AN ASYMPTOTICALLY OPTIMAL COORDINATE DESCENT ALGORITHM FOR LEARNING BAYESIAN NETWORKS FROM GAUSSIAN MODELS*

TONG XU[†], ARMEEN TAEB[‡], SIMGE KÜÇÜKYAVUZ[§], AND ALI SHOJAIE[¶]

Abstract. This paper studies the problem of learning Bayesian networks from continuous observational data, generated according to a linear Gaussian structural equation model. We consider an ℓ_0-penalized maximum likelihood estimator for this problem which is known to have favorable statistical properties but is computationally challenging to solve, especially for medium-sized Bayesian networks. We propose a new coordinate descent algorithm to approximate this estimator and prove several remarkable properties of our procedure: the algorithm converges to a coordinate-wise minimum, and despite the non-convexity of the loss function, as the sample size tends to infinity, the objective value of the coordinate descent solution converges to the optimal objective value of the ℓ_0-penalized maximum likelihood estimator. Finite-sample optimality and statistical consistency guarantees are also established. To the best of our knowledge, our proposal is the first coordinate descent procedure endowed with optimality and statistical guarantees in the context of learning Bayesian networks. Numerical experiments on synthetic and real data demonstrate that our coordinate descent method can obtain near-optimal solutions while being scalable.

Key words. Directed acyclic graphs, ℓ_0-penalization, Non-convex optimization, Structural equation models

MSC codes. 65K10, 68T20, 68Q25

1. Introduction.

1.1. Background and related work. Bayesian networks provide a powerful framework for modeling causal relationships among a collection of random variables. A Bayesian network is typically represented by a directed acyclic graph (DAG), where the random variables are encoded as vertices (or nodes), a directed edge from node i to node j indicates that i causes j, and the acyclic property of the graph prevents the occurrence of circular dependencies. If the DAG is known, it can be used to predict the behavior of the system under manipulations or interventions. However, in large systems such as gene regulatory networks, the DAG is not known a priori, making it necessary to develop efficient and rigorous methods to learn the graph from data. To solve this problem using only observational data, we assume that all relevant variables are observed and that we only have access to observational data.

Three broad classes of methods for learning DAGs from data are constraint-based, score-based, and hybrid. Constraint-based methods use repeated conditional independence tests to determine the presence of edges in a DAG. A prominent example is the PC algorithm and its extensions [20, 21]. While the PC algorithm can be applied in non-parametric settings, testing for conditional independencies is generally hard

*Submitted to the editors on August 21, 2024.

Funding: Simge Küçükyavuz and Tong Xu are supported, in part, by the Office of Naval Research Global [Grant N00014-22-1-2602]. Armeen Taeb is supported by the Royalty Research Fund at the University of Washington. Ali Shojaie is supported by grant R01GM133848 from the National Institutes of Health.

[†]Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL (tongxu2027@u.northwestern.edu).

[‡]Department of Statistics, University of Washington, Seattle, WA (ataeb@uw.edu).

[§]Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL (simge@northwestern.edu).

[¶]Department of Biostatistics, University of Washington, Seattle, WA (ashojaie@uw.edu).

[17]. Furthermore, even in the Gaussian setting, statistical consistency guarantees for the PC algorithm are shown under the *strong faithfulness* condition [12], which is known to be restrictive in high-dimensional settings [22]. Score-based methods often deploy a penalized log-likelihood as a score function and search over the space of DAGs to identify a DAG with an optimal score. These approaches do not require the strong faithfulness assumption. However, statistical guarantees are not provided for many score-based approaches and solving them exactly suffers from high computational complexity. For example, learning an optimal graph using dynamic programming takes about 10 hours for a medium-size problem with 29 nodes [18]. Several papers [13, 25] offer speedup by casting the problem as a convex mixed-integer program, but finding an optimal solution with these approaches can still take an hour for a medium-sized problem. Finally, hybrid approaches combine constraint-based and score-based methods by using background knowledge or conditional independence tests to restrict the DAG search space [21, 16].

Several strategies have been developed to make score-based methods more scalable by finding approximate solutions instead of finding optimally scoring DAGs. One direction to find good approximate solutions is to resort to greedy-based methods, with a prominent example being the Greedy Equivalence Search (GES) algorithm [5]. GES performs a greedy search on the space of completed partially directed acyclic graphs (an equivalence class of DAGs) and is known to produce asymptotically consistent solutions [5]. Despite its favorable properties, GES does not provide optimality or consistency guarantees for any finite sample size. Further, the guarantees of GES assume a fixed number of nodes with sample size going to infinity and do not allow for a growing number of nodes. Another direction is gradient-based approaches [27, 28], which relax the discrete search space over DAGs to a continuous search space, allowing gradient descent and other techniques from continuous optimization to be applied. However, the search space for these problems is highly non-convex, resulting in limited guarantees for convergence, even to a local minimum. Finally, another notable direction is based on *coordinate descent*; that is iteratively maximizing the given score function over a single parameter, while keeping the remaining parameters fixed and checking that the resulting model is a DAG at each update [1, 2, 9, 26]. While coordinate descent algorithms have shown significant promise in learning large-scale Bayesian networks, to the best of our knowledge, they do not come with convergence, optimality, and statistical guarantees.

1.2. Our contributions. We propose a new score-based coordinate descent algorithm for learning Bayesian networks from Gaussian linear structural equation models. Remarkably, unlike prior coordinate descent algorithms for learning Bayesian networks, our procedure provably i) converges to a coordinate-wise minimum, ii) produces optimally scoring DAGs as the sample size tends to infinity despite the non-convex nature of the problem, and iii) yields asymptotically consistent estimates that also provide finite-sample guarantees that allow for a growing number of nodes. As a scoring function for this approach, we deploy an ℓ_0-penalized Gaussian log-likelihood, which implies that optimally-scoring DAGs are solutions to a highly non-convex ℓ_0-penalized maximum likelihood estimator. This estimator is known to have strong statistical consistency guarantees [23], but solving it is, in general, intractable. Thus, our coordinate descent algorithm can be viewed as a scalable and efficient approach to finding approximate solutions to this estimator that are asymptotically optimal (i.e., match the optimal objective value of the ℓ_0 penalized maximum-likelihood estimator as the sample size tends to infinity) and have finite-sample statistical consistency guarantees.

We illustrate the advantages of our method over competing approaches via extensive numerical experiments. The proposed approach is implemented in the python package *micodag*, and all numerical results and figures can be reproduced using the code in https://github.com/AtomXT/coordinate-descent-for-bayesian-networks.git.

2. Problem Setup.

2.1. Preliminaries and Definitions. Consider an unknown DAG whose m nodes correspond to observed random variables $X \in \mathbb{R}^m$. We denote the DAG by $\mathcal{G}^\star = (V, E^\star)$ where $V = \{1, \ldots, m\}$ is the vertex set and $E^\star \subseteq V \times V$ is the directed edge set. We assume that the random variables X satisfy the linear structural equation model (SEM):

$$X = B^{\star \mathrm{T}} X + \epsilon, \qquad (2.1)$$

where $B^\star \in \mathbb{R}^{m\times m}$ is the connectivity matrix with zeros on the diagonal and $B^\star_{jk} \neq 0$ if $(j,k) \in E^\star$. In other words, the sparsity pattern of $B^\star$ encodes the true DAG structure. Further, $\epsilon \sim \mathcal{N}(0, \Omega^\star)$ is a random Gaussian noise vector with zero mean and independent coordinates so that $\Omega^\star$ is a diagonal matrix. Assuming, without loss of generality, that all random variables are centered, each variable X_j in this model can be expressed as the linear combination of its parents—the set of nodes with directed edges pointing to j—plus independent Gaussian noise. By the SEM (2.1) and the Gaussianity of ϵ, the random vector X follows the Gaussian distribution $\mathcal{P}^\star = \mathcal{N}(0, \Sigma^\star)$, with $\Sigma^\star = (I - B^\star)^{-\mathrm{T}} \Omega^\star (I - B^\star)^{-1}$. Throughout, we assume that the distribution $\mathcal{P}^\star$ is non-degenerate, or equivalently, $\Sigma^\star$ is positive definite. Our objective is to estimate the matrix $B^\star$, or as we describe next, an equivalence class when the underlying model is not identifiable.

Multiple SEMs are generally compatible with the distribution $\mathcal{P}^\star$. To formalize this, we need the following definition.

Definition 2.1. *(Graph $\mathcal{G}(B)$ induced by B) Let $B \in \mathbb{R}^{m\times m}$ with zeros on the diagonal. Then, $\mathcal{G}(B)$ is the directed graph on m nodes where the directed edge from i to j appears in $\mathcal{G}(B)$ if and only if $B_{ij} \neq 0$.*

To see why the model (2.1) is generally not identifiable, note that there are multiple tuples (B, Ω) where $\mathcal{G}(B)$ is DAG and Ω is a positive definite diagonal matrix with $\Sigma^\star = (I - B)^{-\mathrm{T}} \Omega (I - B)^{-1}$ [23]. As a result, the SEM given by (B, Ω) yields an equally representative model as the one given by the population parameters $(B^\star, \Omega^\star)$. When $\mathcal{G}^\star$ is *faithful* with respect to the graph $\mathcal{G}^\star$ (see Assumption 7 in Section 4 for a formal definition), the sparsest DAGs that are compatible with $\mathcal{P}^\star$ are precisely $\mathrm{MEC}(\mathcal{G}^\star)$, the *Markov equivalence class* of $\mathcal{G}^\star$ [23]. Next, we formally define the Markov equivalence class.

Definition 2.2. *(Markov equivalence class $\mathrm{MEC}(\mathcal{G})$ [24]) Let $\mathcal{G} = (V, E)$ be a DAG. Then, $\mathrm{MEC}(\mathcal{G})$ consists of DAGs that have the same skeleton and same v-structures as $\mathcal{G}$. The skeleton of $\mathcal{G}$ is the undirected graph obtained from $\mathcal{G}$ by substituting directed edges with undirected ones. Furthermore, nodes i, j, and k form a v-structure if $(i,k) \in E$ and $(j,k) \in E$, and there is no edge between i and j.*

2.2. ℓ_0-Penalized Maximum Likelihood Estimator. Consider n independent and identically distributed observations of the random vector X generated according to (2.1). Let $\hat{\Sigma}$ be the sample covariance matrix obtained from these observations. Further, consider a Gaussian SEM parameterized by connectivity matrix B

and noise variance Ω with $D = \Omega^{-1}$. The parameters (B, D) specify the following precision, or inverse covariance, matrix $\Theta := \Theta(B, D) := (I - B)D(I - B)^{\mathrm{T}}$. The negative log-likelihood of this SEM is proportional to $\ell_n(\Theta) = \mathrm{trace}(\Theta\hat{\Sigma}) - \log\det(\Theta)$. Naturally, we seek a model that not only has a small negative log-likelihood but is also specified by a sparse connectivity matrix containing few nonzero elements. Thus, we deploy the following ℓ_0-penalized maximum likelihood estimator with a regularization parameter $\lambda \geq 0$:

$$(2.2) \qquad \min_{B\in\mathbb{R}^{m\times m}, D\in\mathbb{D}^m_{++}} \ell_n\left((I-B)\,D\,(I-B)^{\mathrm{T}}\right) + \lambda^2\|B\|_{\ell_0} \quad \text{s.t.} \quad \mathcal{G}(B) \text{ is a DAG.}$$

Here, $\mathbb{D}^m_{++}$ denotes the collection of positive definite $m \times m$ diagonal matrices and $\|B\|_{\ell_0}$ denotes the number of non-zeros in B. Note that the ℓ_0 penalty is generally preferred over the ℓ_1 penalty or minimax concave penalty (MCP) for penalizing the complexity of the model. In particular, ℓ_0 regularization exhibits the important property that equivalent DAGs—those in the same Markov equivalence class—have the same penalized likelihood score, while this is not the case for ℓ_1 or MCP regularization [23]. Indeed, this lack of score invariance with ℓ_1 regularization partially explains the unfavorable properties of some existing methods (see Section 5).

The Markov equivalence class $\mathrm{MEC}(\mathcal{G}(\hat{B}^{\mathrm{opt}}))$ of the connectivity matrix $\hat{B}^{\mathrm{opt}}$ obtained from solving (2.2) provides an estimate of $\mathrm{MEC}(\mathcal{G}^\star)$. van de Geer and Bühlmann [23] prove that this estimate has desirable statistical properties; however, solving it is, in general, intractable. As stated, the objective function $\ell_n((I-B)D(I-B)^{\mathrm{T}})$ is non-convex and non-linear function of (B, D). Furthermore, the log det function in the likelihood ℓ_n is not amenable to standard mixed-integer programming optimization techniques. To circumvent the aforementioned challenges, Xu et al. [25] derive the following equivalent optimization model via the change of variables $\Gamma \leftarrow (I - B)D^{1/2}$:

$$(2.3) \qquad \min_{\Gamma\in\mathbb{R}^{m\times m}} f(\Gamma) \quad \text{s.t.} \quad \mathcal{G}\left(\Gamma - \mathrm{diag}\left(\Gamma\right)\right) \text{ is a DAG.}$$

Here $f(\Gamma) := \sum_{i=1}^m -2\log(\Gamma_{ii}) + \mathrm{tr}(\Gamma\Gamma^{\mathrm{T}}\hat{\Sigma}_n) + \lambda^2\|\Gamma - \mathrm{diag}(\Gamma)\|_{\ell_0}$, and $\mathrm{diag}(\Gamma)$ is the diagonal matrix formed by taking the diagonal entries of Γ. The optimal solutions of (2.2) and (2.3) are directly connected: Letting $(\hat{B}^{\mathrm{opt}}, \hat{D}^{\mathrm{opt}})$ be an optimal solution of (2.2), then $\hat{\Gamma}^{\mathrm{opt}} = (I - \hat{B}^{\mathrm{opt}})(\hat{D}^{\mathrm{opt}})^{1/2}$ is an optimal solution of (2.3). Furthermore, the sparsity pattern of $\hat{\Gamma}^{\mathrm{opt}} - \mathrm{diag}(\hat{\Gamma}^{\mathrm{opt}})$ is the same as that of $\hat{B}^{\mathrm{opt}}$; in other words, the Markov equivalence class $\mathrm{MEC}(\mathcal{G}(\hat{B}^{\mathrm{opt}}))$ is the same as the Markov equivalence class $\mathrm{MEC}(\mathcal{G}(\hat{\Gamma}^{\mathrm{opt}} - \mathrm{diag}(\hat{\Gamma}^{\mathrm{opt}})))$.

Xu et al. [25] recast the optimization problem (2.3) as a convex mixed-integer program and provide algorithms to solve (2.3) to optimality. However, solving (2.3) is, in general, NP-hard, and obtaining optimality certificates may take an hour for a problem with 20 nodes [25].

3. A Coordinate Descent Algorithm for DAG Learning.

In this section, we develop a cyclic coordinate descent approach to find a heuristic solution to problem (2.3). The coordinate descent solver is fast and can be scaled to large-scale problems. As we demonstrate in Section 4, it provably converges and produces an asymptotically optimal solution to (2.3). Given the quality of its estimates, the proposed coordinate descent algorithm can also be used as a warm start for the mixed-integer programming framework in [25] to obtain optimal solutions.

3.1. Parameter update without acyclicity constraints. Let us first ignore the acyclicity constraint in (2.3), and consider solving problem (2.3) with respect to a single variable Γ_{uv}, for $u, v = 1, \ldots, m$, with the other coordinates of Γ fixed. Specifically, we are solving

$$\min_{\Gamma_{uv} \in \mathbb{R}} g(\Gamma_{uv}) \coloneqq \sum_{i=1}^{m} -2\log(\Gamma_{ii}) + \operatorname{tr}\left(\Gamma\Gamma^{\mathsf{T}}\hat{\Sigma}\right) + \lambda^2 \|\Gamma - \operatorname{diag}(\Gamma)\|_{\ell_0}, \tag{3.1}$$

with Γ_{ij} being fixed for $i \neq u, j \neq v$.

Proposition 3.1. *The solution to problem* (3.1)*, for* $u, v = 1, \ldots, m$ *and* $v \neq u$ *is given by*

$$\hat{\Gamma}_{uv} = \begin{cases} \frac{-A_{uv}}{2\hat{\Sigma}_{uu}}, & \textit{if } \lambda^2 \leq \frac{A_{uv}^2}{4\hat{\Sigma}_{uu}}, \\ 0, & \textit{otherwise.} \end{cases} \quad ; \quad \hat{\Gamma}_{uu} = \frac{-A_{uu} + \sqrt{A_{uu}^2 + 16\hat{\Sigma}_{uu}}}{4\hat{\Sigma}_{uu}},$$

where $A_{uu} = \sum_{j \neq u} \Gamma_{ju}\hat{\Sigma}_{ju} + \sum_{k \neq u} \Gamma_{ku}\hat{\Sigma}_{uk}$ *and* $A_{uv} = \sum_{j \neq u} \Gamma_{jv}\hat{\Sigma}_{ju} + \sum_{k \neq u} \Gamma_{kv}\hat{\Sigma}_{uk}$.

Proof. For any $u \in V$, we have

$$\operatorname{tr}\left(\Gamma\Gamma^{\mathsf{T}}\hat{\Sigma}\right) = \sum_{i=1}^{m} \Gamma_{ui}\left(\Gamma_{ui}\hat{\Sigma}_{uu} + \sum_{j \neq u} \Gamma_{ji}\hat{\Sigma}_{ju}\right) + \sum_{k \neq u}\sum_{i=1}^{m} \Gamma_{ki}\left(\Gamma_{ui}\hat{\Sigma}_{uk} + \sum_{j \neq u} \Gamma_{ji}\hat{\Sigma}_{jk}\right).$$

We first consider Γ_{uv} for $u \neq v$. The derivative of $g(\Gamma_{uv})$ with respect to Γ_{uv} is:

$$\frac{\partial g(\Gamma_{uv})}{\partial \Gamma_{uv}} = \frac{\partial \operatorname{tr}(\Gamma\Gamma^{\mathsf{T}}\hat{\Sigma})}{\partial \Gamma_{uv}} = 2\hat{\Sigma}_{uu}\Gamma_{uv} + \sum_{j \neq u} \Gamma_{jv}\hat{\Sigma}_{ju} + \sum_{k \neq u} \Gamma_{kv}\hat{\Sigma}_{uk} = 2\hat{\Sigma}_{uu}\Gamma_{uv} + A_{uv}.$$

Setting $\partial g(\Gamma_{uv})/\partial \Gamma_{uv} = 0$, and defining $\hat{\gamma}_{uv} := -A_{uv}/2\hat{\Sigma}_{uu}$, we obtain

$$\arg\min_{\Gamma_{uv}} g(\Gamma_{uv}) = \hat{\Gamma}_{uv} \coloneqq \begin{cases} \hat{\gamma}_{uv}, & \text{if } g(\hat{\gamma}_{uv}) \leq g(0), \\ 0, & \text{otherwise.} \end{cases}$$

The original objective function g with ℓ_0-norm is nonconvex and discontinuous. To find the optimal solution, we compare $g(\hat{\gamma}_{uv})$ with $g(0)$. Given that $g(\hat{\gamma}_{uv})$ represents the optimal objective value for any nonzero Γ_{uv}, comparing it with $g(0)$ allows us to determine the optimal solution. Note that $g(\hat{\gamma}_{uv}) - g(0) = \hat{\gamma}_{uv}^2\hat{\Sigma}_{uu} + \hat{\gamma}_{uv}A_{uv} + \lambda^2$. Thus, $g(\hat{\gamma}_{uv}) \leq g(0)$ is equivalent to $\lambda^2 \leq A_{uv}^2/4\hat{\Sigma}_{uu}$.

Now we consider the update of Γ_{uv} when $u = v$. We have:

$$\frac{\partial g(\Gamma_{uu})}{\partial \Gamma_{uu}} = \frac{-2}{\Gamma_{uu}} + 2\hat{\Sigma}_{uu}\Gamma_{uu} + \sum_{j \neq u} \Gamma_{ju}\hat{\Sigma}_{ju} + \sum_{k \neq u} \Gamma_{ku}\hat{\Sigma}_{uk} = \frac{-2}{\Gamma_{uu}} + 2\hat{\Sigma}_{uu}\Gamma_{uu} + A_{uu}.$$

Setting $\partial g(\Gamma_{uu})/\partial \Gamma_{uu} = 0$, we obtain: $\hat{\Gamma}_{uu} = -A_{uu} + (A_{uu}^2 + 16\hat{\Sigma}_{uu})^{1/2}/4\hat{\Sigma}_{uu}$. ∎

3.2. Accounting for acyclicity and full algorithm description. Algorithm 3.1 fully describes our procedure. The input to our algorithm is the sample covariance $\hat{\Sigma}$, regularization parameter $\lambda \in \mathbb{R}_+$, a super-structure graph E_{super} that is a superset of edges that contains the true edges, and a positive integer C. We allow the user to restrict the set of possible edges to be within a user-specified *super-structure* set of edges E_{super}. A natural choice of the superstructure is the moral graph, which can be efficiently and accurately estimated via existing algorithms such as the graphical lasso [7] or neighborhood selection [15]. This superstructure could also be the complete graph if a reliable superstructure estimate is unavailable.

We start by initializing Γ as the identity matrix. Then, for each pair of indices u and v ranging from 1 to m, we update Γ_{uv} based on specific rules. If $u = v$ (a diagonal entry), we update it directly according to Proposition 3.1. Among the off-diagonal entries, we only update those within the superstructure. Specifically, if $u \neq v$, and (u, v) is in the superstructure, we check if setting Γ_{uv} to a nonzero value violates the acyclicity constraint. (We use the breadth-first search algorithm [e.g., 6, 9] to check for acyclicity.) If it does not, we update Γ_{uv} as per Proposition 3.1; otherwise, we set Γ_{uv} to 0. We refer to a full sequence of coordinate updates as a full loop. The loop is repeated until convergence, when the objective values no longer improve after a complete loop. We keep track of the support of Γs encountered during the algorithm. When the occurrence count of a particular support of Γs reaches a predefined threshold, C, a spacer step [4, 11] is initiated, during which we update every nonzero coordinate iteratively. Note that in the spacer step, we use $\hat{\gamma}_{uv}$, which is the optimal update without considering the sparsity penalty, i.e., we use $\lambda^2 = 0$. The use of spacer steps stabilizes the behavior of updates and ensures convergence. After finishing the spacer step, we reset the counter of the support of the current solution.

Algorithm 3.1 Cyclic coordinate descent algorithm with spacer steps

Input: Sample covariance $\hat{\Sigma}$, regularization parameter $\lambda \in \mathbb{R}_+$, super-structure E_{super}, positive integer C.
Initialize: $\Gamma^0 \leftarrow I$; $t \leftarrow 1$.
while objective function $f(\Gamma^t)$ continue decreasing **do**
  for $u = 1$ to m **do**
    $\Gamma^t_{uu} = \hat{\Gamma}_{uu}$, where $\hat{\Gamma}_{uu}$ is calculated from Proposition 3.1 using the recently updated Γ^t.
    for $v = 1$ to m such that $(u, v) \in E_{\text{super}}$ **do**
      If $\Gamma^t_{uv} \neq 0$ violates acyclicity constraints, set $\Gamma^t_{uv} = 0$.
      If $\Gamma^t_{uv} \neq 0$ would not violate acyclicity constraints, set $\Gamma^t_{uv} = \hat{\Gamma}_{uv}$.
      $t \leftarrow t + 1$
      Count[support(Γ^t)] $\leftarrow$ Count[support(Γ^t)] + 1.
      if Count[support(Γ^t)] $= Cm^2$ **then**
        $\Gamma^{t+1} \leftarrow$ SpacerStep(Γ^t) (Algorithm 3.2)
        Count[support(Γ^t)] = 0.
        $t \leftarrow t + 1$.
      end if
    end for
  end for
end while
Output: $\hat{\Gamma} \leftarrow \Gamma^t$ and the Markov equivalence class $\text{MEC}(\mathcal{G}(\hat{\Gamma} - \text{diag}(\hat{\Gamma})))$

4. Statistical and Optimality Guarantees.

We provide statistical and optimality guarantees for our coordinate descent procedure (Algorithm 3.1). Specifically, we follow a similar proof strategy as [11] to show that Algorithm 3.1 converges. Re-

Algorithm 3.2 SpacerStep

1: **Input:** Γ^t
2: **for** $(u, v) \in \text{support}(\Gamma^t)$ **do**
3: Set $\Gamma_{uv}^{t+1} \leftarrow \hat{\gamma}_{uv}$
4: **end for**
5: **Output:** Γ^{t+1}

markably, we also prove the surprising result that the objective value attained by our coordinate descent algorithm provably converges to the optimal objective value of (2.3). Finally, we build on these results and provide finite-sample statistical consistency guarantees. Throughout, we assume the super-structure E_{super} that is supplied as input to Algorithm 3.1 satisfies $E^\star \subseteq E_{\text{super}}$ where $E^\star$ denotes the true edge set; see [25] for a discussion on how the graphical lasso can yield super-structures that satisfy this property with high probability.

4.1. Convergence and optimality guarantees. Our convergence analysis requires an assumption on the sample covariance matrix:

ASSUMPTION 1 (Positive definite sample covariance). *The sample covariance matrix $\hat{\Sigma}$ is positive definite.*

Assumption 1 is satisfied almost surely if $n \geq m$ and the samples of the random vector X are generated from an absolutely continuous distribution. Under this mild assumption, our coordinate descent algorithm provably converges, as shown next.

THEOREM 4.1 (Convergence of Algorithm 3.1). *Let $\{\Gamma^t\}_{t=1}^{\infty}$ be the sequence of estimates generated by Algorithm 3.1. Suppose that Assumption 1 holds. Then,*

1. *the sequence $\{\text{support}(\Gamma^t)\}_{t=1}^{\infty}$ stabilizes after a finite number of iterations; that is, there exists a positive integer M and a support set $\hat{E} \subseteq \{(i,j) : i,j = 1,2,\ldots,m\}$ such that $\text{support}(\Gamma^t) = \hat{E}$ for all $t \geq M$.*
2. *the sequence $\{\Gamma^t\}_{t=1}^{\infty}$ converges to a matrix Γ with $\text{support}(\Gamma) = \hat{E}$.*

The proof of Theorem 4.1 relies on the following definitions and lemmas, and it closely follows the approach outlined in [11]. With a slight abuse of notation, we let $\ell(\Gamma) := \sum_{i=1}^{m} -2\log(\Gamma_{ii}) + \text{tr}(\Gamma\Gamma^{\mathsf{T}}\hat{\Sigma}_n)$ to be the negative log-likelihood function associated with parameter $\Gamma \in \mathbb{R}^{m\times m}$.

DEFINITION 4.2 (Coordinate-wise (CW) minimum [11]). *A connectivity matrix $\Gamma^{\text{CW}} \in \mathbb{R}^{m\times m}$ of a DAG is the CW minimum of problem (2.3) if for every (u,v), $u,v = 1,\ldots,m$, Γ_{uv}^{CW} is a minimizer of $g(\Gamma_{uv})$ with other coordinates of Γ^{CW} held fixed.*

LEMMA 4.3. *Let $\{\Gamma^j\}_{j=1}^{\infty}$ be the sequence generated by Algorithm 3.1. Then the sequence of objective values $\{f(\Gamma^j)\}_{j=1}^{\infty}$ is decreasing and converges.*

Proof. By Assumption 1, $\ell(\Gamma)$ is strongly convex and thus bounded below, and so is $f(\Gamma)$. If Γ^j is the result of a non-spacer step, then the inequality $f(\Gamma^j) \leq f(\Gamma^{j-1})$ holds trivially. Similarly, we know that if Γ^j results from a spacer step, then, $\ell(\Gamma^j) \leq \ell(\Gamma^{j-1})$. Since a spacer step updates only coordinates on the support, it cannot increase the support size of Γ^{j-1}, i.e., $\|\Gamma^j - \text{diag}(\Gamma^j)\|_{\ell_0} \leq \|\Gamma^{j-1} - \text{diag}(\Gamma^{j-1})\|_{\ell_0}$, thus $f(\Gamma^j) \leq f(\Gamma^{j-1})$. Since $f(\Gamma^j)$ is non-increasing and bounded below, it must converge. ∎

LEMMA 4.4. *The sequence $\{\Gamma^t\}_{t=1}^{\infty}$ generated by Algorithm 3.1 is bounded.*

Proof. By Algorithm 3.1, $\Gamma^t \in G := \{\Gamma \in \mathbb{R}^{m\times m} \mid f(\Gamma) \le f(\Gamma^0)\}$. It suffices to show that the set G is bounded. From Proposition 11.11 in [3], if the function f is coercive, then the set G is bounded. Since $f(\Gamma) \ge \ell(\Gamma)$ for every Γ, it suffices to show that the function ℓ is coercive. By Assumption 1, we have that the function ℓ is strongly convex. The lemma then follows from the classical result in convex analysis that strongly convex functions are coercive. ∎

The following lemma characterizes the limit points of Algorithm 3.1.

LEMMA 4.5. *Let $\hat{E}$ be a support set that is generated infinitely often by the non-spacer steps of Algorithm 3.1, and let $\left\{\Gamma^l\right\}_{l\in L}$ be the estimates from the spacer steps when the support of the input matrix is $\hat{E}$. Then:*

1. *There exists a positive integer M such that for all $l \in L$ with $l \ge M$,* $\operatorname{support}(\Gamma^l) = \hat{E}$.
2. *There exists a subsequence of $\left\{\Gamma^l\right\}_{l\in L}$ that converges to a stationary solution Γ^{CW}, where, Γ^{CW} is the unique minimizer of* $\min_{\operatorname{support}(\Gamma)\subseteq \hat{E}} \ell(\Gamma)$.
3. *Every subsequence of $\{\Gamma^t\}_{t\ge 0}$ with support $\hat{E}$ converges to Γ^{CW}.*

Proof. **Part 1.)** Since spacer steps optimize only over the coordinates in $\hat{E}$, no element outside $\hat{E}$ can be added to the support. Thus, for every $l \in L$ we have $\operatorname{support}(\Gamma^l) \subseteq \hat{E}$. We next show that strict containment is not possible via contradiction. Suppose $\operatorname{Supp}(\Gamma^l) \subsetneq \hat{E}$ occurs infinitely often, and consider some $l \in L$ where this occurs. By the spacer step of Algorithm 3.1, the previous iterate Γ^{l-1} has support $\hat{E}$, implying $\left\|\Gamma^{l-1}\right\|_0 - \left\|\Gamma^l\right\|_0 \ge 1$. Moreover, from the definition of the spacer step, we have $\ell(\Gamma^l) \le \ell(\Gamma^{l-1})$. Therefore, we get $f(\Gamma^{l-1}) - f(\Gamma^l) = \ell(\Gamma^{l-1}) - \ell(\Gamma^l) + \lambda^2(\|\Gamma^{l-1}\|_0 - \|\Gamma^l\|_0) \ge \lambda^2$. Thus, when $\operatorname{support}(\Gamma^l) \subsetneq \hat{E}$ occurs, f decreases by at least λ^2. Therefore, $\Gamma^l \subsetneq \hat{E}$ infinitely many times implies that $f(\Gamma)$ is not lower-bounded, which is a contradiction.

Part 2.) The proof follows the conventional procedure for establishing the convergence of cyclic coordinate descent (CD) [4, 11]. We obtain Γ^l by updating every coordinate in $\hat{E}$ of Γ^{l-1}. Denote the intermediate steps as $\Gamma^{l,1}, \ldots, \Gamma^{l,|\hat{E}|}$, where $\Gamma^{l,|\hat{E}|} = \Gamma^l$. We aim to show that the sequence $\{\Gamma^{l,|\hat{E}|}\}_{l\in L}$ converges to a point Γ^{CW}, and similarly, other sequences $\{\Gamma^{l,i}\}_{l\in L}, i = 1, \ldots, |\hat{E}| - 1$, also converge to Γ^{CW}. By Lemma 4.4, since $\{\Gamma^{l,|\hat{E}|}\}_{l\in L}$ is a bounded sequence, there exists a converging subsequence $\{\Gamma^{l',|\hat{E}|}\}_{l'\in L'}$ with a limit point Γ^{CW}. Without loss of generality, we choose the subsequence satisfying $l' > M$, $\forall l' \in L'$. From Part 1 of the lemma, $\{\Gamma^{l',1}\}_{l'\in L'}, \ldots, \{\Gamma^{l',|\hat{E}|-1}\}_{l'\in L'}$ all have the same support $\hat{E}$. For $\{\Gamma^{l',|\hat{E}|-1}\}_{l'\in L'}$, we have $f(\Gamma^{l',|\hat{E}|-1}) - f(\Gamma^{l',|\hat{E}|}) = \ell(\Gamma^{l'|\hat{E}|-1}) - \ell(\Gamma^{l',|\hat{E}|})$. If the change from $\Gamma^{l',|\hat{E}|-1}$ to $\Gamma^{l',|\hat{E}|}$ is on a diagonal entry, say Γ_{uu}, then, after some algebra, we obtain

$$\ell\left(\Gamma^{l',|\hat{E}|-1}\right) - \ell\left(\Gamma^{l',|\hat{E}|}\right) =$$
$$2\left(-\log \Gamma_{uu}^{l',|\hat{E}|-1}/\Gamma_{uu}^{l',|\hat{E}|} + \Gamma_{uu}^{l',|\hat{E}|-1}/\Gamma_{uu}^{l',|\hat{E}|} - 1\right) + \left(\Gamma_{uu}^{l',|\hat{E}|-1} - \Gamma_{uu}^{l',|\hat{E}|}\right)^2 \hat{\Sigma}_{uu}.$$

Since $a - 1 \ge \log(a)$ for $a \ge 0$, each of the two terms above is non-negative. From Lemma 4.3, as $l' \to \infty$, $f(\Gamma^{l',|\hat{E}|-1}) - f(\Gamma^{l',|\hat{E}|})$ or equivalently $\ell(\Gamma^{l',|\hat{E}|-1}) - \ell(\Gamma^{l',|\hat{E}|})$ converges to 0 as $l' \to \infty$. Combining this with the fact that $\ell(\Gamma^{l',|\hat{E}|-1}) - \ell(\Gamma^{l',|\hat{E}|}) \ge 0$ and that each term in the equality for $\ell(\Gamma^{l',|\hat{E}|-1}) - \ell(\Gamma^{l',|\hat{E}|})$ is non-negative, we conclude that $\Gamma^{l',|\hat{E}|-1}$ must converge to $\Gamma^{l',|\hat{E}|}$ as $l' \to \infty$. Since $\Gamma^{l',|\hat{E}|}$ converges to Γ^{CW}, $\Gamma^{l',|\hat{E}|-1}$ must also converge to Γ^{CW}. Repeating a similar argument, we conclude that $\Gamma^{l',j}$ converges to Γ^{CW} for all $j = 1, 2, \ldots, |\hat{E}|$.

If the change from $\Gamma^{l',|\hat{E}|-1}$ to $\Gamma^{l',|\hat{E}|}$ is on an off-diagonal entry, say Γ_{uv} with $u \neq v$, then, after some algebra,

$$f\left(\Gamma^{l',|\hat{E}|-1}\right) - f\left(\Gamma^{l',|\hat{E}|}\right) = \ell\left(\Gamma^{l',|\hat{E}|-1}\right) - \ell\left(\Gamma^{l',|\hat{E}|}\right) = \left(\Gamma_{uv}^{l',|\hat{E}|-1} - \Gamma_{uv}^{l',|\hat{E}|}\right)^2 \hat{\Sigma}_{uu}\,.$$

Again, appealing to Lemma 4.3 as before, we can conclude that $\Gamma^{l',|\hat{E}|-1}$ converges to Γ^{CW} as $l' \to \infty$. Similarly, $\Gamma^{l',j}$ converges to Γ^{CW} for every $j = 1, 2, \ldots, |\hat{E}| - 1$.

Consider $k, l \in L'$ with $k > l$ such that for the j-th coordinate in $\hat{E}$, $f(\Gamma^k) \leq f(\Gamma^{l,j}) \leq f(\tilde{\Gamma}^{l,j})$. Here, $\tilde{\Gamma}^{l,j}$ equals to $\Gamma^{l,j}$ except for the j-th nonzero coordinate in $\hat{E}$. As $k, l \to \infty$, we have, from the above analysis, that there exists a matrix Γ^{CW} such that $\Gamma^k \to \Gamma^{\mathrm{CW}}$ and $\Gamma^{l,j} \to \Gamma^{\mathrm{CW}}$. Thus, Γ^{CW} and $\lim_{l\to\infty} \tilde{\Gamma}^{l,j}$ differ by only one coordinate in the j-th position. We conclude that $f(\Gamma^{\mathrm{CW}}) \leq f(\lim_{l\to\infty} \tilde{\Gamma}^{l,j})$. In other words, Γ^{CW} is coordinate-wise minimum. Furthermore, since the optimization problem $\min_{\mathrm{support}(\Gamma)\subseteq \hat{E}} \ell(\Gamma)$ is strongly convex by Assumption 1, Γ^{CW} is the unique minimizer of this optimization problem.

Part 3.) Consider any subsequence $\{\Gamma^k\}_{k\in K}$ such that $\mathrm{support}(\Gamma^k) = \hat{E}$. We will show by contradiction that $\{\Gamma^k\}_{k\in K}$ must converge to Γ^{CW}. Suppose $\{\Gamma^k\}_{k\in K}$ has a limit point $\hat{\Gamma} \neq \Gamma^{\mathrm{CW}}$. Then there exist a subsequence $\{\Gamma^{k'}\}_{k'\in K'}$, with $K' \subseteq K$, that converges to $\hat{\Gamma}$. Therefore, $\lim_{k'\to\infty} f(\Gamma^{k'}) = \ell(\hat{\Gamma}) + \lambda^2|\hat{E}|$. From part 1 and part 2, we have that $\lim_{l'\to\infty} f(\Gamma^{l'}) = \ell(\Gamma^{\mathrm{CW}}) + \lambda^2|\hat{E}|$. By Lemma 4.3, we have $\lim_{k'\to\infty} f(\Gamma^{k'}) = \lim_{l'\to\infty} f(\Gamma^{l'})$. Thus, we conclude that $\ell(\hat{\Gamma}) = \ell(\Gamma^{\mathrm{CW}})$, which contradicts the fact that Γ^{CW} is the unique minimizer of $\min_{\mathrm{support}(\Gamma)\subseteq \hat{E}} \ell(\Gamma)$. Therefore, we conclude that any subsequence with support $\hat{E}$ converges to Γ^{CW} as $k \to \infty$. ∎

LEMMA 4.6. *Let* Γ *be a limit point of* $\{\Gamma^k\}_{k=1}^{\infty}$ *with* $\mathrm{support}(\Gamma) = \hat{E}$. *Then we have* $\mathrm{support}(\Gamma^k) = \hat{E}$ *for infinitely many* k*'s.*

Proof. We prove this result by contradiction. Assume that there are only finitely many k's such that $\mathrm{support}(\Gamma^k) = \hat{E}$. Since there are finitely many possible support sets, there is a support $E' \neq \hat{E}$ and a subsequence $\{\Gamma^{k'}\}$ of $\{\Gamma^k\}$ such that $\mathrm{support}(\Gamma^{k'}) = E'$ for all k', and $\lim_{k'\to\infty} \Gamma^{k'} = \Gamma$. However, by Lemma 4.5, the subsequence converges to a minimizer Γ^{CW} with $\mathrm{support}(\Gamma^{\mathrm{CW}}) = E'$ and thus $\Gamma^{\mathrm{CW}} \neq \Gamma$. This is a contradiction. ∎

We are now ready to complete the proof of Theorem 4.1.

Proof of Theorem 4.1. Let Γ be a limit point of $\{\Gamma^k\}$ with the largest support size and denote its support by $\hat{E}$. By Lemma 4.6, there is a subsequence $\{\Gamma^r\}_{r\in R}$ of $\{\Gamma^k\}$ such that $\mathrm{support}(\Gamma^r) = \hat{E}, \forall r \in R$, and $\lim_{r\to\infty} \Gamma^r = \Gamma$. By Lemma 4.5, there exists an integer M such that for every $r \geq M$ and $r + 1$ is a spacer step, we have $\mathrm{support}(\Gamma^r) = \mathrm{support}(\Gamma^{r+1})$. Without loss of generality, we choose the subsequence that $r > M, \forall r \in R$. We will demonstrate by contradiction that any coordinate (u, v) in $\hat{E}$ cannot be dropped infinitely often in $\{\Gamma^k\}$. To this end, assume that $(u, v) \notin \{\mathrm{support}(\Gamma^r)\}_{r>M}$ infinitely often. Let $\{\Gamma^{r'}\}_{r'\in R'}$, where $R' \subseteq R$, be the subsequence with $\mathrm{support}(\Gamma^{r'+1}) = \hat{E} \setminus \{(u, v)\}, \forall r' \in R'$. Since $r' > M$ and the support has been changed, $r' + 1$ is not a spacer step. Therefore, using Proposition 3.1, we have $f(\Gamma^{r'}) - f(\Gamma^{r'+1}) \geq \lambda^2 - A_{uv}^2/4\hat{\Sigma}_{uu} > 0$. By Lemma 4.3, we have $\lim_{r'\to\infty} f(\Gamma^{r'}) - f(\Gamma^{r'+1}) = 0$. Thus, $\lambda^2 = A_{uv}^2/4\hat{\Sigma}_{uu}$, where $A_{uv} = \sum_{j\neq u} \Gamma_{jv}^{r'} \hat{\Sigma}_{ju} + \sum_{k\neq u} \Gamma_{kv}^{r'} \hat{\Sigma}_{uk}$. By Proposition 3.1, in step $r' + 1$, we have $|\Gamma_{uv}^{r'+1}| = \lambda/\sqrt{\hat{\Sigma}_{uu}} > 0$, which contradicts the definition of $\{\Gamma^{r'}\}_{r'\in R'}$. Therefore, no coordinate in $\hat{E}$ can be dropped infinitely often. Moreover, no coordinate can be added to $\hat{E}$ infinitely often as $\hat{E}$ is the largest support. As a result, the support converges to $\hat{E}$. With stabilized support $\hat{E}$, by Lemma 4.5, we have that $\{\Gamma^k\}$ converges to the limit Γ^{CW} with support $\hat{E}$. From Algorithm 3.1 and Proposition 3.1, we have Γ_{uv} is a minimizer of $f(\Gamma_{uv})$ with respect to the coordinate (u, v) and others fixed. Therefore, Γ^{CW} is the CW minimum. ∎

Our analysis for optimality guarantees requires an assumption on the population model. For the set $E \subseteq \{(i, j) : i, j = 1, 2 \ldots, m\}$, consider the optimization problem

$$\Gamma^\star_E = \underset{\Gamma\in\mathbb{R}^{m\times m}}{\arg\min} \sum_{i=1}^{m} -2\log(\Gamma_{ii}) + \operatorname{tr}(\Gamma\Gamma^\intercal\Sigma^\star) \quad \text{s.t.} \quad \text{support}(\Gamma) \subseteq E. \tag{4.1}$$

ASSUMPTION 2. *There exists constants $\bar{\kappa}, \underline{\kappa} > 0$ such that $\sigma_{min}(\Gamma^\star_E) \geq \underline{\kappa}$ and $\sigma_{max}(\Gamma^\star_E) \leq \bar{\kappa}$ for every E where the graph (V, E) is a DAG, where $\sigma_{min}(\cdot)$ and $\sigma_{min}(\cdot)$ are the smallest and largest eigenvalues respectively.*

We further define $d_{\max} := \max_i |\{j : (j, i) \in E_{\text{super}}\}|$.

THEOREM 4.7. *Let $\hat{\Gamma}, \hat{\Gamma}^{\text{opt}}$ be the solution of Algorithm 3.1 and an optimal solution of* (2.3)*, respectively. Suppose Assumption 2 holds and let the regularization parameter be chosen so that $\lambda^2 = \mathcal{O}(\log m/n)$ where m and n denote the number of nodes and number of samples, respectively. Then,*

1. *$f(\hat{\Gamma}) - f(\hat{\Gamma}^{\text{opt}}) \to_P 0$ as $n \to \infty$,*
2. *if $n/\log(n) \geq \mathcal{O}(m^2 \log m)$, with probability greater than $1-1/\mathcal{O}(n)$, we have that: $0 \leq f(\hat{\Gamma}) - f(\hat{\Gamma}^{\text{opt}}) \leq \mathcal{O}(\sqrt{d^2_{max} m^4 \log m/n})$.*

In other words, the objective value of the coordinate descent solution converges in probability to the optimal objective value as $n \to \infty$. Further, assuming the sample size n is sufficiently large, with high probability, the difference in objective value is bounded by $\mathcal{O}(\sqrt{d^2_{max} m^4 \log m/n})$.

Our proof relies on the following lemmas. Throughout, we let $\hat{E}$ be the support of $\hat{\Gamma}$, i.e., $\hat{E} = \{(i, j), \hat{\Gamma}_{ij} \neq 0\}$.

LEMMA 4.8. *Let $\hat{\Gamma}, \hat{\Gamma}^{\text{opt}}$ be the solution of Algorithm 3.1 and optimal solution of* (2.3)*, respectively. Then, i) for any $u, v = 1, 2, \ldots, m$, $A_{uv} + 2\Gamma_{uv}\hat{\Sigma}_{uu} = 2(\hat{\Sigma}\Gamma)_{uv}$ where A_{uv} is defined in Proposition 3.1. ii) if $\hat{\Gamma}_{uv} \neq 0$, then $(\hat{\Sigma}\hat{\Gamma})_{uv} = 0$, and iii) the matrix $\hat{\Gamma}\hat{\Gamma}^\intercal\hat{\Sigma}$ has ones on the diagonal.*

Proof. For $u, v = 1, \ldots, m$, by the definition of A_{uv}, $A_{uv} + 2\Gamma_{uv}\hat{\Sigma}_{uu} = 2(\hat{\Sigma}\Gamma)_{uv}$, proving item i. Since any solution from Algorithm 3.1, $\hat{\Gamma}$ satisfies Proposition 3.1, for any $(u, v) \in \hat{E}$, $(4\hat{\Sigma}_{uu}\hat{\Gamma}_{uu} + A_{uu})^2 = A^2_{uu} + 16\hat{\Sigma}_{uu}$ and $A_{uv} = -2\hat{\Gamma}_{uv}\hat{\Sigma}_{uu}$. Combining the previous relations, we conclude that $(\hat{\Sigma}\hat{\Gamma})_{uv} = 0$. Therefore, for any $(u, v) \in \hat{E}$, we have $\hat{\Gamma}_{uv} \neq 0$ and $(\hat{\Sigma}\hat{\Gamma})_{uv} = 0$, resulting in $\hat{\Gamma}_{uv}(\hat{\Sigma}\hat{\Gamma})_{uv} = 0$. This proves item ii. Plugging A_{uu} into the previous relations, we arrive at $\hat{\Gamma}_{uu}(\hat{\Sigma}\hat{\Gamma})_{uu} = 1$. Thus, $(\hat{\Gamma}\hat{\Gamma}^\intercal\hat{\Sigma})_{ii} = \sum_{j=1}^{m} \hat{\Gamma}_{ij}(\hat{\Gamma}^\intercal\hat{\Sigma})_{ji} = \hat{\Gamma}_{ii}(\hat{\Gamma}^\intercal\hat{\Sigma})_{ii} = 1$, proving item iii. ∎

LEMMA 4.9. *Let $E \subseteq \{(i, j) : i, j = 1, 2, \ldots, m\}$ be any set where the graph indexed by tuple (V, E) is a DAG. Consider the estimator:*

$$\hat{\Gamma}_E = \underset{\Gamma\in\mathbb{R}^{m\times m}}{\arg\min} \sum_{i=1}^{m} -2\log(\Gamma_{ii}) + \operatorname{tr}\left(\Gamma\Gamma^\intercal\hat{\Sigma}\right) \quad s.t. \quad \text{support}(\Gamma) \subseteq E. \tag{4.2}$$

Suppose that $4m\bar{\kappa}\|\hat{\Sigma}-\Sigma^\star\|_2 \le \min\{8\bar{\kappa}^3/m\bar{\kappa}^2, 1/2m\underline{\kappa}\}$ *and that* $\hat{\Sigma}$ *is positive definite. Then,* $\|\hat{\Gamma}_E - \Gamma^\star_E\|_F \le 4m\bar{\kappa}\|\hat{\Sigma}-\Sigma^\star\|_2$.

Proof. The proof follows from standard convex analysis and Brouwer's fixed point theorem; we provide the details below. Since Γ follows a DAG structure, the objective of (4.2) can be written as: $-2\log\det(\Gamma) + \|\Gamma\hat{\Sigma}^{1/2}\|_F^2$. The KKT conditions state that there exists Q with support$(Q)\cap E = \emptyset$ such that the optimal solution $\hat{\Gamma}_E$ of (4.2) satisfies $-2\hat{\Gamma}_E^{-1} + Q + 2\hat{\Gamma}_E\hat{\Sigma} = 0$ and support$(\hat{\Gamma}_E) \subseteq E$. Let $\Delta = \hat{\Gamma}_E - \Gamma^\star_E$. By Taylor series expansion, $\hat{\Gamma}_E^{-1} = (\Gamma^\star_E + \Delta)^{-1} = {\Gamma^\star_E}^{-1} + {\Gamma^\star_E}^{-T}\Delta{\Gamma^\star_E}^{-1} + \mathcal{R}(\Delta)$, where $\mathcal{R}(\Delta) = 2{\Gamma^\star_E}^{-1}\sum_{k=2}^{\infty}(-\Delta\Gamma^\star_E)^k$. For any matrix $M \in \mathbb{R}^{m\times m}$, define the operator $\mathbb{I}^\star$ with $\mathbb{I}^\star(M) := 2{\Gamma^\star_E}^{-T}M{\Gamma^\star_E}^{-1} + 2M\Sigma^\star$. Let $\mathcal{K}$ be the subspace $\mathcal{K} = \{M \in \mathbb{R}^{m\times m}$: support$(M) \subseteq E\}$ and let $P_{\mathcal{K}}$ be the projection operator onto subspace $\mathcal{K}$ that zeros out entries of the input matrix outside of the support set E. From the optimality condition of (4.1), we have $\mathcal{P}_{\mathcal{K}}[2{\Gamma^\star_E}^{-1} - 2\Gamma^\star_E\Sigma^\star] = 0$. Then, the optimality condition of (4.2) can be rewritten as:

$$\mathcal{P}_{\mathcal{K}}\left[\mathbb{I}^\star(\Delta) + 2\Delta(\hat{\Sigma}-\Sigma^\star) + \mathcal{R}(\Delta) + H_n\right] = 0. \tag{4.3}$$

Since $\hat{\Gamma}_E \in \mathcal{K}$ and $\Gamma^\star_E \in \mathcal{K}$, we have that $\Delta \in \mathcal{K}$. We use Brouwer's theorem to obtain a bound on $\|\Delta\|_F$. We define an operator J as $\mathcal{K} \to \mathcal{K}$:

$$J(\delta) = \delta - (\mathcal{P}_{\mathcal{K}}\mathbb{I}^\star\mathcal{P}_{\mathcal{K}})^{-1}\left(\mathcal{P}_{\mathcal{K}}\left[\mathbb{I}^\star\mathcal{P}_{\mathcal{K}}(\delta) + \mathcal{R}(\delta) + H_n + 2\delta(\hat{\Sigma}-\Sigma^\star)\right]\right).$$

Here, the operator $\mathcal{P}_{\mathcal{K}}\mathbb{I}^\star\mathcal{P}_{\mathcal{K}}$ is invertible since $\sigma_{\min}(\mathbb{I}^\star) = \sigma_{\min}({\Gamma^\star_E}^{-1})^2 \ge \frac{1}{\underline{\kappa}^2}$. Notice that any fixed point δ of J satisfies the optimality condition (4.3). Furthermore, since the objective of (4.2) is strictly convex, we have that the fixed point must be unique. In other words, the unique fixed point of J is given by Δ. Now consider the following compact set: $\mathcal{B}_r = \{\delta \in \mathbb{R}^{m\times m}$: support$(\delta) \subseteq E, \|\delta\|_F \le r\}$ for $r = 4m\bar{\kappa}\|\hat{\Sigma}-\Sigma^\star\|_2$. By the assumption, $r \le \min\{8\bar{\kappa}^3/m\underline{\kappa}^2, \frac{1}{2\bar{\kappa}}\}$. Then, for every $\delta \in \mathcal{B}_r$, we have that: $\|\delta\Gamma^\star_S\|_F \le m\bar{\kappa}r \le 1/2$ and additionally, $\|\mathcal{R}(\delta)\|_F \le 2m\|\Gamma^\star_E\|_2^2/\sigma_{\min}(\Gamma^\star_E)\|\delta\|_2^2\frac{1}{1-\|\delta\Gamma^\star_E\|_2} \le 2m\bar{\kappa}_2^2/\underline{\kappa}r^2\frac{1}{1-r\bar{\kappa}} \le 4m\bar{\kappa}_2^2/\underline{\kappa}r^2$. Since $\|H_n\|_F \le 2m\|\Gamma^\star_E\|_2\|\hat{\Sigma}-\Sigma^\star\|_2$ and $\|G(\delta)\|_F \le \frac{1}{\underline{\kappa}^2}[\|H_n\|_F + \|\mathcal{R}(\delta)\|_F + 2\|\delta(\hat{\Sigma}-\Sigma^\star)\|_F]$ we conclude that $\|J(\delta)\|_F \le \frac{4m\bar{\kappa}^2r^2}{\underline{\kappa}^3} + \frac{4m\max\{\bar{\kappa},1\}}{\underline{\kappa}^2}\|\hat{\Sigma}-\Sigma^\star\|_2 \le r$. In other words, we have shown that J maps $\mathcal{B}_r$ onto itself. Appealing to Brouwer's fixed point theorem, we conclude that the fixed point must also lie inside $\mathcal{B}_r$. Thus, we conclude that $\|\Delta\|_F \le r$. ∎

LEMMA 4.10. *With probability greater than* $1-1/\mathcal{O}(n)$*, we have that:* $\|\hat{\Sigma}-\Sigma^\star\|_2 \le \mathcal{O}(\sqrt{m\log(n)/n})$, $\|\hat{\Sigma}\|_\infty \le 2\bar{\kappa}^2$, $\sigma_{min}(\hat{\Sigma}) \ge \underline{\kappa}^2/2$, $\|\hat{\Gamma}\|_\infty \le 2\bar{\kappa}$ *and* $\sigma_{min}(\hat{\Gamma}) \ge \underline{\kappa}/2$.

Proof. From standard Gaussian concentration results that when $n/\log(n) \ge \mathcal{O}(m)$, with probability greater than $1-\mathcal{O}(1/n)$, we have that $\|\hat{\Sigma}-\Sigma^\star\|_2 \le \mathcal{O}(\sqrt{m\log(n)/n})$. By Assumption 2, with probability greater than $1-\mathcal{O}(1/n)$, $\hat{\Sigma}$ is positive definite, with $\|\hat{\Sigma}\|_\infty \le 2\bar{\kappa}^2$ and $\sigma_{\min}(\hat{\Sigma}) \ge \underline{\kappa}^2 - \mathcal{O}(\sqrt{m\log(n)/n}) \ge \underline{\kappa}^2/2$. Furthermore, appealing to Lemma 4.9 and that $n/\log(n) \ge \mathcal{O}(m^3)$, $\|\hat{\Gamma} - \Gamma^\star_{\hat{E}}\|_F \le \mathcal{O}(\sqrt{m^3\log(n)/n})$. Thus $\|\hat{\Gamma}\|_\infty \le \|\Gamma^\star_{\hat{E}}\|_2 + \bar{\kappa} \le 2\bar{\kappa}$ and $\sigma_{\min}(\hat{\Gamma}) \ge \bar{\kappa} - \mathcal{O}(\sqrt{m\log(n)/n}) \ge \underline{\kappa}/2$. ∎

Proof of Theorem 4.7. **Part 1)**. First,

$$0 \le f(\hat{\Gamma}) - f(\hat{\Gamma}^{\text{opt}}) \le f(\hat{\Gamma}) - \log\det(\hat{\Sigma}) - m = -\log\det(\hat{\Gamma}\hat{\Gamma}^{\mathrm{T}}\hat{\Sigma}) + \lambda^2\|\hat{\Gamma} - \text{diag}(\hat{\Gamma})\|_0,$$

where the second inequality follows from $f(\hat{\Gamma}^{\text{opt}}) \geq \min_{\Theta}\{-\log\det(\Theta) + \text{tr}(\Theta\hat{\Sigma})\} = \log\det(\hat{\Sigma})+m$; the equality follows from appealing to item i. of Lemma 4.8 to conclude that $f(\hat{\Gamma}) = -\log\det(\hat{\Gamma}\hat{\Gamma}^{\mathrm{T}}) + m + \lambda^2\|\hat{\Gamma} - \text{diag}(\hat{\Gamma})\|_0$.

Our strategy is to show that as $n \to \infty$, $\hat{\Gamma}\hat{\Gamma}^{\mathrm{T}}\hat{\Sigma}$ converges to a matrix with ones on the diagonal and whose off-diagonal entries induce a DAG. Thus, $\log\det(\hat{\Gamma}\hat{\Gamma}^{\mathrm{T}}\hat{\Sigma}) \to \log\prod_{i=1}^{m} 1 = 0$ as $n \to \infty$. Since $\lambda^2 \to 0$ as $n \to \infty$ and $\|\hat{\Gamma} - \text{diag}(\hat{\Gamma})\|_0 \leq m^2$, we can then conclude the desired result. For any $u, v = 1, 2, \ldots, m$:

$$(\hat{\Gamma}\hat{\Gamma}^{\mathrm{T}}\hat{\Sigma})_{uv} = \sum_{i=1}^{m} \hat{\Gamma}_{ui}(\hat{\Sigma}\hat{\Gamma})_{vi} = \hat{\Gamma}_{uu}(\hat{\Sigma}\hat{\Gamma})_{vu} + \hat{\Gamma}_{uv}(\hat{\Sigma}\hat{\Gamma})_{vv} + \sum_{i\in F_{uv}} \hat{\Gamma}_{ui}(\hat{\Sigma}\hat{\Gamma})_{vi}, \tag{4.4}$$

where $F_{uv} := \{i \mid i \neq u, i \neq v, (u,i) \in \hat{E}, (v,i) \notin \hat{E}\}$. Here, the second equality is due to item ii. of Lemma 4.8; note that if $\hat{\Gamma}_{ui}(\hat{\Sigma}\hat{\Gamma})_{vi} \neq 0$, then $i \in F_{uv}$ as otherwise either $\hat{\Gamma}_{ui} = 0$ or $(\hat{\Sigma}\hat{\Gamma})_{vi} = 0$. We consider the two possible settings for $(u,v), u \neq v$: Setting I) $(u,v) \in \hat{E}$ which implies that $(v,u) \notin \hat{E}$ as $\hat{\Gamma}$ specifies a DAG, and Setting II) $(u,v), (v,u) \notin \hat{E}$. (Note that $(u,v), (v,u) \in \hat{E}$ is not possible since $\hat{\Gamma}$ specifies a DAG.)

Setting I: Since $(u,v) \in \hat{E}$ and $(v,u) \notin \hat{E}$, we have

$$(\hat{\Gamma}\hat{\Gamma}^{\mathrm{T}}\hat{\Sigma})_{vu} = \sum_{i\in F_{vu}} \hat{\Gamma}_{vi}(\hat{\Sigma}\hat{\Gamma})_{ui} = \sum_{i\in F_{vu}} \hat{\Gamma}_{vi}\left(\frac{1}{2}A_{ui} + \hat{\Gamma}_{ui}\hat{\Sigma}_{uu}\right) = \sum_{i\in F_{vu}} \frac{1}{2}\hat{\Gamma}_{vi}A_{ui}.$$

Here, the first equality follows from appealing to (4.4), and noting that $\hat{\Gamma}_{vu} = 0$ and that $(\hat{\Sigma}\hat{\Gamma})_{uv} = 0$ according to item ii. of Lemma 4.8; the second equality follows from item iii. of Lemma 4.8; the final equality follows from noting that $\hat{\Gamma}_{ui} = 0$ for $i \in F_{vu}$.

For each $i \in F_{vu}$, Figure 1 (left) represents the relationships between the nodes u, v, i. Here, the directed edge from u to v from the constraint $(u,v) \in \hat{E}$ is represented by a split line, the directed edge from v to i from the constraint $i \in F_{vu}$ is represented by a solid line, and the directed edge that is disallowed due to the constraint $i \in F_{vu}$ is represented via a cross-out solid line.

Since there is a directed path from u to i, to avoid a cycle, a directed path from i to u cannot exist. Thus, adding the edge from u to i to $\hat{E}$ does not violate acyclicity and the fact that it is missing is due to $\lambda^2 > A_{ui}^2/(4\hat{\Sigma}_{uu})$ according to Proposition 3.1. Then, appealing to Lemma 4.10, we conclude that with probability greater than $1 - \mathcal{O}(1/n)$: $|(\hat{\Gamma}\hat{\Gamma}^{\mathrm{T}}\hat{\Sigma})_{vu}| \leq \sum_{i\in F_{vu}} \frac{1}{2}|\hat{\Gamma}_{vi}| 2\lambda(\hat{\Sigma}_{u,u})^{1/2} \leq 4\lambda\bar{\kappa}d_{\max}$. In other words, in this setting, $|(\hat{\Gamma}\hat{\Gamma}^{\mathrm{T}}\hat{\Sigma})_{vu}| \to 0$ as $n \to \infty$.

Setting II: Since $(u,v), (v,u) \notin \hat{E}$, we have

$$(\hat{\Gamma}\hat{\Gamma}^{\mathrm{T}}\hat{\Sigma})_{uv} = \hat{\Gamma}_{uu}\left(\frac{1}{2}A_{vu} + \hat{\Gamma}_{vu}\hat{\Sigma}_{vv}\right) + \sum_{i\in F_{uv}} \hat{\Gamma}_{ui}\left(\frac{1}{2}A_{vi} + \hat{\Gamma}_{vi}\hat{\Sigma}_{vv}\right)$$

$$= \sum_{\substack{i\in F_{uv}\\ \cup\{u\}}} \frac{\hat{\Gamma}_{ui}A_{vi}}{2}. \tag{4.5}$$

Here, the first equality follows from plugging zero for $\hat{\Gamma}_{uv}$ in (4.4) and appealing to item i. of Lemma 4.8; the second equality follows from plugging in zero for $\hat{\Gamma}_{vi}$ and $\hat{\Gamma}_{vu}$. Since $\hat{\Gamma}$ specifies a DAG, there cannot simultaneously be a directed path from u

to v and from v to u. Thus, either directed edges (u, v) or (v, u) can be added without creating a cycle. We consider the three remaining sub-cases below:

Setting II.1. Adding (u, v) to $\hat{E}$ violates acyclicity but adding (v, u) does not.

For each $i \in F_{uv}$, Figure 1 (middle) represents the relations between nodes u, v, and i. Here, due to the condition of Setting II, nodes u and v are not connected by an edge, so this is displayed by a solid crossed-out undirected edge. Furthermore, the directed edge from u to i from the constraint $i \in F_{uv}$ is represented via a solid directed edge, the directed edge v to i that is disallowed due to the constraint $i \in F_{uv}$ is represented via a cross-out solid line. Finally, the directed edge u to v that is disallowed due to acyclicity is represented via a crossed-out dashed line.

Since adding the directed edge (u, v) to $\hat{E}$ creates a cycle, then we have the following implications: i. adding (v, u) to $\hat{E}$ does not violate acyclicity (as both edges $u \to v$ and $v \to u$ cannot simultaneously create cycles) and ii. there must be a directed path from v to u. Implication i. allows us to conclude that $\hat{\Gamma}_{vu}$ must be equal to zero due to the condition $4\hat{\Sigma}_{vv}\lambda^2 > A_{vu}^2$ from Proposition 3.1. Combining implication ii. and the fact that there is a directed edge from u to i in $\hat{E}$ allows us to conclude that there cannot be a directed path from i to v as we would be creating a direct path from u to itself. Thus, the fact that the directed edge (v, i) is not in $\hat{E}$, or equivalently that $\hat{\Gamma}_{vi} = 0$, is due to $4\hat{\Sigma}_{vv}\lambda^2 > A_{vi}^2$ according to Proposition 3.1. From (4.5) and Lemma 4.10, we conclude with probability greater than $1 - \mathcal{O}(1/n)$, $|(\hat{\Gamma}\hat{\Gamma}^{\mathsf{T}}\hat{\Sigma})_{uv}| \leq 4\bar{\kappa}\lambda(1 + d_{\max})$. In other words, in this setting, $|(\hat{\Gamma}\hat{\Gamma}^{\mathsf{T}}\hat{\Sigma})_{uv}| \to 0$ as $n \to \infty$.

Setting II.2. Adding (u, v) or (v, u) to $\hat{E}$ would not violate acyclicity.

For each $i \in F_{uv}$, Figure 1 (right) represents the relations between the nodes u, v, and i . Here, due to the condition of Setting II, nodes u and v are not connected by an edge, so this is displayed by a solid crossed-out undirected edge. Furthermore, the directed edge from u to i from the constraint $i \in F_{uv}$ is represented via a solid directed edge, the directed edge v to i that is disallowed due to the constraint $i \in F_{uv}$ is represented via a cross-out solid line.

In this setting, recall that the directed edges u to v and v to u are not present in the estimate $\hat{E}$. Since neither of these two edges violates acyclicity according to the condition of this setting, we conclude that $4\hat{\Sigma}_{vv}\lambda^2 > A_{vu}^2$. There cannot be a path from i to v because then there would exist a path from u to v, which contradicts the scenario that an edge from v to u does not create a cycle. As a result, an edge from v to i does not create a cycle and $\hat{\Gamma}_{vi} = 0$ is due to $4\hat{\Sigma}_{vv}\lambda^2 > A_{vi}^2$ according to Proposition 3.1. Thus, from (4.5) and Lemma 4.10, we conclude that, with probability greater than $1-\mathcal{O}(1/n)$, $|(\hat{\Gamma}\hat{\Gamma}^{\mathsf{T}}\hat{\Sigma})_{uv}| \leq 4\bar{\kappa}\lambda(1+d_{\max})$. In other words, $|(\hat{\Gamma}\hat{\Gamma}^{\mathsf{T}}\hat{\Sigma})_{uv}| \to 0$ as $n \to \infty$.

Setting II.3. Adding (v, u) violates acyclicity but adding (u, v) does not.

In this case, even if $(\hat{\Gamma}\hat{\Gamma}^{\mathsf{T}}\hat{\Sigma})_{uv}$ does not converge to zero, we have by the setting assumption that adding (u, v) to $\hat{E}$ does not violate DAG constraint. Since $\hat{E}$ specifies a DAG, the off-diagonal nonzero entries of the matrix $\hat{\Gamma}\hat{\Gamma}^{\mathsf{T}}\hat{\Sigma}$ specifies a DAG as well.

Putting Settings I–II together, we have shown that as $n \to \infty$, the nonzero entries in the off-diagonal of $\hat{\Gamma}\hat{\Gamma}^{\mathsf{T}}\hat{\Sigma}$ specify a DAG. Furthermore, according to item i. of Lemma 4.8, the diagonal entries of this matrix are equal to one. As stated earlier, this then allows us to conclude that $-\log\det(\hat{\Gamma}\hat{\Gamma}^{\mathsf{T}}\hat{\Sigma}) \to 0$ as $n \to \infty$, and consequently that $f(\hat{\Gamma}) - f(\hat{\Gamma}^{\mathrm{opt}}) \to 0$.

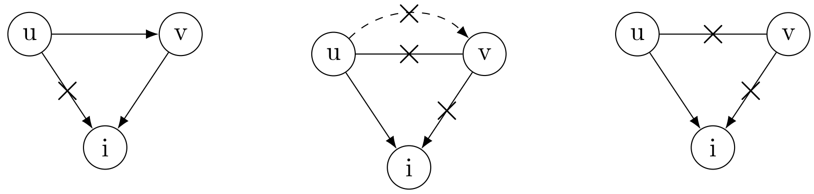


Fig. 1: Left: scenario for Setting I, middle: scenario for setting II.1, and right: scenario for setting II.2; solid directed edges represent directed edges that are assumed to be in the estimate $\hat{E}$, crossed out solid directed edges represent directed edges that are assumed to be excluded in the estimate $\hat{E}$, crossed out solid undirected edges indicate that the corresponding nodes are not connected in $\hat{E}$, and crossed out dotted directed edge indicates that the edge is not present in $\hat{E}$ as adding it would create a cycle.

Part 2) Using the proof of Theorem 4.7 part i), we can immediately conclude that the matrix $\hat{\Gamma}\hat{\Gamma}^{\mathsf{T}}\hat{\Sigma}_n$ can be decomposed as the sum $N + \Delta$. Here, the off-diagonal entries of N specify a DAG, with ones on the diagonal and under the assumption on n, with probability greater than $1 - \mathcal{O}(1/n)$, $\|\Delta\|_\infty \leq 4\bar{\kappa}(1 + d_{\max})\lambda$ with zeros on the diagonal of Δ. Consequently, $\|\Delta\|_\infty \leq 4m\bar{\kappa}^2(1 + d_{\max})\lambda$. Furthermore, by Lemma 4.10, we get $\sigma_{\min}(\hat{\Gamma}\hat{\Gamma}^{\mathsf{T}}\hat{\Sigma}_n) \geq \sigma_{\min}(\hat{\Gamma})^2\sigma_{\min}(\hat{\Sigma}) \geq \underline{\kappa}^4/4$. The reverse triangle inequality yields $\sigma_{\min}(N) \geq \underline{\kappa}^4/4 - 4m\bar{\kappa}^2(1 + d_{\max})\lambda$. Consider any matrix $\bar{N}$ with $|\bar{N}_{ij} - N_{ij}| \leq |\Delta_{ij}|$. Using the reverse triangle inequality again, we get $\sigma_{\min}(\bar{N}) \geq \underline{\kappa}^4 - 8m\bar{\kappa}^2(1 + d_{\max})\lambda$ with probability greater than $1 - \mathcal{O}(1/n)$. By the assumption on the sample size, $\bar{N}$ is invertible, and so we can use first-order Taylor series expansion to obtain $-\log\det(N + \Delta) = -\log\det(N) - \mathrm{tr}(\bar{N}^{-1}\Delta)$. Since $\log\det(N) = 0$, we obtain the bound $-\log\det(N + \Delta) \leq -\mathrm{tr}(\bar{N}^{-1}\Delta) \leq \|\bar{N}^{-1}\|_2\|\Delta\|_\star$ with $\|\cdot\|_\star$ denoting the nuclear norm. Thus, $-\log\det(N + \Delta) \leq \|\bar{N}^{-1}\|_2\|\Delta\|_\star \leq \frac{m}{\sigma_{\min}(\bar{N})}\|\Delta\|_2 \leq \frac{4m^2\bar{\kappa}^2(1+d_{\max})\lambda}{\underline{\kappa}^4/4 - 8m\bar{\kappa}^2(1+d_{\max})\lambda}$. As $\lambda = \mathcal{O}(\log m/n)$, by the assumption on the sample size, $f(\hat{\Gamma}) - f(\hat{\Gamma}^{\mathrm{opt}}) \leq \mathcal{O}(\sqrt{d_{\max}^2 m^4 \log m/n})$. ∎

4.2. Statistical consistency guarantees. Recall from Section 2.1 that there is typically multiple SEMs that are compatible with the distributions $\mathcal{P}^\star$. Each equivalent SEM is specified by a DAG; this DAG defines a total ordering among the variables. Associated to each ordering π is a unique structural equation model that is compatible with the distribution $\mathcal{P}^\star$. We denote the set of parameters of this model as $(\tilde{B}^\star(\pi), \tilde{\Omega}^\star(\pi))$. For the tuple $(\tilde{B}^\star(\pi), \tilde{\Omega}^\star(\pi))$, we define $\tilde{\Gamma}^\star(\pi) := (I - \tilde{B}^\star(\pi))\tilde{\Omega}^\star(\pi)^{-1/2}$. We let $\Pi = \{\text{ordering } \pi : \text{support}(\tilde{B}^\star(\pi)) \subseteq E_{\text{super}}\}$. Throughout, we will use the notation $s^\star = \|B^\star\|_{\ell_0}$ and $\tilde{s} := \tilde{s}^\star(\pi) = \|\tilde{B}^\star(\pi)\|_{\ell_0}$.

Assumption 3. *(Sparsity of every equivalent causal model) There exists some constant* $\tilde{\alpha}$ *such that for any* $\pi \in \Pi$, $\|\tilde{B}^\star_{:j}(\pi)\|_{\ell_0} \leq \tilde{\alpha}\sqrt{n}/\log m$.

Assumption 4. *(Beta-min condition) There exist constants* $0 \leq \eta_1 < 1$ *and* $0 < \eta_0^2 < 1 - \eta_1$, *such that for any* $\pi \in \Pi$, *the matrix* $\tilde{B}^\star(\pi)$ *has at least* $(1 - \eta_1)\|\tilde{B}^\star(\pi)\|_{\ell_0}$ *coordinates* $k \neq j$ *with* $|\tilde{B}^\star_{kj}(\pi)| > \sqrt{\log m/n}(\sqrt{m/s^\star} \vee 1)/\eta_0$.

Assumption 5. *(Sufficiently large noise variances) For every* $\pi \in \Pi$, $\mathcal{O}(1) \geq \min_j[\tilde{\Omega}^\star(\pi)]_{jj} \geq \mathcal{O}(\sqrt{s^\star \log m/n})$.

Assumption 6. *(Sufficiently sparse* $B^\star$ *and super-structure* E_{super}*) For every* $i = 1, 2, \ldots, m$, $\|B^\star_{:,i}\|_{\ell_0} \leq \alpha n/\log(m)$ *and* $|\{j, (j,i) \in E_{super}\}| \leq \alpha n/\log(m)$.

Here, Assumptions 3-4 are similar to those in [23]. Assumption 5 is used to characterize the behavior of the early stopped estimate and is thus new relative to [23]. Assumption 6 ensures that the number of parents for every node both in the true DAG and the super-structure is not too large.

Next, we present our theorem on the finite-sample consistency guarantees of the coordinate descent algorithm. Throughout, we assume that we have obtained a solution after the algorithm has converged. We let GAP denote the difference between the objective value of the coordinate descent output and the optimal objective value of (2.3). We let $\hat{\Gamma}$ be a minimizer of (2.3).

THEOREM 4.11. *Let $\hat{\Gamma}, \hat{\Gamma}^{\text{opt}}$ be the solution of Algorithm 3.1 and the optimal solution of* (2.3)*, respectively. Suppose Assumptions 2-6 are satisfied with constants $\alpha, \tilde{\alpha}, \eta_0$ sufficiently small. Let $\alpha_0 := \min\{4/m, 0.05\}$. Then, for $\lambda^2 \asymp \log m/n$, if $n/\log(n) \geq \mathcal{O}(m^2 \log m)$, with probability greater than $1 - 2\alpha_0$, there exists a π such that*

1. $\|\hat{\Gamma} - \hat{\Gamma}^{\text{opt}}\|_F^2 \leq \mathcal{O}(\sqrt{d_{\max}^2 m^4 \log m/n})$,
2. $\|\hat{\Gamma} - \tilde{\Gamma}^{\star}(\pi)\|_F^2 = \mathcal{O}(\sqrt{d_{\max}^2 m^4 \log m/n})$*, and* $\|\tilde{\Gamma}^{\star}(\pi)\|_{\ell_0} \asymp s^{\star}$.

The proof relies on the following results.

PROPOSITION 4.12. *(Theorem 3.1 of [23]) Suppose Assumptions 2–6 hold with constants $\alpha, \tilde{\alpha}, \eta_0$ sufficiently small. Let $\hat{\Gamma}^{\text{opt}}$ be any optimum of* (2.3) *with the constraint that* $\text{support}(\Gamma) \subseteq E_{super}$*. Let π^{opt} be the associated ordering of $\hat{\Gamma}^{\text{opt}}$ and $(\hat{B}^{\text{opt}}, \hat{\Omega}^{\text{opt}})$ be the associated connectivity and noise variance matrix satisfying $\hat{\Gamma}^{\text{opt}} = (I - \hat{B}^{\text{opt}})\hat{\mathcal{K}}^{\text{opt}^{-1/2}}$. Then, for $\alpha_0 := (4/m) \wedge 0.05$ and $\lambda^2 \asymp \log m/n$, we have, with a probability greater than $1 - \alpha_0$, $\|\hat{B}^{\text{opt}} - \tilde{B}^{\star}(\pi)\|_F^2 + \|\hat{\Omega}^{\text{opt}} - \tilde{\Omega}^{\star}(\pi^{\text{opt}})\|_F^2 = \mathcal{O}(\lambda^2 s^{\star})$, and $\|\tilde{B}^{\star}(\pi)\|_{\ell_0} \asymp s^{\star}$.*

COROLLARY 4.13 (Corollary 6 of [25]). *With the setup in Proposition 4.12,*

$$\left\|\hat{\Gamma}^{\text{opt}} - \tilde{\Gamma}^{\star}(\pi)\right\|_F^2 \leq \frac{16 \max\{1, \|\tilde{B}^{\star}(\pi)\|_F^2, \|\tilde{\Omega}^{\star}(\pi)^{-1/2}\|_F^2\}\lambda^2 s^{\star}}{\min\{1, \min_j(\tilde{\Omega}^{\star}(\pi)_{jj})^3\}}.$$

Proof of Theorem 4.11. The proof is similar to that of [25] and we provide a short description for completeness. For notational simplicity, we let $\Gamma^{\star} := \tilde{\Gamma}^{\star}(\pi)$ where π is the permutation satisfying Proposition 4.12 and $\tilde{\Gamma}^{\star}$ defined earlier. From Theorem 4.7, we have that $0 \leq f(\hat{\Gamma}) - f(\hat{\Gamma}^{\text{opt}}) \leq \mathcal{O}(\sqrt{d_{\max}^2 m^4 \log m/n})$. Let GAP $= \mathcal{O}(\sqrt{d_{\max}^2 m^4 \log m/n})$. For a matrix $\Gamma \in \mathbb{R}^{m\times m}$, let $\ell(\Gamma) := \sum_{i=1}^m -2\log(\Gamma_{ii}) + \text{tr}(\Gamma\Gamma^{\mathrm{T}}\hat{\Sigma}_n)$. Suppose that $\|\hat{\Gamma}\|_{\ell_0} \geq \|\hat{\Gamma}^{\text{opt}}\|_{\ell_0}$. Then, $\ell(\hat{\Gamma}) - \ell(\hat{\Gamma}^{\text{opt}}) \leq$ GAP. On the other hand, suppose $\|\hat{\Gamma}\|_{\ell_0} \leq \|\hat{\Gamma}^{\text{opt}}\|_{\ell_0}$. Then, $\ell(\hat{\Gamma}) - \ell(\hat{\Gamma}^{\text{opt}}) \leq \text{GAP} + \lambda^2\|\hat{\Gamma}\|_{\ell_0} \leq$ 2GAP. So, we conclude the bound $\ell(\hat{\Gamma}) - \ell(\hat{\Gamma}^{\text{opt}}) \leq$ 2GAP.

For notational simplicity, we will consider a vectorized objective. Let $T \subseteq \{1, \ldots, m^2\}$ be indices corresponding to diagonal elements of an $m \times m$ matrix being vectorized. With abuse of notation, let $\hat{\Gamma}$, $\hat{\Gamma}^{\text{opt}}$, and $\Gamma^{\star}$ be the vectorized form of their corresponding matrices. Then, Taylor series expansion yields

$$\begin{aligned}\ell(\hat{\Gamma}) - \ell(\hat{\Gamma}^{\text{opt}}) = (\Gamma^{\star} - \hat{\Gamma}^{\text{opt}})^{\mathrm{T}}\nabla^2\ell(\bar{\Gamma})(\hat{\Gamma} - \hat{\Gamma}^{\text{opt}}) + \nabla\ell(\Gamma^{\star})^{\mathrm{T}}(\hat{\Gamma} - \hat{\Gamma}^{\text{opt}})\\ + 1/2(\hat{\Gamma} - \hat{\Gamma}^{\text{opt}})^{\mathrm{T}}\nabla^2\ell(\tilde{\Gamma})(\hat{\Gamma} - \hat{\Gamma}^{\text{opt}}).\end{aligned}$$

Here, entries of $\tilde{\Gamma}$ lie between $\hat{\Gamma}$ and $\hat{\Gamma}^{\text{opt}}$, and entries of $\bar{\Gamma}$ lie between $\hat{\Gamma}^{\text{opt}}$ and $\Gamma^{\star}$.

Some algebra then gives:

$$1/2(\hat{\Gamma} - \hat{\Gamma}^{\text{opt}})^{\mathsf{T}}\nabla^2\ell(\tilde{\Gamma})(\hat{\Gamma} - \hat{\Gamma}^{\text{opt}}) \leq [\ell(\hat{\Gamma}) - \ell(\hat{\Gamma}^{\text{opt}})] + \|\nabla\ell(\Gamma^\star)\|_{\ell_2}\|\hat{\Gamma} - \hat{\Gamma}^{\text{opt}}\|_{\ell_2} + \|\hat{\Gamma} - \hat{\Gamma}^{\text{opt}}\|_{\ell_2}\|\hat{\Gamma}^{\text{opt}} - \Gamma^\star\|_{\ell_2}\kappa_{\max}(\nabla^2\ell(\bar{\Gamma})).$$

By the convexity of $\ell(\cdot)$, for any Γ, $\nabla^2\ell(\Gamma) \succeq \hat{\Sigma} \otimes I$. Thus appealing to Lemma 4.10, with probability greater than $1 - \mathcal{O}(1/n)$, $\sigma_{\min}(\nabla^2\ell(\Gamma)) \geq \underline{\kappa}^2/2$. Letting $\tau := 4(\|\hat{\Gamma}^{\text{opt}} - \Gamma^\star\|_{\ell_2}\kappa_{\max}(\nabla^2\ell(\bar{\Gamma})) + \|\nabla\ell(\Gamma^\star)\|_{\ell_2})/\underline{\kappa}^2$, with probability greater than $1 - \mathcal{O}(1/n)$: $\|\hat{\Gamma} - \hat{\Gamma}^{\text{opt}}\|_{\ell_2}^2 \leq 4\underline{\kappa}^{-2}\ell(\hat{\Gamma}) - \ell(\hat{\Gamma}^{\text{opt}}) + 4\tau\underline{\kappa}^{-2}\|\hat{\Gamma} - \hat{\Gamma}^{\text{opt}}\|_{\ell_2}\tau$. Note that for non-negative Z, W, Π, the inequality $Z^2 \leq \Pi Z + W$ implies $Z \leq (\Pi + \sqrt{\Pi^2 + 4W})/2$. Using this fact, in conjunction with the previous bound, we obtain with probability greater than $1-\mathcal{O}(1/n)$ the bound $\|\hat{\Gamma}-\hat{\Gamma}^{\text{opt}}\|_{\ell_2} \leq \frac{\tau}{2}+\frac{1}{2}(\tau^2+16\underline{\kappa}^{-2}[\ell(\hat{\Gamma})-\ell(\hat{\Gamma}^{\text{opt}})])^{1/2}$. We next bound τ. From Corollary 4.13, we have control over the term $\|\hat{\Gamma}^{\text{opt}} - \Gamma^\star\|_{\ell_2}$ in τ. It remains to control $\sigma_{\max}(\nabla^2\ell(\bar{\Gamma}))$ and $\|\nabla\ell(\Gamma^\star)\|_{\ell_2}$. Let $\Gamma \in \mathbb{R}^{m^2}$. Suppose that for every $j \in T$, $\Gamma_j \geq \nu$. Then, some calculations yield the bound $\nabla^2\ell(\Gamma) \preceq \hat{\Sigma} \otimes I + \frac{2}{\nu^2}I_{m^2} = \hat{\Sigma} \otimes I + \frac{2}{\nu^2}I_{m^2}$. We have that for every $j \in T$, $\hat{\Gamma}^{\text{opt}}_j \geq \Gamma^\star_j - \|\hat{\Gamma}^{\text{opt}} - \Gamma^\star\|_{\ell_2}$. From Corollary 4.13, Assumption 5, and that $\lambda\sqrt{s^\star} \leq 1$, we then have $\hat{\Gamma}^{\text{opt}}_j \geq \Gamma^\star_j/2 \geq 1/2(\Omega^\star_j)^{-1/2}$. Since the entries of $\bar{\Gamma}$ are between those of $\Gamma^\star$ and $\hat{\Gamma}^{\text{opt}}$ and by Lemma 4.10, $\sigma_{\max}(\nabla^2\ell(\bar{\Gamma})) \leq \sigma_{\max}(\hat{\Sigma}) + 8\min_j \Omega^\star_j = \mathcal{O}(1)$. To control $\nabla\ell(\Gamma^\star)$, we first note that $\mathbb{E}[\nabla\ell(\Gamma^\star)] = 0$. Therefore, $\|\nabla\ell(\Gamma^\star)\|_{\ell_2} = \|\nabla\ell(\Gamma^\star) - \mathbb{E}[\nabla\ell(\Gamma^\star)]\|_{\ell_2}$. Since $\nabla\ell(\Gamma^\star) - \mathbb{E}[\nabla\ell(\Gamma^\star)] = ((\hat{\Sigma} - \Sigma^\star) \otimes I)\Gamma^\star$, letting $K^\star = (\Sigma^\star)^{-1}$ we get $\|\nabla\ell(\Gamma^\star) - \mathbb{E}[\nabla\ell(\Gamma^\star)]\|_{\ell_2}^2 = \text{tr}((\hat{\Sigma}_n - \Sigma^\star)(\hat{\Sigma}_n - \Sigma^\star)^{\mathsf{T}}K^\star) \leq \|\hat{\Sigma} - \Sigma^\star\|_2^2\|K^\star\|_\star \leq m\|\hat{\Sigma} - \Sigma^\star\|_2^2\|K^\star\|_2 \leq \mathcal{O}(m^2\log(n)/n)$. Thus, $\|\nabla\ell(\Gamma^\star) - \mathbb{E}[\nabla\ell(\Gamma^\star)]\|_{\ell_2} \leq \mathcal{O}(m\sqrt{\log n}/\sqrt{n})$. Upper bounding τ and then ultimately using that to upper-bound $\|\hat{\Gamma} - \hat{\Gamma}^{\text{opt}}\|_{\ell_2}$, we conclude that $\|\hat{\Gamma} - \hat{\Gamma}^{\text{opt}}\|_{\ell_2}^2 \leq \mathcal{O}(\sqrt{d_{\max}^2 m^4 \log m/n})$. Combining this bound with Proposition 4.12, we get the first result of the theorem. The second result follows straightforwardly from triangle inequality: $\|\hat{\Gamma} - \Gamma^\star\|_F^2 \leq 2\|\hat{\Gamma} - \hat{\Gamma}^{\text{opt}}\|_F^2 + 2\|\hat{\Gamma}^{\text{opt}} - \Gamma^\star\|_F^2 \leq \mathcal{O}(\sqrt{d_{\max}^2 m^4 \log m/n})$. ∎

The result of Theorem 4.7 guarantees that the estimate from our coordinate descent procedure is close to the optimal solution of (2.3), and that it accurately estimates certain reordering of the population model. For accurately estimating the edges of the population Markov equivalence class MEC($\mathcal{G}^\star$), we need the faithfulness condition and a strictly stronger version of the beta-min condition[23], dubbed the strong beta-min condition.

Assumption 7. *(Faithfulness)The DAG $\mathcal{G}^\star$ is faithful with respect to the data generating distribution $\mathcal{P}^\star$, that is, every conditional independence relationship entailed in $\mathcal{P}^\star$ is encoded $\mathcal{G}^\star$.*

Assumption 8. *(Strong beta-min condition) There exist constant $0 < \eta_0^2 < 1/s^\star$, such that for any $\pi \in \Pi$, the matrix $\tilde{B}^\star(\pi)$ has all of its nonzero coordinates (k, j) satisfy $|\tilde{B}^\star_{kj}(\pi)| > \sqrt{s^\star \log m/n}/\eta_0$.*

Theorem 4.14. *Suppose $\lambda^2 \asymp s^\star \log m/n$, the sample size satisfies $n/\log(n) \geq \mathcal{O}(m^2 \log m)$, and assumptions of Theorem 4.11 hold, with Assumption 4 replaced by Assumption 8. Then, with probability greater than $1 - 2\alpha_0$, there exists a member of the population Markov equivalence class with associated parameter $\Gamma^\star_{\text{mec}}$ such that $\|\hat{\Gamma} - \Gamma^\star_{\text{mec}}\|_F^2 \leq \mathcal{O}(\sqrt{d_{max}^2 m^4 \log m/n})$.*

Appealing to Remark 3.2 of van de Geer and Bühlmann [23], under assumptions of Theorem 4.11, as well as Assumption 8, the graph encoded by any optimal connectivity matrix $\hat{B}^{\text{opt}}$ of this optimization problem encodes, with probability $1-\alpha_0$, a member of the Markov equivalence class of the population directed acyclic graph. Let $(B^\star_{\text{mec}}, \Omega^\star_{\text{mec}})$ be the associated connectivity matrix and noise matrix of this population model. Furthermore, define $\Gamma^\star_{\text{mec}} = (I - B^\star_{\text{mec}}){\Omega^\star_{\text{mec}}}^{-1/2}$. The proof of the theorem relies on the following lemma in [25].

LEMMA 4.15 (Lemma 7 of [25]). *Under the conditions of Theorem 4.14, we have with probability greater than* $1-2\alpha_0$, $\|\hat{\Gamma}^{\text{opt}} - \Gamma^\star_{\text{mec}}\|_F^2 = \mathcal{O}(m^2/n)$.

Proof of Theorem 4.14. First, by Lemma 4.15, with probability greater than $1-2\alpha_0$, $\|\hat{\Gamma}-\Gamma^\star_{\text{mec}}\|_F^2 \le 2\|\hat{\Gamma}-\hat{\Gamma}^{\text{opt}}\|_F^2+2\|\hat{\Gamma}^{\text{opt}}-\Gamma^\star_{\text{mec}}\|_F^2 \le \text{GAP}+\mathcal{O}(m^2/n)$. Since the GAP is on the order $\mathcal{O}(\sqrt{d_{\max}^2 m^4 \log m/n})$, we get $\|\hat{\Gamma} - \Gamma^\star_{\text{mec}}\|_F^2 \le \mathcal{O}(\sqrt{d_{\max}^2 m^4 \log m/n})$.□

We remark that without the faithfulness condition (see Assumption 7), we can guarantee that the estimate from our coordinate descent procedure is close to a member of what is known as the *minimal-edge I-MAP*. The minimal-edge I-MAP is the sparsest set of directed acyclic graphs that induce a structural equation model compatible with the true data distribution. Under faithfulness, the minimal-edge I-MAP coincides with the population Markov equivalence class [23].

5. Experiments. In this section, we illustrate the utility of our method on synthetic and real data and compare its performance with competing methods. We dub our method CD-ℓ_0 as it is a coordinate descent method using ℓ_0 penalized loss function. The competing methods we compare against include Greedy equivalence search (GES) [5], Greedy Sparsest Permutation (GSP) [19], and the mixed-integer convex program (MICODAG) [25]. We also compare our method with other coordinate descent algorithms (CCDr-MCP) [1, 2, 9], which use a minimax concave penalty instead of ℓ_0 norm and are implemented as an `R` package *sparsebn*. All experiments are performed with a MacBook Air (M2 chip) with 8GB of RAM and a 256GB SSD, using `Gurobi` 10.0.0 as the optimization solver.

As the input super-structure E_{super}, we supply an estimated moral graph, computed using the graphical lasso procedure [8]. To make our comparisons fair, we appropriately modify the competing methods so that E_{super} can also be supplied as input. Note that we count the number of support after each update in Algorithm 3.1. Converting the graph into a string key at each iteration is inefficient. Therefore, in the implementation, we count the support only after each full loop, setting the threshold to C instead of Cm^2. Throughout this paper, C is set to 5.

We use the metric d_{cpdag} to evaluate the estimation accuracy as the underlying DAG is generally identifiable up to the Markov equivalence class. The metric d_{cpdag} is the number of different entries between the unweighted adjacency matrices of the estimated completed partially directed acyclic graph (CPDAG) and the true CPDAG. A CPDAG has a directed edge from a node i to a node j if and only if this directed edge is present in every DAG in the associated Markov equivalence class, and it has an undirected edge between nodes i and j if the corresponding Markov equivalence class contains DAGs with both directed edges from i to j and from j to i.

The time limit for the integer programming method MICODAG is set to $50m$. If the algorithm does not terminate within the time limit, we report the solution time (in seconds) and the achieved relative optimality gap, computed as RGAP = (upper bound – lower bound)/lower bound. Here, the upper bound and lower bound refer to the objective value associated with the best feasible solution and best lower bound, obtained respectively by MICODAG. A zero value for RGAP indicates that an optimal solution has been found.

Unless stated otherwise, we use the Bayesian information criterion (BIC) to choose the parameter λ. In our context, the BIC score is given by $-2n\sum_{i=1}^{m}\log(\hat{\Gamma}_{ii}) + n\mathrm{tr}(\hat{\Gamma}\hat{\Gamma}^{\mathrm{T}}\hat{\Sigma}) + k\log(n)$, where k is the number of nonzero entries in the estimated parameter $\hat{\Gamma}$. From theoretical guarantees in [25], λ^2 should be on the order $\log(m)/n$. Hence, we choose λ with the smallest BIC score among $\lambda^2 = c\log m/n$, for $c = 1, 2, \ldots, 15$.

Setup of synthetic experiments: For all the synthetic experiments, once we specify a DAG, we generate data according to the SEM (2.1), where the nonzero entries of $B^\star$ are drawn uniformly at random from the set $\{-0.8, -0.6, 0.6, 0.8\}$ and diagonal entries of $\Omega^\star$ are chosen uniformly at random from the set $\{0.5, 1, 1.5\}$.

5.1. Comparison with benchmarks. We first generate datasets from twelve publicly available networks sourced from [14] and the Bayesian Network Repository (bnlearn). These networks have different numbers of nodes, ranging from $m = 6$ to $m = 70$. We generate 10 independently and identically distributed datasets for each network according to the SEM described earlier with sample size $n = 500$.

Table 1 compares the performance of our method CD-ℓ_0 with the competing ones. First, consider small graphs ($m \leq 20$) for which the integer programming approach MICODAG achieves an optimal or near-optimal solution with a small RGAP. As expected, in terms of the accuracy of the estimated model, MICODAG tends to exhibit the best performance. For these small graphs, CD-ℓ_0 performs similarly to MICODAG but attains the solutions much faster. Next, consider moderately sized graphs ($m > 20$). In this case, MICODAG cannot solve these problem instances within the time limit and hence finds inaccurate models, whereas CD-ℓ_0 obtains much more accurate models much faster. Finally, CD-ℓ_0 outperforms GES, GSP, and CCDr-MCP in most problem instances. The improved performance of CD-ℓ_0 over CCDr-MCP highlights the advantage of using ℓ_0 penalization over a minimax concave penalty: ℓ_0 penalization ensures that DAGs in the same Markov equivalence class have the same score, while the same property does not hold with other penalties.

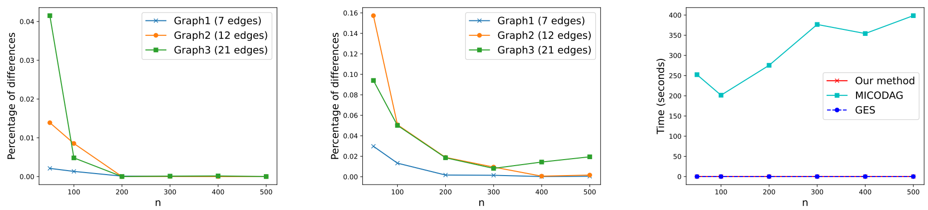
Large graphs: We next demonstrate the scalability of our coordinate descent algorithm for learning large DAGs with over 100 nodes. We consider networks from the Bayesian Network Repository and generate 10 independent datasets similar to the previous experiment. Table 2 presents the results where we see that our method CD-

Table 1: Comparison of our method, CD-ℓ_0, with competing methods

	MICODAG			CCDr-MCP		GES		GSP		CD-ℓ_0	
Network(m)	Time	RGAP	d_{cpdag}	Time	d_{cpdag}	Time	d_{cpdag}	Time	d_{cpdag}	Time	d_{cpdag}
Dsep(6)	≤ 1	0	2.0($\pm$0)	≤ 1	2.0($\pm$0)	≤ 1	1.8($\pm$0.6)	≤ 1	2.0($\pm$0)	≤ 1	2.0($\pm$0)
Asia(8)	≤ 1	0	2.2($\pm$0.6)	≤ 1	2.0($\pm$0)	≤ 1	2.7($\pm$0.9)	≤ 1	4.9($\pm$1.4)	≤ 1	2.0($\pm$0)
Bowling(9)	3	0	2.0($\pm$0)	≤ 1	4.7($\pm$2.4)	≤ 1	2.4($\pm$0.7)	≤ 1	5.6($\pm$2.5)	≤ 1	2.2($\pm$0.4)
InsSmall(15)	≥ 750	.080	7.0($\pm$2.6)	≤ 1	29.9($\pm$4.0)	≤ 1	24.9($\pm$10.3)	≤ 1	17.2($\pm$7.9)	≤ 1	8.0($\pm$0)
Rain(14)	151	0	2.0($\pm$0)	≤ 1	9.5($\pm$2.0)	≤ 1	5.4($\pm$3.7)	≤ 1	17.5($\pm$4.3)	≤ 1	3.3($\pm$2.1)
Cloud(16)	93	0	5.2($\pm$0.6)	≤ 1	11.0($\pm$4.1)	≤ 1	5.0($\pm$1.5)	≤ 1	13.7($\pm$3.0)	≤ 1	6.8($\pm$2.3)
Funnel(18)	70	0	2.0($\pm$0)	≤ 1	2.0($\pm$0)	≤ 1	4.8($\pm$6.5)	≤ 1	13.0($\pm$2.9)	≤ 1	2.0($\pm$0)
Galaxy(20)	237	0	1.0($\pm$0)	≤ 1	4.6($\pm$3.1)	≤ 1	1.5($\pm$1.6)	≤ 1	15.8($\pm$5.2)	≤ 1	1.0($\pm$0)
Insurance(27)	≥ 1350	.340	22.8($\pm$13.5)	≤ 1	38.4($\pm$4.8)	≤ 1	30.5($\pm$14.8)	≤ 1	38.5($\pm$6.7)	≤ 1	14.7($\pm$4.1)
Factors(27)	≥ 1350	.311	56.1($\pm$8.4)	≤ 1	65.3($\pm$7.6)	≤ 1	68.9($\pm$10.5)	≤ 1	52.3($\pm$7.4)	≤ 1	18.1($\pm$6.7)
Hailfinder(56)	≥ 2800	.245	41.4($\pm$12.6)	≤ 1	12.9($\pm$3.5)	≤ 1	26.4($\pm$16.2)	≤ 1	109.1($\pm$10.2)	1.6	2.6($\pm$1.3)
Hepar2(70)	≥ 3500	5.415	76.9($\pm$16.5)	≤ 1	54.6($\pm$12.0)	≤ 1	71.5($\pm$27.4)	≤ 1	66.3($\pm$9.3)	11.4	5.3($\pm$2.2)

Here, MICODAG, mixed-integer convex program [25]; CCDr-MCP, minimax concave penalized estimator with coordinate descent [2]; GES, greedy equivalence search algorithm [5]; GSP, greedy sparsest permutation algorithm [19]; d_{cpdag}, differences between the true and estimated completed partially directed acyclic graphs; RGAP, relative optimality gap. All results are computed over ten independent trials where the average d_{cpdag} values are presented with their standard deviations.

Fig. 2: Convergence of CD-ℓ_0 to an optimal solution



Left: normalized difference, as a function of sample size n, between the optimal objective value of (2.3) found using the integer programming approach MICODAG and the objective value obtained by CD-ℓ_0 for three different graphs; Middle: normalized difference of objectives of solutions obtained from MICODAG and GES; Right: comparison of computational cost of CD-ℓ_0, MICODAG, and GES for the DAG with 21 edges. All results are computed and averaged over ten independent trials.

ℓ_0 can effectively scale to large graphs and obtain better or comparable performance to competing methods, as measured by the d_{cpdag} metric.

5.2. Convergence of CD-ℓ_0 solution to an optimal solution. Theorem 4.7 states that as the sample size tends to infinity, CD-ℓ_0 identifies an optimally scoring model. To see how fast the asymptotic kicks in, we generate three synthetic DAGs with $m = 10$ nodes where the total number of edges is chosen from the set $\{7, 12, 21\}$. We obtain 10 independently and identically distributed datasets according to the SEM described earlier with sample size $n = \{50, 100, 200, 300, 400, 500\}$. In Figure 2(left, middle), we compute the normalized difference $(\text{obj}^{\text{method}} - \text{obj}^{\text{opt}})/\text{obj}^{\text{opt}}$ as a function of n for the three graphs, averaged across the ten independent trials. Here, $\text{obj}^{\text{method}}$ is the objective value obtained by the corresponding method (CD-ℓ_0 or GES), while obj^{opt} is the optimal objective obtained by the integer programming approach MICODAG. For moderately large sample sizes (e.g., $n = 200$), CD-ℓ_0 attains the optimal objective value, whereas GES does not. In Figure 2 (right), for the graph with 21 arcs, we see that CD-ℓ_0 can achieve the same accuracy while being computationally much faster to solve.

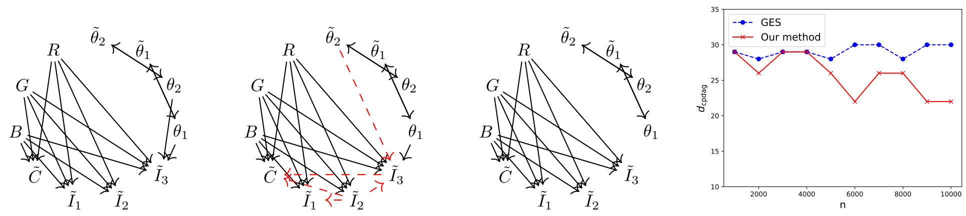
5.3. Real data from causal chambers. Recently, [10] constructed two devices, referred to as causal chambers, allowing us to quickly and inexpensively produce large datasets from non-trivial but well-understood real physical systems. The ground-truth DAG underlying this system is known and shown in Figure 3(a). We collect $n = 1000$ to $n = 10000$ observational samples of $m = 20$ variables at increments of 1000. To maintain clarity, we only plot a subset of the variables in Figure

Table 2: Comparison of our method, CD-ℓ_0, with competing methods for large graphs

	CCDr-MCP		GES		GSP		CD-ℓ_0	
Network(m)	Time	d_{cpdag}	Time	d_{cpdag}	Time	d_{cpdag}	Time	d_{cpdag}
Pathfinder(109)	≤ 1	212.9(±20.7)	≤ 1	275.6(±16.4)	2.0	212.5(±19.5)	11.8	81.6(±16.3)
Andes(223)	1.8	117.9(±9.6)	≤ 1	165.0(±28.3)	6.6	702.0(±42.6)	35.1	107.3(±5.9)
Diabetes(413)	10.4	276.7(±9.7)	3.3	387.1(±22.2)	57.8	1399.8(±19.1)	881.9	286.6(±15.9)

See Table 1 for the description of the methods. All results are computed over ten independent trials where the average d_{cpdag} values are presented with their standard deviations.

Fig. 3: Learning causal models from causal chambers data in [10]



(a) Ground truth (b) GES (c) CD-ℓ_0 (d) d_{cpdag} comparison

Here, a. ground-truth DAG described in [10], b-c. the estimated CPDAGs by GES and CD-ℓ_0 for sample size $n = 10000$, d. comparing the accuracy of the CPDAGs estimated by our method CD-ℓ_0 and GES with different sample sizes n; here the accuracy is computed relative to CPDAG of the ground-truth DAG and uses the metric d_{cpdag}.

3(a, b, c). However, the analysis includes all variables. With this data, we obtain estimates for the Markov equivalence class of the ground-truth DAG using GES and our method CD-ℓ_0 and measure the accuracy of the estimates using the d_{cpdag} metric.

Figures 3(b-c) show the estimated CPDAG for each approach when $n = 10000$. Both methods do not pick up edges between the polarizer angles θ_1, θ_2 and other variables. As mentioned in [10], this phenomenon is likely due to these effects being nonlinear. Figure 3(d) compares the accuracy of CD-ℓ_0 and GES in estimating the Markov equivalence class of the ground-truth DAG. For all sample sizes n, we observe that CD-ℓ_0 is more accurate.

6. Discussion. In this paper, we propose the first coordinate descent procedure with proven optimality and statistical guarantees in the context of learning Bayesian networks. Numerical experiments demonstrate that our coordinate descent method is scalable and provides high-quality solutions.

We showed in Theorem 4.1 that our coordinate descent algorithm converges. It would be of interest to characterize the speed of convergence. In addition, the computational complexity of our algorithm may be improved by updating blocks of variables instead of one coordinate at a time. Finally, an open question is whether, in the context of our statistical guarantees in Theorem 4.7, the sample size requirement can be relaxed.

References.

[1] Bryon Aragam and Qing Zhou. Concave penalized estimation of sparse Gaussian Bayesian networks. *Journal of Machine Learning Research*, 16(1):2273–2328, 2015.

[2] Bryon Aragam, Jiaying Gu, and Qing Zhou. Learning large-scale Bayesian networks with the sparsebn package. *Journal of Statistical Software*, 91:1–38, 2019.

[3] Heinz H. Bauschke and Patrick L. Combettes. Convex analysis and monotone operator theory in Hilbert spaces. In *CMS Books in Mathematics*, 2011.

[4] Dimitri Bertsekas. *Nonlinear Programming*, volume 4. Athena Scientific, 2016.

[5] David Maxwell Chickering. Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3(Nov):507–554, 2002.

[6] Byron Ellis and Wing Hung Wong. Learning causal Bayesian network structures from experimental data. *Journal of the American Statistical Association*, 103

(482):778–789, 2008.
[7] Jerome Friedman, Trevor Hastie, Holger Höfling, and Robert Tibshirani. Pathwise coordinate optimization. *Annals of Applied Statistics*, 1(2):302 – 332, 2007.
[8] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.
[9] Fei Fu and Qing Zhou. Learning sparse causal Gaussian networks with experimental intervention: Regularization and coordinate descent. *Journal of the American Statistical Association*, 108(501):288–300, 2013.
[10] Juan L. Gamella, Jonas Peters, and Peter Bühlmann. The causal chambers: Real physical systems as a testbed for AI methodology, 2024. URL https://arxiv.org/abs/2404.11341.
[11] Hussein Hazimeh and Rahul Mazumder. Fast best subset selection: Coordinate descent and local combinatorial optimization algorithms. *Operations Research*, 68(5):1517–1537, 2020.
[12] Markus Kalisch and Peter Bühlmann. Estimating high-dimensional directed acyclic graphs with the PC-algorithm. *Journal of Machine Learning Research*, 8 (22):613–636, 2007.
[13] Simge Küçükyavuz, Ali Shojaie, Hasan Manzour, Linchuan Wei, and Hao-Hsiang Wu. Consistent second-order conic integer programming for learning Bayesian networks. *Journal of Machine Learning Research*, 24(322):1–38, 2023.
[14] Hasan Manzour, Simge Küçükyavuz, Hao-Hsiang Wu, and Ali Shojaie. Integer programming for learning directed acyclic graphs from continuous data. *INFORMS Journal on Optimization*, 3(1):46–73, 2021.
[15] Nicolai Meinshausen and Peter Buhlmann. High-dimensional graphs and variable selection with the lasso. *Annals of Statistics*, 34:1436–1462, 2006.
[16] Preetam Nandy, Alain Hauser, and Marloes Maathuis. High-dimensional consistency in score-based and hybrid structure learning. *Annals of Statistics*, 46(6A): 3151–3183, 2018.
[17] Rajen Dinesh Shah and J. Peters. The hardness of conditional independence testing and the generalised covariance measure. *Annals of Statistics*, 2018.
[18] Tomi Silander and Petri Myllymäki. A simple approach for finding the globally optimal Bayesian network structure. In *Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence*, UAI'06, page 445–452, Arlington, Virginia, USA, 2006. AUAI Press. ISBN 0974903922.
[19] Liam Solus, Yuhao Wang, and Caroline Uhler. Consistency guarantees for greedy permutation-based causal inference algorithms. *Biometrika*, 108(4):795–814, 2021.
[20] Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, Prediction, and Search*. The MIT Press, 1993. ISBN 978-1-4612-7650-0.
[21] Ioannis Tsamardinos, Laura E. Brown, and Constantin F. Aliferis. The max-min hill-climbing Bayesian network structure learning algorithm. *Machine Learning*, 65:31–78, 2006.
[22] Caroline Uhler, Garvesh Raskutti, Peter Buhlmann, and Bin Yu. Geometry of the faithfulness assumption in causal inference. *Annals of Statistics*, 41:436–463, 2012.
[23] Sara van de Geer and Peter Bühlmann. ℓ_0-penalized maximum likelihood for sparse directed acyclic graphs. *Annals of Statistics*, 41(2):536 – 567, 2013.
[24] Thomas Verma and Judea Pearl. Equivalence and synthesis of causal models. In *Uncertainty in Artificial Intelligence*, pages 255–270, 1990.
[25] Tong Xu, Armeen Taeb, Simge Küçükyavuz, and Ali Shojaie. Integer program-

ming for learning directed acyclic graphs from non-identifiable Gaussian models. arXiv.2404.12592, 2024.

[26] Qiaoling Ye, Arash A Amini, and Qing Zhou. Optimizing regularized Cholesky score for order-based learning of Bayesian networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3555–3572, 2020.

[27] Yue Yu, Jie Chen, Tian Gao, and Mo Yu. DAG-GNN: DAG structure learning with graph neural networks. In *International Conference on Machine Learning*, pages 7154–7163. PMLR, 2019.

[28] Xun Zheng, Bryon Aragam, Pradeep K. Ravikumar, and Eric P Xing. DAGs with NO TEARS: continuous optimization for structure learning. *Advances in Neural Information Processing Systems*, 31:9492–9503, 2018.