"Scalable Intervention Target Estimation in Linear Models" Supplementary Material

A Theoretical analysis

Proof of Lemma 1. This lemma is proved in [14]. We provide an alternative and simplified proof for completeness, and with an approach that fits our subsequent analysis. The first observation is that the noise variance of a terminal node j is the inverse of the corresponding diagonal entry of the precision matrix obtained by removing all descendants of j. Thus, if j has no descendants in a set S, then $\sigma_{S,j}^{-2} = [\Theta_S]_{j,j}$. The second observation is that the noise variance of a node in a restricted SEM is affected by only its ancestors. Therefore, the noise variance of a node j in a restricted SEM over S is equal to that of over a set $S \cap \mathsf{an}(j)$, i.e., $\sigma_{S,j} = \sigma_{S\cap\mathsf{an}(j),j}$. Due to the second observation, we will consider only the restricted SEMs over the sets of the form $S = \mathsf{an}(j) \setminus U_j$, where U_j denotes the ancestors of j that lie out of this restricted SEM. Let us denote the precision matrix of the restricted SEM over $\mathsf{an}(j)$ by $\Phi \triangleq \Theta_{\mathsf{an}(j)}$. We obtain the variance of the noise term for a node $j \in S$ as follows:

$$\Phi_S = \Phi_{S,S} - \Phi_{S,U_j}(\Phi_{U_j,U_j})^{-1}\Phi_{U_j,S} \;, \tag{12}$$

$$\frac{1}{\sigma_{S,j}^2} = \Phi_{j,j} = \frac{1}{\sigma_j^2} - \Phi_{j,U_j}(\Phi_{U_j,U_j})^{-1}\Phi_{U_j,j} \tag{13}$$

$$= \frac{1}{\sigma_j^2} - \frac{B_{U_j,j}^\top [\Theta_{\mathsf{an}(j)}]_{U_j,U_j}^{-1} B_{U_j,j}}{\sigma_j^4} \;, \tag{14}$$

where the first line is due to Schur's complement; the second line is due to the observations mentioned above since j is a terminal node in both sets S and $\mathsf{an}(j)$; and the last line is due to (2) and Proposition 4 of [25].

We use the Markov property to characterize edge weights in a restricted SEM. Conditioned on all of its parents, node j is independent of the remaining nodes. Hence, $[B_{\mathsf{an}(j)}]_{k,j} = B_{k,j}$. We consider the same set $S = \mathsf{an}(j) \setminus U_j$, $\Phi = \Theta_{\mathsf{an}(j)}$ and derive the edge weights as follows:

$$[\Phi_S]_{j,k} = \Phi_{j,k} - \Phi_{j,U_j}(\Phi_{U_j,U_j})^{-1}\Phi_{U_j,k} \tag{15}$$

$$= -\frac{[B_{\mathsf{an}(j)}]_{k,j}}{\sigma_{\mathsf{an}(j),j}^2} + \frac{[B_{\mathsf{an}(j)}]_{U_j,j}^\top}{\sigma_{\mathsf{an}(j),j}^2}[\Theta_{\mathsf{an}(j)}]_{U_j,U_j}^{-1}[\Theta_{\mathsf{an}(j)}]_{U_j,k} \tag{16}$$

$$= -\frac{B_{k,j}}{\sigma_j^2} + \frac{B_{U_j,j}^\top}{\sigma_j^2}[\Theta_{\mathsf{an}(j)}]_{U_j,U_j}^{-1}[\Theta_{\mathsf{an}(j)}]_{U_j,k} \;, \tag{17}$$

$$B_{k,j}^S = \frac{\sigma_{S,j}^2}{\sigma_j^2}(B_{k,j} - B_{U_j,j}^\top [\Theta_{\mathsf{an}(j)}]_{U_j,U_j}^{-1}[\Theta_{\mathsf{an}(j)}]_{U_j,k}) \;, \tag{18}$$

where the last line follows from $[\Phi_S]_{j,k} = -[B_S]_{k,j}/\sigma_{S,j}^2$. Note that this last equality is correct since S contains only the ancestors of j. Similarly, we can write $[\Theta_S]_{j,j} = 1/\sigma_{S,j}^2$ if S contains only the ancestors of j. ■

Proof of Proposition 1. Let us consider the restricted SEM over set S and let $U_j = \mathsf{an}(j) \setminus S$ denote the ancestors of j that are not included in the restricted SEM. Note that the restricted SEM over $\mathsf{an}(j)$ has edge weights $B_{\mathsf{an}(j)} = [B]_{\mathsf{an}(j),\mathsf{an}(j)}$ and noise covariance $\Omega_{\mathsf{an}(j)} = [\Omega]_{\mathsf{an}(j),\mathsf{an}(j)}$.

Therefore, for nodes $u, v \in U_j$, we can use Lemma 1 to obtain,

$$[\Theta_{\mathsf{an}(j)}]_{u,v} = -\frac{[B_{\mathsf{an}(j)}]_{u,v}}{\sigma^2_{\mathsf{an}(j),v}} - \frac{[B_{\mathsf{an}(j)}]_{v,u}}{\sigma^2_{\mathsf{an}(j),u}} + \sum_{l \in \mathsf{an}(j)} \frac{[B_{\mathsf{an}(j)}]_{u,l}[B_{\mathsf{an}(j)}]_{v,l}}{\sigma^2_{\mathsf{an}(j),l}} \tag{19}$$

$$= -\frac{B_{u,v}}{\sigma^2_v} - \frac{B_{v,u}}{\sigma^2_u} + \sum_{l \in \mathsf{an}(j)} \frac{B_{u,l}B_{v,l}}{\sigma^2_l}\,, \tag{20}$$

$$[\Theta_{\mathsf{an}(j)}]_{u,u} = \frac{1}{\sigma^2_{\mathsf{an}(j),u}} + \sum_{l \in \mathsf{an}(j)} \frac{[B_{\mathsf{an}(j)}]^2_{u,l}}{\sigma^2_{\mathsf{an}(j),l}} \tag{21}$$

$$= \frac{1}{\sigma^2_u} + \sum_{l \in \mathsf{an}(j)} \frac{B^2_{u,l}}{\sigma^2_l}\,. \tag{22}$$

Now we will prove the first statement. If S contains $\mathsf{an}_{\mathcal{I}}(j)$ and their parents $\mathsf{pa}(\mathsf{an}_{\mathcal{I}}(j))$, we know that neither u, v nor their children belong to $\mathcal{I}$. Therefore, $[\Theta_{\mathsf{an}(j)}]_{u,v}$ and $[\Theta_{\mathsf{an}(j)}]_{u,u}$ are invariant due to (20) and (22), respectively. Subsequently, we have $[\Delta_{\Theta_{\mathsf{an}(j)}}]_{U_j,U_j} = 0$. Furthermore, since $j \notin \mathcal{I}$, we have that $[\Delta_B]_{k,j} = 0$ for $k \in [p]$. Using the Lemma 1 again, we obtain

$$\sigma^2_{S,j} = \sigma^2_j \left(\sigma^4_j - B^\top_{U_j,j}[\Theta_{\mathsf{an}(j)}]^{-1}_{U_j,U_j} B_{U_j,j}\right)^{-1}, \tag{23}$$

where we note that σ_j, $B_{U_j,j}$, and $[\Theta_{\mathsf{an}(j)}]_{U_j,U_j}$ are all invariant and, subsequently, $\sigma^{(1)}_{S,j} = \sigma^{(2)}_{S,j}$ is invariant. This proves the first statement regarding the invariance of the noise term for a non-intervened node under certain restricted SEMs.

For the last part, Assumption 1 ensures that $\sigma^{(1)}_{S,i} \neq \sigma^{(2)}_{S,i}$ for $i \in \mathcal{I}$, and we have $[\Delta_{\Theta_S}]_{i,i} \neq 0$. Similarly, Assumption 1 states that if $[B_S]_{j,i} \neq 0$ for either model, then $[\Delta_{\Theta_S}]_{j,i} \neq 0$. ■

Proof of Theorem 1. We will follow the steps of the Algorithm 1 to obtain the consistency results. We assume that the covariance estimates are perfect, i.e., they are equal to the population-level statistics. Therefore, we can compute Δ_{Θ_S} for any $S \subseteq [p]$ correctly. Instead of estimating $\mathcal{I}$ directly, we, equivalently, aim to identify its complement $\mathcal{I}^{\mathrm{C}}$.

Forming S_Δ. In Step 1, we first estimate Δ_Θ over $[p]$ to obtain the nodes that are affected by the interventions. Note that $\sigma^{(1)}_i \neq \sigma^{(2)}_i$ for intervened nodes $i \in \mathcal{I}$ and $[\Delta_B]_{k,j} = 0$ for non-intervened nodes $j \notin \mathcal{I}$ and $k \in [p]$. According to (2) and (3), $[\Delta_B]_{k,k} \neq 0$ if and only if either $k \in \mathcal{I}$ or there exists $k \to i$ for which $i \in \mathcal{I}$. In other words, by forming the set $S_\Delta = \{k : k \in [p], [\Delta_\Theta]_{k,k} \neq 0\} = \mathcal{I} \cup \bigcup_{i \in \mathcal{I}} \mathsf{pa}(i)$, we can discard the nodes in $[p] \setminus S_\Delta$. The discarded nodes consist of the non-intervened nodes that do not have children in $\mathcal{I}$. Next, we will show computationally, some of the non-intervened nodes in S_Δ can be identified easier than the others.

Forming non-intervened source nodes J_0. Note that if a node j has an intervened ancestor, the distribution of X_j changes and, subsequently, $\Sigma^{(1)}_{j,j} \neq \Sigma^{(2)}_{j,j}$. If a node i is intervened, the distribution of X_i changes too, and it results in $\Sigma^{(1)}_{i,i} \neq \Sigma^{(2)}_{i,i}$. Therefore, we are able to find non-intervened source nodes directly from $\Sigma^{(1)}$ and $\Sigma^{(2)}$. Since we have already narrowed down our focus to set S_Δ, we define *non-intervened source nodes* as

$$J_0 \triangleq \{j : j \in S_\Delta,\ j \notin \mathcal{I},\ \mathsf{an}_{\mathcal{I}}(j) = \emptyset\} \tag{24}$$

$$= \{j : j \in S_\Delta,\ \Sigma^{(1)}_{j,j} = \Sigma^{(2)}_{j,j}\}\,. \tag{25}$$

Sets S_Δ and J_0 are subsequently fed into the next steps of the algorithm.

Forming source ancestral sets J^k_0. In Proposition 1 we have shown that for any non-intervened node j, there exists sets S that makes $\sigma_{S,j}$ invariant, and the condition is closely related to ancestors of j that are affected by the intervention being included in S. On the other hand, such a restricted SEM does not exist for any intervened node. Therefore, we can identify all the non-intervened nodes in $S_\Delta \setminus J_0$ by finding a proper restricted SEM over a subset of S_Δ. Hence, finding the ancestors of non-intervened nodes is critical. Now consider pair $\{j, k\}$ such that $j \in J_0$, $k \in S_\Delta \setminus J_0$. $\Sigma_{j,j}$ is invariant and $\Sigma_{k,k}$ is changing. If j and k have a common ancestor, which can be j itself, then $\Sigma_{j,k}$ is

nonzero and $[\Delta_{\Theta_{\{j,k\}}}]_{j,k} \neq 0$. Otherwise, $\Sigma_{j,k} = 0$ and $[\Delta_{\Theta_{\{j,k\}}}]_{j,k} = 0$. Subsequently, we define the *source ancestral set* J_0^k for each node $k \in S_\Delta \setminus J_0$, that consists of the nodes in J_0 that have a common ancestor with k, i.e.,

$$\begin{aligned} J_0^k &\triangleq \{j : j \in J_0,\ [\Delta_{\Theta_{\{j,k\}}}]_{j,k} \neq 0\}\ , \quad \forall k \in S_\Delta \setminus J_0 \\ &= \{j : j \in J_0,\ \mathsf{an}(j) \cap \mathsf{an}(k) \neq \emptyset\}\ . \end{aligned} \tag{26}$$

Next, we will use these source ancestral sets to group the nodes that have similar ancestors together.

Forming equivalence classes from J_0. We note that some of the nodes in $S_\Delta \setminus J_0$ will have identical source ancestral sets. Therefore, we can decompose the set $S_\Delta \setminus J_0$ into *equivalence classes* such that all the nodes in a class have the same source ancestral sets. We denote these equivalence classes by $\mathcal{A}_1, \ldots, \mathcal{A}_L$, and the source ancestral set corresponding to the class $\mathcal{A}_\ell$ by $J_0^{\mathcal{A}_\ell}$ for $\ell \in [L]$. Formally,

$$S_\Delta \setminus J_0 = \bigcup_{\ell \in [L]} \mathcal{A}_\ell\ , \tag{27}$$

$$\mathcal{A}_{\ell_1} \cap \mathcal{A}_{\ell_2} = \emptyset\ , \quad \text{for } \ \ell_1 \neq \ell_2\ , \tag{28}$$

$$J_0^{\mathcal{A}_\ell} \triangleq J_0^{k_1} = J_0^{k_2}, \quad \forall k_1, k_2 \in \mathcal{A}_\ell\ , \quad \text{for } \ \ell \in [L]\ . \tag{29}$$

We note that we order these classes according to a topological order such that for $1 \leq \ell < \ell' \leq L$, $J_0^{\mathcal{A}_{\ell'}} \not\subset J_0^{\mathcal{A}_\ell}$. In other words, the class corresponding to the superset of any $J_0^{\mathcal{A}_\ell}$ should appear after $\mathcal{A}_\ell$ in the sequence $\mathcal{A}_1, \ldots, \mathcal{A}_L$. This ordering is important since we do not need descendants of a non-intervened node in a restricted SEM to conclude its invariance. In the next step, we will show how searching for such restricted SEMs for non-intervened nodes is simplified with this decomposition to equivalence classes.

d-separation property for invariance. We establish the connection between d-separation in interventional graphs and the precision differences. Consider the augmented graph characterization of interventions presented in [7]. A new node, F, is introduced to the graph to represent the interventional distribution. There are edges from F to i for any intervened node $i \in \mathcal{I}$ in the augmented graph. As there is no edge between F and non-intervened node j, there exists a set S that d-separates F and j in the augmented graph. This implies that the probability distribution of the node j is invariant given $S \setminus \{j\}$, which in turn implies that both conditional mean and variance of the node j does not change. Subsequently, $\sigma_{S,j}$ is invariant for this set S. Applying the results of [26] and [9], $[\Theta_S]_{j,j} = \sigma_{S,j}^{-2}$ is also invariant. Therefore, the set S that d-separates F and non-intervened j results in $[\Delta_{\Theta_S}]_{j,j} = 0$.

Processing equivalence classes. We process equivalence classes $\mathcal{A}_1, \ldots, \mathcal{A}_L$ individually, i.e., at stage ℓ, we consider the nodes in $\mathcal{A}_\ell$. Let us define $\mathcal{M}_\ell = J_0 \cup \bigcup_{1 \leq b < \ell} \mathcal{A}_b$. We will prove that for a non-intervened node $j \in \mathcal{A}_\ell$, we can determine its invariance via $2^{|\mathcal{A}_\ell|}$ PDE. Due to our ordering of the equivalence classes, any ancestor of j in S_Δ will lie in either $\mathcal{M}_\ell$ or $\mathcal{A}_\ell$. Consider the set $S = \mathcal{M}_\ell \cup \mathsf{an}_\mathcal{I}(j) \cup \mathsf{pa}(\mathsf{an}_\mathcal{I}(j))$ which is also of the form $\mathcal{M}_\ell \cup A$ for some $A \subseteq \mathcal{A}_\ell$. Note that S does not contain any descendant of j.

We will use *d-separation property for invariance* to show that this set S yields $[\Delta_{\Theta_S}]_{j,j} = 0$. Specifically, we will show that there does not exist a d-connecting path between the augmented node F and j. Suppose the contrary and let $\pi : \langle F \to i \ldots j \rangle$ be a d-connecting path where $i \in \mathcal{I}$. If j has a tail end on π, there is a collider node k on the path that is a descendant of j. Since S does not contain any descendant of j, neither node k nor its descendants are in S, and it blocks the path. Therefore, the path should be of the form $\langle F \to i \cdots \to j \rangle$. If i is a collider and not in S, it means it is not an ancestor of S. Therefore, its descendants are also not in S, and i blocks the path. If i is a collider and in S, it is either in $\mathcal{M}_\ell$ or in $\mathsf{an}_\mathcal{I}(j)$. In either case, the parent of i on the path is also in S and it blocks the path. If i is not a collider, the path will be $\langle F \to i \to \cdots \to j \rangle$. If i is in S, it blocks the path. If i is not in S, it is not an ancestor of j. Then, there is a collider k on the path that is a descendant of i. Since i is not in S, none of its descendants are neither in S. Therefore, k blocks the path. We have ruled out all possible active paths and shown that there does not exist a d-connecting path between F and j for $S = \mathcal{M}_\ell \cup \mathsf{an}_\mathcal{I}(j) \cup \mathsf{pa}(\mathsf{an}_\mathcal{I}(j))$. Subsequently, $[\Delta_{\Theta_S}]_{j,j} = 0$ due to d-separation for invariance property. As we have noted before, set S can be written as $S = \mathcal{M}_\ell \cup A$ for some $A \subseteq \mathcal{A}_\ell$, and we can check the existence of such A, i.e., whether j is non-intervened by

using PDE only $2^{|\mathcal{A}_\ell|}$ times. Formally, the *process equivalence class* returns

$$\mathcal{I}_\ell = \{i : i \in \mathcal{A}_\ell \cap \mathcal{I}\}\,, \quad \text{and} \quad J_\ell = \{j : j \in \mathcal{A}_\ell \cap \mathcal{I}^{\mathrm{C}}\}\,. \tag{30}$$

This concludes the proof that Algorithm 1 consistently estimates $\mathcal{I}$ set. ■

Remark 3 *After forming $\mathcal{A}_1, \ldots, \mathcal{A}_L$ classes with corresponding sets $J_0^{\mathcal{A}_1}, \ldots, J_0^{\mathcal{A}_L}$, consider a pair $\mathcal{A}_\ell, \mathcal{A}'_\ell$ where $1 \leq \ell < \ell' \leq L$. Note that for any node pair (u, v) where $u \in \mathcal{A}_\ell$ and $v \in \mathcal{A}_{\ell'}$, u is not a descendant of v. Additionally, if $J_0^{\mathcal{A}_\ell} \not\subset J_0^{\mathcal{A}_{\ell'}}$, u is not an ancestor of v. Hence, while considering $\mathcal{A}_\ell$ step of Algorithm 1, taking $\mathcal{M}_\ell = J_0^{\mathcal{A}_\ell} \cup \bigcup_{b \in \mathcal{B}_\ell} \mathcal{A}_b$ where $\mathcal{B}_\ell \triangleq \{b : J_0^{\mathcal{A}_b} \subset J_0^{\mathcal{A}_\ell},\ \ 1 \leq b < \ell\}$ is equivalent to taking $\mathcal{M}_\ell = J_0 \cup \bigcup_{1 \leq b < \ell} \mathcal{A}_b$. We use the former simplified approach to reduce the computational burden by having fewer nodes for subsequent Δ_Θ estimates.*

Proof of Theorem 2. While processing a class $\mathcal{A}_\ell$ in Algorithm 1, we declare a node j non-intervened if there exist a set $A \subset \mathcal{A}_\ell$ such that $[\Delta_{\Theta_{\mathcal{M}_\ell \cup A}}]_{j,j} = 0$. Note that there may exist more than one such $\mathcal{M}_\ell \cup A$, in which case we denote the smallest one by $\mathcal{N}_j$.

Now, define $c_j \triangleq \ell$ for all $j \in \mathcal{A}_\ell$, where ℓ is the index of the equivalence class that contains node j. We have shown in Section 4.2 that finding $\{j \to i\}_{j \notin \mathcal{I}, i \in \mathcal{I}}$ is sufficient to update MEC into $\mathcal{I}$-MEC. Therefore, our goal for a non-intervened node is to find all of its intervened children. Consider $j \in J_{c_j}$ and $i \in \mathcal{I}_{c_i}$ such that $c_j \leq c_i$. If $i \in \mathcal{N}_j$, it immediately implies that j is not a parent of i. Suppose that $i \notin \mathcal{N}_j$.

Consider $S = \mathcal{M}_{c_i} \cup \mathsf{pa}(i) \cup \{i\}$ that is also of the form $\mathcal{M}_{c_i} \cup A$ for some $A \subseteq \mathcal{A}_{c_i}$. Therefore, we compute PDE for this S in c_i-th stage of *process equivalence class*. If $j \notin \mathsf{pa}(i)$, all the paths $j \cdots \to i$ are blocked with a parent of i that is given in S. On the other hand, if the path ends with $\leftarrow i$, the path contains a collider node k that is a descendant of i. Since i is the youngest node in S, that collider k blocks the path. Therefore, $[\Theta_S]_{j,i} = 0$ and $[\Delta_{\Theta_S}]_{j,i} = 0$ if $j \notin \mathsf{pa}(i)$. From Assumption 1, if $j \in \mathsf{pa}(i)$, $[\Delta_{\Theta_S}]_{j,i} \neq 0$. Therefore, we identify all the non-intervened parents of intervened node i.

Orienting more edges. In addition to finding $\{j \to i\}_{j \notin \mathcal{I}, i \in \mathcal{I}}$, which is the main objective of Theorem 2, we can also recover the edges $\{k \to i\}_{\{k,i\} \in \mathcal{I}, c_k \neq c_i}$. Consider nodes $k \in \mathcal{I}_{c_k}$ and $i \in \mathcal{I}_{c_i}$ such that $c_k < c_i$. In other words, k and i are both intervened but they belong to different equivalence classes. Similar to the previous case, by considering set $S = \mathcal{M}_{c_i} \cup \mathsf{pa}(i) \cup i$, we obtain $[\Delta_{\Theta_S}]_{k,i} \neq 0$ if $k \in \mathsf{pa}(i)$ and $[\Delta_{\Theta_S}]_{k,i} = 0$ otherwise. Therefore, $k \notin \mathsf{pa}(i)$, and we can orient all $k \to i$ edges if both nodes are intervened and belong to different equivalence classes.

Proof of Theorem 3. We use the ADMM-based approach of [12] as our PDE function to estimate $\Delta = \Theta^{(1)} - \Theta^{(2)}$. Theorem 1 of [12] gives the sample complexity of this estimation as $O(M_\Sigma M_{\Gamma,\Gamma^T} d^4 \log p)$. In Theorem 3, we further assume that the product $M_\Sigma M_{\Gamma,\Gamma^T}$ is bounded. Accordingly, with $n = O\left(\frac{d^4}{\varepsilon^2} \frac{\log p}{\delta}\right)$ samples, PDE's output $\hat{\Delta}$ satisfies $\|\hat{\Delta} - \Delta\|_\infty < \varepsilon$ with a probability at least $1 - \delta$. We note that the conditions in Theorem 3 are given for the linear SEM over $[p]$ and the associated covariance matrices. If these conditions hold, they also hold for the restricted SEM over any $S \subset [p]$. Therefore, if we have $\|\hat{\Delta}_\Theta - \Delta_\Theta\|_\infty < \varepsilon$, we also have $\|\hat{\Delta}_{\Theta_S} - \Delta_{\Theta_S}\|_\infty < \varepsilon$ for any set S. Subsequently, we can threshold PDE outputs $\hat{\Delta}_{\Theta_S}$ by ε to exactly recover the support of Δ_{Θ_S} for any set S.

Note that Algorithm 1 requires only the support of Δ_{Θ_S} for a number of sets S. Therefore, with $n = O\left(\frac{d^4}{\varepsilon^2} \frac{\log p}{\delta}\right)$ samples, Algorithm 1 identifies $\mathcal{I}$ with a probability at least $1 - \delta$. We have shown in the proof of Theorem 2 that finding $\{j \to i\}_{j \notin \mathcal{I}, i \in \mathcal{I}}$ does not require any additional Δ_Θ estimates. Therefore, with $n = O\left(\frac{d^4}{\varepsilon^2} \frac{\log p}{\delta}\right)$ samples, Algorithm 1 also identifies the non-intervened parents of the intervened nodes $\{j \to i\}_{j \notin \mathcal{I}, i \in \mathcal{I}}$ with a probability at least $1 - \delta$. ■

We finally note that Corollary 1 of [12] explicitly assumes that both M_Σ and M_{Γ,Γ^T} are bounded to remove $M_\Sigma M_{\Gamma,\Gamma^T}$ from the sample complexity. However, it can be readily relaxed to $M_\Sigma M_{\Gamma,\Gamma^T} < \ +\infty$ since both terms always appear within the same product. We note that this relaxation brings about a significant level of flexibility in choosing covariance matrices. Indeed, this

product is closely related to the condition number of the estimation problem. Two terms correspond to the norm of the inverse of Hessian of the optimization problem and the norm of the covariance, respectively. Product of these terms, the condition number, appears in similar matrix inference problems such as graphical lasso [27].

B Additional experiments

B.1 Intervention recovery

We have compared the results of our algorithm and those of UT-IGSP for estimating intervention targets under shift intervention model in Section 5.1. We expand the simulations to various settings in this subsection. Specifically, we report the results for shift intervention model with higher density $c = 2.5$ in Table 2, increased variance setting with $c = 2.5$ in Table 3, and randomized intervention setting with with $c = 2.5$ in Table 4.

Our algorithm works well in all settings. Especially, increasing the dimension does not adversely affect accuracy and time complexity.

Table 2: $\mathcal{I}$ estimation in the shift intervention model - 50 repetitions with 5000 samples - density 2.5

	UT-IGSP ([19])				Algorithm 1			
p	Precision	Recall	F1	Time(s)	Precision	Recall	F1	Time(s)
20	0.95	0.99	0.97	0.2	0.90	0.86	0.88	0.2
40	0.89	0.99	0.94	0.6	0.87	0.91	0.89	0.3
60	0.88	1	0.94	2.0	0.86	0.96	0.91	0.4
80	0.80	1	0.89	7.0	0.86	0.94	0.90	0.5
100	0.77	1	0.87	17.7	0.87	0.98	0.92	0.5

Table 3: $\mathcal{I}$ estimation in the increased variance model - 50 repetitions with 5000 samples - density 2.5

	UT-IGSP ([19])				Algorithm 1			
p	Precision	Recall	F1	Time(s)	Precision	Recall	F1	Time(s)
20	0.90	0.99	0.95	0.2	0.89	0.86	0.87	0.2
40	0.85	1	0.92	0.6	0.87	0.93	0.90	0.3
60	0.88	1	0.93	2.4	0.89	0.97	0.92	0.3
80	0.80	1	0.89	5.8	0.86	0.97	0.91	0.4

Table 4: $\mathcal{I}$ estimation in the randomized intervention - 50 repetitions with 5000 samples - density 2.5

	UT-IGSP ([19])				Algorithm 1			
p	Precision	Recall	F1	Time(s)	Precision	Recall	F1	Time(s)
20	0.92	1	0.96	0.2	0.86	0.91	0.88	0.2
40	0.82	1	0.90	0.7	0.88	0.94	0.91	0.3
60	0.81	1	0.90	2.8	0.84	0.96	0.90	0.5
80	0.74	1	0.85	8.4	0.86	0.92	0.89	0.6

Comparison with Ghoshal's algorithm [14]. Ghoshal's algorithm in [14] is designed to estimate Δ_B, and its performance critically hinges on the noise variances to be invariant. Even though it is not designed to return intervention targets, we can define the estimated intervention set of Ghoshal's algorithm as $\hat{\mathcal{I}} \triangleq \{i : i,\ \exists\, j, (\Delta_B)_{j,i} \neq 0\}$. We run our algorithm and Ghoshal's algorithm on the randomized intervention setting described in Section 5.1 and report the results in Table 5. Expectedly, Ghoshal's algorithm does not perform well due to violation of the invariant noise variance assumption.

Table 5: $\mathcal{I}$ estimation in the randomized intervention model - 100 repetitions with 10000 samples - density 2.5

	Ghoshal [14]				Algorithm 1			
p	Precision	Recall	F1	Time(s)	Precision	Recall	F1	Time(s)
20	0.74	0.62	0.67	<0.1	0.92	0.92	0.92	0.6
40	0.73	0.68	0.70	0.1	0.91	0.94	0.93	0.6
60	0.70	0.69	0.69	0.2	0.91	0.96	0.94	0.6
80	0.69	0.66	0.67	0.3	0.91	0.96	0.93	0.6
100	0.66	0.63	0.64	0.4	0.91	0.95	0.93	0.7

Increased number of samples. Theorem 1 states that our algorithm is consistent. Figure 1 shows that the performance of the algorithm increases significantly with the increased number of samples in all of the considered settings. We provide additional evidence of this fact. We generate 50 random graphs with density $c = 2.5$ for each of the shift intervention, increased variance, and randomized intervention settings. We report the F1 scores for each setting with 5000, 10000, and 20000 samples in Table 6.

Table 6: $\mathcal{I}$ estimation with increased number of samples - 50 repetitions - density 2.5

	Shift Intervention			Increased Variance			Randomized Intervention		
p	5000	10000	20000	5000	10000	20000	5000	10000	20000
40	0.87	0.90	0.91	0.95	0.96	0.96	0.90	0.92	0.94
60	0.90	0.92	0.93	0.93	0.96	0.97	0.91	0.93	0.95
80	0.90	0.91	0.94	0.93	0.97	0.98	0.91	0.94	0.95
100	0.93	0.94	0.96	0.94	0.97	0.97	0.89	0.93	0.92

B.2 Causal structure learning

In Section 4.2, we have shown that our method recovers the new information that can be gained through interventions. Hence, Algorithm 1 refines the given MEC into the $\mathcal{I}$-MEC. Accordingly, we test our algorithm for the causal structure recovery task in this subsection.

First, we take the correct CPDAG of $\mathcal{G}^{(1)}$ and apply our algorithm's findings to obtain $\mathcal{I}$-CPDAG. We run 100 realizations of Erdős-Rényi graphs with $c = 2$ and 10000 samples. For different values of graph size p, we consider fixed target set size $|\mathcal{I}| = 5$ or growing target set size $|\mathcal{I}| = p/10$. We report the results for recovery of $\mathcal{I}$-directed edges in Table 7.

Table 7: Recovery of $\mathcal{I}$-directed edges in the increased variance model

	$\lvert\mathcal{I}\rvert = 5$				$\lvert\mathcal{I}\rvert = p/10$			
p	Precision	Recall	F1	Time(s)	Precision	Recall	F1	Time(s)
40	0.69	0.93	0.80	0.15	0.73	0.94	0.82	0.11
60	0.73	0.93	0.82	0.24	0.73	0.93	0.82	0.25
80	0.75	0.93	0.83	0.28	0.73	0.96	0.83	0.45
100	0.82	0.97	0.89	0.42	0.72	0.93	0.81	0.86

Next, we consider recovering the non-intervened parents of the intervened nodes, i.e., $\{j \to i\}_{j \notin \mathcal{I}, i \in \mathcal{I}}$. We note that we do not use any given MEC information in this setting. Therefore, a comparison with UT-IGSP algorithm becomes feasible. We report the results for $|\mathcal{I}| = 5$ in Table 8. Similar to the intervention recovery task, our algorithm's runtime does not suffer from increasing the dimension while the runtime of UT-IGSP grows very quickly.

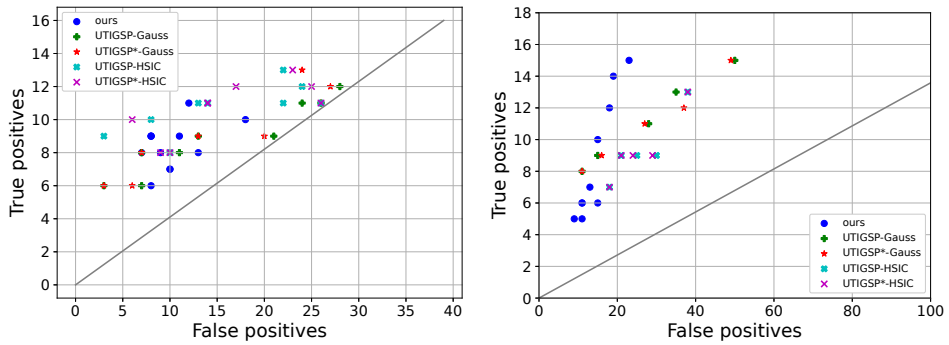
Table 8: Recovery of non-intervened parents of intervened nodes

	UT-IGSP ([19])				Algorithm 1			
p	Precision	Recall	F1	Time(s)	Precision	Recall	F1	Time(s)
20	0.76	0.98	0.86	0.32	0.81	0.81	0.81	0.15
40	0.82	0.98	0.89	2.30	0.85	0.79	0.82	0.22
60	0.84	0.98	0.91	10.11	0.88	0.85	0.86	0.26
80	0.89	0.99	0.93	32.97	0.92	0.78	0.85	0.28

B.3 Application to real data

We have investigated directed edge recovery results for two real biological datasets in Section 5.3. In this subsection, we give the skeleton recovery results for the same datasets. Figure 3 illustrates that our observations from the directed edge recovery hold for the skeleton recovery as well. Comparison of figures 2 and 3 reveals that our algorithm orients fewer number of edges incorrectly with respect to UT-IGSP algorithm.

Hyperparameters. We have defined the regularization parameters λ_1, λ_2, and λ_3 for our algorithm and cut-off value α for UT-IGSP in Section 5. Specifically, we have used $\lambda_1 \in [0.1, 0.3]$, $\lambda_2 = 0.2$, and $\lambda_3 \in [0.05, 0.2]$ for Algorithm 1, and $\alpha \in [0.0001, 0.5]$ for UT-IGSP while creating figures 2a and 3a. Similarly, we have used $\lambda_1 = 0.1$, $\lambda_2 = 0.05$, and $\lambda_3 \in [0.005, 0.1]$ for Algorithm 1, and $\alpha \in [0.005, 0.1]$ for UT-IGSP while creating figures 2b and 3b.



(a) Sachs [22] dataset skeleton recovery

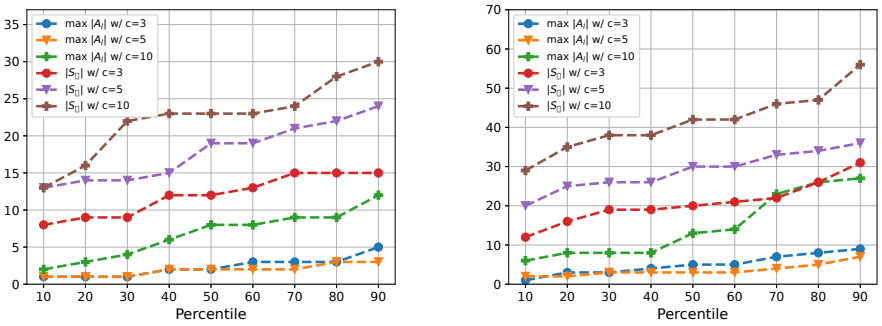
(b) Dixit [24] dataset skeleton recovery

Figure 3: ROC curves for skeleton recovery. The solid grey line corresponds to random guessing.

B.4 Computational complexity

We have stated in Section 4 that the computational complexity of our algorithm is exponential in the size of the largest equivalence class, $\max |\mathcal{A}_\ell|$. This can be as large as p_Δ in some extreme examples. One possible scenario for this case is if the parents of intervention targets are also intervened. In this case, J_0 will be the empty set and all nodes in S_Δ will belong to the same group. However, this requires the interventions to concentrate in one neighborhood such that parents of the intervened nodes will also be intervened. In reality, such scenarios happen rarely, and interventions are generally distributed.

We generate 1000 instances of random graphs with $p = 100$, various densities, and target set sizes to demonstrate the much smaller size of $\mathcal{A}_\ell$ groups with respect to S_Δ. Figure 4 illustrates that $\max |\mathcal{A}_\ell|$ is much smaller than p_Δ. Indeed, Fig. 4 also shows the limitations of some of the related work that has computational complexity exponential in p_Δ strictly. For instance, for $p = 100, |\mathcal{I}| = 5$, and $c = 5$ in Fig. 4a, the 90%-th percentile of p_Δ is 25, whereas $\max_\ell |\mathcal{A}_\ell|$ is only 4. Gains of our algorithm become more dramatic when the target set is larger. For instance, for $p = 100, |\mathcal{I}| = 10$, and $c = 5$ in Fig. 4b, the 50%-th percentile of p_Δ is 34, whereas the 90%-th percentile of $\max_\ell |\mathcal{A}_\ell|$ is only 10. Therefore, our algorithm can scale up to higher dimensions.



(a) $p = 100, |\mathcal{I}| = 5$ (b) $p = 100, |\mathcal{I}| = 10$

Figure 4: Exponential factor in the computational complexity of our algorithm, $\max_\ell |\mathcal{A}_\ell|$, is much smaller than the size of the affected nodes $p_\Delta = |S_\Delta|$. x-axis shows the percentile values over 1000 different random DAG instances. Largest class size $\max_l |\mathcal{A}_\ell|$ and p_Δ are plotted for three different density values.

We finally comment on the computational complexity of the PDE routine. The ADMM-based PDE algorithm of [12] has $O(p^3)$ complexity. We note that we run PDE with all $[p]$ nodes only once during the S_Δ estimation in Step 1. Hence, the estimation with $O(p^3)$ complexity will only be performed once. The rest of the PDE instances require much smaller number of nodes as stated in Remark 3. We note that a related study in [14] uses another PDE algorithm that has complexity $O(p^4)$. Reducing it to $O(p^3)$ is a significant gain, which allows us to process hundreds of nodes.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See the last paragraph in Section 6.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See the last paragraph in Section 6.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Section 3 for exact problem definition and Assumption 1 in Section 4 for assumptions.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix A.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See the link at the end of the abstract.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 5 and the code in supplemental material.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] Standard deviation of the precision and recall rates of intervention recovery are given in Table 1 in Section 5.1.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See first paragraph in Section 5.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 5.

(b) Did you mention the license of the assets? [Yes] See the footnote in Section 5.3.

(c) Did you include any new assets either in the supplemental material or as a URL? [Yes] The code for our algorithm and simulations are provided in the supplemental material.

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]