Mathematical Programming (2019) 176:175–205
https://doi.org/10.1007/s10107-018-1331-z

FULL LENGTH PAPER

Series B

Blessing of massive scale: spatial graphical model estimation with a total cardinality constraint approach

Ethan X. Fang[1] · Han Liu[2] · Mengdi Wang[2]

Received: 10 September 2017 / Accepted: 15 September 2018 / Published online: 1 October 2018

Abstract
We consider the problem of estimating high dimensional spatial graphical models with a total cardinality constraint (i.e., the ℓ_0-constraint). Though this problem is highly nonconvex, we show that its primal-dual gap diminishes linearly with the dimensionality and provide a convex geometry justification of this "blessing of massive scale" phenomenon. Motivated by this result, we propose an efficient algorithm to solve the dual problem (which is concave) and prove that the solution achieves optimal statistical properties. Extensive numerical results are also provided.

Mathematics Subject Classification 65C60 · 65K05 · 78M50

1 Introduction

We consider the problem of estimating high dimensional spatial graphical models. More specifically, let $\boldsymbol{X} = (X_1, \ldots, X_d)^T$ be a d-dimensional random vector on a spatial field (e.g., a lattice). We aim to find an undirected graph $G = (V, E)$ with vertex set $V = \{1, 2, \ldots, d\}$ and edge set $E \subset V \times V$ to encode the conditional independence of $\boldsymbol{X}$, i.e., $(j, k) \in E$ if and only if X_j and X_k are conditionally dependent given the remaining variables. A spatial graphical model also requires the graph G to be

Electronic supplementary material The online version of this article (https://doi.org/10.1007/s10107-018-1331-z) contains supplementary material, which is available to authorized users.

✉ Mengdi Wang
mengdiw@princeton.edu

Ethan X. Fang
xxf13@psu.edu

Han Liu
hanliu@princeton.edu

[1] Department of Statistics, Department of Industrial and Manufacturing Engineering, Pennsylvania State University, University Park, PA 16802, USA

[2] Department of Operations Research and Financial Engineering, Princeton University, Princeton, NJ 08544, USA

conformed with the spatial proximity. In other words, a necessary condition for the existence of edge $(j, k) \in E$ is that vertices j and k are spatially closed (more details are provided later).

1.1 Motivating applications of spatial graphical models

Spatial graphical models find important real-world applications. We provide two concrete motivating examples. The first application is to infer the topology of sensor network on a 2D surface. The wireless sensor network is widely used in various applications including agriculture [10], military [11] and environmental science [9]. See Yick et al. [25] for a survey. In these applications, it is important to understand how the sensors interact with one another. In practice, each sensor can only communicate with other sensors that are geographically close. Also, in applications such as agricultural and environmental studies, all sensors' corresponding locations are known. Thus, the spatial proximity information of these sensors are available a priori. See Fig. 1a for a simple illustration.

Another example is to estimate the short-range brain network. In these networks, the vertices are voxels or ROIs (region of interest) embedded in the 3D space. Given the brain imaging data and the spatial information, we aim to estimate the graphical model under the constraint that each vertex can only connect to the nodes that are physically close [4]. See Fig. 1b for an illustrative example.

There are many other applications of spatial graphical models. For example, in global weather analysis, we are interested in how the weathers at various locations interact with one another, and two different locations can be conditionally dependent only if they are sufficiently close.

1.2 Main contributions

Under many statistical models, such as Gaussian or Ising models, we estimate spatial graphical models by solving the following problem:

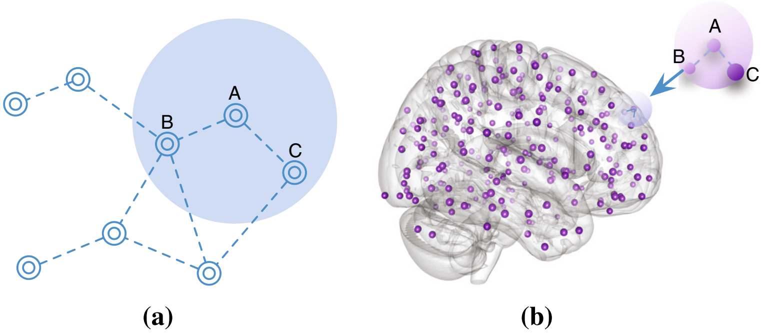

Fig. 1 **a** Sensor network example: in the network, each sensor can only connect to another sensor if they are physically close on the plane. In the figure, each dashed line represents a possible connection. For example, sensor A can only possibly connect to B or C, but not others. **b** Brain network example: for a brain network, we may impose an assumption that each vertex can only connect to another if they are close in the 3-D space. Taking vertex A for example, it can only possibly connect to vertices B or C

$$\min_{\boldsymbol{\beta}_j \in \mathcal{C}_j} \frac{1}{d} \sum_{j=1}^{d} \mathcal{L}_j(\boldsymbol{\beta}_j), \text{ subject to } \sum_{j=1}^{d} \mathcal{R}_j(\boldsymbol{\beta}_j) \le K. \qquad (1.1)$$

For each vertex j, $\mathcal{L}_j : \mathbb{R}^{d_j} \to \mathbb{R}$ is some convex loss function associated with the statistical model. $\mathcal{R}_j(\cdot)$ is some nonconvex function, such as smoothly clipped absolute deviation (SCAD), minimax concave penalty (MCP) or the ℓ_0-(pseudo)norm function. The feasible set $\mathcal{C}_j$ is a closed set. Throughout this paper, for ease of presentation, we implicitly assume $\mathcal{C}_j = \mathbb{R}^{d_j}$. The parameter K is some tuning parameter representing the desired sparsity level. Denote the global minimizer of the problem by $\{\widetilde{\boldsymbol{\beta}}_j\}_{j=1}^d$. The corresponding estimated graph is $\widetilde{G} = (V, \widetilde{E})$, where $(j, k) \in \widetilde{E}$ if and only if $\widetilde{\beta}_{jk} \neq 0$ or $\widetilde{\beta}_{kj} \neq 0$. Given the spatial proximity information, we have that each node j can only connect to a set of vertices $\mathcal{N}_j \subset \{1, \ldots, d\}$, where $|\mathcal{N}_j| = d_j \ll d$. Then, each set $\mathcal{C}_j \subset \mathbb{R}^{d_j}$ is of small dimensions, and this makes each subproblem $\min_{\boldsymbol{\beta}_j \in \mathcal{C}_j} \mathcal{L}_j(\boldsymbol{\beta}_j)$ small dimensional. For example, under the Gaussian model, we let $\mathcal{L}_j(\boldsymbol{\beta}_j) = n^{-1}\|\mathbb{X}_j - \mathbb{X}_{\mathcal{N}_j}\boldsymbol{\beta}_j\|_2^2$, where $\mathbb{X}_j \in \mathbb{R}^n$ is the data corresponding to vertex X_j, and $\mathbb{X}_{\mathcal{N}_j} \in \mathbb{R}^{n \times d_j}$ is the data which corresponds to the potential neighbors of vertex X_j.

Our first contribution is the characterization of the complexity of problem (1.1) in a general setting. We prove that the decision version of problem (1.1) is NP-complete and thus difficult in general. However, for the special and most interesting case where $\mathcal{R}_j(\boldsymbol{\beta}_j) = \|\boldsymbol{\beta}_j\|_0$, we discover that the problem becomes polynomial-time solvable by a dynamic programming algorithm, although this algorithm is practically expensive. We further prove that if the constraint of problem (1.1) is vector-valued, the problem becomes fundamentally more difficult, in which case even finding a fully polynomial-time approximation scheme to solve the problem is NP-hard. For example, we cannot solve the problem if the constraint in (1.1) is changed to $\sum_{j=1}^d (\|\boldsymbol{\beta}_j\|_0, \sum_{k=1}^{d_j-1} \|\beta_{j,k+1} - \beta_{jk}\|_0)^T \le (K_1, K_2)^T$ unless P = NP.

Our second contribution is a scalable algorithm to solve problem (1.1). Although there exists a polynomial-time dynamic programming algorithm to solve the problem when $\mathcal{R}_j(\boldsymbol{\beta}_j) = \|\boldsymbol{\beta}_j\|_0$, the algorithm is not tractable in practice. To achieve a more practical algorithm, we develop a Splitting-Communicating (SPICA) algorithm which solves the Lagrangian dual of the primal problem (1.1). The algorithm utilizes the separable structure of the dual problem and converges to a dual optimal solution geometrically. Since problem (1.1) is nonconvex, there exists a positive duality gap between the primal and dual optimal solutions. Suppose that the number of potential neighbors per-vertex is fixed. We prove that the average-per-vertex duality gap diminishes at the rate of $\mathcal{O}(d^{-1})$ as the graph dimension d increases. As a result, if the dimension d is large, the dual optimal solution is close to the primal optimal solution, and achieves optimal statistical properties. This reveals a "blessing of massive scale" phenomenon.

1.3 Relationship with existing literature

Problem (1.1) is highly nonconvex and raises computational challenges. To overcome such challenges, several existing works rely on solving optimization problems derived from convex relaxations, such as the ℓ_1-relaxation. The motivation of different convex relaxations is to avoid solving nonconvex or combinatorial optimization problems while still achieves fast statistical rates. Extensive works study the theoretical guarantees of different relaxations various models, and achieve optimal minimax lower bounds under certain regularity assumptions. See [12,15–17,27,29]. However, some results prove that there are some unavoidable statistical losses for the estimators derived from these methods. For example, in linear regression, we have that the ℓ_0-constrained estimator $\widehat{\boldsymbol{\beta}}_0$ obtains optimal rate of convergence that $n^{-1}\mathbb{E}\big\{\|\mathbb{X}\widehat{\boldsymbol{\beta}}_0 - \mathbb{X}\boldsymbol{\beta}^*\|_2^2\big\} = \mathcal{O}(n^{-1}s\log d)$, where $s = \|\boldsymbol{\beta}^*\|_0$. In comparison, without restricted eigenvalue-type assumptions, methods based on convex relaxations only achieve a slower rate [28].

To summarize, our work develops a novel framework to estimate high-dimensional spatial graphical models by directly attacking the nonconvex problem under the total cardinality constraint. The proposed algorithm produces a near-optimal solution to the nonconvex optimization problem, and achieves optimal statistical properties. The characterization of the complexity of problem (1.1) provides fundamental understandings and insights of the problem.

Paper Organization. The rest of this paper is organized as follows. Section 2 characterizes the complexity of problem (1.1) under general settings. Section 3 describes the SPICA algorithm In Sect. 4, we provide theoretical guarantees of the "blessing of massiveness" phenomenon and analyze the statistical properties of the estimators. Section 5 provides extensive numerical experiments using both synthetic and sensor network data.

Notations. Let $\mathbb{X} \in \mathbb{R}^{n\times d}$ be the data matrix, and $\mathbb{X}_j$ denotes the jth column of $\mathbb{X}$. Also, $\mathbb{X}_{\mathcal{N}_j}$ denotes the columns of possible neighbors of X_j, where $\mathcal{N}_j$ is the set of possible neighbors of X_j. For a matrix $\mathbf{A} \in \mathbb{R}^{d\times d}$, we denote its maximum eigenvalue by $\Lambda_{\max}(\mathbf{A})$, and its minimum eigenvalue by $\Lambda_{\min}(\mathbf{A})$. We denote by $[d] = \{1, \ldots, d\}$. The norms of $\mathbf{v} = (v_1, \ldots, v_d)^T \in \mathbb{R}^d$ are defined as $\|\mathbf{v}\|_0 = \sum_{j\in[d]} \mathbf{1}\{x_j \neq 0\}$, and $\|\mathbf{v}\|_p = \{\sum_{j\in[d]} v_j^p\}^{1/p}$ for $p \geq 1$.

2 The complexity of spatial-graphical model problem

In this section, we study the complexity of problem (1.1) by relating it to a classical discrete NP-complete problem—the knapsack problem. For general constraints $\mathcal{R}_j$'s, we show that the decision version of problem (1.1) is NP-complete. In the special case of $\mathcal{R}_j(\boldsymbol{\beta}_j) = \|\boldsymbol{\beta}_j\|_0$, we show that the problem admits a polynomial-time algorithm. In the general setting where the nonconvex constraints $\mathcal{R}_j$'s are vector-valued, we prove that the problem does not admit a fully polynomial-time approximation scheme unless P = NP, and the problem is fundamentally more difficult. For example, we cannot solve problems under both the cardinality and a fused-type cardinality constraints, where $\mathcal{R}_j(\boldsymbol{\beta}_j) = (\|\boldsymbol{\beta}_j\|_0, \sum_{k=1}^{d_0-1}\|\beta_{j,k+1} - \beta_{jk}\|_0)^T$.

2.1 Knapsack problem and complexity

The knapsack problem plays an important role in combinatorics. It is motivated from applications in resource allocation, where the goal is to maximize the total utility under capacity constraints. Its simplest form is the following 0–1 knapsack problem:

$$\max_{x_j} \sum_{j=1}^{d} c_j x_j, \text{ subject to } \sum_{j=1}^{d} b_j x_j \le b_0, \ x_j \in \{0,1\}, \ \text{ for } j = 1, \dots, d, \quad (2.1)$$

where c_j's, b_j's and b_0 are positive integers. The input to the 0–1 knapsack problem includes: the constant c_j which is the value of the jth item; the constant b_j which is the cost of the jth item, and the constant b_0 which is the total budget. Let $\mathbf{c} = (c_1, \dots, c_d)^T$ and $\mathbf{b} = (b_1, \dots, b_d)^T \in \mathbb{R}^d$. We refer to problem (2.1) as the 0–1 knapsack problem with input $(\mathbf{c}, \mathbf{b}, b_0)$. This problem is known to be NP-complete [24].

An important variant of the 0–1 knapsack problem is the *multiple-row knapsack problem* (also known as the multiple-dimensional knapsack problem):

$$\max_{x_j} \sum_{j=1}^{d} c_j x_j, \text{ subject to } \sum_{j=1}^{d} b_j^{(\ell)} x_j \le b_0^{(\ell)}, \ \text{ for all } \ell = 1, \dots, L, \ x_j \in \{0,1\}. \quad (2.2)$$

In comparison with the 0–1 knapsack problem, this problem has multiple-row constraints. The multiple-row knapsack problem is fundamentally more difficult than the 0–1 knapsack problem. It is NP-hard to solve the problem to an arbitrary precision. More specifically, it is shown that finding a fully polynomial time approximation scheme for the multiple-row knapsack problem is NP-hard [14], which is defined below.

Definition 2.1 An approximation scheme for a maximization problem (P) is an algorithm that takes two inputs: One is the problem instance P, and the other is a desired numerical accuracy $\epsilon > 0$. Denote by $f^* > 0$ the optimal value of P. The algorithm produces a solution for P with objective value $f(P)$ such that $\left\{f^* - f(P)\right\}/f^* \le \epsilon$. If the running time for the algorithm is bounded by a polynomial function of $1/\epsilon$ and the problem size, it is a fully polynomial time approximation scheme.

To facilitate our discussion, we briefly review some definitions in computational complexity theory in Supplementary Materials Section A. We refer to Williamson and Shmoys [24] for more detailed discussion about the knapsack problem and computational complexity theory.

2.2 NP-completeness of problem (1.1)

In this subsection, we prove that problem (1.1) is NP-complete. To prove that problem (1.1) is NP-complete, we shall construct a two-way polynomial time reduction between problem (1.1) and 0–1 knapsack problem (2.1). We show that given one instance of the 0–1 knapsack problem or problem (1.1), we can construct another instance of the other problem within a polynomial-time, and by solving the other instance we can recover the solution to the original instance. As we discussed in the introduction, the form of loss functions $\mathcal{L}_j$'s depends on the specific statistical model. Without loss of generality, we assume all $\mathcal{L}_j$'s are of a same form (least square or logistic loss for example), and each $\mathcal{L}_j$ only depends on some input data $(\mathbb{X}_j, \mathbb{Y}_j)$. Thus, each $\mathcal{L}_j(\boldsymbol{\beta}_j)$ can be represented as $\mathcal{L}(\boldsymbol{\beta}_j; \mathbb{X}_j, \mathbb{Y}_j)$. We consider finding an ϵ-optimal solution to the problem

$$\min_{\boldsymbol{\beta}_j} \sum_{j=1}^{d} \mathcal{L}(\boldsymbol{\beta}_j; \mathbb{X}_j, \mathbb{Y}_j), \text{ subject to } \sum_{j=1}^{d} \mathcal{R}_j(\boldsymbol{\beta}_j) \le b_0, \tag{2.3}$$

with input $\big(\{\mathbb{X}_j, \mathbb{Y}_j\}_{j\in[d]}, b_0, \epsilon\big)$, where we say a solution is ϵ-optimal if its corresponding objective value is within ϵ of the optimal value, and the solution is feasible. Problem (2.3) can be continuous since both objective and constraint functions in (2.3) can be continuous, and 0–1 knapsack problem is discrete. To connect the two problems, we need to "discretize" problem (1.1). We first consider the loss functions. We impose the following assumption.

- (A.1) Given positive constants c_j's for all $j \in [d]$, we can find $\mathbb{X}_j$, $\mathbb{Y}_j$ and a constant c_0 within a polynomial time such that

$$\mathcal{L}(0; \mathbb{X}_j, \mathbb{Y}_j) = c_0 \quad \text{and} \quad \min_{\beta_j \in \mathbb{R}} \mathcal{L}(\beta_j; \mathbb{X}_j, \mathbb{Y}_j) = -c_j + c_0.$$

This assumption is satisfied for most statistical models. For example, if $\mathcal{L}(\beta_j; \mathbb{X}_j, \mathbb{Y}_j) = \|\mathbb{Y}_j - \mathbb{X}_j\beta_j\|_2^2/n$, it is easy to verify that letting $\mathbb{Y}_j = (\sqrt{c_0}, \sqrt{c_0})^T$ and $\mathbb{X}_j = \big(\sqrt{c_j + c_j'}, \sqrt{c_j - c_j'}\big)^T$, where $c_j' = \sqrt{2c_jc_0 - c_0^2}$, satisfy this assumption.

Next, we look at constraint functions $\mathcal{R}_j$'s. Given a problem instance of (2.3), we need to efficiently construct a knapsack problem of which the constraint is similar to the problem instance (2.3). Since the knapsack problem is discrete, and problem (2.3) is possibly continuous, we assume that we can efficiently "discretize" the constraint functions $\mathcal{R}_j$'s, where we impose the following assumption.

- (A.2) For any j, given any $\delta > 0$ and any set $[-r, r]^{d_0}$ for some $r > 0$, we can find a finite discretization $\mathcal{B}$ of the set that for any point $\boldsymbol{\beta} \in [-r, r]^{d_0}$, there exists a point $\mathbf{p} \in \mathcal{B}$ such that $\|\mathbf{p} - \boldsymbol{\beta}\|_2 \le \delta$ and $\mathcal{R}_j(\mathbf{p}) \le \mathcal{R}_j(\boldsymbol{\beta})$ in a polynomial time.

This assumption holds for most common $\mathcal{R}_j$'s in statistical applications. For example, suppose $\mathcal{R}_j$'s are SCAD functions. We have that the discretization $\{0, \pm\sqrt{\delta/d_0}, \pm 2\sqrt{\delta/d_0}, \ldots, \pm p^*\sqrt{\delta/d_0}\}^{d_0}$ satisfies the assumption, where $p^* = \operatorname{argmax}_{p\in\mathbb{N}}\{p\sqrt{\delta/d_0} \le r\}$, and $\mathbb{N}$ is the set of natural numbers.

Next, we provide the main theorem of this section. We show that given one instance of 0–1 knapsack problem (2.1), we can construct an instance of problem (2.3) within a polynomial-time, and by solving the instance of problem (2.3) we can recover the solution to the original instance of 0–1 knapsack problem, and vice versa. This proves that problem (2.3) is NP-complete since 0–1 knapsack problem (2.1) is known to be NP-complete.

Theorem 2.2 *Under assumptions (A.1)–(A.2), the decision version of the nonconvex constrained optimization problem* (2.3) *is NP-complete.*

Proof See Supplementary Materials Section B for the detailed proof. □

2.3 Polynomial-time algorithm in the case of ℓ_0-constrained problem

Though problem (2.3) is NP-complete, we show that the special case of problem (2.3) under a total cardinality constraint, i.e., the case where $\mathcal{R}_j(\boldsymbol{\beta}_j) = \|\boldsymbol{\beta}_j\|_0$, admits a polynomial-time algorithm. Specifically, given an instance of problem (2.3), we map it to an instance of multiple-choice knapsack problem, and by solving the instance of multiple-choice knapsack problem efficiently, we recover the solution to the instance of problem (2.3).

Let us first introduce multiple-choice knapsack problem. Denote by $\mathbf{C} = (\mathbf{c}_1, \ldots, \mathbf{c}_d)^T \in \mathbb{R}^{d \times d_0}$ and $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_d)^T \in \mathbb{R}^{d \times d_0}$, where $\mathbf{c}_j = (c_{j1}, \ldots, c_{jd_0})^T$ and $\mathbf{b}_j = (b_{j1}, \ldots, b_{jd_0})^T$. Consider the multiple-choice knapsack problem with input $(\mathbf{C}, \mathbf{B}, b_0)$, where all b_{jk}'s and b_0 are positive integers:

$$\max_{x_{jk}} \sum_{j=1}^{d} \sum_{k=1}^{d_0} c_{jk} x_{jk}, \text{ subject to } \sum_{j=1}^{d} \sum_{k=1}^{d_0} b_{jk} x_{jk} \le b_0, \sum_{k=1}^{d_0} x_{jk} \le 1, \; x_{jk} \in \{0, 1\}, \tag{2.4}$$

for all $j \in [d]$ and all $k \in [d_0]$. Given an instance of problem (2.3) under the ℓ_0-constraint, we map the instance to an instance of multiple-choice knapsack problem (2.4). For each j, we solve the subproblems

$$\widehat{\boldsymbol{\beta}}_j(k) = \operatorname*{argmin}_{\boldsymbol{\beta}_j \in \mathcal{C}_j} \mathcal{L}(\boldsymbol{\beta}_j; \mathbb{X}_j, \mathbb{Y}_j), \text{ subject to } \|\boldsymbol{\beta}_j\|_0 \le k, \text{ for } k = 0, 1, \ldots, d_0.$$

Since we assume that d_0 is a constant, the cost of computing all $\widehat{\boldsymbol{\beta}}_j(k)$'s increases linearly as d increases. Let $b_{jk} = k$, $b_0 = K$ and $c_{jk} = -\mathcal{L}\big(\widehat{\boldsymbol{\beta}}_j(k); \mathbb{X}_j, \mathbb{Y}_j\big) + c_0$, where $c_0 > \max_{j,k} \mathcal{L}\big(\widehat{\boldsymbol{\beta}}_j(k); \mathbb{X}_j, \mathbb{Y}_j\big)$ for $j \in [d]$ and $k \in [d_0]$. We obtain a multiple-choice knapsack problem of the form (2.4). Denote by $\{x^*_{jk}\}$ an optimal solution to the multiple-choice knapsack problem. We have that $\{x^*_{jk}\}$ recovers an optimal solution to the ℓ_0-constrained problem by setting $\widehat{\boldsymbol{\beta}}_j = \widehat{\boldsymbol{\beta}}_j(k)$ if $x^*_{jk} = 1$.

Next, we present a dynamic programming approach to solve the multiple-choice knapsack problem, which is a variant of Pisinger [19]. We formulate a dynamic program with the state variable (d', k'), where $1 \le d' \le d$ and $0 \le k' \le K$. The dimension of the state space is $d(K+1)$. We define the value function of a state (d', k') to be the optimal value for the multiple-choice knapsack problem considering only multiple-choice sets 1 to d' with constraint k'. In another words, let

$$V(d', k') = \max_{x_{jk}} \sum_{j=1}^{d'} \sum_{k=0}^{d_0} c_{jk} x_{jk}, \text{ subject to } \sum_{j=1}^{d'} \sum_{k=0}^{d_0} k x_{jk} \le k',$$
$$\sum_{k=0}^{d_0} x_{jk} \le 1, \; x_{jk} \in \{0, 1\}.$$

Thus, $V(d, K)$ is the optimal value for the original multiple-choice knapsack problem. To facilitate our discussion, fixing c_{jk}'s, we denote the knapsack problem with first d' multiple choice set and constraint variable k' by $(MK_{d',k'})$, i.e., we let

$$(MK_{d',k'}):\quad \max_{x_{jk}} \sum_{j=1}^{d'}\sum_{k=0}^{d_0} c_{jk}x_{jk},\ \text{subject to}\ \sum_{j=1}^{d'}\sum_{k=0}^{d_0} kx_{jk} \le k',$$

$$\sum_{k=0}^{d_0} x_{jk} \le 1,\ x_{jk} \in \{0, 1\}.$$

Denote by $\{x_{jk}^*\}$ the optimal solution to the problem $(MK_{d,K})$. Let $k' = \sum_{j=1}^{d'}\sum_{k=0}^{d_0} kx_{jk}^*$. To efficiently solve the problem, a key observation is that the partial solution $\{x_{jk}^*\}$ for $j = 1, \dots, d'$ and $k = 0, \dots, d_0$ is the optimal solution solution for the problem $(MK_{d',k'})$. This can be proved by contradiction that if the assertion does not hold, we can replace the partial solution with the optimal solution to $(MK_{d',k'})$, and we keep the rest of the original optimal solution to $(MK_{d,K})$ the same. The sum of the corresponding objectives of the two partial solutions is greater than the original optimal objective. This leads to a contradiction.

Based on this observation, we find the optimal value $V(d, K)$ by a recursive algorithm based on following recursive equations:

$$V(1, k') = \max\left\{V(1, k'-1), \max\{c_{1k} : k \le k'\}\right\},$$

and for $d' > 1$

$$V(d', k') = \max_k \left\{V(d', k'-1), \max\{V(d'-1, k'-k) + c_{d'k} : k \le k'\right\}. \quad (2.5)$$

The dynamic programming algorithm for solving the problem $(MK_{d,K})$ is summarized in Algorithm 1. The total number of states is $d(K + 1)$ for the problem, and the computational complexity for computing each $V(d', k')$ is $\mathcal{O}(d_0 + 1)$. Thus, the complexity of computing $V(d, K)$ is of the order $\mathcal{O}(dd_0K)$. In our problem, the number K is upper-bounded by dd_0, so the computational complexity is of the order $\mathcal{O}(d^2d_0^2)$. Note that this does not include the computation for the coefficients c_{jk}'s. To compute all c_{jk}'s, for each sub-problem $\mathcal{L}_j$, we need to enumerate all 2^{d_0} possible combinations of the support of $\boldsymbol{\beta}_j \in \mathbb{R}^{d_0}$. Thus, applying dynamic programming techniques, the total computational complexity for solving the ℓ_0-constraint problem is of order $\mathcal{O}(2^{d_0}d + d^2d_0^2)$, which is still a polynomial order of the dimension d.

In summary, Algorithm 1 is a dynamic programming algorithm that runs in a polynomial-time. However, it can be very expensive in practice as it requires enumerating and solving all subproblems. In the next section, we will propose a more practical algorithm.

Meanwhile, we point out that the dynamic programming approach becomes significantly more expensive when $\mathcal{R}_j$'s are some continuous functions instead of the ℓ_0-norm. When $\mathcal{R}_j$ is continuous, our reduction to the multiple-choice knapsack problem requires a fine discretization of $\mathcal{R}_j$. This may result in a large number of choices

Algorithm 1 Dynamic Programming Algorithm for Problem (2.4)

```
**Input:** $c_{jk} \in \mathbb{R}_+$, $K$
**Output:** $x^*_{jk}$
$V(d', -1) \leftarrow 0$, $V(0, k') \leftarrow 0$, $\mathcal{S}(d', k') = 0$ for all $d'$ and $k'$. $d' \leftarrow 0$.
Let $\mathcal{S}(1, k') \leftarrow \mathrm{argmax}_k \{c_{1k} : k \leq k'\}$.
**while** $d' < d$ **do**
    Let $d' \leftarrow d' + 1$. Solve (2.5) for $1 \leq k' \leq K$.
    **for** $k' = 0 : K$ **do**
        **if** $\max_k \left\{V(d'-1, k'-k) + c_{d'k} : k \leq k', V(d'-1, k'-k) > 0\right\} > V(d', k'-1)$ **then**
            Let $\mathcal{S}(d', k') \leftarrow \mathrm{argmax}_k \left\{V(d'-1, k'-k) + c_{d'k} : k \leq k', V(d'-1, k'-k) > 0\right\}$.
        **end if**
    **end for**
**end while**
$k' \leftarrow K$.
**while** $d' > 0$ **do**
    Let $x^*_{d', \mathcal{S}(d', k')} = 1$, $k' \leftarrow k' - \mathcal{S}(d', k')$, $d' \leftarrow d' - 1$.
**end while**
```

in the constructed knapsack problem, making the dynamic programming approach inefficient. In comparison, when $\mathcal{R}_j$ is the ℓ_0-constraint, the values of $\mathcal{R}_j$ are naturally discrete. The resulting knapsack problem has at most d_0 choices, which is a relatively small number. In general, the dynamic programming approach to problem (2.3) is practically slow, even though it is a polynomial-time algorithm.

2.4 A "Harder" result in the case of vector-valued constraint

In this subsection, we consider the case where the functions $\mathcal{R}_j(\boldsymbol{\beta}_j)$'s are vector-valued. Specifically, we consider the problem:

$$\min_{\beta_j} \sum_{j=1}^{d} \mathcal{L}(\boldsymbol{\beta}_j; \mathbb{X}_j, \mathbb{Y}_j), \text{ subject to } \sum_{j=1}^{d} \mathcal{R}_j^{(\ell)}(\boldsymbol{\beta}_j) \leq b_0^{(\ell)}, \tag{2.6}$$

for $\ell = 1, \ldots, L$, where $L \geq 1$. This problem contains (2.3) as a special case. Intuitively speaking, finding an ϵ-optimal solution to the problem (2.6) should not be more difficult than problem (2.3). However, the next theorem proves that the multiple-row constraints case (2.6) is fundamentally more difficult.

Theorem 2.3 *Under assumptions (A.1)–(A.2), if $L > 1$, finding a fully polynomial-time approximation scheme for problem* (2.6) *is NP-hard.*

Proof The proof is based on constructing a two-way polynomial-time reduction between the multiple-row knapsack problem (2.2) and the problem (2.6). The argument is analogous to the proof of Theorem 2.2, and we omit it to avoid repetition. Then, as shown in Magazine and Chern [14], there does not exist a fully polynomial-time approximation scheme to solve the two-row multiple-choice knapsack problem unless we assume P = NP. □

This theorem establishes one of the strongest forms of complexity, and shows the problem (2.6) is fundamentally hard to solve. In comparison, when there exists only one total cardinality constraint, we can solve the problem within a polynomial-time by dynamic programming.

3 SPICA algorithm for spatial-graph estimation

In this section, we describe an efficient duality-based algorithm to directly attack the nonconvex problem (1.1) and prove that it achieves a near-optimal solution, even though problem (1.1) belongs to an NP-complete class of problems as shown in Sect. 2. We illustrate the geometric intuition on why this algorithm generates a near-optimal solution when the dimension d is large. We focus our discussion on the case where the problem (1.1) is subject to the total cardinality constraint

$$\sum_{j=1}^{d} \mathcal{R}_j(\boldsymbol{\beta}_j) = \sum_{j=1}^{d} \|\boldsymbol{\beta}_j\|_0 \le K.$$

Note that all the analyses of this section can be generalized to general nonconvex constraints. We focus our discussion on the case of ℓ_0-constraint, in which the solution achieves optimal statistical properties. For ease of presentation, in what follows, we assume that all $\boldsymbol{\beta}_j$'s are of identical dimensions d_0, i.e., $\boldsymbol{\beta}_j \in \mathbb{R}^{d_0}$ for all $j \in [d]$, where d_0 is a given constant.

3.1 SPICA algorithm

In this subsection, we propose an algorithm to solve problem (1.1) subject to total cardinality constraint. It is practically efficient and can handle problems with large dimension d. We consider the Lagrangian dual of (1.1),

$$\max_{\lambda \ge 0} \sum_{j=1}^{d} \mathcal{Q}_j(\lambda) - \lambda K, \ \text{ where } \mathcal{Q}_j(\lambda) = \inf_{\boldsymbol{\beta}_j \in \mathcal{C}_j} \{\mathcal{L}_j(\boldsymbol{\beta}_j) + \lambda \|\boldsymbol{\beta}_j\|_0\}, \tag{3.1}$$

for all $j = 1, \ldots, d$. The variable λ is the Lagrangian multiplier. According to literatures on duality theory [1], even though the primal problem (1.1) is nonconvex, letting $\mathcal{Q}(\lambda) = \sum_{j=1}^{d} \mathcal{Q}_j(\lambda)$, the dual $\widehat{\mathcal{Q}}(\lambda) = \mathcal{Q}(\lambda) - \lambda K$ is a concave function of λ. We aim to obtain the dual optimal solution-multiplier pair $(\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}, \widehat{\lambda})$ defined as

$$\begin{gathered} \widehat{\lambda} = \operatorname{argmax}_{\lambda \ge 0} \sum_{j=1}^{d} \mathcal{Q}_j(\lambda) - \lambda K, \ \text{ where } \mathcal{Q}_j(\lambda) = \mathcal{L}_j\big\{\widehat{\boldsymbol{\beta}}_j(\lambda)\big\} + \lambda \big\|\widehat{\boldsymbol{\beta}}_j(\lambda)\big\|_0, \\ \text{and } \widehat{\boldsymbol{\beta}}_j(\lambda) = \operatorname*{argmin}_{\boldsymbol{\beta}_j \in \mathcal{C}_j} \ \mathcal{L}_j(\boldsymbol{\beta}_j) + \lambda \|\boldsymbol{\beta}_j\|_0, \ \widehat{\boldsymbol{\beta}}_j = \operatorname*{argmin}_{\boldsymbol{\beta}_j \in \mathcal{C}_j} \ \mathcal{L}_j(\boldsymbol{\beta}_j) + \widehat{\lambda} \|\boldsymbol{\beta}_j\|_0 \end{gathered} \tag{3.2}$$

In what follows, with a slight abuse of notation, we refer $\widehat{\boldsymbol{\beta}}_j$ defined above as the dual optimal solution for ease of presentation.

We adopt the "golden section search" method to solve the dual problem (3.1), which runs iteratively. Let $\xi = (-1+\sqrt{5})/2$. Given two initial points λ_1 and λ_2, let $\lambda_3 = \lambda_2 + \xi(\lambda_1 - \lambda_2)$ and $\lambda_4 = \lambda_1 + \xi(\lambda_2 - \lambda_1)$. During each iteration, if $\widehat{\mathcal{Q}}(\lambda_3) > \widehat{\mathcal{Q}}(\lambda_4)$, then we move the points $\{\lambda_2, \widehat{\mathcal{Q}}(\lambda_2)\}$ to $\{\lambda_4, \widehat{\mathcal{Q}}(\lambda_4)\}$, and $\{\lambda_4, \widehat{\mathcal{Q}}(\lambda_4)\}$ to $\{\lambda_3, \widehat{\mathcal{Q}}(\lambda_3)\}$, and we update λ_3 to $\lambda_2 + \xi(\lambda_1 - \lambda_2)$. Otherwise, let $\{\lambda_1, \widehat{\mathcal{Q}}(\lambda_1)\}$ be $\{\lambda_3, \widehat{\mathcal{Q}}(\lambda_3)\}$, and $\{\lambda_3, \widehat{\mathcal{Q}}(\lambda_3)\}$ be $\{\lambda_4, \widehat{\mathcal{Q}}(\lambda_4)\}$, and we update λ_4 to $\lambda_1 + \xi(\lambda_2 - \lambda_1)$. Specifically, at each iteration, we first compute the values of $\{\widehat{\mathcal{Q}}_j(\lambda_i)\}_{i=1}^4$ for all j. This can be conducted efficiently since the dual problem (3.1) "splits" the Lagrangian minimization problem into d nonconvex problems with small dimension d_0, and we can compute $\mathcal{Q}_j(\lambda_i)$'s in parallel. We call this a "splitting" step. Next, we centrally update λ_i's according to the golden section search method, which is a "communicating" step. Thus we call it a "splitting-communicating" (SPICA) algorithm, which is summarized in Algorithm 2. This algorithm finds a narrow interval that contains the optimal multiplier of the problem (3.1) after some iterations, and the output solution is the midpoint of the interval. It is well known that the golden section search method converges ξ-geometrically to the dual optimal solution $(\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}, \widehat{\lambda})$ [1], i.e., we have

$$\widehat{\lambda}^{(t)} - \widehat{\lambda} \leq \xi^t |\lambda_2^{(0)} - \lambda_1^{(0)}|,$$

where $\lambda_i^{(0)}$'s denote the initial points, $\widehat{\lambda}^{(t)} = |\lambda_1^{(t)} - \lambda_2^{(t)}|/2$, and $(\lambda_1^{(t)}, \lambda_2^{(t)})$ denotes the corresponding point after t iterations.

Algorithm 2 SPICA Algorithm

```
Input: $\lambda_1, \lambda_2 \in \mathbb{R}_+, \epsilon > 0, \xi = (-1+\sqrt{5})/2$
Output: $\widehat{\lambda}, \{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}$
$\lambda_3 \leftarrow \lambda_2 + \xi(\lambda_1 - \lambda_2), \lambda_4 \leftarrow \lambda_1 + \xi(\lambda_2 - \lambda_1)$.
while $|\lambda_1 - \lambda_2| > \epsilon$ do
    if $\mathcal{Q}(\lambda_3) - \lambda_3 K > \mathcal{Q}(\lambda_4) - \lambda_4 K$ then
        $\{\lambda_2, \mathcal{Q}(\lambda_2) - \lambda_2 K\} \leftarrow \{\lambda_4, \mathcal{Q}(\lambda_4) - \lambda_4 K\}, \{\lambda_4, \mathcal{Q}(\lambda_4) - \lambda_4 K\} \leftarrow \{\lambda_3, \mathcal{Q}(\lambda_3) - \lambda_3 K\}$.
        $\lambda_3 \leftarrow \lambda_2 + \xi(\lambda_1 - \lambda_2)$.
    else
        $\{\lambda_1, \mathcal{Q}(\lambda_1) - \lambda_1 K\} \leftarrow \{\lambda_3, \mathcal{Q}(\lambda_3) - \lambda_3 K\}, \{\lambda_3, \mathcal{Q}(\lambda_3) - \lambda_3 K\} \leftarrow \{\lambda_4, \mathcal{Q}(\lambda_4) - \lambda_4 K\}$.
        $\lambda_4 \leftarrow \lambda_1 + \xi(\lambda_2 - \lambda_1)$.
    end if
end while
$\widehat{\lambda} \leftarrow (\lambda_1 + \lambda_2)/2, \widehat{\boldsymbol{\beta}}_j = \operatorname*{argmin}_{\boldsymbol{\beta}_j} \mathcal{L}_j(\boldsymbol{\beta}_j) + \widehat{\lambda}\|\boldsymbol{\beta}_j\|_0$ for all $j \in [d]$.
```

The SPICA algorithm provides significant computational advantages. However, instead of a global optimal solution to problem (1.1), it generates an optimal solution to dual problem (3.1). Since the total cardinality constraint is nonconvex, there might exists some duality gap between the dual and the primal optimal solutions. In the next section, we illustrate a convexification phenomenon that, as d increases, the duality gap diminishes and does not impair any statistical loss in a wide range of problems.

3.2 The convexification phenomenon

Before rigorously proving that the dual optimal solution obtained by the SPICA algorithm is close to the primal optimal solution of problem (1.1), we illustrate some geometric intuition. The intuition traces back to some early convex geometry work, namely, the Shapley–Folkman Lemma [22]. Consider the averaged Minkowski sum of d sets $\mathcal{A}_1,\ldots,\mathcal{A}_d$ defined as $\{d^{-1}\sum_{j=1}^{d} a_j : a_j \in \mathcal{A}_j \text{ for } j \in [d]\}$. The lemma reveals a geometric fact that the average of many nonconvex sets tends to be convex. In particular, letting $\rho(\mathcal{A})$ be a metric of the nonconvexity of the set $\mathcal{A}$, we have

$$\rho\Big(\frac{\mathcal{A}_1 + \mathcal{A}_2 + \cdots + \mathcal{A}_d}{d}\Big) \to 0, \text{ as } d \to \infty.$$

We provide an example to illustrate this convexification effect. Let the maximum distance between two sets $\mathcal{A}$ and $\mathcal{B}$ be $d(\mathcal{A}, \mathcal{B}) = \sup_{b\in\mathcal{B}} \inf_{a\in\mathcal{A}}\{\|a - b\| : a \in \mathcal{A}, b \in \mathcal{B}\}$. We measure the nonconvexity of a set $\mathcal{A}$ by the maximum distance between $\mathcal{A}$ and its convex hull. Since this distance is 0 if and only if $\mathcal{A}$ is convex, the maximum distance is a reasonable measure of how convex a set is. Considering the discrete set $\mathcal{A} = \{0, 1\}$ and its convex hull $\overline{\mathcal{A}} = [0, 1]$, we have $\rho(\mathcal{A}) = d(\mathcal{A}, \overline{\mathcal{A}}) = 1/2$. The maximum distance between the average of the Minkowski sum of two $\mathcal{A}$'s, which is $\mathcal{A}_2 = \{0, 1/2, 1\}$, and its convex hull is $\rho(\mathcal{A}_2) = d(\mathcal{A}_2, \overline{\mathcal{A}}) = 1/4$. Let the average of d $\mathcal{A}$'s be $\mathcal{A}_d$. We have $\rho(\mathcal{A}_d) = 1/2d$, which converges to 0 as d increases. We thus conclude that the average of d $\mathcal{A}$'s tend to be more convex as d increases. In Fig. 2, we provide an geometric illustration of such increase of convexity, and we provide the mathematical description of Shapley–Folkman Lemma in Supplementary Materials Section C.

Let us return to problem (1.1). The duality gap between the primal problem (1.1) and its dual can be bounded by the nonconvexity of the set $d^{-1}\sum_{j=1}^{d}\mathcal{A}_j$, where each $\mathcal{A}_j = \big\{\big(\|\boldsymbol{\beta}_j\|_0, \mathcal{L}_j(\boldsymbol{\beta}_j)\big) : \boldsymbol{\beta}_j \in \mathcal{C}_j\big\}$ characterizes the joint nonconvexity of $\big(\|\boldsymbol{\beta}_j\|_0, \mathcal{L}_j(\boldsymbol{\beta}_j)\big)$. By the intuition above, as d increases, the set $d^{-1}\sum_{j=1}^{d}\mathcal{A}_j$ tends to be convex, and we expect a diminishing duality gap. This convexification phenomenon provides a hint that solving the dual problem might be as good as solving the primal problem.

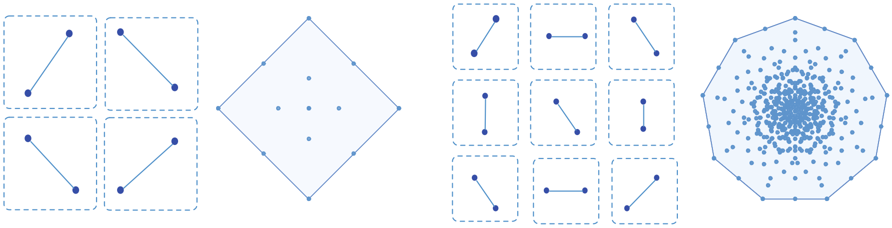

Fig. 2 Left two: the shaded area of the second figure is the convex hull of the averaged Minkowski sum of four sets illustrated on the first figure. Each of the four sets contains two points, and the line between them represents the convex hull. Right two: The shaded area on the right is the convex hull of the averaged Minkowski sum of nine sets. The maximum distance between the averaged Minkowski sum and its convex hull decreases as the number of sets increases

4 Theoretical justification of SPICA algorithm

In this section, we provide theoretical justications for the SPICA algorithm. We prove that the average-per-vertex duality gap diminishes as d increases. We analyze the statistical properties of the estimators computed by the SPICA algorithm. We also discuss the computational complexities of the dynamic programming approach and the SPICA algorithm in Supplementary Materials Section E.

4.1 Diminishing duality gap

In this subsection, we prove that the average-per-vertex duality gap diminishes at a rate of $\mathcal{O}(1/d)$. This result provides the theoretical justification that the estimator obtained by the SPICA algorithm (Algorithm 2) is near-optimal, i.e., it is close to the primal optimal solution $\{\widetilde{\boldsymbol{\beta}}_j\}_{j\in[d]}$ of problem (1.1) defined as

$$\{\widetilde{\boldsymbol{\beta}}_j\}_{j=1}^d = \operatorname*{argmin}_{\boldsymbol{\beta}_j\in\mathcal{C}_j} \frac{1}{d}\sum_{j=1}^d \mathcal{L}_j(\boldsymbol{\beta}_j), \text{ subject to } \sum_{j=1}^d \|\boldsymbol{\beta}_j\|_0 \le K,$$

As we discussed in the previous subsections, we consider the Lagrangian dual problem (3.1), and the SPICA algorithm (Algorithm 2) finds the dual optimal solution-multiplier pair $\big(\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}, \widehat{\lambda}\big)$ as defined in (3.2). Since the problem is nonconvex, strong duality does not hold. In this case, we only have weak duality that

$$\frac{1}{d}\sum_{j=1}^d \mathcal{L}_j(\widehat{\boldsymbol{\beta}}_j) + \widehat{\lambda}\Big(\sum_{j=1}^d \|\widehat{\boldsymbol{\beta}}_j\|_0 - K\Big) \le \frac{1}{d}\sum_{j=1}^d \mathcal{L}_j(\widetilde{\boldsymbol{\beta}}_j), \tag{4.1}$$

where both $\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}$ and $\{\widetilde{\boldsymbol{\beta}}_j\}_{j\in[d]}$ satisfy the total cardinality constraint, but some duality gap might exist in this case. Note that the duality gap is the difference between primal and dual optimal objective values. We provide an example to illustrate that the primal and dual optimal solutions do not necessarily match, which results in a positive duality gap. Let

$$\mathbb{X} = \begin{pmatrix} 1 & 6 & 5 & 5 \\ 8 & 9 & 3 & 2 \\ 7 & 10 & 8 & 8 \end{pmatrix} \quad \text{and} \quad \mathbf{y} = \begin{pmatrix} 12 \\ 20 \\ 25 \end{pmatrix}.$$

Considering the ℓ_0-constrained problem,

$$\min_{\boldsymbol{\beta}} \|\mathbb{X}\boldsymbol{\beta} - \mathbf{y}\|_2^2, \text{ subject to } \|\boldsymbol{\beta}\|_0 \le 2,$$

the primal solution is $\widetilde{\boldsymbol{\beta}} = (857/497, 0, 292/165, 0)^T$. For the dual solution,

$$\widehat{\boldsymbol{\beta}}(\lambda) = \|\mathbb{X}\boldsymbol{\beta} - \mathbf{y}\|_2^2 + \lambda\|\boldsymbol{\beta}\|_0.$$

it is not difficult to check that when $\lambda \ge 1169 - 1669/217$, $\widehat{\boldsymbol{\beta}}(\lambda) = \mathbf{0}$, if $\lambda \in [1669/434, 1669/217)$, $\widehat{\boldsymbol{\beta}}(\lambda) = (0, 502/217, 0, 0, 0)^T$, if $\lambda < 1669/434$, $\widehat{\boldsymbol{\beta}}(\lambda) =$

$(1, 1, 1, 0)^T$. This implies that the primal and dual optimal solutions do not match, and there exists a strictly positive duality gap equals 449/894.

The next theorem proves that, as d increases, the average-per-vertex duality gap vanishes at the rate of $\mathcal{O}(1/d)$. This gives a strong evidence that the dual solution obtained by the SPICA algorithm is a fairly good approximation to the primal solution, especially when d is large. Usually, such a large d would cause the "curse of dimensionality" in nonconvex optimization, but our result reveals a "blessing of massive scale" phenomenon.

Theorem 4.1 *The solution $\big(\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}, \widehat{\lambda}\big)$ obtained by the SPICA algorithm (Algorithm 2) is a dual optimal solution-multiplier pair, which solves the dual problem (3.1), and satisfies*

$$\frac{1}{d}\sum_{j=1}^{d}\mathcal{L}_j(\widehat{\boldsymbol{\beta}}_j) \le \frac{1}{d}\sum_{j=1}^{d}\mathcal{L}_j(\widetilde{\boldsymbol{\beta}}_j) + \frac{C_g}{d}, \quad \textit{and} \quad \sum_{j=1}^{d}\|\boldsymbol{\beta}_j\|_0 \le K, \tag{4.2}$$

where $\{\widetilde{\boldsymbol{\beta}}\}_{j\in[d]}$ is the primal optimal solution, and the constant C_g is

$$C_g = \max_{j\in[d]}\Big|\mathcal{L}_j(\mathbf{0}) - \min_{\boldsymbol{\beta}_j\in\mathcal{C}_j}\mathcal{L}_j(\boldsymbol{\beta}_j)\Big|. \tag{4.3}$$

Proof First, we prove the existence of the optimal dual solution. This is proved in Lemma D.1 in Supplementary Materials. Next, if $\widehat{\lambda} = 0$, we have $\widehat{\boldsymbol{\beta}}_j = \operatorname{argmin}_{\boldsymbol{\beta}_j}\mathcal{L}_j(\boldsymbol{\beta}_j)$ for all j's as defined in (3.2). Since $\sum_{j\in[d]}\|\widehat{\boldsymbol{\beta}}_j\|_0 \le K$ by the feasibility of $\widehat{\boldsymbol{\beta}}_j$'s, we have $\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}$ is also the primal optimal solution. This implies $\widehat{\boldsymbol{\beta}}_j = \widetilde{\boldsymbol{\beta}}_j$ for all j, and our claim follows as desired.

If $\widehat{\lambda} > 0$, we prove in Lemma D.3 in Supplementary Materials that one of the two cases must hold:

(i) There exists a dual optimal solution-multiplier pair $(\widehat{\lambda}, \{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]})$, such that $\sum_{j\in[d]}\|\widehat{\boldsymbol{\beta}}_j\|_0 = K$.

(ii) Case (i) does not hold, and there exist at least two solutions achieve dual optimal objective, denoted as $\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}$ and $\{\widehat{\boldsymbol{\beta}}'_j\}_{j\in[d]}$, such that $\sum_{j\in[d]}\|\widehat{\boldsymbol{\beta}}_j\|_0 < K$ and $\sum_{j\in[d]}\|\widehat{\boldsymbol{\beta}}'_j\|_0 > K$.

Next, we consider the two cases separately. For case (i), there exists a dual optimal solution $\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}$ satisfying $\sum_{j\in[d]}\|\widehat{\boldsymbol{\beta}}_j\|_0 = K$. By the weak duality (4.1), we have

$$\frac{1}{d}\sum_{j=1}^{d}\mathcal{L}_j(\widehat{\boldsymbol{\beta}}_j) \le \frac{1}{d}\sum_{j=1}^{d}\mathcal{L}_j(\widetilde{\boldsymbol{\beta}}_j) + \widehat{\lambda}\Big(\sum_{j=1}^{d}\|\widehat{\boldsymbol{\beta}}_j\|_0 - K\Big) = \frac{1}{d}\sum_{j=1}^{d}\mathcal{L}_j(\widetilde{\boldsymbol{\beta}}_j),$$

where the first inequality holds by the definition of dual optimal solution that $\widehat{\boldsymbol{\beta}}_j = \operatorname{argmin}_{\boldsymbol{\beta}_j}\mathcal{L}_j(\boldsymbol{\beta}_j) + \widehat{\lambda}\|\boldsymbol{\beta}_j\|_0$, and the assertion of the theorem follows as desired. We also point out that since $\{\widetilde{\boldsymbol{\beta}}_j\}_{j\in[d]}$ is the primal optimal solution, the above inequality and the feasibility of $\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}$ guarantee the primal optimality of $\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}$, i.e., the dual optimal solution $\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}$ is also a primal optimal solution. This also leads to the certificate of primal optimality result stated in Corollary 4.2.

In the remaining proof, we focus our discussion on case (ii). This case is more complicated and requires more careful analysis due to the existence of multiple solutions. Recall that, given the multiplier $\widehat{\lambda}$, a dual solution is obtained by solving d subproblems of the ℓ_0-penalized form:

$$\min_{\boldsymbol{\beta}_j} \mathcal{L}_j(\boldsymbol{\beta}_j) + \widehat{\lambda}\|\boldsymbol{\beta}_j\|_0, \text{ for all } j = 1, \dots, d.$$

Since there are multiple solutions which achieve dual optimal objective, as shown in Lemma D.3 in Supplementary Materials, we have that there is at least one j, such that the above ℓ_0-penalized optimization problem has multiple solutions, i.e., for some j, there exist $\widehat{\boldsymbol{\beta}}_j^{(1)}$ and $\widehat{\boldsymbol{\beta}}_j^{(2)}$ such that

$$\mathcal{L}_j(\widehat{\boldsymbol{\beta}}_j^{(1)}) + \widehat{\lambda}\|\widehat{\boldsymbol{\beta}}_j^{(1)}\|_0 = \mathcal{L}_j(\widehat{\boldsymbol{\beta}}_j^{(2)}) + \widehat{\lambda}\|\widehat{\boldsymbol{\beta}}_j^{(2)}\|_0. \tag{4.4}$$

In addition, any combination of the optimal solutions of the subproblems provides a dual optimal objective without satisfying the feasibility. In what follows, we show that we can select a dual optimal solution from all possible combinations, such that the selected solution achieves the error bound (4.2).

Suppose that there exist m solutions achieve dual optimal objective. Let $\{\widehat{\boldsymbol{\beta}}_j^{(1)}\}_{j\in[d]}$, $\{\widehat{\boldsymbol{\beta}}_j^{(2)}\}_{j\in[d]}$,…, $\{\widehat{\boldsymbol{\beta}}_j^{(m)}\}_{j\in[d]}$ be the sequence of solutions ranked by their corresponding primal objective values, i.e.,

$$\sum_{j=1}^{d} \mathcal{L}_j\big(\widehat{\boldsymbol{\beta}}_j^{(1)}\big) \leq \sum_{j=1}^{d} \mathcal{L}_j\big(\widehat{\boldsymbol{\beta}}_j^{(2)}\big) \leq, \dots, \leq \sum_{j=1}^{d} \mathcal{L}_j\big(\widehat{\boldsymbol{\beta}}_j^{(m)}\big). \tag{4.5}$$

Meanwhile, by the dual optimality, we have,

$$\begin{aligned}\sum_{j=1}^{d}\Big[\mathcal{L}_j\{\widehat{\boldsymbol{\beta}}_j^{(1)}\} + \widehat{\lambda}\|\widehat{\boldsymbol{\beta}}_j^{(1)}\|_0\Big] &= \sum_{j=1}^{d}\Big[\mathcal{L}_j\{\widehat{\boldsymbol{\beta}}_j^{(2)}\} + \widehat{\lambda}\|\widehat{\boldsymbol{\beta}}_j^{(2)}\|_0\Big]\\ &= \cdots = \sum_{j=1}^{d}\Big[\mathcal{L}_j\{\widehat{\boldsymbol{\beta}}_j^{(m)}\} + \widehat{\lambda}\|\widehat{\boldsymbol{\beta}}_j^{(m)}\|_0\Big].\end{aligned}$$

Since $\widehat{\lambda} > 0$ by assumption, we have

$$\sum_{j=1}^{d} \|\widehat{\boldsymbol{\beta}}_j^{(1)}\|_0 \geq \sum_{j=1}^{d} \|\widehat{\boldsymbol{\beta}}_j^{(2)}\|_0 \geq, \dots, \geq \sum_{j=1}^{d} \|\widehat{\boldsymbol{\beta}}_j^{(m)}\|_0.$$

Consequently, by the assumption that case (ii) holds, we have

$$\sum_{j=1}^{d} \|\widehat{\boldsymbol{\beta}}_j^{(1)}\|_0 > K > \sum_{j=1}^{d} \|\widehat{\boldsymbol{\beta}}_j^{(m)}\|_0.$$

To prove our claim, a key observation is that, for any $m_1 \in \{1, \ldots, m-1\}$, $\sum_{j\in[d]} \mathcal{L}_j\big\{\widehat{\boldsymbol{\beta}}_j^{(m_1+1)}\big\} - \sum_{j\in[d]} \mathcal{L}_j\big\{\widehat{\boldsymbol{\beta}}_j^{(m_1)}\big\} \le C_g$, where C_g is defined in (4.3). This is proved in Lemma D.5 in Supplementary Materials.

Thus, by the assumption that case (ii) holds, there exist two consecutive solutions $\{\widehat{\boldsymbol{\beta}}_j^{(m_1)}\}_{j\in[d]}$ and $\{\widehat{\boldsymbol{\beta}}_j^{(m_1+1)}\}_{j\in[d]}$ for some $m_1 \in [m]$, such that

$$\Big| \sum_{j=1}^{d} \mathcal{L}_j\big\{\widehat{\boldsymbol{\beta}}_j^{(m_1)}\big\} - \sum_{j=1}^{d} \mathcal{L}_j\big\{\widehat{\boldsymbol{\beta}}_j^{(m_1+1)}\big\} \Big| \le C_g,$$
$$\text{and } \sum_{j=1}^{d} \|\widehat{\boldsymbol{\beta}}_j^{(m_1)}\|_0 > K > \sum_{j=1}^{d} \|\widehat{\boldsymbol{\beta}}_j^{(m_1+1)}\|_0.$$

In addition, by the dual optimality of the two solutions, it holds that

$$\begin{aligned}&\sum_{j=1}^{d} \mathcal{L}_j\big\{\widehat{\boldsymbol{\beta}}_j^{(m_1)}\big\} + \widehat{\lambda}\Big\{ \sum_{j=1}^{d} \|\widehat{\boldsymbol{\beta}}_j^{(m_1)}\|_0 - K \Big\} \\ &\quad = \sum_{j=1}^{d} \mathcal{L}_j\big\{\widehat{\boldsymbol{\beta}}_j^{(m_1+1)}\big\} + \widehat{\lambda}\Big(\sum_{j=1}^{d} \|\widehat{\boldsymbol{\beta}}_j^{(m_1+1)}\|_0 - K \Big).\end{aligned}$$

Consequently, as $\sum_{j\in[d]} \|\widehat{\boldsymbol{\beta}}_j^{(m_1)}\|_0 > K > \sum_{j\in[d]} \|\widehat{\boldsymbol{\beta}}_j^{(m_1+1)}\|_0$, and $\widehat{\lambda} > 0$, we further obtain that

$$\begin{aligned}0 &\le -\widehat{\lambda}\Big(\sum_{j=1}^{d} \|\widehat{\boldsymbol{\beta}}_j^{(m_1+1)}\|_0 - K \Big) \le \widehat{\lambda}\Big(\sum_{j=1}^{d} \|\widehat{\boldsymbol{\beta}}_j^{(m_1)}\|_0 - \sum_{j=1}^{d} \|\widehat{\boldsymbol{\beta}}_j^{(m_1+1)}\|_0 \Big) \\ &= \sum_{j=1}^{d} \mathcal{L}_j\big\{\widehat{\boldsymbol{\beta}}_j^{(m_1+1)}\big\} - \sum_{j=1}^{d} \mathcal{L}_j\big\{\widehat{\boldsymbol{\beta}}_j^{(m_1)}\big\} \le C_g.\end{aligned} \tag{4.6}$$

We have

$$\sum_{j=1}^{d} \mathcal{L}_j\big\{\widehat{\boldsymbol{\beta}}_j^{(m_1+1)}\big\} \le \sum_{j=1}^{d} \mathcal{L}_j\big(\widetilde{\boldsymbol{\beta}}_j\big) - \widehat{\lambda}\Big(\sum_{j=1}^{d} \|\widehat{\boldsymbol{\beta}}_j^{(m_1+1)}\|_0 - K \Big) \le \sum_{j=1}^{d} \mathcal{L}_j\big(\widetilde{\boldsymbol{\beta}}_j\big) + C_g,$$

where the first inequality holds by the weak duality (4.1), and the second inequality holds by (4.6).

To conclude, in both cases (i) and (ii), we prove that there exists a dual optimal solution $\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}$ that achieves the total cardinality constraint, and approximates the primal solution within a constant error bound even if d increases. □

To interpret the constant C_g, each $\left|\mathcal{L}_j(\mathbf{0}) - \min_{\boldsymbol{\beta}_j\in\mathcal{C}_j} \mathcal{L}_j(\boldsymbol{\beta}_j)\right|$ is some "divergence" related to vertex j. It essentially measures the information gain by using neighboring vertices to explain uncertainties of vertex j. The constant C_g is the maximal divergence among all vertices.

This result indicates that when the maximal divergence C_g is bounded, the average-per-vertex duality gap decreases to 0 as d increases. By the proof of Theorem 4.1 for case (i), i.e., when $\sum_{j\in[d]} \|\boldsymbol{\beta}_j\|_0 = K$, the next corollary follows immediately. This provides a criterion to determine if the primal optimality holds for $\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}$.

Corollary 4.2 (Certificate for Primal Optimality) *Let $\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}$ be the dual optimal solution for problem* (1.1) *obtained by the SPICA algorithm (Algorithm* 2*). When the equality $\sum_{j=1}^{d} \|\widehat{\boldsymbol{\beta}}_j\|_0 = K$ holds, it holds that the dual optimal solution also achieves the primal optimality, i.e.,*

$$\sum_{j=1}^{d} \mathcal{L}_j\big(\widehat{\boldsymbol{\beta}}_j\big) = \sum_{j=1}^{d} \mathcal{L}_j\big(\widetilde{\boldsymbol{\beta}}_j\big), \ \textit{if} \sum_{j=1}^{d} \|\widehat{\boldsymbol{\beta}}_j\|_0 = K.$$

In Sect. 5, we find that empirically, the dual solution $\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}$ satisfies the certificate $\sum_{j=1}^{d} \|\widehat{\boldsymbol{\beta}}_j\|_0 = K$ with high probability.

4.2 Statistical properties

In this subsection, we provide theoretical justifications of the estimators derived from the SPICA algorithm. We provide the statistical guarantee that under weak assumptions, the duality gap does not sacrifice any statistical efficiency when d is large. This matches Theorem 4.1. We discuss the rates of convergence for the estimators provided by the SPICA algorithm under Gaussian and Ising graphical models. Most technical proofs are provided in Supplementary Materials Section F.

4.2.1 Gaussian graphical model

We first apply the SPICA algorithm to estimate Gaussian graphical model. Consider a d-dimensional random vector $\boldsymbol{X} = (X_1, X_2, \ldots, X_d)^T \sim N(\mathbf{0}, \boldsymbol{\Sigma})$. Under the Gaussian assumption, the conditional independence between X_j and X_k holds if and only if $\boldsymbol{\Theta}_{jk} = 0$, where $\boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1}$. Extensive literatures study this problem from different approaches [3,5,12,13,15,20,21,26]. More recently, Fan et al. [7] proposes the innovated scalable efficient estimation (ISEE) method motivated by the innovated transformation [8]. This approach transforms the problem into a covariance matrix estimation problem, and is shown to be statistically efficient and practically scalable.

Under the spatial graphical modeling setting, taking a neighborhood pursuit approach, we formulate the graph estimation problem as

$$\min_{\{\boldsymbol{\beta}_j\}_{j\in[d]}} \frac{1}{dn}\sum_{j=1}^{d}\|\mathbb{X}_j - \mathbb{X}_{\mathcal{N}_j}\boldsymbol{\beta}_j\|_2^2, \text{ subject to } \sum_{j=1}^{d}\|\boldsymbol{\beta}_j\|_0 \le K,$$

where $\mathbb{X} \in \mathbb{R}^{n\times d}$ is the data matrix; $\mathbb{X}_{\mathcal{N}_j} \in \mathbb{R}^{n\times d_j}$ denotes the columns of $\mathbb{X}$ which correspond to the potential neighbors of X_j, and K is a pre-specified total cardinality. Given a solution $\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}$, we obtain the connected neighbors of each X_j by taking the corresponding nonzero components of $\widehat{\boldsymbol{\beta}}_j$. Consequently, we construct the graph estimator by either "OR" or "AND" rule on combining the neighborhoods for all X_j's. This approach is based on the fact that

$$\begin{aligned} &X_j = \boldsymbol{X}_{\mathcal{N}_j}^T\boldsymbol{\beta}_j^* + \epsilon_j, \text{ where } \boldsymbol{\beta}_j^* = \big(\boldsymbol{\Sigma}_{\mathcal{N}_j,\mathcal{N}_j}\big)^{-1}\boldsymbol{\Sigma}_{\mathcal{N}_j,j} \in \mathbb{R}^{d-1}, \\ &\epsilon_j \sim N\big(0,\sigma_j^2\big), \text{ and } \sigma_j^2 = \boldsymbol{\Sigma}_{jj} - \boldsymbol{\Sigma}_{j,\mathcal{N}_j}\big(\boldsymbol{\Sigma}_{\mathcal{N}_j,\mathcal{N}_j}\big)^{-1}\boldsymbol{\Sigma}_{\mathcal{N}_j,j}, \end{aligned} \tag{4.7}$$

and by the block matrix inversion formula, it holds that

$$\boldsymbol{\Theta}_{jj} = \{\mathrm{Var}(\epsilon_j)\}^{-1} = \sigma_j^{-2}, \text{ and } \boldsymbol{\Theta}_{\mathcal{N}_j,j} = -\{\mathrm{Var}(\epsilon_j)\}^{-1}\boldsymbol{\beta}_j^* = -\sigma_j^{-2}\boldsymbol{\beta}_j^*.$$

Thus, $\boldsymbol{\Theta}_{jk} = 0$ if and only if the corresponding component of $\boldsymbol{\beta}_j^*$ is 0.

We point out that there are several advantages of the total cardinality approach over the ℓ_1 or other penalized approaches: (i) Imposing the total cardinality constraint directly handles the estimator's sparsity level. This provides a more intuitive approach than penalized methods, where tuning parameters do not give very interpretable meanings. (ii) Total cardinality constraint approach does not incur any estimation bias. In comparison, penalized approach induces some estimation biases. Although such biases are asymptotically negligible under appropriate scaling, the finite-sample behavior of the penalized approach is indeed outperformed by the total cardinality approach as demonstrated in simulation studies in Sect. 5.

Next, we analyze the statistical properties of the estimator obtained by the SPICA algorithm. As the neighborhood pursuit approach formulates the problem as a regression problem, we first bound the "prediction risk" of the estimator. This leads to the estimator's fast rate of convergence.

In the following discussion, for ease of presentation, we assume that the numbers of potential neighbors of the vertices are the same, i.e., $|\mathcal{N}_1| =, \ldots, = |\mathcal{N}_d| = d_0$. The next therem guarantees that the average-per-vertex risk of our estimator converges at the minimax optimal rate, and justifies the vanishing gap does not incur statistical loss if the dimension d is large.

Theorem 4.3 *Suppose that we have n independent samples of* $\boldsymbol{X} \sim N(\mathbf{0}, \boldsymbol{\Sigma}) \in \mathbb{R}^d$, *and the spatial information that each vertex j can only connect to a set of vertices* $\mathcal{N}_j \subset \{1, \ldots, d\}$ *and* $|\mathcal{N}_j| = d_0$. *Let* $\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}$ *be the estimator obtained by the SPICA algorithm. Assume* $s \le K$, *and* $2K \le dd_0$, *where* $\sum_{j=1}^{d}\|\boldsymbol{\beta}_j^*\|_0 = s$, *and* $\boldsymbol{\beta}_j^*$*'s are defined in* (4.7). *We further assume* $\mathrm{diag}(\boldsymbol{\Sigma}) \le \sigma^2$. *Then, with probability at least* $1 - \mathcal{O}(d^{-1})$, *we have*

$$\frac{1}{dn}\sum_{j=1}^{d}\|\mathbb{X}_{\mathcal{N}_j}\widehat{\boldsymbol{\beta}}_j-\mathbb{X}_{\mathcal{N}_j}\boldsymbol{\beta}_j^*\|_2^2\le C_1\cdot\frac{K\log d}{dn}+C_2\cdot\frac{\log d}{d}, \tag{4.8}$$

where C_1 and C_2 are two constants, and do not depend on K, d and n.

Proof The proof is based on the following two lemmas. The first lemma quantifies the risk of the estimator, which involves the duality gap. The second lemma quantifies the duality gap C_g incurred by the SPICA algorithm. The two lemmas are proved in Supplementary Materials Section F.1.

Lemma 4.4 *Suppose we have n independent samples of $\boldsymbol{X}\sim N(\mathbf{0},\boldsymbol{\Sigma})\in\mathbb{R}^d$, and the prior information that each X_j can only connect to a set of nodes $\mathcal{N}_j\subset\{1,\dots,d\}$, i.e., $\boldsymbol{\Theta}_{jk}=0$ if $k\notin\mathcal{N}_j$. Let $\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}$ be the estimator obtained by the SPICA algorithm. Assume $2K\le dd_0$, $s\le K$, $\sum_{j\in[d]}\|\boldsymbol{\beta}_j^*\|_0=s$, where $\boldsymbol{\beta}_j^*$'s are defined in (4.7). We further assume $\operatorname{diag}(\boldsymbol{\Sigma})\le\sigma^2$. Then, for any $\delta>0$, with probability at least $1-\delta$, we have*

$$\begin{aligned}&\frac{1}{n}\sum_{j=1}^{d}\|\mathbb{X}_{\mathcal{N}_j}\widehat{\boldsymbol{\beta}}_j-\mathbb{X}_{\mathcal{N}_j}\boldsymbol{\beta}_j^*\|_2^2\\&\le 64\frac{\sigma^2}{n}\log\left\{\sum_{j=1}^{2K}\binom{dd_0}{j}\right\}+\frac{128\sigma^2K}{n}\log 6+\frac{64\sigma^2}{n}\log(\sigma^{-1})+2C_g,\end{aligned} \tag{4.9}$$

where the constant C_g is the duality gap incurred by the SPICA algorithm.

Lemma 4.5 *Suppose we have n independent samples of $\boldsymbol{X}\sim N(\mathbf{0},\boldsymbol{\Sigma})\in\mathbb{R}^d$. Let $\mathcal{L}_j(\boldsymbol{\beta}_j)=\|\mathbb{X}_j-\mathbb{X}_{\mathcal{N}_j}\boldsymbol{\beta}_j\|_2^2$ be the least square loss. We have,*

$$\max_j\left\{\mathcal{L}_j(\mathbf{0})-\min_{\boldsymbol{\beta}_j}\mathcal{L}_j(\boldsymbol{\beta}_j)\right\}\le n\sigma^2+C\cdot n\log d,$$

with probability at least $1-\mathcal{O}(d^{-1})$, where C is a constant.

Combining the above two lemmas, and plugging (G.1) in Supplementary Materials Section G into (4.9), our claim follows as desired. □

This theorem proves that if $n<d$ and if the average-per-vertex degree is larger than 1, the estimator $\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}$ obtains the optimal rate of convergence. Note that this result does not require any restricted-eigenvalue type assumptions on $\mathbb{X}$. In comparison, it is shown in Zhang et al. [28] that if we do not impose such assumptions, other estimators based on convex relaxations, such as the Lasso estimator, cannot achieve the optimal rate unless P = NP. In addition, if we impose the sparse eigenvalue condition that the minimum eigenvalue of the sub-covariance matrices $\boldsymbol{\Sigma}_{\mathcal{N}_j,\mathcal{N}_j}$'s are all bounded below, i.e., there exists a constant $\rho>0$, such that

$$\Lambda_{\min}\big(\boldsymbol{\Sigma}_{\mathcal{N}_j,\mathcal{N}_j}\big)>\rho,\ \text{for all } j=1,\dots,d.$$

We have that the estimator $\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}$ obtains the fast rate of convergence that

$$\frac{1}{d}\sum_{j=1}^{d}\|\widehat{\boldsymbol{\beta}}_j-\boldsymbol{\beta}_j^*\|_2^2\le \underbrace{C_1\cdot\frac{K\log d}{dn}}_{\text{statistical error}}+\underbrace{C_2\cdot\frac{\log d}{d}}_{\text{duality gap}},$$

where C_1 and C_2 are two constants, and do not depend on K, d and n. Note that if the certificate of primal optimality (Corollary 4.2) holds, the duality gap term disappears.

In graphical model estimation, support recovery is of significant importance. The next corollary provides the support recovery guarantee of the estimator.

Corollary 4.6 *Assume that all the assumptions in Theorem 4.3 and the sparse eigenvalue condition hold and* $K = s$, *where* $\sum_{j\in[d]}\|\boldsymbol{\beta}_j^*\|_0 = s$. *Suppose that we have the minimal signal strength that for all* j,

$$\|\boldsymbol{\beta}_j^*(\mathcal{S}_j)\|_{\min} > C\cdot\sqrt{\frac{\log d}{n}}, \tag{4.10}$$

where $\mathcal{S}_j$ *denotes the support of* $\boldsymbol{\beta}_j^*$; $\|\mathbf{v}\|_{\min} = \min_j |v_j|$, *and* C *is a constant which does not depend on* K, d *and* n. *We have, with probability at least* $1-\mathcal{O}(d^{-1})$,

$$\left|supp\big(\{\widehat{\boldsymbol{\beta}}_j\}_{j\in[d]}\big)\cap supp\big(\{\boldsymbol{\beta}_j^*\}_{j\in[d]}\big)\right|\ge s-d_0,$$

where $supp(\mathbf{v})$ *denotes the support of the vector* $\mathbf{v}$.

Proof By the proof of Theorem 4.1, the estimator obtained by the SPICA algorithm is an optimal solution under the constraint $\sum_{j\in[d]}\|\boldsymbol{\beta}_j\|_0 = s - s'$ for some $s'\in\{0,\ldots,d_0\}$. Thus, the corollary follows by analyzing the property of such minimizer. See Supplementary Materials Section F.3 for the proof. □

This corollary proves that the SPICA algorithm almost exactly recovers the support of the graph with high probability. As d and s increase, if d_0 is fixed, the ratio between the number of correctly estimated support over the number of true support converges to 1 with high probability. Also, similar to the estimation results, if the certificate of primal optimality (Corollary 4.2) holds, the estimator exactly recovers the true support with high probability.

In comparison with recent work Fan et al. [7] from the theoretical perspective, the ISEE procedure also achieves optimal statistical rate of convergence. However, the assumption (Condition 2 in Fan et al. [7]) on the ℓ_∞-norm cone invertibility factor is slightly different from our imposed sparse eigenvalue condition. The two methods thus achieve optimality under different conditions.

We also point out that by similar proof techniques in Fan and Lv [6], we can show that if SCAD or MCP is adopted, then asymptotically, the method is equivalent to the cardinality constrained method. However, when the signal is low, the cardinality constrained method introduces less estimation bias if the tuning parameter in SCAD/MCP is not properly selected.

Another advantage of the proposed ℓ_0-constrained method is that the tuning parameter, which is the cardinality, gives a direct interpretable meaning for practitioners, which is like the best K-subset selection.

4.2.2 Ising graphical model

In this subsection, we consider the spatial Ising graphical model. Ising graphical model studies the conditional independences among random variables $X_j \in \{\pm 1\}$ for $j \in [d]$. Under Ising graphical model, the joint distribution of $\boldsymbol{X} = (X_1, \dots, X_d)^T$ is

$$\mathbb{P}(X_1 = x_1, \dots, X_d = x_d) = \frac{1}{Z(\boldsymbol{\beta})} \exp\Big(\sum_{j \neq k} \frac{\beta_{jk} x_j x_k}{4}\Big),$$

where $Z(\boldsymbol{\beta})$ is some unknown partition function; each β_{jk} describes the interaction between vertex j and vertex k, and $\beta_{jk} = \beta_{kj}$.

Since the function $Z(\boldsymbol{\beta})$ is not given, directly estimating β_{jk}'s is not tractable. For the ith observation $\mathbf{x}_i = (x_{i1}, \dots, x_{id})^T \in \{\pm 1\}^d$, let $\theta_{ij} = \mathbb{P}(X_j = x_{ij} | \boldsymbol{X}_{\setminus j} = \mathbf{x}_{i, \setminus j})$ be the conditional distribution of the jth vertex given others. Adopting the composite likelihood idea, we have

$$\theta_{ij} = \frac{\exp\big(\sum_{k:k\neq j} \beta_{jk} x_{ij} x_{ik}\big)}{\exp\big(\sum_{k:k\neq j} \beta_{jk} x_{ij} x_{ik}\big) + 1}.$$

We have that the negative conditional log-likelihood of the jth vertex is

$$\mathcal{L}_j(\boldsymbol{\beta}_j) = -\frac{1}{n} \sum_{i=1}^{n} \log(\theta_{ij}),$$

Incorporating the spatial information, we have the prior information that $\boldsymbol{\beta}_{jk} = 0$ if $(j, k) \notin \mathcal{N}_j$ for each j, where $|\mathcal{N}_j| = d_0$. Adopting the total cardinality approach, we estimate $\boldsymbol{\beta}_j$'s by solving the following problem

$$\min_{\boldsymbol{\beta}_j} \frac{1}{d} \sum_{j=1}^{d} \mathcal{L}_j(\boldsymbol{\beta}_j), \text{ subject to } \sum_{j=1}^{d} \|\boldsymbol{\beta}_j\|_0 \leq K.$$

Next, we analyze the statistical properties of the estimators $\{\widehat{\boldsymbol{\beta}}_j\}_{j=1}^d$ obtained by the SPICA algorithm. We impose the following mild assumptions:

Assumption 4.7 Under Ising model with parameters $\{\boldsymbol{\beta}_j^*\}_{j\in[d]}$, assume:

- (B.1) $\|\boldsymbol{\beta}_j^*\|_\infty \leq R$ for some $R \in (0, \infty)$.
- (B.2) The population Hessian matrix with respect to any subset $\mathcal{K} \subset \{1, \dots, dd_0\}$, satisfies the local sparse eigenvalue condition that $\Lambda_{\min}\big\{\mathbb{E}\big[\nabla^2_{\mathcal{K}\mathcal{K}} \mathcal{L}(\boldsymbol{\beta}^*)\big]\big\} > 2\rho$, where $|\mathcal{K}| = 2K$, $\mathcal{L}(\boldsymbol{\beta}^*) = \sum_{j=1}^d \mathcal{L}_j(\boldsymbol{\beta}_j^*)$, $\boldsymbol{\beta}^* = (\boldsymbol{\beta}_1^{*T}, \dots, \boldsymbol{\beta}_d^{*T})^T$, and $\rho > 0$ is a constant.

Note that assumption (B.1) is used in most literatures. For assumption (B.2), we only assume such a sparse eigenvalue condition at the point $\boldsymbol{\beta}^*$. This is essential for the identifiability of $\boldsymbol{\beta}^*$. The next theorem provides the fast rate of convergence of the estimator obtained by the SPICA algorithm.

Theorem 4.8 *Suppose that Assumption 4.7 holds, and assume that the n independent samples are generated from a Ising model with parameters* $\boldsymbol{\beta}_j^* \in \mathbb{R}^{d_0}$ *for all* $j \in [d]$ *and* $\sum_{j\in[d]} \|\boldsymbol{\beta}_j^*\|_0 = s \leq K$. *We have, with probability at least* $1 - \mathcal{O}(d^{-1})$,

$$\frac{1}{d}\sum_{j=1}^{d} \|\widehat{\boldsymbol{\beta}}_j - \boldsymbol{\beta}_j^*\|_2^2 \leq \underbrace{C_1 \cdot \frac{K \log d}{dn}}_{\textit{statistical error}} + \underbrace{C_2 \cdot \frac{1}{d}}_{\textit{duality gap}},$$

where C_1 *and* C_2 *are two constants, and do not depend on* K, d *and* n.

Proof See Supplementary Materials Section F.4 for the proof. □

5 Numerical results

In this section, we conduct extensive numerical experiments to test the SPICA algorithm in comparison with ℓ_1-penalized method. We compare the parameter estimation and graph recovery performances of these two methods using both synthetic and real datasets. For ease of presentation, we provide the numerical studies under the Gaussian graphical model.

5.1 Synthetic data

We first use synthetic data. We consider three different sets of parameters: (i) $n = 100$, $d = 1,000$; (ii) $n = 100$, $d = 2,000$; (iii) $n = 100$, $d = 5,000$, and we let the number of potential neighbors $d_0 = 10$. We further consider three different models for generating undirected graphs and precision matrices as discussed below. Figure 3 illustrates sample graphs under these models. We repeat each setting for 100 times and report the averaged performance.

$*$ **Scale-free graph**. We generate the graph by the preferential attachment mechanism. We begin with a graph with a chain of 2 vertices. At iteration j, we add a new vertex to the graph. The new vertex j connects to one of the previous d_0 vertices, with a

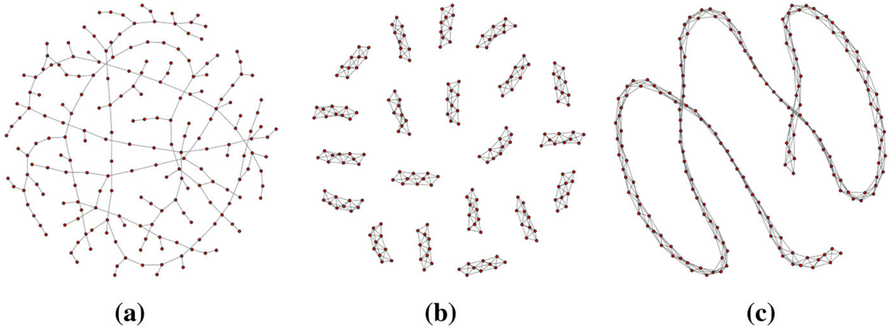

Fig. 3 Examples of the three graph patterns we consider in the simulation study. **a** Scale-Free. **b** Block. **c** Band

probability which is proportional to the number of degrees of the existing vertex. Mathematically, let p_i be the probability that the new vertex j will connect to the existing vertex i is $p_i = k_i / \sum_{i'=\min\{1,j-d_0/2\}}^{j-1} k_{i'}$, where k_i is the current degree of the vertex i. Thus, the resulting graph has $d-1$ edges. Given the graph, we generate the corresponding adjacency matrix $\mathbf{A}$ by setting the diagonal elements to be 0, and we set the nonzero off-diagonal elements to be $\rho = 0.3$, 0.5 or 0.7. Then, we construct the precision matrix $\boldsymbol{\Theta}$ as

$$\boldsymbol{\Theta} = \mathbf{D}\big[\mathbf{A} + \big\{\big|\Lambda_{\min}(\mathbf{A})\big| + 0.2\big\} \cdot \mathbf{I}_d\big]\mathbf{D}, \tag{5.1}$$

where $\Lambda_{\min}(\mathbf{A})$ denotes the smallest eigenvalue of $\mathbf{A}$; $\mathbf{I}_d$ denotes the identity matrix, and $\mathbf{D}$ is a diagonal matrix with $D_{jj} = 1$ for $j = 1, \ldots, d/2$ and $D_{jj} = 3$ for $j = d/2+1, \ldots, d$. Finally, we generate the multivariate Gaussian samples: $\mathbf{x}_1, \ldots, \mathbf{x}_n \sim N_d(\mathbf{0}, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma} = \boldsymbol{\Theta}^{-1}$.

$*$ **Block graph**. We construct an adjacency matrix $\mathbf{A}$ as a block diagonal matrix. Each block is of the size 10, and within each block, two nodes j and k are connected if they are within a same block and $|j-k| \leq 3$. We set the nonzero off-diagonal entries of $\mathbf{A}$ as $\rho = 0.3$, 0.5 or 0.7, and diagonal entries to be 1. This matrix is positive definite. The graph has $4.8d$ edges, and we let the precision matrix be $\boldsymbol{\Theta} = \mathbf{DAD}$, where $\mathbf{D}$ is generated by the same procedure as in the Scale-Free graph.

$*$ **Band graph**. Given d vertices indexed by $j = 1, \ldots, d$, we generate edges between the vertices whose corresponding coordinates are at distance less than or equal to 3, i.e., a node j is connected to node k if and only if $|j-k| \leq 3$. The resulting graph has $3d-6$ edges. Given the graph, we construct the corresponding precision matrix $\boldsymbol{\Theta}$ by the same procedure as in generating the scale-free graph.

We first consider the graph recovery performances of the SPICA algorithm and the ℓ_1-penalized method, where the spatial information is also used. In particular, we evaluate the graph recovery performance by looking at the false positive and false negative rates. In particular, let $\widehat{G}^K = (V, \widehat{E}^K)$ be an estimated graph under the total cardinality constraint with tuning parameter K. The number of false positive discoveries using tuning parameter K is $\mathrm{FP}(K) = |\widehat{E}^K \backslash E|$, where $A\backslash B = \{a : a \in A \text{ and } a \notin B\}$, and the number of false negative discoveries with K is $\mathrm{FN}(K) = |E\backslash \widehat{E}^K|$. Consequently, we define the corresponding false positive rate (FPR) and the false negative rate (FNR) as

$$\mathrm{FPR}(K) = \frac{\mathrm{FP}(K)}{\binom{d}{2} - |E|} \quad \text{and} \quad \mathrm{FNR}(K) = \frac{\mathrm{FN}(K)}{|E|}.$$

We plot the receiver operating characteristic (ROC) curves using the averaged $\big\{\mathrm{FPR}(K), \mathrm{TPR}(K)\big\}$ generated by the SPICA algorithm for 100 times. Also, we plot the averaged ROC curves for the ℓ_1-penalized method for comparisons. Figure 4 corresponds to the results from different settings of the scale-free graph. Figures 5 and 6 present the results from different settings of block and band graphs.

By Figs. 4, 5 and 6, we see that the SPICA algorithm performs better than ℓ_1-penalized method under the block and band models, and the two methods perform

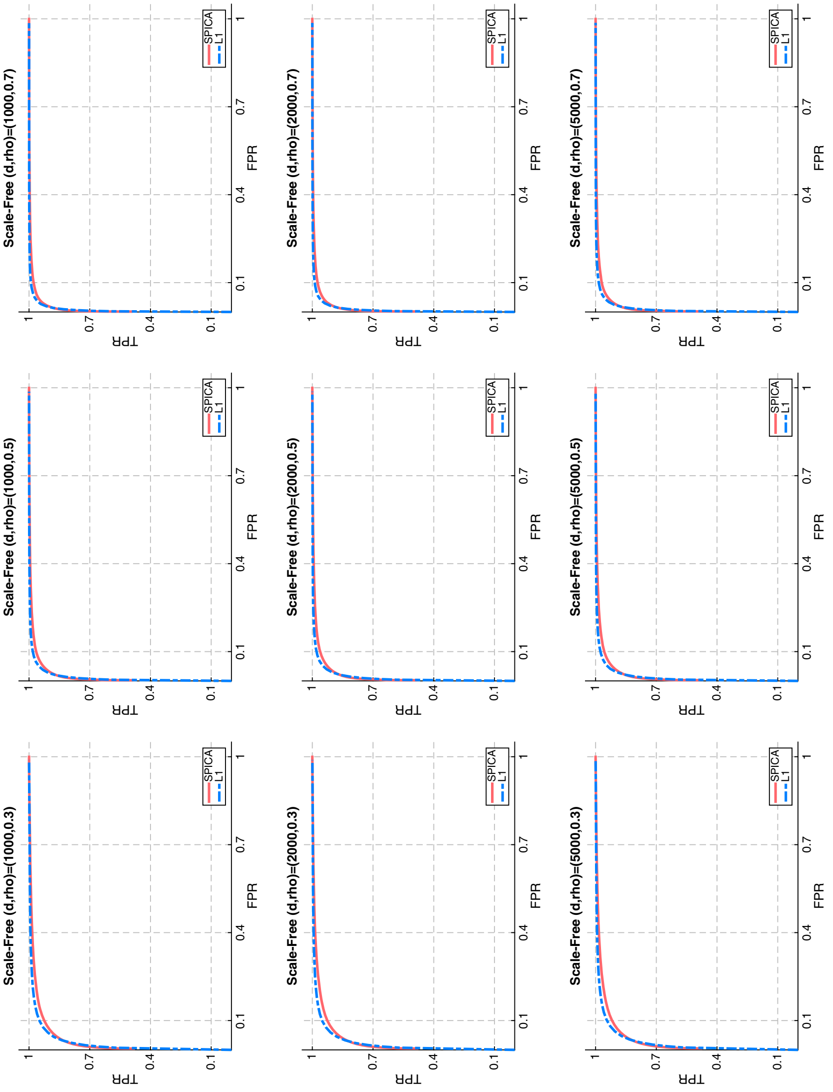

Fig. 4 ROC curves for scale-free model under different settings

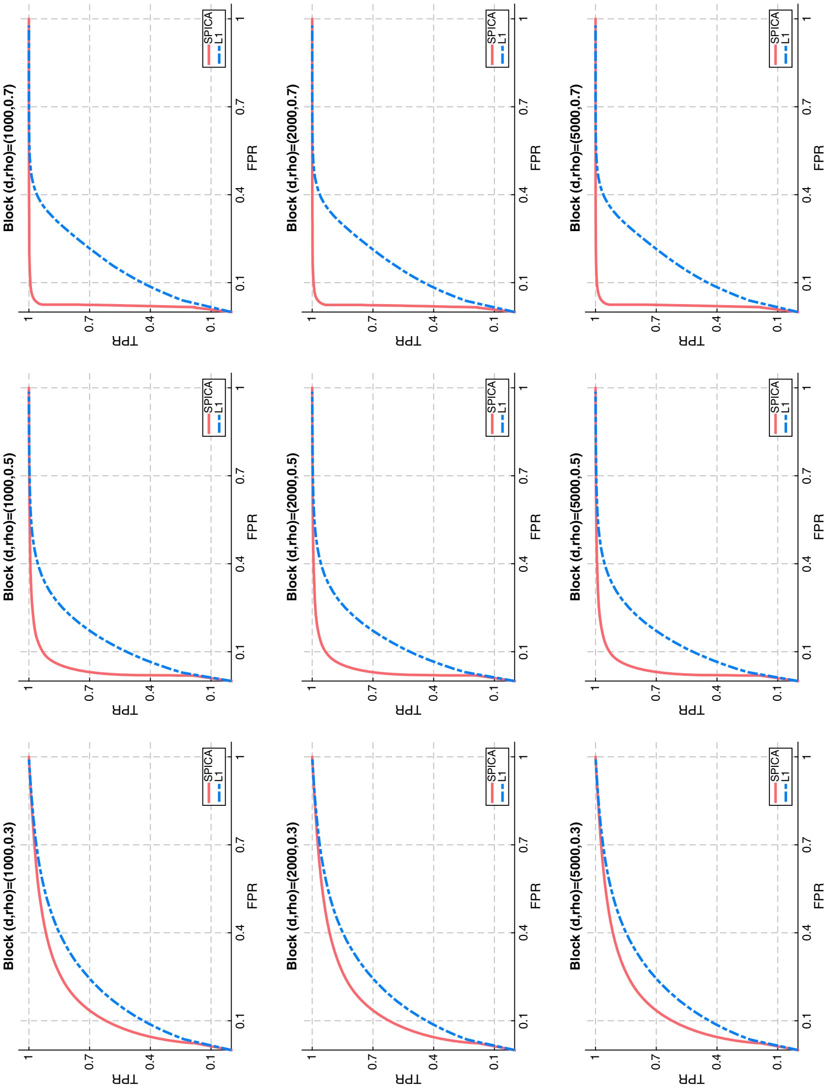

Fig. 5 ROC curves for block model under different settings

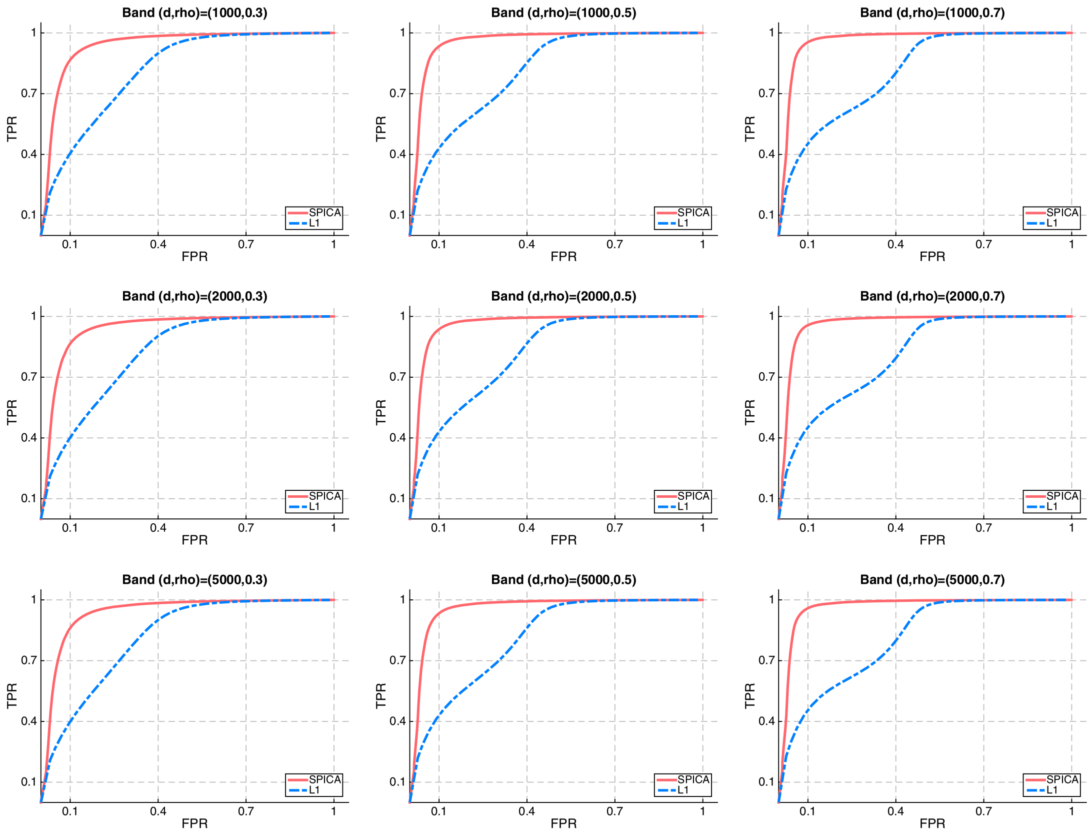

Fig. 6 ROC curves for band model under different settings

similarly under the scale-free model. Since the degree of the block and band models are larger than the degree of the scale-free model, this gives us a hint that the SPICA algorithm works better than the ℓ_1-penalized method when the number of edges of the graph is larger. Also, we notice that the spatial information is more naturally adopted in the block and band models due to their structures. In addition, for block and band models, we observe that the margin of the SPICA algorithm over the ℓ_1-penalized method increases when ρ increases. This phenomenon has an intuitive explanation that the penalization term $\lambda \sum_{j\in[d]} \|\boldsymbol{\beta}_j\|_1$ increases with the signal strength of $\boldsymbol{\beta}_j^*$'s, which induces more estimation bias, and results a worse performance in graph recovery.

We then compare the SPICA algorithm with the ℓ_1-penalized method and the ISEE method from the perspective of parameter estimation. The tuning parameters are chosen by five-fold cross-validation for SPICA and the ℓ_1-penalized methods. Note that the ISEE method selects the tuning parameter based on the scaled Lasso [23]. We repeat each setting mentioned above 100 times, and we report the averaged errors $\sum_{j\in[d]} \|\widehat{\boldsymbol{\beta}}_j - \boldsymbol{\beta}_j^*\|_2^2$ and stand deviations in Table 1. We observe that the ISEE method always outperforms the ℓ_1-penalized method, and the SPICA algorithm performs better than the ℓ_1-penalized and ISEE methods as the degree or the signal strength increases. In addition, we observe some interesting phenomenon. In the scale-free and block models, the errors decrease as ρ increases for all three methods. This is intuitive that as the increase in signal strength helps graph recovery, and consequently it also helps parameters estimation. In the band model, same as the scale-free and block models, the errors decrease as ρ increases for the SPICA algorithm. However, the errors increase as ρ increases for the ℓ_1-penalized and the ISEE method. This again confirms the intuition that as the ℓ_1-norm $\sum_{j\in[d]} \|\boldsymbol{\beta}_j^*\|_1$ increases, the penalization terms induce more biases. In comparison, the total cardinality constraint approach does not induce any bias.

To summarize, the advantage of the SPICA algorithm over convex methods is well illustrated from the perspectives of both graph support recovery and parameters estimation. We also point out that the certificate of primal optimality, i.e., $\sum_{j\in[d]} \|\widehat{\boldsymbol{\beta}}_j\|_0 = K$, holds in more than 98% cases, which means that the SPICA algorithm generates the optimal solution to the problem with total cardinality constraint in these cases. This further shows the reliability of the SPICA algorithm.

Meanwhile, from the computational perspective, we see that the SPICA algorithm runs 10–15 times longer than the convex methods. However, in our current implementation, we directly use `gurobi` [18] to solve the problems in each step. Meanwhile, as pointed out in recent literature such as Bertsimas et al. [2], it is shown that by utilizing modern mixed integer program techniques, the computational speed can be substantially accelerated. We believe that our method can be substantially accelerated if such techniques are fully implemented.

5.2 Sensor network data

We also use wireless sensor network data to conduct tests. Our goal is to estimate how the sensors are connected. In practical applications, depending on the sensor type, the communication network of sensors might be known or not. In our data, the

Table 1 Quantitative comparisons of the SPICA, ℓ_1-penalized method and the ISEE method on different models

Model	n	d	Method	$\rho = 0.3$		$\rho = 0.5$		$\rho = 0.7$	
				Error	Time (s)	Error	Time (s)	Error	Time (s)
Scale-free	100	1000	SPICA	59.859	35.347	53.751	34.494	50.834	36.294
			ℓ_1	60.124	2.573	53.844	2.358	50.472	2.857
			ISEE	59.849	1.459	53.785	1.577	50.339	1.736
		2000	SPICA	120.412	71.471	108.872	69.207	102.521	70.384
			ℓ_1	120.635	5.184	108.746	4.781	102.418	5.804
			ISEE	120.357	3.397	108.559	3.274	102.276	3.716
		5000	SPICA	307.534	173.466	275.815	178.791	256.142	174.583
			ℓ_1	305.464	11.481	272.615	13.841	254.370	11.728
			ISEE	304.395	8.957	271.389	7.817	253.201	8.893
Block	100	1000	SPICA	99.644	38.489	83.205	37.489	71.874	40.188
			ℓ_1	99.418	3.184	91.815	3.368	79.562	2.957
			ISEE	99.391	2.857	89.405	2.503	75.491	2.490
		2000	SPICA	201.841	77.694	169.942	73.909	144.481	80.544
			ℓ_1	202.031	6.530	181.418	6.819	157.906	5.938
			ISEE	201.958	5.788	175.356	5.491	157.906	5.185
		5000	SPICA	516.913	198.437	431.546	195.359	365.976	203.495
			ℓ_1	517.096	15.651	465.967	16.892	396.485	14.859
			ISEE	516.830	14.185	458.359	13.378	384.092	13.408
Band	100	1,000	SPICA	88.146	35.398	82.792	33.420	76.849	32.810
			ℓ_1	90.421	3.530	95.655	3.394	98.615	3.429
			ISEE	89.823	3.049	93.459	2.894	88.759	2.949
		2000	SPICA	175.706	72.586	162.097	68.108	158.951	63.450
			ℓ_1	181.277	7.322	186.695	6.579	205.439	6.993
			ISEE	178.595	5.930	173.341	5.849	184.672	6.189
		5000.	SPICA	448.765	178.418	415.445	167.390	396.709	159.423
			ℓ_1	458.162	17.937	482.276	16.859	512.680	17.359
			ISEE	451.392	15.595	454.108	14.819	463.410	15.049

We report the averaged Frobenius norm $\sum_{j\in[d]} \|\widehat{\boldsymbol{\beta}}_j - \boldsymbol{\beta}_j^*\|_2^2$ and running times in seconds after repeating the simulation 100 times

communication network is given. The reason we choose this type of data is that our primary goal is to evaluate the two different methods, and without such information, it is difficult to tell which method works better. In the implementation of different methods, we do not use the information of how the sensors are connected, and we only use such information to evaluate the results at a later stage.

As discussed in the introduction, in a sensor network, each sensor can only connect to another if they are sufficiently close as illustrated in Fig. 7. Thus, estimating the network of sensors fits into the spatial graphical model framework. In our data, we have $d = 3592$ sensors, and each sensor can only connect with another if they are

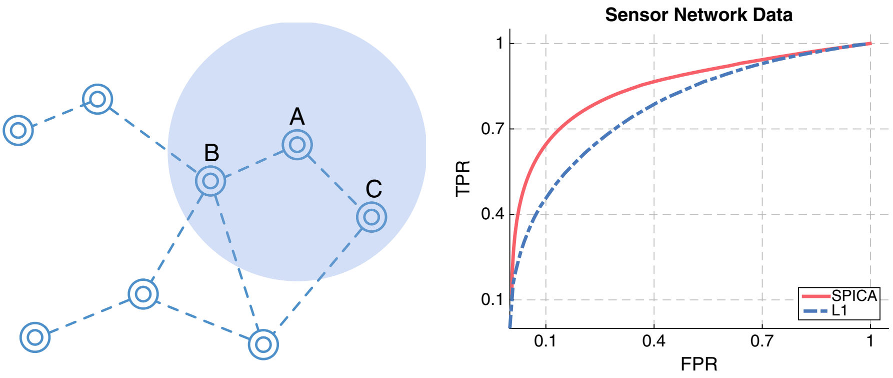

Fig. 7 Left: in the sensor network, each sensor can only connect to another sensor if they are physically close on the plane. Each dashed line represents a possible connection. For example, sensor A can only possibly connect to B or C, but not others. Right: ROC curve for sensor network data

within 3 m. On average, each sensor has 24 potential neighbors. We have in total $n = 98$ samples. Each sample contains a signal strength of each sensor. Taking a Gaussian graphical model approach, we test the SPICA algorithm and ℓ_1-penalized method. We plot the ROC curves of the SPICA algorithm and the ℓ_1-penalized method in Fig. 7. It is clear that the SPICA algorithm performs significantly better than the ℓ_1-penalized method. This shows that the SPICA algorithm is capable of estimating spatial graphical models in practice.

6 Conclusion

To conclude, we propose a practical SPICA algorithm for spatial graphical model estimation for the total cardinality constraint approach. We solve the problem by considering the Lagrangian dual problem. Though the problem is nonconvex, we prove that the average-per-vertex duality gap decreases as the dimension d increases, and we achieve optimal statistical properties if the dimension d is sufficiently large. We conduct thorough numerical experiments to backup our theory. We further provide several new fundamental results to better understand the total cardinality constraint approach for the spatial graphical model. We prove that the decision version of problem (1.1) is NP-complete. For the case $\mathcal{R}_j(\boldsymbol{\beta}_j) = \|\boldsymbol{\beta}_j\|_0$, we show that the problem is polynomial-time solvable.

In our current implementation, we use cross-validation to choose the tuning parameter, which works well empirically. We also tried the BIC method, which gives similar results empirically. To the best of our knowledge, there is not much literature for tuning parameter selection for discrete methods under the high-dimensional setting, which is a challenging problem. We will work on this problem in the future to develop theoretically justified method for the tuning parameter selection.

For future work, we will continue to develop efficient algorithms to attack different cardinality constrained problems without sacrificing statistical efficiencies. We also plan to apply the proposed method to conduct real-world applications, such as brain functional region partition.

References

1. Bertsekas, D.P.: Nonlinear Programming. Athena Scientific, Belmont (1999)
2. Bertsimas, D., King, A., Mazumder, R.: Best subset selection via a modern optimization lens. Ann. Stat. **44**, 813–852 (2016)
3. Cai, T., Liu, W., Luo, X.: A constrained ℓ_1 minimization approach to sparse precision matrix estimation. J. Am. Stat. Assoc. **106**, 594–607 (2011)
4. Cao, L., Fei-Fei, L.: Spatially coherent latent topic model for concurrent segmentation and classification of objects and scenes. In: IEEE 11th International Conference on Computer Vision, 2007. ICCV 2007. IEEE (2007)
5. Fan, J., Feng, Y., Wu, Y.: Network exploration via the adaptive lasso and scad penalties. Ann Appl Stat **3**, 521 (2009)
6. Fan, Y., Lv, J.: Asymptotic equivalence of regularization methods in thresholded parameter space. J. Am. Stat. Assoc. **108**, 1044–1061 (2013)
7. Fan, Y., Lv, J.: Innovated scalable efficient estimation in ultra-large Gaussian graphical models. Ann. Stat. **44**, 2098–2126 (2016)
8. Hall, P., Jin, J.: Innovated higher criticism for detecting sparse signals in correlated noise. Ann. Stat. **38**, 1686–1732 (2010)
9. Howard, A., Matarić, M. J., Sukhatme, G. S.: Mobile sensor network deployment using potential fields: a distributed, scalable solution to the area coverage problem. In: Asama, H., Arai, T., Fukuda, T., Hasegawa, T. (eds.) Distributed Autonomous Robotic Systems, Vol. 5. Springer, pp. 299–308 (2002)
10. Langendoen, K., Baggio, A., Visser, O.: Murphy loves potatoes: experiences from a pilot sensor network deployment in precision agriculture. In: Proceedings 20th IEEE International Parallel and Distributed Processing Symposium. IEEE (2006)
11. Lee, S.H., Lee, S., Song, H., Lee, H.S.: Wireless sensor network design for tactical military applications: remote large-scale environments. In: Military Communications Conference, 2009. MILCOM 2009. IEEE. IEEE (2009)
12. Liu, H., Wang, L.: Tiger: a tuning-insensitive approach for optimally estimating Gaussian graphical models. arXiv preprint arXiv:1209.2437 (2012)
13. Liu, W.: Gaussian graphical model estimation with false discovery rate control. Ann Stat **41**, 2948–2978 (2013)
14. Magazine, M.J., Chern, M.-S.: A note on approximation schemes for multidimensional knapsack problems. Math. Oper. Res. **9**, 244–247 (1984)
15. Meinshausen, N., Bühlmann, P.: High-dimensional graphs and variable selection with the Lasso. Ann. Stat. **34**, 1436–1462 (2006)
16. Meinshausen, N., Bühlmann, P.: Stability selection. J. R. Stat. Soc. Ser. B Stat. Methodol. **72**, 417–473 (2010)
17. Meinshausen, N., Yu, B.: Lasso-type recovery of sparse representations for high-dimensional data. Ann. Stat. **37**, 246–270 (2009)
18. Optimization, G.: Inc.,"gurobi optimizer reference manual," 2015. (2014). http://www.gurobi.com. Accessed 29 Sept 2018
19. Pisinger, D.: A minimal algorithm for the multiple-choice knapsack problem. Eur. J. Oper. Res. **83**, 394–410 (1995)
20. Ravikumar, P., Wainwright, M.J., Raskutti, G., Yu, B.: High-dimensional covariance estimation by minimizing ℓ_1-penalized log-determinant divergence. Electron. J. Stat. **5**, 935–980 (2011)
21. Ren, Z., Sun, T., Zhang, C.-H., Zhou, H.H.: Asymptotic normality and optimalities in estimation of large Gaussian graphical models. Ann. Stat. **43**, 991–1026 (2015)
22. Starr, R.M.: Quasi-equilibria in markets with non-convex preferences. Econometrica **37**(1), 25–38 (1969)
23. Sun, T., Zhang, C.-H.: Scaled sparse linear regression. Biometrika **99**, 879–898 (2012)
24. Williamson, D.P., Shmoys, D.B.: The Design of Approximation Algorithms. Cambridge University Press, Cambridge (2011)

25. Yick, J., Mukherjee, B., Ghosal, D.: Wireless sensor network survey. Comput. Netw. **52**, 2292–2330 (2008)
26. Yuan, M., Lin, Y.: Model selection and estimation in the Gaussian graphical model. Biometrika **94**, 19–35 (2007)
27. Zhang, T.: On the consistency of feature selection using greedy least squares regression. J. Mach. Learn. Res. **10**, 555–568 (2009)
28. Zhang, Y., Wainwright, M.J., Jordan, M.I.: Lower bounds on the performance of polynomial-time algorithms for sparse linear regression. In: Proceedings of Annual Conference on Learning Theory (2014)
29. Zhao, P., Yu, B.: On model selection consistency of Lasso. J. Mach. Learn. Res. **7**, 2541–2563 (2006)