Latent Hierarchical Causal Structure Discovery with Rank Constraints

Biwei Huang [*1] **Charles Low** [*1], **Feng Xie**[3], **Clark Glymour**[1], **Kun Zhang**[1,2]
[1] Carnegie Mellon University
[2] Mohamed bin Zayed University of Artificial Intelligence
[3] Beijing Technology and Business University, China
{bwei.huang, charleslow88, xiefeng009}@gmail.com,
cg09@andrew.cmu.edu, kunz1@cmu.edu

Abstract

Most causal discovery procedures assume that there are no latent confounders in the system, which is often violated in real-world problems. In this paper, we consider a challenging scenario for causal structure identification, where some variables are latent and they form a hierarchical graph structure to generate the measured variables; the children of latent variables may still be latent and only leaf nodes are measured, and moreover, there can be multiple paths between every pair of variables (i.e., it is beyond tree structure). We propose an estimation procedure that can efficiently locate latent variables, determine their cardinalities, and identify the latent hierarchical structure, by leveraging rank deficiency constraints over the measured variables. We show that the proposed algorithm can find the correct Markov equivalence class of the whole graph asymptotically under proper restrictions on the graph structure.

1 Introduction

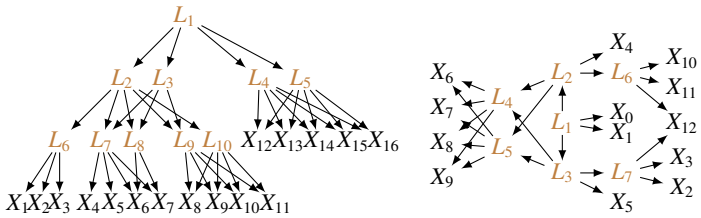
In many cases, the common assumption in causal discovery algorithms–no latent confounders–may not hold. For example, in complex systems, it is usually hard to enumerate and measure all task-related variables, so there may exist latent variables that influence multiple measured variables, the ignorance of which may introduce spurious correlations among measured variables. Much effort has been made to handle the confounding problem in causal structure learning. One research line considers the causal structure over measured variables, including FCI and its variants [Spirtes et al., 2000, Pearl, 2000, Colombo et al., 2012, Akbari et al., 2021], matrix decomposition-based approaches [Chandrasekaran et al., 2011, 2012, Frot et al., 2019], and over-complete ICA-based ones [Hoyer et al., 2008, Salehkaleybar et al., 2020].

Another line focuses on identifying the causal structure among latent variables, including Tetrad condition-based approaches [Silva et al., 2006, Kummerfeld and Ramsey, 2016], high-order moments-based ones [Shimizu et al., 2009, Cai et al., 2019, Xie et al., 2020, Adams et al., 2021, Chen et al., 2022], matrix decomposition-based approach Anandkumar et al. [2013], copula model-based approach [Cui et al., 2018], mixture oracles-based approach [Kivva et al., 2021], and multiple domains-based approach [Zeng et al., 2021]. Moreover, regarding the scenario of latent hierarchical structures, previous work along this line assumes a tree structure [Pearl, 1988, Zhang, 2004, Choi et al., 2011, Drton et al., 2017], where there is one and only one undirected path between every pair of variables. This assumption is rather restrictive and the structure in real-world problems could be more complex–beyond a tree.

In this paper, we consider a more challenging scenario where latent causal variables form a hierarchical graph structure to generate measured variables—the children of latent variables may still be latent

[*]These authors contributed equally to this work.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).



(a) (b)

Figure 1: Example hierarchical graphs that our method can handle, where X_i are measured variables and L_i are latent variables.

Pa:	Parents
PCh:	Pure children
PDe:	Pure descendants
Gp:	Grandparents
Sib:	Siblings
$\mathcal{M}$:	Measured pure descendants

Table 1: Graphical notations.

and only the leaf nodes are measured, and moreover, there can be multiple paths between every pair of variables (see the example hierarchical graphs in Figure 1). We aim to find out identifiability conditions of the hierarchical structure that are as mild as possible, and meanwhile, develop an efficient algorithm with theoretical guarantees to answer the following questions. (1) How can we locate latent parents for both measured and latent variables, as well as determining the cardinality of the latent parents, by only providing the leaf nodes? (2) How can we identify the causal relationships among latent variables and those from latent variables to measured variables?

Interestingly, we can answer these questions by properly making use of rank deficiency constraints; finding and leveraging rank properties in specific ways enable us to identify the Markov equivalence class of the whole graph, under appropriate conditions. Our contribution is mainly two-fold:

- We propose a structure identification algorithm that can efficiently locate latent variables (including their cardinalities) and identify the latent hierarchical structure, by leveraging the rank deficiency.
- We show that the proposed algorithm can find the correct graph asymptotically under mild restrictions of the graph structure. Roughly speaking, we show that it is sufficient to have $k+1$ pure children (which can be latent), as well as another $k+1$ neighbors, to identify the latent variable set with size k (see the detailed conditions in Definition 4 and Condition 1).

It is worth mentioning that rank constraints have been used in previous methods [Silva et al., 2006, Kummerfeld and Ramsey, 2016], but they assume that each latent variable has three measured ones as children and each measured variable has only one latent parent. There are also other methods for latent structure learning; for instance, Anandkumar et al. [2013], which uses matrix decomposition, needs $3k$ measured children, and the GIN-based method [Xie et al., 2020], which makes use of high-order statistics, needs $2k$ measured children. However, all those developments require that every latent variable should have measured variables as children. Very recently, Xie et al. [2022] proposes an approach for latent hierarchical structure by leveraging the GIN condition under linear non-Gaussian models, but it assumes that each variable has only one parent, where both figures in Figure 1 do not satisfy.

This paper is organized as follows. In Section 2, we give formal definitions of the latent hierarchical causal model under investigation and give conditions that are essential to the identifiability of the graph structure. In Section 3, we propose an efficient algorithm that makes use of rank-deficiency constraints to identify the latent hierarchical structure. Moreover, we show theoretically in Section 4 that the proposed algorithm outputs the correct Markov equivalence class of the whole graph asymptotically. In Section 5, we empirically validate the proposed approach on synthetic data. Notations for graphical representations that are used in the paper are provided in Table 1.

2 Latent Hierarchical Causal Model

In this paper, we focus on latent hierarchical causal model with graph structure $\mathcal{G}$, where both measured variables $\mathbf{X}_{\mathcal{G}} = \{X_1, \cdots, X_m\}$ and latent variables $\mathbf{L}_{\mathcal{G}} = \{L_1, \cdots, L_n\}$ are generated by their latent parents in a directed acyclic graph (DAG) with linear relationships:

$$X_i = \sum_{L_j \in Pa(X_i)} b_{ij} L_j + \varepsilon_{X_i}, \quad L_j = \sum_{L_k \in Pa(L_j)} c_{jk} L_k + \varepsilon_{L_j}, \tag{1}$$

where b_{ij} and c_{jk} are the causal strength from L_j to X_i and from L_k to L_j, respectively, and ε_{X_i} and ε_{L_j} are noise terms that are independent of each other. Without loss of generality, we assume that all variables in $\mathbf{X}_{\mathcal{G}}$ and $\mathbf{L}_{\mathcal{G}}$ have zero mean.

Below, we first give the general definition of a linear latent hierarchical graphical model in Definition 1. Then we give more detailed conditions on the graph structure (Definitions 2-4) that are essential to formalize the identifiability condition of the latent hierarchical structure.

Definition 1 (Linear Latent Hierarchical (L^2H) Model)**.** *A graphical model, with its graph $\mathcal{G} = (\mathbf{V}_\mathcal{G}, \mathbf{E}_\mathcal{G})$, is a linear latent hierarchical model if:*

1. *$\mathbf{V}_\mathcal{G} = \mathbf{X}_\mathcal{G} \bigcup \mathbf{L}_\mathcal{G}$, where $\mathbf{X}_\mathcal{G}$ is the set of measured variables and $\mathbf{L}_\mathcal{G}$ is the set of latent variables,*
2. *there is at least one undirected path between every pair of variables, and*
3. *each variable in $\mathbf{X}_\mathcal{G}$ and $\mathbf{L}_\mathcal{G}$ are generated by the structural equation models in Eq. 1.*

Generally, without further constraints, the causal structure of the L^2H model is hard to be identified. It has been shown that if the underlying graph structure satisfies a tree [Pearl, 1988], then the structure is identifiable. However, this structural constraint may be too strong to hold in many real-world problems. In this paper, we give sufficient conditions that are much milder than previous ones, as well as an efficient search algorithm, for the identifiability of the causal structure.

We now give the corresponding definitions, including *pure children*, *pure descendants*, and *effective cardinality*, that will be used in the identifiability condition, together with illustrative examples.

Definition 2 (Pure Children)**.** *Variables $\mathbf{V}$ are pure children of a set of latent variables $\mathbf{L}$ in a graph $\mathcal{G}$, if $Pa_\mathcal{G}(\mathbf{V}) = \mathbf{L}$ and $\mathbf{V} \cap \mathbf{L} = \emptyset$. That is, $\mathbf{V}$ have no other parents than $\mathbf{L}$. We denote the pure children of $\mathbf{L}$ by $PCh_\mathcal{G}(\mathbf{L})$.*

Accordingly, *Pure Descendants* of a set of latent variables $\mathbf{L}$ are defined as all recursive pure children of $\mathbf{L}$ (including $PCh_\mathcal{G}(\mathbf{L}), PCh_\mathcal{G}(PCh_\mathcal{G}(\mathbf{L}))$, etc.), denoted by $PDe_\mathcal{G}(\mathbf{L})$. Furthermore, measured variables that are pure descendent of $\mathbf{L}$ are called *Measured Pure Descendants*, denoted by $\mathcal{M}_\mathcal{G}(\mathbf{L})$.

Example 1. *In Figure 1(a), the pure children of $\{L_2, L_3\}$ are $\{L_6, \cdots, L_{10}\}$, its pure descendants are $\{L_6, \cdots, L_{10}, X_1, \cdots, X_{11}\}$, and its measured pure descendants are $\{X_1, \cdots, X_{11}\}$.*

Definition 3 (Effective Cardinality)**.** *For a set of latent variables $\mathbf{L}$, denote by $\mathbf{C}$ the largest subset of $PCh_\mathcal{G}(\mathbf{L})$ such that there is no subset $\mathbf{C}' \subseteq \mathbf{C}$ satisfying $|\mathbf{C}'| > |Pa_\mathcal{G}(\mathbf{C}')|$ and $|Pa_\mathcal{G}(\mathbf{C}')| < |\mathbf{L}|$. Then, the effective cardinality of $\mathbf{L}$'s pure children is $|\mathbf{C}|$.*

In the case when $\mathbf{L}$ and its pure children are fully connected, the effective cardinality is just the cardinality of the pure children of $\mathbf{L}$. However, for Figure 1(a), the effective cardinality of the pure children of $\{L_7, L_8\}$ is 3, because here the largest subset that satisfies the condition is $\{X_5, X_6, X_7\}$.

We further define *latent atomic cover* that constrains the number of pure children and neighbours for latent variables, which are essential for structural identifiability.

Definition 4 (Latent Atomic Cover)**.** *Let $\mathbf{L} = \{L_1, ..., L_k\}$ be a set of latent variables in graph $\mathcal{G}$, with $|\mathbf{L}| = k$. We say that $\mathbf{L}$ is a latent atomic cover if the following conditions are met:*

1. *there exists a subset of pure children $\mathbf{C}' \subseteq PCh_\mathcal{G}(\mathbf{L})$ with effective cardinality $\geq k + 1$;*
2. *there exists a neighbour set $\mathbf{B}$ to $\mathbf{L}$ s.t. $\mathbf{B} \bigcap \mathbf{C}' = \emptyset$ and $|\mathbf{B}| = k + 1$;*
3. *there does not exist a partition of $\mathbf{L} = \mathbf{L}_1 \cup \mathbf{L}_2$, so that both $\mathbf{L}_1, \mathbf{L}_2$ satisfy conditions 1 and 2 and $\{PCh_\mathcal{G}(\mathbf{L}_1) \cup PCh_\mathcal{G}(\mathbf{L}_2)\} \backslash \mathbf{L} = PCh_\mathcal{G}(\mathbf{L})$.*

Example 2. *In Figure 1(a), $\mathbf{L} = \{L_2, L_3\}$ is a latent atomic cover with $k = 2$, because (1) there exists a subset of pure children $\mathbf{C}' = \{L_6, L_7, L_8\}$ with effective cardinality $3 = k + 1$, (2) there exists a neighbor set $\mathbf{B} = \{L_1, L_9, L_{10}\}$, s.t. $\mathbf{B} \cap \mathbf{C}' = \emptyset$ and $|\mathbf{B}| = 3 = k + 1$, and (3) neither $\{L_2\}$ or $\{L_3\}$ satisfies the above two conditions.*

We now give the conditions for structural identifiability from measured variables $\mathbf{X}_\mathcal{G}$ alone, including those on the structural constraints (Condition 1) and rank faithfulness assumption (Condition 2).

Condition 1 (Irreducible Linear Latent Hierarchical (IL^2H) Graph)**.** *An L^2H graph $\mathcal{G}$ is an IL^2H graph if*

1. *every latent variable $L \in \mathbf{L}$ in $\mathcal{G}$ belongs to at least one latent atomic cover,*
2. *for any pair of latent atomic covers $(\mathbf{L}_A, \mathbf{L}_B)$, if $PDe_\mathcal{G}(\mathbf{L}_A) \bigcap PDe_\mathcal{G}(\mathbf{L}_B) \neq \emptyset$, then either (a) $\mathbf{L}_A \subset \mathbf{L}_B$ or (b) $\mathbf{L}_A \subset PDe_\mathcal{G}(\mathbf{L}_B)$ or (c) $\mathbf{L}_B \subset \mathbf{L}_A$ or (d) $\mathbf{L}_B \subset PDe_\mathcal{G}(\mathbf{L}_A)$, and*
3. *for any three latent atomic covers $\mathbf{L}_A, \mathbf{L}_B, \mathbf{L}_C$, if the causal structure satisfies $\mathbf{L}_A \rightarrow \mathbf{L}_B \rightarrow \mathbf{L}_C$ or $\mathbf{L}_A \leftarrow \mathbf{L}_B \rightarrow \mathbf{L}_C$, and $|\mathbf{L}_B| = k$, then $\mathbf{L}_B$ has $2k$ neighbors, except for $\mathbf{L}_A, \mathbf{L}_C$ and the parents in the v structure where $\mathbf{L}_B$ is a collider.*

Next, we give the faithfulness assumption, which holds for generic covariance matrices consistent with $\mathcal{G}$ [Spirtes, 2013].

Condition 2 (Rank Faithfulness)**.** *A probability distribution P is rank faithful to a DAG $\mathcal{G}$ if every rank constraint on a sub-covariance matrix that holds in P is entailed by every linear structural model with respect to $\mathcal{G}$.*

We will show in Section 4 that if the underlying latent hierarchical graph satisfies an IL^{2}H graph and the rank faithfulness holds, then the location and cardinality of latent variables, and the causal structure among latent atomic covers and that from latent atomic covers to measured variables, are identifiable with appropriate search procedures. Below, let us first present the identification procedure.

3 Structure Identification with Rank-Deficiency Constraints

We propose an identification algorithm (Algorithm 1) to identify the structure of IL^{2}H graphs, by leveraging rank-deficiency constraints of measured variables. In particular, the algorithm includes three phases: finding causal clusters and assigning latent atomic covers in a greedy manner ("*findCausalClusters*"), refining incorrect clusters and covers due to the greedy search ("*refineClusters*"), and refining edges and finding v structures ("*refineEdges*").

Algorithm 1: Latent Hierarchical Causal Structure Discovery

Input : Date from a set of measured variables $\mathbf{X}_{\mathcal{G}}$
Output : Markov equivalence class $\mathcal{G}'$

```
G' = findCausalClusters (X_G) ;        // find clusters and assign latent covers greedily
G' = refineClusters (G') ;     // refine incorrect clusters and covers from greedy search
G' = refineEdges (G') ;                        // refine some edges and find v structures
```

Before describing the identification algorithm, we first give a theorem that relates the graphical structure of an IL^{2}H graph to the rank constraints over the covariance matrix of measured variables.

Theorem 1 (Graphical Implication of Rank Constraints in IL^{2}H Graphs)**.** *Suppose $\mathcal{G}$ satisfies an IL^{2}H graph. Under the rank faithfulness assumption, the cross-covariance matrix $\Sigma_{\mathbf{X}_A,\mathbf{X}_B}$ over measured variables $\mathbf{X}_A$ and $\mathbf{X}_B$ in $\mathcal{G}$ (with $|\mathbf{X}_A|, |\mathbf{X}_B| > r$) has rank r, if and only if there exists a subset of latent variables $\mathbf{L}$ with $|\mathbf{L}| = r$ such that $\mathbf{L}$ d-separates $\mathbf{X}_A$ from $\mathbf{X}_B$, and there is no $\mathbf{L}'$ with $|\mathbf{L}'| < |\mathbf{L}|$ that d-separates $\mathbf{X}_A$ from $\mathbf{X}_B$. That is,*

$$rank(\Sigma_{\mathbf{X}_A,\mathbf{X}_B}) = min\{|\mathbf{L}| : \mathbf{L} \text{ d-separates } \mathbf{X}_A \text{ from } \mathbf{X}_B\}.$$

For instance, in Figure 1(a), $rank(\Sigma_{\{X_1,X_2\},\{X_3,X_4\}}) = 1$, because L_6 d-separates $\{X_1, X_2\}$ from $\{X_3, X_4\}$ with $|L_6| = 1$.

Rank Test. We test rank deficiency by leveraging canonical correlations [Anderson, 1984]. Specifically, the number of non-zero canonical correlations between two random vectors is equal to the rank. Denote by α_i the i-th canonical correlation coefficient between $\mathbf{X}_A$ and $\mathbf{X}_B$. Then under the null hypothesis that $\texttt{rank}(\Sigma_{\mathbf{X}_A,\mathbf{X}_B}) \leq r$, the statistics $-(N-(p+q+3)/2)\sum_{i=r+1}^{\max(p,q)} \log(1-\alpha_i^2)$ is approximately χ^2 distributed with $(p-r)(q-r)$ degrees of freedom, where $p = |\mathbf{X}_A|$, $q = |\mathbf{X}_B|$, and N is the sample size.

We further show that for any subset of latent variables in an IL^{2}H graph, we can use the measured variables as surrogates to estimate the rank, as indicated in the following theorem.

Theorem 2 (Measurement as a surrogate)**.** *Suppose $\mathcal{G}$ is an IL^{2}H graph. Denote by $\mathbf{A}, \mathbf{B} \subseteq \mathbf{V}_{\mathcal{G}}$ two subsets of variables in $\mathcal{G}$, with $\mathbf{A} \cap \mathbf{B} = \emptyset$. Furthermore, denote by $\mathbf{X}_A$ the set of measured variables that are d-separated by $\mathbf{A}$ from all other measures, and by $\mathbf{X}_B$ the set of measured variables that are d-separated by $\mathbf{B}$ from all other measures. Then $\texttt{rank}(\Sigma_{\mathbf{A},\mathbf{B}}) = \texttt{rank}(\Sigma_{\mathbf{X}_A,\mathbf{X}_B})$.*

A special case when Theorem 2 holds is that $\mathbf{X}_A$ and $\mathbf{X}_B$ are the measured pure descendants of $\mathbf{A}$ and $\mathbf{B}$, respectively. Note that Theorem 1 is a special case of Theorem 2.8 in Sullivant et al. [2010] when applied to IL^{2}H graphs. Different from the setting in Sullivant et al. [2010] where access to the full covariance matrix $\Sigma_{\mathbf{V}_{\mathcal{G}},\mathbf{V}_{\mathcal{G}}}$ is assumed, we only have access to the covariance matrix $\Sigma_{\mathbf{X}_{\mathcal{G}},\mathbf{X}_{\mathcal{G}}}$ over the measured variables $\mathbf{X}_{\mathcal{G}}$, which we will use to infer the causal structure over the entire graph $\mathcal{G}$. For this reason, although we can infer the number of latent variables that d-separate any two sets of measured variables $\mathbf{X}_A, \mathbf{X}_B$, we cannot directly know the exact location of these variables in the graph. Fortunately, the structure constraints of the IL^{2}H graph and Theorem 2 will allow us to reconstruct the graph with certain search procedures, as shown below.

3.1 Phase I: Finding Causal Clusters

We start to discover clusters in a recursive and greedy manner, by performing rank-deficiency tests over measured variables. We denote by $\mathcal{S}$ a set of active variables that is under investigation; $\mathcal{S}$ is set to $\mathbf{X}_{\mathcal{G}}$ initially and will be updated to include latent atomic covers when rank deficiency is discovered. Below, we first give the definition about set size and the definition of *atomic rank-deficiency set*, which will be used in Rule 1 and Algorithm 2.

Definition 5 (Set Size)**.** *Suppose $\mathcal{S}$ is a set. We denote by $|\mathcal{S}|$ the cardinality of $\mathcal{S}$; that is, $|\mathcal{S}|$ is the number of elements S_i in $\mathcal{S}$, for $S_i \in \mathcal{S}$. We denote by $\|\mathcal{S}\|$ the number of variables in $\mathcal{S}$, where $\|\mathcal{S}\| = |\bigcup_i S_i|$, for $S_i \in \mathcal{S}$.*

For example, suppose $\mathcal{S} = \{\{L_1\}, \{L_6\}, \{L_7, L_8\}, \{L_9, L_{10}\}\}$. Then $|\mathcal{S}| = 4$ and $\|\mathcal{S}\| = 6$.

Definition 6 (Atomic Rank-Deficiency Set)**.** *Given a graph $\mathcal{G}'$. Denote by $\mathcal{S} \subseteq V_{\mathcal{G}'}$ an active set of variables that is under investigation. Let $\mathbf{A} \subset \mathcal{S}$ and $\mathbf{B} = \mathcal{S} \backslash \mathbf{A}$. Denote by $\mathbf{X}_A = \mathcal{M}_{\mathcal{G}'}(\mathbf{A})$ and $\mathbf{X}_B = \mathcal{M}_{\mathcal{G}'}(\mathbf{B})$ the measured pure descendants of $\mathbf{A}$ and $\mathbf{B}$, respectively. If (1) $\Sigma_{\mathbf{A},\mathbf{B}}$ is rank deficient, i.e., $\texttt{rank}(\Sigma_{\mathbf{A},\mathbf{B}}) < min\{\|\mathbf{A}\|, \|\mathbf{B}\|\}$, and (2) no proper subset of $\mathbf{A}$ is rank deficient, then $\mathbf{A}$ is called an atomic rank-deficient set, and $\Sigma_{\mathbf{A},\mathbf{B}}$ can be estimated by $\Sigma_{\mathbf{X}_A,\mathbf{X}_B}$.*

For example, in Figure 1(a), suppose now we have the measured variables $\mathbf{X}_{\mathcal{G}}$ under investigation, so the active variable set $\mathcal{S} = \mathbf{X}_{\mathcal{G}}$. $\mathbf{A} = \{X_1, X_2\} \subset \mathcal{S}$ forms an atomic rank-deficiency set because the cross-covariance matrix of $\mathbf{X}_A$ against all other measures has rank 1 (i.e., rank deficient). According to Theorem 1, the rank deficiency occurs because the latent parent L_6 of $\{X_1, X_2\}$ d-separates them from all other measures. This naturally leads to the following rule that assigns a latent atomic cover over the rank-deficiency set:

Rule 1: If $\mathbf{A}$ is a rank-deficiency set with $\texttt{rank}(\Sigma_{\mathbf{X}_A,\mathbf{X}_B}) = k$, then assign a latent atomic cover $\mathbf{L}$ of size k as the parent of every variable $A_i \in \mathbf{A}$.

Later in Phase II, we will show that Rule 1 may not correctly identify the latent atomic cover in some cases, and, accordingly, we will further provide an efficient revision procedure in Phase II. At the current phase, we use Rule 1, together with certain search procedures, to identify the clusters and latent atomic covers. The detailed search procedure of Phase I is given in Algorithm 2 (*findCausalClusters*), which tests for rank deficiency recursively to discover clusters of variables and their latent atomic clusters. The set of active variables $\mathcal{S}$ is set to $\mathbf{X}_{\mathcal{G}}$ initially (line 1) and will be updated as the search goes on (line 14). We start to identify the latent atomic cover with size $k = 1$ (line 1). We consider any subset of the latent atomic covers in $\mathcal{S}$ and replace them with their pure children, resulting in $\tilde{\mathcal{S}}$ (line 4). Then we draw a subset of variables $\mathbf{A} \subset \tilde{\mathcal{S}}$ with cardinality at least $k + 1$ and conduct a rank deficiency test of $\mathbf{A}$ against $\mathcal{S} \backslash \mathbf{A}$ by estimating the rank of $\texttt{rank}(\Sigma_{\mathbf{X}_A,\mathbf{X}_B})$ (lines 6-7). Note that here $\mathbf{X}_A$ and $\mathbf{X}_B$ are the measured pure descendants of $\mathbf{A}$ and $\mathbf{B}$, respectively, in the currently learned graph. This step is repeated until all subsets are tested (lines 5-9). If rank deficiency is found, we merge the overlapping groups into a cluster and add latent covers over the cluster (lines 10-14). We further reset $k = 1$ and resume the search (line 15). Otherwise, if no latent cover is found, we increment $k = k + 1$ (line 17). This procedure is repeated until no more clusters are found. Finally, we connect the elements in $\mathcal{S}$ into a chain structure (line 19). Figure 2 illustrates an example procedure of finding new clusters by applying *findCausalClusters* to the measured variables $\mathbf{X}_{\mathcal{G}}$ generated from the structure in Figure 1(a).

3.2 Phase II: Refining Clusters

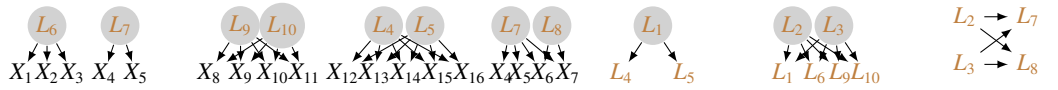
As we have mentioned above, the naive assignment of causal clusters in Algorithm 2 (*findCausal-Clusters*) may not be correct in some cases. In this section, we provide a precise characterization of the cases where *findCausalClusters* incorrectly clusters variables and, accordingly, propose an efficient algorithm to correct such cases.

Take Figure 3(a) as an example to illustrate the issue. By applying *findCausalClusters* to the measured variables $\mathbf{X}_{\mathcal{G}}$, at $k = 2$, we discovered that $\{X_9, X_{10}, X_{11}\}$ form a cluster and then set a 2-latent atomic cover $\{L'_1, L'_2\}$ over it (Figure 3(b)). This is not correct, because X_9, X_{10}, X_{11} actually belong to three different 1-latent atomic covers, L_1, L_2, L_3, respectively. The incorrect clustering and covering are because when discovering the rank-deficiency set $\{X_9, X_{10}, X_{11}\}$, the latent covers $\{L_4, L_5\}, \{L_6, L_7\}$ have not been identified yet. If $\{L_4, L_5\}$ had already been discovered, we would correctly find that $\{X_9, L_4, L_5\}$ form a 1-latent atomic cover, and the algorithm would proceed correctly thereafter. More generally, *findCausalClusters* may set incorrect cluster for the rank-deficiency set $\{X_9, X_{10}, X_{11}\}$, in

Algorithm 2: Phase I: findCausalClusters

Input : Data from a set of measured variables $\mathbf{X}_{\mathcal{G}}$
Output : Graph $\mathcal{G}'$

Active set $\mathcal{S} \leftarrow \mathbf{X}_{\mathcal{G}}$; $k \leftarrow 1$;
repeat
  repeat
    draw a set of latent atomic covers $\tilde{\mathbf{L}} \subset \mathcal{S}$; let $\tilde{\mathcal{S}} = (\mathcal{S} \backslash \tilde{\mathbf{L}}) \cup (\cup_{L_i \in \tilde{\mathbf{L}}} PCh(L_i))$;
    repeat
      draw a set of test variables $\mathbf{A} \subset \tilde{\mathcal{S}}$, with $\|\mathbf{A}\| \geq k+1$, and let $\mathbf{B} \leftarrow \tilde{\mathcal{S}} \backslash \mathbf{A}$;
      $k' \leftarrow \texttt{rank}(\Sigma_{\mathbf{X}_A, \mathbf{X}_B})$ estimated by rank deficiency test;
      if $k' < k+1$ **then** rank deficiency found and keep track of this set $\mathbf{A}$;
    until *all subsets* $\mathbf{A}$ *exhausted*;
    if *any groups with rank deficiency are found* **then**
      merge overlapping clusters; identify lowest rank k' found;
      foreach *discovered cluster of variables* $\mathbf{A}$ *with rank* k' **do**
        create latent cover $\mathbf{L}$ with cardinality k' as parents of $\mathbf{A}$;
        $\mathcal{S} = (\mathcal{S} \backslash \mathbf{A}) \cup \mathbf{L}$;
      $k = 1$; **break**;
  until *all subsets* $\tilde{\mathbf{L}}$ *exhausted*;
  if *no group with rank deficiency is found* **then** $k = k+1$;
until *no more clusters are found*;
for all $S_i \in \mathcal{S}$, connect them into a chain structure;
return $\mathcal{G}'$



(a) $k = 1$, $\mathcal{S} = \{X_1, \cdots, X_{16}\}$. (b) $k=2$, $\mathcal{S} = \{\{L_6\}, \{L_7\}, X_6, \cdots, X_{16}\}$. (c) $k=1$, $\mathcal{S} = \{\{L_4, L_5\}, \{L_6\}, \{L_7, L_8\}, \{L_9, L_{10}\}\}$. (d) $k = 2$, $\mathcal{S} = \{\{L_1\}, \{L_7, L_8\}, \{L_6\}, \{L_9, L_{10}\}\}$. (e) Connect $\{L_2, L_3\}$ and $\{L_7, L_8\}$.

Figure 2: An illustration of Algorithm 2 that discovers new clusters in sequence (marked with gray circle) by applying *findCausalClusters* to the measured variables $\mathbf{X}_{\mathcal{G}}$ generated from the structure in Figure 1(a). Specifically, we first set $k = 1$ and the active set is $\mathcal{S} = \{X_1, \cdots, X_{16}\}$ and $\tilde{\mathcal{S}} = \mathcal{S}$, and we can find the clusters in (a), and no further cluster can be found with $k = 1$. Then we increase k to 2 with the active set $\mathcal{S} = \{\{L_6\}, \{L_7\}, X_6, \cdots, X_{16}\}$ and $\tilde{\mathcal{S}} = \mathcal{S}$, and then we can find the clusters in (b). Then, the active set is $\mathcal{S} = \{\{L_4, L_5\}, \{L_6\}, \{L_7, L_8\}, \{L_9, L_{10}\}\}$ and we set back $k = 1$, and when $\tilde{\mathcal{S}} = \{\{L_4, L_5\}, X_1, \cdots, X_{11}\}$ we find the cluster in (c). Note that when testing the rank over $\{L_4, L_5\}$ against other variables, we use their measured pure descendants in the currently estimated graph instead. The above procedure is repeated to further find the cluster in (d). Finally, when there are no enough variables for testing, we connect the elements in the active variable set: connecting $\{L_2, L_3\}$ to $\{L_7, L_8\}$ in (e).

the case when their parents L_1, L_2, L_3 split the graph into two or more disconnected graphs. This result is formally stated in the following definition and theorem.

Definition 7 (Bond Set)**.** *Consider a set of measured variables* $\mathbf{X} \subseteq \mathbf{X}_{\mathcal{G}}$*, and a minimal set of latent variables* $\mathbf{L} \subseteq \mathbf{L}_{\mathcal{G}}$ *that d-separate* $\mathbf{X}$ *from all other measures* $\mathbf{X}' := \mathbf{X}_{\mathcal{G}} \backslash \mathbf{X}$*. We say that* $\mathbf{X}$ *is a bond set if* $\mathbf{L}$ *also d-separates some partition* $\mathbf{X}_A \subset \mathbf{X}', \mathbf{X}_B \subset \mathbf{X}'$ *from one another.*

From the previous example, we have seen that with the existence of bond sets, *findCausalClusters* may end up with incorrect clusters and covers. The following theorem shows that the presence of bond sets is the only reason for incorrect clusters and covers with *findCausalClusters*.

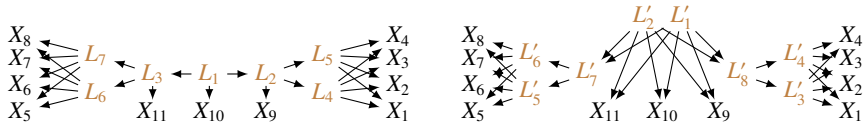
Theorem 3 (Correct Cluster Condition)**.** *Suppose* $\mathcal{G}$ *is an* IL^2H *graph with measured variables* $\mathbf{X}_{\mathcal{G}}$*. Consider the output* $\mathcal{G}'$ *from applying findCausalClusters over* $\mathbf{X}_{\mathcal{G}}$*. If none of the clusters in* $\mathcal{G}'$ *is the bond set in* $\mathcal{G}$*, then all latent atomic covers have been correctly identified.*

However, without access to the true graph $\mathcal{G}$, we are not able to identify which clusters formed from *findCausalClusters* are bond sets. Fortunately, the following properties of the clusters formed over bond sets provide a way to remove bond sets in our discovered graph even without such knowledge, so that the clusters and latent covers can be correctly identified.

Theorem 4 (Correcting Clusters)**.** *Denote by $\mathcal{G}'$ the output from findCausalClusters and by $\mathcal{G}$ the true graph. For a latent atomic cover $\mathbf{L}'$ in $\mathcal{G}'$, if the measured pure descendants of $\mathbf{L}'$ is a bond set in the true graph $\mathcal{G}$, then there exist a set of siblings $\mathbf{S}$ of $\mathbf{L}'$ in $\mathcal{G}'$, a set of children $\mathbf{C}$ of $\mathbf{L}'$, and a set of grandparents $\mathbf{P}$ of $\mathbf{L}'$, such that $\mathcal{M}_{\mathcal{G}'}(\mathbf{S} \cup \mathbf{C} \cup \mathbf{P})$ forms a cluster that is not a bond set in $\mathcal{G}$.*

The above theorem shows that whenever forming a bond set, the siblings and grandparents of the latent cover are the key to correcting the incorrect clusters and covers. Specifically, we will remove such a cover and use its children to form new clusters with the siblings and grandparents (see details in Algorithm 3), and Theorem 4 guarantees that these new clusters will not contain bond sets, and thus, providing correct clusters.

Example 3. *In Figure 3(b), the formed cluster $\{X_9, X_{10}, X_{11}\}$ is the bond set in the true graph in Figure 3(a), and the covers $\{L'_1, L'_2\}$, $\{L'_7\}$, and $\{L'_8\}$ are not correct. Illustration of the algorithm for this example, including refining the incorrect clusters and covers, is given in Appendix A5.7.*



(a) Original graph (b) Output from findCausalClusters

Figure 3: An example where findCausalClusters fails.

Furthermore, with Theorem 4, we give the following rule for correcting the clusters and covers.

Rule 2: For each discovered latent atomic cover $\mathbf{L}$, let $\mathbf{V} = Gp_{\mathcal{G}'}(\mathbf{L}) \bigcup Sib_{\mathcal{G}'}(\mathbf{L}) \bigcup Ch_{\mathcal{G}'}(\mathbf{L})$ and apply *findCausalClusters* to $\mathbf{V}$ to refine the clusters.

Before introducing the detailed refining algorithm *refineClusters* based on Rule 2, we first introduce an operator *makeRoot* that will be used in the algorithm, which will not change the rank deficiency constraints (Lemma 5).

Definition 8 (*makeRoot*)**.** *Give a graph $\mathcal{G}'$ and a latent atomic cover $\mathbf{L}$ in $\mathcal{G}'$. A makeRoot operator of $\mathbf{L}$, denoted by makeRoot($\mathbf{L}$), reorients all outgoing edges of $\mathbf{L}$ to $\mathbf{L}$, such that $\mathbf{L}$ is a root variable.*

Lemma 5 (Rank Invariance)**.** *Denote by $\mathcal{G}'$ the output from findCausalClusters and by $\mathbf{L}$ a latent atomic cover in $\mathcal{G}'$. Then all rank constraints, that are possible to be executed by findCausalClusters prescribed by $\mathcal{G}'$, before and after the operator makeRoot($\mathbf{L}$) are identical.*

Algorithm 3 introduces the procedure of refining clusters based on Rule 2. Starting with the output $\mathcal{G}'$ from *findCausalClusters*, we proceed in a breadth-first search from the root variable (lines 1-3, 5). Because the root is trivially not over a bond set, Theorem 4 guarantees that for each child of the root, we will form new clusters that are not over bond sets. For each cover $\mathbf{L}$, if it has only one child, we will recursively consider the children of its child instead, since if $\mathbf{L}$ is over a bond set then its single child will also be. We add the first set of children into the search set $\mathbf{V}$ (line 4). We construct the search set (line 6) and remove the covers we are refining (lines 7-8). Finally, we make $\mathbf{L}$ the root (line 8) and conduct the search to form new clusters (line 9). The algorithm ends after refining every latent cover in $\mathcal{G}'$. With this refining procedure, we will derive the correct latent covers for the graph in Figure 3(a).

3.3 Phase III: Refining Edges

With *findCausalClusters* and *refineClusters*, the output $\mathcal{G}'$ correctly identifies the latent variables. Moreover, $\mathcal{G}'$ correctly identifies the following d-separation: for every $\mathbf{L} \in \mathbf{L}_{\mathcal{G}'}$, its parents $Pa_{\mathcal{G}'}(\mathbf{L})$ d-separates $\mathbf{L}$ and its descendants from the ancestors of $Pa_{\mathcal{G}'}(\mathbf{L})$, and thus, there cannot be any edges from each $\mathbf{L}$ to any of its ancestors beyond its own parents. However, $\mathcal{G}'$ may still have incorrect edges *locally*. Specifically, each time we create an atomic cover $\mathbf{L}$ in *findCausalClusters*, the implicit assumption is that each of its children is conditionally independent of the other children given the parents $\mathbf{L}$. However, this assumption is not necessarily true, as (i) the children of $\mathbf{L}$ may be directly connected to one another, and (ii) $\mathbf{L}$ may only be directly connected to a subset of its children. In other words, previous steps did not consider condition independence relationships across clusters. Hence, we need a further step to correct edges over each $\mathbf{L}$ and its children.

For example, consider the true graph in Figure 4(a), which ends up with the graph in Figure 4(b) with the first two phases. However, note that the first two phases did not consider the d-separation between L'_2 and L'_3, and thus the edges among L'_1, L'_2, L'_3, L'_4 may not be correct, including the v structure. In

Algorithm 3: Phase II: refineClusters

Input : Output graph $\mathcal{G}'$ from Phase I
Output : Refined graph $\mathcal{G}'$

`refineClusters` ($\mathcal{G}'$):

1 Let Q be an empty queue; Q.enqueue($PCh_{\mathcal{G}'}(Root(\mathcal{G}'))$);
2 **repeat**
3 $\quad$ $\mathbf{L} \leftarrow Q$.dequeue();
4 $\quad$ **while** $Ch_{\mathcal{G}'}(\mathbf{L})$ *is a single atomic cover* **do** remove $Ch_{\mathcal{G}'}(\mathbf{L})$ and add $Ch_{\mathcal{G}'}(Ch_{\mathcal{G}'}(\mathbf{L}))$ as children of $\mathbf{L}$;
5 $\quad$ **for** *each atomic cover* $C \in Ch_{\mathcal{G}'}(\mathbf{L})$ **do** Q.enqueue($\mathbf{C}$) ;
6 $\quad$ $\mathbf{V} \leftarrow Gp_{\mathcal{G}'}(\mathbf{L}) \bigcup Sib_{\mathcal{G}'}(\mathbf{L}) \bigcup Ch_{\mathcal{G}'}(\mathbf{L})$; `// variables for finding new clusters`
7 $\quad$ $\mathbf{R} \leftarrow \mathbf{L} \bigcup Pa_{\mathcal{G}}(\mathbf{L})$; `// atomic covers to be removed`
8 $\quad$ $\mathcal{G}' \leftarrow$ `makeRoot`$_{\mathcal{G}'}(\mathbf{L})$; remove $\mathbf{R}$ and all adjoining edges from $\mathcal{G}'$;
9 $\quad$ $\mathcal{G}'' \leftarrow$ `findCausalClusters`($\mathbf{V}$); update $\mathcal{G}'$ with new clusters from $\mathcal{G}''$; `// find new clusters`
10 **until** *Q is empty*;
11 **return** $\mathcal{G}'$

particular, in the true graph, L_1 d-separates L_2 from L_3, while $\{L_1, L_4\}$ does not d-separate L_2 from L_3, but these d-separations are not reflected in the discovered $\mathcal{G}'$ in the first two phases. Thus, we need to refine the edges. To this end, we first set L'_1, L'_2, L'_3 and L'_2, L'_3, L'_4 to be fully connected, and consider testing $\mathcal{A} = \{L'_2, X_1\}$ against $\mathcal{B} = \{L'_3, X_2\}$, where we partition the children of L'_1 into two sets and put them in $\mathcal{A}$ and $\mathcal{B}$, respectively. By doing so, we force L'_1 in $\mathcal{G}'$ to be in the separating set. Since $\texttt{rank}(\Sigma_{\mathcal{A},\mathcal{B}}) = 1$, it implies that no other variable is in the separating set, and therefore we can conclude that L'_1 d-separates L'_2 from L'_3. This principle is characterized in the following lemma.

Lemma 6 (Cross-Cover Test)**.** *Given a set of variables $\mathcal{S}$, consider two latent atomic covers $\mathbf{L}_A, \mathbf{L}_B \in \mathcal{S}$, and a potential separating set $\mathbf{C} = \{\mathbf{L}_{C_i}\} \subseteq \mathcal{S} \backslash \{\mathbf{L}_A, \mathbf{L}_B\}$. For each $\mathbf{L}_{C_i}$, consider $\mathbf{C}_i^A, \mathbf{C}_i^B \subseteq PCh(\mathbf{L}_{C_i})$ with $\mathbf{C}_i^A, \mathbf{C}_i^B \neq \emptyset$ and $\mathbf{C}_i^A \cap \mathbf{C}_i^B = \emptyset$, and denote the cardinality $k_i^A := min(|\mathbf{L}_{C_i}|, |\mathbf{C}_i^A|)$, $k_i^B := min(|\mathbf{L}_{C_i}|, |\mathbf{C}_i^B|)$, respectively. Then there is no edge between $\mathbf{L}_A$ and $\mathbf{L}_B$ if and only if there exists a separating set $\mathbf{C}$ such that $\texttt{rank}(\Sigma_{\mathcal{A},\mathcal{B}}) < min(|\mathbf{L}_A| + \sum_i k_i^A, |\mathbf{L}_B| + \sum_i k_i^B)$, where $\mathcal{A} = \{\mathbf{L}_A, \mathbf{C}_1^A, \mathbf{C}_2^A, ...\}$ and $\mathcal{B} = \{\mathbf{L}_B, \mathbf{C}_1^B, \mathbf{C}_2^B, ...\}$. In this case, we say that $\mathbf{C}$ satisfies the cross-cover test of $\mathbf{L}_A$ against $\mathbf{L}_B$.*

Note that in order to find rank deficiency when performing the cross-cover test, $\mathbf{L}_{C_i}$ needs to satisfy the third condition in Condition 1. Based on Lemma 6, we use the following rule to refine the edges.

Rule 3: For a pair of latent covers $(\mathbf{L}_A, \mathbf{L}_B)$, let $\mathcal{A} \leftarrow \{\mathbf{L}_A, \mathbf{C}_1^A, \mathbf{C}_2^A, ...\}$ and $\mathcal{B} \leftarrow \{\mathbf{L}_B, \mathbf{C}_1^B, \mathbf{C}_2^B, ...\}$. If there exists such $\mathcal{A}, \mathcal{B}$ such that $\texttt{rank}_{\mathcal{G}'}(\Sigma_{\mathcal{A},\mathcal{B}})$ is rank deficient, then remove all edges between $\mathbf{L}_A, \mathbf{L}_B$ in $\mathcal{G}'$.

Furthermore, with Rule 3, Algorithm 4a (*CrossCoverTest*) in Appendix A gives the procedure of refining the edges over a set of latent variables $\mathcal{S}$ to correct the causal skeleton.

Furthermore, we are going to identify the v structures among latent atomic covers. In the output $\mathcal{G}'$ from phase II, for any $\mathbf{L}'$ with a child $\mathbf{C}'_i$ and parent $\mathbf{P}'$, it is not possible to have a collider $\mathbf{C}'_i \rightarrow \mathbf{L}' \leftarrow \mathbf{P}'$ in the ground-truth graph, because in this case the cluster would not have been rank deficient. Therefore, the only v structures in the graph are amongst the variables in $\mathbf{L}' \cup PCh_{\mathcal{G}'}(\mathbf{L}')$, and similar to *crossCoverTest*, we only need to test for v structures locally. Continue to consider the example in Figure 4. The edge between L'_2 and L'_3 is missing because the rank over $\{L'_2, X_1\}$ and $\{L'_3, X_2\}$ is 1, implying that L'_1 d-separates L'_2 from L'_3, as is done in *crossCoverTest*. Now, since $L'_2 - L'_4 - L'_3$ forms an unshielded triplet, we want to test if a collider exists at L'_4. We find that the rank over $\mathcal{A} = \{L'_2, L'_4, X_1\}$ and $\mathcal{B} = \{L'_3, X_2\}$ is $2 > 1$, and the rank over $\mathcal{A} = \{L'_2, X_1\}$ and $\mathcal{B} = \{L'_3, L'_4, X_2\}$ is $2 > 1$, so $L'_2 \rightarrow L'_4 \leftarrow L'_3$. For general cases, the rule for finding v structures are formulated in the following lemma and Algorithm 4b (*findColliders*) in Appendix A.

Lemma 7 (V-Structure Test)**.** *For any unshielded triangle $\mathbf{L}_A - \mathbf{L}_C - \mathbf{L}_B$, let $\mathcal{A}, \mathcal{B}$ be the set of variables in Lemma 6 such that $\Sigma_{\mathcal{A},\mathcal{B}}$ was rank deficient. Let $k = \texttt{rank}(\Sigma_{\mathcal{A},\mathcal{B}})$, $k_1 = \texttt{rank}(\Sigma_{\mathcal{A}\cup\mathbf{L}_C,\mathcal{B}})$, and $k_2 = \texttt{rank}(\Sigma_{\mathcal{A},\mathcal{B}\cup\mathbf{L}_C})$. Then, $\mathbf{L}_A \rightarrow \mathbf{L}_B \leftarrow \mathbf{L}_C$ if and only if $k < \min(k_1, k_2)$.*

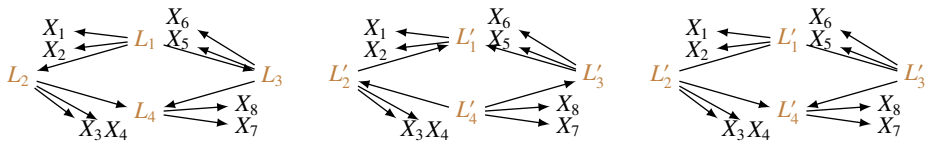
As mentioned above, we only need to perform cross-cover test and v-structure test locally in the estimated graph $\mathcal{G}'$. Algorithm 4 (Phase III: *refineEdges*) combines the search procedure of the two components together with the output $\mathcal{G}'$ from *refineClusters* as the input. Specifically, for each latent cover $\mathbf{L}'$ in $\mathcal{G}'$, we only need to consider testing for edges amongst $\mathbf{L}'$ and its children $\mathbf{C} = Ch_{\mathcal{G}'}(\mathbf{L}')$. Thus, we perform a depth-first traversal of the output graph $\mathcal{G}'$ starting from $Root(\mathcal{G}')$ (lines 1-3), and

apply the cross-cover test at each $\mathbf{L}$ to refine the skeleton of $\mathcal{G}'$ (lines 4-7, 11) and the collider test to identify the v structures (lines 10, 11). After determining v-structures, we can find more directions by applying Meek's rule (line 12), analogous to that in the PC algorithm [Spirtes et al., 2000].

Algorithm 4: Phase III: refineEdges

Input : Learned graph $\mathcal{G}'$ from phase II
Output : Markov equivalence class $\mathcal{G}'$

`refineEdges` ($\mathcal{G}'$):
1 **foreach** *latent atomic cover* $\mathbf{L}'$ *in* $\mathcal{G}'$ **do**
2 | **if** $\mathbf{L}'$ *does not have latent children* **then return** $\mathcal{G}'$;
3 | **foreach** *latent child* $\mathbf{C}_i$ *of* $\mathbf{L}'$ **do** $\mathcal{G}' \leftarrow$ `refineEdges`$(\mathcal{G}', \mathbf{C}_i)$;
4 | **if** $\mathbf{C} := PCh_{\mathcal{G}'}(L')$ *is a single latent cover* **then** $\mathcal{S} \leftarrow \mathbf{L}' \cup \mathbf{C} \cup PCh_{\mathcal{G}'}(\mathbf{C})$;
5 | **else** $\mathcal{S} \leftarrow \mathbf{L}' \cup \mathbf{C}$;
6 | $\mathcal{G}'' \leftarrow$ `makeRoot`$_{\mathcal{G}'}(\mathbf{L}')$ and remove all edges amongst $\mathcal{S}$ in $\mathcal{G}''$; `// temp graph`
7 | Edgeset $\mathcal{E} \leftarrow$ `crossCoverTest`$(\mathcal{S}, \mathcal{G}'')$;
8 | **if** *no conditional independencies found* **then return** $\mathcal{G}'$;
9 | **else**
10 | | Collider set $\mathcal{C} \leftarrow$ `findColliders`$(\mathcal{S}, \mathcal{E}, \mathcal{G}'')$;
11 | | in $\mathcal{G}'$, remove all edges amongst $\mathcal{S}$ and use $\mathcal{E}, \mathcal{C}$ to connect variables in $\mathcal{S}$;
12 | | apply Meek's rule to $\mathcal{G}'$;
13 convert $\mathcal{G}'$ to its Markov equivalence class;
14 **return** $\mathcal{G}'$



(a) An IL^{2}H graph with v structure. (b) Possible Output from Phase II. (c) Output from phase III.

Figure 4: Illustrations for refineEdges.

4 Theoretical Results

In this section, we show the correctness of the algorithms proposed in Section 3. In particular, by making use of the rank constraints of only the measured variables, the proposed algorithms output the correct Markov equivalence class of the IL^{2}H graph asymptotically, under the *minimal-graph operator* and *skeleton operator*, with their definitions given below.

Definition 9 (Markov Equivalence Class of IL^{2}H graphs)**.** *Two IL^{2}H graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ are in the same Markov equivalence class, denoted by $\mathcal{G}_1 \approx \mathcal{G}_2$, if and only (1) they have the same set of variables (both measured and latent variables), (2) have the same causal skeleton, and (3) have the same V-structures $\mathbf{L}_i \rightarrow \mathbf{L}_k \leftarrow \mathbf{L}_j$, where $\mathbf{L}_i, \mathbf{L}_j, \mathbf{L}_k$ represent latent atomic covers.*

Definition 10 (Minimal-Graph Operator)**.** *Suppose $\mathcal{G}$ is an IL^{2}H graph. For every latent atomic cover $\mathbf{L}$ in $\mathcal{G}$, merge $\mathbf{L}$ to its parents $\mathbf{P}$ if the following conditions hold: (1) $\mathbf{L}$ is the pure children of $\mathbf{P}$, (2) $|\mathbf{L}| = |\mathbf{P}|$, and (3) the pure children of $\mathbf{L}$ form one latent atomic cover, or the siblings of $\mathbf{L}$ form one latent atomic cover. We call such operator the minimal-graph operator and denote it by $O_{min}(\mathcal{G})$.*

Definition 11 (Skeleton Operator)**.** *Suppose $\mathcal{G}$ is an IL^{2}H graph. A skeleton operator of $\mathcal{G}$, denoted by $O_s(\mathcal{G})$ is defined as follows: for any latent atomic cover $\mathbf{L}$, draw an edge from $l_j \in \mathbf{L}$ to $c_k \in PCh_{\mathcal{G}}(\mathbf{L}_i)$, if l_j and c_k are not directly connected in $\mathcal{G}$.*

Note that the minimal-graph operator and the skeleton operator will not change the rank constraints, or in other words, graphs before and after applying the operators are indistinguishable with rank constraints. This result is shown in the following lemma.

Lemma 8. *Suppose $\mathcal{G}$ is an IL^{2}H graph. The rank constraints are invariant with the minimal-graph operator and the skeleton operator; that is, $\mathcal{G}$ and $O_{skeleton}(O_{min}(\mathcal{G}))$ are rank equivalent.*

These two operators have already been achieved in Algorithm 1, so the output of the algorithm is the rank-equivalent graph with the minimal number of latent atomic covers.

We next proceed to show that phases I-III will output a graph $\mathcal{G}'$ such that $\mathcal{G}'$ will be in the same Markov equivalence class as $O_{min}(O_s(\mathcal{G}))$, denoted by $\mathcal{G}' \approx \mathcal{G}$. We have already shown earlier in

Theorem 3 that *findCausalClusters* gives correct latent covers when there is no bond set. Furthermore, Theorem 4 shows that even in the presence of bond sets, Phase II *refineClusters* is able to refine clusters into those without bond sets. Therefore, Phase I-II can correctly identify the latent atomic covers of $O_{min}(\mathcal{G})$, which is given in the following theorem.

Theorem 9 (Identifiability of Latent Variables)**.** *Suppose $\mathcal{G}$ is an IL^{2}H graph with measured variables $\mathbf{X}_{\mathcal{G}}$. Phases I-II in Algorithm 1 over $\mathbf{X}_{\mathcal{G}}$ can asymptotically identify the latent atomic covers of $O_{min}(\mathcal{G})$, with only the first two conditions in Condition 1.*

Moreover, Phase III *refineEdges* further guarantees correct skeletons and v structures. Therefore, the following theorem shows that Algorithm 1, which includes Phases I-III, can asymptotically identify the Markov equivalence class, up to the skeleton operator and the minimal-graph operator.

Theorem 10 (Identifiability of Causal Graph)**.** *Suppose $\mathcal{G}$ is an IL^{2}H graph with measured variables $\mathbf{X}_{\mathcal{G}}$. Algorithm 1, including Phases I-III, over $\mathbf{X}_{\mathcal{G}}$ can asymptotically identify the Markov equivalence class of $O_{min}(O_s(\mathcal{G}))$.*

5 Experimental Results

We applied the proposed algorithm to synthetic data to learn the latent hierarchical causal graph. Specifically, we considered different types of latent graphs and different sample sizes. The causal strength was generated uniformly from $[-5, -0.5] \cup [0.5, 5]$, and the noise term was randomly chosen from a normal distribution with noise variance uniformly sampled from $[1, 5]$. To the best of our knowledge, this is the first algorithm that can identify such general latent hierarchical structures, so to fairly compare with other methods, besides general IL^{2}H graphs, we also considered tree structures and measurement models. We compared the proposed method with tree-based method–Chow-Liu Recursive Grouping (CLRG) [Choi et al., 2011], as well as measurement-model-based methods, including FOFC [Kummerfeld and Ramsey, 2016] and GIN [Xie et al., 2020].

Table 1 gives the estimation results evaluated on three types of latent graphs: IL^{2}H graphs, tree structures, and measurement models. The performance is measured by the percentage of correctly identified causal clusters over only measured variables. Our method gives the best results on all types of graphs, indicating that it can handle not only the tree-based and measurement-based structures, but also the latent hierarchical structure. The CLRG algorithm does not perform well on tree-based structure because the metric is rather strict–even a single misclustered variable outputs an error. Complete experimental settings and more results, including performance measured by other metrics and the case when the noise terms are uniformly distributed, are given in Appendix B.

Table 1: Performance (mean (standard deviation)) on learning different types of latent graphs.

	IL^{2}H				Tree				Measurement model			
Sample size	Ours	CLRG	FOFC	GIN	Ours	CLRG	FOFC	GIN	Ours	CLRG	FOFC	GIN
2k	0.70 (0.22)	0	0.12	0.35	0.89 (0.12)	0	0.38	0.13	0.92 (0.08)	0	0.38	0.3
5k	0.83 (0.15)	0	0.17	0.40	1.0 (0.00)	0	0.75	0.23	1.0 (0.00)	0	0.65	0.7
10k	0.86 (0.13)	0	0.29	0.44	1.0 (0.00)	0.13	0.87	0.50	1.0 (0.00)	0	1.0	0.7

6 Conclusions and Future Work

In this paper, we formulated a specific type of latent hierarchical causal model and proposed a method to identify its graph by making use of rank deficiency constraints. Theoretically, we show that the proposed algorithm can find the correct Markov equivalence class of the whole graph asymptotically under mild restrictions of the graph structure. For more general graphs, only using the second-order statistics may result in a rank equivalence class that contains multiple DAGs, so how to further leverage high-order statistics to distinguish between causal graphs within the equivalence class will be our future work. Other future research directions include allowing nonlinear causal relationships and allowing measured variables to cause latent variables (existing techniques, e.g., Adams et al. [2021], Squires et al. [2022], may help to mitigate this issue).

Acknowledgement

BH would like to acknowledge the support of Apple Scholarship. Kun Zhang was partially supported by the National Institutes of Health (NIH) under Contract R01HL159805, by the NSF-Convergence Accelerator Track-D award #2134901, by a grant from Apple Inc., and by a grant from KDDI Research Inc. FX acknowledges the support by National Natural Science Foundation of China (NSFC 62006051).

References

Peter Spirtes, Clark N Glymour, and Richard Scheines. *Causation, Prediction, and Search*. MIT press, 2000.

Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, New York, NY, USA, 2000. ISBN 0-521-77362-8.

Diego Colombo, Marloes H Maathuis, Markus Kalisch, and Thomas S Richardson. Learning high-dimensional directed acyclic graphs with latent and selection variables. *The Annals of Statistics*, pages 294–321, 2012.

Sina Akbari, Ehsan Mokhtarian, AmirEmad Ghassami, and Negar Kiyavash. Recursive causal structure learning in the presence of latent variables and selection bias. In *Advances in Neural Information Processing Systems*, volume 34, pages 10119–10130, 2021.

V. Chandrasekaran, S. Sanghavi, P. A. Parrilo, and A. S. Willsky. Rank-sparsity incoherence for matrix decomposition. *SIAM Journal on Optimization*, 21(2):572–596, 2011.

V. Chandrasekaran, P. A. Parrilo, and A. S. Willsky. Latent variable graphical model selection via convex optimization. *Annals of Statistics*, 40(4):1935–1967, 2012.

Benjamin Frot, Preetam Nandy, and Marloes H Maathuis. Robust causal structure learning with some hidden variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 81 (3):459–487, 2019.

Patrik O Hoyer, Shohei Shimizu, Antti J Kerminen, and Markus Palviainen. Estimation of causal effects using linear non-gaussian causal models with hidden variables. *International Journal of Approximate Reasoning*, 49(2):362–378, 2008.

Saber Salehkaleybar, AmirEmad Ghassami, Negar Kiyavash, and Kun Zhang. Learning linear non-gaussian causal models in the presence of latent variables. *Journal of Machine Learning Research*, 21(39):1–24, 2020.

Ricardo Silva, Richard Scheine, Clark Glymour, and Peter Spirtes. Learning the structure of linear latent variable models. *Journal of Machine Learning Research*, 7(Feb):191–246, 2006.

Erich Kummerfeld and Joseph Ramsey. Causal clustering for 1-factor measurement models. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1655–1664. ACM, 2016.

Shohei Shimizu, Patrik O Hoyer, and Aapo Hyvärinen. Estimation of linear non-gaussian acyclic models for latent factors. *Neurocomputing*, 72(7-9):2024–2027, 2009.

Ruichu Cai, Feng Xie, Clark Glymour, Zhifeng Hao, and Kun Zhang. Triad constraints for learning causal structure of latent variables. In *Advances in Neural Information Processing Systems*, pages 12863–12872, 2019.

Feng Xie, Ruichu Cai, Biwei Huang, Clark Glymour, Zhifeng Hao, and Kun Zhang. Generalized independent noise condition for estimating latent variable causal graphs. In *Advances in Neural Information Processing Systems*, pages 14891–14902, 2020.

Jeffrey Adams, Niels Hansen, and Kun Zhang. Identification of partially observed linear causal models: Graphical conditions for the non-gaussian and heterogeneous cases. *Advances in Neural Information Processing Systems*, 34, 2021.

Zhengming Chen, Feng Xie, Jie Qiao, Zhifeng Hao, Kun Zhang, and Ruichu Cai. Identification of linear latent variable model with arbitrary distribution. In *Proceedings 36th AAAI Conference on Artificial Intelligence (AAAI)*, 2022.

Animashree Anandkumar, Daniel Hsu, Adel Javanmard, and Sham Kakade. Learning linear bayesian networks with latent variables. In *International Conference on Machine Learning*, pages 249–257, 2013.

Ruifei Cui, Perry Groot, Moritz Schauer, and Tom Heskes. Learning the causal structure of copula models with latent variables. In *UAI*, pages 188–197. AUAI Press, 2018.

Bohdan Kivva, Goutham Rajendran, Pradeep Ravikumar, and Bryon Aragam. Learning latent causal graphs via mixture oracles. *Advances in Neural Information Processing Systems*, 34, 2021.

Yan Zeng, Shohei Shimizu, Ruichu Cai, Feng Xie, Michio Yamamoto, and Zhifeng Hao. Causal discovery with multi-domain lingam for latent factors. In *IJCAI*, pages 2097–2103, 2021.

Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988. ISBN 1558604790.

Nevin L Zhang. Hierarchical latent class models for cluster analysis. *The Journal of Machine Learning Research*, 5:697–723, 2004.

Myung Jin Choi, Vincent Y.F. Tan, Animashree Anandkumar, and Alan S. Willsky. Learning latent tree graphical models. *Journal of Machine Learning Research*, 12(49):1771–1812, 2011. URL `http://jmlr.org/papers/v12/choi11b.html`.

Mathias Drton, Shaowei Lin, Luca Weihs, and Piotr Zwiernik. Marginal likelihood and model selection for gaussian latent tree and forest models. *Bernoulli*, 23(2):1202–1232, 2017.

F. Xie, B. Huang, Z. Chen, Y. He, Z. Geng, and K. Zhang. Identification of linear non-gaussian latent hierarchical structure. In *ICML*, 2022.

Peter Spirtes. Calculation of entailed rank constraints in partially non-linear and cyclic models. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*, pages 606–615. AUAI Press, 2013.

T. W. Anderson. *An Introduction to Multivariate Statistical Analysis. 2nd ed.* John Wiley & Sons, 1984.

Seth Sullivant, Kelli Talaska, and Jan Draisma. Trek separation for Gaussian graphical models. *The Annals of Statistics*, 38(3):1665 – 1685, 2010. doi: 10.1214/09-AOS760. URL `https://doi.org/10.1214/09-AOS760`.

C. Squires, A. Yun, E. Nichani, R. Agrawal, and C. Uhler. Causal structure discovery between clusters of nodes induced by latent factors. In *CLeaR*, 2022.

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] We proposed a method to identify latent hierarchical causal graphs by making use of rank deficiency constraints. Theoretically, we show that the proposed algorithm can find the correct Markov equivalence class asymptotically for more general graph structures, compared to previous methods.
   (b) Did you describe the limitations of your work? [Yes] See Section 6 on future work. One limitation is that here we only allowed linear models, and we will extend it to allow nonlinear causal relations as well. Another limitation is that here we did not consider direct causal edges among the measurement variables. This is because, for such cases, only using the second-order statistics may only identified to the rank equivalence class, which may contain multiple DAGs, so how to further leverage high-order statistics to distinguish between causal graphs within the equivalence class will be our future work.
   (c) Did you discuss any potential negative societal impacts of your work? [No] We do not think there is any negative societal impact at the current moment.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes] We have read the ethics review guidelines.
2. If you are including theoretical results...

(a) Did you state the full set of assumptions of all theoretical results? [Yes] We clearly stated the assumptions in Section 2.

(b) Did you include complete proofs of all theoretical results? [Yes] We gave complete proofs of all theoretical results in Appendix.

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] We gave the detailed experimental setting in Appendix. Code will be made public.

(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] We specified all the details in Appendix.

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Section B in Appendix.

(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]

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? [N/A]

(b) Did you mention the license of the assets? [N/A]

(c) Did you include any new assets either in the supplemental material or as a URL? [N/A]

(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]

Appendix for "Latent Hierarchical Causal Structure Discovery with Rank Constraints"

Biwei Huang [*1] **Charles Low** [*1], **Feng Xie**[3], **Clark Glymour**[1], **Kun Zhang**[1,2]
[1] Carnegie Mellon University
[2] Mohamed bin Zayed University of Artificial Intelligence
[3] Beijing Technology and Business University, China
{bwei.huang, charleslow88, xiefeng009}@gmail.com,
cg09@andrew.cmu.edu, kunz1@cmu.edu

Organization of Appendix:

- Section A: Algorithms that are not listed in the main text because of page limits
- Section B: Complete experimental results
- Section C: Proofs
- Section D: Related work
- Section E: More explanations on identifiability conditions, algorithms, and theorems
- Section F: Illustrative examples of the entire algorithm

Pa: Parents	Sib: Siblings	$\mathbf{V}_{\mathcal{G}}$: All variables in graph $\mathcal{G}$
PCh: Pure children	$\mathcal{M}$: Measured pure descendants	$\mathbf{X}$: A set of measured variables
PDe: Pure descendants	$\mathbf{X}_{\mathcal{G}}$: All measured variables in graph $\mathcal{G}$	$\mathbf{L}$: A set of latent variables
Gp: Grandparents	$\mathbf{L}_{\mathcal{G}}$: All latent variables in graph $\mathcal{G}$	$\mathbf{V}$: A set of variables

Table 3: Complete graphical notations used in the paper.

A Algorithms on *crossCoverTest* and *findColliders*

Algorithm 4a: (*CrossCoverTest*) gives the procedure of refining the edges over a set of latent variables $\mathcal{S}$. It first fully connects the latent covers in $\mathcal{S}$ (line 2). Then for every pair of latent covers, it performs the cross-cover test (lines 3-17). If rank deficiency is found, then remove the corresponding edges (lines 10-12).

Algorithm 4b: (*findColliders*) gives the procedure of finding v structures. For every unshielded triangle $\mathbf{L}_1 - \mathbf{L}_3 - \mathbf{L}_2$, it performs v structure tests and compares with the rank which does not involve $\mathbf{L}_3$ (lines 22-23). If the rank that involves $\mathbf{L}_3$ is larger, then the unshielded triangle forms a v structure (lines 24-25).

B Complete Experimental Results

We applied the proposed algorithm to synthetic data to learn the latent hierarchical causal graph. Specifically, we considered different types of latent graphs and different sample sizes (with $N = 2k, 5k, 10k$). The causal strength was generated uniformly from $[-5, -0.5] \cup [0.5, 5]$, and the noise term either follows a Gaussian distribution (with noise variance uniformly sampled from $[1, 5]$) or a uniform distribution $\mathcal{U}(-2, 2)$.

[*]These authors contributed equally to this work.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

Algorithm 4a: crossCoverTest

Input : A set of latent variables $\mathcal{S}$, currently learned graph $\mathcal{G}'$
Output : An edgeset $\mathcal{E}$ among the variables

`crossCoverTest` ($\mathcal{S}$, $\mathcal{G}'$):

$\mathcal{E} \leftarrow \emptyset$;
add undirected edges between every pair of latent atomic covers in $\mathcal{S}$ to $\mathcal{E}$;
foreach *pair of latent atomic covers* $\mathbf{L}_A, \mathbf{L}_B \in \mathcal{S}$ **do**
  k = 0
  repeat
    repeat
      draw a potential separating set of k atomic covers $\mathbf{C} = \{\mathbf{C}_1, \mathbf{C}_2, \cdots, \mathbf{C}_k\} \subseteq \mathcal{S} \backslash \{\mathbf{L}_A, \mathbf{L}_B\}$;
        foreach *atomic cover* $\mathbf{C}_i \in \mathbf{C}$ **do**
          partition $PCh_{\mathcal{G}'}(\mathbf{C}_i)$ into $\mathbf{C}_i^A, \mathbf{C}_i^B$; `// Remark on clever choice`
      $\mathcal{A} \leftarrow \{\mathbf{L}_A, \mathbf{C}_1^A, \mathbf{C}_2^A, ...\}$ and $\mathcal{B} \leftarrow \{\mathbf{L}_B, \mathbf{C}_1^B, \mathbf{C}_2^B, ...\}$;
      if *there exists such* $\mathcal{A}, \mathcal{B}$ *such that* $\texttt{rank}_{\mathcal{G}'}(\Sigma_{\mathcal{A},\mathcal{B}})$ *is rank deficient* **then**
        remove all edges between $\mathbf{L}_A, \mathbf{L}_B$ in $\mathcal{E}$;
        break;
    until *all sets* $\mathbf{C}$ *with k atomic covers tested*;
    if *rank deficiency found* **then**
      break;
    k += 1;
  until $k >$ *number of variables in* $\mathcal{S} \backslash \{\mathbf{L}_A, \mathbf{L}_B\}$;
return *Edgeset* $\mathcal{E}$

Algorithm 4b: findColliders

Input : A set of latent variables $\mathcal{S}$, edgeset $\mathcal{E}$, currently learned graph $\mathcal{G}'$
Output : A set of colliders $\mathcal{C}$

`findColliders` ($\mathcal{S}$, $\mathcal{E}$, $\mathcal{G}'$):

Collider set $\mathcal{C} \leftarrow \emptyset$;
foreach *unshielded triangle* $\mathbf{L}_1 - \mathbf{L}_3 - \mathbf{L}_2$ *in* $\mathcal{S}$ *based on* $\mathcal{E}$ **do**
  let $\mathcal{A}, \mathcal{B}$ be the set of variables in Algorithm 4a: such that $\Sigma_{\mathcal{A},\mathcal{B}}$ was rank deficient with rank k;
  $k_1 \leftarrow \texttt{rank}_{\mathcal{G}'}(\Sigma_{\mathcal{A} \cup \mathbf{L}_3, \mathcal{B}})$;
  $k_2 \leftarrow \texttt{rank}_{\mathcal{G}'}(\Sigma_{\mathcal{A}, \mathcal{B} \cup \mathbf{L}_3})$;
  if $k < \min(k_1, k_2)$ **then**
    add collider $\mathbf{L}_1 \rightarrow \mathbf{L}_3 \leftarrow \mathbf{L}_2$ to $\mathcal{C}$;
return *Collider set* $\mathcal{C}$

To the best of our knowledge, this is the first algorithm that can identify such general latent hierarchical structures, so to fairly compare with other methods, besides general IL^{2}H graphs (see Figure 7), we also considered tree structures (see Figure 5) and measurement models (see Figure 6). We compared the proposed method with the tree-based method–Chow-Liu Recursive Grouping (CLRG) [Choi et al., 2011], as well as measurement-model-based methods, including FOFC [Kummerfeld and Ramsey, 2016] and GIN Xie et al. [2020].

We used the following metrics to evaluate the performance:

- *Causal cluster recovery rate over measured variables (metric 1)*: measured by the percentage of correctly identified causal clusters over measured variables, with

$$m_1 = \frac{\text{correctly found \# clusters over measured variables}}{\text{total \# clusters over measured variables}}.$$

- *Causal cluster recovery rate over all variables (metric 2)*, measured by the percentage of correctly identified causal clusters over all variables, with

$$m_2 = \frac{\text{correctly found \# clusters over all variables}}{\text{total \# clusters over all variables}}.$$

- *Percentage differences between estimated and true adjacency matrices (metric 3)*, with

$$m_3 = \sum_{i,j} (Adj_{\mathcal{G}}(i,j) \sim= Adj_{\mathcal{G}'}(i,j))/((n_{\mathbf{X}} + n_{\mathbf{L}})^2 - n_{\mathbf{X}}^2),$$

  where Adj denotes the adjacency matrix, i and j denote the i-th and j-th entry, respectively, and $n_{\mathbf{X}}$ and $n_{\mathbf{L}}$ are the number of measured variables and latent variables, respectively. Note that the indices of the latent variables in the estimated graph may not be aligned to those in the true graph. To remove this ambiguity, we tried all permutations of the latent indices in the estimated graph and used the one which has the smallest difference from the true graph. Moreover, if the estimated number of latent variables is smaller than the true number of latent variables, add extra latent variables to $\mathcal{G}'$ that do not have edges with others. If the estimated number of latent variables is larger than the true number of latent variables, then find a subset of the latent variables in $\mathcal{G}'$ that best aligns the true ones.

It is worth mentioning that how to measure the performance of the estimated latent hierarchical graph is a nontrivial problem and will be further investigated.

The experimental results were reported in Tables 4 and 5, where the noise terms are Gaussian distributed and uniformly distributed, respectively. Our method gives the best results on all types of graphs, indicating that it can handle not only the tree-based and measurement-based structures, but also the latent hierarchical structure. The CLRG algorithm does not perform well on tree-based structure because the first two metrics are rather strict–even a single mis-clustered variable outputs an error.

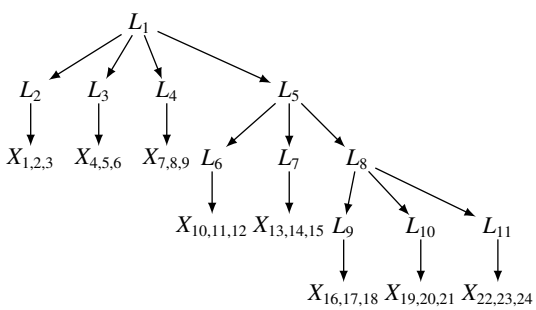


Figure 5: Tree. Note that $X_{i,j,k}$ means X_i, X_j, X_k.

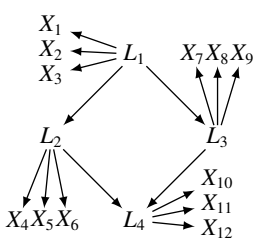


Figure 6: Measurement model.

C Related Work

Identification of causal relationships from observational data, known as *causal discovery*, is attractive for the reason that traditional randomized control trials may be hard or even impossible to do. Most state-of-the-art approaches in causal discovery assume that the measured variables are the underlying causal variables and that no latent confounders influence the measured variables [Spirtes et al., 2000, Chickering, 2002, Shimizu et al., 2006, Hoyer et al., 2009, Zhang and Hyvärinen, 2009]. However, in many real-world problems, this assumption may not hold.

For example, in complex systems, it is usually hard to enumerate and measure all task-related variables, so there may exist latent variables that influence multiple observed variables, the ignorance of which may introduce spurious correlations between measured variables. A more complex scenario is that the variables form a hierarchical structure, where the latent variables may generate latent variables in a hierarchical way, while only the leaf nodes are measured, which is common in real-world scenarios. For instance, in fMRI data analysis, hundreds of thousands of voxels are recorded, where these micro-variables may not be necessary to have clear semantic meaning. Therefore, from the measured voxels, we aim to automatically identify conceptually meaningful functional brain regions of different levels, where the lower level represents simpler functional regions and the higher level represents more abstract and complex functional regions, which thus form a hierarchical structure.

Table 4: Performance (mean (standard deviation)) on learning different types of latent graphs, where noise terms were generated from Gaussian distributions.

		metric 1 ↑			
Algorithm		Ours	CLRG	FOFC	GIN
IL^{2}H	2k	0.70 (0.22)	0.00 (0.00)	0.12 (0.09)	0.35 (0.23)
	5k	0.83 (0.15)	0.00 (0.00)	0.17 (0.10)	0.40 (0.24)
	10k	0.86 (0.13)	0.00 (0.00)	0.29 (0.10)	0.44 (0.21)
Tree	2k	0.89 (0.12)	0.00 (0.00)	0.38 (0.25)	0.13 (0.30)
	5k	1.0 (0.00)	0.00 (0.00)	0.75 (0.23)	0.23 (0.30)
	10k	1.0 (0.00)	0.13 (0.04)	0.87 (0.20)	0.50 (0.00)
Measurement Model	2k	0.92 (0.08)	0.00 (0.00)	0.38 (0.22)	0.30 (0.16)
	5k	1.0 (0.00)	0.00 (0.00)	0.65 (0.32)	0.70 (0.43)
	10k	1.0 (0.00)	0.00 (0.00)	1.0 (0.00)	0.70 (0.43)

		metric 2 ↑			
Algorithm		Ours	CLRG	FOFC	GIN
IL^{2}H	2k	0.60 (0.16)	0.00 (0.00)	0.09 (0.07)	0.26 (0.19)
	5k	0.69 (0.22)	0.00 (0.00)	0.09 (0.07)	0.28 (0.19)
	10k	0.73 (0.17)	0.00 (0.00)	0.12 (0.07)	0.35 (0.17)
Tree	2k	0.79 (0.19)	0.00 (0.00)	0.28 (0.19)	0.09 (0.11)
	5k	0.83 (0.16)	0.00 (0.00)	0.55 (0.17)	0.17 (0.22)
	10k	0.89 (0.09)	0.09 (0.03)	0.63 (0.18)	0.36 (0.20)
Measurement Model	2k	0.92 (0.08)	0.00 (0.00)	0.38 (0.22)	0.30 (0.16)
	5k	1.0 (0.00)	0.00 (0.00)	0.65 (0.32)	0.70 (0.43)
	10k	1.0 (0.00)	0.00 (0.00)	1.0 (0.00)	0.70 (0.43)

		metric 3 ↓			
Algorithm		Ours	CLRG	FOFC	GIN
IL^{2}H	2k	0.11 (0.02)	0.18 (0.04)	0.26 (0.21)	0.16 (0.03)
	5k	0.10 (0.02)	0.18 (0.04)	0.26 (0.21)	0.15 (0.03)
	10k	0.10 (0.02)	0.18 (0.04)	0.20 (0.27)	0.15 (0.03)
Tree	2k	0.02 (0.00)	0.10 (0.00)	0.09 (0.00)	0.15 (0.02)
	5k	0.02 (0.00)	0.10 (0.00)	0.09 (0.00)	0.10 (0.02)
	10k	0.01 (0.00)	0.09 (0.00)	0.08 (0.00)	0.09 (0.00)
Measurement Model	2k	0.02 (0.00)	0.28 (0.00)	0.18 (0.04)	0.30 (0.08)
	5k	0.00 (0.00)	0.28 (0.00)	0.13 (0.05)	0.17 (0.15)
	10k	0.00 (0.00)	0.28 (0.00)	0.00 (0.00)	0.17 (0.15)

Note: ↑ means a higher value is better, and vice versa.

We may also see similar structures in image representation learning–image pixels are dependent, and it seems sensible to consider them as observations generated by multiple-layer hidden concepts.

Previous causal discovery approaches that can handle latent confounders are mainly based on the following criteria.

- Conditional independence constraints. The FCI algorithm [Spirtes et al., 2000], as well as its variants [Colombo et al., 2012, Pearl, 2000, Akbari et al., 2021], makes use of conditional independence tests over observed variables to identify the causal structure over observed variables up to a maximal ancestral graph. This type of methods can handle both linear and nonlinear causal relationships, but the limitation is that there are large indeterminacies in the resulting graph about the existence of an edge, as well as the existence of confounders. In practice, it is often the case that the resulting graph contains many undetermined edges, denoted by $\circ\!\!-\!\!\circ$, where the circle can be either tail or arrow. Moreover, they do not consider the causal relationships among latent variables.

- Tetrad condition. With the Tetrad condition, i.e., the rank constraints of every 2×2 off-diagonal sub-covariance matrix, one is able to locate latent variables and identify the causal skeleton among them in linear-Gaussian models [Silva et al., 2006, Kummerfeld and Ramsey, 2016, Wang, 2020]. These methods assume that each observed variable is influenced by only one latent parent, and

Table 5: Performance (mean (standard deviation)) on learning different types of latent graphs, where noise terms were generated from uniform distributions.

		metric 1 ↑			
Algorithm		Ours	CLRG	FOFC	GIN
IL^{2}H	2k	0.72 (0.19)	0.00 (0.00)	0.10 (0.10)	0.45 (0.18)
	5k	0.87 (0.13)	0.00 (0.00)	0.17 (0.10)	0.46 (0.18)
	10k	0.88 (0.10)	0.00 (0.00)	0.29 (0.10)	0.52 (0.16)
Tree	2k	0.93 (0.07)	0.00 (0.00)	0.40 (0.27)	0.79 (0.22)
	5k	1.0 (0.00)	0.00 (0.00)	0.75 (0.23)	0.99 (0.04)
	10k	1.0 (0.00)	0.00 (0.00)	0.87 (0.20)	0.95 (0.08)
Measurement Model	2k	0.95 (0.06)	0.00 (0.00)	0.25 (0.21)	0.87 (0.20)
	5k	1.0 (0.00)	0.00 (0.00)	0.70 (0.27)	1.00 (0.00)
	10k	1.0 (0.00)	0.00 (0.00)	1.0 (0.00)	1.00 (0.00)

		metric 2 ↑			
Algorithm		Ours	CLRG	FOFC	GIN
IL^{2}H	2k	0.61 (0.20)	0.00 (0.00)	0.07 (0.07)	0.34 (0.15)
	5k	0.71 (0.19)	0.00 (0.00)	0.09 (0.07)	0.35 (0.15)
	10k	0.76 (0.20)	0.00 (0.00)	0.12 (0.07)	0.39 (0.14)
Tree	2k	0.83 (0.12)	0.00 (0.00)	0.29 (0.20)	0.57 (0.16)
	5k	0.85 (0.11)	0.00 (0.00)	0.55 (0.17)	0.72 (0.03)
	10k	0.89 (0.15)	0.00 (0.00)	0.63 (0.18)	0.69 (0.06)
Measurement Model	2k	0.95 (0.06)	0.00 (0.00)	0.25 (0.21)	0.87 (0.20)
	5k	1.0 (0.00)	0.00 (0.00)	0.70 (0.27)	1.00 (0.00)
	10k	1.0 (0.00)	0.00 (0.00)	1.0 (0.00)	1.00 (0.00)

		metric 3 ↓			
Algorithm		Ours	CLRG	FOFC	GIN
IL^{2}H	2k	0.10 (0.02)	0.18 (0.04)	0.23 (0.21)	0.15 (0.03)
	5k	0.10 (0.02)	0.18 (0.04)	0.26 (0.21)	0.15 (0.03)
	10k	0.10 (0.02)	0.18 (0.04)	0.20 (0.27)	0.14 (0.03)
Tree	2k	0.02 (0.00)	0.11 (0.00)	0.10 (0.00)	0.11 (0.00)
	5k	0.01 (0.00)	0.10 (0.00)	0.09 (0.00)	0.11 (0.00)
	10k	0.01 (0.00)	0.10 (0.00)	0.08 (0.00)	0.11 (0.00)
Measurement Model	2k	0.01 (0.00)	0.28 (0.00)	0.25 (0.04)	0.07 (0.05)
	5k	0.00 (0.00)	0.28 (0.00)	0.25 (0.04)	0.07 (0.05)
	10k	0.00 (0.00)	0.28 (0.00)	0.00 (0.00)	0.00 (0.00)

Note: ↑ means a higher value is better, and vice versa.

each latent variable has at least three pure measured children. Moreover, the Tetrad condition can also be used to identify a latent tree structure [Pearl, 1988].

- Matrix decomposition. It has been shown that, under certain conditions, the precision matrix can be decomposed into a low-rank matrix and a sparse matrix, where the low-rank matrix represents the causal structure from latent variables to observed variables and the sparse matrix gives the structural relationships over observed variable. To achieve such decomposition, however, certain assumptions are imposed on the structure [Chandrasekaran et al., 2011, 2012]. A related work [Anandkumar et al., 2013] decomposed the covariance matrix into a low-rank matrix and a diagonal matrix, which requires three times more measured variables than latent variables. [Anandkumar et al., 2013] can also handle multi-level DAGs that some latent variables do not have measured variables as children, but it requires that the underlying graph can be partitioned into multiple levels such that all the edges are between nodes in adjacent layers; the graphs in Figure 1(b) in the main text, and Figures 5-6 and Figures 7(a, b, d) in Appendix are not satisfied.
- Over-complete independent component analysis (ICA)-based methods. Several methods [Shimizu et al., 2009] make use of over-complete ICA to learn the causal structure with latent variables, since it allows more source signals than observed variables. These methods do not consider the causal structure among latent variables and the size of the equivalence class of the identified structure could be large. In addition, in practice, the estimation of over-complete ICA models is easy to

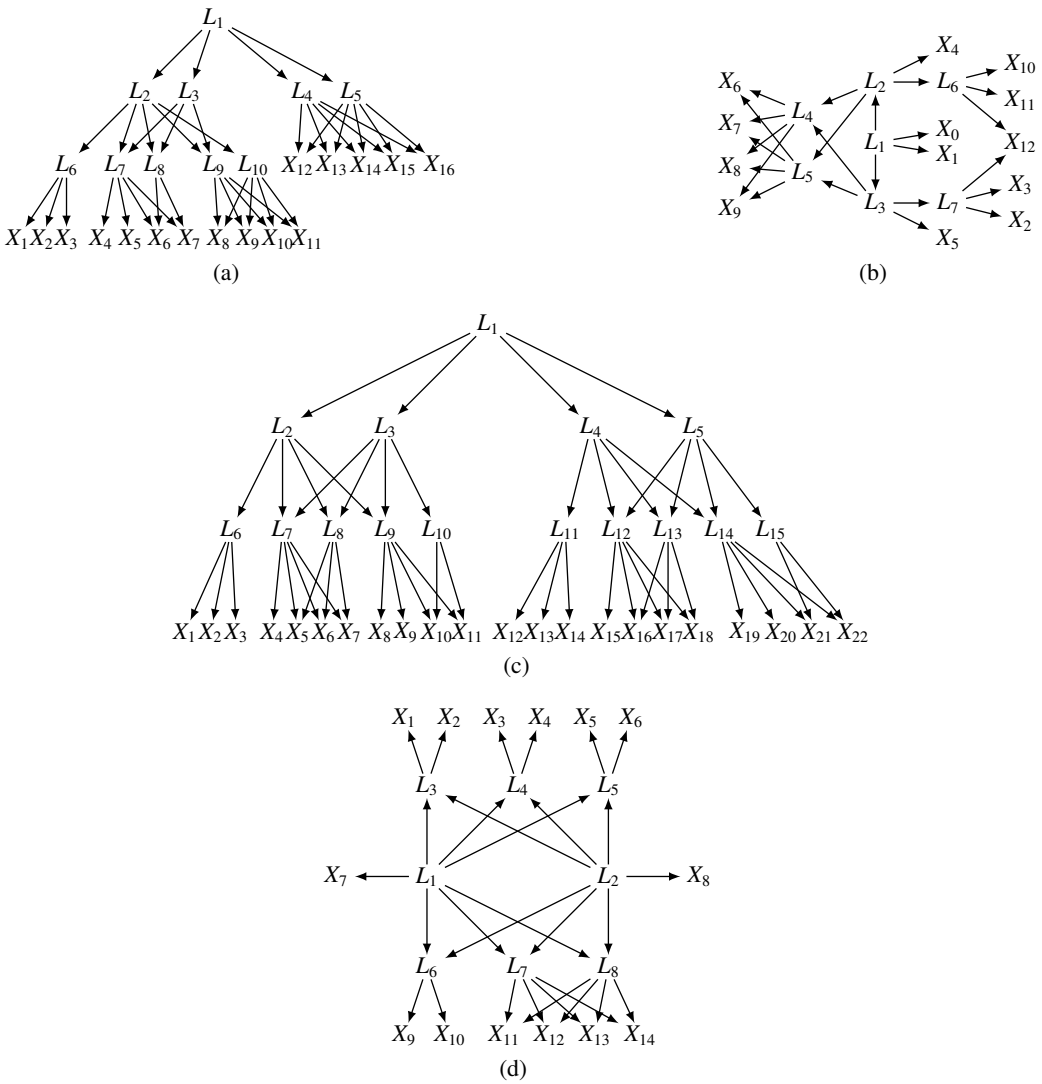


Figure 7: Example IL^{2}H graphs.

get stuck in local optima, unless the underlying sources are very sparse [Entner and Hoyer, 2010, Tashiro et al., 2014].

- Generalized independent noise (GIN) condition. The GIN condition is an extension of the independent noise condition in the existence of latent confounders. It assumes the noise terms are non-Gaussian and leverages higher-order statistics to identify latent structures. In particular, Xie et al. [2020] proposes a GIN-based approach that allows multiple latent parents behind every pair of observed variables and can identify causal directions among latent variables, but it still requires that each latent variable set should have at least twice more measured variables as children.
- Moreover, Huang* et al. [2020] considered a special type of confounders in heterogeneous data, where the confounder can be represented as a function of domain index or a smooth function of time, so one may use the known domain index or time index as a surrogate variable to remove the influence from those confounders and thus identify causal structure over observed variables.
- Mixture oracles-based method. Recently, Kivva et al. [2021] proposed a mixture oracles-based method to identify the latent variable graph that allows nonlinear causal relationships. It is based on assumptions that the latent variables are discrete and each latent variable has measured variables as children. Thanks to the discreteness assumption, it can handle more general DAGs over latent variables.

D Proofs

D.1 Proof of Theorem 1

Theorem 1 (Graphical Implication of Rank Constraints in IL^{2}H Graphs)**.** *Suppose $\mathcal{G}$ satisfies an IL^{2}H graph. Under the rank faithfulness assumption, the cross-covariance matrix $\Sigma_{\mathbf{X}_A,\mathbf{X}_B}$ over measured variables $\mathbf{X}_A$ and $\mathbf{X}_B$ in $\mathcal{G}$ (with $|\mathbf{X}_A|, |\mathbf{X}_B| > r$) has rank r, if and only if there exists a subset of latent variables $\mathbf{L}$ with $|\mathbf{L}| = r$ such that $\mathbf{L}$ d-separates $\mathbf{X}_A$ from $\mathbf{X}_B$, and there is no $\mathbf{L}'$ with $|\mathbf{L}'| < |\mathbf{L}|$ that d-separates $\mathbf{X}_A$ from $\mathbf{X}_B$. That is,*

$$rank(\Sigma_{\mathbf{X}_A,\mathbf{X}_B}) = min\{|\mathbf{L}| : \mathbf{L} \text{ d-separates } \mathbf{X}_A \text{ from } \mathbf{X}_B\}.$$

Proof. Theorem 1 is a special case of Theorem 2.8 in Sullivant et al. [2010] when applied to IL^{2}H graphs. Different from the setting in Sullivant et al. [2010] which access to the full covariance matrix $\Sigma_{\mathbf{V}_\mathcal{G},\mathbf{V}_\mathcal{G}}$ is assumed, we only have access to the covariance matrix $\Sigma_{\mathbf{X}_\mathcal{G},\mathbf{X}_\mathcal{G}}$ over the measured variables $\mathbf{X}_\mathcal{G}$.

It is enough to show that for IL^{2}H graphs, (C_A, C_B) t-separating $\mathbf{X}_A$ from $\mathbf{X}_B$ is equivalent to $\mathbf{L}$ d-separating $\mathbf{X}_A$ from $\mathbf{X}_B$, where $C_A, C_B \subset \mathbf{L}_\mathcal{G}$.

Since $\mathbf{L}$ d-separates $\mathbf{X}_A$ from $\mathbf{X}_B$ and since any $X \in \mathbf{X}_\mathcal{G}$ cannot be the choke point, we can choose $C_A = \mathbf{L}$ and $C_B = \emptyset$, so that (C_A, C_B) t-separates $\mathbf{X}_A$ from $\mathbf{X}_B$ in IL^{2}H graphs.

Therefore, combining Theorem 2.8 in Sullivant et al. [2010] and the above equivalence, the theorem is proved. □

D.2 Proof of Theorem 2

Theorem 2 (Measurement as a surrogate)**.** *Suppose $\mathcal{G}$ is an IL^{2}H graph. Denote by $\mathbf{A}, \mathbf{B} \subseteq \mathbf{V}_\mathcal{G}$ two subsets of variables in $\mathcal{G}$, with $\mathbf{A} \cap \mathbf{B} = \emptyset$. Furthermore, denote by $\mathbf{X}_A$ the set of measured variables that are d-separated by $\mathbf{A}$ from all other measures, and by $\mathbf{X}_B$ the set of measured variables that are d-separated by $\mathbf{B}$ from all other measures. Then $\texttt{rank}(\Sigma_{\mathbf{A},\mathbf{B}}) = \texttt{rank}(\Sigma_{\mathbf{X}_A,\mathbf{X}_B})$.*

Proof. According to Theorem 1, the rank of $\Sigma_{X_\mathbf{A},X_\mathbf{B}}$ is the minimal number of latent variables $\mathbf{L}$ that d-separate $X_\mathbf{A}$ and $X_\mathbf{B}$; that is, $\mathbf{L}$ block all paths between $X_\mathbf{A}$ and $X_\mathbf{B}$ with the smallest cardinality. Furthermore, since $\mathbf{A}$ d-separates $\mathbf{X}_A$ from all other measures, $\mathbf{X}_A$ are the measured pure descendants of $\mathbf{A}$. Similarly, $\mathbf{X}_B$ are the measured pure descendants of $\mathbf{B}$. Moreover, given the structure of IL^{2}H graphs, $\mathbf{L}$ also block all paths between $\mathbf{A}$ and $\mathbf{B}$ with the smallest cardinality. Therefore, the rank of $\Sigma_{\mathbf{A},\mathbf{B}}$ is also $|\mathbf{L}|$, equivalent to that of $\Sigma_{\mathbf{X}_A,\mathbf{X}_B}$. □

D.3 Proof of Theorem 3

Theorem 3 (Correct Cluster Condition)**.** *Suppose $\mathcal{G}$ is an IL^{2}H graph with measured variables $\mathbf{X}_\mathcal{G}$. Consider the output $\mathcal{G}'$ from applying findCausalClusters over $\mathbf{X}_\mathcal{G}$. If none of the clusters in $\mathcal{G}'$ is the bond set in $\mathcal{G}$, then all latent atomic covers have been correctly identified.*

Before the proof of Theorem 3, we first give the following lemma which shows that if $\mathbf{X} \subseteq \mathbf{X}_\mathcal{G}$ is not a correct cluster in the true graph $\mathcal{G}$, then $\mathbf{X}$ forms a bond set. In other words, If a set of measured variables does not form a bond set of $\mathcal{G}$, then it must form a correct cluster.

Lemma 1 (Fake-Cluster ⇒ Bond Set)**.** *A set of measured variables $\mathbf{X} \subseteq \mathbf{X}_\mathcal{G}$ that is mistakely tested as a rank-deficient set by findCausalClusters but does not form a cluster in the true graph $\mathcal{G}$ (that is, a fake cluster), only if $\mathbf{X}$ forms a bond set of $\mathcal{G}$.*

Below, we first give the proof of Lemma 1.

Proof. Suppose we found a rank deficient set of variables $\mathbf{X} = \{X_1, X_2, \cdots\}$ which is not a cluster in $\mathcal{G}$. This implies that (i) there exists at least two disjoint latent atomic covers, $\mathbf{L}_1, \mathbf{L}_2$, which d-separate variables in $\mathbf{X}$ from all other measures, and (ii) $\mathbf{X}$ does not contain all the pure children of either $\mathbf{L}_1$ nor $\mathbf{L}_2$, because otherwise, due to the IL^{2}H requirement that each $\mathbf{L}_i$ has $> |\mathbf{L}_i|$ pure children, we would have discovered that a subset of $\mathbf{X}$ was rank deficient and clustered together earlier by

findCausalClusters. This implies that $\mathbf{L}_1, \mathbf{L}_2$ will d-separate the remaining pure child of $\mathbf{L}_1$ from that of $\mathbf{L}_2$, implying that $\mathbf{X}$ is a bond set. □

Now we are ready to prove Theorem 3.

Proof. We will prove that all latent atomic covers can be correctly identified *from bottom to top*, if there is no bond set.

First, from Lemma 1 we know that if a set of measured variables is not a bond set, then they form a correct causal cluster, and thus the corresponding identified latent cover is correct. Denote the latent atomic covers identified at this step by $\mathbf{L}'_1$, and denote by $\mathcal{G}'_1$ the currently estimated graph.

After identifying the latent atomic covers at the downmost level, we next continue to form causal clusters from root variables in $\mathcal{G}'_1$, including $\mathbf{L}'_1$ and the remaining measured variables that did not form clusters in the previous step. If any of the latent covers in $\mathbf{L}'_1$ have latent children $\mathbf{L}_i$ in the true graph $\mathcal{G}$ that have not been identified in the current step, then reverse these edges such that the children become parents. Such an operation will not affect the discovery of latent covers, and $\mathbf{L}_i$ will be found in later steps. So by further leveraging Lemma 1 and by treating $\mathbf{L}'_1$ and the remaining measured variables that did not form clusters in the previous step as "measured variables", this step results in correctly identified latent covers $\mathbf{L}'_2$ and estimated graph $\mathcal{G}'_2$.

We can now iteratively repeat the previous step to discover new latent atomic covers from the root variables in the estimated graph in the previous step, until no more rank deficient sets can be found. □

D.4 Proof of Theorem 4

Theorem 4 (Correcting Clusters)**.** *Denote by $\mathcal{G}'$ the output from findCausalClusters and by $\mathcal{G}$ the true graph. For a latent atomic cover $\mathbf{L}'$ in $\mathcal{G}'$, if the measured pure descendants of $\mathbf{L}'$ is a bond set in the true graph $\mathcal{G}$, then there exist a set of siblings $\mathbf{S}$ of $\mathbf{L}'$ in $\mathcal{G}'$, a set of children $\mathbf{C}$ of $\mathbf{L}'$, and a set of grandparents $\mathbf{P}$ of $\mathbf{L}'$, such that $\mathcal{M}_{\mathcal{G}'}(\mathbf{S} \cup \mathbf{C} \cup \mathbf{P})$ forms a cluster that is not a bond set in $\mathcal{G}$.*

Proof. Suppose the measured pure descendants of $\mathbf{L}'$, $\mathbf{X} := \mathcal{M}_{\mathcal{G}'}(\mathbf{L}')$, is a bond set in the true graph $\mathcal{G}$. Denote by $\mathbf{L}_S \subseteq \mathbf{L}_{\mathcal{G}}$ the minimal set of latent variables in $\mathcal{G}$ that d-separates $\mathbf{X}$ from all other measures $\mathbf{X}' := \mathbf{X}_{\mathcal{G}} \backslash \mathbf{X}$, and since $\mathbf{X}$ is a bond set, $\mathbf{L}_S$ also d-separates some disjoint partition of measures $\mathbf{X}_i \subset \mathbf{X}'$ from $\mathbf{X}_j \subset \mathbf{X}'$, and accordingly, denote by $\mathcal{G}_i$ the subgraph that contains measures $\mathbf{X}_i$ and by $\mathcal{G}_j$ the subgraph that contains measures $\mathbf{X}_j$. Moreover, denote by $\mathbf{L}_i \subset \mathbf{L}_S$ the set of latent variables that d-separates $\mathbf{X}_i$ from other measures. The proof contains three steps.

In step 1, we show that for each $\mathbf{L}'$'s sibling, its measured pure descendants are the same set of measured variables as that in the subgraph $\mathcal{G}_i$ for some i. To this end, we first show that variables in each subgraph $\mathcal{G}_i$ only formed clusters with variables in the same $\mathcal{G}_i$. Without loss of generality, suppose there are only two subgraphs. Suppose for contradiction that we discovered a cluster of variables $\mathbf{V}$ such that $\mathcal{M}_{\mathcal{G}'}(\mathbf{V})$ comprises measures from $\mathcal{G}_1, \mathcal{G}_2$. Then $\mathbf{V}$ can be separated into variable sets $\mathbf{V}_1, \mathbf{V}_2$ belonging to $\mathcal{G}_1, \mathcal{G}_2$ respectively. However, the minimal d-separating set for $\mathbf{V}_1, \mathbf{V}_2$ must not overlap, since they are in different subgraphs. This implies that either $\mathbf{V}_1, \mathbf{V}_2$ forms a rank deficient set by itself, which should have been discovered earlier. Hence, we reach a contradiction.

We next show that for each $\mathbf{L}'$'s sibling, its measured pure descendants are not a proper subset of the measured variables as that in $\mathcal{G}_i$. If it was the case, the variables in $\mathcal{G}_i$ would continue to form variables with one another until $\mathbf{L}_i$ is in the minimal d-separating set, where $\mathcal{M}_{\mathcal{G}'}(\mathbf{A}') = \mathbf{X}_i$, hence showing the claim.

In step 2, we show that the cardinality of each sibling of $\mathbf{L}'$ is equal to $|\mathbf{L}_i|$ for some i. We first show that for a variable set $\mathbf{V}_i$ such that $\mathcal{M}(\mathbf{V}_i) \subseteq \mathcal{M}(\mathcal{G}_i)$, it is not possible that $|\mathbf{V}_i| < |\mathbf{L}_i|$. This is because if this was the case, it implies that there exists a latent set of smaller cardinality that separates the measured variables in $\mathcal{G}_i$ from the rest of the graph, which contradicts the fact that $\mathbf{L}_i$ is minimal. So it is always the case that $|\mathbf{V_i}| \geq |\mathbf{L}_i|$.

Next, we show that as long as any such variable set $|\mathbf{V}_i| > |\mathbf{L}_i|$, it will be able to form a cover with cardinality $< |\mathbf{V}_i|$. This is because if no rank deficient sets exist for cardinality $k < |\mathbf{L}_i|$,

$\{\mathbf{V}_i$: the measured variables in $\mathbf{V}_i$ = the measured variables in $\mathcal{G}_i\}$ will form a cluster of cardinality $|\mathbf{L}_i|$, since $|\mathbf{L}| > |\mathbf{L}_i| < |\mathbf{V_i}|$.

Thus, by combining step 1 and step 2, we have shown that each sibling of $\mathbf{L}'$ corresponding to $\mathbf{L}_i$ for some i.

Finally, in step 3, we show that, for $\mathbf{L}'$, there exist a set of siblings $\mathbf{S}$, a set of children $\mathbf{C}$, and a set of grandparent $\mathbf{P}$, so that their union $\mathbf{S} \cup \mathbf{C} \cup \mathbf{P}$ will not form a bond cover. Note that the measured pure descendants of siblings or grandparents are $\mathbf{X}_i$, which is the reason why we need to consider the siblings and grandparents. Moreover, note that the reason $\mathbf{L}'$ is formed as bond cover is that when it is formed, its slibings have not been found yet. Intuitively, now its siblings have been found, so we can find the correct clusters.

Suppose for contradiction that refining clusters will discover a new bond cover $\mathbf{L}_{bond}$, and without loss of generality, suppose the minimal d-separating set involves some distinct covers $\mathbf{L}_A, \mathbf{L}_B$. Each of $\mathbf{L}_A, \mathbf{L}_B$ must respectively d-separate some partition of variables $\mathbf{V}_A, \mathbf{V}_B \subset \mathbf{V}$ from all other variables remaining in $\mathbf{V}$. We also know that in order to find rank deficiency, $\|\mathbf{L}_A\| + \|\mathbf{L}_B\| < \|\mathbf{V}_A\| + \|\mathbf{V}_B\|$, implying that $\|\mathbf{L}_A\| < \|\mathbf{V}_A\|$ or $\|\mathbf{L}_B\| < \|\mathbf{V}_B\|$. However, since $\mathbf{L}_A, \mathbf{L}_B$ d-separates $\mathbf{V}_A, \mathbf{V}_B$ from all other variables respectively, testing either $\mathbf{V}_A, \mathbf{V}_B$ must have been rank deficient. Since either of them are over a smaller latent cardinality ($\|\mathbf{L}_A\|, \|\mathbf{L}_B\| < \|\mathbf{L}_A\| + \|\mathbf{L}_B\|$), one of them must have been discovered as a cluster earlier. Hence, we reach a contradiction. □

D.5 Proof of Lemma 5

Lemma 5 (Rank Invariance)**.** *Denote by $\mathcal{G}'$ the output from findCausalClusters and by $\mathbf{L}$ a latent atomic cover in $\mathcal{G}'$. Then the rank constraints over $\mathbf{X}_{\mathcal{G}}$ prescribed by $\mathcal{G}'$ before and after the operation makeRoot($\mathbf{L}$) are identical.*

Proof. For an IL^{2}H graph $\mathcal{G}'$ and a latent atomic cover $\mathbf{L}$ in $\mathcal{G}'$, after applying the makeRoot operator to $\mathbf{L}$, which results in $\mathcal{G}''$, $\mathcal{G}'$ and $\mathcal{G}''$ are in the same Markov equivalence class. Therefore, $\mathcal{G}'$ and $\mathcal{G}''$ have the same rank constraints. □

D.6 Proof of Lemma 6

Lemma 6 (Cross-Cover Test)**.** *Given a set of variables $\mathcal{S}$, consider two latent atomic covers $\mathbf{L}_A, \mathbf{L}_B \in \mathcal{S}$, and a potential separating set $\mathbf{C} = \{\mathbf{L}_{C_i}\} \subseteq \mathcal{S}\backslash\{\mathbf{L}_A, \mathbf{L}_B\}$. For each $\mathbf{L}_{C_i}$, consider $\mathbf{C}_i^A, \mathbf{C}_i^B \subseteq PCh(\mathbf{L}_{C_i})$ with $\mathbf{C}_i^A, \mathbf{C}_i^B \neq \emptyset$ and $\mathbf{C}_i^A \cap \mathbf{C}_i^B = \emptyset$, and denote the cardinality $k_i^A := min(|\mathbf{L}_{C_i}|, |\mathbf{C}_i^A|)$, $k_i^B := min(|\mathbf{L}_{C_i}|, |\mathbf{C}_i^B|)$, respectively. Then there is no edge between $\mathbf{L}_A$ and $\mathbf{L}_B$ if and only if there exists a separating set $\mathbf{C}$ such that $\texttt{rank}(\Sigma_{\mathcal{A},\mathcal{B}}) < min(|\mathbf{L}_A| + \sum_i k_i^A, |\mathbf{L}_B| + \sum_i k_i^B)$, where $\mathcal{A} = \{\mathbf{L}_A, \mathbf{C}_1^A, \mathbf{C}_2^A, ...\}$ and $\mathcal{B} = \{\mathbf{L}_B, \mathbf{C}_1^B, \mathbf{C}_2^B, ...\}$. In this case, we say that $\mathbf{C}$ satisfies the cross-cover test of $\mathbf{L}_A$ against $\mathbf{L}_B$.*

Proof. We first show that if there is no edge between $\mathbf{L}_A$ and $\mathbf{L}_B$, then $\texttt{rank}(\Sigma_{\mathcal{A},\mathcal{B}}) < min(|\mathbf{L}_A| + \sum_i k_i^A, |\mathbf{L}_B| + \sum_i k_i^B)$.

Since there is no edge between $\mathbf{L}_A$ and $\mathbf{L}_B$, there exists a set $\mathbf{C} = \{\mathbf{L}_{C_i}\}$, so that given $\mathbf{C}$, $\mathbf{L}_A$ and $\mathbf{L}_B$ are d-separated. Since $\mathbf{C}_i^A$ and $\mathbf{C}_i^B$ are the children of $\mathbf{L}_{C_i}$, $\mathbf{C}$ d-separates $\mathcal{A}$ from $\mathcal{B}$ as well. Then according to Theorem 7.1, $\texttt{rank}(\Sigma_{\mathcal{A},\mathcal{B}}) = |\mathbf{C}|$. Moreover, since $|\mathbf{C}| < min(|\mathbf{L}_A| + \sum_i k_i^A, |\mathbf{L}_B| + \sum_i k_i^B)$, we have $\texttt{rank}(\Sigma_{\mathcal{A},\mathcal{B}}) < min(|\mathbf{L}_A| + \sum_i k_i^A, |\mathbf{L}_B| + \sum_i k_i^B)$.

Next we show that $\texttt{rank}(\Sigma_{\mathcal{A},\mathcal{B}}) < min(|\mathbf{L}_A| + \sum_i k_i^A, |\mathbf{L}_B| + \sum_i k_i^B)$, then there is no edge between $\mathbf{L}_A$ and $\mathbf{L}_B$.

Now suppose that there are edges between $\mathbf{L}_A$ and $\mathbf{L}_B$. Then $\Sigma_{\mathcal{A},\mathcal{B}}$ is not rank deficient; that is $\texttt{rank}(\Sigma_{\mathcal{A},\mathcal{B}}) = min(|\mathbf{L}_A| + \sum_i k_i^A, |\mathbf{L}_B| + \sum_i k_i^B)$. Therefore, if $\texttt{rank}(\Sigma_{\mathcal{A},\mathcal{B}}) < min(|\mathbf{L}_A| + \sum_i k_i^A, |\mathbf{L}_B| + \sum_i k_i^B)$, then there is no edge between $\mathbf{L}_A$ and $\mathbf{L}_B$. □

D.7 Proof of Lemma 7

Lemma 7 (V-Structure Test)**.** *For any unshielded triangle $\mathbf{L}_A - \mathbf{L}_C - \mathbf{L}_B$, let $\mathcal{A}, \mathcal{B}$ be the set of variables in Lemma **??** such that $\Sigma_{\mathcal{A},\mathcal{B}}$ was rank deficient. Let $k = \texttt{rank}(\Sigma_{\mathcal{A},\mathcal{B}})$, $k_1 = \texttt{rank}(\Sigma_{\mathcal{A}\cup\mathbf{L}_C,\mathcal{B}})$, and $k_2 = \texttt{rank}(\Sigma_{\mathcal{A},\mathcal{B}\cup\mathbf{L}_C})$. Then, $\mathbf{L}_A \rightarrow \mathbf{L}_B \leftarrow \mathbf{L}_C$ if and only if $k < \min(k_1, k_2)$.*

Proof. We first show that if $\mathbf{L}_A \to \mathbf{L}_C \leftarrow \mathbf{L}_B$, then $k < \min(k_1, k_2)$.

Since $\mathbf{L}_A \to \mathbf{L}_C \leftarrow \mathbf{L}_B$, $\mathbf{L}_C$ cannot be in the separation set of $\mathbf{L}_A$ and $\mathbf{L}_B$; that is, given $\mathbf{L}_C$, $\mathbf{L}_A$ and $\mathbf{L}_B$ are d-connected. Hence, $k < k_1$ and $k < k_2$, and thus $k < \min(k_1, k_2)$.

Next we show that if $k < \min(k_1, k_2)$, then $\mathbf{L}_A \to \mathbf{L}_C \leftarrow \mathbf{L}_B$.

Suppose $\mathbf{L}_A, \mathbf{L}_C, \mathbf{L}_B$ do not form a v-structure; that is $\mathbf{L}_A \to \mathbf{L}_C \to \mathbf{L}_B$ or $\mathbf{L}_A \leftarrow \mathbf{L}_C \leftarrow \mathbf{L}_B$. Then $k = \min(k_1, k_2)$, since $\mathbf{L}_C$ has been considered before in order to achieve rank deficiency of $\Sigma_{\mathcal{A},\mathcal{B}}$. Therefore, if $k < \min(k_1, k_2)$, then $\mathbf{L}_A \to \mathbf{L}_C \leftarrow \mathbf{L}_B$. □

D.8 Proof of Lemma 8

Lemma 8. *Suppose $\mathcal{G}$ is an IL^{2}H graph. The rank constraints are invariant with the minimal-graph operator and the skeleton operator; that is, $\mathcal{G}$ and $\mathcal{O}_{skeleton}(\mathcal{O}_{min}(\mathcal{G}))$ are rank equivalent.*

Proof. We first show that the minimal-graph operator will not change rank deficiency constraints. Denote by $\mathcal{G}_1$ and $\mathcal{G}_2$ the graph before and after applying the minimal-graph operator, respectively. For every latent atomic cover $\mathbf{L}$ in $\mathcal{G}_1$, since those three conditions hold, for any $\mathbf{C} \subseteq PCh_{\mathcal{G}'}(\mathbf{L}')$ and for any $\mathbf{S} \subseteq Sib_{\mathcal{G}'}(\mathbf{L}')$ with $\mathbf{C}, \mathbf{S} \neq \emptyset$, $\texttt{rank}(\Sigma_{\mathcal{A},\mathcal{B}}) = |\mathbf{P}|$, where $\mathcal{A} = \mathbf{C} \cup \mathbf{S}$ and $\mathcal{B} = \mathbf{X}_{\mathcal{G}'} \backslash \mathcal{M}_{\mathcal{G}'}(\mathcal{A})$. So, after merging $\mathbf{L}$ to its parents $\mathbf{P}$, the cardinality of the d-separation set between any two sets of variables does not change. Thus, according to Theorem 1, the rank constraints will not change after merging $\mathbf{L}$ to its parents $\mathbf{P}$.

Moreover, it is trivial to show that the skeleton operator will not change rank deficiency constraints, because the d-separation set between any two sets of variables will not change. □

D.9 Proof of Theorem 9

Theorem 9. *Suppose $\mathcal{G}$ is an IL^{2}H graph with measured variables $\mathbf{X}_{\mathcal{G}}$. Phases I-II in Algorithm 1 over $\mathbf{X}_{\mathcal{G}}$ can asymptotically identify the latent atomic covers of $\mathcal{O}_{min}(\mathcal{G})$, with the first two conditions in Condition 1.*

Proof. Theorem 3 has shown that *findCausalClusters* gives correct latent covers when there is no bond set. Furthermore, Theorem 4 shows that even in the presence of bond sets, refining the set over $Ch(\mathbf{L}') \cup Sib(\mathbf{L}') \cup Gp(\mathbf{L}')$, where $\mathcal{M}_{\mathcal{G}'}(\mathbf{L}')$ forms a bond set, can correct the clusters.

Phase II *refineClusters* refines clusters over $Ch(\mathbf{L}') \cup Sib(\mathbf{L}') \cup Gp(\mathbf{L}')$ in a breadth-first search from the root variable, and it ends after refining every latent cover in $\mathcal{G}'$. Therefore, with this refining procedure, we will derive correct latent covers. □

D.10 Proof of Theorem 10

Theorem 10. *Suppose $\mathcal{G}$ is an IL^{2}H graph with measured variables $\mathbf{X}_{\mathcal{G}}$. Algorithm* **??**, *including Phases I-III, over $\mathbf{X}_{\mathcal{G}}$ can asymptotically identify the Markov equivalence class of $\mathcal{O}_{min}(\mathcal{O}_s(\mathcal{G}))$.*

Proof. Theorem 5 has shown that Phases I-II can find the correct clusters and latent atomic covers of $\mathcal{O}_{min}(\mathcal{G})$. Moreover, Lemma 2 and Lemma 3 have shown that by performing Cross-Cover Test and V-Structure Test, the skeleton and v structure among every triple of latent variables can be correctly identified.

Phase III *refineEdges* refines the edges over $\mathbf{L}' \cup PCh(\mathbf{L}') \cup PCh(PCh(\mathbf{L}'))$ by performing Cross-Cover Test and V-Structure Test in a depth-first search from the root variable, and it ends until $\mathbf{L}'$ does not have latent children. Therefore, with this refining procedure, we will derive correct skeletons and v structures. □

E More Explanations on Identifiability Conditions, Algorithms, and Theorems

E.1 More Explanations on Definition 3 (*Effective Cardinality*)

The effective cardinality, defined in Definition 3, can be estimated with the following procedure.

$j \leftarrow 1$;
$\mathbf{C} \leftarrow PCh_{\mathcal{G}}(\mathbf{L})$;
while $j < |\mathbf{L}|$ **do**
  Find the largest subset of variables $\mathbf{C}' \subseteq \mathbf{C}$ such that $|\mathbf{C}'| > |Pa_{\mathcal{G}}(\mathbf{C}')| = j$;
  Introduce a set of latents $\mathbf{L}'$ with $|\mathbf{L}'| = |Pa_{\mathcal{G}}(\mathbf{C}')|$;
  add $\mathbf{L}'$ as new children of $Pa_{\mathcal{G}}(\mathbf{C}')$;
  $\mathbf{C} \leftarrow \mathbf{C} \backslash \mathbf{C}' \bigcup \mathbf{L}'$;
  $j \mathrel{+}= 1$;
end while
return $|\mathbf{C}|$

For example, for Figure 1(a), the effective cardinality of the pure children of $\{L_7, L_8\}$ is 3, because $|\{X_4, X_5\}| > |\{L_7\}|$ and we replace $\{X_4, X_5\}$ with a single latent variable L', so the cardinality of the resulting children set is $|\{X_6, X_7, L'\}| = 3$.

E.2 More Explanations on Definition 4 (*Latent Atomic Cover*)

The first two conditions in Definition 4 ensure that there are enough variables in the current active variable set to find the rank deficiency, so that we can determine the latent atomic cover with size k. However, note that they may not be the necessary conditions. For example, for the graphs in Figure 8, some of the latent atomic covers only have k neighbors (except for the $k+1$ pure children), but they are still identifiable.

The first half of the third condition, "there does not exist a partition of $\mathbf{L} = \mathbf{L}_1 \cup \mathbf{L}_2$, so that both $\mathbf{L}_1, \mathbf{L}_2$ satisfy conditions 1 and 2", ensure that the latent atomic cover $\mathbf{L}$ is atomic.

The second half of the third condition, "there does not exist a partition of $\mathbf{L} = \mathbf{L}_1 \cup \mathbf{L}_2$, so that $\{PCh_{\mathcal{G}}(\mathbf{L}_1) \cup PCh_{\mathcal{G}}(\mathbf{L}_2)\} \backslash \mathbf{L} = PCh_{\mathcal{G}}(\mathbf{L})$", is needed in the overlapping case. Consider the following graph: $L_1 \rightarrow \{X_1, X_2, X_3\}, L_2 \rightarrow \{X_3, X_4, X_5\}$. Here, $\mathbf{L}_1 = \{L_1\}$, $\mathbf{L}_2 = \{L_2\}$, and $\mathbf{L} = \{L_1, L_2\}$ are latent atomic covers; however, $\{PCh_{\mathcal{G}}(\mathbf{L}_1) = \{X_1, X_2\}$, $\{PCh_{\mathcal{G}}(\mathbf{L}_2) = \{X_4, X_5\}$, and $\{PCh_{\mathcal{G}}(\mathbf{L}) = \{X_1, X_2, X_3, X_4, X_5\}$, so $\{PCh_{\mathcal{G}}(\mathbf{L}_1) \cup PCh_{\mathcal{G}}(\mathbf{L}_2)\} \backslash \mathbf{L} \neq PCh_{\mathcal{G}}(\mathbf{L})$. Therefore, although both $\mathbf{L}_1$ and $\mathbf{L}_2$ satisfy the first two conditions, $\mathbf{L}$ satisfies Definition 4.

E.3 More Explanations on Condition 1 (*IL^{2}H graph*)

The first two conditions in Condition 1 guarantee to identify the latent atomic covers, while the last one is used to identify the edges among latent atomic covers when performing Cross-Cover Test and V-Structure Test.

Specifically, the second condition says that two latent atomic covers that are partly overlapped is not allowed, except for the case that one latent atomic cover is contained in another one; otherwise, the cover creation rule can be non-trivial. For example, for the graph in Figure 9(a), although $PDe_{\mathcal{G}}(\{L_3, L_4\}) \bigcap PDe_{\mathcal{G}}(\{L_4\}) = \{X_3, X_4\}$, it satisfies the second condition in Condition 1 because $\{L_3\} \subset \{L_2, L_3\}$. In contrast, for the graph in Figure 9(b), $PDe_{\mathcal{G}}(\{L_3, L_4\}) \bigcap PDe_{\mathcal{G}}(\{L_4, L_5\}) = \{X_3\}$, but $\{L_3, L_4\}, \{L_4, L_5\}$ are not subsets or descendants of one another. Hence, it does not satisfy the second condition of an IL^{2}H graph.

E.4 More Explanations on Algorithm 2 (*findCausalClusters*)

The search procedure in Algorithm 2 (*findCausalClusters*) contains the following two key updates.

- *Latent-atomic-cover size update.* We start to identify latent atomic covers with size $k = 1$. If a rank-deficiency set is not found, then increment $k = k + 1$; otherwise, reset $k = 1$.
- *Active variable set update.* The set of active variables $\mathcal{S}$ is set to $\mathbf{X}_{\mathcal{G}}$ initially. We consider any subset of the latent atomic covers in $\mathcal{S}$ and replace them with their pure children, resulting in $\tilde{\mathcal{S}}$. At

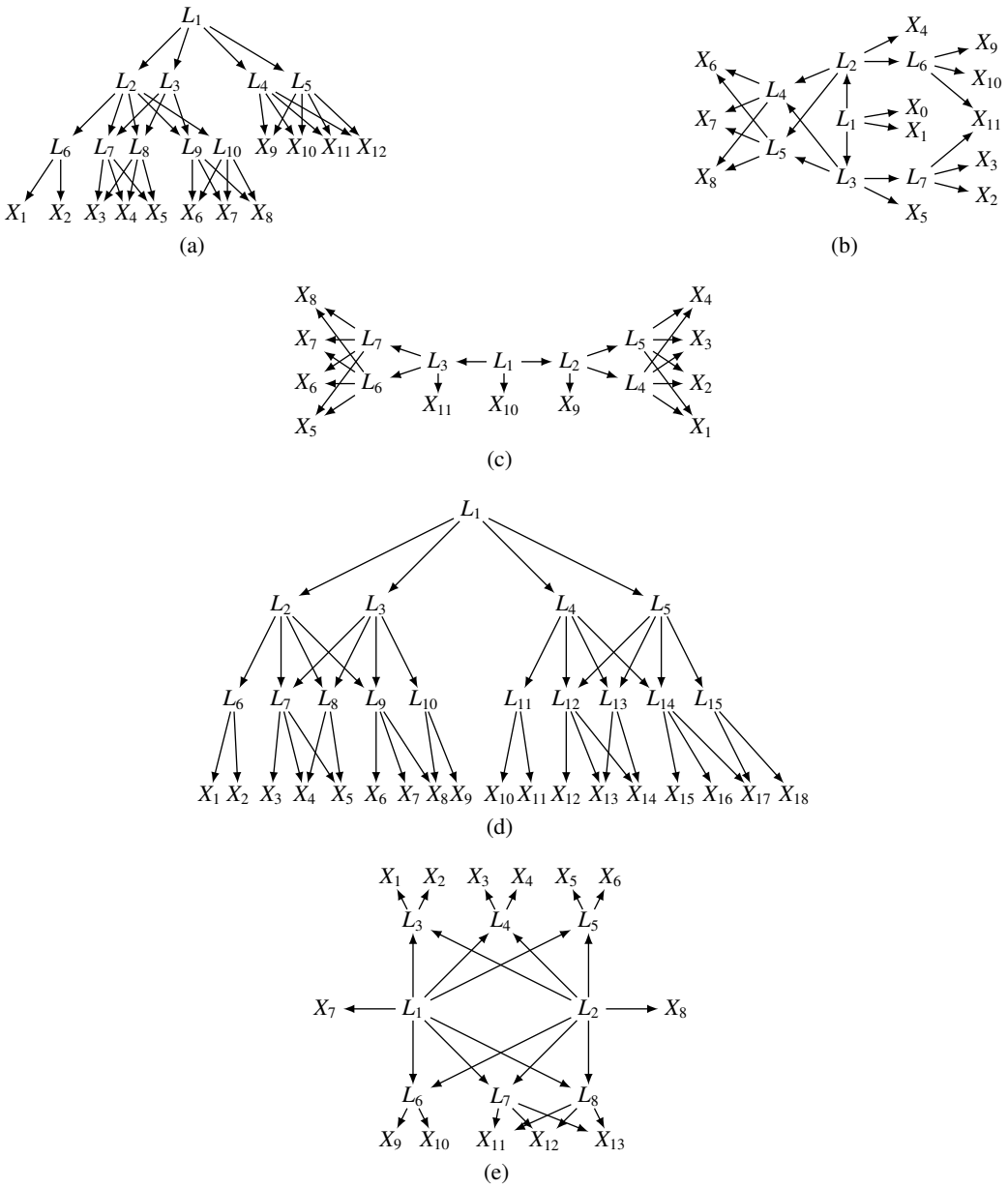


Figure 8: Latent hierarchical graphs where some latent atomic covers only have k neighbors, except for the $k+1$ pure children, but the graph structure is still identifiable. Note that in many cases, k-neighbor is enough; the condition "$k+1$ neighbors" is sufficient, but not necessary.

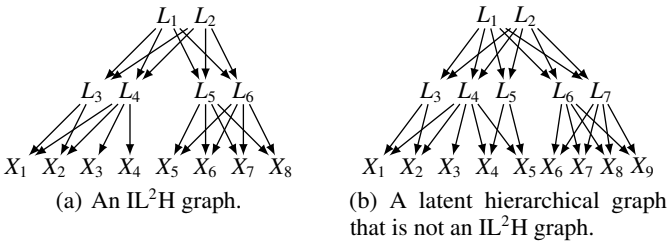


Figure 9: Examples and counter-examples of IL^{2}H graphs.

the same time, we search for the rank-deficiency set $\mathbf{A}$ with rank k in $\tilde{\mathcal{S}}$; if it is found, then assign a latent atomic cover $\mathbf{L}$ of size k as the parent $\mathbf{A}$, and accordingly, the active variable set is updated as $(\mathcal{S}\backslash\mathbf{A}) \cup \mathbf{L}$.

Note that in Algorithm 2 *findCausalClusters*, we search over $\tilde{\mathcal{S}}$, instead of $\mathcal{S}$, to avoid adding a certain type of redundant latent variables when some latent atomic covers have overlapping variables, including the v structure; see Figure 10 for an illustration.

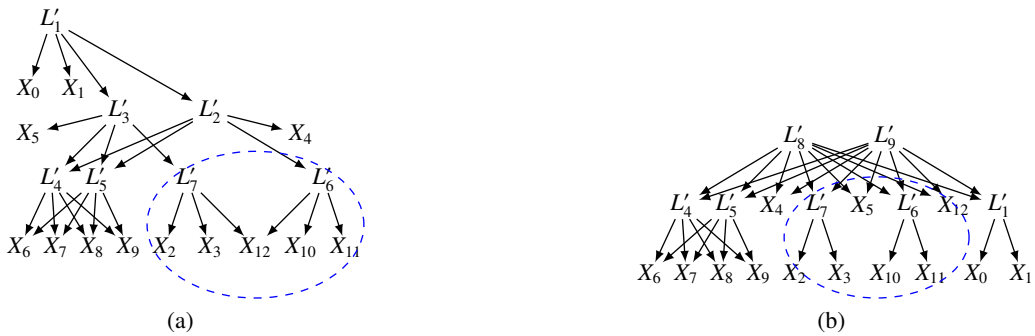


Figure 10: (a) Output from Phase I *findCausalClusters* with ground truth graph in Figure 7(b), where we consider the children of the active variable set $\mathcal{S}$ (i.e., $\tilde{\mathcal{S}}$) to search for the rank deficiency set. (b) The output graph if directly searching over $\mathcal{S}$ in Phase I; that is, without considering line 4 in Algorithm 2. The output graph without considering $\tilde{\mathcal{S}}$ is not correct; for instance, X_{12} will not be considered as the children of $\{L_6, L_7\}$.

Moreover, in Algorithm 2 *findCausalClusters*, if there are conflicts when the search goes on, then we just ignore it, and such conflicts will be handled in Algorithm 3 *refineClusters*. Also, note that except for the v structure where the measured variable is a collider, other "v structures" in the intermediate output in Algorithms 2 and 3 are not true v structures.

For the illustration of Algorithm 2 given in Figure 2, here we give more detailed explanations. Specifically, we first set $k = 1$ and the active set is $\mathcal{S} = \{X_1, \cdots, X_{16}\}$ and $\tilde{\mathcal{S}} = \mathcal{S}$, and we can find the clusters in (a), and no further cluster can be found with $k = 1$. Then we increase k to 2 with the active set $\mathcal{S} = \{\{L_6\}, \{L_7\}, X_6, \cdots, X_{16}\}$ and $\tilde{\mathcal{S}} = \mathcal{S}$, and then we can find the clusters in (b). Then, the active set is $\mathcal{S} = \{\{L_4, L_5\}, \{L_6\}, \{L_7, L_8\}, \{L_9, L_{10}\}\}$ and we set back $k = 1$, and when $\tilde{\mathcal{S}} = \{\{L_4, L_5\}, X_1, \cdots, X_{11}\}$ we find the cluster in (c). Note that when testing the rank over $\{L_4, L_5\}$ against other variables, we use their measured pure descendants in the currently estimated graph instead. The above procedure is repeated to further find the cluster in (d). Finally, when there are no enough variables for testing, we connect the elements in the active variable set: connecting $\{L_2, L_3\}$ to $\{L_7, L_8\}$ in (e).

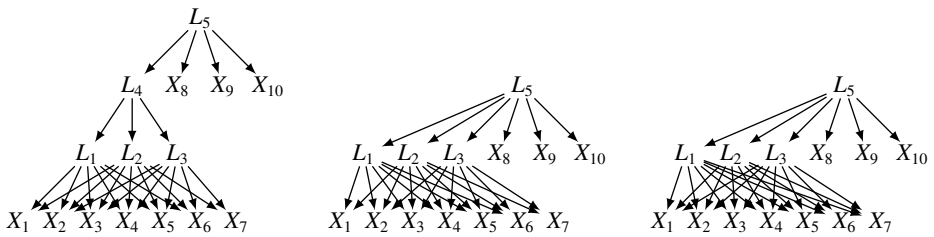
E.5 Explanation of Minimal-Graph Operator and Skeleton Operator

The minimal-graph operator and the skeleton operator will not change the rank constraints, or in other words, graphs before and after applying the operators are indistinguishable with rank constraints.

Give an IL^{2}H graph in Figure 11(a), after applying the minimal-graph operator, the latent atomic cover L_4 will be merged to its parent L_5, resulting in the graph in Figure 11(b), while the rank constraints will not change. Furthermore, after applying the skeleton operator to the graph in Figure 11(b), L_1 has an edge to X_7 and L_3 has an edge to X_1, resulting in the graph in Figure 11(c), which also does not change the rank constaints.

E.6 Complexity of Algorithm 1

The time complexity of the algorithm is upper bounded by $O(r \sum_{k=2}^{l+1} \binom{m}{k})$, and this bound is further upper bounded by $O(r(1+m)^{l+1})$, where m is the number of measured variables, l is the cardinality of the largest latent cover of the estimated graph, with $l \ll m$, and r is the number of levels of the estimated hierarchical graph, with $r \ll m$.



(a) An IL^{2}H graph.

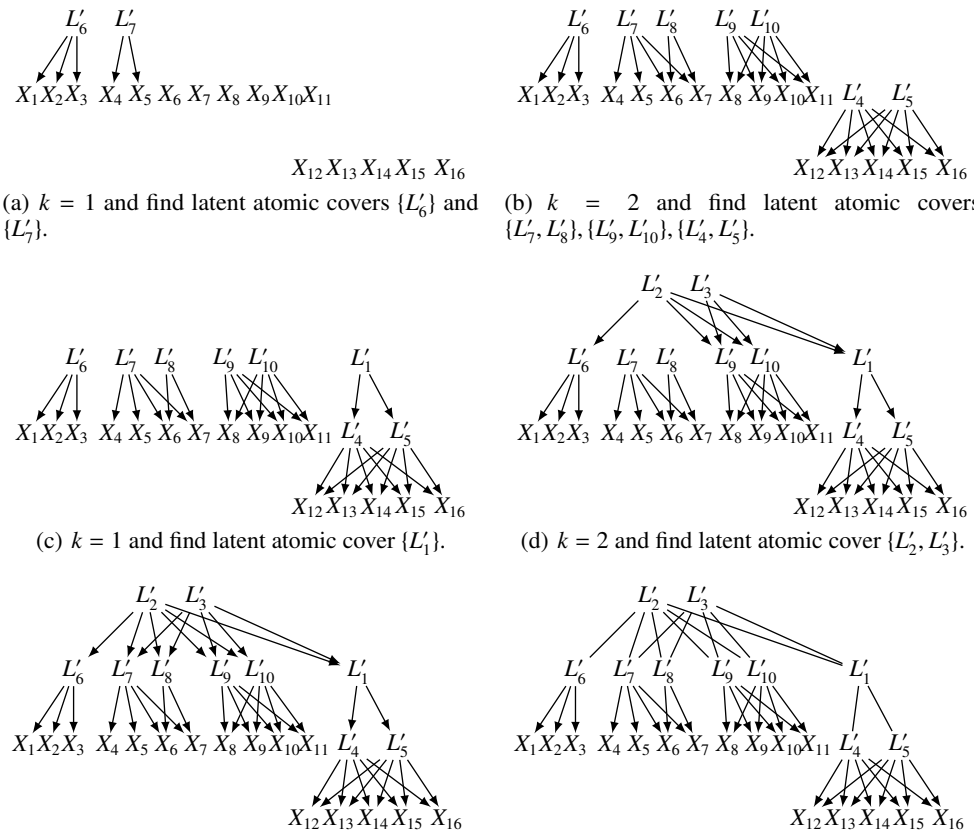
(b) After applying the minimal-graph operator to the graph in (a), L_4 is merged to its parent L_5, and the rank constraints do not change.

(c) After applying the skeleton operator to the graph in (b), L_1 has an edge to X_7 and L_3 has an edge to X_1, and the rank constraints do not change.

Figure 11: Examples of applying the minimal-graph operator and the skeleton operator to an IL^{2}H graph.

E.7 Illustrative Examples of the Entire Algorithm

Figure 12 and Figure 13 give two illustrative examples of the entire algorithm, showing how each step proceeds.



(a) $k = 1$ and find latent atomic covers $\{L'_6\}$ and $\{L'_7\}$.

(b) $k = 2$ and find latent atomic covers $\{L'_7, L'_8\}, \{L'_9, L'_{10}\}, \{L'_4, L'_5\}$.

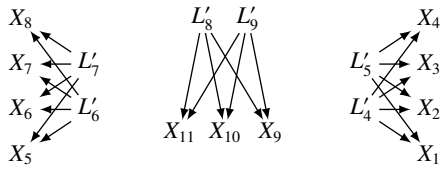
(c) $k = 1$ and find latent atomic cover $\{L'_1\}$.

(d) $k = 2$ and find latent atomic cover $\{L'_2, L'_3\}$.

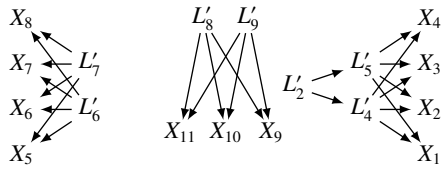
(e) Connect $\{L'_2, L'_3\}$ to $\{L'_7, L'_8\}$ and *findCausalClusters* ends.

(f) There is no change after *refineClusters*, and then *refineEdges* refines the edges and outputs the Markov equivalence class.

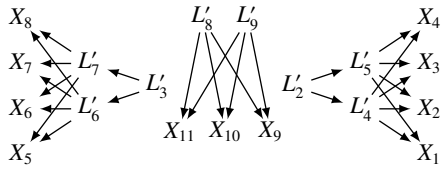
Figure 12: An illustrative example by applying Algorithm 1 to the measured variables in Figure 7(a).



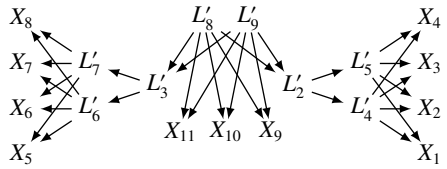
(a) $k = 2$ and find latent atomic covers $\{L'_6, L'_7\}$, $\{L'_4, L'_5\}$, and $\{L'_8, L'_9\}$.



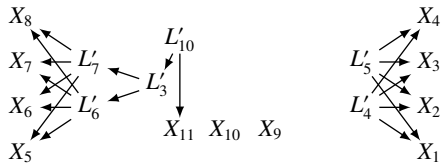
(b) $k = 1$ and find a latent atomic cover $\{L'_2\}$.



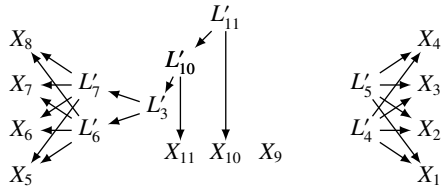
(c) $k = 1$ and find a latent atomic cover $\{L'_3\}$.



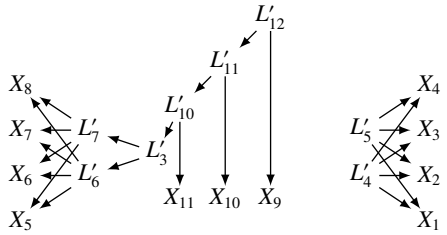
(d) Connect $\{L'_8, L'_9\}$ to $\{L'_2\}$ and to $\{L'_3\}$, and *findCausalClusters* ends.



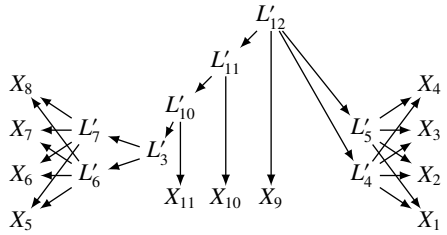
(e) Refine $\{L'_2\}$ by first removing $\{L'_2\}$ and its parents $\{L'_8, L'_9\}$ and performing *findCausalClusters* over $\{L'_3, L'_4, L'_5, X_9, X_{10}, X_{11}\}$, and then we can find a latent cover $\{L'_{10}\}$.



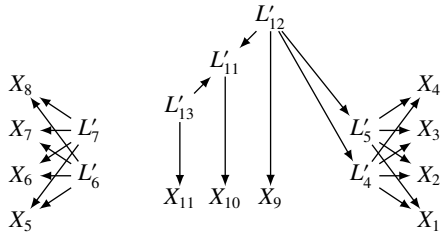
(f) Next perform *findCausalClusters* over $\{L'_{10}, L'_4, L'_5, X_9, X_{10}\}$ we can find a latent cover $\{L'_{11}\}$.



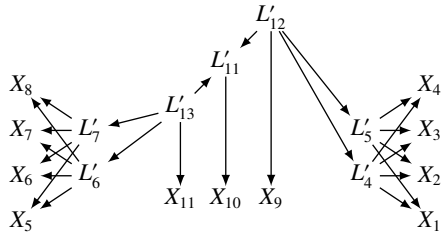
(g) Next perform *findCausalClusters* over $\{L'_{11}, L'_4, L'_5, X_9\}$ we can find a latent cover L'_{12}.



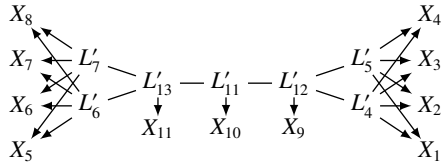
(h) Connect $\{L'_{12}\}$ to $\{L'_4, L'_5\}$.



(i) Refine $\{L'_3\}$ by first removing $\{L'_3\}$ and its parents $\{L'_{10}\}$ and performing *findCausalClusters* over $\{L'_6, L'_7, L'_{11}, X_{11}\}$, and then we can find a latent cover $\{L'_{13}\}$.



(j) Connect $\{L'_{13}\}$ to $\{L'_6, L'_7\}$.



(k) Perform *refineEdges* to refine the edges and output the Markov equivalence class.

Figure 13: An illustrative example by applying Algorithm 1 to the measured variables in Figure 8(c).

References

Myung Jin Choi, Vincent Y.F. Tan, Animashree Anandkumar, and Alan S. Willsky. Learning latent tree graphical models. *Journal of Machine Learning Research*, 12(49):1771–1812, 2011. URL `http://jmlr.org/papers/v12/choi11b.html`.

Erich Kummerfeld and Joseph Ramsey. Causal clustering for 1-factor measurement models. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1655–1664. ACM, 2016.

Feng Xie, Ruichu Cai, Biwei Huang, Clark Glymour, Zhifeng Hao, and Kun Zhang. Generalized independent noise condition for estimating latent variable causal graphs. In *Advances in Neural Information Processing Systems*, pages 14891–14902, 2020.

Peter Spirtes, Clark N Glymour, and Richard Scheines. *Causation, Prediction, and Search*. MIT press, 2000.

David Maxwell Chickering. Optimal structure identification with greedy search. *Journal of machine learning research*, 3(Nov):507–554, 2002.

Shohei Shimizu, Patrik O. Hoyer, Aapo Hyvärinen, and Antti Kerminen. A linear non-gaussian acyclic model for causal discovery. *J. Mach. Learn. Res.*, 7:2003–2030, December 2006. ISSN 1532-4435.

Patrik O Hoyer, Dominik Janzing, Joris M Mooij, Jonas Peters, and Bernhard Schölkopf. Nonlinear causal discovery with additive noise models. In *Advances in neural information processing systems*, pages 689–696, 2009.

Kun Zhang and Aapo Hyvärinen. On the identifiability of the post-nonlinear causal model. In *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*, pages 647–655. AUAI Press, 2009.

Diego Colombo, Marloes H Maathuis, Markus Kalisch, and Thomas S Richardson. Learning high-dimensional directed acyclic graphs with latent and selection variables. *The Annals of Statistics*, pages 294–321, 2012.

Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, New York, NY, USA, 2000. ISBN 0-521-77362-8.

Sina Akbari, Ehsan Mokhtarian, AmirEmad Ghassami, and Negar Kiyavash. Recursive causal structure learning in the presence of latent variables and selection bias. In *Advances in Neural Information Processing Systems*, volume 34, pages 10119–10130, 2021.

Ricardo Silva, Richard Scheine, Clark Glymour, and Peter Spirtes. Learning the structure of linear latent variable models. *Journal of Machine Learning Research*, 7(Feb):191–246, 2006.

S. Wang. Causal clustering for 1-factor measurement models on data with various types. *arXiv preprint arXiv:2009.08606*, 2020.

Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988. ISBN 1558604790.

V. Chandrasekaran, S. Sanghavi, P. A. Parrilo, and A. S. Willsky. Rank-sparsity incoherence for matrix decomposition. *SIAM Journal on Optimization*, 21(2):572–596, 2011.

V. Chandrasekaran, P. A. Parrilo, and A. S. Willsky. Latent variable graphical model selection via convex optimization. *Annals of Statistics*, 40(4):1935–1967, 2012.

Animashree Anandkumar, Daniel Hsu, Adel Javanmard, and Sham Kakade. Learning linear bayesian networks with latent variables. In *International Conference on Machine Learning*, pages 249–257, 2013.

Shohei Shimizu, Patrik O Hoyer, and Aapo Hyvärinen. Estimation of linear non-gaussian acyclic models for latent factors. *Neurocomputing*, 72(7-9):2024–2027, 2009.

Doris Entner and Patrik O Hoyer. Discovering unconfounded causal relationships using linear non-gaussian models. In *JSAI International Symposium on Artificial Intelligence*, pages 181–195. Springer, 2010.

Tatsuya Tashiro, Shohei Shimizu, Aapo Hyvärinen, and Takashi Washio. ParceLiNGAM: a causal ordering method robust against latent confounders. *Neural Computation*, 26(1):57–83, 2014.

B. Huang*, K. Zhang*, J. Zhang, R. Sanchez-Romero, C. Glymour, and B. Schölkopf. Causal discovery from heterogeneous/nonstationary data. In *JMLR*, volume 21(89), 2020.

Bohdan Kivva, Goutham Rajendran, Pradeep Ravikumar, and Bryon Aragam. Learning latent causal graphs via mixture oracles. *Advances in Neural Information Processing Systems*, 34, 2021.

Seth Sullivant, Kelli Talaska, and Jan Draisma. Trek separation for Gaussian graphical models. *The Annals of Statistics*, 38(3):1665 – 1685, 2010. doi: 10.1214/09-AOS760. URL `https://doi.org/10.1214/09-AOS760`.