehat{z}_M^c$ is stronger. In practice, PMUs are much more accurate than other sensors (see [ZTW$^+$19]), and the measurement error is controlled within the range of $[-0.6°, 0.6°]$; and the number of PMUs to be installed is usually small due to budget constraints, i.e., s is small. Thus, we identify the continuous-relaxation value $\widehat{z}_M$ of M-DDF as a better upper bound for the PMU placement problem in power systems that will be used in the following numerical study, where we also set the PMU standard deviation to $0.02°$, a known PMU standard error in the literature (see [ZMST10]).
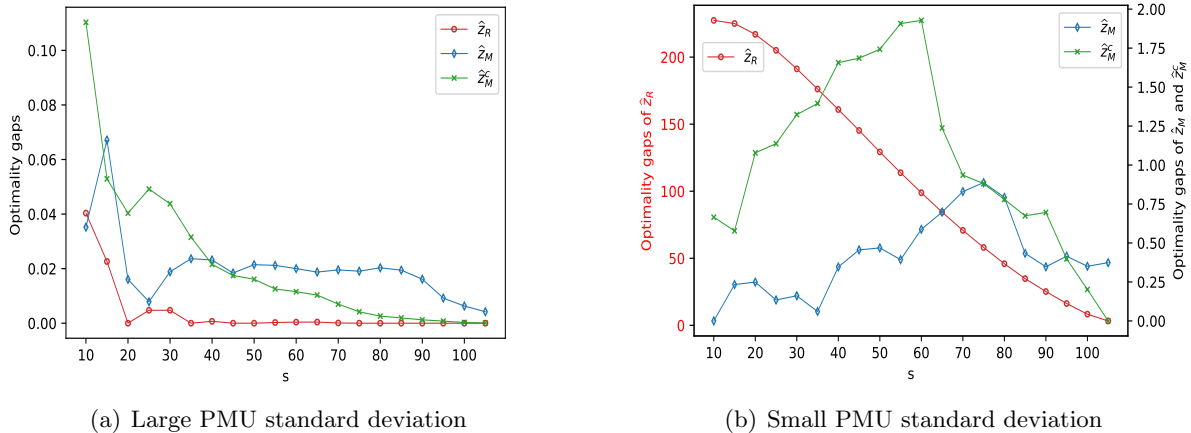

(a) Large PMU standard deviation   (b) Small PMU standard deviation

Figure 4   Comparison between continuous-relaxation bounds on IEEE 118-bus instance

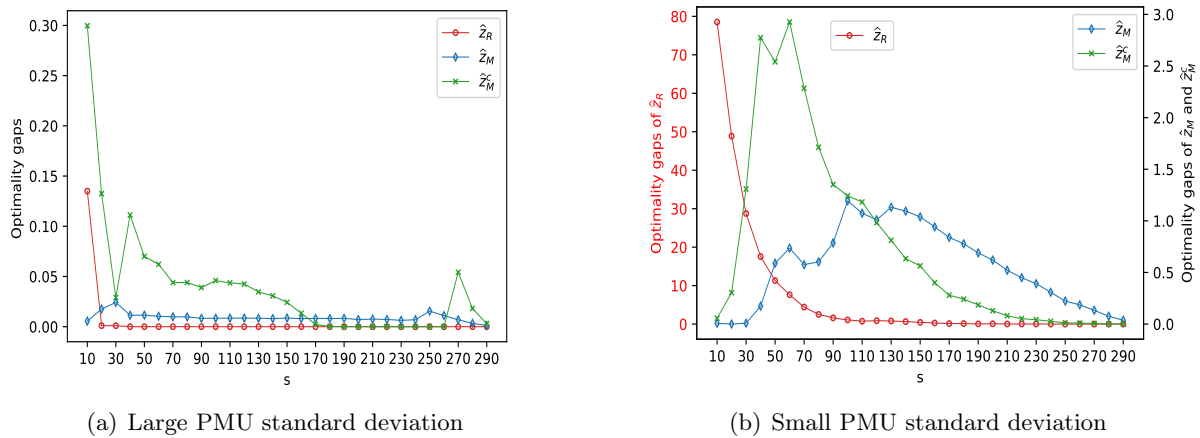(a) Large PMU standard deviation   (b) Small PMU standard deviation

Figure 5   Comparison between continuous-relaxation bounds on IEEE 300-bus instance

5.2. Testing submodular and optimality cuts: IEEE 118- and 300-bus instances

Next, we tested the submodular and optimality cuts introduced in Section 3. For our experiments, the proposed primal and dual selection strategies of subsets S^1 and S^0 only depend on the continuous relaxation of M-DDF, and its Lagrangian dual, due to their strength for the power-system instances. Similarly, for the selected subsets and their resulting restricted problems (10), we only computed the Lagrangian dual value $\widehat{z}_M(S^1, S^0)$ in (12) to check whether this value is less than a lower bound returned by the local-search Algorithm 1. That is, for a pair (S^1, S^0), we applied the

Frank-Wolfe algorithm with a warm start to compute the restricted dual bound $\widehat{z}_M(S^1, S^0)$, unless the optimality cuts based on (S^1, S^0) could be directly obtained using Proposition 7.

We considered different settings to generate a pair (S^1, S^0) to construct optimality cuts, including: (a) $S^1 = \emptyset$, S^0 is singleton; (b) S^1 is singleton, $S^0 = \emptyset$; (c) $S^1 = \emptyset$, S^0 has 2 elements; (d) S^1 has 2 elements, $S^0 = \emptyset$; and (e) both S^1 and S^0 are singletons. The subsets are set to be small due to the effectiveness of their corresponding optimality cuts and tightness of the continuous-relaxation bounds. Besides, except setting (e), as one of subsets is empty, we did not need auxiliary binary variables to construct the optimality cuts (11). We used both the primal and dual strategies for selecting the subsets S^1 and S^0, because each one has its own advantage.

The well-known variable-fixing technique is a combination of settings (a) and (b), thus it can be viewed as a special case of our optimality cuts. It was originated for MESP with [AFLW96, AFLW99] and has recently been applied to MESP in [Ans18, Ans20, CFL23]. These existing works do not consider optimizing the restricted Lagrangian dual problem to strengthen the upper bound, nor explore the subsets based on the primal continuous-relaxation solution. Figure 6 presents the number of fixed variables on two IEEE instances by using our strategy (i.e., the integration of settings (a) and (b)), compared to that of [CFL23]. Because both methods manage to fix s variables to one and the remaining $n - s$ variables to zero within one second, when $s \leq 4$ in 118-bus instance and $s \leq 31$ in 300-bus instances, we do not display these results.

In Figure 6, we see that the number of variables fixed to one tends to steadily increase and the number of variables fixed to zero decreases as s increases. From Figure 6(a) and Figure 6(b), we observe that our primal selection strategy successfully finds several optimal variables equal to one, but the variable-fixing by [CFL23] fails to fix any variable to one in our test instances. It is worth mentioning that the variables being fixed to one can significantly reduce the problem size and represent the best buses in power systems for installation of new PMU sensors. In Figure 6(c) and Figure 6(d), our dual strategy slightly outperforms that of [CFL23] when fixing variables to zero, and we are able to fix 13 more variables compared to that of [CFL23] for some cases. Finally, we compare the time for both methods in Figure 6(e) and Figure 6(f). We see that the overall performance of our primal and dual strategies is better than that of [CFL23], and our method takes more time but is still negligible compared to what it takes to solve DDF to optimality. We present the "Gap" in Figure 7, i.e., the difference between M-DDF continuous relaxation value $\widehat{z}_M$ and a lower bound returned by the local search Algorithm 1. The decrease of gaps precisely explains the sudden jumps in Figure 6(a)-Figure 6(d) at points of $s = 32, 63$, because the optimality gap used in Proposition 7 is a key factor in the variable-fixing performance, i.e., a lower gap implies more variable fixings.

For the other three settings (c), (d), and (e) that involve two binary variables, their number of optimality cuts and overall time are illustrated in Figure 8. We see that when s increases, the number of optimality cuts tends to increase, but as seen in Figure 6, the variable-fixing gets worse, implying the fact that optimality cuts can be complementary to each other. Thus, in order to ensure the effectiveness of optimality cuts in our algorithmic framework, we may need to try various types of optimality cuts.

5.3. Exact and approximation algorithms: IEEE 118- and 300-bus instances

We tested the two IEEE instances and found that with the benefit of submodular and optimality cuts, the LP/NLP B&B can efficiently solve DDF and approximation algorithms can find high-quality solutions within one minute.

In Table 2, we compare the time and the optimality gap of the LP/NLP B&B algorithm with and without submodular and optimality cuts. For LP/NLP B&B with submodular and optimality cuts, we also present the time to generate the optimality cuts. The time to generate the submodular cuts is negligible. We used the optimality cuts based on subsets from settings (a) and (b) (i.e., variable-fixing) to reduce the problem size of DDF, and we added the other optimality cuts from settings (c), (d), and (e) to the initial MILP relaxation M^0 solved in LP/NLP B&B. Columns "#a
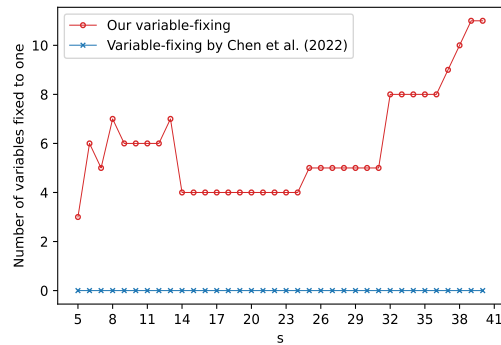
(a) Number of variables fixed at one for $n=117$
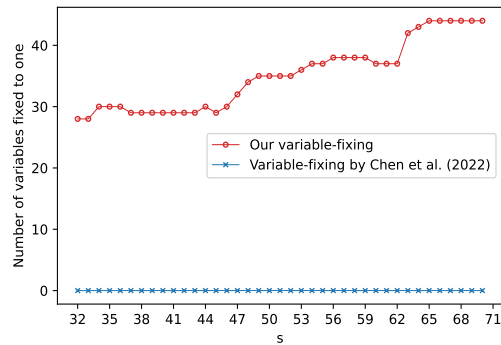
(b) Number of variables fixed at one for $n=299$
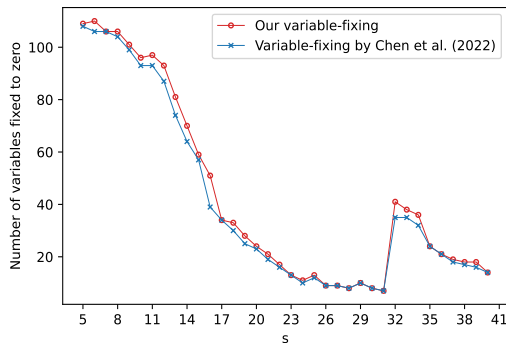
(c) Number of variables fixed at zero for $n=117$
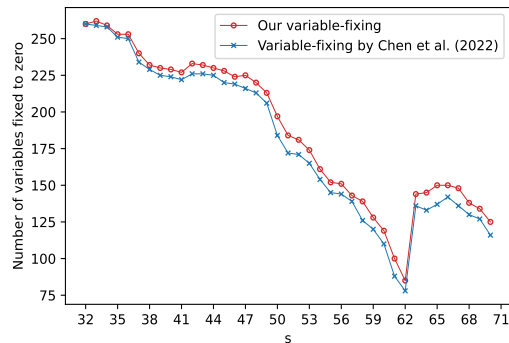
(d) Number of variables fixed at zero for $n=299$
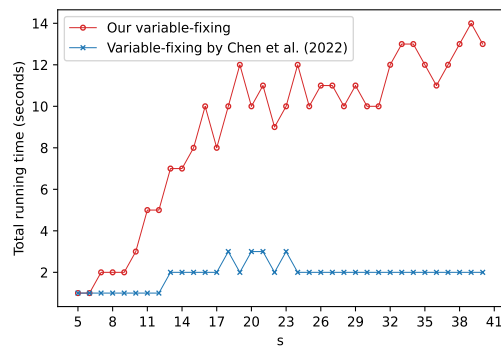
(e) Time for $n=117$
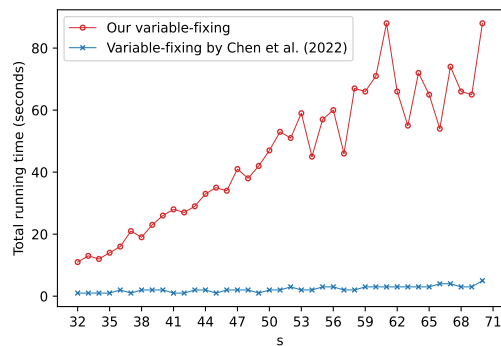
(f) Time for $n=299$

Figure 6 Comparison between variable-fixing techniques on IEEE instances

$- \#e$" present the number of optimality cuts corresponding to settings (a)$-$(e). For each test case, the time limit was set to four hours.

Just using submodular cuts, we can solve five more cases within the time limit, and even for the cases that are not solved to optimality, the MIPgap is dramatically reduced. When additionally, optimality cuts are included, all cases are solved to optimality within the time limit. For the four cases that took a substantial amount of time using only submodular cuts, with optimality cuts we could solve them much faster. We can see that many variables were fixed for all test cases using optimality cuts based on settings (a) and (b). According to [LNI11], it is difficult to compute the

(a) Gap for n=117

(b) Gap for n=299
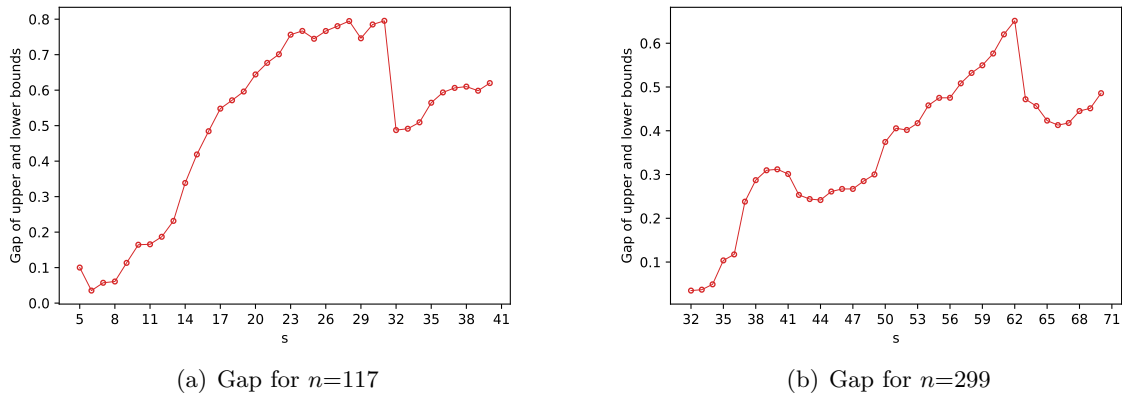
Figure 7   Gap between $\widehat{z}_M$ and the lower bound returned by Algorithm 1 on IEEE instances
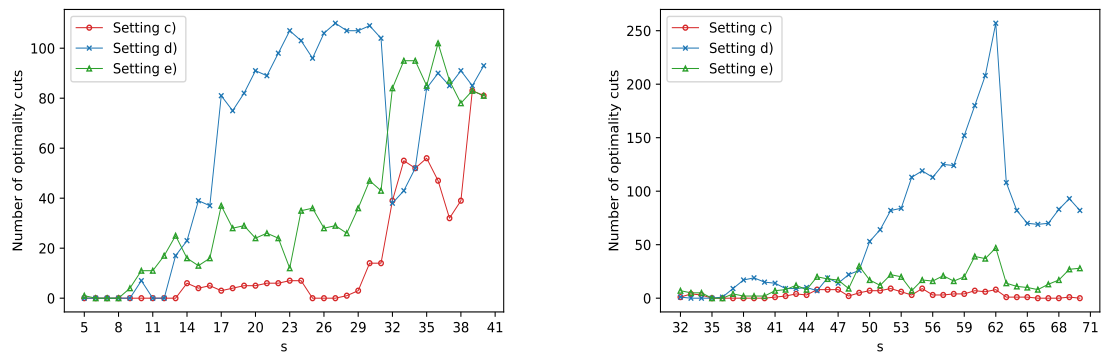
(a) Optimality cuts with two variables for n=117

(b) Optimality cuts with two variables for n=299
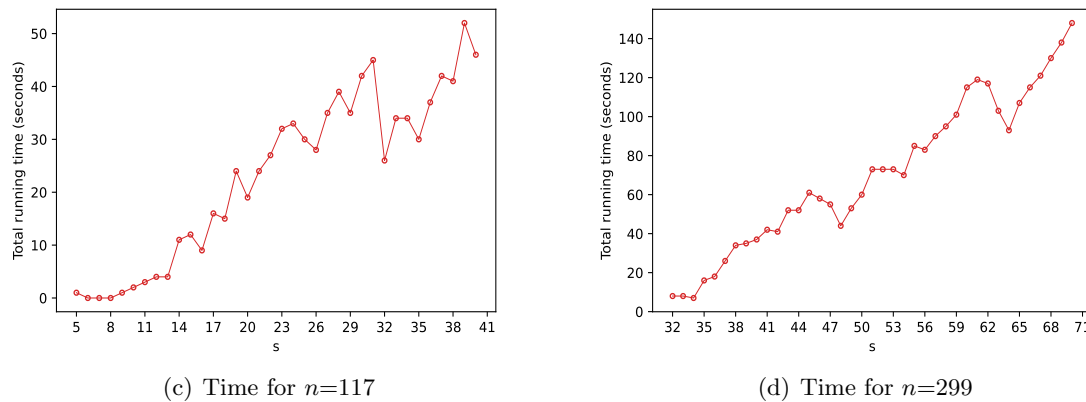
(c) Time for n=117

(d) Time for n=299

Figure 8   Performance of optimality cuts with two binary variables on IEEE instances

optimal value of DDF for a power system with more than 100 buses. However, our LP/NLP B&B using the submodular and optimality cuts enables us to effectively solve these cases to optimality. More details on this experiment are in Table EC.1, in e-companion EC.3.

Using the optimal values of DDF in Table 2, we evaluate the bounds given by the continuous relaxation of M-DDF and the performance of the approximation algorithms on the same testing

Table 2 Impact of submodular and optimality cuts on LP/NLP B&B on IEEE instances

n	s	LP/NLP B&B		LP/NLP B&B + submod. cuts		LP/NLP B&B + submod. and opt. cuts							
		MIPgap[1]	time[2]	MIPgap[1]	time[2]	MIPgap[1]	time[2]	#a	#b	#c	#d	#e	cut time[3]
117	5	0.00	11	0.00	1	0.00	2	3	109	0	0	1	2
117	10	0.00	423	0.00	5	0.00	5	6	96	0	7	11	5
117	15	3.18	-	0.00	1019	0.00	34	4	59	4	39	13	20
117	16	4.27	-	0.00	5368	0.00	58	4	51	5	37	16	19
117	17	5.19	-	0.61	-	0.00	344	4	34	3	81	37	24
117	18	4.81	-	1.34	-	0.00	626	4	33	4	75	28	25
117	19	8.47	-	2.16	-	0.00	1542	4	28	5	82	29	36
117	20	11.11	-	2.20	-	0.00	7111	4	24	5	91	24	29
299	35	7.76	-	0.00	33	0.00	31	30	253	0	0	0	30
299	40	83.52	-	0.00	505	0.00	69	29	229	0	15	2	63
299	45	48.06	-	0.00	1334	0.00	109	29	228	8	7	20	96
299	50	52.67	-	0.88	-	0.00	135	35	197	7	53	17	107
299	51	40.41	-	0.90	-	0.00	218	35	184	7	64	12	126
299	52	140.05	-	2.44	-	0.00	219	35	181	9	82	22	124
299	53	186.13	-	3.27	-	0.00	328	36	174	6	84	20	132
299	54	168.61	-	3.55	-	0.00	560	37	161	3	113	7	115
299	55	158.67	-	4.60	-	0.00	1240	37	152	9	119	17	142
299	56	143.81	-	5.29	-	0.00	5072	38	151	3	113	16	143
299	57	129.59	-	4.57	-	0.00	11462	38	143	3	125	21	136

[1] MIPgap = upper bound − best feasible-solution value (both obtained by LP/NLP B&B)

[2] total time in seconds; "-": instance not solved within four hours

[3] time to generate optimality cuts in seconds

cases. The computational results are displayed in Table 3. The "gap" for M-DDF is the difference between the continuous relaxation value $\widehat{z}_M$ and the optimal value z^*. The "gap" for the approximation algorithms is the difference between z^* and the solution returned by them. For comparison purposes, we also tested the greedy algorithm for solving the PMU placement problem in power systems studied in [LNI11, LCW$^+$12], whose implementations recalculate the objective function value at each iteration without exploring the rank-one update. We feed the continuous-relaxation solution of M-DDF to Algorithm 2, and thus the time for Algorithm 2 includes that of solving the continuous relaxation. We see that the upper bound given by the solution of the continuous-relaxation of M-DDF is always close to the optimal value. We also see that Algorithm 1 consistently gives better times and gaps than Algorithm 2. Although Algorithm 1 and the greedy algorithm have the same gap for each instance, in our implementations Algorithm 1 is much faster due to the rank-one updating technique.

5.4. Scalability and near-optimality of approximation algorithms: IEEE 118-, 300-, and Polish 2383-bus instances

To better understand the overall performance of our approximation algorithms, we tested cases with a full range of selected data points (i.e., s) on the two IEEE instances and on a larger instance with 2383 buses, one of the largest power systems in the literature (see [ZMSG97]). Note that LP/NLP B&B is unable to solve many of the large-s cases to optimality within a four-hour time limit. Thus, here we use the continuous-relaxation value $\widehat{z}_M$ of M-DDF to evaluate the quality of the feasible solutions obtained using the approximation algorithms. The value for "gap" in

Table 3   Relaxation and approximation algorithms on IEEE instances

n	s	M-DDF relaxation		local-search Algorithm 1		sampling Algorithm 2		greedy algorithm	
		gap	time[1]	gap	time[1]	gap	time[1]	gap	time[1]
117	5	0.10	< 1	0.00	< 1	0.00	5	0.00	< 1
117	10	0.16	1	0.00	< 1	0.00	5	0.00	< 1
117	15	0.42	1	0.00	< 1	0.08	6	0.00	< 1
117	16	0.48	1	0.00	< 1	0.32	6	0.00	< 1
117	17	0.55	1	0.00	< 1	0.17	6	0.00	< 1
117	18	0.57	2	0.00	< 1	0.23	7	0.00	< 1
117	19	0.60	1	0.00	< 1	0.79	6	0.00	< 1
117	20	0.64	2	0.00	< 1	0.87	6	0.00	< 1
299	35	0.11	6	0.00	< 1	0.00	34	0.00	14
299	40	0.31	11	0.00	< 1	2.04	46	0.00	13
299	45	0.26	5	0.00	< 1	4.40	41	0.00	12
299	50	0.37	5	0.00	< 1	3.98	37	0.00	11
299	51	0.41	7	0.00	< 1	2.25	39	0.00	12
299	52	0.40	6	0.00	< 1	4.18	45	0.00	11
299	53	0.42	8	0.00	< 1	4.36	45	0.00	12
299	54	0.46	10	0.00	< 1	3.56	38	0.00	12
299	55	0.47	10	0.00	< 1	2.29	46	0.00	12
299	56	0.47	14	0.00	< 1	7.41	42	0.00	13
299	57	0.46	11	0.05	< 1	5.07	46	0.05	17

[1] time in seconds; "< 1": time less than one second

Table 4 and Figure 9 is equal to the difference between $\widehat{z}_M$ and the output values produced by the approximation algorithm.

We compare our proposed approximation algorithms with the greedy algorithm in Figure 9, for which we test 23 cases with number of installed PMUs from $5, 10, \ldots, 115$ and 29 cases with number of installed PMUs from $10, 20, \ldots, 290$ for the two IEEE instances with 117 and 299 non-reference buses, respectively. We see that Algorithm 1 clearly outperforms the other two, considering the gaps and times. Besides, we see that the time of sampling Algorithm 2 decreases when s becomes close to n. This is because when $s \to n$ is large, the Frank-Wolfe converges quickly and takes less time to compute the M-DDF continuous relaxation value $\widehat{z}_M$, which serves as Step 1 of the sampling Algorithm 2 and accounts for a large portion of its running time. Such numerical phenomenon is supported by the complementary property of DDF. Specifically, it is expected that cases with small $s \to 1$ has the similar performances as those with large $s \to n$ due to the complement. In addition, the optimality gap of M-DDF continuous relaxation value $\widehat{z}_M$ starts to decrease with large s as shown in Figure 3.

In Table 4, we present results where we tested a large-scale instance with 2383 buses in the Polish power system (see [ZMSG97]), which admits $n = 2382$ non-reference buses. The greedy algorithm is omitted in this table because it could not finish on these cases within four hours. On the other hand, our proposed approximation algorithms scale well. It is evident that Algorithm 1 performs very well in time and solution quality, dominating the performance of Algorithm 2 in both respects. Thus, we recommend using Algorithm 1 (with an efficient implementation) to solve practical PMU placement problems. We note that the gaps for Algorithm 1 could be even smaller if we could compare to the optimal value. Another observation, is that the small gaps for Algorithm 1 establish the quality of the M-DDF relaxation on these cases.
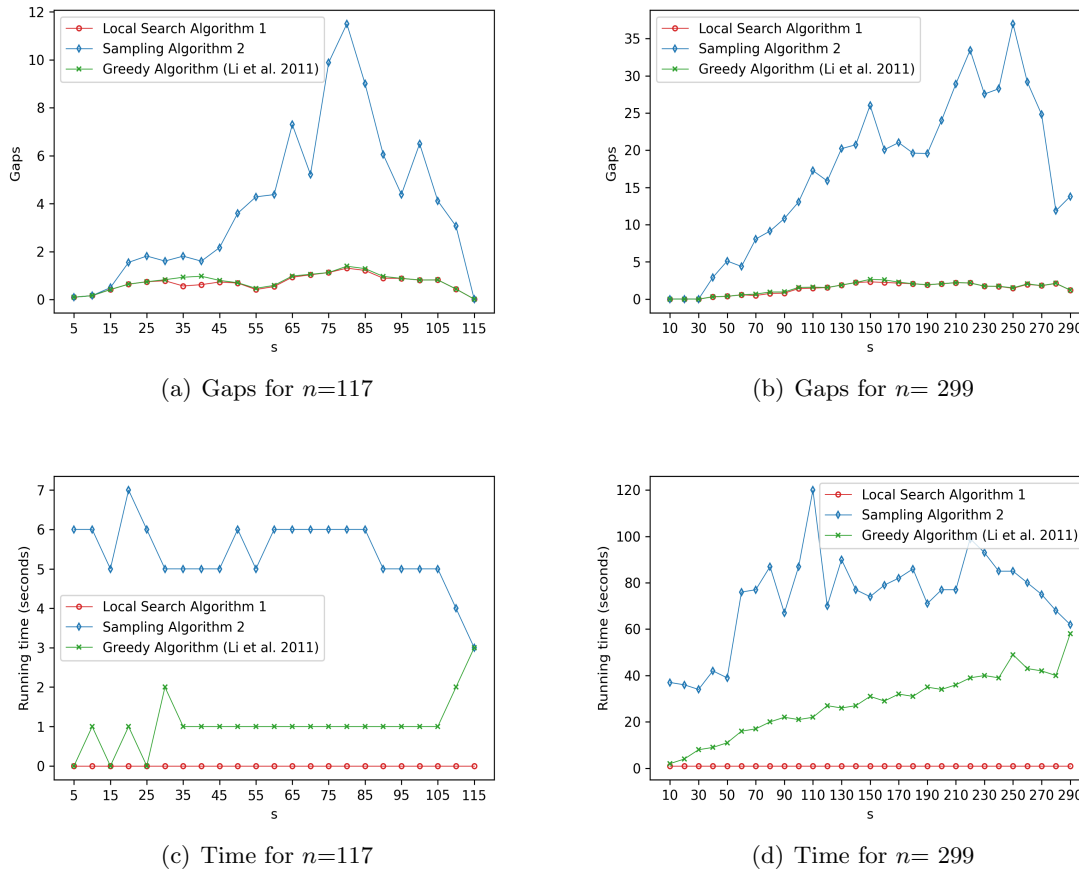
(a) Gaps for n=117

(b) Gaps for n= 299

(c) Time for n=117

(d) Time for n= 299

Figure 9   Comparison between approximation algorithms on IEEE instances

Table 4   Relaxation and approximation algorithms on the Polish instance

n	s	M-DDF relaxation		local-search Algorithm 1		sampling Algorithm 2	
		$\widehat{z}_M$	time[1]	gap	time[1]	gap	time[1]
2382	75	750.11	421	0.98	9	14.24	424
2382	100	1165.74	545	1.24	17	33.82	552
2382	125	1576.29	608	1.43	17	41.81	675
2382	150	1982.08	666	1.93	17	62.18	740
2382	175	2383.14	807	2.58	25	78.23	853
2382	200	2779.27	794	3.25	28	61.03	822
2382	225	3172.16	1064	4.08	31	93.11	1101
2382	250	3561.82	1301	4.95	37	105.06	1368
2382	275	3948.40	1333	5.15	46	122.23	1426
2382	300	4332.33	1244	5.41	47	135.21	1448
2382	325	4713.75	1293	6.11	45	130.45	1514
2382	350	5092.40	1392	6.92	49	149.43	1661
2382	375	5468.35	1510	8.02	51	123.27	1768

[1] time in seconds

6. Conclusion

We studied the D-optimal data fusion problem, which can be of vital importance in many fields, such as monitoring, operation, planning, control, and decision making of various environmental, structural, agricultural, food processing, and manufacturing systems. The developed exact and approximation algorithms come with theoretical performance guarantees. Our numerical study confirms the efficacy of the proposed algorithms. We expect the proposed methods can be applicable to many machine learning problems under a cardinality constraint such as sparse PCA, sparse regression, sparse matrix completion, and so on. Besides, the work in [NS16] provides an interesting geometric relaxation for MESP under more general matroid constraints. Thus, one possible future direction is to explore this relaxation for solving DDF under general settings.

Acknowledgments

References

[AA11]   Shabbir Ahmed and Alper Atamtürk. Maximizing a class of submodular utility functions. *Mathematical Programming*, 128(1):149–169, 2011.

[ABG+20]   Tobias Achterberg, Robert E Bixby, Zonghao Gu, Edward Rothberg, and Dieter Weninger. Presolve reductions in mixed integer programming. *INFORMS Journal on Computing*, 32(2):473–506, 2020.

[AFLW96]   Kurt M. Anstreicher, Marcia Fampa, Jon Lee, and Joy Williams. Continuous relaxations for constrained maximum-entropy sampling. In *Integer programming and Combinatorial Optimization (Vancouver, BC, 1996)*, volume 1084 of *Lecture Notes in Comput. Sci.*, pages 234–248. Springer, Berlin, 1996.

[AFLW99]   Kurt M. Anstreicher, Marcia Fampa, Jon Lee, and Joy Williams. Using continuous nonlinear relaxations to solve constrained maximum-entropy sampling problems. *Mathematical Programming, Series A*, 85(2):221–240, 1999.

[AKFFS09] Farrokh Aminifar, Amin Khodaei, Mahmud Fotuhi-Firuzabad, and Mohammad Shahidehpour. Contingency-constrained PMU placement in power networks. *IEEE Transactions on Power Systems*, 25(1):516–523, 2009.

[AL04]   Kurt M. Anstreicher and Jon Lee. A masked spectral bound for maximum-entropy sampling. In *mODa 7 — Advances in Model-Oriented Design and Analysis*, Contributions in Statistics, pages 1–12. Physica, Heidelberg, 2004.

[Ans18]   Kurt M Anstreicher. Maximum-entropy sampling and the Boolean quadric polytope. *Journal of Global Optimization*, 72(4):603–618, 2018.

[Ans20]   Kurt M Anstreicher. Efficient solution of maximum-entropy sampling problems. *Operations Research*, 68(6):1826–1835, 2020.

[BL07]   Samuel Burer and Jon Lee. Solving maximum-entropy sampling problems using factored masks. *Mathematical Programming*, 109(2-3, Ser. B):263–281, 2007.

[CFL23]   Zhongzhu Chen, Marcia Fampa, and Jon Lee. On computing with some convex relaxations for the maximum-entropy sampling problem. *INFORMS Journal on Computing*, 35(2):368–385, 2023.

[CFLL21]   Zhongzhu Chen, Marcia Fampa, Amélie Lambert, and Jon Lee. Mixing convex-optimization bounds for maximum-entropy sampling. *Mathematical Programming, Series B*, 188:539–568, 2021.

[CHB08]   Han-Lim Choi, Jonathan How, and Paul Barton. An outer-approximation algorithm for generalized maximum entropy sampling. In *Proc. of ACC 2008*, pages 1818–1823. IEEE, 2008.

[CHB13]   Han-Lim Choi, Jonathan How, and Paul Barton. An outer-approximation approach for information-maximizing sensor selection. *Optimization Letters*, 7:745–764, 2013.

[CMI13]   Ali Civril and Malik Magdon-Ismail. Exponential inapproximability of selecting a maximum volume sub-matrix. *Algorithmica*, 65(1):159–176, 2013.

[DLRCTP10] Jaime De La Ree, Virgilio Centeno, James S Thorp, and Arun G Phadke. Synchronized phasor measurement applications in power systems. *IEEE Transactions on Smart Grid*, 1(1):20–27, 2010.

[FL10]   Matteo Fischetti and Andrea Lodi. Heuristics in mixed integer programming. *Wiley Encyclopedia of Operations Research and Management Science*, 2010.

[FL22a]   Marcia Fampa and Jon Lee. *Maximum-Entropy Sampling: Algorithms and Application*. Springer, 2022.

[FL22b]   Marcia Fampa and Jon Lee. An outer-approximation algorithm for generalized maximum entropy sampling. In *Proc. of ISCO 2022*. Springer, Lecture Notes in Computer Science, 2022.

[He09]   Xiaofei He. Laplacian regularized d-optimal design for active learning and its application to image retrieval. *IEEE Transactions on Image Processing*, 19(1):254–263, 2009.

[HLW01]   Alan Hoffman, Jon Lee, and Joy Williams. New upper bounds for maximum-entropy sampling. In *mODa 6—Advances in Model-Oriented Design and Analysis (Puchberg/Schneeberg, 2001)*, Contrib. Statist., pages 143–153. Physica, Heidelberg, 2001.

[JH85]   Charles R Johnson and Roger A Horn. *Matrix Analysis*. Cambridge University Press, 1985.

[Joh74]   Charles R. Johnson. The Hadamard product of a and a^T. *Pacific Journal of Mathematics*, 51(2):477–481, 1974.

[KG11]   Vassilis Kekatos and Georgios B Giannakis. A convex relaxation approach to optimal placement of phasor measurement units. In *2011 4th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pages 145–148. IEEE, 2011.

[KLQ95]   Chun-Wa Ko, Jon Lee, and Maurice Queyranne. An exact algorithm for maximum entropy sampling. *Operations Research*, 43(4):684–691, 1995.

[LC21]   J Lapsins and S Cakula. Active machine learning in regression problems. In *2021 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, pages 1020–1023, 2021.

[LCW$^+$12] Qiao Li, Tao Cui, Yang Weng, Rohit Negi, Franz Franchetti, and Marija D Ilic. An information-theoretic approach to PMU placement in electric power systems. *IEEE Transactions on Smart Grid*, 4(1):446–456, 2012.

[Lee98]   Jon Lee. Constrained maximum-entropy sampling. *Operations Research*, 46(5):655–664, 1998.

[LNI11]   Qiao Li, Rohit Negi, and Marija D Ilić. Phasor measurement units placement for power system state estimation: A greedy approach. In *2011 IEEE Power and Energy Society General Meeting*, pages 1–8. IEEE, 2011.

[LW03]   Jon Lee and Joy Williams. A linear integer programming bound for maximum-entropy sampling. *Mathematical Programming, Series B*, 94(2–3):247–256, 2003.

[LX20]   Yongchun Li and Weijun Xie. Exact and approximation algorithms for sparse PCA. *Preprint:*, 2020. https://arXiv:2008.12438.

[LX23]   Yongchun Li and Weijun Xie. Best principal submatrix selection for the maximum entropy sampling problem: scalable algorithms and performance guarantees. *Operations Research*, 2023.

[MAL19]   Stanford Martinez, Adel Alaeddini, and Kristina Langer. A sequential weighted Laplacian-regularized optimal design for response surface modeling of expensive functions with outliers: An application in linear elastic fracture mechanics. *Quality and Reliability Engineering International*, 35(6):1911–1928, 2019.

[MFR20]   Wendel Melo, Marcia Fampa, and Fernanda Raupp. An overview of MINLP algorithms and their implementation in Muriqui Optimizer. *Annals of Operations Research*, 286:217–241, 2020.

[Mon00]   Alcir Monticelli. Electric power system state estimation. *Proceedings of the IEEE*, 88(2):262–282, 2000.

[MSTX19]   Vivek Madan, Mohit Singh, Uthaipon Tantipongpipat, and Weijun Xie. Combinatorial algorithms for optimal design. In *Conference on Learning Theory*, pages 2210–2258, 2019.

[Nik15]   Aleksandar Nikolov. Randomized rounding for the largest simplex problem. In *Proceedings of the Forty-Seventh Annual ACM Symposium on the Theory of Computing*, pages 861–870, 2015.

[NS16]   Aleksandar Nikolov and Mohit Singh. Maximizing determinants under partition constraints. In *Proceedings of the Forty-Eighth Annual ACM Symposium on the Theory of Computing*, pages 192–201, 2016.

[NST19]   Aleksandar Nikolov, Mohit Singh, and Uthaipon Tao Tantipongpipat. Proportional volume sampling and approximation algorithms for A-optimal design. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1369–1386. SIAM, 2019.

[Nuq01]   Reynaldo Francisco Nuqui. *State estimation and voltage security monitoring using synchronized phasor measurements*. PhD thesis, Virginia Polytechnic Institute and State University, 2001. https://vtechworks.lib.vt.edu/bitstream/handle/10919/28266/rnuqui_dissertation.pdf.

[NWF78]   George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions — I. *Mathematical Programming*, 14(1):265–294, 1978.

[Pal81]   Juni Palmgren. The fisher information matrix for log linear models arguing conditionally on observed explanatory variable. *Biometrika*, 68(2):563–566, 1981.

[PFL22]   Gabriel Ponte, Marcia Fampa, and Jon Lee. Exact and heuristic solution approaches for the D-optimality problem, 2022. Preprint.

[QG92]   Ignacio Quesada and Ignacio E. Grossmann. An LP/NLP based branch and bound algorithm for convex MINLP optimization problems. *Computers & Chemical Engineering*, 16(10–11):937–947, 1992.

[SW87]   Michael C Shewry and Henry P Wynn. Maximum entropy sampling. *Journal of Applied Statistics*, 14(2):165–170, 1987.

[SX20]   Mohit Singh and Weijun Xie. Approximation algorithms for D-optimal design. *Mathematics of Operations Research*, 45(4):1512–1534, 2020.

[Tan20]   Uthaipon Tantipongpipat. λ-regularized A-optimal design and its approximation by λ-regularized proportional volume sampling. *Preprint:*, 2020. `https://arXiv:2006.11182`.

[The17]   Nikolaos P Theodorakatos. Application of synchronized phasor measurements units in power systems. *The International Journal of Engineering and Science (IJES)*, 6(3):25–39, 2017.

[TVC$^+$10]   Vladimir Terzija, Gustavo Valverde, Deyu Cai, Pawel Regulski, Vahid Madani, John Fitch, Srdjan Skok, Miroslav M Begovic, and Arun Phadke. Wide-area monitoring, protection, and control of future electric power networks. *Proceedings of the IEEE*, 99(1):80–93, 2010.

[Var12]   Pramod K Varshney. *Distributed Detection and Data Fusion.* Springer, 2012.

[WN99]   Laurence A Wolsey and George L Nemhauser. *Integer and Combinatorial Optimization.* John Wiley & Sons, 1999.

[WP17]   Jing Wang and Eunsik Park. Active learning for penalized logistic regression via sequential experimental design. *Neurocomputing*, 222:183–190, 2017.

[YBT06]   Kai Yu, Jinbo Bi, and Volker Tresp. Active learning via transductive experimental design. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 1081–1088, 2006.

[YKBB13]   Chun Yang, Lance Kaplan, Erik Blasch, and Michael Bakich. Optimal placement of heterogeneous sensors for targets with Gaussian priors. *IEEE Transactions on Aerospace and Electronic Systems*, 49(3):1637–1653, 2013.

[ZCTP06]   Ming Zhou, Virgilio A Centeno, James S Thorp, and Arun G Phadke. An alternative for including phasor measurements in state estimators. *IEEE Transactions on Power Systems*, 21(4):1930–1937, 2006.

[Zha95]   Hong Zhang. Two-dimensional optimal sensor placement. *IEEE Transactions on Systems, Man, and Cybernetics*, 25(5):781–792, 1995.

[ZMSG97]   Ray D Zimmerman, Carlos E Murillo-Sánchez, and Deqiang Gan. Matpower: A MATLAB power system simulation package. *Manual, Power Systems Engineering Research Center, Ithaca NY*, 1997.

[ZMST10]   Ray Daniel Zimmerman, Carlos Edmundo Murillo-Sánchez, and Robert John Thomas. Matpower: Steady-state operations, planning, and analysis tools for power systems research and education. *IEEE Transactions on Power Systems*, 26(1):12–19, 2010.

[ZTW$^+$19]   Jiecheng Zhao, Jin Tan, Ling Wu, Lingwei Zhan, Wenxuan Yao, and Yilu Liu. Impact of the measurement errors on synchrophasor-based WAMS applications. *IEEE Access*, 7:143960–143972, 2019.

EC.1. Proofs

EC.1.1. Proof of Proposition 2

PROPOSITION 2. *The Lagrangian dual of the continuous relaxation of R-DDF is*

$$\widehat{z}_R := \operatorname{ldet} \boldsymbol{C} + \min_{\substack{\boldsymbol{\Lambda} \in \mathbb{S}^d_{++}, \\ \nu, \boldsymbol{\mu} \in \mathbb{R}^n_+}} \left\{ -\operatorname{ldet}(\boldsymbol{\Lambda}) + \operatorname{tr} \boldsymbol{\Lambda} + s\nu + \sum_{i \in [n]} \mu_i - d : \boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i \le \nu + \mu_i, \ i \in [n] \right\}. \tag{3}$$

Proof. First, we introduce an auxiliary matrix variable $\boldsymbol{X} \in \mathbb{S}^d_{++}$ and reformulate R-DDF as

$$\operatorname{ldet} \boldsymbol{C} + \max_{\boldsymbol{X} \in \mathbb{S}^d_{++}, \boldsymbol{x} \in \mathcal{Z}_s} \left\{ \operatorname{ldet}(\boldsymbol{X}) : \boldsymbol{I}_d + \sum_{i \in [n]} x_i \boldsymbol{b}_i \boldsymbol{b}_i^\top \succeq \boldsymbol{X} \right\}.$$

Then we derive the Lagrangian dual of the above maximization problem over $\boldsymbol{x}, \boldsymbol{X}$. Let $\boldsymbol{\Lambda} \in \mathbb{S}^d_{++}$, $\nu \in \mathbb{R}$, $\boldsymbol{v} \in \mathbb{R}^n_+$, $\boldsymbol{\mu} \in \mathbb{R}^n_+$ denote the Lagrangian multipliers. The Lagrangian function L is

$$L(\boldsymbol{x}, \boldsymbol{X}, \boldsymbol{\Lambda}, \nu, \boldsymbol{v}, \boldsymbol{\mu}) = \operatorname{ldet}(\boldsymbol{X}) + \operatorname{tr} \boldsymbol{\Lambda} - \operatorname{tr}(\boldsymbol{X}\boldsymbol{\Lambda}) + \sum_{i \in [n]} \left(\boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i - \nu + v_i - \mu_i \right) x_i + s\nu + \sum_{i \in [n]} \mu_i.$$

Maximizing L over $(\boldsymbol{x}, \boldsymbol{X})$ yields

$$\boldsymbol{\Lambda} = \boldsymbol{X}^{-1}, \ \boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i - \nu + v_i - \mu_i = 0, \forall i \in [n].$$

Then the Lagrangian dual problem can be obtained by plugging the above result into L, removing $\boldsymbol{v}$, and minimizing L over $(\boldsymbol{\Lambda}, \nu, \boldsymbol{\mu})$. $\square$

EC.1.2. Proof of Proposition 3

PROPOSITION 3. *For any $\boldsymbol{x} \in [0,1]^n$, the gradient $\boldsymbol{g} \in \mathbb{R}^n$ and the Hessian $\boldsymbol{H} \in \mathbb{R}^{n \times n}$ of the objective function in the continuous relaxation of R-DDF are*

$$\boldsymbol{g}(\boldsymbol{x}) := [\boldsymbol{b}_1^\top \boldsymbol{X}^{-1} \boldsymbol{b}_1, \cdots, \boldsymbol{b}_n^\top \boldsymbol{X}^{-1} \boldsymbol{b}_n], \quad \boldsymbol{H}(\boldsymbol{x}) := -(\boldsymbol{B}^\top \boldsymbol{X}^{-1} \boldsymbol{B}) \circ (\boldsymbol{B}^\top \boldsymbol{X}^{-1} \boldsymbol{B}), \ and \ \boldsymbol{H}(\boldsymbol{x}) \succeq -\delta^2 \boldsymbol{I}_n,$$

where $\boldsymbol{X} := \boldsymbol{I}_d + \sum_{i \in [n]} x_i \boldsymbol{b}_i \boldsymbol{b}_i^\top$ and $\delta := \lambda_{\max}(\boldsymbol{B}^\top \boldsymbol{B})$.

Proof. For any $\boldsymbol{Y} \in \mathbb{S}^d_{++}$, it is well-known that the function $\operatorname{ldet}(\boldsymbol{Y})$ has first- and second-order derivatives: $\boldsymbol{Y}^{-1}$ and $-\boldsymbol{Y}^{-1} \partial \boldsymbol{Y} \boldsymbol{Y}^{-1}$. Thus, the gradient and Hessian of the objective function over $\boldsymbol{x}$ in R-DDF can be derived.

According to [Joh74], the least eigenvalue of Hessian matrix $\boldsymbol{H}(\boldsymbol{x})$ satisfies

$$\lambda_{\min}(\boldsymbol{H}(\boldsymbol{x})) = -\lambda_{\max}(-\boldsymbol{H}(\boldsymbol{x})) \ge -\lambda^2_{\max}(\boldsymbol{B}^\top \boldsymbol{X}^{-1} \boldsymbol{B}) \ge -\lambda^2_{\max}(\boldsymbol{B}^\top \boldsymbol{B}),$$

where the last inequality is due to $\boldsymbol{X} \succeq \boldsymbol{I}_d$. $\square$

EC.1.3. Proof of Theorem 5

THEOREM 5. *Let $\widehat{S}$ denote the output of the local-search Algorithm 1, and let $\bar{s} := \min\{s, n-s\}$, then the set $\widehat{S}$ yields a $\min\{d\log(1+(\bar{s}/d)\sigma^2_{\max}/(1+\sigma_{\max})), \bar{s}\log(\bar{s})\}$-approximation bound for DDF, i.e.,*

$$\operatorname{ldet}\left(\boldsymbol{C} + \sum_{i \in \widehat{S}} \boldsymbol{a}_i \boldsymbol{a}_i^\top \right) \ge z^* - \min\left\{ d\log\left(1 + \frac{\bar{s}\sigma^2_{\max}}{d(1+\sigma_{\max})} \right), \bar{s}\log(\bar{s}) \right\},$$

where the constant $\sigma_{\max} := \max_{i \in [n]} \boldsymbol{a}_i^\top \boldsymbol{C}^{-1} \boldsymbol{a}_i$.

Proof. We will prove the two approximation bounds (i) $n \log(1 + (\bar{s}/n)\sigma_{\max}^2/(1 + \sigma_{\max}))$ and (ii) $\bar{s} \log(\bar{s})$, using R-DDF, M-DDF, and their complements, respectively.

(i) $d \log(1 + (\bar{s}/d)\sigma_{\max}^2/(1 + \sigma_{\max}))$-approximation bound.

Using the Lagrangian dual problem (3) of R-DDF (R-DDF), let us first show a non-symmetric approximation bound, $d \log(1 + (\bar{s}/d)\sigma_{\max}^2/(1 + \sigma_{\max}))$.

Given the output $\widehat{S}$ of the local-search Algorithm 1, let $\boldsymbol{X} = \sum_{\ell \in \widehat{S}} \boldsymbol{b}_\ell \boldsymbol{b}_\ell^\top$ and $\boldsymbol{\Lambda} = (\boldsymbol{I}_d + \boldsymbol{X})^{-1}$. Because $\widehat{S}$ is a locally-optimal solution of DDF, for any $i \in \widehat{S}$ and $j \in [n]\backslash\widehat{S}$, following the inequalities in (13), the local-optimality conditions can be written as

$$\boldsymbol{b}_j^\top \boldsymbol{\Lambda} \boldsymbol{b}_j - \boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i \boldsymbol{b}_j^\top \boldsymbol{\Lambda} \boldsymbol{b}_j + \boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_j \boldsymbol{b}_j^\top \boldsymbol{\Lambda} \boldsymbol{b}_i \le \boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i. \tag{EC.1}$$

Next, we will explore the inequality (EC.1) to construct a feasible solution to the Lagrangian dual problem (3) of the R-DDF. For any $i \in \widehat{S}$ and $j \in [n]\backslash\widehat{S}$, by dropping the nonnegative term $\boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_j \boldsymbol{b}_j^\top \boldsymbol{\Lambda} \boldsymbol{b}_i$ in the left-hand side of inequality (13), we obtain

$$\boldsymbol{b}_j^\top \boldsymbol{\Lambda} \boldsymbol{b}_j \le \frac{1}{1 - \boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i} \boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i. \tag{EC.2}$$

On the other hand, for each $i \in \widehat{S}$, we have

$$\boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i \le \boldsymbol{b}_i^\top (\boldsymbol{I}_d + \boldsymbol{b}_i \boldsymbol{b}_i^\top)^{-1} \boldsymbol{b}_i = \frac{\boldsymbol{b}_i^\top \boldsymbol{b}_i}{1 + \boldsymbol{b}_i^\top \boldsymbol{b}_i} = \frac{\boldsymbol{a}_i^\top \boldsymbol{C}^{-1} \boldsymbol{a}_i}{1 + \boldsymbol{a}_i^\top \boldsymbol{C}^{-1} \boldsymbol{a}_i} \le \frac{\sigma_{\max}}{1 + \sigma_{\max}}, \tag{EC.3}$$

where the last inequality is due to $\sigma_{\max} := \max_{i \in [n]} \boldsymbol{a}_i^\top \boldsymbol{C}^{-1} \boldsymbol{a}_i$ and the fact that the function $h(t) = t/(1 + t)$ is non-decreasing in $t \in \mathbb{R}_+$.

Combining the results (EC.2) and (EC.3), it follows that

$$\boldsymbol{b}_j^\top \boldsymbol{\Lambda} \boldsymbol{b}_j \le \frac{1}{1 - \boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i} \boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i \le (1 + \sigma_{\max})\boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i, \forall i \in \widehat{S}, \forall j \in [n] \setminus \widehat{S}, \tag{EC.4}$$

where the second inequality is due to non-decreasing of $h(t) = 1/(1 - t)$ over $t \in [0, 1)$.

Let us denote $\beta_{\min} := \min_{i \in \widehat{S}} \boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i$. Then, according to the inequality (EC.4), a feasible solution to the Lagrangian dual problem (3) can be constructed by

$$\widehat{\boldsymbol{\Lambda}} = t\boldsymbol{\Lambda}, \widehat{\nu} = t(1 + \sigma_{\max})\beta_{\min}, \widehat{\mu}_i = t(1 + \sigma_{\max})\boldsymbol{b}_i^\top \boldsymbol{\Lambda} \boldsymbol{b}_i - \widehat{\nu}, \forall i \in \widehat{S}, \widehat{\mu}_i = 0, \forall i \in [n] \setminus \widehat{S},$$

where $t > 0$ is a scalar and will be specified later.

Plugging solution $(\widehat{\boldsymbol{\Lambda}}, \widehat{\nu}, \widehat{\boldsymbol{\mu}})$ to problem (3), the objective value with the scalar t satisfies

$$\widehat{z}_R \le \min_t \{\text{ldet}\,\boldsymbol{C} - \text{ldet}\,\boldsymbol{\Lambda} - d\log(t) + t\,\text{tr}\,\boldsymbol{\Lambda} + t(1 + \sigma_{\max})\,\text{tr}(\boldsymbol{X}\boldsymbol{\Lambda}) - d\}$$
$$= \min_t \{\text{ldet}\,\boldsymbol{C} - \text{ldet}\,\boldsymbol{\Lambda} - d\log(t) + td + t\sigma_{\max}\,\text{tr}(\boldsymbol{X}\boldsymbol{\Lambda}) - d\},$$

where the equation is from the fact $\text{tr}((\boldsymbol{I}_d + \boldsymbol{X})\boldsymbol{\Lambda}) = d$.

Minimizing the left-hand side above over t, the optimal scalar is $t^* = d/(d + \sigma_{\max}\,\text{tr}(\boldsymbol{X}\boldsymbol{\Lambda}))$ and thus the final objective value with t^* of problem (3) is equal to

$$z^* \le \widehat{z}_R \le \text{ldet}\,\boldsymbol{C} + \text{ldet}(\boldsymbol{I}_d + \boldsymbol{X}) + d\log(1 + \sigma_{\max}/d\,\text{tr}(\boldsymbol{X}\boldsymbol{\Lambda}))$$
$$\le \text{ldet}\,\boldsymbol{C} + \text{ldet}(\boldsymbol{I}_d + \boldsymbol{X}) + d\log\left(1 + \frac{\sigma_{\max}}{d}\frac{s\sigma_{\max}}{1 + \sigma_{\max}}\right),$$

where the first inequality is from Proposition 2 and the last one is due to inequality (EC.3).

Now we use the Lagrangian dual problem (4) of R-DDF-comp to establish a complementary approximation bound of the previous one, $d\log(1 + (n - s)/d\sigma_{\max}^2/(1 + \sigma_{\max}))$. Because the

objective function of R-DDF-comp is $\mathrm{ldet}(\boldsymbol{I}_d - \sum_{i\in[n]\setminus\widehat{S}}\boldsymbol{q}_i\boldsymbol{q}_i^\top)$, for any $i \in [n]\setminus\widehat{S}$ and $j\in\widehat{S}$, the local-optimality condition becomes

$$\det(\boldsymbol{I}_d + \boldsymbol{X}) \geq \det(\boldsymbol{I}_d + \boldsymbol{X} + \boldsymbol{q}_i\boldsymbol{q}_i^\top - \boldsymbol{q}_j\boldsymbol{q}_j^\top) \Longleftrightarrow \boldsymbol{q}_j^\top\boldsymbol{\Lambda}\boldsymbol{q}_j - \boldsymbol{q}_i^\top\boldsymbol{\Lambda}\boldsymbol{q}_j\boldsymbol{q}_j^\top\boldsymbol{\Lambda}\boldsymbol{q}_i \geq \boldsymbol{q}_i^\top\boldsymbol{\Lambda}\boldsymbol{q}_i(1 - \boldsymbol{q}_j^\top\boldsymbol{\Lambda}\boldsymbol{q}_j),$$

where we define $\boldsymbol{X} := -\sum_{\ell\in[n]\setminus\widehat{S}}\boldsymbol{q}_\ell\boldsymbol{q}_\ell^\top$ and $\boldsymbol{\Lambda} := (\boldsymbol{I}_d + \boldsymbol{X})^{-1}$ here.

Then, by dropping the term $\boldsymbol{q}_i^\top\boldsymbol{\Lambda}\boldsymbol{q}_j\boldsymbol{q}_j^\top\boldsymbol{\Lambda}\boldsymbol{q}_i$ above, we obtain

$$-\boldsymbol{q}_j^\top\boldsymbol{\Lambda}\boldsymbol{q}_j \leq -\boldsymbol{q}_i^\top\boldsymbol{\Lambda}\boldsymbol{q}_i(1 - \boldsymbol{q}_j^\top\boldsymbol{\Lambda}\boldsymbol{q}_j) \leq -\boldsymbol{q}_i^\top\boldsymbol{\Lambda}\boldsymbol{q}_i\frac{1}{1 + \sigma_{\max}}, \forall i \in [n]\setminus\widehat{S}, \forall j\in\widehat{S}, \tag{EC.5}$$

where the second inequality is from by plugging the expressions of $\boldsymbol{q}_j$ and $\boldsymbol{q}_i$, i.e.,

$$\boldsymbol{q}_j\boldsymbol{\Lambda}\boldsymbol{q}_j = \boldsymbol{a}_j^\top(\boldsymbol{C} + \boldsymbol{A}\boldsymbol{A}^\top)^{-\frac{1}{2}}\left[\boldsymbol{I}_d - (\boldsymbol{C} + \boldsymbol{A}\boldsymbol{A}^\top)^{-\frac{1}{2}}\boldsymbol{A}_{[n]\setminus\widehat{S}}\boldsymbol{A}_{[n]\setminus\widehat{S}}^\top(\boldsymbol{C} + \boldsymbol{A}\boldsymbol{A}^\top)^{-\frac{1}{2}}\right]^{-1}$$

$$(\boldsymbol{C} + \boldsymbol{A}\boldsymbol{A}^\top)^{-\frac{1}{2}}\boldsymbol{a}_j = \boldsymbol{a}_j^\top\left(\boldsymbol{C} + \boldsymbol{A}\boldsymbol{A}^\top - \boldsymbol{A}_{[n]\setminus\widehat{S}}\boldsymbol{A}_{[n]\setminus\widehat{S}}^\top\right)^{-1}\boldsymbol{a}_j \leq \boldsymbol{a}_j^\top(\boldsymbol{C} + \boldsymbol{a}_j\boldsymbol{a}_j^\top)^{-1}\boldsymbol{a}_j \leq \frac{\sigma_{\max}}{1 + \sigma_{\max}}.$$

Using inequality (EC.5), we can construct a feasible solution to the Lagrangian dual problem (4) as below

$$\widehat{\boldsymbol{\Lambda}} = t\boldsymbol{\Lambda}, \widehat{\nu} = t\frac{1}{1 + \sigma_{\max}}\beta_{\min}^c, \widehat{\mu}_i = -t\frac{1}{1 + \sigma_{\max}}\boldsymbol{q}_i^\top\boldsymbol{\Lambda}\boldsymbol{q}_i - \widehat{\nu}, \forall i \in [n]\setminus\widehat{S}, \widehat{\mu}_i = 0, \forall i\in\widehat{S},$$

where $\beta_{\min}^c := -\max_{i\in[n]\setminus\widehat{S}}\boldsymbol{q}_i^\top\boldsymbol{\Lambda}\boldsymbol{q}_i$.

Analogous to analyzing the previous approximation bound, we calculate the optimal scalar $t^* = d/[d - \sigma_{\max}/(1 + \sigma_{\max})\,\mathrm{tr}(\boldsymbol{X}\boldsymbol{\Lambda})]$ and plugging the optimal scalar, the objective value of problem (4) satisfies

$$z^* \leq \widehat{z}_R \leq \mathrm{ldet}(\boldsymbol{C} + \boldsymbol{A}\boldsymbol{A}^\top) + \mathrm{ldet}(\boldsymbol{I}_d + \boldsymbol{X}) + d\log\left(1 - \frac{\sigma_{\max}}{d(1 + \sigma_{\max})}\,\mathrm{tr}(\boldsymbol{X}\boldsymbol{\Lambda})\right)$$

$$= \mathrm{ldet}(\boldsymbol{C} + \boldsymbol{A}\boldsymbol{A}^\top) + \mathrm{ldet}(\boldsymbol{I}_d + \boldsymbol{X}) + d\log\left(1 + \frac{\sigma_{\max}}{d(1 + \sigma_{\max})}\sum_{i\in[n]\setminus\widehat{S}}\boldsymbol{q}_i^\top\boldsymbol{\Lambda}\boldsymbol{q}_i\right)$$

$$\leq \mathrm{ldet}\left(\boldsymbol{C} + \sum_{i\in\widehat{S}}\boldsymbol{a}_i\boldsymbol{a}_i^\top\right) + d\log\left(1 + \frac{(n-s)\sigma_{\max}^2}{d(1 + \sigma_{\max})}\right),$$

where the last inequality is because for any $i \in [n]\setminus\widehat{S}$, we have

$$\boldsymbol{q}_i\boldsymbol{\Lambda}\boldsymbol{q}_i = \boldsymbol{a}_i^\top\left(\boldsymbol{C} + \boldsymbol{A}\boldsymbol{A}^\top - \boldsymbol{A}_{[n]\setminus\widehat{S}}\boldsymbol{A}_{[n]\setminus\widehat{S}}^\top\right)^{-1}\boldsymbol{a}_i \leq \boldsymbol{a}_i^\top\boldsymbol{C}^{-1}\boldsymbol{a}_i \leq \sigma_{\max}.$$

Minimizing the two approximation bounds gives us the symmetric one, $d\log(1 + \frac{\bar{s}\sigma_{\max}^2}{d(1 + \sigma_{\max})})$.

(ii) $\bar{s}\log(\bar{s})$-approximation bound.

According to M-DDF, the output value of the local-search Algorithm 1 becomes

$$z^* \leq \widehat{z}_M \leq \log f\left(\sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top\right) + \mathrm{ldet}(\boldsymbol{C}) + s\min\left\{\log(s), \log\left(n - s - \frac{n}{s} + 2\right)\right\},$$

where the inequality results from [**?**][Theorem 7].

Using the complementary M-DDF, we can also show

$$z^* \leq \widehat{z}_M^c \leq \log f\left(\sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top\right) + \mathrm{ldet}(\boldsymbol{C}) + (n-s)\min\left\{\log(n-s), \log\left(s - \frac{n}{n-s} + 2\right)\right\}.$$

Taking the minimum of the two bounds above and using the identity $\log f(\sum_{i\in\widehat{S}}\boldsymbol{v}_i\boldsymbol{v}_i^\top) + \mathrm{ldet}(\boldsymbol{C}) = \mathrm{ldet}(\boldsymbol{C} + \sum_{i\in\widehat{S}}\boldsymbol{a}_i\boldsymbol{a}_i^\top)$ complete the proof.

$\square$

EC.1.4. Proof of Theorem 6

THEOREM 6. *Let $\widehat{\boldsymbol{x}}_D, \widehat{\boldsymbol{x}}_M, \widehat{\boldsymbol{x}}_M^c$ denote optimal continuous-relaxation solutions of R-DDF, M-DDF, and M-DDF-comp, respectively. Suppose that Algorithm 2 generates random sets $\tilde{S}_D$, $\tilde{S}_M$, $\tilde{S}_{M^c}$ with $\widehat{\boldsymbol{x}}_D, \widehat{\boldsymbol{x}}_M, \widehat{\boldsymbol{x}}_M^c$ as inputs, respectively, then we have*

(i) $\log \mathbb{E}\left[\det\left(\boldsymbol{C} + \sum_{i \in \tilde{S}_D} \boldsymbol{a}_i \boldsymbol{a}_i^\top\right)\right] \geq z^* + n \log(x_{\min}) - (n-s)\log(1+\delta)$, *where* $x_{\min} > 0$ *is the least non-zero entry in* $\widehat{\boldsymbol{x}}_D$ *and* $\delta := \lambda_{\max}(\boldsymbol{A}^\top \boldsymbol{C}^{-1}\boldsymbol{A})$;

(ii) $\log \mathbb{E}\left[\det\left(\boldsymbol{C} + \sum_{i \in \tilde{S}_M} \boldsymbol{a}_i \boldsymbol{a}_i^\top\right)\right] \geq z^* - s \log\left(\frac{s}{n}\right) - \log\left(\binom{n}{s}\right)$;

(iii) $\log \mathbb{E}\left[\det\left(\boldsymbol{C} + \sum_{i \in \tilde{S}_M^c} \boldsymbol{a}_i \boldsymbol{a}_i^\top\right)\right] \geq z^* - (n-s) \log\left(\frac{n-s}{n}\right) - \log\left(\binom{n}{n-s}\right)$;

(iv) *Algorithm 2 can be derandomized as a polynomial-time algorithm with the same performance guarantees.*

To facilitate the analysis of approximation bounds, let us first introduce the elementary symmetric polynomial function and its relation with eigenvalues.

DEFINITION EC.1 (ELEMENTARY SYMMETRIC POLYNOMIAL). For a vector $\boldsymbol{y} \in \mathbb{R}^n$ and an integer $k \in [n]$, let $e_k(\boldsymbol{y})$ denote the degree k elementary symmetric polynomial, i.e.,

$$e_k(\boldsymbol{y}) := \sum_{S \in \binom{[n]}{k}} \prod_{i \in S} y_i.$$

For any symmetric matrix $\boldsymbol{X} \in \mathbb{S}^n$ with its eigenvalue vector $\boldsymbol{\lambda} \in \mathbb{R}^n$ and an integer $k \in [n]$, it is well-known that $e_k(\boldsymbol{\lambda})$ is equal to

$$e_k(\boldsymbol{\lambda}) = \sum_{S \in \binom{[n]}{k}} \det(\boldsymbol{X}_{S,S}). \tag{EC.6}$$

Now we are ready to prove Theorem 6.

Proof. The proof can be split into four parts, corresponding to three approximation bounds and derandomization.

Part (i) Let $T := \mathrm{supp}(\widehat{\boldsymbol{x}}_D)$ be the index set of nonzero entries in $\widehat{\boldsymbol{x}}_D$.

For notational convenience, we define two matrices $\boldsymbol{X} \in \mathbb{S}_+^n$ and $\boldsymbol{Y} \in \mathbb{S}_+^n$ as

$$\boldsymbol{X} := \mathrm{Diag}(\widehat{\boldsymbol{x}}_D) + \boldsymbol{Y}, \quad \boldsymbol{Y} := \mathrm{Diag}^{\frac{1}{2}}(\widehat{\boldsymbol{x}}_D) \boldsymbol{B}^\top \boldsymbol{B} \, \mathrm{Diag}^{\frac{1}{2}}(\widehat{\boldsymbol{x}}_D).$$

For each $i \in [n] \setminus T$, we can show that the i-th column and row vectors of $\boldsymbol{X}$ and $\boldsymbol{Y}$ consist of zeros as $(\widehat{x}_D)_i = 0$. It follows that both $\boldsymbol{X}$ and $\boldsymbol{Y}$ have rank at most $|T|$. Thus, we let $\lambda_1 \geq \cdots \geq \lambda_{|T|} \geq 0 = \lambda_{|T|+1} = \cdots = \lambda_n$ and $\beta_1 \geq \cdots \geq \beta_{|T|} \geq 0 = \beta_{|T|+1} = \cdots = \beta_n$ denote the eigenvalues of $\boldsymbol{X}$ and $\boldsymbol{Y}$, respectively. Furthermore, according to Weyl's inequalities and the fact that $0 < x_{\min} \leq 1$, two eigenvalue vectors $\boldsymbol{\lambda}$ and $\boldsymbol{\beta}$ satisfy

$$\lambda_i \geq x_{\min} + \beta_i \geq x_{\min}(1 + \beta_i), \forall i = 1, \cdots, |T|. \tag{EC.7}$$

Next, the exponential expectation of the objective value of R-DDF is equal to

$$\mathbb{E}\left[\det\left(\boldsymbol{I}_d + \sum_{i \in \tilde{S}_D} \boldsymbol{b}_i \boldsymbol{b}_i^\top\right)\right] = \sum_{S \in \binom{[n]}{s}} \mathbb{P}[\tilde{S}_D = S] \det\left(\boldsymbol{I}_d + \sum_{i \in S} \boldsymbol{b}_i \boldsymbol{b}_i^\top\right)$$

$$= \frac{1}{\sum_{\bar{S} \in \binom{[n]}{s}} \prod_{i \in \bar{S}} (\widehat{x}_D)_i} \sum_{S \in \binom{[n]}{s}} \prod_{i \in S} (\widehat{x}_D)_i \det\left(\boldsymbol{I}_n + \boldsymbol{B}^\top \boldsymbol{B}\right)_{S,S}$$

$$= \frac{1}{e_s(\widehat{\boldsymbol{x}}_D)} \sum_{S \in \binom{[n]}{s}} \det(\boldsymbol{X}_{S,S}) = \frac{1}{e_s(\widehat{\boldsymbol{x}}_D)} e_s(\boldsymbol{\lambda}) = \frac{1}{e_s(\widehat{\boldsymbol{x}}_D)} \lambda_1^s e_s\left(\frac{1}{\lambda_1}\boldsymbol{\lambda}\right)$$

$$\geq \frac{1}{e_s(\widehat{\boldsymbol{x}}_D)} \lambda_1^s \binom{|T|}{s} e_{|T|}\left(\frac{1}{\lambda_1}\boldsymbol{\lambda}\right) = \frac{1}{e_s(\widehat{\boldsymbol{x}}_D)} \lambda_1^{s-|T|} \binom{|T|}{s} e_{|T|}(\boldsymbol{\lambda})$$

$$= \frac{1}{e_s(\widehat{\boldsymbol{x}}_D)} \lambda_1^{s-|T|} \binom{|T|}{s} e_n\left([\lambda_1, \cdots, \lambda_{|T|}, 1, \cdots, 1]^\top\right)$$

$$\geq \frac{1}{e_s(\widehat{\boldsymbol{x}}_D)} \lambda_1^{s-|T|} \binom{|T|}{s} e_n\left([x_{\min}(\beta_1 + 1), \cdots, x_{\min}(1 + \beta_{|T|}), 1, \cdots, 1]^\top\right)$$

$$\geq \frac{1}{\left(\frac{s}{|T|}\right)^s \binom{|T|}{s}} \lambda_1^{s-|T|} \binom{|T|}{s} x_{\min}^{|T|} e_n(\boldsymbol{1} + \boldsymbol{\beta}) = \left(\frac{|T|}{s}\right)^s \lambda_1^{s-|T|} x_{\min}^{|T|} \det(\boldsymbol{I}_n + \boldsymbol{Y})$$

$$= \left(\frac{|T|}{s}\right)^s \lambda_1^{s-|T|} x_{\min}^{|T|} \det(\boldsymbol{I}_d + \boldsymbol{B}\,\mathrm{Diag}(\widehat{\boldsymbol{x}}_D)\boldsymbol{B}^\top)$$

$$\geq \left(\frac{|T|}{s}\right)^s x_{\min}^n \lambda_1^{s-|T|} \exp(\widehat{z}_R - \mathrm{ldet}\,\boldsymbol{C})$$

$$\geq \left(\frac{|T|}{s}\right)^s x_{\min}^n \lambda_1^{s-|T|} \exp(z^* - \mathrm{ldet}\,\boldsymbol{C}),$$

where the first inequality stems from the fact that each element of vector $\frac{1}{\lambda_1}\boldsymbol{\lambda}$ is no larger than 1, the sixth equality is due to $\lambda_i = 0$ for all $i \in [|T| + 1, n]$, the second inequality is from (EC.7), the third inequality is obtained by Maclaurin's inequality, the last equation is from the fact that matrices $\boldsymbol{Y}$ and $\boldsymbol{B}\,\mathrm{Diag}(\widehat{\boldsymbol{x}}_D)\boldsymbol{B}^\top$ have the same nonzero eigenvalues, the fourth inequality is because $\widehat{\boldsymbol{x}}_D$ is an optimal solution and $x_{\min} \leq 1$, and the last inequality is due to $\widehat{z}_R \geq z^*$.

Taking logarithm on both sides of the inequality above, we obtain

$$\mathrm{ldet}\,\boldsymbol{C} + \log \mathbb{E}\left[\det\left(\boldsymbol{I}_n + \sum_{i \in \tilde{S}_D} \boldsymbol{b}_i \boldsymbol{b}_i^\top\right)\right] \geq z^* + n \log(x_{\min}) + (s - |T|)\log(\lambda_1) - s \log\left(\frac{s}{|T|}\right)$$

$$\geq z^* + n \log(x_{\min}) - (n - s)\log(1 + \lambda_{\max}(\boldsymbol{B}^\top \boldsymbol{B})),$$

where the inequality is because $\lambda_1 \leq 1 + \beta_1 \leq 1 + \lambda_{\max}(\boldsymbol{B}^\top \boldsymbol{B})$.

Part (ii) The objective value led by the output $\tilde{S}_M$ of Algorithm 2 with $\widehat{\boldsymbol{x}} = \widehat{\boldsymbol{x}}_M$ is bounded by

$$\log \mathbb{E}\left[f\left(\sum_{i \in \tilde{S}_M} \boldsymbol{v}_i \boldsymbol{v}_i^\top\right)\right] + \mathrm{ldet}(\boldsymbol{C}) \geq \widehat{z}_M - s \log\left(\frac{s}{n}\right) - \log\left(\binom{n}{s}\right),$$

where the inequality is from [**?**, Theorem 5] and using the fact $\widehat{z}_M \geq z^*$, we obtain the approximation bound.

Part (iii) Given the output $\tilde{S}_M^c$ of Algorithm 2 with $\widehat{\boldsymbol{x}} = \widehat{\boldsymbol{x}}_M^c$, the objective value of Complementary DDF satisfies

$$\log \mathbb{E}\left[f\left(\sum_{i \in \tilde{S}_M^c} \boldsymbol{v}_i \boldsymbol{v}_i^\top\right)\right] + \mathrm{ldet}(\boldsymbol{C}) \geq \widehat{z}_M - (n - s)\log\left(\frac{n - s}{n}\right) - \log\left(\binom{n}{n - s}\right),$$

where the inequality is from [**?**, Theorem 5] and using the fact $\widehat{z}_M \geq z^*$, we obtain the approximation bound.

Part (iv) According the proof of Theorem 3, for any cardinality-s subset $S \subseteq [n]$, we must have

$$\det\left(\boldsymbol{C} + \sum_{i \in S} \boldsymbol{a}_i \boldsymbol{a}_i^\top\right) = f\left(\sum_{i \in [n]} x_i \boldsymbol{v}_i \boldsymbol{v}_i^\top\right).$$

The remainder of the proof follows the de-randomization procedure of [**?**, Thm. 6]. $\qquad\square$

EC.2. NP-Hardness of sensor fusion in power systems

In this section, we show that the sensor fusion problem in power systems, as a special case of DDF, is still NP-hard. In this case, plugging the FIM (14) into DDF exactly formulates the sensor fusion problem as

$$z^* := \max_{S \subseteq [n], |S|=s} \mathrm{ldet}\left(\boldsymbol{C} + \sum_{i \in S} \frac{1}{\widehat{\sigma}_i^2} \boldsymbol{e}_i \boldsymbol{e}_i^\top\right), \tag{EC.8}$$

where $\boldsymbol{C} \in \mathbb{S}_{++}^n$ and $\boldsymbol{e}_i \in \mathbb{R}^n$ denotes the ith column of matrix $\boldsymbol{I}_n$.

PROPOSITION EC.1. *Even when all standard deviations $\{\widehat{\sigma}_i > 0\}_i$ are equal, the problem (EC.8), as a special case of our DDF, is NP-hard.*

Proof. Suppose that $\widehat{\sigma}_i = \widehat{\sigma}$ holds for all $i \in [n]$. We will show that there is $\widehat{\sigma} > 0$ such that the problem (EC.8) is NP-hard.

As shown in Theorem 2, the problem (EC.8) is equivalent to

$$z^* := \max_{S \subseteq [n], |S|=s} \mathrm{ldet}\left(\boldsymbol{C} + \sum_{i \in S} \frac{1}{\widehat{\sigma}_i^2} \boldsymbol{e}_i \boldsymbol{e}_i^\top\right) = \mathrm{ldet}\,\boldsymbol{C} + \max_{S \subseteq [n], |S|=s} \mathrm{ldet}\left(\boldsymbol{I}_n + \frac{1}{\widehat{\sigma}^2} \boldsymbol{C}^{-1}\right)_{S,S}.$$

Then following the proof of [KLQ95][theorem 1], one can find proper $\widehat{\sigma} > 0$ and $\boldsymbol{C}^{-1} \in \mathbb{S}_{++}^n$ to reduce the problem (EC.8) to the well-known NP-hard stable set problem. □

EC.3. Supplementary numerical results

As a supplement of Table 2, we present in Table EC.1 detailed numerical results for LP/NLP B&B with and without submodular cuts. In particular, we directly solve the maximization problem in M-DDF or R-DDF and thus the optimal value returned by the B&B is equal to the difference of z^* and the constant $\mathrm{ldet}(\boldsymbol{C})$. The results expose the quality of the lower bound (LB) and the upper bound (UB) computed by LP/NLP B&B when the instances are not solved to optimality.

Table EC.1   LP/NLP B&B with and without submodular cuts on IEEE instances

n	s	$z^* -$ $\mathrm{ldet}(\boldsymbol{C})$	LP/NLP B&B				LP/NLP B&B + submod. cuts			
			LB	UB	MIPgap[1]	time[2]	LB	UB	MIPgap[1]	time[2]
117	5	80.15	80.15	80.15	0.00	11	80.15	80.15	0.00	1
117	10	156.90	156.90	156.90	0.00	423	156.90	156.90	0.00	5
117	15	231.63	230.68	233.86	3.18	-	231.63	231.63	0.00	1019
117	16	246.31	244.65	248.92	4.27	-	246.31	246.31	0.00	5368
117	17	260.94	259.77	264.96	5.19	-	260.94	261.55	0.61	-
117	18	275.56	275.01	279.82	4.81	-	275.30	276.64	1.34	-
117	19	290.15	287.34	295.82	8.47	-	289.53	291.69	2.16	-
117	20	304.69	300.83	311.94	11.11	-	304.07	306.27	2.20	-
299	35	367.49	360.53	368.29	7.76	-	367.49	367.49	0.00	33
299	40	404.02	326.62	410.14	83.52	-	404.02	404.02	0.00	505
299	45	439.81	394.85	442.91	48.06	-	439.81	439.81	0.00	1334
299	50	474.49	425.78	478.45	52.67	-	474.31	475.19	0.88	-
299	51	481.24	444.43	484.84	40.41	-	481.17	482.07	0.90	-
299	52	487.98	364.09	504.14	140.05	-	487.64	490.08	2.44	-
299	53	494.66	323.89	510.02	186.13	-	493.93	497.20	3.27	-
299	54	501.27	345.42	514.02	168.61	-	500.36	503.91	3.55	-
299	55	507.84	360.22	518.89	158.67	-	506.21	510.81	4.60	-
299	56	514.37	381.03	524.84	143.81	-	512.14	517.43	5.29	-
299	57	520.89	400.04	529.63	129.59	-	519.55	524.12	4.57	-

[1] MIPgap = UB − LB

[2] time in seconds; "-": instance not solved within four hours