# Local Causal Structure Learning in the Presence of Latent Variables

**Feng Xie** [1] **Zheng Li** [1] **Peng Wu** [1] **Yan Zeng** [1] **Chunchen Liu** [2] **Zhi Geng** [1]

## Abstract

Discovering causal relationships from observational data, particularly in the presence of latent variables, poses a challenging problem. While current local structure learning methods have proven effective and efficient when the focus lies solely on the local relationships of a target variable, they operate under the assumption of causal sufficiency. This assumption implies that all the common causes of the measured variables are observed, leaving no room for latent variables. Such a premise can be easily violated in various real-world applications, resulting in inaccurate structures that may adversely impact downstream tasks. In light of this, our paper delves into the primary investigation of locally identifying potential parents and children of a target from observational data that may include latent variables. Specifically, we harness the causal information from m-separation and V-structures to derive theoretical consistency results, effectively bridging the gap between global and local structure learning. Together with the newly developed stop rules, we present a principled method for determining whether a variable is a direct cause or effect of a target. Further, we theoretically demonstrate the correctness of our approach under the standard causal Markov and faithfulness conditions, with infinite samples. Experimental results on both synthetic and real-world data validate the effectiveness and efficiency of our approach.

## 1. Introduction

Inferring causal relations, known as causal discovery, has drawn much attention in several fields, such as computer science (Jonas et al., 2017; Pearl, 2018; Schölkopf, 2022), social science (Spirtes et al., 2000), epidemiology (Hernán & Robins, 2010), biology (Glymour et al., 2019), and neuroscience (Smith et al., 2011; Sanchez-Romero et al., 2019). The discovered causal relationships are useful for predicting the behavior of a system under external interventions, which is a crucial step in both understanding and manipulating that system (Pearl, 2009). Learning such relations from purely observational data is challenging, especially when latent confounders can be present (Spirtes & Zhang, 2016).

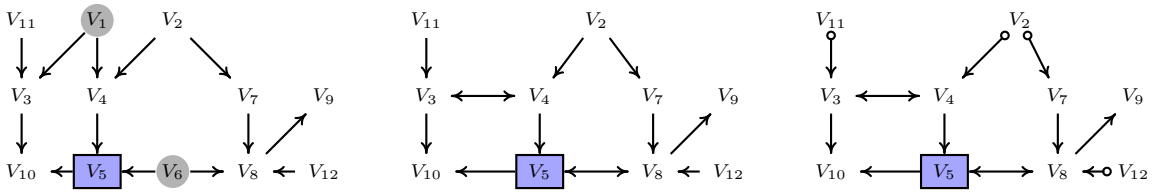
There exists work in the literature that has attempted to recover causal structure among observed variables in the presence of latent variables. Spirtes et al. (2000) proposed the seminal FCI (Fast Causal Inference) algorithm that can learn a partial ancestral graph (PAG) [1] in the presence of latent variables by performing conditional independence tests. Later, a faster algorithm, called Really Fast Causal Inference (RFCI), was developed (Colombo et al., 2012). Other interesting developments along this line include (Claassen & Heskes, 2011; Claassen et al., 2013; Raghu et al., 2018; Akbari et al., 2021; Mokhtarian et al., 2023). These works focus on learning the whole causal graph rather than the local causal graph. However, in many real-world scenarios, researchers are usually interested in the local causal relationships (Walters et al., 2007; Peter & Davidson, 2011; Ma et al., 2023).

Several contributions have been made to learn the local causal structure other than the global causal structure. For instance, the Local Causal Discovery (LCD) algorithm (Cooper, 1997) and its variants (Silverstein et al., 2000; Mani & Cooper, 2004) are proposed to find causal edges among every four-variable set in a causal graph. Although these algorithms primarily aim to identify a subset of causal edges through specific structures among all variables, our focus is on discovering all causal edges adjacent to a single target variable. Yin et al. (2008) and Zhou et al. (2010) designed the PCD-by-PCD to find sets of parents, children, and maybe some of the descendants (PCD) of variables of the target variable. Later, Wang et al. (2014) proposed a more efficient approach, called MB-by-MB, for discovering direct cause and effect variables of the target. Additional

[1]Department of Applied Statistics, Beijing Technology and Business University, Beijing, China [2]LingYang Co.Ltd, Alibaba Group, Hangzhou, China. Correspondence to: Yan Zeng <yanazeng013@btbu.edu.cn>.

*Proceedings of the 41st International Conference on Machine Learning*, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

[1]A PAG represents a Markov equivalence class of maximal ancestral graphs (MAGs) which encode the causal relations among the observed variables. See the example in Figure 1 or Section 3.1 for more details.




(a) DAG (b) MAG (c) PAG

*Figure 1.* (a) Underlying causal DAG from a selected part of ANDES network (Conati et al., 1997), where $V_1$ and $V_6$ are hidden and $V_5$ is the target variable of interest. (b) The corresponding MAG of the DAG in (a). (c) The inferred PAG from observed variables .

significant contributions to this field have been made, including the Causal Markov Blanket (CMB) algorithm (Gao & Ji, 2015), the Efficient Local Causal Structure (ELCS) algorithm (Yang et al., 2021), and the GradieNt-based LCS (GraN-LCS) algorithm (Liang et al., 2023). Although these methods have been used in a range of fields, they usually assume the assumption of causal sufficiency, i.e., we have measured all the common causes of the measured variables in the system. However, in various real-world scenarios, including Gene Expression network (Wille et al., 2004), , etc, this assumption is usually violated.

In this paper, we address the challenge of locally learning the causes and effects of a given target variable in a more complex scenario where certain variables may be unmeasured. Specifically, our primary contributions can be summarized in three key aspects:

1. We propose a novel MMB-by-MMB algorithm for learning the direct causes and effects of a target variable based only on the estimated local structure, allowing for the existence of latent variables.
2. We theoretically demonstrate that the proposed algorithm is a complete local discovery algorithm and can identify the same direct causes and effects for a target variable as global methods under standard assumptions.
3. We conduct extensive experiments and demonstrate the efficacy of our algorithm on both benchmark network structures and real-world data.

## 2. Related Works

This paper focuses on local causal structure (LCS) learning. Our investigation intersects with broader themes, such as global causal structure (GCS) learning and Markov Blanket (MB) learning. In this context, we here provide a brief review of these three interconnected areas. For a comprehensive review of causal structure learning or MB learning, see (Spirtes & Zhang, 2016; Heinze-Deml et al., 2018; Yu et al., 2020; Kitson et al., 2023)

**LCS learning.** Existing LCS learning methods can be roughly divided into two categories, namely Y-structure-based methods including LCD algorithm (Cooper, 1997) and its variants (Silverstein et al., 2000; Mani & Cooper, 2004; Versteeg et al., 2022), and constraint-based ones such as PCD-by-PCD (Yin et al., 2008; Zhou et al., 2010), MB-by-MB (Wang et al., 2014), CMB (Gao & Ji, 2015), ELCS (Yang et al., 2021), and GraN-LCS algorithm (Liang et al., 2023). Methods in the first category typically focus on learning causal edges among sets of four variables, while our approach targets all causal edges related to a specific variable. Moreover, methods in the second category generally assume that all common causes of the measured variables are observed, an assumption not required by our method.

**GCS learning.** When latent confounding is present, well-known algorithms along this line include the seminal FCI algorithm (Spirtes et al., 2000), RFCI (Colombo et al., 2012), FCI$^+$ (Claassen et al., 2013), and its variants (Claassen & Heskes, 2011; Ogarrio et al., 2016; Raghu et al., 2018; Akbari et al., 2021). Some further studies are also conducted by introducing the data-generating mechanism (Chen et al., 2023; Kaltenpoth & Vreeken, 2023; Chen et al., 2021) or distribution of data (Hoyer et al., 2008; Salehkaleybar et al., 2020; Maeda & Shimizu, 2020; Cai et al., 2023). While these algorithms are efficient in their operation, identifying the global structure can be unnecessary and wasteful when our primary interest lies in understanding the local structure surrounding a single target variable, which can be clearly observed in the $nTest$ in our experimental results.

**MB learning.** MB learning algorithm aims to learn parents, children, and spouses of target T simultaneously. Along this line include GSMB (Margaritis & Thrun, 1999), IAMB (Tsamardinos & Aliferis, 2003), Fast-IAMB (Yaramakala & Margaritis, 2005), and Total Conditioning (TC)(Pellet & Elisseeff, 2008), and other variants (Aliferis et al., 2003; Pena et al., 2007; Gao & Ji, 2016; Wu et al., 2019). Recently, Yu et al. (2018) proposed an algorithm, M3B to mine the MAG MB (MMB) of a target variable in MAGs. However, the above methods do not distinguish parents from children. In contrast, our method has to differentiate the direct parents (cause) and children (effect).

To the best of our knowledge, there is currently no method for learning the local causal structure in the presence of latent confounders that can effectively identify the direct

causes and effects of a target variable under standard assumptions.

## 3. Preliminaries

### 3.1. Graph Terminology and Notations

**Ancestral Graphs.** A ***mixed graph*** $\mathcal{G}$ over the set of vertices $\mathbf{V}$ containing three types of edges between pairs of nodes: directed edges ($\rightarrow$), bi-directed edges ($\leftrightarrow$), and undirected edges ($-$). A is a ***spouse*** of B if $A \leftrightarrow B$ is in $\mathcal{G}$. A mixed graph is ***ancestral*** if it doesn't contain a directed or almost directed cycle [2]. Let $\mathcal{V}$ be any subset of vertices in $\mathcal{G}$. An ***inducing path*** relative to $\mathcal{V}$ is a path on which every vertex not in $\mathcal{V}$ (except for the endpoints) is a collider on the path and every collider is an ancestor of an endpoint of the path. An ancestral graph is a ***Maximal Ancestral Graph (MAG)*** $\mathcal{M}$ if there is no inducing path between any two non-adjacent vertices. A MAG is called a directed acyclic graph (DAG) if it has only directed edges. A causal MAG represents a set of causal models with the same set of observed variables that entail the same independence and ancestral relations among the observed variables. Two MAGs are called ***Markov equivalent*** if they impose the same independence model. A ***Partial Ancestral Graph (PAG)***$\mathcal{P}$ represents an equivalence class of MAGs $[\mathcal{M}]$. A partial ancestral graph for $[\mathcal{M}]$ is a graph $\mathcal{P}$ with possible three kinds of marks ($\circ$, $>$, $-$)[3], such that 1) $\mathcal{P}$ has the same adjacencies as $\mathcal{M}$ (and hence any member of $[\mathcal{M}]$) does, and every non-circle mark in $\mathcal{P}$ is an invariant mark in $[\mathcal{M}]$. For convenience, we use an asterisk (*) to denote any possible mark of a PAG ($\circ, >, -$) or a MAG ($>, -$).

**Definition 1** (m-separation)**.** *In a mixed graph $\mathcal{G}$, a path $p$ between vertices $X$ and $Y$ is active (m-connecting) relative to a (possibly empty) set of vertices $\mathbf{Z}$ ($X, Y \notin \mathbf{Z}$) if 1) every non-collider on $p$ is not a member of $\mathbf{Z}$, and 2) every collider on $p$ has a descendant in $\mathbf{Z}$.*

A set $\mathbf{Z}$ m-separates $\mathbf{X}$ and $\mathbf{Y}$ in $\mathcal{G}$, denoted by $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y}|\mathbf{Z})_{\mathcal{G}}$, if there is no active path between any vertices in $\mathbf{X}$ and any vertices in $\mathbf{Y}$ relative to $\mathbf{Z}$. The criterion of m-separation is a generalization of Pearl's d-separation criterion in DAG to ancestral graphs. Two MAGs are called Markov equivalent if they impose the same m-separations.

Related concepts used here can be found in sources (Richardson & Spirtes, 2002; Zhang, 2008b).

**Markov Blanket.** In a DAG, the ***Markov blanket*** of a vertice $T$, noted $MB(T)$, is the set of parents, children, and children's parents (spouses) of $T$. In a MAG, the Markov blanket of a vertice $T$, noted as $MMB(T)$, consists of 1) parents of $T$; 2) children of $T$; and 3) a set of variables that for $\forall V_i$ within the set, $V_i$ is not adjacent to $T$ and has a collider path to $T$. See the example in the Definition 2.

**Notations.** Given a graph $\mathcal{G}$, two vertices are said to be adjacent in $\mathcal{G}$ if there is an edge between them. We use $Adj(T)$ to denote the set of adjacent vertices of $T$. $X$ is called an ancestor of $Y$ and $Y$ a descendant of $X$ if there is a directed path from $X$ to $Y$ or $X = Y$. We use $Pa(T)$, $Ch(T)$, $Sp(T)$, $An(T)$, $De(T)$ to denote the set of ***parents, children, spouses, ancestors***, and ***descendants*** of vertex $T$ in $\mathcal{G}$, respectively. We use the notation $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y}|\mathbf{Z})_P$ for "$\mathbf{X}$ is statistically independent of $\mathbf{Y}$ given $\mathbf{Z}$", and $(\mathbf{X} \not\perp\!\!\!\perp \mathbf{Y}|\mathbf{Z})_P$ for the negation of the same sentence (Dawid, 1979). We drop the subscript $P$ whenever it is clear from context. We use $MMB^{+}(T)$ to denote the set of $\{MMB(T) \cup T\}$.

**Standard Assumption.** In terms of m-separation, the ***causal Markov condition*** says that m-separation in a graph $\mathcal{G}$ implies conditional independence in the population distribution. The ***causal Faithfulness condition*** says that m-connection in a graph $\mathcal{G}$ implies conditional dependence in the population distribution (Zhang, 2008a). Under the above two conditions, conditional independence relations among the observed variables correspond exactly to m-separation in the MAG $\mathcal{G}$, i.e., $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y}|\mathbf{Z})_P \Leftrightarrow (\mathbf{X} \perp\!\!\!\perp \mathbf{Y}|\mathbf{Z})_{\mathcal{G}}$.

**Identification of Global Learning for PAG.** Assuming the causal Markov condition and the causal Faithfulness condition, the PAG (that represents an equivalence class of MAGs) can be uniquely identified by using the independence-constraint-based algorithm, such as FCI (Spirtes et al., 2000; Zhang, 2008b), from an oracle of conditional independence relations.

### 3.2. Problem Definition

We consider a Structural Causal Model (SCM) (Pearl, 2009) with the set of variables $\mathbf{V} = \mathbf{O} \cup \mathbf{L}$, and the joint distribution $P(\mathbf{V})$, where $\mathbf{O}$ and $\mathbf{L}$ denote the set of observed variables and latent variables, respectively. We here assume that there is no selection bias in the system. Thus, the SCM is associated with a DAG where each node is a variable in $\mathbf{V}$ and each edge is a function $f$. That is to say, each variable $V_i \in \mathbf{V}$ is generated as $V_i = f_i(Pa(V_i), u_i)$, where $u_i$ represent errors (or "disturbances") due to omitted factors, and all errors are independent from each other.

**Goal.** Given a target variable $T \in \mathbf{O}$, we are interested in the local structure of the target variable. In particular, our goal is to establish the local criteria for identifying the potential direct causes and effects of a target based only on the local structure instead of the entire graph.

[2] An almost directed cycle happens when A is both a spouse and an ancestor of B.

[3] $\circ$ represents undetermined edge marks.

## 4. MMB-by-MMB Algorithm

In this section, we propose a sequential algorithm, MMB-by-MMB, for discovering the direct causes and effects of a target variable $T$. We use **Waitlist** to store nodes that are potentially relevant for identifying the direct causes and effects of $T$. Let **Donelist** store nodes removed from **Waitlist** and $\mathcal{P}$ store valid causal information.

**Basic idea.** The process proceeds through a series of sequential repeats. Initially, **Waitlist** $= \{T\}$ and **Donlist** is empty. In each iteration, we focus on the first variable $V_X$ in the **Waitlist**. Specifically, we first learn the MMB of $V_X$ and a local causal structure $\mathcal{L}_{MMB^+(V_X)}$ based on this MMB. Subsequently, employing m-separation (see Theorem 1) and V-structure (see Theorem 2), we select true causal information from $\mathcal{L}_{MMB^+(V_X)}$ and store them in $\mathcal{P}$. Next, $\mathcal{P}$ is oriented using standard orientation criteria. Finally, we update **Waitlist** and **Donlist**. When the stop rules are met (see Theorem 3), the algorithm stops running. The specific pseudocode is provided in Algorithm 1 and 2.

We outline the principle of the algorithm in Section 4.1. Furthermore, we present the detailed steps of the algorithm in Section 4.2. Under standard assumptions, we show that the proposed algorithm can locally identify the same direct causes and effects for a target variable as global learning methods. Finally, in Section 4.3, we analyze the complexity of the algorithm. To improve readability, we defer all proofs to Appendix C.

### 4.1. Principle of the Algorithm

In this section, we present the theoretical results that serve as the principle for our sequential approach. Specifically, we answer the following 3 questions:

$\mathcal{Q}1$. What causal information of ***m-separation*** in local structure learning is consistent with those in global learning?

$\mathcal{Q}2$. What causal information of ***V-structures*** in local structure learning is consistent with those in global learning?

$\mathcal{Q}3$. How to design a ***stop*** criteria to ensure that our local learning structure is consistent with the global one?

We first give the following theorem about m-separation in both local and global learning, which answers question $\mathcal{Q}1$.

**Theorem 1** (**M-separation**). *Let $T$ be any node in $\mathbf{O}$, and $X$ be a node in $MMB(T)$. Then $T$ and $X$ are m-separated by a subset of $\mathbf{O} \setminus \{T, X\}$ if and only if they are m-separated by a subset of $MMB(T) \setminus \{X\}$.*

Theorem 1 implies that the existence of an edge connecting $T$ to any other node $X \in MMB(T)$ can be equivalently determined through both the full distribution of $\mathbf{O}$ and the marginal distribution of $MMB^+(T)$. Consequently, it becomes feasible to accurately assess the presence of these connecting edges to $T$ by utilizing the observed data from $MMB^+(T)$.

**Example 1.** *Consider the MAG shown in Figure 1(b). Let $V_5$ be the target variable $T$. Suppose that we can correctly check conditional independencies from data and thus find the $MMB(V_5)$, i.e., $MMB(V_5) = \{V_3, V_4, V_7, V_8, V_{10}, V_{12}\}$. According to Theorem 1, we deduce the existence of edges $V_5 \circ\!\!\circ V_4$, $V_5 \circ\!\!\circ V_8$ and $V_5 \circ\!\!\circ V_{10}$, while there are no connecting edges between $V_5$ and $V_3$, $V_7$, or $V_{12}$. These results are consistent with the conclusions of global learning.*

**Remark 1.** *It is noteworthy that the connecting edges between nodes in $MMB(T)$ through the marginal distribution of $MMB^+(T)$ do not align with those identified through the full distribution of $\mathbf{O}$. For instance, considering the connection between $V_4$ and $V_7$, we will obtain the spurious edge $V_4 \circ\!\!-\!\!\circ V_7$ from $MMB^+(T)$. However, because $V_4 \perp\!\!\!\perp V_7 | V_2$, we know there is no direct edge between $V_4$ and $V_7$ in the ground-truth MAG.*

Next, we discuss the solution for the question $\mathcal{Q}2$, and the illustrative examples are given accordingly. Let $\mathcal{V}$ be a subset of $\mathbf{V}$. We say that a V-structure $X \rightarrow Z \leftarrow Y$ can be identified or found by the marginal distribution $P(\mathcal{V})$ if the conditional independence and dependence of the V-structure can be checked in the $P(\mathcal{V})$, i.e., $X \perp\!\!\!\perp Y | \mathbf{S}$ and $X \not\perp\!\!\!\perp Y | \mathbf{S} \cup \{Z\}$ for $\{X, Y, Z\} \cup \mathbf{S} \subseteq \mathcal{V}$.

**Theorem 2** (**Fully Correct V-structures**). *Consider a sub-MAG of $\mathcal{M}'$ over $MMB^+(T)$. Let $V_a, V_b$ be two nodes in $MMB(T)$. The following statements hold.*

$\mathcal{S}1$. *The V-structure $V_a * \!\!\rightarrow T \leftarrow\!\!* V_b$ that identified by the marginal distribution of $MMB^+(T)$ are true V-structures in the ground-truth MAG $\mathcal{M}$.*

$\mathcal{S}2$. *The V-structure $T * \!\!\rightarrow V_a \leftarrow\!\!* V_b$ can be successfully identified by the marginal distribution of $MMB^+(T)$.*

Statement $\mathcal{S}1$ shows that if $T$ is a collider in the identified V-structures using the observational data of $MMB^+(T)$, then these V-structures are equivalent determined by the full observational data of $\mathbf{O}$. Statement $\mathcal{S}2$ says that a special type of V-structure, in which the collider $V_a$ within the V-structure is not an ancestor of $T$, can certainly be identified from the observational data of $MMB^+(T)$.

**Example 2** (Statements $\mathcal{S}1$ and $\mathcal{S}2$). *Continue to consider the causal diagram shown in Figure 1(b). We have known $MMB^+(V_5) = \{V_5, V_3, V_4, V_7, V_8, V_{10}, V_{12}\}$. According to Statement $\mathcal{S}1$, we can determine the V-structure $V_4 * \!\!\rightarrow V_5 \leftarrow\!\!* V_8$ form the marginal distribution of $MMB^+(V_5)$, since $V_4 \perp\!\!\!\perp V_8 | V_7$ and $V_4 \not\perp\!\!\!\perp V_8 | \{V_7, V_5\}$. Furthermore, according to Statement $\mathcal{S}2$, we can obtain the V-structure $V_5 * \!\!\rightarrow V_8 \leftarrow\!\!* V_7$ from the marginal distribution of $MMB^+(V_5)$, since $V_5 \perp\!\!\!\perp V_7 | V_4$ and $V_5 \not\perp\!\!\!\perp V_7 | \{V_4, V_8\}$.*

**Remark 2.** *Theorem 2 merely states that the locally identified V-structures containing T are correct. That is to say,*

*during the orientation of local structures, some V-structures may be incorrect or missing, as stated in the following two examples.*

1. *V-structures identified that do not include $T$ cannot be guaranteed to be correct from the observational data of $MMB^+(T)$. For instance, consider the graph shown in Figure 1.(b), one may obtain $V_3 {*\!\!\rightarrow} V_4 \leftarrow\!\!* V_7$ from the observational data of $MMB^+(V_5)$ since $V_3 \perp\!\!\!\perp V_7 | \emptyset$ and $V_3 \not\perp\!\!\!\perp V_7 | V_4$. However, the V-structure $V_3 {*\!\!\rightarrow} V_4 \leftarrow\!\!* V_7$ is not entirely accurate. In global structural learning, the identified V-structure is $V_3 {*\!\!\rightarrow} V_4 \leftarrow\!\!* V_2$.*
2. *Not all V-structures that include collider $T$ can be identified from the observational data of $MMB^+(T)$. For instance, let $V_{10}$ be the target variable in Figure 1(b). Then we have $MMB(V_{10}) = \{V_3, V_5\}$. The V-structure $V_3 {*\!\!\rightarrow} V_{10} \leftarrow\!\!* V_5$ is unidentifiable within $MMB^+(V_{10})$, as the separating set for $V_3$ and $V_5$ (i.e., $V_4$) is not encompassed in $MMB(V_{10})$.*

We now address the last question $\mathcal{Q}3$ about the criteria for stopping rules.

**Theorem 3** (**Stop Rules**). *Let $T$ be the target node of interest within $\mathbf{O}$ and $\mathbf{Waitlist}$ represent the collection of nodes that need to be checked by Theorem 1 and Theorem 2. If any of the subsequent rules are met, the local structure identified for $T$, encompassing its direct causes and effects, will be equivalent to the structure identified through global learning methods.*

$\mathcal{R}1$. *The structures around the target $T$ are all determined.*
$\mathcal{R}2$. *The* $\mathbf{Waitlist}$ *is empty.*
$\mathcal{R}3$. *All paths from the target $T$, which include undirected edges (connected to the target $T$), are blocked by edges $*\!\!\rightarrow$.*

Rules $\mathcal{R}1$ and $\mathcal{R}2$ are both direct stopping criteria, meaning that all causal information of interest has been identified, or all nodes have been fully utilized. Roughly speaking, R3 states that when the paths connecting the surrounding undirected edges of $T$ are blocked by directed edges ($*\!\!\rightarrow$), integrating the joint distribution of the remaining nodes connected on these paths is equivalent to omitting these nodes (see Lemma 2 in the appendix). In other words, continuing to learn the local structure of the remaining nodes connected on these paths will not help determine the direction of the undirected edges around $T$.

Below, we give an example to illustrate the rule $\mathcal{R}3$ in Theorem 3.

**Example 3** ($\mathcal{R}3$). *Consider the graph shown in Figure 2(a), where $T$ is the target variable. Assuming that we have already checked nodes $T$ according to Theorem 1 and Theorem 2, we then obtain subgraph (b). Note that the left tail of edge $T \circ\!\!\rightarrow V_1$ is not directed. However, we can find that the path $T \circ\!\!\rightarrow V_1$ is blocked by the edge $\circ\!\!\rightarrow$. According to $\mathcal{R}3$ of Theorem 3, we will stop the learning process, even though nodes $V_3$ and $V_4$ have not yet been checked.*

(a) Underlying MAG (b) The final local PAG around $T$

*Figure 2.* The illustrative example for $\mathcal{R}3$ in Theorem 3.

**Algorithm 1** MMB-by-MMB

**Input:** Target $T$, observed data $\mathbf{O}$ of $\mathbf{V}$
Initialize : $\mathbf{Waitlist} := \{T\}$, $\mathbf{Donelist} = \emptyset$, $\mathcal{P} = \emptyset$.
**repeat**
  $V_X \leftarrow$ the head node of $\mathbf{Waitlist}$;
  $MMB^+(V_X) \leftarrow MMB_{alg}(V_X)$ ;
  **if** $\exists V_Y \in \mathbf{Donelist}, MMB^+(V_X) \subseteq MMB^+(V_Y)$ **then**
    $\mathcal{L}_X \leftarrow$ the substructure of $\mathcal{L}_Y$ over $MMB^+(X)$;
  **else if** $MMB(V_X) \subseteq \mathbf{Donelist}$ **then**
    $\mathcal{L}_X \leftarrow$ the substructure of $\mathcal{P}$ over $MMB^+(X)$;
  **else**
    Learn $\mathcal{L}_X$ over $MMB^+(X)$.
  **end if**
  $\mathcal{P} \leftarrow$ select the edges connected to $V_X$ and the V-structures containing $V_X$.
  $\mathcal{P} \leftarrow$ orient maximally the edge marks using the orientation rules of Zhang (2008b).
  Add $V_X$ to $\mathbf{Donelist}$, and remove $V_X$ from the $\mathbf{Waitlist}$.
  Add $\{Adj(V_X) \setminus (\mathbf{Waitlist} \cup \mathbf{Donelist})\}$ to $\mathbf{Waitlist}$
**until** One of the stop Rules $\mathcal{R}1 \sim \mathcal{R}3$ is met
**Output:** The local structure $\mathcal{P}$ around $T$

**Algorithm 2** $MMB_{alg}$(Pellet & Elisseeff, 2008)

**Input:** Target $V_x$, observed data $\mathbf{O}$ of $\mathbf{V}$
Initialize : $MMB(V_X) := \emptyset$.
**for** each $V_Y \in \mathbf{O} \setminus V_X$ **do**
  **if** $V_X \not\perp\!\!\!\perp V_Y \mid \mathbf{O} \setminus \{V_X, V_Y\}$ **then**
    Add $V_Y$ to $MMB(V_X)$
  **end if**
**end for**
$MMB^+(V_X) = \{V_X \cup MMB(V_X)\}$
**Output:** $MMB^+(V_X)$

### 4.2. Our Sequential Approach

In this section, we leverage the above theoretical results and propose a sequential algorithm to learn the potential parents and children of a target node in models that include latent variables. We use $\mathcal{L}_\mathcal{V}$ to denote the local structure

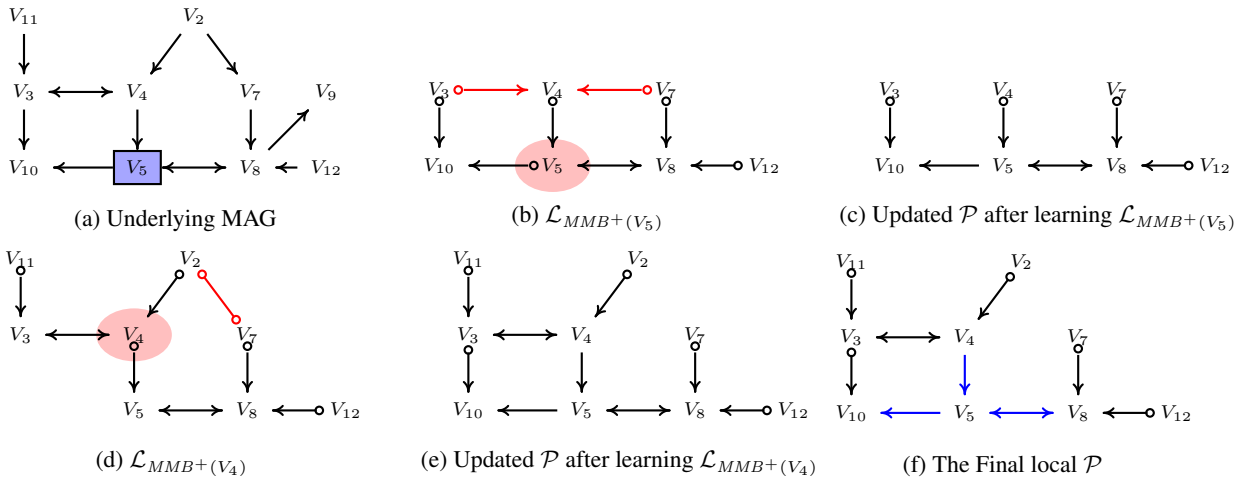



*Figure 3.* The sequential process for finding the parents and children of the target $V_5$ in the graph of Figure 1 (a), where the red edges indicate that the current local results cannot be guaranteed to be consistent with the global learning results.

learned from a subset $\mathcal{V}$ of $\mathbf{V}$, utilizing the test of conditional independence and orientation of V-structures. We use $\mathcal{P}$ to store the local causal structure around $T$, where the causal information preserved in this structure is consistent with the global learning structure. Roughly speaking, for target $T$, our method contains the following key steps:

**S1.** Finding a MAG Markov blanket $MMB(T)$ of the target $T$ and learning the local structure $\mathcal{L}_{MMB^+(T)}$.

**S2.** Putting the edges connected to $T$ and the V-structures containing $T$ in $\mathcal{L}_{MMB^+(T)}$ to $\mathcal{P}$, according to Theorem 1 and Theorem 2.

**S3.** Orienting maximally the edge marks in $\mathcal{P}$ using the standard orientation rules of Zhang (2008b).

Three steps S1 $\sim$ S3 are repeated sequentially until any one of the stop rules is met in Theorem 3. For notational convenience, let **WaitList** be the list of nodes to be checked by Theorem 1 and Theorem 2. Let **DoneList** denote the list of nodes whose local structures have been found. Additionally, $MMB_{alg}$ refers to the algorithm used for learning $MMB$. The complete procedure is summarized in Algorithm 1 and 2, and a complete example is given in Example 4.

**Example 4.** *We illustrate our MMB-by-MMB algorithm with the causal MAG in Figure 3(a). We assume Oracle tests for conditional independence tests. In this structure, There are latent variables between nodes $V_3$ and $V_4$, $V_5$ and $V_8$. We here are interested in the local structure around $V_5$. The learning process is as follows:*

- *It first initializes sets* **Waitlist** $= \{V_5\}$, **Donelist** $= \emptyset$, *and graph $\mathcal{P} = \emptyset$ (Line 1).*
- *After initialization, it runs $MMB_{alg}(V_5)$ and obtain $MMB(V_5)$=$\{V_3, V_4, V_7, V_8, V_{10}, V_{12}\}$ (Lines $3 \sim 4$).*
- *It then learns $\mathcal{L}_{MMB^+(V_5)}$ over $MMB^+(V_5)$, as depicted in Figure 3(b) (Line 10).*
- *Next, it updates $\mathcal{P}$ by selecting the edges connected to $V_5$ and the V-structures containing $V_5$ (Line 12).*
- *It now orients $V_5 \circ\!\!\rightarrow V_{10}$ as $V_5 \rightarrow V_{10}$ by orientation rules (Line 13). Consequently, it obtains $\mathcal{P}$ as shown in Figure 3(c).*
- *Then, it updates* **Donelist** $= \{V_5\}$, *and* **Waitlist** $= \{V_4, V_8, V_{10}\}$ *(Lines $14 \sim 15$).*
- *Sequentially, it runs $MMB_{alg}(V_4)$ and obtain $MMB(V_4)$=$\{V_2, V_3, V_5, V_7, V_8, V_{11}, V_{12}\}$.*
- *It then learns the local structure $\mathcal{L}_{MMB^+(V_4)}$ as shown in Figure 3(d) (Line 10).*
- *Next, it pools the determined edges together and orients $V_4 \circ\!\!\rightarrow V_5$ as $V_4 \rightarrow V_5$ (Lines $12 \sim 13$). Following this, it derives the local structure $\mathcal{P}$, as illustrated in Figure 3(e).*
- *Next, it updates sets* **Waitlist** $= \{V_8, V_{10}, V_2, V_3\}$, **Donelist**$=\{V_5, V_4\}$ *(Lines $14 \sim 15$).*
- *Finally, the algorithm terminates because stop $\mathcal{R}1$ is satisfied. Output the resulting local structure $\mathcal{P}$, which is depicted in Figure 3(f).*

*More details of the example are given in the Appendix D.*

**Theorem 4.** ***The Correctness of MMB-by-MMB Algorithm.*** *We assume Oracle tests for conditional independence tests and accurately obtain the MMB of the target variable $T$ through the MMB discovery algorithm, MMB-by-MMB will identify the direct causes and effects of the target under causal faithfulness and no selection bias assumptions.*

Theorem 4 shows that the local $\mathcal{P}$ obtained through the MMB-by-MMB algorithm is correct, meaning that the edges connected to the target node $T$ and their directions in the output $\mathcal{P}$ are identical to those in the Markov equivalence class of the underlying causal MAG $\mathcal{M}$.

### 4.3. Complexity of MMB-by-MMB Algorithm

This algorithm's complexity can be divided into two parts: the first part involves finding the MAG Markov blanket, and the second part involves learning the local structure. Let $r$ denote the number of local structures to be learned sequentially in our algorithm. In our experiment, we used the TC algorithm(Pellet & Elisseeff, 2008) to search for MMB. The time complexity of finding MMB among $r$ nodes out of $n$ total nodes is $\mathcal{O}\left(\frac{r(2n-r-1)}{2}\right)$, where $n$ denotes the size of observed node set $\mathbf{O}$. When learning local structure, we apply the logic of the PC algorithm to identify the adjacent edges in $MMB^{+}(T)$. In the worst case, the complexity of the PC algorithm for learning a local structure over $n$ nodes is $\mathcal{O}\left(2\binom{n}{2}\sum_{i=0}^{k}\binom{n-1}{i}\right)$. Let $k$ be the maximal degree of any variable and let $|MMB^{+}|$ denote the size of $MMB^{+}(V_X)$. The complexity of the MMB-by-MMB algorithm is the total complexity of finding MAG Markov blankets plus constructing local structures. In the worst case, this is, $\mathcal{O}\left[\frac{r(2n-r-1)}{2}+2r\binom{|MMB^{+}|}{2}\sum_{i=0}^{k}\binom{|MMB^{+}|-1}{i}\right]$.

## 5. Experimental Result

To demonstrate the accuracy and efficiency of our algorithm, we compared the proposed MMB-by-MMB algorithm with global learning methods, such as PC-stable (Colombo & Maathuis, 2014), FCI (Spirtes et al., 2000), and RFCI (Colombo et al., 2012) [4]. We also compared with local learning methods, such as the MB-by-MB algorithm (Wang et al., 2014), the Causal Markov Blanket (CMB) algorithm (Gao & Ji, 2015), and the GradieNt-based LCS (GraN-LCS) algorithm (Liang et al., 2023) [5].

We here use the existing implementation (Pellet & Elisseeff, 2008) of the Total Conditioning (TC) discovery algorithm to find the MB of a target variable. Our source code is available from `https://github.com/fengxie009/MMB-by-MMB`.

### 5.1. Synthetic Data Generated from Benchmark Network Structures

***Experimental setup:*** We select four networks ranging from low to high dimensionality: MILDEW, ALARM, WIN95PTS, and ANDES, containing 35, 37, 76, and 223 nodes, respectively[6]. The network structures are parameterized as a linear Gaussian structural causal model. The causal strength of each edge is drawn from Uniform $([-1,-0.5]\cup[0.5,1])$. For each graph, we randomly select 4, 4, 6,10 nodes as latent variables, and others as observed variables. We here choose nodes with more adjacent nodes as target nodes. Each experiment was repeated 100 times with randomly generated data, and the reported results were averaged. The best results are highlighted in boldface.

***Metrics:*** We evaluate the performance of the algorithms using the following typical metrics: **Precision**: the ratio of true edges [7] in the output to the total number of edges in the algorithm's output. **Recall**: the ratio of true edges in the output to the total number of edges in the ground-truth structure of a target. **F1**: the harmonic average of *Precision* and *Recall*, calculated as

$$F1 = 2 * Precision * Recall/(Prescision + Recall).$$

**Distance**: the Euclidean distance between Recall and Precision, computed as

$$Distance = \sqrt{(1-Recall)^2+(1-Precision)^2}.$$

**nTest**: the number of conditional independence tests implemented by an algorithm.

***Results:*** Due to space constraints, we here present only partial results for each network with two targets. These results are shown in Tables 1 $\sim$ 4. The complete results are given in the Appendix E. From the tables, we can see that our proposed MMB-by-MMB algorithm outperforms other methods with almost all evaluation metrics in all four structures and in all sample sizes, indicating the effectiveness of our method. As expected, the number of conditional independence tests in our method is far less than that in the methods FCI and RFCI, which are used for global learning structures involving latent variables. It is worth noting that although the *nTest* of CMB method is fewer than our method in the WIN95PTS network when $Size = 5000$, the other four metrics of our method outperform CMB. Furthermore, the results of local learning methods, i.e., the MB-by-MB, CMB, and GraN-LCS, are not satisfactory, indicating that these methods cannot address situations involving latent variable structures.

### 5.2. Gene Expression Data

In this section, we apply our method on the gene expression data from Wille et al. (2004), which comprises gene expression measurements of Arabidopsis thaliana grown under 118 different conditions, such as variations in light and darkness, and exposure to growth hormones. Wille et al. (2004) focused particularly on the genes involved in isoprenoid

[4]For PC-stable algorithms, we used the implementations in the MATLAB package at `https://github.com/kuiy/CausalLearner`. FCI algorithm is from Python-package causallearn (Zheng et al., 2023). RFCI algorithm is from R-package pcalg (Kalisch et al., 2012).

[5]We utilized the Python pyCausalFS package (Yu et al., 2020) for MB-by-MB and CMB algorithms. The source code is available at `https://github.com/kuiy/pyCausalFS`, and the GraN-LCS algorithm from `https://www.sdu-idea.cn/codes.php?name=GraN-LCS`.

[6]The details of those networks can be found at `https://www.bnlearn.com/bnrepository/`.

[7]A true edge implies the correct estimation of tails on both sides.

*Table 1.* Performance Comparisons on MILDEW.Net

| | | Size=1000 | | | | | Size=5000 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Target | Algorithm | **Precision↑** | **Recall↑** | **F1↑** | **Distance↓** | **nTest↓** | **Precision↑** | **Recall↑** | **F1↑** | **Distance↓** | **nTest↓** |
| dm-1 | PC-stable | 0.28±0.08 | 0.28±0.09 | 0.28±0.08 | 1.02±0.12 | 5032.70 | 0.27±0.07 | 0.27±0.07 | 0.27±0.07 | 1.04±0.10 | 8930.25 |
| | FCI | 0.83±0.27 | 0.83±0.27 | 0.83±0.27 | 0.24±0.39 | 10260.86 | 0.85±0.27 | 0.85±0.26 | 0.85±0.27 | 0.21±0.38 | 16637.09 |
| | RFCI | 0.81±0.28 | 0.81±0.28 | 0.81±0.28 | 0.27±0.39 | 5032.70 | 0.84±0.26 | 0.84±0.27 | 0.84±0.27 | 0.22±0.38 | 8930.25 |
| | MB-by-MB | 0.51±0.13 | 0.57±0.18 | 0.52±0.14 | 0.67±0.20 | 4596.90 | 0.50±0.12 | 0.58±0.18 | 0.51±0.13 | 0.68±0.19 | 15864.89 |
| | CMB | 0.49±0.15 | 0.52±0.17 | 0.50±0.15 | 0.71±0.22 | 3440.64 | 0.50±0.13 | 0.54±0.16 | 0.51±0.14 | 0.69±0.20 | 3661.29 |
| | GraN-LCS | 0.70±0.22 | 0.78±0.20 | 0.73±0.22 | 0.39±0.30 | - | 0.67±0.21 | 0.74±0.20 | 0.69±0.20 | 0.43±0.29 | - |
| | MMB-by-MMB | **0.95±0.15** | **0.95±0.15** | **0.95±0.15** | **0.07±0.22** | **392.95** | **0.97±0.13** | **0.97±0.13** | **0.97±0.13** | **0.04±0.18** | **613.61** |
| dm-4 | PC-stable | 0.33±0.18 | 0.33±0.18 | 0.33±0.18 | 0.94±0.26 | 5071.66 | 0.33±0.18 | 0.33±0.18 | 0.33±0.18 | 0.95±0.25 | 8972.31 |
| | FCI | 0.77±0.20 | 0.70±0.20 | 0.72±0.20 | 0.39±0.29 | 10338.28 | 0.84±0.15 | 0.77±0.20 | 0.80±0.18 | 0.29±0.26 | 16762.44 |
| | RFCI | 0.72±0.21 | 0.64±0.19 | 0.67±0.20 | 0.47±0.28 | 5071.66 | 0.81±0.15 | 0.73±0.20 | 0.76±0.18 | 0.35±0.26 | 8972.31 |
| | MB-by-MB | 0.59±0.14 | 0.69±0.18 | 0.62±0.14 | 0.53±0.20 | 7075.49 | 0.60±0.15 | 0.71±0.18 | 0.63±0.16 | 0.51±0.22 | 28815.65 |
| | CMB | 0.60±0.15 | 0.63±0.16 | 0.61±0.15 | 0.55±0.22 | 2325.96 | 0.58±0.15 | 0.63±0.15 | 0.60±0.15 | 0.57±0.21 | 3638.17 |
| | GraN-LCS | 0.57±0.17 | 0.59±0.19 | 0.56±0.16 | 0.62±0.23 | - | 0.60±0.20 | 0.61±0.21 | 0.59±0.19 | 0.58±0.26 | - |
| | MMB-by-MMB | **0.95±0.13** | **0.91±0.16** | **0.92±0.15** | **0.11±0.21** | **527.14** | **0.99±0.05** | **0.98±0.09** | **0.98±0.08** | **0.03±0.11** | **690.57** |

Note: The symbol '-' indicates that GraN-LCS does not output this information. MMB-by-MMB with the second best result is underlined. ↑ means a higher value is better, and vice versa.

*Table 2.* Performance Comparisons on ALARM.Net

| | | Size=1000 | | | | | Size=5000 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Target | Algorithm | **Precision↑** | **Recall↑** | **F1↑** | **Distance↓** | **nTest↓** | **Precision↑** | **Recall↑** | **F1↑** | **Distance↓** | **nTest↓** |
| LVEDVOLUME | PC-stable | 0.56±0.16 | 0.55±0.14 | 0.56±0.15 | 0.63±0.20 | 3515.85 | 0.56±0.16 | 0.55±0.14 | 0.56±0.15 | 0.63±0.20 | 4878.73 |
| | FCI | 0.78±0.25 | 0.77±0.25 | 0.77±0.25 | 0.32±0.35 | 7552.66 | 0.84±0.24 | 0.83±0.24 | 0.83±0.24 | 0.24±0.34 | 10173.30 |
| | RFCI | 0.78±0.25 | 0.77±0.24 | 0.77±0.24 | 0.32±0.34 | 3515.85 | 0.83±0.23 | 0.83±0.24 | 0.83±0.24 | 0.24±0.33 | 4878.73 |
| | MB-by-MB | 0.46±0.18 | 0.46±0.18 | 0.45±0.17 | 0.77±0.25 | 1531.45 | 0.44±0.15 | 0.44±0.16 | 0.43±0.14 | 0.81±0.20 | 4196.87 |
| | CMB | 0.44±0.21 | 0.43±0.19 | 0.43±0.20 | 0.80±0.28 | 1471.71 | 0.44±0.20 | 0.43±0.18 | 0.43±0.18 | 0.81±0.26 | 1992.32 |
| | GraN-LCS | 0.58±0.15 | 0.57±0.14 | 0.57±0.14 | 0.61±0.20 | - | 0.58±0.13 | 0.58±0.13 | 0.58±0.13 | 0.60±0.18 | - |
| | MMB-by-MMB | **0.97±0.12** | **0.96±0.12** | **0.96±0.12** | **0.05±0.16** | **324.09** | **0.98±0.09** | **0.98±0.09** | **0.98±0.09** | **0.03±0.12** | **344.51** |
| STROKEVOLUME | PC-stable | 0.57±0.17 | 0.56±0.13 | 0.56±0.14 | 0.62±0.20 | 3470.71 | 0.57±0.17 | 0.56±0.13 | 0.56±0.14 | 0.62±0.20 | 4840.23 |
| | FCI | 0.78±0.25 | 0.77±0.25 | 0.77±0.25 | 0.33±0.35 | 7483.00 | 0.84±0.23 | 0.84±0.23 | 0.84±0.23 | 0.23±0.33 | 10171.28 |
| | RFCI | 0.72±0.25 | 0.71±0.24 | 0.71±0.25 | 0.41±0.35 | 3470.71 | 0.83±0.24 | 0.83±0.24 | 0.83±0.24 | 0.24±0.34 | 4840.23 |
| | MB-by-MB | 0.43±0.18 | 0.48±0.21 | 0.43±0.18 | 0.79±0.25 | 2076.76 | 0.37±0.15 | 0.41±0.17 | 0.38±0.15 | 0.88±0.21 | 6377.99 |
| | CMB | 0.44±0.20 | 0.45±0.20 | 0.44±0.19 | 0.79±0.27 | 2197.08 | 0.37±0.19 | 0.37±0.19 | 0.37±0.19 | 0.89±0.26 | 2597.72 |
| | GraN-LCS | 0.42±0.13 | 0.46±0.16 | 0.43±0.14 | 0.80±0.20 | - | 0.42±0.13 | 0.46±0.16 | 0.43±0.14 | 0.80±0.20 | - |
| | MMB-by-MMB | **0.95±0.15** | **0.95±0.16** | **0.95±0.16** | **0.08±0.22** | **566.39** | **0.98±0.09** | **0.98±0.09** | **0.98±0.09** | **0.02±0.12** | **698.17** |

Note: The symbol '-' indicates that GraN-LCS does not output this information. MMB-by-MMB with the second best result is underlined. ↑ means a higher value is better, and vice versa.

*Table 3.* Performance Comparisons on WIN95PTS.Net

| | | Size=1000 | | | | | Size=5000 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Target | Algorithm | **Precision↑** | **Recall↑** | **F1↑** | **Distance↓** | **nTest↓** | **Precision↑** | **Recall↑** | **F1↑** | **Distance↓** | **nTest↓** |
| Problem5 | Pc-stable | 0.52±0.07 | 0.52±0.07 | 0.52±0.07 | 0.68±0.10 | 12657.62 | 0.53±0.07 | 0.53±0.08 | 0.53±0.07 | 0.67±0.10 | 26398.40 |
| | FCI | 0.69±0.24 | 0.69±0.24 | 0.68±0.24 | 0.45±0.34 | 25417.23 | 0.77±0.27 | 0.76±0.27 | 0.76±0.27 | 0.34±0.38 | 43850.55 |
| | RFCI | 0.67±0.23 | 0.66±0.23 | 0.66±0.23 | 0.48±0.33 | 12657.62 | 0.77±0.27 | 0.75±0.28 | 0.76±0.28 | 0.35±0.39 | 26398.40 |
| | MB-by-MB | 0.46±0.19 | 0.50±0.22 | 0.47±0.20 | 0.75±0.28 | 13633.52 | NA | NA | NA | NA | NA |
| | CMB | 0.57±0.16 | 0.60±0.18 | 0.58±0.17 | 0.59±0.23 | 4757.95 | 0.54±0.15 | 0.58±0.17 | 0.56±0.16 | 0.63±0.22 | **5413.78** |
| | GraN-LCS | 0.48±0.14 | 0.50±0.15 | 0.48±0.14 | 0.73±0.20 | - | 0.48±0.16 | 0.48±0.17 | 0.48±0.16 | 0.74±0.22 | - |
| | MMB-by-MMB | **0.90±0.20** | **0.90±0.20** | **0.89±0.20** | **0.15±0.29** | **3907.42** | **0.93±0.18** | **0.92±0.19** | **0.92±0.19** | **0.11±0.27** | 12372.20 |
| HrglssDrtnAftrPrnt | PC-stable | 0.78±0.08 | 0.77±0.06 | 0.78±0.07 | 0.32±0.10 | 12637.44 | 0.77±0.06 | 0.76±0.04 | 0.76±0.05 | 0.34±0.07 | 25058.00 |
| | FCI | 0.80±0.24 | 0.80±0.24 | 0.80±0.24 | 0.28±0.34 | 25651.52 | 0.83±0.24 | 0.83±0.25 | 0.83±0.25 | 0.24±0.35 | 42604.71 |
| | RFCI | 0.81±0.24 | 0.81±0.24 | 0.81±0.24 | 0.27±0.34 | 12637.44 | 0.82±0.25 | 0.82±0.25 | 0.82±0.25 | 0.25±0.35 | 25058.00 |
| | MB-by-MB | 0.46±0.09 | 0.46±0.09 | 0.46±0.09 | 0.77±0.13 | 14169.84 | 0.45±0.11 | 0.45±0.11 | 0.45±0.11 | 0.78±0.15 | 45118.80 |
| | CMB | 0.48±0.23 | 0.48±0.23 | 0.48±0.23 | 0.73±0.33 | 7933.74 | 0.42±0.19 | 0.42±0.19 | 0.42±0.19 | 0.82±0.26 | 11783.44 |
| | GraN-LCS | 0.39±0.14 | 0.39±0.14 | 0.39±0.14 | 0.86±0.20 | - | 0.43±0.13 | 0.43±0.13 | 0.43±0.13 | 0.80±0.18 | - |
| | MMB-by-MMB | **0.92±0.18** | **0.92±0.18** | **0.92±0.18** | **0.11±0.25** | **1054.52** | **0.92±0.20** | **0.92±0.20** | **0.92±0.20** | **0.11±0.29** | **2029.79** |

Note: The symbol '-' indicates that GraN-LCS does not output this information. MMB-by-MMB with the second best result is underlined. ↑ means a higher value is better, and vice versa. NA entries for MB-by-MB demonstrate that the runtime exceeds a certain threshold.

*Table 4.* Performance Comparisons on ANDES.Net

| | | Size=1000 | | | | | Size=5000 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Target | Algorithm | **Precision↑** | **Recall↑** | **F1↑** | **Distance↓** | **nTest↓** | **Precision↑** | **Recall↑** | **F1↑** | **Distance↓** | **nTest↓** |
| RApp3($V_5$) | PC-stable | 0.23±0.07 | 0.23±0.07 | 0.23±0.07 | 1.09±0.10 | 234677.37 | 0.23±0.07 | 0.23±0.07 | 0.23±0.07 | 1.09±0.10 | 439483.08 |
| | FCI | 0.70±0.24 | 0.68±0.25 | 0.68±0.24 | 0.45±0.34 | 901063.34 | 0.79±0.24 | 0.78±0.25 | 0.79±0.24 | 0.30±0.34 | 1584682.77 |
| | RFCI | 0.66±0.24 | 0.64±0.24 | 0.65±0.24 | 0.50±0.34 | 234677.37 | 0.78±0.24 | 0.77±0.25 | 0.77±0.25 | 0.32±0.35 | 439483.08 |
| | MB-by-MB | 0.34±0.07 | 0.43±0.12 | 0.36±0.08 | 0.89±0.12 | 24239.83 | 0.39±0.08 | 0.56±0.12 | 0.43±0.09 | 0.78±0.12 | 44225.00 |
| | CMB | 0.33±0.06 | 0.38±0.09 | 0.34±0.07 | 0.92±0.09 | 79932.47 | 0.32±0.05 | 0.39±0.09 | 0.34±0.07 | 0.93±0.09 | 145631.64 |
| | GraN-LCS | 0.39±0.12 | 0.46±0.16 | 0.40±0.12 | 0.84±0.17 | - | 0.38±0.13 | 0.42±0.16 | 0.39±0.13 | 0.86±0.19 | - |
| | MMB-by-MMB | **0.91±0.15** | **0.90±0.17** | **0.89±0.17** | **0.16±0.24** | **5043.44** | **0.98±0.07** | **0.98±0.07** | **0.98±0.07** | **0.03±0.10** | **4595.15** |
| RApp4 | PC-stable | 0.46±0.09 | 0.46±0.09 | 0.46±0.09 | 0.77±0.13 | 234677.37 | 0.46±0.09 | 0.46±0.09 | 0.46±0.09 | 0.77±0.13 | 439483.08 |
| | FCI | 0.79±0.24 | 0.78±0.24 | 0.78±0.24 | 0.32±0.34 | 901063.34 | 0.84±0.23 | 0.84±0.23 | 0.84±0.24 | 0.23±0.33 | 1584682.77 |
| | RFCI | 0.79±0.24 | 0.79±0.24 | 0.79±0.24 | 0.32±0.34 | 234677.37 | 0.84±0.23 | 0.83±0.23 | 0.83±0.24 | 0.24±0.33 | 439483.08 |
| | MB-by-MB | 0.26±0.14 | 0.29±0.16 | 0.26±0.14 | 1.04±0.20 | 18761.00 | NA | NA | NA | NA | NA |
| | CMB | 0.25±0.16 | 0.26±0.17 | 0.25±0.16 | 1.06±0.23 | 87956.70 | 0.23±0.12 | 0.23±0.12 | 0.23±0.12 | 1.09±0.17 | 212387.71 |
| | GraN-LCS | 0.36±0.16 | 0.45±0.19 | 0.38±0.16 | 0.87±0.23 | - | 0.38±0.17 | 0.48±0.22 | 0.40±0.18 | 0.84±0.26 | - |
| | MMB-by-MMB | **0.91±0.22** | **0.93±0.19** | **0.91±0.21** | **0.12±0.30** | **3153.11** | **0.97±0.12** | **0.98±0.09** | **0.97±0.11** | **0.04±0.15** | **1430.12** |

Note: The symbol '-' indicates that GraN-LCS does not output this information. MMB-by-MMB with the second best result is underlined. ↑ means a higher value is better, and vice versa. NA entries for MB-by-MB demonstrate that the runtime exceeds a certain threshold.

synthesis. In Arabidopsis thaliana, isoprenoid synthesis is carried out by two distinct pathways in separate organs: the mevalonate pathway (MVA) and the nonmevalonate pathway (MEP). The dataset we used contains 33 genes. We here employ the model in Wille et al. (2004) as a baseline (See Figure 3 of Wille et al. (2004)). It should be noted that in this model, some edges are undirected. We selected two genes, *DXR* and *MECPS*, as target nodes, respectively. The findings are as follows:

**Target=DXR.** Our method obtains that $Pa(DXR) = \{HMGS\}$ and $Ch(DXR)$={*DXPS2, CMK, MECPS, HDS*}. We found that the connections among four genes *DXPS2*, *CMK*, *MECPS*, and *HDS*, as well as the information that *DXR* is the ancestral node of *CMK*, *MECPS*, and *HDS*, are consistent with the conclusions in Wille et al. (2004). In the baseline, the nodes with edge connections to *DXR* are: {*DXPS1, DXPS2, DXPS3, MCT, CMK, MECPS, HDS, UPPS1*}. The nodes connected by directed edges pointing to *DXR* are {*DXPS1, DXPS2, DXPS3*}, and the node *MCT* is connected by directed edges pointing from *DXR*. Undirected edges connect other nodes.

**Target=PPDS1.** Our method gets $Pa(PPDS1) = \{HDR\}$, and $Ch(PPDS1)$={*PPDS2, DPPS2*}. We found that the connections among three genes *HDR*, *PPDS2*, and *DPPS2* are consistent with the conclusions in Wille et al. (2004). In the baseline, the nodes with edge connections to *PPDS1* are: {*HDR, IPPI1, PPDS2, DPPS2*}. And the node *IPPI1* is connected by directed edges pointing from *PPDS1*. Other nodes are connected by undirected edges.

**Target=MECPS.** Our method gets $Pa(MECPS)$={*DXR, FPPS2*}, and $Ch(MECPS)$={*MCT*}. We found that the connections among two genes *DXR* and *MCT*, as well as the information that *DXR* is the ancestral node of *MECPS*, are consistent with the conclusions in Wille et al. (2004). In the baseline, the nodes with edge connections to *MECPS* are: {*DXR, MCT, CMK, HDS, ACCT1, HMGR2*}. The node *HDS* is connected by directed edges pointing to *MECPS*, and the node *CMK* is connected by directed edges pointing from *MECPS*. Undirected edges connect other nodes.

## 6. Conclusion and and Further Work

We introduce a novel local causal discovery algorithm, MMB-by-MMB, designed to be effective in models with the presence of latent variables. Unlike existing global algorithms, MMB-by-MMB method demonstrates the capability to identify causal structures under equivalent identification conditions, yet it accomplishes this with significantly lower computational expense. Furthermore, we provide proof validating the correctness of the MMB-by-MMB algorithm.

It should be noted that due to the presence of latent variables, the results of the proposed method still include some instances where it is challenging to determine the causes and effects from purely observational data without any further assumptions. Therefore, exploring how to utilize background knowledge, such as leveraging data generation mechanisms (Kaltenpoth & Vreeken, 2023) or expert knowledge (Wang et al., 2023), to further aid in identifying causes and effects within local structures remains a future research direction. Additionally, leveraging theories on combining interventional and observational data (Hauser & Bühlmann, 2015) to learn the local causal structure in the presence of latent variables is interesting for future work.

## Acknowledgements

This research was supported by the Natural Science Foundation of China (62306019) and the Disciplinary funding of Beijing Technology and Business University (STKY202302). Peng Wu is supported by the National Natural Science Foundation of China (No. 12301370). We appreciate the comments from anonymous reviewers, which greatly helped to improve the paper.

## Impact Statement

Observational causal discovery is crucial for analyzing systems where experimental manipulation is ethically impermissible. Our research concentrates on learning local causal structures in the presence of latent variables. A significant strength of our work is its practical approach to learning the parents and children of a target variable of interest from finite samples. Our work can be particularly valuable in the realm of complex systems, such as gene and social networks, to improve the interpretability of machine learning systems when deployed in real-world settings.

## References

Akbari, S., Mokhtarian, E., Ghassami, A., and Kiyavash, N. Recursive causal structure learning in the presence of latent variables and selection bias. *Advances in Neural Information Processing Systems*, 34:10119–10130, 2021.

Aliferis, C. F., Tsamardinos, I., and Statnikov, A. Hiton: a novel markov blanket algorithm for optimal variable selection. In *AMIA annual symposium proceedings*, volume 2003, pp. 21. American Medical Informatics Association, 2003.

Cai, R., Huang, Z., Chen, W., Hao, Z., and Zhang, K. Causal discovery with latent confounders based on higher-order cumulants. In *International Conference on Machine Learning*, pp. 3380–3407. PMLR, 2023.

Chen, L., Li, C., Shen, X., and Pan, W. Discovery and inference of a causal network with hidden confounding.

*Journal of the American Statistical Association*, pp. 1–13, 2023.

Chen, R., Dash, S., and Gao, T. Integer programming for causal structure learning in the presence of latent variables. In *International Conference on Machine Learning*, pp. 1550–1560. PMLR, 2021.

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

Claassen, T. and Heskes, T. A logical characterization of constraint-based causal discovery. In *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, pp. 135–144, 2011.

Claassen, T., Mooij, J. M., and Heskes, T. Learning sparse causal models is not np-hard. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 172. Citeseer, 2013.

Colombo, D. and Maathuis, M. H. Order-independent constraint-based causal structure learning. *J. Mach. Learn. Res.*, 15(1):3741–3782, 2014.

Colombo, D., Maathuis, M. H., Kalisch, M., and Richardson, T. S. Learning high-dimensional directed acyclic graphs with latent and selection variables. *The Annals of Statistics*, pp. 294–321, 2012.

Conati, C., Gertner, A. S., VanLehn, K., and Druzdzel, M. J. On-line student modeling for coached problem solving using bayesian networks. In *User Modeling: Proceedings of the Sixth International Conference UM97 Chia Laguna, Sardinia, Italy June 2–5 1997*, pp. 231–242. Springer, 1997.

Cooper, G. F. A simple constraint-based algorithm for efficiently mining observational databases for causal relationships. *Data Mining and Knowledge Discovery*, 1: 203–224, 1997.

Dawid, A. P. Conditional independence in statistical theory. *Journal of the Royal Statistical Society: Series B (Methodological)*, 41(1):1–15, 1979.

Gao, T. and Ji, Q. Local causal discovery of direct causes and effects. *Advances in Neural Information Processing Systems*, 28, 2015.

Gao, T. and Ji, Q. Efficient markov blanket discovery and its application. *IEEE transactions on Cybernetics*, 47(5): 1169–1179, 2016.

Glymour, C., Zhang, K., and Spirtes, P. Review of causal discovery methods based on graphical models. *Frontiers in genetics*, 10:524, 2019.

Hauser, A. and Bühlmann, P. Jointly interventional and observational data: estimation of interventional markov equivalence classes of directed acyclic graphs. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 77(1):291–318, 2015.

Heinze-Deml, C., Maathuis, M. H., and Meinshausen, N. Causal structure learning. *Annual Review of Statistics and Its Application*, 5:371–391, 2018.

Hernán, M. A. and Robins, J. M. Causal inference, 2010.

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

Isozaki, T. A robust causal discovery algorithm against faithfulness violation. *Information and Media Technologies*, 9(1):121–131, 2014.

Jonas, P., Dominik, J., and Schölkopf, B. *Elements of Causal Inference*. MIT Press, 2017.

Kalisch, M., Mächler, M., Colombo, D., Maathuis, M. H., and Bühlmann, P. Causal inference using graphical models with the r package pcalg. *Journal of statistical software*, 47:1–26, 2012.

Kaltenpoth, D. and Vreeken, J. Nonlinear causal discovery with latent confounders. In *International Conference on Machine Learning*, pp. 15639–15654. PMLR, 2023.

Kitson, N. K., Constantinou, A. C., Guo, Z., Liu, Y., and Chobtham, K. A survey of bayesian network structure learning. *Artificial Intelligence Review*, pp. 1–94, 2023.

Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.

Liang, J., Wang, J., Yu, G., Domeniconi, C., Zhang, X., and Guo, M. Gradient-based local causal structure learning. *IEEE Transactions on Cybernetics*, 2023.

Ma, S., Wang, J., Bieganek, C., Tourani, R., and Aliferis, C. Local causal pathway discovery for single-cell rna sequencing count data: a benchmark study. *Journal of Translational Genetics and Genomics*, 7(1):50–65, 2023.

Maeda, T. N. and Shimizu, S. Rcd: Repetitive causal discovery of linear non-gaussian acyclic models with latent confounders. In *International Conference on Artificial Intelligence and Statistics*, pp. 735–745. PMLR, 2020.

Mani, S. and Cooper, G. F. Causal discovery using a bayesian local causal discovery algorithm. In *MEDINFO 2004*, pp. 731–735. IOS Press, 2004.

Margaritis, D. and Thrun, S. Bayesian network induction via local neighborhoods. *Advances in neural information processing systems*, 12, 1999.

Mokhtarian, E., Khorasani, M., Etesami, J., and Kiyavash, N. Novel ordering-based approaches for causal structure learning in the presence of unobserved variables. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 12260–12268, 2023.

Ogarrio, J. M., Spirtes, P., and Ramsey, J. A hybrid causal search algorithm for latent variable models. In *Conference on probabilistic graphical models*, pp. 368–379. PMLR, 2016.

Pearl, J. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, New York, 2nd edition, 2009.

Pearl, J. Theoretical impediments to machine learning with seven sparks from the causal revolution. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pp. 3–3, 2018.

Pellet, J.-P. and Elisseeff, A. Using markov blankets for causal structure learning. *Journal of Machine Learning Research*, 9(7), 2008.

Pena, J. M., Nilsson, R., Björkegren, J., and Tegnér, J. Towards scalable and data efficient learning of markov boundaries. *International Journal of Approximate Reasoning*, 45(2):211–232, 2007.

Peter, I. S. and Davidson, E. H. A gene regulatory network controlling the embryonic specification of endoderm. *Nature*, 474(7353):635–639, 2011.

Raghu, V. K., Ramsey, J. D., Morris, A., Manatakis, D. V., Sprites, P., Chrysanthis, P. K., Glymour, C., and Benos, P. V. Comparison of strategies for scalable causal discovery of latent variable models from mixed data. *International journal of data science and analytics*, 6:33–45, 2018.

Richardson, T. and Spirtes, P. Ancestral graph markov models. *The Annals of Statistics*, 30(4):962–1030, 2002.

Salehkaleybar, S., Ghassami, A., Kiyavash, N., and Zhang, K. Learning linear non-gaussian causal models in the presence of latent variables. *The Journal of Machine Learning Research*, 21(1):1436–1459, 2020.

Sanchez-Romero, R., Ramsey, J. D., Zhang, K., Glymour, M. R., Huang, B., and Glymour, C. Estimating feedforward and feedback effective connections from fmri time series: Assessments of statistical methods. *Network Neuroscience*, 3(2):274–306, 2019.

Schölkopf, B. Causality for machine learning. In *Probabilistic and Causal Inference: The Works of Judea Pearl*, pp. 765–804. 2022.

Silverstein, C., Brin, S., Motwani, R., and Ullman, J. Scalable techniques for mining causal structures. *Data Mining and Knowledge Discovery*, 4:163–192, 2000.

Smith, S. M., Miller, K. L., Salimi-Khorshidi, G., Webster, M., Beckmann, C. F., Nichols, T. E., Ramsey, J. D., and Woolrich, M. W. Network modelling methods for fmri. *Neuroimage*, 54(2):875–891, 2011.

Spirtes, P. and Zhang, K. Causal discovery and inference: concepts and recent methodological advances. *Applied Informatics*, 3(1):1–28, 2016.

Spirtes, P., Glymour, C., and Scheines, R. *Causation, Prediction, and Search*. MIT press, 2000.

Tsamardinos, I. and Aliferis, C. F. Towards principled feature selection: Relevancy, filters and wrappers. In *International Workshop on Artificial Intelligence and Statistics*, pp. 300–307. PMLR, 2003.

Versteeg, P., Mooij, J., and Zhang, C. Local constraint-based causal discovery under selection bias. In *Conference on Causal Learning and Reasoning*, pp. 840–860. PMLR, 2022.

Walters, G. R., Consortium, W. T. C. C., et al. Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls. *Nature*, 447(7145): 661–678, 2007.

Wang, C., Zhou, Y., Zhao, Q., and Geng, Z. Discovering and orienting the edges connected to a target variable in a dag via a sequential local learning approach. *Computational Statistics & Data Analysis*, 77:252–266, 2014.

Wang, T.-Z., Qin, T., and Zhou, Z.-H. Sound and complete causal identification with latent variables given local background knowledge. *Artificial Intelligence*, pp. 103964, 2023.

Wille, A., Zimmermann, P., Vranová, E., Fürholz, A., Laule, O., Bleuler, S., Hennig, L., Prelić, A., von Rohr, P., Thiele, L., et al. Sparse graphical gaussian modeling of the isoprenoid gene network in arabidopsis thaliana. *Genome biology*, 5(11):1–13, 2004.

Wu, X., Jiang, B., Yu, K., Chen, H., et al. Accurate markov boundary discovery for causal feature selection. *IEEE transactions on cybernetics*, 50(12):4983–4996, 2019.

Xie, X. and Geng, Z. A recursive method for structural learning of directed acyclic graphs. *The Journal of Machine Learning Research*, 9:459–483, 2008.

Yang, S., Wang, H., Yu, K., Cao, F., and Wu, X. Towards efficient local causal structure learning. *IEEE Transactions on Big Data*, 8(6):1592–1609, 2021.

Yaramakala, S. and Margaritis, D. Speculative markov blanket discovery for optimal feature selection. In *Fifth IEEE International Conference on Data Mining (ICDM'05)*, pp. 4–pp. IEEE, 2005.

Yin, J., Zhou, Y., Wang, C., He, P., Zheng, C., and Geng, Z. Partial orientation and local structural learning of causal networks for prediction. In *Causation and Prediction Challenge*, pp. 93–105. PMLR, 2008.

Yu, K., Liu, L., Li, J., and Chen, H. Mining markov blankets without causal sufficiency. *IEEE transactions on neural networks and learning systems*, 29(12):6333–6347, 2018.

Yu, K., Guo, X., Liu, L., Li, J., Wang, H., Ling, Z., and Wu, X. Causality-based feature selection: Methods and evaluations. *ACM Computing Surveys (CSUR)*, 53(5): 1–36, 2020.

Zhalama, Zhang, J., and Mayer, W. Weakening faithfulness: some heuristic causal discovery algorithms. *International journal of data science and analytics*, 3:93–104, 2017.

Zhang, J. Causal reasoning with ancestral graphs. *Journal of Machine Learning Research*, 9:1437–1474, 2008a.

Zhang, J. On the completeness of orientation rules for causal discovery in the presence of latent confounders and selection bias. *Artificial Intelligence*, 172(16-17): 1873–1896, 2008b.

Zheng, Y., Huang, B., Chen, W., Ramsey, J., Gong, M., Cai, R., Shimizu, S., Spirtes, P., and Zhang, K. Causal-learn: Causal discovery in python. *arXiv preprint arXiv:2307.16405*, 2023.

Zhou, Y., Wang, C., Yin, J., and Geng, Z. Discover local causal network around a target to a given depth. In *Causality: Objectives and Assessment*, pp. 191–202. PMLR, 2010.

## A. Notations and Definitions

| Symbol | Description |
|---|---|
| $\mathcal{G}$ | A mixed graph |
| $\mathcal{M}$ | A Maximal Ancestral Graph (MAG) |
| $\mathcal{M}_L$ | A local Maximal Ancestral Graph (MAG) |
| $\mathcal{P}$ | A Partial Ancestral Graph (PAG) |
| $\mathbf{V}$ | The set of all variables |
| $\mathbf{O}$ | The set of observed variables |
| $\mathbf{L}$ | The set of latent variables |
| $Pa(T)$,$Ch(T)$ | The set of all parents and children of $T$, respectively |
| $Sp(T)$ | The set of all spouses of $T$ |
| $An(T)$, $De(T)$ | The set of all ancestors and descendants of $T$, respectively |
| $Adj(T)$ | The set of adjacent vertices of $T$ |
| $MB(T)$ | The Markov blanket of a vertice $T$ in a DAG |
| $MMB(T)$ | The Markov blanket of a vertice $T$ in a MAG |
| $MMB^{+}(T)$ | The set of $\{MMB(T) \cup T\}$ |
| $S_{X,Y}$ | The set of m-separates X and Y |
| $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z})_{\mathcal{G}}$ | A set $\mathbf{Z}$ m-separates $\mathbf{X}$ and $\mathbf{Y}$ in $\mathcal{G}$ |
| $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z})_{P}$ | $\mathbf{X}$ is statistically independent of $\mathbf{Y}$ given $\mathbf{Z}$. We drop the subscript $P$ whenever it is clear from context. |
| $(\mathbf{X} \not\perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z})_{P}$ | $\mathbf{X}$ is not statistically independent of $\mathbf{Y}$ given $\mathbf{Z}$ |
| $A \rightarrow B$ in $\mathcal{G}$ | $A$ is a cause of $B$, but $B$ is not a cause of $A$ |
| $A \leftrightarrow B$ in $\mathcal{G}$ | $A$ is not a cause of $B$, and $B$ is not a cause of $A$ |
| $A \quad B$ in $\mathcal{P}$ | $A$ and $B$ are not adjacent |
| $A \circ\!\!\rightarrow B$ in $\mathcal{P}$ | $B$ is not an ancestor of $A$ |
| $A \circ\!\!-\!\!\circ B$ in $\mathcal{P}$ | No set m-separates $A$ and $B$ |
| $A \rightarrow B$ in $\mathcal{P}$ | $A$ is a cause of $B$ |
| $A \leftrightarrow B$ in $\mathcal{P}$ | There is a latent common cause of $A$ and $B$ |
| $\mathcal{L}_{\mathcal{V}}$ | The local structure learned from a subset $\mathcal{V}$ of $\mathbf{V}$, utilizing the test of conditional independence and orientation of V-structures |
| $MMB_{alg}$ | The algorithm used for learning $MMB$ |
| **WaitList** | The list of nodes to be checked by Theorem 1 and Theorem 2 |
| **DoneList** | The list of nodes whose local structures have been found |

*Table 5.* The list of main symbols used in this paper

**Definition 2.** ***MAG Markov Blanket (MMB)*** *(Pellet & Elisseeff, 2008) In a MAG, the Markov blanket of a vertice $T$, noted as $MMB(T)$, consists of the set of parents, children, children's parents of $T$, as well as the district of $T$ and of the children of $T$, and the parents of each node of these districts, where the district of a node $X$ is the set of all nodes reachable from $X$ using only bidirected edges.*

In Figure 4, the MAG Markov Blanket of $T$ is specifically illustrated.

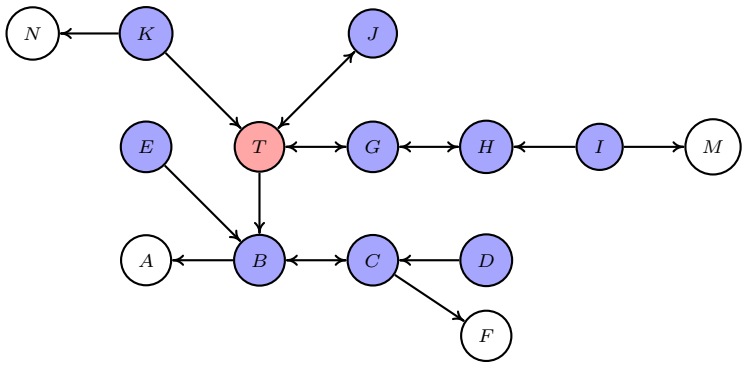



*Figure 4.* The illustrative example for MMB, where T is the target of interest and the blue nodes belong to $MMB(T)$.

## B. Discussion of violating the faithfulness assumption

First, we would like to mention that classical causal discovery, such as PC, FCI, and RFCI, is usually dependent on the causal faithfulness assumption, and these methods have been used in a range of fields (Spirtes & Zhang, 2016).

Second, reducing unnecessary conditional independence tests can mitigate statistically weak violations of the causal faithfulness assumption (Isozaki, 2014), which is precisely the focus of our paper. Moreover, experimental results also validate this point (nTest is minimal, and other metrics are superior to existing methods).

Lastly, incorporating elements of the Greedy Equivalence Search (GES) algorithm (Chickering, 2002), a representative score-based method, could be made more robust against violation of faithfulness (Zhalama et al., 2017), which is the direction of our future work.

## C. Proofs

### C.1. Proof of Theorem 1

Before presenting the proof, we quote the Theorem 1 of Xie & Geng (2008).

**Lemma 1.** *Suppose that $A \perp\!\!\!\perp B|C$. Let $u \in A$ and $v \in A \cup C$. Then $u$ and $v$ are m-separated by a subset of $A \cup B \cup C$ if and only if they are m-separated by a subset of $A \cup C$.*

We begin to utilize Lemma 1 to prove Theorem 1.

*Proof.* From the property of MMB, we know $T \perp\!\!\!\perp \{\mathbf{O} \setminus MMB^+(T)\}|MMB(T)$. Let $X$ be a node in $MMB(T)$. According to Lemma 1, we directly obtain that $T$ and $X$ are m-separated by a subset of $\mathbf{O}$ if and only if they are m-separated by a subset of $MMB(T) \setminus \{X\}$. □

### C.2. Proof of Theorem 2

*Proof.* We prove statements $\mathcal{S}1$ and $\mathcal{S}2$ in Theorem 2 separately below. For notational convenience, let $S_{X,Y}$ denote the set of nodes that m-separates $X$ and $Y$.

**Statement $\mathcal{S}1$.** Without loss of generality, we assume that $V_a{*\!\!\rightarrow} T \leftarrow\!\!*V_b$ is a V-structure in the ground-truth MAG $\mathcal{M}$ over $\mathbf{O}$. Because $V_a$ and $V_b$ are two nodes in $MMB(T)$ and according to Theorem 1, we can ascertain the presence of direct edges $V_a - T$ and $T - V_b$ by the marginal distribution of $MMB^+(T)$. Next, we need to discuss the following two cases:

- $S_{V_a,V_b} \subset MMB(T)$. Because $S_{V_a,V_b} \subset MMB(T)$ in the sub-MAG $\mathcal{M}'$, we can directly verify the condition $V_a \perp\!\!\!\perp V_b|S_{V_a,V_b}$ from the marginal distribution of $MMB^+(T)$. Due to $T \in MMB^+(T)$, we obtain $V_a \not\perp\!\!\!\perp V_b|S_{V_a,V_b} \cup \{T\}$. This will imply that the $V_a{*\!\!\rightarrow} T \leftarrow\!\!*V_b$ identified by the marginal distribution of $MMB^+(T)$ is exactly the V-structure in the ground-truth MAG $\mathcal{M}$.
- $S_{V_a,V_b} \nsubseteq MMB(T)$. Because $S_{V_a,V_b} \nsubseteq MMB(T)$ in the sub-MAG $\mathcal{M}'$, we can directly deduce that it is impossible to find a separating set for $V_a$ and $V_b$ in $MMB^+(T)$, implying that we are unable to identify such V-structures, even if they exist in the ground-truth MAG $\mathcal{M}$.

Based on the above analysis, the V-structures we identify from the marginal distribution of $MMB^+(T)$ must be consistent with those in the ground-truth MAG $\mathcal{M}$.

**Statement $\mathcal{S}2$.** Without loss of generality, we assume that $T{*\!\!\rightarrow} V_a \leftarrow\!\!*V_b$ is a V-structure in the ground-truth MAG $\mathcal{M}$ over $\mathbf{O}$. To identify this V-structure from the marginal distribution of $MMB^+(T)$, we need to verify the following four conditions:

- 1. $\forall S \subseteq MMB(T), T \not\perp\!\!\!\perp V_a \mid S$
- 2. $\exists S_{T,V_b} \subseteq MMB(T), T \perp\!\!\!\perp V_b \mid S_{T,V_b}$
- 3. $V_a \notin S_{T,V_b}$
- 4. $\forall S \subseteq MMB^+(T), V_a \not\perp\!\!\!\perp V_b \mid S$

According to Theorem 1, we can directly conclude that there exists the direct edge $T - V_a$, and the direct edge $T - V_b$ does not exist, by the marginal distribution of $MMB^+(T)$. These results will imply that the above two conditions hold, i.e., $\forall S \subseteq MMB(T), T \not\perp\!\!\!\perp V_a \mid S$ and $\exists S_{T,V_b} \subseteq MMB(T), T \perp\!\!\!\perp V_b \mid S_{T,V_b}$. Because of the property of V-structure, we

obtain the third condition hold, i.e., $V_a \notin S_{T,V_b}$.

We next show that condition 4 can be verified by the marginal distribution of $MMB^+(T)$. Let's consider the scenario of a spurious edge between $V_a$ and $V_b$, meaning there is no direct edge between them, but for any subset $\forall S \subseteq MMB^+(T), V_a \not\perp\!\!\!\perp V_b \mid S$. Our objective is to demonstrate that if it is an active path connecting $V_a$ and $V_b$ instead of a direct edge, then $\exists S \subseteq MMB^+(T), V_a \perp\!\!\!\perp V_b \mid S$. We can examine active paths in the following three cases:

- 1. $V_a \leftarrow \cdots V_b$
- 2. $V_a \leftrightarrow V_x \rightarrow \cdots V_b$
- 3. $V_a \rightarrow \cdots V_b$

Given that $T{*}\!\!\rightarrow V_a$, we have $Pa(V_a), Sp(V_a), Pa(Sp(V_a)) \subset MMB^+(T)$. Each path in case 1 is blocked by $Pa(V_a)$ and, consequently, by a subset of $MMB^+(T)$. For case 2, all paths are obstructed by $Sp(V_a)$, which is also a subset of $MMB^+(T)$. Thus, we only need to demonstrate that there exists a subset of $MMB^+(T)$ that blocks each path $V_a \rightarrow V_x \cdots V_b$. By conditions 2 and 3, $S_{T,V_b}$ blocks each path $V_a \rightarrow V_x \cdots V_b$ since $V_a \notin S_{T,V_b}$. In summary, in the absence of a direct edge between $V_a$ and $V_b$, we can obtain that $\exists S \subseteq MMB^+(T), V_a \perp\!\!\!\perp V_b \mid S$. This will imply that condition 4 can be verified by the marginal distribution of $MMB^+(T)$.

In conclusion, the V-structure $T{*}\!\!\rightarrow V_a \leftarrow\!\!{*}V_b$ we identify from the marginal distribution of $MMB^+(T)$ must be consistent with those in the ground-truth MAG $\mathcal{M}$.

□

### C.3. Proof of Theorem 3

Rule $\mathcal{R}1$ implies that all the causal information of interest, i.e., the edges and directions connected to the target, has been found. Rule $\mathcal{R}2$ asserts that all nodes have been effectively utilized, leaving no node for sequential learning. Both rules $\mathcal{R}1$ and $\mathcal{R}2$ are self-evident. Therefore, our task is to establish the validity of $\mathcal{R}3$. Before that, we quote the following lemma since it is used to prove Theorem 3.

**Lemma 2.** *In a MAG $\mathcal{M}$ with a set of vertices $\mathbf{X}$, consider $Y$ as a leaf node (i.e., $Y$ is not an ancestor of any node in $\mathbf{X}$). Let $\mathcal{M}'$ be the new MAG obtained by removing $Y$ from $\mathcal{M}$, and $\mathbf{X}'$ be the set of all nodes in $\mathcal{M}'$, then the following condition holds:*

$$P_{\mathcal{M}}(\mathbf{X}') = P_{\mathcal{M}'}(\mathbf{X}') \tag{1}$$

Lemma 2 implies that the joint probability distribution of the remaining node set $\mathbf{X}'$ in the new MAG $\mathcal{M}'$ is equivalent to the joint probability distribution of the same node set $\mathbf{X}'$ in the original MAG $\mathcal{M}$. In other words, removing the leaf node $Y$ from $\mathcal{M}$ does not alter the joint probability distribution of the remaining node set $\mathbf{X}'$.

We now prove Lemma 2.

*Proof.*

$$\begin{aligned}
P_{\mathcal{M}}(\mathbf{X}') &= \sum_{Y} P_{\mathcal{M}}(\mathbf{X}', Y) \\
&= \sum_{Y} \prod_{X \in \mathbf{X}'} P(X \mid Pa(X)) P(Y \mid Pa(Y)) \\
&= \prod_{X \in \mathbf{X}'} P(X \mid Pa(X)) \sum_{Y} P(Y \mid Pa(Y)) \\
&\text{(Because } Y \text{ is a leaf node, then } \forall X \in \mathbf{X}', Y \notin Pa\,(X)) \\
&= \prod_{X \in \mathbf{X}'} P(X \mid Pa(X)) \\
&= P_{\mathcal{M}'}(\mathbf{X}')
\end{aligned} \tag{2}$$

□

**Remark 3.** *Lemma 2 inspires us the following facts:*

- *Firstly, let* $\mathbf{O}$ *be a set of observed data,* $\mathcal{M}$ *be the MAG graph over* $\mathbf{O}$*, and* $\mathbf{X}' = \{\mathbf{O} \setminus \mathbf{Y}\}$*.* $\mathbf{Y}$ *represents the set of leaf nodes relative to* $\mathbf{X}'$ *in* $\mathcal{M}$*. According to Lemma 2, we can deduce that* $P_{\mathcal{M}}(\mathbf{X}') = P_{\mathcal{M}'}(\mathbf{X}')$*, where* $\mathcal{M}'$ *is the new MAG obtained by removing* $\mathbf{Y}$ *from* $\mathcal{M}$*.*
- *Secondly, let* $\mathbf{X}'' = \{\mathbf{O} \setminus \mathbf{Y} \cup \mathbf{Y}'\}$*, and* $\mathbf{Y}'$ *denote the set of leaf nodes relative to* $\mathbf{X}''$ *in* $\mathcal{M}'$*. Subsequently, we can infer* $P_{\mathcal{M}'}(\mathbf{X}'') = P_{\mathcal{M}''}(\mathbf{X}'')$ *using Lemma 2, where* $\mathcal{M}''$ *is the new MAG obtained by removing* $\mathbf{Y}'$ *from* $\mathcal{M}'$*.*
- *Finally, by repeating the above steps multiple times, we can obtain a local MAG* $\mathcal{M}_L$*, and* $P_{\mathcal{M}}(\mathbf{X}) = P_{\mathcal{M}'}(\mathbf{X}) = P_{\mathcal{M}''}(\mathbf{X}) = P_{\mathcal{M}_L}(\mathbf{X})$*, where* $\mathbf{X} = \{\mathbf{O} \setminus \mathcal{Y}\}$*, and* $\mathcal{Y}$ *denotes the leaf nodes that are removed during the repetition process.*

Based on the above analysis, we now prove the rule $\mathcal{R}3$ of Theorem 3.

*Proof.* Based on the above description of Remark 3 and relying on the faithfulness assumption, it can be inferred that after iteratively deleting all leaf nodes in the MAG, the relationship among the remaining nodes will remain unchanged.

First, let's provide a more detailed explanation of $\mathcal{R}3$. In our sequential approach, assuming the sequential learning process terminates due to satisfying $\mathcal{R}3$, the result we learn is a local MAG's PAG $\mathcal{P}$. Assuming that we have identified a node set $\mathbf{O}'$ around $T$ in the sequential learning process, and there exists a path between each node in the set $\mathbf{Waitlist}$ and $T$. Suppose $Y \in \mathbf{Waitlist}$, let $V_i$ represent the nodes excluding $T$ and $Y$ in the path, where $i \in [1, n]$ and $V_i \in \mathbf{Donelist}$. Let $V_1$ denote the node closest to $T$ on the path, and $V_n$ denote the node closest to $Y$. If we identify that all paths connected by undirected edges around $T$ possess the following characteristics: the edges between $T$ and $V_1$ on the path are undirected, while the edges between $V_n$ and $Y$ are either $V_n \rightarrow Y$ or $V_n \circ\!\!\rightarrow Y$. Hence, upon satisfaction of $\mathcal{R}3$, we can conclude the sequential learning algorithm.

Then, We proceed to demonstrate why the sequential learning algorithm can be halted when $\mathcal{R}3$ is satisfied. In our learning process, we identify that the edges between $T$ and $V_1$ on the paths are undirected, while $Y$ are not ancestors of $V_n$. These paths from $T$ to $Y$ in underlying $\mathcal{M}_L$ can be considered in the following two cases:

- 1. $T \cdots V_1 \cdots V_n \leftrightarrow Y$
- 2. $T \cdots V_1 \cdots V_n \rightarrow Y$

Since we have identified $*\!\!\rightarrow Y$, we can infer that these $Y$ nodes belong to the leaf nodes of the underlying $\mathcal{M}_L$. Combining these $Y$ nodes into a set $\mathbf{Y}$, according to Lemma 2, we can deduce that $P_{\mathcal{M}'_L}(\mathbf{O}'') = P_{\mathcal{M}_L}(\mathbf{O}'')$, where $\mathbf{O}'' = \{\mathbf{O}' \setminus \mathbf{Y}\}$ is the new MAG obtained by removing $\mathbf{Y}$ from $\mathcal{M}_L$.

This implies that the joint probability distribution of the remaining nodes set $\mathbf{O}''$ in $\mathcal{M}'_L$ is equivalent to the joint probability distribution of the same node set $\mathbf{O}''$ in the $\mathcal{M}_L$. Then, through $P_{\mathcal{M}}(\mathbf{X}) = P_{\mathcal{M}'}(\mathbf{X}) = P_{\mathcal{M}''}(\mathbf{X}) = P_{\mathcal{M}_L}(\mathbf{X})$ in Remark 3, we can get $P_{\mathcal{M}'_L}(\mathbf{O}'') = P_{\mathcal{M}}(\mathbf{O}'')$.

However, we failed to identify, based on the marginal distribution of $\mathbf{O}''$, that all paths involving undirected edges connected to $T$ are blocked by $*\!\!\rightarrow$. Therefore, we continue the learning process until all paths involving undirected edges connected to $T$ are blocked by $*\!\!\rightarrow$ through the marginal distribution of $\mathbf{O}'$. To summarize, when the situation satisfying $\mathcal{R}3$ is identified, we can get $P_{\mathcal{M}_L}(\mathbf{O}') = P_{\mathcal{M}}(\mathbf{O}')$ which implies that continuing this algorithm will not contribute to orienting the undirected edges in $\mathcal{P}$. Hence, upon satisfaction of $\mathcal{R}3$, we can conclude the sequential learning algorithm. □

### C.4. Proof of Theorem 4

*Proof.* To establish the correctness of the MMB-by-MMB approach, it is imperative to demonstrate the correctness of all edges and orientations in the resulting graph $\mathcal{P}$. Additionally, it is crucial to assert that the undirected edges linked to the target node $T$ remain unaltered, defying further orientation even as the algorithm progresses.

Following Theorem 1, it is established that all edges connected to nodes in the Donelist are accurate. Given that $T$ is encompassed within the Donelist, the edges linked to $T$ are deemed correct.

Subsequently, relying on Theorem 2, it can be inferred that all v-structures in $\mathcal{P}$ are correct, and those v-structures having at least one node that does not belong to the ancestors of the collider within the Donelist are correctly identified. Following Zhang's orientation methodology, the undirected edges in $\mathcal{P}$ are oriented by checking the presence of edges in $\mathcal{P}$.

Ultimately, we demonstrate that continuing the algorithm cannot orient the undirected edges connected to $T$ in the output

$\mathcal{P}$ if they are present. As outlined earlier, we accurately ascertain all edges and v-structures along with their orientations. Thus, a PAG $\mathcal{P}$ representing the Markov equivalence class of the underlying MAG is obtained when the Donelist equals the complete set **O** of all nodes. In cases where the Donelist is a subset of **O**, the algorithm stops, as nodes in the Donelist do not establish connections with nodes outside $\mathcal{P}$, or their paths to nodes outside $\mathcal{P}$ are all blocked by $* \rightarrow$. In such instances, the edges and orientations identified through the continuing algorithm do not contribute to orienting the undirected edges in $\mathcal{P}$, as these undirected edges have already been enveloped by previously determined directed edges.

Hence, the correctness of the MMB-by-MMB algorithm is proven. □

## D. Illustration of MMB-by-MMB Algorithm

In this section, we illustrate our MMB-by-MMB with the graph in Figure 3(a). We assume Oracle tests for conditional independence conditions.

- It first initializes sets **Waitlist** = $\{V_5\}$, **Donelist** = $\emptyset$, and graph $\mathcal{P} = \emptyset$ (Line 1).
- After initialization, it runs $MMB_{alg}(V_5)$ and obtain $MMB(V_5)$=$\{V_3, V_4, V_7, V_8, V_{10}, V_{12}\}$ (Lines $3 \sim 4$).
- It then learns $L_{MMB^+(V_5)}$ over $MMB^+(V_5)$: $V_4 \circ\!\!\rightarrow V_5 \circ\!\!\rightarrow V_{10} \leftarrow\!\!\circ V_3 \circ\!\!\rightarrow V_4$ and $V_{12} \circ\!\!\rightarrow V_8 \leftrightarrow V_5 \leftarrow\!\!\circ V_4 \leftarrow\!\!\circ V_7 \circ\!\!\rightarrow V_8$, as depicted in Figure 3(b) (Line 10).
- Next, it updates $\mathcal{P}$ by selecting the edges connected to $V_X$ and the V-structures containing $V_X$ (Line 12). According to Theorem 1 and Theorem 2, these edges can be determined :$V_3 \circ\!\!\rightarrow V_{10} \leftarrow\!\!\circ V_5 \leftarrow\!\!\circ V_4$, $V_5 \leftrightarrow V_8 \leftarrow\!\!\circ V_{12}$ and $V_7 \circ\!\!\rightarrow V_8$.
- It now orients $V_5 \circ\!\!\rightarrow V_{10}$ as $V_5 \rightarrow V_{10}$ by orientation rules (Line 13). Consequently, it obtains $\mathcal{P}$ as shown in Figure 3(c).
- Then, it updates **Donelist**=$\{V_5\}$, and **Waitlist**=$\{V_4, V_8, V_{10}\}$ (Lines $14 \sim 15$).
- Sequentially, it runs $MMB_{alg}(V_4)$ and obtain $MMB(V_4)$=$\{V_2, V_3, V_5, V_7, V_8, V_{11}, V_{12}\}$.
- It then learns the local structure $\mathcal{L}_{MMB^+(V_4)}$ : $V_{11} \circ\!\!\rightarrow V_3 \leftrightarrow V_4 \circ\!\!\rightarrow V_5 \leftrightarrow V_8$ and $V_4 \leftarrow\!\!\circ V_2 \circ\!\!-\!\!\circ V_7 \circ\!\!\rightarrow V_8 \leftarrow\!\!\circ V_{12}$, as shown in Figure 3 (Line 10).
- According to Theorem 1 and Theorem 2, these edges can be determined : $V_{11} \circ\!\!\rightarrow V_3 \leftrightarrow V_4 \circ\!\!\rightarrow V_5 \leftrightarrow V_8 \leftarrow\!\!\circ V_{12}$, $V_2 \circ\!\!\rightarrow V_4$ and $V_{12} \circ\!\!\rightarrow V_8$. Next, it pools the determined edges together and orients $V_4 \circ\!\!\rightarrow V_5$ as $V_4 \rightarrow V_5$ (Lines $12 \sim 13$). Following this, it derives the local structure $\mathcal{P}$, as illustrated in Figure 3(e).
- Next, it updates sets **Waitlist**=$\{V_8, V_{10}, V_2, V_3\}$, **Donelist**=$\{V_5, V_4\}$ (Lines $14 \sim 15$).
- Finally, the algorithm terminates because stop $\mathcal{R}1$ is satisfied. Output the resulting local structure $\mathcal{P}$, which is depicted in Figure 3(f).

The ultimate local $\mathcal{P}$, acquired through orienting rules, is presented in Figure 3.(f). As all edges connected to the target $V_5$ have been oriented (i.e., satisfying stop $\mathcal{R}1$), the learning process can be concluded.

## E. More Results on Experiments

All experiments were performed with Intel 2.90GHz and 2.89 GHz CPUs and 128 GB of memory. We give more experimental results here.

Table 6 provides a detailed overview of the network statistics used in this paper.

*Table 6.* Statistics on the Networks.

| Networks | Num.Variables | Avg degree | Max in-degree |
|---|---|---|---|
| *MILDEW* | 35 | 2.63 | 3 |
| *ALARM* | 37 | 2.49 | 4 |
| *WIN95PTS* | 76 | 2.95 | 7 |
| *ANDES* | 223 | 3.03 | 6 |

Tables $7 \sim 10$ provide the complete results in Section 5.1.

*Table 7.* Performance Comparisons on MILDEW.Net

| | | Size=1000 | | | | | Size=5000 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Target | Algorithm | **Precision↑** | **Recall↑** | **F1↑** | **Distance↓** | **nTest↓** | **Precision↑** | **Recall↑** | **F1↑** | **Distance↓** | **nTest↓** |
| dm-2 | PC-stable | 0.28±0.09 | 0.28±0.09 | 0.28±0.09 | 1.02±0.13 | 5066.55 | 0.27±0.07 | 0.27±0.07 | 0.27±0.07 | 1.04±0.10 | 8917.40 |
| | FCI | 0.70±0.25 | 0.64±0.29 | 0.66±0.28 | 0.48±0.39 | 10307.75 | 0.77±0.25 | 0.72±0.29 | 0.73±0.28 | 0.38±0.39 | 16663.83 |
| | RFCI | 0.69±0.26 | 0.62±0.29 | 0.64±0.28 | 0.51±0.39 | 5066.55 | 0.73±0.25 | 0.68±0.28 | 0.70±0.27 | 0.43±0.38 | 8917.40 |
| | MB-by-MB | 0.55±0.11 | 0.62±0.14 | 0.57±0.12 | 0.61±0.17 | 6085.94 | 0.55±0.11 | 0.63±0.15 | 0.57±0.11 | 0.60±0.16 | 21849.32 |
| | CMB | 0.55±0.13 | 0.58±0.14 | 0.56±0.13 | 0.62±0.18 | 2169.34 | 0.56±0.12 | 0.60±0.13 | 0.57±0.12 | 0.60±0.17 | 2573.37 |
| | GraN-LCS | 0.68±0.17 | 0.72±0.18 | 0.68±0.16 | 0.46±0.22 | - | 0.69±0.18 | 0.72±0.19 | 0.68±0.17 | 0.46±0.24 | - |
| | MMB-by-MMB | **0.94±0.15** | **0.93±0.17** | **0.93±0.17** | **0.10±0.24** | **732.89** | **0.98±0.08** | **0.97±0.11** | **0.97±0.10** | **0.04±0.15** | **1064.33** |
| dm-1 | PC-stable | 0.28±0.08 | 0.28±0.09 | 0.28±0.08 | 1.02±0.12 | 5032.70 | 0.27±0.07 | 0.27±0.07 | 0.27±0.07 | 1.04±0.10 | 8930.25 |
| | FCI | 0.83±0.27 | 0.83±0.27 | 0.83±0.27 | 0.24±0.39 | 10260.86 | 0.85±0.27 | 0.85±0.26 | 0.85±0.27 | 0.21±0.38 | 16637.09 |
| | RFCI | 0.81±0.28 | 0.81±0.28 | 0.81±0.28 | 0.27±0.39 | 5032.70 | 0.84±0.26 | 0.84±0.27 | 0.84±0.27 | 0.22±0.38 | 8930.25 |
| | MB-by-MB | 0.51±0.13 | 0.57±0.18 | 0.52±0.14 | 0.67±0.20 | 4596.90 | 0.50±0.12 | 0.58±0.18 | 0.51±0.13 | 0.68±0.19 | 15864.89 |
| | CMB | 0.49±0.15 | 0.52±0.17 | 0.50±0.15 | 0.71±0.22 | 3440.64 | 0.50±0.13 | 0.54±0.16 | 0.51±0.14 | 0.69±0.20 | 3661.29 |
| | GraN-LCS | 0.70±0.22 | 0.78±0.20 | 0.73±0.22 | 0.39±0.30 | - | 0.67±0.21 | 0.74±0.20 | 0.69±0.20 | 0.43±0.29 | - |
| | MMB-by-MMB | **0.95±0.15** | **0.95±0.15** | **0.95±0.15** | **0.07±0.22** | **392.95** | **0.97±0.13** | **0.97±0.13** | **0.97±0.13** | **0.04±0.18** | **613.61** |
| foto-4 | Pc-stable | 0.25±0.04 | 0.25±0.04 | 0.25±0.04 | 1.06±0.06 | 4983.90 | 0.25±0.04 | 0.25±0.04 | 0.25±0.04 | 1.06±0.06 | 8852.61 |
| | FCI | 0.71±0.27 | 0.70±0.27 | 0.70±0.27 | 0.43±0.39 | 10187.20 | 0.77±0.30 | 0.77±0.30 | 0.77±0.30 | 0.33±0.43 | 16496.33 |
| | RFCI | 0.62±0.28 | 0.61±0.28 | 0.61±0.28 | 0.56±0.40 | 4983.90 | 0.74±0.30 | 0.74±0.30 | 0.74±0.30 | 0.37±0.43 | 8852.61 |
| | MB-by-MB | 0.29±0.07 | 0.37±0.14 | 0.30±0.08 | 0.97±0.13 | 6715.11 | 0.27±0.04 | 0.34±0.13 | 0.28±0.05 | 1.00±0.09 | 23781.90 |
| | CMB | 0.30±0.09 | 0.35±0.13 | 0.31±0.09 | 0.97±0.13 | 2335.02 | 0.29±0.07 | 0.34±0.12 | 0.30±0.08 | 0.99±0.11 | 3097.41 |
| | GraN-LCS | 0.32±0.12 | 0.40±0.18 | 0.34±0.13 | 0.92±0.19 | - | 0.33±0.12 | 0.40±0.18 | 0.34±0.14 | 0.92±0.19 | - |
| | MMB-by-MMB | **0.93±0.12** | **0.92±0.14** | **0.92±0.14** | **0.11±0.20** | **820.24** | **0.95±0.15** | **0.95±0.15** | **0.95±0.15** | **0.07±0.22** | **1207.26** |
| dm-4 | Pc-stable | 0.33±0.18 | 0.33±0.18 | 0.33±0.18 | 0.94±0.26 | 5071.66 | 0.33±0.18 | 0.33±0.18 | 0.33±0.18 | 0.95±0.25 | 8972.31 |
| | FCI | 0.77±0.20 | 0.70±0.20 | 0.72±0.20 | 0.39±0.29 | 10338.28 | 0.84±0.15 | 0.77±0.20 | 0.80±0.18 | 0.29±0.26 | 16762.44 |
| | RFCI | 0.72±0.21 | 0.64±0.19 | 0.67±0.20 | 0.47±0.28 | 5071.66 | 0.81±0.15 | 0.73±0.20 | 0.76±0.18 | 0.35±0.26 | 8972.31 |
| | MB-by-MB | 0.59±0.14 | 0.69±0.18 | 0.62±0.14 | 0.53±0.20 | 7075.49 | 0.60±0.15 | 0.71±0.18 | 0.63±0.16 | 0.51±0.22 | 28815.65 |
| | CMB | 0.60±0.15 | 0.63±0.16 | 0.61±0.15 | 0.55±0.22 | 2325.96 | 0.58±0.15 | 0.63±0.15 | 0.60±0.15 | 0.57±0.21 | 3638.17 |
| | GraN-LCS | 0.57±0.17 | 0.59±0.19 | 0.56±0.16 | 0.62±0.23 | - | 0.60±0.20 | 0.61±0.21 | 0.59±0.19 | 0.58±0.26 | - |
| | MMB-by-MMB | **0.95±0.13** | **0.91±0.16** | **0.92±0.15** | **0.11±0.21** | **527.14** | **0.99±0.05** | **0.98±0.09** | **0.98±0.08** | **0.03±0.11** | **690.57** |

Note: The symbol '-' indicates that GraN-LCS does not output this information. MMB-by-MMB with the second best result is underlined. ↑ means a higher value is better, and vice versa.

*Table 8.* Performance Comparisons on ALARM.Net

| | | Size=1000 | | | | | Size=5000 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Target | Algorithm | **Precision↑** | **Recall↑** | **F1↑** | **Distance↓** | **nTest↓** | **Precision↑** | **Recall↑** | **F1↑** | **Distance↓** | **nTest↓** |
| LVEDVOLUME | PC-stable | 0.56±0.16 | 0.55±0.14 | 0.56±0.15 | 0.63±0.20 | 3515.85 | 0.56±0.16 | 0.55±0.14 | 0.56±0.15 | 0.63±0.20 | 4878.73 |
| | FCI | 0.78±0.25 | 0.77±0.25 | 0.77±0.25 | 0.32±0.35 | 7552.66 | 0.84±0.24 | 0.83±0.24 | 0.83±0.24 | 0.24±0.34 | 10173.30 |
| | RFCI | 0.78±0.25 | 0.77±0.24 | 0.77±0.24 | 0.32±0.34 | 3515.85 | 0.83±0.23 | 0.83±0.24 | 0.83±0.24 | 0.24±0.33 | 4878.73 |
| | MB-by-MB | 0.46±0.18 | 0.46±0.18 | 0.45±0.17 | 0.77±0.25 | 1531.45 | 0.44±0.15 | 0.44±0.16 | 0.43±0.14 | 0.81±0.20 | 4196.87 |
| | CMB | 0.44±0.21 | 0.43±0.19 | 0.43±0.20 | 0.80±0.28 | 1471.71 | 0.44±0.20 | 0.43±0.18 | 0.43±0.18 | 0.81±0.26 | 1992.32 |
| | GraN-LCS | 0.58±0.15 | 0.57±0.14 | 0.57±0.14 | 0.61±0.20 | - | 0.58±0.13 | 0.58±0.13 | 0.58±0.13 | 0.60±0.18 | - |
| | MMB-by-MMB | **0.97±0.12** | **0.96±0.12** | **0.96±0.12** | **0.05±0.16** | **324.09** | **0.98±0.09** | **0.98±0.09** | **0.98±0.09** | **0.03±0.12** | **344.51** |
| VENTTUBE | PC-stable | 0.58±0.18 | 0.56±0.15 | 0.57±0.16 | 0.61±0.22 | 3529.88 | 0.58±0.18 | 0.57±0.15 | 0.57±0.16 | 0.61±0.22 | 4846.68 |
| | FCI | 0.76±0.27 | 0.75±0.26 | 0.74±0.26 | 0.36±0.37 | 7629.62 | 0.80±0.25 | 0.78±0.26 | 0.79±0.26 | 0.30±0.36 | 10199.60 |
| | RFCI | 0.74±0.27 | 0.73±0.27 | 0.73±0.27 | 0.39±0.38 | 3529.88 | 0.76±0.26 | 0.74±0.27 | 0.74±0.26 | 0.37±0.37 | 4846.68 |
| | MB-by-MB | 0.36±0.12 | 0.37±0.14 | 0.36±0.12 | 0.90±0.17 | 3081.04 | 0.32±0.11 | 0.32±0.11 | 0.32±0.11 | 0.97±0.15 | 8077.43 |
| | CMB | 0.32±0.13 | 0.31±0.11 | 0.31±0.12 | 0.97±0.17 | 3537.93 | 0.29±0.10 | 0.29±0.10 | 0.29±0.09 | 1.00±0.13 | 4630.07 |
| | GraN-LCS | 0.43±0.13 | 0.46±0.16 | 0.44±0.14 | 0.80±0.20 | - | 0.44±0.13 | 0.47±0.15 | 0.45±0.13 | 0.79±0.18 | - |
| | MMB-by-MMB | **0.87±0.24** | **0.85±0.24** | **0.85±0.24** | **0.21±0.34** | **570.23** | **0.89±0.21** | **0.87±0.22** | **0.87±0.22** | **0.18±0.31** | **736.22** |
| CATECHOL | PC-stable | 0.24±0.05 | 0.24±0.05 | 0.24±0.05 | 1.07±0.07 | 3482.37 | 0.24±0.05 | 0.24±0.05 | 0.24±0.05 | 1.07±0.07 | 4758.09 |
| | FCI | 0.63±0.22 | 0.52±0.18 | 0.55±0.19 | 0.64±0.27 | 7565.43 | 0.76±0.17 | 0.66±0.19 | 0.69±0.17 | 0.44±0.25 | 10110.46 |
| | RFCI | 0.59±0.20 | 0.47±0.16 | 0.51±0.17 | 0.70±0.23 | 3482.37 | 0.73±0.17 | 0.63±0.18 | 0.66±0.17 | 0.49±0.24 | 4758.09 |
| | MB-by-MB | 0.31±0.09 | 0.38±0.13 | 0.33±0.10 | 0.94±0.14 | 6336.96 | 0.30±0.08 | 0.38±0.12 | 0.32±0.09 | 0.95±0.13 | 20601.93 |
| | CMB | 0.29±0.07 | 0.35±0.11 | 0.30±0.08 | 0.97±0.11 | 5036.08 | 0.30±0.07 | 0.36±0.10 | 0.32±0.08 | 0.96±0.11 | 4204.28 |
| | GraN-LCS | 0.26±0.07 | 0.26±0.09 | 0.26±0.08 | 1.05±0.11 | - | 0.25±0.07 | 0.25±0.06 | 0.25±0.06 | 1.06±0.09 | - |
| | MMB-by-MMB | **0.92±0.15** | **0.87±0.17** | **0.88±0.16** | **0.17±0.23** | **936.64** | **0.95±0.11** | **0.94±0.13** | **0.94±0.13** | **0.09±0.18** | **1128.77** |
| STROKEVOLUME | PC-stable | 0.57±0.17 | 0.56±0.13 | 0.56±0.14 | 0.62±0.20 | 3470.71 | 0.57±0.17 | 0.56±0.13 | 0.56±0.14 | 0.62±0.20 | 4840.23 |
| | FCI | 0.78±0.25 | 0.77±0.25 | 0.77±0.25 | 0.33±0.35 | 7483.00 | 0.84±0.23 | 0.84±0.23 | 0.84±0.23 | 0.23±0.33 | 10171.28 |
| | RFCI | 0.72±0.25 | 0.71±0.24 | 0.71±0.25 | 0.41±0.35 | 3470.71 | 0.83±0.24 | 0.83±0.24 | 0.83±0.24 | 0.24±0.34 | 4840.23 |
| | MB-by-MB | 0.43±0.18 | 0.48±0.21 | 0.43±0.18 | 0.79±0.25 | 2076.76 | 0.37±0.15 | 0.41±0.17 | 0.38±0.15 | 0.88±0.21 | 6377.99 |
| | CMB | 0.44±0.20 | 0.45±0.20 | 0.44±0.19 | 0.79±0.27 | 2197.08 | 0.37±0.19 | 0.37±0.19 | 0.37±0.19 | 0.89±0.26 | 2597.72 |
| | GraN-LCS | 0.42±0.13 | 0.46±0.16 | 0.43±0.14 | 0.80±0.20 | - | 0.42±0.13 | 0.46±0.16 | 0.43±0.14 | 0.80±0.20 | - |
| | MMB-by-MMB | **0.95±0.15** | **0.95±0.16** | **0.95±0.16** | **0.08±0.22** | **566.39** | **0.98±0.09** | **0.98±0.09** | **0.98±0.09** | **0.02±0.12** | **698.17** |

Note: The symbol '-' indicates that GraN-LCS does not output this information. MMB-by-MMB with the second best result is underlined. ↑ means a higher value is better, and vice versa.

*Table 9.* Performance Comparisons on WIN95PTS.Net

| | | Size=1000 | | | | | Size=5000 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Target | Algorithm | **Precision↑** | **Recall↑** | **F1↑** | **Distance↓** | **nTest↓** | **Precision↑** | **Recall↑** | **F1↑** | **Distance↓** | **nTest↓** |
| Problem5 | Pc-stable | 0.52±0.07 | 0.52±0.07 | 0.52±0.07 | 0.68±0.10 | 12657.62 | 0.53±0.07 | 0.53±0.08 | 0.53±0.07 | 0.67±0.10 | 26398.40 |
| | FCI | 0.69±0.24 | 0.69±0.24 | 0.68±0.24 | 0.45±0.34 | 25417.23 | 0.77±0.27 | 0.76±0.27 | 0.76±0.27 | 0.34±0.38 | 43850.55 |
| | RFCI | 0.67±0.23 | 0.66±0.23 | 0.66±0.23 | 0.48±0.33 | 12657.62 | 0.77±0.27 | 0.75±0.28 | 0.76±0.28 | 0.35±0.39 | 26398.40 |
| | MB-by-MB | 0.46±0.19 | 0.50±0.22 | 0.47±0.20 | 0.75±0.28 | 13633.52 | NA | NA | NA | NA | NA |
| | CMB | 0.57±0.16 | 0.60±0.18 | 0.58±0.17 | 0.59±0.23 | 4757.95 | 0.54±0.15 | 0.58±0.17 | 0.56±0.16 | 0.63±0.22 | **5413.78** |
| | GraN-LCS | 0.48±0.14 | 0.50±0.15 | 0.48±0.14 | 0.73±0.20 | - | 0.48±0.16 | 0.48±0.17 | 0.48±0.16 | 0.74±0.22 | - |
| | MMB-by-MMB | **0.90±0.20** | **0.90±0.20** | **0.89±0.20** | **0.15±0.29** | **3907.42** | **0.93±0.18** | **0.92±0.19** | **0.92±0.19** | **0.11±0.27** | 12372.20 |
| HrglssDrtnAftrPrnt | PC-stable | 0.78±0.08 | 0.77±0.06 | 0.78±0.07 | 0.32±0.10 | 12637.44 | 0.77±0.06 | 0.76±0.04 | 0.76±0.05 | 0.34±0.07 | 25058.00 |
| | FCI | 0.80±0.24 | 0.80±0.24 | 0.80±0.24 | 0.28±0.34 | 25651.52 | 0.83±0.24 | 0.83±0.25 | 0.83±0.25 | 0.24±0.35 | 42604.71 |
| | RFCI | 0.81±0.24 | 0.81±0.24 | 0.81±0.24 | 0.27±0.34 | 12637.44 | 0.82±0.25 | 0.82±0.25 | 0.82±0.25 | 0.25±0.35 | 25058.00 |
| | MB-by-MB | 0.46±0.09 | 0.46±0.09 | 0.46±0.09 | 0.77±0.13 | 14169.84 | 0.45±0.11 | 0.45±0.11 | 0.45±0.11 | 0.78±0.15 | 45118.80 |
| | CMB | 0.48±0.23 | 0.48±0.23 | 0.48±0.23 | 0.73±0.33 | 7933.74 | 0.42±0.19 | 0.42±0.19 | 0.42±0.19 | 0.82±0.26 | 11783.44 |
| | GraN-LCS | 0.39±0.14 | 0.39±0.14 | 0.39±0.14 | 0.86±0.20 | - | 0.43±0.13 | 0.43±0.13 | 0.43±0.13 | 0.80±0.18 | - |
| | MMB-by-MMB | **0.92±0.18** | **0.92±0.18** | **0.92±0.18** | **0.11±0.25** | **1054.52** | **0.92±0.20** | **0.92±0.20** | **0.92±0.20** | **0.11±0.29** | **2029.79** |
| Problem1 | PC-stable | 0.72±0.08 | 0.72±0.08 | 0.72±0.08 | 0.40±0.12 | 12629.93 | 0.73±0.07 | 0.73±0.07 | 0.73±0.07 | 0.39±0.10 | 25137.35 |
| | FCI | 0.97±0.10 | 0.97±0.12 | 0.97±0.11 | 0.05±0.15 | 25540.04 | 0.97±0.11 | 0.97±0.12 | 0.97±0.11 | 0.04±0.16 | 42790.89 |
| | RFCI | 0.94±0.15 | 0.93±0.17 | 0.94±0.16 | 0.09±0.23 | 12629.93 | 0.97±0.11 | 0.96±0.12 | 0.96±0.12 | 0.05±0.17 | 25137.35 |
| | MB-by-MB | 0.72±0.22 | 0.78±0.24 | 0.73±0.22 | 0.37±0.31 | 16082.27 | NA | NA | NA | NA | NA |
| | CMB | 0.62±0.13 | 0.63±0.14 | 0.62±0.13 | 0.54±0.19 | 11702.02 | 0.63±0.13 | 0.63±0.13 | 0.63±0.13 | 0.53±0.18 | 103663.78 |
| | GraN-LCS | 0.57±0.15 | 0.57±0.16 | 0.57±0.15 | 0.61±0.21 | - | 0.65±0.21 | 0.66±0.22 | 0.65±0.21 | 0.49±0.30 | - |
| | MMB-by-MMB | **0.97±0.11** | **0.97±0.14** | **0.97±0.13** | **0.05±0.19** | **2895.81** | **0.99±0.06** | **0.99±0.07** | **0.99±0.07** | **0.02±0.10** | **6931.07** |
| GDIOUT | PC-stable | 0.23±0.06 | 0.23±0.06 | 0.23±0.06 | 1.08±0.08 | 12715.64 | 0.23±0.06 | 0.23±0.06 | 0.23±0.06 | 1.08±0.08 | 26274.96 |
| | FCI | 0.79±0.16 | 0.70±0.19 | 0.73±0.18 | 0.39±0.25 | 25501.34 | 0.78±0.21 | 0.73±0.22 | 0.74±0.22 | 0.37±0.31 | 43773.13 |
| | RFCI | 0.71±0.19 | 0.62±0.20 | 0.65±0.19 | 0.50±0.27 | 12715.64 | 0.75±0.23 | 0.70±0.24 | 0.71±0.23 | 0.41±0.33 | 26274.96 |
| | MB-by-MB | 0.31±0.12 | 0.44±0.18 | 0.34±0.13 | 0.92±0.19 | 14327.88 | NA | NA | NA | NA | NA |
| | CMB | 0.27±0.09 | 0.37±0.15 | 0.29±0.10 | 0.99±0.15 | 6279.86 | 0.26±0.07 | 0.33±0.14 | 0.27±0.08 | 1.02±0.13 | **6850.83** |
| | GraN-LCS | 0.24±0.08 | 0.26±0.11 | 0.25±0.08 | 1.06±0.12 | - | 0.25±0.10 | 0.28±0.14 | 0.26±0.10 | 1.05±0.15 | - |
| | MMB-by-MMB | **0.87±0.17** | **0.87±0.19** | **0.85±0.18** | **0.22±0.26** | **5514.87** | **0.87±0.25** | **0.87±0.26** | **0.86±0.25** | **0.20±0.36** | 14635.54 |
| PrData | PC-stable | 0.24±0.10 | 0.24±0.10 | 0.24±0.10 | 1.07±0.14 | 12566.90 | 0.24±0.10 | 0.24±0.10 | 0.24±0.10 | 1.07±0.14 | 26315.51 |
| | FCI | 0.80±0.18 | 0.68±0.17 | 0.71±0.17 | 0.40±0.24 | 25398.65 | 0.77±0.12 | 0.71±0.14 | 0.73±0.13 | 0.39±0.18 | 44028.39 |
| | RFCI | 0.72±0.22 | 0.61±0.20 | 0.64±0.20 | 0.51±0.29 | 12566.90 | 0.76±0.13 | 0.69±0.14 | 0.71±0.13 | 0.41±0.19 | 26315.51 |
| | MB-by-MB | 0.32±0.14 | 0.47±0.20 | 0.35±0.15 | 0.90±0.22 | 69692.41 | NA | NA | NA | NA | NA |
| | CMB | 0.30±0.15 | 0.39±0.19 | 0.31±0.15 | 0.96±0.22 | 11953.29 | 0.27±0.11 | 0.37±0.17 | 0.28±0.11 | 0.99±0.17 | 69763.20 |
| | GraN-LCS | 0.24±0.08 | 0.25±0.09 | 0.24±0.08 | 1.07±0.11 | - | 0.25±0.10 | 0.26±0.11 | 0.25±0.10 | 1.06±0.14 | - |
| | MMB-by-MMB | **0.84±0.15** | **0.77±0.15** | **0.77±0.14** | **0.32±0.19** | **6833.29** | **0.90±0.13** | **0.90±0.14** | **0.88±0.14** | **0.18±0.19** | **15581.97** |

Note: The symbol '-' indicates that GraN-LCS does not output this information. MMB-by-MMB with the second best result is underlined. ↑ means a higher value is better, and vice versa. NA entries for MB-by-MB demonstrate that the runtime exceeds a certain threshold.

*Table 10.* Performance Comparisons on ANDES.Net

| | | Size=1000 | | | | | Size=5000 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Target | Algorithm | **Precision↑** | **Recall↑** | **F1↑** | **Distance↓** | **nTest↓** | **Precision↑** | **Recall↑** | **F1↑** | **Distance↓** | **nTest↓** |
| RApp3($V_5$) | PC-stable | 0.23±0.07 | 0.23±0.07 | 0.23±0.07 | 1.09±0.10 | 234677.37 | 0.23±0.07 | 0.23±0.07 | 0.23±0.07 | 1.09±0.10 | 439483.08 |
| | FCI | 0.70±0.24 | 0.68±0.25 | 0.68±0.24 | 0.45±0.34 | 901063.34 | 0.79±0.24 | 0.78±0.25 | 0.79±0.24 | 0.30±0.34 | 1584682.77 |
| | RFCI | 0.66±0.24 | 0.64±0.24 | 0.65±0.24 | 0.50±0.34 | 234677.37 | 0.78±0.24 | 0.77±0.25 | 0.77±0.25 | 0.32±0.35 | 439483.08 |
| | MB-by-MB | 0.34±0.07 | 0.43±0.12 | 0.36±0.08 | 0.89±0.12 | 24239.83 | 0.39±0.08 | 0.56±0.12 | 0.43±0.09 | 0.78±0.12 | 44225.00 |
| | CMB | 0.33±0.06 | 0.38±0.09 | 0.34±0.07 | 0.92±0.09 | 79932.47 | 0.32±0.05 | 0.39±0.09 | 0.34±0.07 | 0.93±0.09 | 145631.64 |
| | GraN-LCS | 0.39±0.12 | 0.46±0.16 | 0.40±0.12 | 0.84±0.17 | - | 0.38±0.13 | 0.42±0.16 | 0.39±0.13 | 0.86±0.19 | - |
| | MMB-by-MMB | **0.91±0.15** | **0.90±0.17** | **0.89±0.17** | **0.16±0.24** | **5043.44** | **0.98±0.07** | **0.98±0.07** | **0.98±0.07** | **0.03±0.10** | **4595.15** |
| SNode-27 | PC-stable | 0.26±0.04 | 0.26±0.04 | 0.26±0.04 | 1.05±0.06 | 234677.37 | 0.25±0.00 | 0.25±0.00 | 0.25±0.00 | 1.06±0.00 | 439483.08 |
| | FCI | 0.87±0.16 | 0.80±0.21 | 0.82±0.19 | 0.25±0.27 | 901063.34 | 0.90±0.16 | 0.87±0.19 | 0.88±0.18 | 0.17±0.25 | 1584682.77 |
| | RFCI | 0.85±0.18 | 0.79±0.22 | 0.81±0.21 | 0.28±0.29 | 234677.37 | 0.88±0.17 | 0.85±0.20 | 0.86±0.19 | 0.20±0.27 | 439483.08 |
| | MB-by-MB | 0.57±0.08 | 0.63±0.12 | 0.58±0.08 | 0.58±0.12 | 34010.75 | NA | NA | NA | NA | NA |
| | CMB | 0.56±0.06 | 0.61±0.10 | 0.58±0.08 | 0.60±0.11 | 126675.88 | 0.56±0.07 | 0.61±0.09 | 0.57±0.07 | 0.60±0.10 | 257776.30 |
| | GraN-LCS | 0.67±0.14 | 0.70±0.16 | 0.66±0.13 | 0.47±0.19 | - | 0.71±0.15 | 0.71±0.16 | 0.69±0.14 | 0.44±0.20 | - |
| | MMB-by-MMB | **0.95±0.15** | **0.93±0.18** | **0.94±0.17** | **0.09±0.24** | **4794.07** | **0.95±0.16** | **0.95±0.17** | **0.95±0.17** | **0.07±0.24** | **7104.74** |
| SNode-21 | PC-stable | 0.25±0.00 | 0.25±0.00 | 0.25±0.00 | 1.06±0.00 | 234677.37 | 0.25±0.00 | 0.25±0.00 | 0.25±0.00 | 1.06±0.00 | 439483.08 |
| | FCI | 0.83±0.22 | 0.84±0.21 | 0.83±0.22 | 0.24±0.31 | 901063.34 | 0.87±0.22 | 0.87±0.22 | 0.87±0.22 | 0.19±0.31 | 1584682.77 |
| | RFCI | 0.80±0.24 | 0.81±0.24 | 0.80±0.24 | 0.28±0.34 | 234677.37 | 0.87±0.23 | 0.86±0.23 | 0.86±0.23 | 0.19±0.32 | 439483.08 |
| | MB-by-MB | 0.31±0.12 | 0.42±0.20 | 0.33±0.14 | 0.93±0.20 | 18424.49 | 0.33±0.10 | 0.49±0.13 | 0.36±0.11 | 0.88±0.15 | 82493.83 |
| | CMB | 0.31±0.14 | 0.36±0.17 | 0.32±0.14 | 0.96±0.20 | 94615.12 | 0.30±0.13 | 0.35±0.16 | 0.31±0.13 | 0.97±0.19 | 232544.08 |
| | GraN-LCS | 0.47±0.16 | 0.57±0.19 | 0.50±0.17 | 0.71±0.24 | - | 0.47±0.18 | 0.56±0.21 | 0.50±0.18 | 0.71±0.26 | - |
| | MMB-by-MMB | **0.90±0.19** | **0.92±0.16** | **0.90±0.18** | **0.14±0.26** | **5086.75** | **0.96±0.10** | **0.97±0.09** | **0.96±0.10** | **0.05±0.14** | **7601.29** |
| RApp4 | PC-stable | 0.46±0.09 | 0.46±0.09 | 0.46±0.09 | 0.77±0.13 | 234677.37 | 0.46±0.09 | 0.46±0.09 | 0.46±0.09 | 0.77±0.13 | 439483.08 |
| | FCI | 0.79±0.24 | 0.78±0.24 | 0.78±0.24 | 0.32±0.34 | 901063.34 | 0.84±0.23 | 0.84±0.23 | 0.84±0.24 | 0.23±0.33 | 1584682.77 |
| | RFCI | 0.79±0.24 | 0.79±0.24 | 0.79±0.24 | 0.32±0.34 | 234677.37 | 0.84±0.23 | 0.83±0.23 | 0.83±0.24 | 0.24±0.33 | 439483.08 |
| | MB-by-MB | 0.26±0.14 | 0.29±0.16 | 0.26±0.14 | 1.04±0.20 | 18761.00 | NA | NA | NA | NA | NA |
| | CMB | 0.25±0.16 | 0.26±0.17 | 0.25±0.16 | 1.06±0.23 | 87956.70 | 0.23±0.12 | 0.23±0.12 | 0.23±0.12 | 1.09±0.17 | 212387.71 |
| | GraN-LCS | 0.36±0.16 | 0.45±0.19 | 0.38±0.16 | 0.87±0.23 | - | 0.38±0.17 | 0.48±0.22 | 0.40±0.18 | 0.84±0.26 | - |
| | MMB-by-MMB | **0.91±0.22** | **0.93±0.19** | **0.91±0.21** | **0.12±0.30** | **3153.11** | **0.97±0.12** | **0.98±0.09** | **0.97±0.11** | **0.04±0.15** | **1430.12** |
| SNode-4 | Pc-stable | 0.46±0.09 | 0.46±0.09 | 0.46±0.09 | 0.76±0.13 | 234677.37 | 0.46±0.09 | 0.46±0.09 | 0.46±0.09 | 0.76±0.13 | 439483.08 |
| | FCI | 0.89±0.17 | 0.88±0.18 | 0.88±0.18 | 0.17±0.26 | 901063.34 | 0.93±0.15 | 0.91±0.18 | 0.91±0.17 | 0.12±0.24 | 1584682.77 |
| | RFCI | 0.87±0.20 | 0.86±0.20 | 0.86±0.20 | 0.19±0.29 | 234677.37 | 0.92±0.16 | 0.91±0.18 | 0.91±0.17 | 0.13±0.25 | 439483.08 |
| | MB-by-MB 0.23±0.12 | 0.24±0.13 | 0.23±0.12 | 1.08±0.18 | 21378.94 | 0.27±0.03 | 0.31±0.10 | 0.28±0.05 | 1.02±0.08 | 55156.33 | |
| | CMB | 0.25±0.16 | 0.26±0.17 | 0.26±0.16 | 1.05±0.23 | 120151.52 | 0.24±0.17 | 0.25±0.17 | 0.24±0.16 | 1.07±0.23 | 248697.33 |
| | GraN-LCS | 0.35±0.12 | 0.47±0.14 | 0.37±0.12 | 0.87±0.17 | - | 0.37±0.15 | 0.45±0.18 | 0.38±0.15 | 0.86±0.22 | - |
| | MMB-by-MMB | **0.90±0.22** | **0.92±0.17** | **0.90±0.21** | **0.14±0.29** | **3125.87** | **0.98±0.07** | **0.98±0.07** | **0.98±0.07** | **0.03±0.11** | **1913.94** |
| SNode-47 | PC-stable | 0.47±0.08 | 0.47±0.08 | 0.47±0.08 | 0.75±0.12 | 234677.37 | 0.47±0.08 | 0.47±0.08 | 0.47±0.08 | 0.75±0.12 | 439483.08 |
| | FCI | 0.91±0.15 | 0.89±0.17 | 0.90±0.16 | 0.15±0.24 | 901063.34 | 0.94±0.12 | 0.92±0.16 | 0.93±0.14 | 0.10±0.21 | 1584682.77 |
| | RFCI | 0.91±0.15 | 0.89±0.17 | 0.90±0.16 | 0.15±0.24 | 234677.37 | 0.94±0.12 | 0.92±0.16 | 0.93±0.14 | 0.10±0.21 | 439483.08 |
| | MB-by-MB | 0.27±0.12 | 0.30±0.14 | 0.28±0.12 | 1.01±0.18 | 14689.66 | 0.23±0.13 | 0.26±0.14 | 0.24±0.13 | 1.07±0.18 | 55457.00 |
| | CMB | 0.26±0.13 | 0.26±0.13 | 0.26±0.13 | 1.04±0.19 | 124314.84 | 0.25±0.12 | 0.25±0.13 | 0.25±0.12 | 1.06±0.17 | 247077.40 |
| | GraN-LCS | 0.32±0.13 | 0.41±0.18 | 0.34±0.14 | 0.92±0.20 | - | 0.32±0.15 | 0.38±0.21 | 0.33±0.16 | 0.94±0.24 | - |
| | MMB-by-MMB | **0.94±0.18** | **0.95±0.13** | **0.94±0.16** | **0.08±0.23** | **1693.23** | **1.00±0.00** | **1.00±0.00** | **1.00±0.00** | **0.00±0.00** | **1590.80** |
| SNode-24 | PC-stable | 0.47±0.08 | 0.47±0.08 | 0.47±0.08 | 0.75±0.11 | 234677.37 | 0.47±0.08 | 0.47±0.08 | 0.47±0.08 | 0.75±0.12 | 439483.08 |
| | FCI | 0.65±0.17 | 0.70±0.16 | 0.66±0.16 | 0.48±0.23 | 901063.34 | 0.72±0.19 | 0.79±0.15 | 0.74±0.17 | 0.37±0.25 | 1584682.77 |
| | RFCI | 0.63±0.18 | 0.68±0.17 | 0.65±0.17 | 0.50±0.24 | 234677.37 | 0.71±0.19 | 0.79±0.16 | 0.74±0.18 | 0.37±0.25 | 439483.08 |
| | MB-by-MB | 0.40±0.16 | 0.48±0.23 | 0.42±0.17 | 0.82±0.25 | 20934.91 | NA | NA | NA | NA | NA |
| | CMB | 0.52±0.11 | 0.58±0.17 | 0.54±0.13 | 0.65±0.18 | 54165.45 | 0.54±0.09 | 0.61±0.15 | 0.56±0.10 | 0.62±0.14 | 69412.77 |
| | GraN-LCS | 0.51±0.12 | 0.60±0.13 | 0.53±0.12 | 0.66±0.17 | - | 0.56±0.11 | 0.63±0.15 | 0.58±0.12 | 0.60±0.17 | - |
| | MMB-by-MMB | **0.88±0.20** | **0.90±0.18** | **0.88±0.20** | **0.17±0.28** | **7202.50** | **0.99±0.06** | **0.99±0.06** | **0.99±0.07** | **0.02±0.09** | **15345.62** |

Note: The symbol '-' indicates that GraN-LCS does not output this information. MMB-by-MMB with the second best result is underlined. ↑ means a higher value is better, and vice versa. NA entries for MB-by-MB demonstrate that the runtime exceeds a certain threshold.