Mach Learn (2016) 104:25–54
DOI 10.1007/s10994-016-5545-0

Swamping and masking in Markov boundary discovery

Xuqing Liu[1,2] · **Xinsheng Liu**[1]

Received: 9 May 2014 / Accepted: 16 January 2016 / Published online: 9 March 2016

Abstract This paper considers the problems of swamping and masking in Markov boundary discovery for a target variable. There are two potential reasons for swamping and masking: one is incorrectness of some conditional independence (CI) tests, and the other is violation of local composition. First, we explain why the incorrectness of CI tests may lead to swamping and masking, analyze how to reduce the incorrectness of CI tests, and build an algorithm called LRH under local composition. For convenience, we integrate the two existing algorithms, IAMB and KIAMB, and our LRH into an algorithmic framework called LCMB. Second, since LCMB may prematurely stop searching if local composition is violated, a theoretical improvement on LCMB is made as follows: we analyze how to resume the stopped search of LCMB, construct a corresponding algorithmic framework called WLCMB, and show that its correctness only needs a more relaxed condition than that of LCMB. Finally, we apply LCMB and WLCMB to a number of Bayesian networks. The experimental results reveal that LRH is much more efficient than the existing two LCMB algorithms and that WLCMB can further improve LCMB.

Keywords Bayesian network · Markov blanket · Markov boundary · Masking · Swamping

This research was supported by the National Natural Science Foundation of China (No.: 61374183) and the Humanistic and Social Science Foundation of Ministry of Education of China (No.: 12YJA630122).

Editor: James Cussens.

✉ Xinsheng Liu
xsliu@nuaa.edu.cn

Xuqing Liu
liuxuqing688@163.com

[1] State Key Laboratory of Mechanics and Control of Mechanical Structures, Institute of Nano Science and Department of Mathematics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China

[2] Faculty of Mathematics and Physics, Huaiyin Institute of Technology, Huai'an 223003, China

1 Introduction

Markov blankets (Mb) and Markov boundaries (MB) are two basic concepts in Bayesian networks (BNs). For a target variable T, its Mb is a variable set conditioned on which all other variables are probabilistically independent of T, and its MB is a minimal Mb; that is, an MB is the smallest set containing all variables carrying the information about T that cannot be obtained from other variables (Pearl 1988).

The discovery of MBs plays a central role in feature selection (Pellet and Elisseeff 2008; Aliferis et al. 2010a, b; Fu and Desmarais 2010). Feature selection aims to identify the minimal subset of features required for probabilistic classification, with the following three-fold objective (Guyon and Elisseeff 2003): improving the prediction performance of the predictors, providing faster and more cost-effective predictors, and facilitating a better understanding of the underlying process that generated the data. Pearl (1988) showed the conditional probability for the target variable given other variables can be replaced by the one with an MB as the conditional set. Pellet and Elisseeff (2008) proved an MB is the theoretically optimal set of features if the faithfulness condition is satisfied. Further, under certain assumptions about the learner and the loss function, MB is the solution to the feature selection problem (Tsamardinos and Aliferis 2003; Masegosa and Moral 2012; Statnikov et al. 2013). Hence, MB discovery techniques are receiving more and more attention in recent years.

In the literature, there have been lots of MB discovery approaches, including independence-based and score-based ones, as well as some hybrid methods. This paper focuses on the former.

The Koller–Sahami (KS) algorithm, put forward by Koller and Sahami (1996), is the first technique of creating a framework used to define the theoretically optimal filter method for a feature selection problem. It provides no theoretical guarantees to soundness (Tsamardinos et al. 2003a). The grow-shrink (GS) algorithm, which was proposed by Margaritis and Thrun (1999, 2000), consists of the growing phase and the shrinking phase. In its growing phase, as long as there exists a variable conditionally dependent on the target given the candidate Markov blanket (CMb), this variable will be added to the CMb until no more such variables exist. All members of an MB as well as some false positives enter the CMb at the end of the growing phase. The shrinking phase detects those false positives and removes them. The GS algorithm was theoretically proven by Margaritis and Thrun (1999) to be correct under the assumption that all the conditional independence (CI) tests are correct. Here, a CI test for a hypothesis is said to be correct, if the corresponding statistical decision is correctly made by using a testing method.

Tsamardinos et al. (2003a) pointed out that GS uses a static and potentially inefficient heuristic in the growing phase, and then they presented a variant of GS called the incremental association Markov boundary (IAMB) algorithm by employing a dynamic heuristic: IAMB reorders the remaining variables by means of an *association* function at each iteration such that the spouses of the target can enter the CMb early and thus fewer false positives are added to the CMb during the growing phase. HITON (Aliferis et al. 2003) also uses a similar static but slightly more efficient heuristic compared to GS.

Similar dynamic heuristics are employed by some variants of IAMB (Tsamardinos and Aliferis 2003; Yaramakala and Margaritis 2005; Zhang et al. 2010). This strategy is also used by divide-and-conquer search techniques, such as the max–min Markov boundary algorithm (Tsamardinos et al. 2003b), the parents and children based Markov boundary (PCMB) algorithm (Peña et al. 2007), the breadth first search of Markov boundary algorithm introduced by

(Fu and Desmarais 2007), and the algorithms included in the algorithmic framework called GLL (Aliferis et al. 2010a).

Under the faithfulness condition, most of these algorithms efficiently retrieve an approximate MB. Peña et al. (2007) relaxed the faithfulness condition to the composition assumption. Based on this relaxation, they put forward a stochastic version of IAMB called KIAMB by introducing a randomization parameter $K \in [0, 1]$. Here, K specifies the trade-off between greediness and randomness in the search: KIAMB with $K = 1$ coincides with IAMB which is completely greedy, while KIAMB with $K = 0$ is a completely random approach expected to discover all the MBs of the target variable with a nonzero probability if running repeatedly for enough times. Further, Statnikov et al. (2013) relaxed the condition for IAMB (also suitable for KIAMB) to be correct to local composition. Another stochastic search technique is the Bayesian stochastic search of Markov boudaries algorithm (Masegosa and Moral 2012), which tries to get all MBs by running a large number of times; it provides some alternative results by scoring the different obtained solutions.

Usually, these algorithms perform well in MB discovery. However, there are two potential problems for them, either of which may lead to what we call *swamping* and *masking* in some situations, and such cases may frequently arise in practice. See (P1) and (P2) below for these two problems. Here, swamping means a true positive becomes a false negative, while masking means a true negative becomes a false positive. These two terminologies are often used for ***outlier detection*** (Ben-Gal 2005; Hadi et al. 2009): *swamping* means some non-outliers are identified as outliers, while *masking* means some outliers are not identified; outliers mask themselves by swamping some non-outliers. We borrow them here to characterize the two results of (P1) (P2) in MB discovery because of their similar behaviors in "masking" themselves and "swamping" others. Definition 2 gives the mathematical description for them.

P1 *Incorrect CI tests may lead to swamping and masking.* Each MB discovery algorithm assumes that all CI tests are correct. This assumption requires the data efficiency of an algorithm. The parents and children based algorithms, such as PCMB and the algorithms in the GLL framework, are data efficient but not time efficient; in contrast, IAMB and KIAMB are time efficient but not data efficient (Schlüter 2014). Once one or more false positives with spuriously high dependence on the target enter the CMb, the cascading errors (Bromberg and Margaritis 2009) caused by them may lead to the exclusion of some true positives. Example 1 provides an illustration.

P2 *Violation of the faithfulness condition (or the local composition assumption) may also lead to swamping and masking.* The faithfulness condition is usually required by the parents and children based algorithms (Peña et al. 2007; Aliferis et al. 2010a), while the relaxed assumption, local composition, is needed by IAMB and KIAMB. However, the faithfulness condition and the local composition assumption may be violated in practice. Example 2 illustrates this possibility.

Example 1 Yaramakala (2004) considered the following scenario: in a BN over $\{T, X, Y_1, Y_2, Z\}$ with the graph given in (a) of Fig. 1 as its directed acyclic graph (DAG), the node Z is a nonmember of the MB for the target T, but it may have the highest association with T because there exist multiple paths for the flow of information between T and Z: $T \rightarrow Y_1 \rightarrow Z$ and $T \rightarrow Y_2 \rightarrow Z$. In this case, Z becomes the first node entering the CMb of IAMB. Peña et al. (2007) instantiated the same scenario to a problem of signal transmission and reception. Yaramakala (2004) and Peña et al. (2007) thought that there may be some true negatives entering the CMb in the growing phase such that the time cost increases. This is natural. However, a more important but neglected problem is that these false positives may bring some cascading errors (Bromberg and Margaritis 2009), which may further cause incorrectness of CI tests

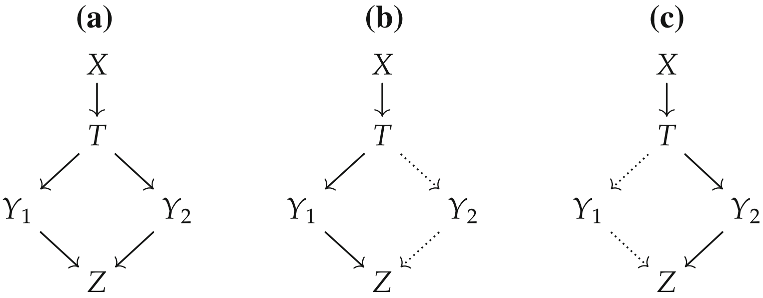

Fig. 1 An illustration on why incorrect CI tests may lead to swamping and masking

and thus the exclusion of some true positives. For example, Y_1 or Y_2 may eventually become a false negative. Hence, it is meaningful to consider the problem of (P1), and what we can do is to prevent too many true negatives with spuriously high dependence on the target from entering the CMb in the growing phase.

Example 2 Consider a target variable T which has three potential features X, Y, and Z. As we know, the total information about T carried by X and Y can be decomposed into: (a) the unique information carried by X, (b) the unique information carried by Y, (c) the redundant information shared by X and Y, and (d) the synergistic information carried jointly by X and Y (Williams and Beer 2010; Rauh et al. 2014). Assume Z carries all of (a)(b)(c) and some (but not all) of (d). It follows that: (1) Z has the highest association with T; (2) T is conditionally independent of X given Z; (3) T is conditionally independent of Y given Z; (4) T is conditionally dependent on $\{X, Y\}$ given Z; (5) T is conditionally independent of Z given X and Y. Then, $\{X, Y\}$ is the unique MB of T in $\{T, X, Y, Z\}$. However, IAMB can not find this MB correctly. Specifically, in the growing phase of IAMB, Z enters the CMb and then it excludes X and Y; in the shrinking phase, Z remains in the CMb. Similarly, it follows that KIAMB can not find $\{X, Y\}$ with a probability not <66.67 % for any value of $K \in [0, 1]$. We no longer consider other above-mentioned algorithms because of the violation of the faithfulness condition. Therefore, it is meaningful to consider the problem of (P2).

These two examples indicate that both the incorrectness of some CI tests and the violation of local composition may lead to swamping and masking. This motivates us to build novel algorithms which are expected to (1) reduce the incorrectness of CI tests, and (2) overcome swamping and masking to a large extent in the case of violating the local composition assumption.

The remainder of this paper is organized as follows. Section 2 provides necessary preliminaries. Section 3 presents the IAMB and KIAMB algorithms, relaxes the notions of Mb and MB, and proves some new results for IAMB and KIAMB. Section 4 addresses the problem of (P1), puts forward a method of including as few true negatives as possible in the growing phase, and builds an algorithm called LRH, which is proven to be correct under the relaxed local composition assumption. The ALARM network is employed to show the data efficiency and time efficiency of LRH. In addition, this section gives a post-processing technique to reduce incorrectness of CI tests kept in the shrinking phase. For convenience, IAMB, KIAMB, and LRH are integrated into an algorithmic framework called LCMB. To resume the search stopped in the growing phase of LCMB, Sect. 5 considers (P2) and constructs an efficient algorithmic framework called WLCMB. The application to ALARM indicates WLCMB can further improve LCMB in data efficiency. Section 6 applies LCMB and WLCMB to several large networks. Section 7 concludes this paper.

2 Preliminary

In the paper, we denote a variable and its value by upper-case and lower-case letters in italics (e.g., X, x), a set of variables and its value by upper-case and lower-case bold letters in italics (e.g., $\boldsymbol{X}$, $\boldsymbol{x}$). The difference between $\boldsymbol{X}$ and $\boldsymbol{Y}$ is denoted by $\boldsymbol{X}\backslash\boldsymbol{Y}$. For brevity, we write $(\boldsymbol{X}\backslash\boldsymbol{Y})\backslash\boldsymbol{Z}$ as $\boldsymbol{X}\backslash\boldsymbol{Y}\backslash\boldsymbol{Z}$. In addition, we use $|\boldsymbol{X}|$ to denote the number of variables involved in $\boldsymbol{X}$.

Suppose we have a joint probability distribution $\mathbb{P}$ over $\boldsymbol{V} \triangleq \{X_1, \ldots, X_p\}$ and a DAG $\mathbb{G}$ with the variables in $\boldsymbol{V}$ as its nodes. We say $(\mathbb{G}, \mathbb{P})$ satisfies the Markov condition if every $X \in \boldsymbol{V}$ is conditionally independent of its nondescendants given its parents. Further, $(\mathbb{G}, \mathbb{P})$ is called a BN if it satisfies the Markov condition. Furthermore, $(\mathbb{G}, \mathbb{P})$ satisfies the faithfulness condition if, based on the Markov condition, $\mathbb{G}$ entails all and only CIs in $\mathbb{P}$ (Pearl 1988; Neapolitan 2004).

Denote $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y}|\boldsymbol{Z}$ (*resp.*, $\boldsymbol{X} \not\perp\!\!\!\perp \boldsymbol{Y}|\boldsymbol{Z}$), if $\boldsymbol{X}$ and $\boldsymbol{Y}$ are conditionally independent (*resp.*, dependent) given $\boldsymbol{Z}$. The following properties describe the relations among CI statements (Pearl 1988; Statnikov et al. 2013). For any $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{W} \subseteq \boldsymbol{V}$, we have (1) *symmetry*: $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y}|\boldsymbol{Z} \Leftrightarrow \boldsymbol{Y} \perp\!\!\!\perp \boldsymbol{X}|\boldsymbol{Z}$; (2) *decomposition*: $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y} \cup \boldsymbol{W}|\boldsymbol{Z}$ implies $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y}|\boldsymbol{Z}$ and $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{W}|\boldsymbol{Z}$; (3) *weak union*: $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y} \cup \boldsymbol{W}|\boldsymbol{Z}$ implies $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y}|\boldsymbol{Z} \cup \boldsymbol{W}$; (4) *contraction*: $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y}|\boldsymbol{Z} \cup \boldsymbol{W}$ and $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{W}|\boldsymbol{Z}$ imply $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y} \cup \boldsymbol{W}|\boldsymbol{Z}$. Further, if $\mathbb{P}$ is strictly positive, then besides (1)∼(4) we also have (5) *intersection*: $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y}|\boldsymbol{Z} \cup \boldsymbol{W}$ and $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{W}|\boldsymbol{Z} \cup \boldsymbol{Y}$ imply $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y} \cup \boldsymbol{W}|\boldsymbol{Z}$. Furthermore, if $\mathbb{P}$ is faithful to a DAG $\mathbb{G}$, then besides (1)∼(5) we also have (6) *composition*:

$$\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y}|\boldsymbol{Z} \;\&\; \boldsymbol{X} \perp\!\!\!\perp \boldsymbol{W}|\boldsymbol{Z} \Rightarrow \boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y} \cup \boldsymbol{W}|\boldsymbol{Z} \tag{1}$$

As we know, faithfulness implies composition, but not vice versa. For composition, Statnikov et al. (2013) provided a relaxed version called *local composition*: we say $\boldsymbol{T} \subseteq \boldsymbol{V}$ satisfies the local composition property, if (1) holds for any $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z} \subseteq \boldsymbol{V}\backslash\boldsymbol{T}$. We also say $\boldsymbol{T} \subseteq \boldsymbol{V}$ satisfies the local composition property with respect to some particular $\boldsymbol{Z} \subseteq \boldsymbol{V}\backslash\boldsymbol{T}$, if (1) holds for any $\boldsymbol{X}, \boldsymbol{Y} \subseteq \boldsymbol{V}\backslash\boldsymbol{Z}\backslash\boldsymbol{T}$.

Conditional mutual information (CMI) is one of the basic tools for testing CIs. Denote the CMI between $\boldsymbol{X}$ and $\boldsymbol{Y}$ conditioned on $\boldsymbol{Z}$ by $\mathbb{I}(\boldsymbol{X}; \boldsymbol{Y}|\boldsymbol{Z})$. Then $\mathbb{I}(\boldsymbol{X}; \boldsymbol{Y}|\boldsymbol{Z}) \geqslant 0$, with equality holding if and only if $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y}|\boldsymbol{Z}$ (Zhang and Guo 2006). For a practical problem, we cannot access to the true CMI; instead, we use its empirical estimate, denoted by $\mathbb{I}_{\boldsymbol{D}}(\boldsymbol{X}; \boldsymbol{Y}|\boldsymbol{Z})$, based on the data $\boldsymbol{D}$ (Cheng et al. 2002). Note that $\mathbb{I}_{\boldsymbol{D}}(\boldsymbol{X}; \boldsymbol{Y}|\boldsymbol{Z}) \geqslant 0$ also holds for any $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z} \subseteq \boldsymbol{V}$. Denote the G^2 statistic by $G^2(\boldsymbol{X}; \boldsymbol{Y}|\boldsymbol{Z}) \triangleq 2n \cdot \mathbb{I}_{\boldsymbol{D}}(\boldsymbol{X}; \boldsymbol{Y}|\boldsymbol{Z})$, which approximates to the chi-square variate with $r \triangleq (r_{\boldsymbol{X}}-1)(r_{\boldsymbol{Y}}-1)r_{\boldsymbol{Z}}$ degrees of freedom, namely $G^2(\boldsymbol{X}; \boldsymbol{Y}|\boldsymbol{Z}) \dot{\sim} \chi^2(r)$, where $r_{\boldsymbol{\xi}}$ represents the number of configurations for $\boldsymbol{\xi}$ (de Campos 2006). Denote the p value by $p_{\boldsymbol{D}}(\boldsymbol{X}; \boldsymbol{Y}|\boldsymbol{Z}) = \mathbb{P}\{\chi^2(r) \geqslant G^2(\boldsymbol{X}; \boldsymbol{Y}|\boldsymbol{Z})\}$. Then, the G^2 test asserts $\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y}|\boldsymbol{Z}$ if $p_{\boldsymbol{D}}(\boldsymbol{X}; \boldsymbol{Y}|\boldsymbol{Z}) > \alpha$ for a significance level α, and concludes $\boldsymbol{X} \not\perp\!\!\!\perp \boldsymbol{Y}|\boldsymbol{Z}$ if $p_{\boldsymbol{D}}(\boldsymbol{X}; \boldsymbol{Y}|\boldsymbol{Z}) \leqslant \alpha$. In this paper, α is set to be 0.05. Accordingly, the *negative p value* is used as the association function, $f_{\boldsymbol{D}}$, as Tsamardinos et al. (2006), Aliferis et al. (2010a, b), and Statnikov et al. (2013) did: $f_{\boldsymbol{D}}(\boldsymbol{X}; \boldsymbol{Y}|\boldsymbol{Z}) \triangleq -\mathbb{P}\{\chi^2(r) \geqslant G^2(\boldsymbol{X}; \boldsymbol{Y}|\boldsymbol{Z})\}$.

The chain rule for CMI (Cover and Thomas 2006) is useful to prove the main results of this paper: $\mathbb{I}(\boldsymbol{X}; \boldsymbol{Y}_1 \cup \boldsymbol{Y}_2|\boldsymbol{Z}) = \mathbb{I}(\boldsymbol{X}; \boldsymbol{Y}_1|\boldsymbol{Z}) + \mathbb{I}(\boldsymbol{X}; \boldsymbol{Y}_2|\boldsymbol{Z} \cup \boldsymbol{Y}_1)$ holds for any four sets of variables $\boldsymbol{X}, \boldsymbol{Y}_1, \boldsymbol{Y}_2$, and $\boldsymbol{Z}$ from $\boldsymbol{V}$. This formula remains valid if we replace $\mathbb{I}(\cdot)$ with $\mathbb{I}_{\boldsymbol{D}}(\cdot)$.

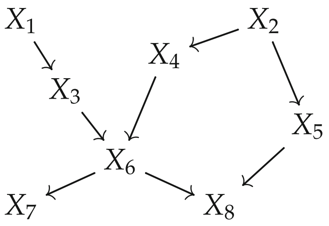

Fig. 2 The DAG of the ASIA network used to illustrate the notions of d-separation and MB

Another notion closely related to CI is d-separation (Pearl 1988; Neapolitan 2004). For a DAG $\mathbb{G}$ over $\boldsymbol{V}$, letting $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z} \subseteq \boldsymbol{V}$ be disjoint, we say $\boldsymbol{Z}$ d-separates $\boldsymbol{X}$ and $\boldsymbol{Y}$ if it blocks every path between $\boldsymbol{X}$ and $\boldsymbol{Y}$, and if this is the case we write $\boldsymbol{X} \perp \boldsymbol{Y}|\boldsymbol{Z}$. Here, $\boldsymbol{Z}$ blocking a path $\mathbb{p}$ means that $\mathbb{p}$ has a head-to-tail node or a tail-to-tail node belonging to $\boldsymbol{Z}$, or that $\mathbb{p}$ has a head-to-head node C such that C and its all descendants are not in $\boldsymbol{Z}$. As well known, $\boldsymbol{X} \perp \boldsymbol{Y}|\boldsymbol{Z} \Rightarrow \boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y}|\boldsymbol{Z}$, if $(\mathbb{G}, \mathbb{P})$ is a BN (Neapolitan 2004). This implication provides a convenient way of identifying CIs. For example, consider a BN with the graph presented in Fig. 2 as its DAG. Then, X_2 and X_8 are d-separated by $\{X_4, X_5\}$, meaning $X_2 \perp X_8|\{X_4, X_5\}$ and thus $X_2 \perp\!\!\!\perp X_8|\{X_4, X_5\}$; X_3 and X_4 are d-separated by $\varnothing$, meaning $X_3 \perp X_4$, so $X_3 \perp\!\!\!\perp X_4$. Note that these two probabilistic CIs can not be directly derived from the Markov condition.

In what follows, the concepts of Mb and MB are presented (Pearl 1988; Neapolitan 2004).

Definition 1 For $T \in \boldsymbol{V}$, we call $\boldsymbol{M} \subseteq \boldsymbol{V}\backslash\{T\}$ a Markov blanket (Mb) of T if $T \perp\!\!\!\perp \boldsymbol{V}\backslash\boldsymbol{M}\backslash\{T\}|\boldsymbol{M}$. Further, a Markov boundary (MB) of $\boldsymbol{T}$ is any Mb such that none of its proper subsets is an Mb of T.

According to Definition 1, an Mb, saying $\boldsymbol{M}$, of T is a set of variables which can shield T from all other variables, while an MB is a minimal Mb. Moreover, by means of the chain rule for CMI, it can be easily shown that $\mathbb{I}(T; \boldsymbol{M}) = \max_{\boldsymbol{N} \subseteq \boldsymbol{V}\backslash\{T\}} \mathbb{I}(T; \boldsymbol{N}) = \mathbb{I}(T; \boldsymbol{V}\backslash\{T\})$, so $\boldsymbol{M}$ carries all information about T carried by all the variables. Furthermore, the following results are well known in the literature (Pearl 1988; Neapolitan 2004; Statnikov et al. 2013): (a) if $(\mathbb{G}, \mathbb{P})$ is a BN, then for $T \in \boldsymbol{V}$ the set of its all parents, children, and spouses is an Mb of T (we denote it by $\boldsymbol{M}_T$); (b) if $\mathbb{P}$ satisfies the intersection property, then T has a unique MB; (c) if $(\mathbb{G}, \mathbb{P})$ satisfies the faithfulness condition, then $\boldsymbol{M}_T$ is the unique MB of T.

Consider again the BN with the graph presented in Fig. 2 as its DAG. In this BN, it is seen that $\boldsymbol{M}_{X_4} \triangleq \{X_2, X_6, X_3\}$ is an Mb of X_4; further, $\boldsymbol{M}_{X_4}$ is the unique MB of X_4 if the faithfulness condition is satisfied. Similarly, $\boldsymbol{M}_{X_2} \triangleq \{X_4, X_5\}$ is the unique MB of X_2 under the faithfulness condition.

Based on the notion of MB, we give the definition for swamping and masking:

Definition 2 (*Swamping and masking*) For $T \in \boldsymbol{V}$, let $\boldsymbol{M} \subseteq \boldsymbol{V}\backslash\{T\}$ be a true MB of T, $\boldsymbol{M}_{\mathbb{A}} \triangleq (\boldsymbol{M}\backslash\boldsymbol{X}) \cup \boldsymbol{Y}$ be the output of an MB discovery algorithm, $\mathbb{A}$, with $\boldsymbol{X} \subseteq \boldsymbol{M}$ and $\boldsymbol{Y} \subseteq \boldsymbol{V}\backslash\boldsymbol{M}\backslash\{T\}$. Assume $\boldsymbol{M}_{\mathbb{A}}$ is not an MB of T. Then, we say (1) *swamping* occurs with respect to $\boldsymbol{M}$, if $\boldsymbol{X} \neq \varnothing$; and (2) *masking* occurs with respect to $\boldsymbol{M}$, if $\boldsymbol{Y} \neq \varnothing$.

The MB of a target may not be unique. This is why we use "a" or "an" in Definition 2. This definition is applicable whether the MB is unique or not. Lemeire (2007) provided a case of violating the uniqueness of MB called *information equivalence*. $\boldsymbol{X}$ and $\boldsymbol{Y}$ are called information equivalent with respect to T given $\boldsymbol{Z} \subseteq \boldsymbol{V}\backslash\boldsymbol{X}\backslash\boldsymbol{Y}\backslash\{T\}$ if the following four conditions hold: $T \not\perp\!\!\!\perp \boldsymbol{X}|\boldsymbol{Z}$, $T \not\perp\!\!\!\perp \boldsymbol{Y}|\boldsymbol{Z}$, $T \perp\!\!\!\perp \boldsymbol{X}|\boldsymbol{Y} \cup \boldsymbol{Z}$, and $T \perp\!\!\!\perp \boldsymbol{Y}|\boldsymbol{X} \cup \boldsymbol{Z}$.

3 Two typical algorithms and a further discussion

In this section, we concisely present two typical MB discovery algorithms: IAMB (Tsamardinos et al. 2003a) and KIAMB (Peña et al. 2007). Then, we make a further discussion about them. Considering that these two algorithms are correct under the local composition assumption (Theorem 1) or the Markov local composition assumption (Definition 4), we put them into an algorithmic framework called LCMB. Here, “LC” means “Markov local composition”.

IAMB is an enhanced variant of GS. Tsamardinos et al. (2003a) showed the correctness of IAMB under the faithfulness condition; Peña et al. (2007) relaxed the condition to the composition assumption; Statnikov et al. (2013) further relaxed the condition to the local composition assumption. The pseudo code for IAMB is described in Algorithm 1. In the

Algorithm 1: LCMB and its three instantiations: IAMB, KIAMB, and LRH

Procedure: $\boldsymbol{M} \leftarrow$ LCMB($\mathbb{A}$; $\boldsymbol{D}, T, \boldsymbol{W}, \boldsymbol{B}$)

Input: $\mathbb{A}$ is a two-phase MB algorithm including the required parameters (for IAMB, $\mathbb{A} = \langle$IAMB$\rangle$; for KIAMB, $\mathbb{A} = \langle$KIAMB, $K\rangle$, in which $K \in [0, 1]$ is a randomization parameter; for LRH, $\mathbb{A} = \langle$LRH, $k\rangle$, in which k ($\geqslant 1$) is an integer denoting the number of nodes entering the CMb at each iteration); $\boldsymbol{D}$ is a data matrix; T is a target; $\boldsymbol{W}$ is a whitelist; $\boldsymbol{B}$ is a blacklist.

Output: The output, $\boldsymbol{M}$, is an MB of T under the Markov local composition assumption.

```
//main procedure:
    $M \leftarrow$ LCMB($\mathbb{A}$; $D, T, W, B$)
$M \leftarrow$ FW($\mathbb{A}$; $D, T, W, B$)
$M \leftarrow$ BW($D, T, M, W$)
return $M$
//IAMB:  $M \leftarrow$ LCMB(⟨IAMB⟩; $D, T, W, B$)
$M \leftarrow$ FW(⟨IAMB⟩; $D, T, W, B$)
$M \leftarrow$ BW($D, T, M, W$)
return $M$
//KIAMB:
    $M \leftarrow$ LCMB(⟨KIAMB, $K$⟩; $D, T, W, B$)
$M \leftarrow$ FW(⟨KIAMB, $K$⟩; $D, T, W, B$)
$M \leftarrow$ BW($D, T, M, W$)
return $M$
//LRH:  $M \leftarrow$ LCMB(⟨LRH, $k$⟩; $D, T, W, B$)
$M \leftarrow$ FW(⟨LRH, $k$⟩; $D, T, W, B$)
$M \leftarrow$ BW($D, T, M, W$)
return $M$
//growing phase of IAMB:
    $M \leftarrow$ FW(⟨IAMB⟩; $D, T, W, B$)
$M \leftarrow W$
while $M$ has changed do
    $Y \leftarrow \arg\max_{X \in V \backslash M \backslash B \backslash \{T\}} f_D(T; X | M)$
    if $T \not\perp\!\!\!\perp Y | M$ then
        $M \leftarrow M \cup \{Y\}$
    end
end
//growing phase of KIAMB:
    $M \leftarrow$ FW(⟨KIAMB, $K$⟩; $D, T, W, B$)
$M \leftarrow W$
while $M$ has changed do
    $M_1 \leftarrow \{X \in V \backslash M \backslash B \backslash \{T\} : T \not\perp\!\!\!\perp X | M\}$
    if $M_1 \neq \varnothing$ then
        $M_2 \leftarrow K^*$ nodes randomly from $M_1$
        $Y \leftarrow \arg\max_{X \in M_2} f_D(T; X | M)$
        $M \leftarrow M \cup \{Y\}$
    end
end
//growing phase of LRH:
    $M \leftarrow$ FW(⟨LRH, $k$⟩; $D, T, W, B$)
$M \leftarrow W$
while $M$ has changed do
    $M_1 \leftarrow \{X \in V \backslash M \backslash B \backslash \{T\} : T \not\perp\!\!\!\perp X | M\}$
    if $M_1 \neq \varnothing$ then
        $M_2 \leftarrow$ refined $M_1$ according to
        exclusion of SEI
        $Y \leftarrow k^*$ nodes from $M_2$ with highest
        associations
        $M \leftarrow M \cup Y$
    end
end
//shrinking phase: $M \leftarrow$ BW($D, T, M, W$)
foreach $X \in M \backslash W$ do
    if $T \perp\!\!\!\perp X | M \backslash \{X\}$ then
        $M \leftarrow M \backslash \{X\}$
    end
end
```

algorithm, the function $f_{\boldsymbol{D}}$ denotes a heuristic measurement of the association between variables based on the data $\boldsymbol{D}$ (Tsamardinos et al. 2003a; Peña et al. 2007). Two widely used selections for $f_{\boldsymbol{D}}$ are CMI (Cheng et al. 2002; Tsamardinos et al. 2003a) and the negative p value (Tsamardinos et al. 2006; Aliferis et al. 2010a, b; Statnikov et al. 2013). This paper employs the latter. Yaramakala (2004) also suggested an equivalent version of the negative p value.

KIAMB is a stochastic extension of IAMB. It embeds a randomization parameter $K \in [0, 1]$ used to trade off greediness and randomness. If taking $K = 1$, KIAMB reduces to IAMB. Peña et al. (2007) proved the correctness of KIAMB under the composition assumption. By the proof, the local composition assumption is sufficient for this algorithm to be correct. Its pseudo code is also described in Algorithm 1. In the growing phase of KIAMB, $K^* = \max\{1, \lfloor |\boldsymbol{M}_1| \cdot K \rfloor\}$.

It is noted here that Algorithm 1 predefines a whitelist $\boldsymbol{W}$ and a blacklist $\boldsymbol{B}$, which can be determined by virtue of expert knowledge or empirical information. In the original IAMB and KIAMB, both $\boldsymbol{W}$ and $\boldsymbol{B}$ are taken as the empty set by default.

Recall that a CI test for a hypothesis is said to be correct if the corresponding statistical decision is correctly made by using a testing method. Based on this terminology, the correctness of IAMB and KIAMB is presented as follows (Tsamardinos et al. 2003a; Peña et al. 2007; Statnikov et al. 2013).

Theorem 1 (Correctness of IAMB and KIAMB) *Assume T satisfies the local composition assumption, and all CI tests are correct. Then* (i) IAMB *outputs an MB of T;* (ii) KIAMB *outputs an MB of T for any $K \in [0, 1]$.*

By this theorem and the two examples presented in Sect. 1, IAMB and KIAMB may fail to output an MB when some CI tests are incorrect or local composition is violated. In what follows, we give a naive definition for the outputs of these algorithms and then make a further discussion. Note that an MB can be equivalently defined to be any Mb such that $T \not\perp\!\!\!\perp \boldsymbol{N}|\boldsymbol{M}\backslash\boldsymbol{N}$ holds for any nonempty $\boldsymbol{N} \subseteq \boldsymbol{M}$, in view of the contraction property and the decomposition property.

Definition 3 For $T \in \boldsymbol{V}$, we call $\boldsymbol{M} \subseteq \boldsymbol{V}\backslash\{T\}$ a weak Markov blanket (WMb) of T if $T \perp\!\!\!\perp X|\boldsymbol{M}$ for any $X \in \boldsymbol{V}\backslash\boldsymbol{M}\backslash\{T\}$. Further, a weak Markov boundary (WMB) of T is any WMb such that $T \not\perp\!\!\!\perp \boldsymbol{N}|\boldsymbol{M}\backslash\boldsymbol{N}$ holds for any nonempty $\boldsymbol{N} \subseteq \boldsymbol{M}$.

This definition is introduced to characterize the true output of an existing MB discovery algorithm (such as IAMB or KIAMB) in the case that local composition is violated. One did not care about such a definition in the early literature because the faithfulness condition or the composition property (and thus local composition) was usually assumed to be a precondition in an MB algorithm; but this definition becomes necessary if we try to explore what are influencing the efficiency of the existing MB discovery algorithms. "Appendix 1" gives a further explanation about why we define the notion of WMB in this way. Clearly, a WMb is an Mb under local composition, while a WMB is an MB under the same assumption. The following theorem describes the relation between Definition 3 and Algorithm 1.

Theorem 2 *Assume all CI tests are correct. Then* IAMB *or* KIAMB *for any $K \in [0, 1]$ outputs a WMB of T.*

Proof Denoting the output of IAMB or KIAMB at the end of the growing phase by $\boldsymbol{M}$, it is clear that $\boldsymbol{M}$ is a WMb, since $T \perp\!\!\!\perp X|\boldsymbol{M}$ holds for any $X \in \boldsymbol{V}\backslash\boldsymbol{M}\backslash\{T\}$ owing to the exit condition. Let the final output of either algorithm be $\boldsymbol{N} \subseteq \boldsymbol{M}$. Without loss of generality, assume that $\boldsymbol{M} \backslash \boldsymbol{N} = \{X_1, \ldots, X_k\}$ and that $k \geqslant 1$, in which $X_1, \ldots, X_k$ are removed from $\boldsymbol{M}$ in sequence, that is, $T \perp\!\!\!\perp X_i | \boldsymbol{M} \backslash \{X_1, \ldots, X_i\}$ holds for $i = 1, \ldots, k$. By the chain rule for CMI, we have

$$\begin{aligned}\mathbb{I}(T; \boldsymbol{M} \backslash \boldsymbol{N} | \boldsymbol{N}) &= \mathbb{I}(T; \{X_1, \ldots, X_k\} | \boldsymbol{M} \backslash \{X_1, \ldots, X_k\}) \\ &= \textstyle\sum_{i=k}^{1} \mathbb{I}(T; X_i | \boldsymbol{M} \backslash \{X_1, \ldots, X_i\}) = 0,\end{aligned}$$

so $T \perp\!\!\!\perp \boldsymbol{M} \backslash \boldsymbol{N} | \boldsymbol{N}$, which combined with $T \perp\!\!\!\perp X | \boldsymbol{M}$ (or equivalently, $T \perp\!\!\!\perp X | (\boldsymbol{M} \backslash \boldsymbol{N}) \cup \boldsymbol{N}$) and the contraction property implies $T \perp\!\!\!\perp (\boldsymbol{M} \backslash \boldsymbol{N}) \cup \{X\} | \boldsymbol{N}$. By the decomposition property, this further means $T \perp\!\!\!\perp X | \boldsymbol{N}$ holds for any $X \in (\boldsymbol{V} \backslash \boldsymbol{M} \backslash \{T\}) \cup (\boldsymbol{M} \backslash \boldsymbol{N}) = \boldsymbol{V} \backslash \boldsymbol{N} \backslash \{T\}$. Hence, $\boldsymbol{N}$ is WMb.

Finally, we prove $\boldsymbol{N}$ is a WMB. In fact, suppose there is some nonempty $\{Y_1, \ldots, Y_\ell\} \triangleq \boldsymbol{Y} \subseteq \boldsymbol{N}$ such that $T \perp\!\!\!\perp \boldsymbol{Y} | \boldsymbol{N} \backslash \boldsymbol{Y}$. Here, the exit condition of the shrinking phase means $\ell \geqslant 2$. It follows that

$$\begin{aligned}0 &= \mathbb{I}(T; \boldsymbol{Y} | \boldsymbol{N} \backslash \boldsymbol{Y}) = \mathbb{I}(T; \{Y_1, \ldots, Y_\ell\} | \boldsymbol{N} \backslash \{Y_1, \ldots, Y_\ell\}) \\ &= \mathbb{I}(T; \{Y_2, \ldots, Y_\ell\} | \boldsymbol{N} \backslash \{Y_1, \ldots, Y_\ell\}) + \mathbb{I}(T; Y_1 | \boldsymbol{N} \backslash \{Y_1\}) \\ &\geqslant \mathbb{I}(T; Y_1 | \boldsymbol{N} \backslash \{Y_1\}) \geqslant 0.\end{aligned}$$

Therefore, $\mathbb{I}(T; Y_1 | \boldsymbol{N} \backslash \{Y_1\}) = 0$, which contradicts $T \not\perp\!\!\!\perp Y_1 | \boldsymbol{N} \backslash \{Y_1\}$ according to the exit condition of the shrinking phase. This indicates $\boldsymbol{N}$ is a WMB. The proof is completed. □

Based on the notion of WMb, we relax the local composition assumption as follows:

Definition 4 (*Markov local composition*) We say $T \in \boldsymbol{V}$ satisfies the Markov local composition property, if T satisfies the local composition property with respect to any WMb of T or, equivalently, if every WMb of T in $\boldsymbol{V}$ is an Mb.

As seen, IAMB and KIAMB remain correct under the Markov local composition assumption. This is why we call them both LCMB algorithms.

4 LRH algorithm: lessen swamping, resist masking, and highlight the true positives

This section addresses the problem of (P1) posed in Sect. 1. First, we exemplify the situations that some CI tests are incorrect, even when the data size is large. Then, we analyze how to add as few false positives as possible to the CMb and thus to reduce the incorrectness of CI tests such that swamping and masking get alleviated. Finally, we present the resulting algorithm called LRH, which can lessen swamping, resist masking, and highlight the true positives.

4.1 An exemplification

Consider the well-known ALARM network (Beinlich et al. 1989), which is shown in Fig. 3. Observe that there are many situations of multiple channels for the flow of information. In these situations, IAMB may suffer swamping and masking caused by the incorrectness of some associated CI tests. For example, taking $T \triangleq X_2$ with $\boldsymbol{M}_T \triangleq \{X_{23}, X_{27}, X_{29}\}$ as its unique MB under the faithfulness condition, the detailed operating steps of discovering

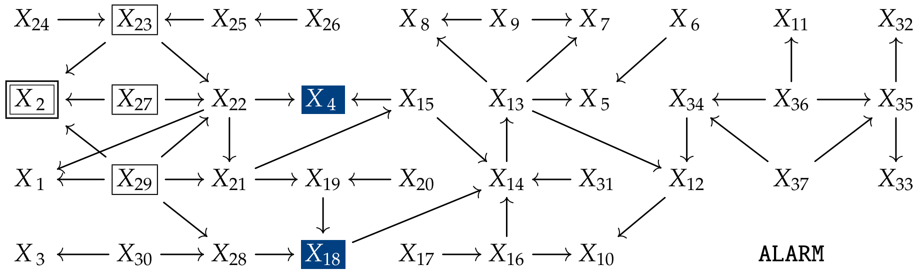

Fig. 3 ALARM network with 37 nodes and 46 edges used to illustrate that the incorrectness of some CI tests may lead to swamping and masking due to the multiple channels for the flow of information. For example, $\{X_{23}, X_{27}, X_{29}\}$ is the unique MB of X_2 under the faithfulness condition. However, `IAMB` outputs an incorrect MB, $\{X_4, X_{18}, X_{23}\}$, while `LRH` outputs the true MB

Table 1 Details of `IAMB` for discovering the MB, $\{X_{23}, X_{27}, X_{29}\}$, of $T \triangleq X_2$ in the ALARM network with $\alpha = 0.05$, based on a data set of size 5000

Phase	Iteration	Results of `IAMB`
Growing	1	$\boldsymbol{M} = \varnothing$ $f_{\boldsymbol{D}}(T; X_1 \mid \boldsymbol{M}) = \max_{X \in \boldsymbol{V} \backslash \boldsymbol{M} \backslash \{T\}} f_{\boldsymbol{D}}(T; X \mid \boldsymbol{M}) \geqslant -\alpha$. Conclusion: $\boldsymbol{M} \leftarrow \boldsymbol{M} \cup \{X_1\}$.
	2	$\boldsymbol{M} = \{X_1\}$ $f_{\boldsymbol{D}}(T; X_4 \mid \boldsymbol{M}) = \max_{X \in \boldsymbol{V} \backslash \boldsymbol{M} \backslash \{T\}} f_{\boldsymbol{D}}(T; X \mid \boldsymbol{M}) \geqslant -\alpha$. Conclusion: $\boldsymbol{M} \leftarrow \boldsymbol{M} \cup \{X_4\}$.
	3	$\boldsymbol{M} = \{X_1, X_4\}$ $f_{\boldsymbol{D}}(T; X_{18} \mid \boldsymbol{M}) = \max_{X \in \boldsymbol{V} \backslash \boldsymbol{M} \backslash \{T\}} f_{\boldsymbol{D}}(T; X \mid \boldsymbol{M}) \geqslant -\alpha$. Conclusion: $\boldsymbol{M} \leftarrow \boldsymbol{M} \cup \{X_{18}\}$.
	4	$\boldsymbol{M} = \{X_1, X_4, X_{18}\}$ $f_{\boldsymbol{D}}(T; X_{23} \mid \boldsymbol{M}) = \max_{X \in \boldsymbol{V} \backslash \boldsymbol{M} \backslash \{T\}} f_{\boldsymbol{D}}(T; X \mid \boldsymbol{M}) \geqslant -\alpha$. Conclusion: $\boldsymbol{M} \leftarrow \boldsymbol{M} \cup \{X_{23}\}$.
	5	$\boldsymbol{M} = \{X_1, X_4, X_{18}, X_{23}\}$ $\max_{X \in \boldsymbol{V} \backslash \boldsymbol{M} \backslash \{T\}} f_{\boldsymbol{D}}(T; X \mid \boldsymbol{M}) \approx -1.0000 < -\alpha$ Conclusion: the growing phase ends, and the obtained Mb is $\{X_1, X_4, X_{18}, X_{23}\}$.
Shrinking	1	$\boldsymbol{M} = \{X_1, X_4, X_{18}, X_{23}\}$ $f_{\boldsymbol{D}}(T; X_1 \mid \boldsymbol{M} \backslash \{X_1\}) \approx -1.0000 < -\alpha$ Conclusion: X_1 is removed from the CMb.
	2	$\boldsymbol{M} = \{X_4, X_{18}, X_{23}\}$ $f_{\boldsymbol{D}}(T; X_i \mid \boldsymbol{M} \backslash \{X_i\}) \approx -0.0000 \geqslant -\alpha$, for $i = 4, 18, 23$. Conclusion: the shrinking phase ends, and the obtained MB is $\{X_4, X_{18}, X_{23}\}$.
Conclusion: `IAMB` outputs $\{X_4, X_{18}, X_{23}\}$ as the MB of T, which is incorrect.		

The association is taken as the negative p value of the G^2-test. In the growing phase, `IAMB` adds the nodes X_1, X_4, X_{18}, and X_{23} in sequence to the CMb; in the shrinking phase, `IAMB` removes X_1 from the CMb. Thus, `IAMB` incorrectly outputs $\{X_4, X_{18}, X_{23}\}$ as the MB of T. The same result is returned when taking α as 0.01 or 0.005 or 0.001

the MB is presented in Table 1. Following the steps in the table, `IAMB` first adds X_1, X_4, X_{18}, and X_{23} to the CMb, and then removes X_1. This algorithm outputs an incorrect MB, $\{X_4, X_{18}, X_{23}\}$, for the target T. As seen, swamping occurs since the two true positives, X_{27}

and X_{29}, become false negatives; masking also follows because the two true negatives, X_4 and X_{18}, become false positives by the end of the shrinking phase.

We now analyze why swamping and masking happen by virtue of Fig. 3. Note that T contains the information propagated by X_{23}, X_{27}, and X_{29}. By means of these three nodes, there are no fewer than two disjoint links without any converging nodes for the flow of information between T and one of X_1, X_4, and X_{18}. Specifically, we have

- $T \leftarrow X_{23}$ (or X_{27}) $\rightarrow X_{22} \rightarrow X_1$ and $T \leftarrow X_{29} \rightarrow X_1$ connect T and X_1;
- $T \leftarrow X_{23}$ (or X_{27}) $\rightarrow X_{22} \rightarrow X_4$ and $T \leftarrow X_{29} \rightarrow X_{21} \rightarrow X_{15} \rightarrow X_4$ connect T and X_4;
- $T \leftarrow X_{29} \rightarrow X_{28} \rightarrow X_{18}$ and $T \leftarrow X_{23}$ (or X_{27}) $\rightarrow X_{22} \rightarrow X_{21} \rightarrow X_{19} \rightarrow X_{18}$ connect T and X_{18}.

This means X_1 or X_4 or X_{18} has higher association with T than each of X_{23}, X_{27}, and X_{29}, so X_1, X_4, and X_{18} enter the CMb in sequence in the growing phase. After adding X_{23}, the remaining two true positives (i.e., X_{27} and X_{29}) are excluded, due to the incorrectness of the following two CI tests:

- The true CMI, $\mathbb{I}(T; X_{27}|X_1, X_4, X_{18}, X_{23}) \approx 0.0331 > 0$, indicating $T \not\perp\!\!\!\perp X_{27}|\{X_1, X_4, X_{18}, X_{23}\}$, but the p value of the G^2-test, $p_{\boldsymbol{D}}(T; X_{27}|X_1, X_4, X_{18}, X_{23}) \approx 1.0000$, is far larger than α, meaning the opposite assertion $T \perp\!\!\!\perp X_{27}|\{X_1, X_4, X_{18}, X_{23}\}$;
- The true CMI, $\mathbb{I}(T; X_{29}|X_1, X_4, X_{18}, X_{23}) \approx 0.0352 > 0$, indicating $T \not\perp\!\!\!\perp X_{29}|\{X_1, X_4, X_{18}, X_{23}\}$. On the other hand, $p_{\boldsymbol{D}}(T; X_{29}|X_1, X_4, X_{18}, X_{23}) \approx 1.0000 \gg \alpha$ asserts $T \perp\!\!\!\perp X_{29}|\{X_1, X_4, X_{18}, X_{23}\}$.

This explains why the incorrectness of some CI tests may lead to swamping. Further, in the shrinking phase, the two false positives, X_4 and X_{18}, can not be identified, because not all information about T is shielded since X_{27} and X_{29} are excluded. This means masking may follow if swamping occurs.

This analysis shows the incorrectness of CI tests may bring swamping and masking. However, we need to use "all CI tests are correct" as a precondition for an MB algorithm. Hence, what we can do is to reduce the incorrectness of CI tests as far as possible. Considering an incorrect CI test is usually the case of accepting a false hypothesis (Cochran 1954; Bromberg and Margaritis 2009), a good MB algorithm should add as few false positives as possible to the CMb in the growing phase, because too many false positives may make the detection of a true dependence hard.

4.2 Method

Example 1 presents a simplified scenario where swamping and masking happen due to the incorrectness of CI tests. By the graphical structure that (a) of Fig. 1 illustrates, the target T propagates its information to X, Y_1, and Y_2. Then, Y_1 and Y_2 transmit the information to Z. In other words, Z collects the information about T through Y_1 and Y_2, so it may carry more information about T than either Y_1 or Y_2. Mathematically, $\mathbb{I}(T; Z) \geqslant \max\{\mathbb{I}(T; Y_1), \mathbb{I}(T; Y_2)\}$ may hold. This indicates Z has spuriously high association with T. For a larger BN such as the ALARM network, there may be many similar nodes to Z. Hence, we can add as few false positives as possible to the CMb by identifying such nodes.

Suppose the transmission via Y_2 is blocked as (b) of Fig. 1 shows. That is, $T \rightarrow Y_1 \rightarrow Z$ becomes the only remaining channel between T and Z. In this case, the data-processing inequality (Cover and Thomas 2006) gives $\mathbb{I}(T; Z|Y_2) \leqslant \mathbb{I}(T; Y_1|Y_2)$. Similarly, if the transmission via Y_1 is blocked as shown in (c) of Fig. 1, then $\mathbb{I}(T; Z|Y_1) \leqslant \mathbb{I}(T; Y_2|Y_1)$. This means Z can no longer effectively collect the information about T once one or more channels between T and Z are blocked, so Y_1 or Y_2 will enter the CMb before Z. Without loss of generality, suppose the CMb is obtained as $\boldsymbol{M} \triangleq \{X, Y_1\}$ after two steps of the growing phase. Then, further blocking implies $T \not\perp\!\!\!\perp Y_2|\boldsymbol{M} \cup \{Z\}$ and $T \perp\!\!\!\perp Z|\boldsymbol{M} \cup \{Y_2\}$. Hence, Y_2 enters $\boldsymbol{M}$ and thus $\boldsymbol{M} = \{X, Y_1, Y_2\}$. Finally, $T \perp\!\!\!\perp Z|\boldsymbol{M}$, meaning the growing phase ends.

As seen, the method of blocking one or more information channels can add as few false positives as possible to the CMb in the growing phase, because the remaining information (after blocking information channels) about T carried by one node is closer to the true unique information about T carried by this node. Therefore, this method can reduce swamping and masking caused by the problem of (P1).

Motivated by this idea, denoting the CMb by $\boldsymbol{M}$, we select the subsequent additions according to the following selection-exclusion-inclusion (SEI) procedure:

(a) *Selection* Let $\boldsymbol{M}_1 \triangleq \{X \in \boldsymbol{V}\backslash\boldsymbol{M}\backslash\boldsymbol{B}\backslash\{T\} : T \not\perp\!\!\!\perp X|\boldsymbol{M}\}$ be the set of all nodes having information channels reaching T other than those through $\boldsymbol{M}$. The nodes in $\boldsymbol{M}_1$ are the candidates preparing to enter the CMb in the current step.

(b) *Exclusion* If $\boldsymbol{M}_1$ is empty, the shrinking phase ends; if $|\boldsymbol{M}_1| = 1$, add the only node in $\boldsymbol{M}_1$ to $\boldsymbol{M}$ and then go to (a) of the next iteration; otherwise, the method of blocking information channels is used. Put $\boldsymbol{M}_2 \triangleq \{X \in \boldsymbol{M}_1 : T \not\perp\!\!\!\perp X|\boldsymbol{M} \cup \{Z\}$ holds for any $Z \in \boldsymbol{N}_X\}$ and $\boldsymbol{M}_3 \triangleq \boldsymbol{M}_1\backslash\boldsymbol{M}_2$, in which $\boldsymbol{N}_X \triangleq \{Y \in \boldsymbol{M}_1\backslash\{X\} : X \not\perp\!\!\!\perp Y|\boldsymbol{M}\}$ denotes the set of all nodes having information channels reaching T and X other than those through $\boldsymbol{M}$. This heuristic is inspired by the notion of 1-step dependence coefficient (de Campos 2006; Martínez-Rodríguez et al. 2008; Lee et al. 2012). If $\boldsymbol{M}_2 = \varnothing$, modify it as $\boldsymbol{M}_2 \triangleq \{Y\}$ with $Y = \arg\max_{X \in \boldsymbol{M}_1} f_{\boldsymbol{D}}(T; X|\boldsymbol{M})$. All nodes in $\boldsymbol{M}_3$ (with spuriously high dependence on T) are excluded. This step can effectively reduce the possibility of adding too many false positives to the CMb. A further discussion about the exclusion procedure is given in Sect. 7.

(c) *Inclusion* Let $\boldsymbol{Y}$ be a set of $k^* \triangleq \min\{k, |\boldsymbol{M}_2|\}$ nodes from $\boldsymbol{M}_2$ with the highest associations with T: take

$$g_{\boldsymbol{D}}(T; X|\boldsymbol{M}, \boldsymbol{N}_X) = \min_{Z \in \boldsymbol{N}_X} f_{\boldsymbol{D}}(T; X|\boldsymbol{M} \cup \{Z\}) \tag{2}$$

and let $\boldsymbol{Y} = \{X_{(1)}, \ldots, X_{(k^*)}\}$, with $g_{\boldsymbol{D}}(T; X_{(1)}|\boldsymbol{M}, \boldsymbol{N}_{X_{(1)}}) \geqslant \cdots \geqslant g_{\boldsymbol{D}}(T; X_{(|\boldsymbol{M}_2|)} |\boldsymbol{M}, \boldsymbol{N}_{X_{(|\boldsymbol{M}_2|)}})$. Add the nodes in $\boldsymbol{Y}$ to $\boldsymbol{M}$. Here, k ($\geqslant 1$) is the maximal number of nodes entering the CMb at each iteration. This paper uses $k = 3$.

Repeat (a)(b)(c) until the exit condition stated in (b) is satisfied (i.e., $\boldsymbol{M}_1$ is empty). After that, refine $\boldsymbol{M}$ by virtue of the shrinking phase.

This is the basic method of designing the new algorithm, LRH (presented in the next subsection). It will be seen that the algorithm performs well in lessening swamping, resisting masking, and highlighting the true positives. This is why we call it the LRH algorithm.

4.3 LRH algorithm with application to the ALARM network

By the description given in Sect. 4.2, we present the LRH algorithm in Algorithm 1. LRH consists of two phases: in the growing phase, the SEI procedure is iteratively implemented to search an Mb which contains as few false positives as possible; in the shrinking phase, the Mb is refined to become an MB. Specifically, the selection, exclusion, and inclusion procedures of SEI are implemented in Line 3, Line 5, and Line 7, respectively, in the growing phase of LRH. As the following theorem shows, LRH is correct under the local composition assumption or the Markov local composition assumption. Hence, LRH is also an LCMB algorithm. The proof of this theorem is similar to that of Theorem 2, so we omit it here.

Theorem 3 (Correctness of LRH) *Assume all CI tests are correct. Then* LRH *outputs a WMb of T for any $k \geqslant 1$. Further, if T satisfies the (Markov) local composition assumption, then* LRH *outputs an MB of T.*

Now we consider the computational complexities of the three LCMB algorithms. Usually, the number of CI tests can be employed to measure the complexity of a CI-based MB discovery algorithm (Tsamardinos et al. 2003a, 2006; Aliferis et al. 2010a). In this sense, IAMB and KIAMB have the same complexity $O(|\boldsymbol{V}| \cdot |\boldsymbol{M}_T|)$ in the average case. By direct analysis, the complexity of LRH is $O[(|\boldsymbol{V}| + |\boldsymbol{M}_T|^2) \cdot |\boldsymbol{M}_T|/k]$. As we can see, LRH may need more CI tests than IAMB or KIAMB in each iteration; however, there may be fewer iterations in the growing phase of LRH, since multiple nodes are allowed to enter the CMb in each iteration (see, e.g., Tables 1 and 2 for an illustration). Also, the shrinking phase may also need fewer iterations because LRH usually add fewer true negatives to the CMb than IAMB or KIAMB by the end of the growing phase. Therefore, LRH is also time efficient like IAMB and KIAMB.

To demonstrate how LRH works, we apply this algorithm to the ALARM network. The detailed operating steps of LRH for discovering the MB of $T \triangleq X_2$ are presented in Table 2. Following the steps in the table, LRH first adds $\{X_{29}, X_{23}, X_{21}\}$ and $\{X_{27}\}$ to the CMb, and then removes the only one false positive, X_{21}. As expected, the two nodes, X_4 and X_{18}, with spuriously high dependence on the target are successfully identified before the *inclusion* procedure of SEI and, therefore, they will no longer swamp the true positives, X_{27} and X_{29}. In comparison, IAMB adds these two (plus another) nonmembers of the true MB, namely, X_1, X_4, and X_{18}; the two true positives, X_{27} and X_{29}, are then swamped. Although X_1 is finally removed, X_4 and X_{18} continue to mask themselves, so IAMB gives an incorrect output.

There are many other similar situations for this network. Table 3 lists the results of the three LCMB algorithms (i.e., IAMB, KIAMB, and LRH) for all the 37 nodes as targets. For KIAMB, we take K as 0.2, 0.5, and 0.8; all results for each KIAMB are averaged over five runs. This table consists of two aspects: MB and relative efficiency (RE), in which the RE of an obtained MB, $\boldsymbol{M}$, is defined as

$$\mathrm{RE}_{\boldsymbol{D}}(\boldsymbol{M}, T) \triangleq \min\{\mathbb{I}_{\boldsymbol{D}}(T;\ \boldsymbol{M})/\mathbb{I}_{\boldsymbol{D}}(T;\ \boldsymbol{M}_T), 1\}. \tag{3}$$

This statistic is a naive estimate for $\mathrm{RE}(\boldsymbol{M}, T) \triangleq \mathbb{I}(T;\ \boldsymbol{M})/\mathbb{I}(T;\ \boldsymbol{M}_T)$, which measures the performance that $\boldsymbol{M}$ carries the information about T. Table 3 indicates that:

- LRH retrieves 21 and 3 Mbs; IAMB retrieves 8 and 5 Mbs; and KIAMB with $K = 0.5$ or $K = 0.8$ performs nearly as well as IAMB, while KIAMB with $K = 0.2$ performs poorly. Further, LRH possesses 34 out of the 37 maximal REs; IAMB possesses 14 maximal REs; and KIAMB with K taken as 0.2, 0.5, and 0.8 possess 5, 12, and 12 maximal REs, respectively. This indicates LRH improves on IAMB and KIAMB greatly. The results also reveal that it is reasonable to use RE to measure the performance that a potential MB carries the information about the target. In addition, it should be mentioned here that each LCMB algorithm outputs several Mbs (supersets of the true MBs) after implementing the BW function (see Algorithm 1 for details). This type of masking is the consequence of the incorrectness of some associated CI tests. The next subsection will discuss this issue and provides an effective post-processing technique, called PostBW, to alleviate this type of masking.
- There are 12 cases where LRH outputs a proper subset of the true MB, and 9 such cases for IAMB. As seen, in any one such case for IAMB, LRH also outputs a proper subset

Table 2 Details of LRH for discovering the MB of $T \triangleq X_2$ on the ALARM network with $k = 3$ and $\alpha = 0.05$, based on a data set of size 5000

Phase	Iteration	Results of LRH	
Growing	1	Selection	$\boldsymbol{M} = \varnothing$
			$\boldsymbol{M}_1 = \{X \in \boldsymbol{V} \backslash \boldsymbol{M} \backslash \{T\} \colon T \not\perp\!\!\!\perp X \vert \boldsymbol{M}\} = \{X_i : i = 1, 4, 7, 8, 10, 12, \ldots, 15, 18, 19, 21, \ldots, 29, 31, 34, 36\}$
		Exclusion	$\boldsymbol{M}_2 = \{X \in \boldsymbol{M}_1 : T \not\perp\!\!\!\perp X \vert \boldsymbol{M} \cup \{Z\}, \ \forall\, Z \in \boldsymbol{N}_X\} = \{X_i :$ $i = 29, 23, 21, 18, 4, 27, 8, 24, 36\}$, in which the nodes are sorted according to $g_{\boldsymbol{D}}$ from high to low. See Eq. (2) for details
			Conclusion: X_i is excluded from $\boldsymbol{M}_1$, for $i = 1, 7, 10, 12, 13, 14, 15, 19, 22, 25, 26, 28, 31, 34$
		Inclusion	$\boldsymbol{Y} =$ "a set of at most k nodes from $\boldsymbol{M}_2$ with the highest associations with $T'' = \{X_{29}, X_{23}, X_{21}\}$
			Conclusion: X_{29}, X_{23}, and X_{21} are included into $\boldsymbol{M}$
	2	Selection	$\boldsymbol{M} = \{X_{29}, X_{23}, X_{21}\}$
			$\boldsymbol{M}_1 = \{X_{27}\}$
		Exclusion	$\boldsymbol{M}_2 = \{X_{27}\}$
			Conclusion: no node is excluded from $\boldsymbol{M}_1$ since $\vert\boldsymbol{M}_1\vert = 1$
		Inclusion	$\boldsymbol{Y} = \{X_{27}\}$
			Conclusion: X_{27} is included into $\boldsymbol{M}$
	3	$\boldsymbol{M} = \{X_{29}, X_{23}, X_{21}, X_{27}\}$	
		$\boldsymbol{M}_1 = \varnothing$, so the search stops	
		Conclusion: the growing phase ends, and the obtained Mb is $\{X_{21}, X_{23}, X_{27}, X_{29}\}$	
Shrinking	1	$\boldsymbol{M} = \{X_{21}, X_{23}, X_{27}, X_{29}\}$	
		$f_{\boldsymbol{D}}(T; X_{21} \mid \boldsymbol{M} \backslash \{X_{21}\}) \ = \ -1.0000 \ < \ -\alpha;$	
		$f_{\boldsymbol{D}}(T; X_{23} \mid \boldsymbol{M} \backslash \{X_{23}\}) \ = \ -0.0000 \ \geqslant \ -\alpha$	
		$f_{\boldsymbol{D}}(T; X_{27} \mid \boldsymbol{M} \backslash \{X_{27}\}) \ = \ -0.0000 \ \geqslant \ -\alpha;$	
		$f_{\boldsymbol{D}}(T; X_{29} \mid \boldsymbol{M} \backslash \{X_{29}\}) \ = \ -0.0000 \ \geqslant \ -\alpha$	
		Conclusion: X_{21} is removed from the CMb	
	2	$\boldsymbol{M} = \{X_{23}, X_{27}, X_{29}\}$	

Table 2 continued

Phase	Iteration	Results of LRH
		$f_{\boldsymbol{D}}(T; X_{23} \mid \boldsymbol{M}\backslash\{X_{23}\}) = -0.0000 \geqslant -\alpha$; $f_{\boldsymbol{D}}(T; X_{27} \mid \boldsymbol{M}\backslash\{X_{27}\}) = -0.0000 \geqslant -\alpha$; $f_{\boldsymbol{D}}(T; X_{29} \mid \boldsymbol{M}\backslash\{X_{29}\}) = -0.0000 \geqslant -\alpha$ Conclusion: the shrinking phase ends, and the obtained MB is $\{X_{23}, X_{27}, X_{29}\}$
Conclusion: LRH outputs $\{X_{23}, X_{27}, X_{29}\}$ as the MB of T, which is correct		

The association is taken as the negative p value of the G^2-test. In the growing phase, LRH adds the nodes $\{X_{29}, X_{23}, X_{21}\}$ and $\{X_{27}\}$ in sequence to the CMb; in the shrinking phase, LRH removes X_{21} from the CMb. Thus, LRH retrieves the true MB, $\{X_{23}, X_{27}, X_{29}\}$, of T. The same result is returned when taking k as 1 or 2 or 4 or 5 and taking α as 0.01 or 0.005 or 0.001

Table 3 Results of the `LCMB` algorithms applied to the ALARM network based on a data set of size 5000 ($\alpha = 0.05$)

Target	MB (subscripts)			RE (Equation 3)				
	True	`IAMB`	`LRH`	`IAMB`	`KIAMB`			`LRH`
					$K = 0.2$	$K = 0.5$	$K = 0.8$	
X_1	22, 29	22	22, 29	0.9896	0.9307	0.9364	0.9047	1.0000
X_2	23, 27, 29	4, 18, 23	23, 27, 29	0.7672	0.7910	0.8396	0.7625	1.0000
X_3	30	4, 7, 30	4, 7, 30	1.0000	1.0000	1.0000	1.0000	1.0000
X_4	15, 22	1, 15	15, 22	0.9691	0.9351	0.9514	0.9323	1.0000
X_5	6, 13	6, 13	6, 13	1.0000	0.9770	1.0000	0.9865	1.0000
X_6	5, 13	5, 13, 30	5, 13	1.0000	0.9675	1.0000	1.0000	1.0000
X_7	9, 13	9, 13	9, 13	1.0000	0.8421	0.9633	0.9920	1.0000
X_8	9, 13	5, 9, 13	9, 13	1.0000	0.9529	0.9700	0.9892	1.0000
X_9	7, 8, 13	7, 8, 13	7, 8, 13	1.0000	0.9542	0.9889	0.9973	1.0000
X_{10}	12, 16	12, 16	12, 16	1.0000	0.9531	0.9095	0.9794	1.0000
X_{11}	36	32, 34, 36	34, 36	1.0000	1.0000	1.0000	1.0000	1.0000
X_{12}	10, 13, 16, 34	10, 13, 34	10, 13, 16, 34	0.9083	0.8549	0.9107	0.9399	1.0000
X_{13}	5, 6, 7, 8, 9, 12, 14, 34	5, 7, 8, 9, 14	7, 8, 12, 14, 34	0.8591	0.7617	0.8153	0.8478	0.9587
X_{14}	13, 15, 16, 18, 31	7, 13	13, 15, 16, 19, 28, 31	0.5757	0.7314	0.8330	0.8991	0.9657
X_{15}	4, 14, 16, 18, 21, 22, 31	1, 4, 19	4, 21	0.9098	0.9135	0.8903	0.8872	0.9305
X_{16}	10, 12, 14, 15, 17, 18, 31	7, 10, 12	14, 15, 17, 18, 31	0.3922	0.6712	0.8334	0.7685	0.9125
X_{17}	16	16	16	1.0000	1.0000	1.0000	1.0000	1.0000
X_{18}	14, 15, 16, 19, 28, 31	19, 28	19, 28	0.9302	0.8794	0.9354	0.9103	0.9302
X_{19}	18, 20, 21, 28	4, 18, 28	18, 20, 21, 28	0.9533	0.9622	0.9474	0.9334	1.0000
X_{20}	19, 21	19, 21	19, 21	1.0000	0.9282	0.9592	1.0000	1.0000
X_{21}	15, 19, 20, 22, 29	2, 4, 18, 28	15, 19, 22, 29	0.8729	0.9030	0.9146	0.9178	0.9901
X_{22}	1, 4, 15, 21, 23, 27, 29	1, 4	4, 21, 29	0.8980	0.8850	0.9183	0.9232	0.9023
X_{23}	2, 22, 24, 25, 27, 29	1, 2, 25	2, 22, 25	0.8844	0.8515	0.8086	0.8595	0.9281
X_{24}	23, 25	2, 23	23, 25	0.9705	0.9535	0.9467	0.9725	1.0000
X_{25}	23, 24, 26	23, 26	23, 24, 26	0.9636	0.9218	0.9855	1.0000	1.0000
X_{26}	25	25	25	1.0000	0.8799	1.0000	1.0000	1.0000
X_{27}	2, 22, 23, 29	2, 23	2, 23	0.7431	0.7103	0.6994	0.7431	0.7431
X_{28}	18, 19, 29, 30	2, 18, 19	18, 19, 29	0.9882	0.9314	0.9908	0.9714	0.9968
X_{29}	1, 2, 21, 22, 23, 27, 28, 30	2, 18, 19	1, 21, 28	0.7419	0.8723	0.9023	0.8307	0.9252
X_{30}	3, 28, 29	3, 28	3, 28, 29	0.8988	0.9372	0.9632	0.9061	1.0000
X_{31}	14, 15, 16, 18	4, 14, 16, 18	14, 15, 16, 18	0.9345	0.9613	0.9239	0.8713	1.0000
X_{32}	35	12, 35	34, 35	1.0000	0.9889	1.0000	1.0000	1.0000
X_{33}	35	35	35	1.0000	1.0000	1.0000	1.0000	1.0000
X_{34}	12, 13, 36, 37	11, 12, 13, 37	12, 13, 36, 37	0.8645	0.7532	0.9761	0.9756	1.0000
X_{35}	32, 33, 36, 37	32, 33	32, 33, 37	0.9823	0.9910	0.9889	0.9974	0.9935
X_{36}	11, 34, 35, 37	11, 32, 34, 37	34, 35, 37	0.9717	0.9823	0.9713	0.9813	0.9936
X_{37}	34, 35, 36	11, 32, 34	34, 35, 36	0.8574	0.9696	1.0000	0.9901	1.0000

The backcolors are specified as follows: "■" denotes here is the largest RE; "🟦" denotes the true MB is found; "🟩" denotes a proper subset of the true MB is found; "🟧" denotes an Mb (instead of an MB) is found; others have no backcolor

of the true MB (5 cases) or the true MB (4 cases), but not vice versa. Therefore, `LRH` performs better in lessening swamping than `IAMB`. Taking X_{12} as the target for example, `IAMB` adds X_5, X_7, X_8, X_9, X_{10}, X_{13}, and X_{34} to the CMb in its growing phase, and then removes X_5, X_7, X_8, and X_9 in its shrinking phase to output $\{X_{10}, X_{13}, X_{34}\}$ as the MB of X_{12}; `LRH` adds X_{10}, X_{13}, X_{16}, and X_{34} to the CMb in the growing phase, and no nodes are removed in the shrinking phase. As seen, X_{16} is a spouse node of X_{12}

Table 4 Average REs and RTs of the LCMB algorithms applied to the ALARM network based on 10 data sets of different sizes (from 500 to 5000): each result is averaged over all the 37 nodes as targets

LCMB		Data size									
		500	1000	1500	2000	2500	3000	3500	4000	4500	5000
Average REs											
IAMB		0.9047	0.9144	0.9190	0.9074	0.9070	0.9167	0.9119	0.9192	0.9186	0.9142
KIAMB	$K = 0.2$	0.8192	0.8438	0.8403	0.8858	0.9123	0.8763	0.9050	0.9109	0.8870	0.9111
	$K = 0.5$	0.8687	0.8682	0.9346	0.9133	0.9225	0.9154	0.9465	0.9269	0.9174	0.9163
	$K = 0.8$	0.9048	0.9224	0.9229	0.9290	0.9439	0.9225	0.9147	0.9371	0.9334	0.9354
LRH		0.9250	0.9415	0.9412	0.9621	0.9636	0.9628	0.9693	0.9702	0.9710	0.9776
Average RTs (in seconds)											
IAMB		0.0383	0.0482	0.0603	0.0667	0.0797	0.0953	0.1022	0.1149	0.1265	0.1376
KIAMB	$K = 0.2$	0.0438	0.0566	0.0703	0.0779	0.0907	0.1015	0.1181	0.1283	0.1513	0.1541
	$K = 0.5$	0.0411	0.0532	0.0618	0.0719	0.0826	0.0964	0.1018	0.1133	0.1231	0.1423
	$K = 0.8$	0.0389	0.0491	0.0608	0.0675	0.0803	0.0965	0.1071	0.1135	0.1201	0.1335
LRH		0.0490	0.0581	0.0746	0.0739	0.0819	0.1051	0.1211	0.1268	0.1583	0.1617

The backcolor indicates the performance of each algorithm with black corresponding to the maximal RE and blue to the minimal (for each case of data size)

but IAMB fails to include it. This is because IAMB adds too many false positives (i.e., X_5, X_7, X_8, and X_9) in its growing phase such that the CI test for the true dependence $X_{12} \not\perp\!\!\!\perp X_{16} | \{X_5, X_7, X_8, X_9 X_{10}, X_{13}, X_{34}\}$ incorrectly accepts the false hypothesis. LRH outputs the true MB of X_{12}. Similarly, if taking X_{25} as the target, IAMB also fails to include the spouse node X_{24}, while LRH can output the true MB, $\{X_{23}, X_{24}, X_{26}\}$.

- For IAMB, there are 15 cases in which the outputs excludes one or more true positives and includes some false positives. This means IAMB suffers swamping and masking severely. In comparison, LRH yields only one such output; it successfully prevents those spuriously dependent variables from entering the CMb in most situations by virtue of the SEI procedure. Therefore, the heuristic involved in SEI is effective in lessening swamping, resisting masking, and highlighting the true positives.

By the above results, although LRH needs the same conditions for its correctness as IAMB and KIAMB, this algorithm is expected to be of higher performance than the other two LCMB algorithms for most situations in collecting the information about the target. To check whether this assertion holds for various data sizes, we implement the three LCMB algorithms on the ALARM network based on 10 data sets of sizes from 500 to 5000. The first part of Table 4 presents the corresponding REs, in which each result is averaged over all the 37 nodes as targets; the results for KIAMB are averaged over five different runs. By the table, LRH performs much better than IAMB and KIAMB in all cases; it can collect over 96 % of the information about the targets when $n = 2000$. This shows the data efficiency of the LRH algorithm.

Additionally, the average running time (RT; in seconds) of each LCMB algorithm is listed in the second part of Table 4. By the table, there is no significant difference between the RT of LRH and that of IAMB and KIAMB. Therefore, all the three LCMB algorithms are time efficient.

4.4 `PostBW`: a post-processing technique

As we know, any Mb (*resp.*, WMb) of a target will become an MB (*resp.*, WMB) after being processed by `BW`, if all the CI tests involved are correct. However, Table 3 reveals that some false positives may remain in the Mb if some CI tests are incorrect. This subsection puts forward a post-processing technique, called `PostBW`, to alleviate such a type of masking. This technique is pseudo-coded in Algorithm 2.

Theorem 4 *Let $\boldsymbol{M}$ be any WMb (resp., an Mb) of T. Then, $\boldsymbol{M}$ is a WMB (resp., MB) of T if and only if, for any $X \in \boldsymbol{M}$, there is no $Y \in \boldsymbol{V} \backslash \boldsymbol{M} \backslash \{T\}$ such that*

$$T \perp\!\!\!\perp X \,|\, (\boldsymbol{M} \backslash \{X\}) \cup \{Y\} \quad \text{and} \tag{4}$$

$$T \perp\!\!\!\perp Y \,|\, \boldsymbol{M} \backslash \{X\} \tag{5}$$

hold simultaneously.

Before presenting the proof, we explain why `PostBW` may work after implementing `BW`. In fact, if a WMb (*resp.*, an Mb), $\boldsymbol{M}$, is not a WMB (*resp.*, an MB) of T, then

$$T \perp\!\!\!\perp X \,|\, \boldsymbol{M} \backslash \{X\} \tag{6}$$

holds for some $X \in \boldsymbol{M}$. However, if the corresponding CI test is incorrect, X will remain in $\boldsymbol{M}$ and thus masking occurs; in this case, the way of identifying false positives from a WMb or Mb by means of the `BW` procedure becomes invalid. Theorem 4 provides an alternative way for this purpose. Imagine that a false positive can enter the CMb in the growing phase because it possesses the aptitude of masking itself, and this false positive continues to mask itself in the shrinking phase. In this scenario, `BW` only employs the members of $\boldsymbol{M}$, so it may fail to identify all false positives; alternatively, `PostBW` employs the members and nonmembers of $\boldsymbol{M}$ simultaneously, so it may find some false positives that are accepted by `BW`. This is why `PostBW` may further work after implementing the `BW` procedure.

Proof We first prove the necessity. Suppose there is some $X \in \boldsymbol{M}$ and some $Y \in \boldsymbol{V} \backslash \boldsymbol{M} \backslash \{T\}$ such that (4) and (5) hold simultaneously. Then, $T \perp\!\!\!\perp \{X, Y\} \,|\, \boldsymbol{M} \backslash \{X\}$, in view of the contraction property, so (6) follows from the decomposition property. Consider the case that $\boldsymbol{M}$ is a WMb, that is, $T \perp\!\!\!\perp Z \,|\, \boldsymbol{M}$ holds for any $Z \in \boldsymbol{V} \backslash \boldsymbol{M} \backslash \{T\}$. Equivalently, we have $T \perp\!\!\!\perp Z \,|\, (\boldsymbol{M} \backslash \{X\}) \cup \{X\}$.

Algorithm 2: `PostBW` and `InterPostBW`

Procedure: $\boldsymbol{M} \leftarrow$ `PostBW`$(\boldsymbol{D}, T, \boldsymbol{M}, \boldsymbol{W}, \boldsymbol{B})$ and $\boldsymbol{M} \leftarrow$ `InterPostBW`$(\boldsymbol{D}, T, \boldsymbol{M}, \boldsymbol{W}, \boldsymbol{B})$
Input: $\boldsymbol{D}$ is a data matrix; T is the target; $\boldsymbol{M}$ is a WMb (*resp.*, an Mb) of T; $\boldsymbol{W}$ is a whitelist; $\boldsymbol{B}$ is a blacklist.
Output: The output, $\boldsymbol{M}$, is a WMB (*resp.*, MB) of T.

```
//$\boldsymbol{M} \leftarrow$ PostBW($\boldsymbol{D}, T, \boldsymbol{M}, \boldsymbol{W}, \boldsymbol{B}$)
while $\boldsymbol{M}$ has changed do
    if $\exists\, X \in \boldsymbol{M} \backslash \boldsymbol{W}$ & $Y \in \boldsymbol{V} \backslash \boldsymbol{M} \backslash \boldsymbol{B} \backslash \{T\}$ s.t.
    (4)(5) then
        $\boldsymbol{M} \leftarrow \boldsymbol{M} \backslash \{X\}$
    end
end
return $\boldsymbol{M}$

//$\boldsymbol{M} \leftarrow$ InterPostBW($\boldsymbol{D}, T, \boldsymbol{M}, \boldsymbol{W}, \boldsymbol{B}$)
while $\boldsymbol{M}$ has changed do
    $\boldsymbol{M} \leftarrow$ BW($\boldsymbol{D}, T, \boldsymbol{M}, \boldsymbol{W}$)
    if $\boldsymbol{M}$ has not changed then
        if $\exists\, X \in \boldsymbol{M} \backslash \boldsymbol{W}$ & $Y \in \boldsymbol{V} \backslash \boldsymbol{M} \backslash \boldsymbol{B} \backslash \{T\}$ s.t.
        (4)(5) then
            $\boldsymbol{M} \leftarrow \boldsymbol{M} \backslash \{X\}$
        end
    end
end
return $\boldsymbol{M}$
```

This combined with (6) and the contraction property implies $T \perp\!\!\!\perp \{Z, X\}|\boldsymbol{M}\backslash\{X\}$. Hence, $T \perp\!\!\!\perp U|\boldsymbol{M}\backslash\{X\}$ holds for any $U \in (\boldsymbol{V}\backslash\boldsymbol{M}\backslash\{T\}) \cup \{X\} = \boldsymbol{V}\backslash(\boldsymbol{M}\backslash\{X\}) \backslash \{T\})$. This contradicts that $\boldsymbol{M}$ is a WMB. In the case that $\boldsymbol{M}$ is an Mb, we can similarly verify $T \perp\!\!\!\perp \boldsymbol{V}\backslash(\boldsymbol{M}\backslash\{X\}) \backslash \{T\}|\boldsymbol{M}\backslash\{X\}$, which contradicts that $\boldsymbol{M}$ is an MB. The proof of the necessity is completed.

Now we show the sufficiency. For any $X \in \boldsymbol{M}$ and $Y \in \boldsymbol{V}\backslash\boldsymbol{M}\backslash\{T\}$, (4) and (5) do not hold simultaneously. In other words, $\mathbb{I}(T; X|(\boldsymbol{M}\backslash\{X\}) \cup \{Y\}) > 0$ or $\mathbb{I}(T; Y|\boldsymbol{M}\backslash\{X\} > 0$. On the other hand, $T \perp\!\!\!\perp Y|\boldsymbol{M}$ holds since $\boldsymbol{M}$ is a WMb (or an Mb), so $\mathbb{I}(T; Y|\boldsymbol{M}) = 0$. By the chain rule for CMI, we have

$$\begin{aligned}\mathbb{I}(T; X|\boldsymbol{M}\backslash\{X\}) &= \mathbb{I}(T; \{X, Y\}|\boldsymbol{M}\backslash\{X\}) - \mathbb{I}(T; Y|(\boldsymbol{M}\backslash\{X\}) \cup \{X\}) \\ &= \mathbb{I}(T; \{X, Y\}|\boldsymbol{M}\backslash\{X\}) - \mathbb{I}(T; Y|\boldsymbol{M}) \\ &= \mathbb{I}(T; Y|\boldsymbol{M}\backslash\{X\}) + \mathbb{I}(T; X|(\boldsymbol{M}\backslash\{X\}) \cup \{Y\}) - 0 \\ &> 0.\end{aligned}$$

Therefore, $T \not\perp\!\!\!\perp X|\boldsymbol{M}\backslash\{X\}$ holds for any $X \in \boldsymbol{M}$. By Theorem 2 (*resp.*, Theorem 1), $\boldsymbol{M}$ is a WMB (*resp.*, an MB) of T. The proof of the sufficiency is also completed. □

To examine the performance of PostBW, we apply this procedure to the Mbs of X_3, X_6, X_8, X_{11}, and X_{32} outputted by IAMB and LRH (see Table 3 for details). All the false positives accepted by BW are identified by PostBW and all the true MBs for these five targets are correctly discovered, except that the MB of X_{11} is obtained as $\{X_{34}, X_{36}\}$. This shows that PostBW improves on BW substantially.

The computational complexity will increase if using PostBW: this procedure needs to do $O(|\boldsymbol{V}|\cdot|\boldsymbol{M}_T|)$ extra CI tests. A feasible solution for alleviating the resulted computational cost is to interleave PostBW with BW. Following this idea, we first implement BW in each iteration, and then activate PostBW if BW stops (in each iteration). For convenience, we call this interleaved procedure to be InterPostBW, and present its pseudo code in Algorithm 2. Finally, we apply InterPostBW to the Mbs of X_3, X_6, X_8, X_{11}, and X_{32} outputted by IAMB and LRH. The results indicate InterPostBW has the same performance as PostBW in the sense of RE but it needs less RT for most situations.

5 WLCMB algorithmic framework

Section 4 considered the problem of (P1) and proposed the LRH algorithm. As we saw, LRH is time efficient and much more data efficient than IAMB and KIAMB. However, as Example 2 shows, the Markov local composition assumption may be violated in practice and, if this is the case, LRH and the other two LCMB algorithms will stop to search before finding a true MB. In this section, we consider the problem of (P2) as follows: analyze why swamping and masking occur in the case of violating the Markov local composition assumption, discuss how to overcome them by resuming the stopped search of LCMB, and build a corresponding algorithmic framework.

Recalling Example 2 considered in Sect. 1, IAMB incorrectly outputs $\{Z\}$ as the MB of T, meaning the two true positives (i.e., X and Y) are swamped by Z, and the false positive (i.e., Z) successfully masks itself. This indicates the dynamic heuristic in the growing phase of IAMB may lead to swamping, which may further bring masking. Similarly, LRH incorrectly outputs $\{Z\}$ as the MB of T, so LRH is also invalid for this type of swamping and masking. We also find that KIAMB as a random version of IAMB may discover the true MB if implementing it repeatedly; but this possibility is low. In addition, GS may find the true MB but this depends on the preassigned priority of variables checked in every search; swamping and masking will happen if, for example, the priority is "Z, X, Y" or "Z, Y, X". Thus, LCMB may prematurely terminate the growing phase if the CMb shields T from every remaining single variable.

Let $\boldsymbol{M}$ be a true MB of T in $\boldsymbol{V}$, and $\boldsymbol{M}_{\mathbb{A}} \triangleq (\boldsymbol{M}\backslash\boldsymbol{X}) \cup \boldsymbol{Y}$ be the output of an MB discovery algorithm, $\mathbb{A}$. Under the assumption that all CI tests are correct, Theorems 2 and 3 show that $\boldsymbol{M}_{\mathbb{A}}$ is a WMB of T in $\boldsymbol{V}$. Further, $\boldsymbol{M}_{\mathbb{A}}$ is not an MB (and thus also not an Mb), if the local composition assumption with respect to $\boldsymbol{M}_{\mathbb{A}}$ is violated. In this case, $\boldsymbol{X} \neq \varnothing$, so swamping must occur. The questions are then: (1) why some useful information about T carried by some variables in $\boldsymbol{V}\backslash\boldsymbol{M}\backslash\{T\}$ can not be captured successfully by $\mathbb{A}$? (2) how to resume the stopped search of $\mathbb{A}$?

For convenience, we denote $\boldsymbol{X} \triangleq \{X_{i_1}, \ldots, X_{i_k}\}$. First, we note the following conclusions:

- $T \not\perp\!\!\!\perp \boldsymbol{X}|\boldsymbol{M}_{\mathbb{A}}$: On the one hand, $\boldsymbol{M}$ is an MB, meaning $T \perp\!\!\!\perp \boldsymbol{V}\backslash\boldsymbol{M}\backslash\{T\}|\boldsymbol{M}$, so $T \perp\!\!\!\perp \boldsymbol{V}\backslash(\boldsymbol{M} \cup \boldsymbol{Y})\backslash\{T\}|(\boldsymbol{M}\cup\boldsymbol{Y})$ in view of the weak union property. Equivalently, we have $T \perp\!\!\!\perp \boldsymbol{V}\backslash(\boldsymbol{M}_{\mathbb{A}} \cup \boldsymbol{X})\backslash\{T\}|(\boldsymbol{M}_{\mathbb{A}} \cup \boldsymbol{X})$. On the other hand, $\boldsymbol{M}_{\mathbb{A}}$ is not an Mb. Suppose $T \perp\!\!\!\perp \boldsymbol{X}|\boldsymbol{M}_{\mathbb{A}}$, then the contraction property indicates
$$\left.\begin{array}{l} T \perp\!\!\!\perp \boldsymbol{V}\backslash(\boldsymbol{M}_{\mathbb{A}} \cup \boldsymbol{X})\backslash\{T\}|(\boldsymbol{M}_{\mathbb{A}} \cup \boldsymbol{X}) \\ T \perp\!\!\!\perp \boldsymbol{X}|\boldsymbol{M}_{\mathbb{A}} \end{array}\right\} \Rightarrow T \perp\!\!\!\perp \boldsymbol{V}\backslash\boldsymbol{M}_{\mathbb{A}}\backslash\{T\}|\boldsymbol{M}_{\mathbb{A}},$$
which contradicts that $\boldsymbol{M}_{\mathbb{A}}$ is not an Mb of T. Hence, $T \not\perp\!\!\!\perp \boldsymbol{X}|\boldsymbol{M}_{\mathbb{A}}$.
- $k \geqslant 2$, and $T \perp\!\!\!\perp X_{i_\ell}|\boldsymbol{M}_{\mathbb{A}}$ holds for any $\ell = 1, \ldots, k$: $\boldsymbol{M}_{\mathbb{A}}$ is a WMB (and thus a WMb) of T.
- $T \not\perp\!\!\!\perp \boldsymbol{N}|\boldsymbol{M}_{\mathbb{A}}\backslash\boldsymbol{N}$ holds for any nonempty $\boldsymbol{N} \subseteq \boldsymbol{M}_{\mathbb{A}}$: This is because $\boldsymbol{M}_{\mathbb{A}}$ is a WMB of T.

The first two conclusions mean those true positives in $\boldsymbol{X}$ are swamped by $\boldsymbol{M}_{\mathbb{A}}$; the idea of the third one will be used in Definition 5. As seen in Example 2, the local composition assumption will be violated, if $\boldsymbol{M}_{\mathbb{A}}$ contains all unique information and all redundant information about T carried by each X_{i_ℓ} as well as some (but not all) synergistic information about T carried jointly by $X_{i_1}, \ldots, X_{i_k}$. That synergistic information about T carried jointly by $X_{i_1}, \ldots, X_{i_k}$ and also by $\boldsymbol{M}_{\mathbb{A}}$ swamps the remaining useful information about T carried by $X_{i_1}, \ldots, X_{i_k}$ but not by $\boldsymbol{M}_{\mathbb{A}}$. In this sense, swamping occurs and the search of LCMB ends; masking may then follow.

Definition 5 (*WMB-supplementary*) For $T \in \boldsymbol{V}$, let $\boldsymbol{M}_{\mathbb{A}}$ be a WMB of T in $\boldsymbol{V}$. For $\boldsymbol{S} \subseteq \boldsymbol{M}_{\mathbb{A}}$, we call $\boldsymbol{N}_{\boldsymbol{S}}$ ($\subseteq \boldsymbol{V}\backslash\boldsymbol{M}_{\mathbb{A}}\backslash\{T\}$) a WMB-supplementary of $\boldsymbol{S}$, if the following two conditions hold: (1) $(\boldsymbol{M}_{\mathbb{A}}\backslash\boldsymbol{S}) \cup \boldsymbol{N}_{\boldsymbol{S}}$ is a WMb of T in $\boldsymbol{V}\backslash\boldsymbol{S}$; and (2) $T \not\perp\!\!\!\perp \boldsymbol{N}|(\boldsymbol{M}_{\mathbb{A}}\backslash\boldsymbol{S}) \cup (\boldsymbol{N}_{\boldsymbol{S}}\backslash\boldsymbol{N})$ holds for any nonempty $\boldsymbol{N} \subseteq \boldsymbol{N}_{\boldsymbol{S}}$, if $\boldsymbol{N}_{\boldsymbol{S}} \neq \varnothing$.

The analysis before Definition 5 provides a method of resuming the search of LCMB: if putting a set of all (or part) of nodes from $\boldsymbol{M}_{\mathbb{A}}$, say $\boldsymbol{S}$, into the blacklist temporarily, some swamped information may be detected, so the search can continue. For convenience, we call $\boldsymbol{S}$ a *swamping set*. Observing again Example 2, X and Y are no longer swamped if removing Z temporarily. Example 3 presented in the appendix gives a similar inspiration. If we can find one swamping set, $\boldsymbol{S}$, then we remove it temporarily to search the variables swamped by $\boldsymbol{S}$.

In practice, we can not seek $\boldsymbol{S}$ directly; instead, we may check every possible subset of $\boldsymbol{M}_{\mathbb{A}}$. The resulted heuristic is as follows: for any nonempty $\boldsymbol{S} \subseteq \boldsymbol{M}_{\mathbb{A}}$, find a WMB-supplementary of it, $\boldsymbol{N}_{\boldsymbol{S}}$. (1) If $\boldsymbol{S}$ is a swamping set, some of those variables (saying $X_{i_1}, \ldots, X_{i_k}$) swamped by $\boldsymbol{S}$ may be found and thus enter $\boldsymbol{N}_{\boldsymbol{S}}$. $X_{i_1}, \ldots, X_{i_k}$ contain some synergistic information about T, some of which is carried by $\boldsymbol{S}$ and thus by $\boldsymbol{M}_{\mathbb{A}}$, and some other may not be carried by $\boldsymbol{M}_{\mathbb{A}}$. This means $\boldsymbol{T} \not\perp\!\!\!\perp \boldsymbol{N_S}|\boldsymbol{M}_{\mathbb{A}}$. (2) Conversely, if $\boldsymbol{S}$ is not a swamping set, then $\boldsymbol{T} \perp\!\!\!\perp \boldsymbol{N_S}|\boldsymbol{M}_{\mathbb{A}}$. In this sense, if $\boldsymbol{T} \perp\!\!\!\perp \boldsymbol{N_S}|\boldsymbol{M}_{\mathbb{A}}$ holds for every $\boldsymbol{S}(\subseteq \boldsymbol{M}_{\mathbb{A}})$, we may think T has no swamping set, so in this case it is reasonable to assume T satisfies local composition with respect to such a WMB, $\boldsymbol{M}_{\mathbb{A}}$. Otherwise, once a swamping set, $\boldsymbol{S}$, is found, the growing phase stopped in $\mathbb{A}$ may resume, and we can update $\boldsymbol{M}_{\mathbb{A}}$ based on $\boldsymbol{M}_{\mathbb{A}}$ and $\boldsymbol{S}$. Repeat this procedure until no swamping sets can be found. This is a potentially feasible solution to the problem of (P2).

Following this way, we construct an `LCMB`-based algorithmic framework called `WLCMB`. Here, “`W`” refers to as “weak”; we call it `WLCMB` because it can output an MB under the *weak Markov local composition* assumption defined as below.

Definition 6 (*Weak Markov local composition*) We say T satisfies the weak Markov local composition property, if every WMB, $\boldsymbol{M}_{\mathbb{A}}$, of T satisfying the following condition is an MB of T: any $\boldsymbol{S} \subseteq \boldsymbol{M}_{\mathbb{A}}$ has a WMB-supplementary $\boldsymbol{N_S}$ such that $T \perp\!\!\!\perp \boldsymbol{N_S}|\boldsymbol{M}_{\mathbb{A}}$.

The pseudo code of `WLCMB` is described in Algorithm 3. By the algorithm, `WLCMB` interleaves `LCMB` (i.e., Line 7 of Algorithm 3) with the search-resuming procedure (i.e., Line 9, and Line 10 of Algorithm 3) by virtue of the `ImpWMB` function. If taking $\mathbb{A}$ as `IAMB`, `KIAMB`, and `LRH`, the corresponding `WLCMB` algorithm will be called `WIAMB`, `WKIAMB`, and `WLRH`, respectively. Moreover, the `BW` procedure in Algorithm 3 can also be replaced with `PostBW` or `InterPostBW`.

Theorem 5 (Correctness of `WLCMB`) *Assume all CI tests are correct. Then* `WLCMB` *outputs a WMB of T for any* `LCMB` *algorithm taken from* {`IAMB`, `KIAMB`, `LRH`}. *Further, if T satisfies the weak Markov local composition assumption, then* `WLCMB` *outputs an MB of T.*

Proof Put $\boldsymbol{N_S} \triangleq \boldsymbol{M_S}\backslash(\boldsymbol{M}\backslash\boldsymbol{S})$, where $\boldsymbol{M_S}$ is derived in Line 7 of Algorithm 3. Similar to the proofs of Theorems 2 and 3, it can be shown that $\boldsymbol{N_S}$ is a WMB-supplementary of $\boldsymbol{S}$. Denote the outputs of Line 1, Line 7, Line 9, and Line 10 in the k-th iteration ($k \geqslant 1$, if possible) by $\boldsymbol{M}^{(0)}$, $\boldsymbol{M}_{\boldsymbol{S}}^{(k)}$, $\boldsymbol{M}_{\mathtt{FW}}^{(k)}$, and $\boldsymbol{M}^{(k)}$, respectively, before the iterated procedure ends. Then, by the fact that `FW` collects more information about T while `BW` removes redundant variables only, we have $\mathbb{I}(T; \boldsymbol{M}^{(0)}) < \mathbb{I}(T; \boldsymbol{M}^{(1)})$, and

Algorithm 3: `WLCMB`

Procedure: $\boldsymbol{M} \leftarrow$ `WLCMB`($\mathbb{A}$; $\boldsymbol{D}, T, \boldsymbol{W}, \boldsymbol{B}$)
Input: $\mathbb{A}$ is an `LCMB` algorithm;
$\boldsymbol{D}$ is a data matrix; T is the target; $\boldsymbol{W}$ is a whitelist; $\boldsymbol{B}$ is a blacklist.
Output: The output, $\boldsymbol{M}$, is an MB of T under the weak Markov local composition assumption.

```
   //main procedure: $M \leftarrow$ WLCMB($\mathbb{A}$; $D, T, W, B$)
1  $M \leftarrow$ LCMB($\mathbb{A}$; $D, T, W, B$)
2  while $M$ has changed do
3      $M \leftarrow$ ImpWMB($\mathbb{A}$; $D, T, M, W, B$)
4  end
5  return $M$

   //sub-routine:
      $M \leftarrow$ ImpWMB($\mathbb{A}$; $D, T, M, W, B$)
6  foreach $S \subseteq M \backslash W$ do
7      $M_S \leftarrow$ LCMB($\mathbb{A}$; $D, T, M \backslash S, B \cup S$)
8      if $T \not\perp\!\!\!\perp M_S \backslash (M \backslash S) | M$ then
9          $M_{FW} \leftarrow$ FW($\mathbb{A}$; $D, T, M_S \cup S, B$)
10         $M \leftarrow$ BW($D, T, M_{FW}, W$)
11         break
12     end
13 end
```

$$
\begin{aligned}
&\mathbb{I}\left(T; \boldsymbol{M}^{(k-1)}\right) \\
&\quad < \mathbb{I}\left(T; \boldsymbol{M}^{(k-1)}\right) + \mathbb{I}\left(T; \boldsymbol{M}_{\boldsymbol{S}}^{(k)} \backslash \left(\boldsymbol{M}^{(k-1)} \backslash \boldsymbol{S}\right) | \boldsymbol{M}^{(k-1)}\right) && (\text{“} <'' \text{ is due to Line 8}) \\
&\quad = \mathbb{I}\left(T; \left[\boldsymbol{M}_{\boldsymbol{S}}^{(k)} \backslash \left(\boldsymbol{M}^{(k-1)} \backslash \boldsymbol{S}\right)\right] \cup \boldsymbol{M}^{(k-1)}\right) && (\text{using the chain rule for } \mathit{CMI}) \\
&\quad = \mathbb{I}\left(T; \boldsymbol{M}_{\boldsymbol{S}}^{(k)} \cup \boldsymbol{S}\right) && \left(\text{since } \boldsymbol{M}^{(k-1)} \backslash \boldsymbol{S} \subseteq \boldsymbol{M}_{\boldsymbol{S}}^{(k)}\right) \\
&\quad \leqslant \mathbb{I}\left(T; \boldsymbol{M}_{\mathtt{FW}}^{(k)}\right) && (\text{“} \leqslant'' \text{ is due to Line 9}) \\
&\quad = \mathbb{I}\left(T; \boldsymbol{M}^{(k)}\right) && (\text{“} ='' \text{ is due to Line 10})
\end{aligned}
$$

Therefore, the exit condition, $T \perp\!\!\!\perp \boldsymbol{N}_{\boldsymbol{S}} | \boldsymbol{M}$ holding for any $\boldsymbol{S} \subseteq \boldsymbol{M}$ shown in Line 8, will be satisfied after a number of iterations. Once the exit condition is satisfied, the weak Markov local composition assumption indicates that the output of WLCMB is then an MB of T. The proof is completed. □

Recall Example 2 presented in Sect. 1, where the Markov local composition assumption is violated. Specifically, $T \perp\!\!\!\perp X|Z$, $T \perp\!\!\!\perp Y|Z$, $T \not\perp\!\!\!\perp \{X, Y\}|Z$, and $T \perp\!\!\!\perp Z|\{X, Y\}$. Using the notations employed in the proof of Theorem 5, we have

- IAMB and WIAMB: First, IAMB outputs $\boldsymbol{M}^{(0)} = \{Z\}$, which is not the MB of T. Taking $\boldsymbol{S} = \{Z\} \subseteq \boldsymbol{M}^{(0)}$, we obtain $\boldsymbol{M}_{\boldsymbol{S}}^{(1)} = \{X, Y\}$, meaning $T \not\perp\!\!\!\perp \boldsymbol{M}_{\boldsymbol{S}}^{(1)} \backslash (\boldsymbol{M}^{(0)} \backslash \boldsymbol{S}) | \boldsymbol{M}^{(0)}$ (i.e., $T \not\perp\!\!\!\perp \{X, Y\}|Z$). Further, $\boldsymbol{M}_{\mathtt{FW}}^{(1)} = \{X, Y, Z\}$ and $\boldsymbol{M}^{(1)} = \{X, Y\}$. Similarly, $\boldsymbol{M}^{(2)} = \{X, Y\} = \boldsymbol{M}^{(1)}$. Thus, WIAMB ends, outputing $\{X, Y\}$ correctly.
- KIAMB and WKIAMB: The output of KIAMB may be $\boldsymbol{M}^{(0)} = \{X, Y\}$ or $\boldsymbol{M}^{(0)} = \{Z\}$. In either case, WKIAMB can output the correct MB. The details are omitted here.
- LRH and WLRH: First, LRH selects $\{X, Y, Z\}$ and excludes $\{X, Y\}$ in its SEI procedure. Therefore, LRH outputs $\boldsymbol{M}^{(0)} = \{Z\}$. The remaining process of WLRH is similar to that of WIAMB. Finally, WLRH outputs $\{X, Y\}$ correctly.

This illustrates how WLCMB works when the Markov local composition assumption is violated.

To examine the performance of WLCMB algorithms, we apply them to the ALARM network. Table 5 presents the corresponding results, including the outputted MBs of WIAMB and WLRH for all the 37 nodes as targets. The REs of each WLCMB are also given. By Tables 3 and 5, it is concluded that each WLCMB improves on the corresponding LCMB substantially. Specifically, IAMB retrieves 8 and 5 MBs, and yields 15 incorrect outputs, while WIAMB retrieves 17 and 1 MBs, and yields 12 incorrect outputs; LRH gives 21, 3 MBs, and only one incorrect output, while WLRH yields 25, 1 MBs, and no incorrect outputs. The results also show WLRH performs best.

Similar to Table 4, the first part of Table 6 lists the average REs of the three WLCMB algorithms applied to the ALARM network based on the same 10 data sets, in which each result is averaged over all the 37 nodes as targets. Comparing Table 6 with Table 4, it is seen that WLCMB performs better than LCMB for all cases of data sizes and thus is more data efficient.

We mention that WLCMB has a higher computational complexity than LCMB: the complexity of WLCMB is that of the associated LCMB multiplied by $2^{|\boldsymbol{M}|}$ in the average case. Hence, WLCMB usually needs longer RT to yield a better output than LCMB. This can be seen from the second part of Table 6, which provides the average RT of the three WLCMB algorithms applied to the ALARM network. The experimental results on several large networks given in Sect. 6 also show this assertion. This means we should trade off the expected RE and RT before deciding to select which MB discovery algorithm in practice.

Table 5 Results of the WLCMB algorithms applied to the ALARM network based on a data set of size 5000 ($\alpha = 0.05$)

Target	MB (subscripts)			RE				
	True	WIAMB	WLRH	WIAMB	WKIAMB			WLRH
					$K = 0.2$	$K = 0.5$	$K = 0.8$	
X_1	22, 29	22	22, 29	0.9896	0.9896	0.9610	0.9896	1.0000
X_2	23, 27, 29	4, 18, 23	23, 27, 29	0.7672	0.8852	0.9767	0.8977	1.0000
X_3	30	30	30	1.0000	1.0000	1.0000	1.0000	1.0000
X_4	15, 22	1, 15	15, 22	0.9691	0.9437	0.9732	0.9680	1.0000
X_5	6, 13	6, 13	6, 13	1.0000	1.0000	0.9918	0.9865	1.0000
X_6	5, 13	5, 13	5, 13	1.0000	0.9908	0.9908	1.0000	1.0000
X_7	9, 13	9, 13	9, 13	1.0000	1.0000	0.9920	1.0000	1.0000
X_8	9, 13	9, 13	9, 13	1.0000	1.0000	1.0000	0.9916	1.0000
X_9	7, 8, 13	7, 8, 13	7, 8, 13	1.0000	0.9916	0.9973	0.9973	1.0000
X_{10}	12, 16	12, 16	12, 16	1.0000	1.0000	0.9739	1.0000	1.0000
X_{11}	36	34, 36	34, 36	1.0000	1.0000	1.0000	1.0000	1.0000
X_{12}	10, 13, 16, 34	10, 13, 16, 34	10, 13, 16, 34	1.0000	1.0000	1.0000	1.0000	1.0000
X_{13}	5, 6, 7, 8, 9, 12, 14, 34	5, 7, 8, 9, 14	7, 8, 12, 14, 34	0.8591	0.9085	0.9472	0.9554	0.9587
X_{14}	13, 15, 16, 18, 31	4, 13, 16, 18, 31	13, 15, 16, 18, 31	0.9737	0.8182	0.9639	0.9625	1.0000
X_{15}	4, 14, 16, 18, 21, 22, 31	1, 4, 19	4, 21	0.9098	0.9048	0.9281	0.9280	0.9305
X_{16}	10, 12, 14, 15, 17, 18, 31	14, 15, 17, 18, 31	14, 15, 17, 18, 31	0.9125	0.9041	0.8573	0.8698	0.9125
X_{17}	16	16	16	1.0000	1.0000	1.0000	1.0000	1.0000
X_{18}	14, 15, 16, 19, 28, 31	19, 28	19, 28	0.9302	0.9249	0.9255	0.9359	0.9302
X_{19}	18, 20, 21, 28	4, 18, 28	18, 20, 21, 28	0.9533	0.9840	0.9877	0.9308	1.0000
X_{20}	19, 21	19, 21	19, 21	1.0000	0.9614	0.9807	0.9592	1.0000
X_{21}	15, 19, 20, 22, 29	2, 4, 18, 28	15, 19, 22, 29	0.8729	0.9029	0.8959	0.9273	0.9901
X_{22}	1, 4, 15, 21, 23, 27, 29	1, 4	1, 21, 23	0.8980	0.8997	0.9144	0.9008	0.9614
X_{23}	2, 22, 24, 25, 27, 29	1, 2, 25	2, 22, 25	0.8844	0.8436	0.9214	0.8813	0.9281
X_{24}	23, 25	23, 25	23, 25	1.0000	0.9707	0.9553	0.9338	1.0000
X_{25}	23, 24, 26	23, 24, 26	23, 24, 26	1.0000	1.0000	1.0000	1.0000	1.0000
X_{26}	25	25	25	1.0000	1.0000	1.0000	1.0000	1.0000
X_{27}	2, 22, 23, 29	2, 23	2, 23	0.7431	0.6901	0.7354	0.7144	0.7431
X_{28}	18, 19, 29, 30	2, 18, 19	18, 19, 29	0.9882	0.9563	0.9671	0.9942	0.9968
X_{29}	1, 2, 21, 22, 23, 27, 28, 30	2, 18, 19	21, 22, 28	0.7419	0.8465	0.9191	0.8866	0.9558
X_{30}	3, 28, 29	3, 28, 29	3, 28, 29	1.0000	0.9657	0.9801	0.9834	1.0000
X_{31}	14, 15, 16, 18	4, 14, 16, 18	14, 15, 16, 18	0.9345	0.9176	0.9283	1.0000	1.0000
X_{32}	35	35	35	1.0000	1.0000	1.0000	1.0000	1.0000
X_{33}	35	35	35	1.0000	1.0000	1.0000	1.0000	1.0000
X_{34}	12, 13, 36, 37	12, 13, 36, 37	12, 13, 36, 37	1.0000	0.9934	0.9561	0.9946	1.0000
X_{35}	32, 33, 36, 37	32, 33	32, 33, 36, 37	0.9823	0.9935	0.9935	0.9913	1.0000
X_{36}	11, 34, 35, 37	11, 32, 34, 37	34, 35, 37	0.9717	0.9820	0.9917	0.9830	0.9936
X_{37}	34, 35, 36	32, 33, 34, 36	34, 35, 36	0.9908	0.9918	0.9945	1.0000	1.0000

The backcolors are the same as in Table 3

6 Experimental results on large networks

Tables 4 and 6 showed the superiority of LRH over IAMB and KIAMB in discovering an MB of the target for small BNs. The results also demonstrated the effectiveness of our WLCMB algorithmic framework in further improving the data efficiency of LCMB. This section applies the algorithms to some large BNs, based on the data sets of size 5000 used by Tsamardinos et al. (2006) and Aliferis et al. (2010a). These data sets are available at http://www.dsl-lab.org/supplements/JMLR2008/. The used networks are representatives of a wide range of problem domains; Table 7 lists the numbers of nodes and edges for them. Tsamardinos et al. (2006)

Table 6 Average REs and RTs of the WLCMB algorithms applied to the ALARM network based on 10 data sets of different sizes (from 500 to 5000): each result is averaged over all the 37 nodes as targets

WLCMB		Data size									
		500	1000	1500	2000	2500	3000	3500	4000	4500	5000
Average REs											
WIAMB		0.9088	0.9343	0.9240	0.9289	0.9220	0.9536	0.9386	0.9507	0.9460	0.9533
WKIAMB	$K = 0.2$	0.8580	0.9052	0.9421	0.9477	0.9210	0.9353	0.9542	0.9515	0.9624	0.9631
	$K = 0.5$	0.8909	0.9383	0.9269	0.9517	0.9552	0.9665	0.9656	0.9662	0.9586	0.9531
	$K = 0.8$	0.9160	0.9196	0.9496	0.9538	0.9512	0.9664	0.9503	0.9622	0.9603	0.9530
WLRH		0.9286	0.9460	0.9545	0.9621	0.9681	0.9722	0.9768	0.9790	0.9807	0.9811
Average RTs (in seconds)											
WIAMB		0.1982	0.2926	0.3586	0.4363	0.5664	0.6347	0.6885	0.7788	0.8112	0.9762
WKIAMB	$K = 0.2$	0.2386	0.3654	0.4090	0.5475	0.6314	0.6960	0.8379	0.9130	1.0280	1.1869
	$K = 0.5$	0.1964	0.3107	0.3750	0.4676	0.5790	0.6714	0.8359	0.8255	1.1606	1.2241
	$K = 0.8$	0.1915	0.2982	0.4070	0.4806	0.5653	0.6385	0.8050	0.8723	1.0192	1.0786
WLRH		0.2492	0.3997	0.5114	0.6568	0.7370	0.8110	0.9506	1.1313	1.2881	1.3929

and Aliferis et al. (2010a) provided more details about these networks and the used data sets. For each BN, we also use a data set of size 2500 randomly drawn from the original data set to evaluate the performance of the algorithms in data efficiency.

For each network, 10 nodes are randomly selected as targets. These targets are listed in Table 7. Besides the REs and RTs, we also compute the weighted area under ROC curve (AUC) based on the naive Bayes classifier. For the case of size 5000, we randomly select 4000 instances as the training set and use the others as the testing set; for the case of size 2500, we randomly select 2000 and 500 out of 5000 instances as the training set and the testing set, respectively. Table 7 presents all the results. For each case, the AUC of the true MB, $\boldsymbol{M}_T$, of the target is provided to compare how the performance of an algorithm is close to the best. Each result is averaged over that of the 10 targets. In addition, according to the recommendation of Peña et al. (2007) and the results given in Sects. 4 and 5, we take $K = 0.8$ in KIAMB and WKIAMB.

Table 7 indicates our algorithms are applicable to large BNs. By the table, it is concluded that: (1) for the three LCMB algorithms, LRH performs best in the senses of RE and AUC; (2) for the three WLCMB algorithms, WLRH performs best in both senses; (3) for each LCMB and its corresponding WLCMB, the latter improves the data efficiency of the former. In addition, we note a natural conclusion that the results on the case of a larger data size are more desirable in most situations than that on the case of a smaller data size. In brief, LRH and WLRH have the best performances in solving (P1) and (P2), respectively. Considering that WLRH usually needs a longer RT than LRH as Sect. 5 analyzes, we should first trade off the RE and the RT in practice and then choose between these two algorithms.

7 Conclusion and discussions

This paper considered two potential reasons for causing swamping and masking. For the problem of (P1) that incorrect CI tests may lead to swamping and masking, we proposed the LRH algorithm to alleviate the influence that swamping and masking brings under the local composition assumption. The application to the ALARM network shows the superiority of

Table 7 Results of LCMBs and WLCMBs applied to the six large BNs based on two data sets of sizes 5000 and 2500: average AUCs, average REs, and average RTs (in seconds)

BN	Number of nodes	Number of edges	Targets (subscripts)	Results	Data size	True	LCMB algorithm			WLCMB algorithm		
							IAMB	KIAMB	LRH	WIAMB	WKIAMB	WLRH
ALARM10	370	570	12, 44, 112, 116, 130, 160, 200, 234, 308, 333	AUC	5000	0.9503	0.8943	0.9310	0.9484	0.9134	0.9432	0.9492
					2500	0.9481	0.9220	0.9335	0.9428	0.9384	0.9247	0.9428
				RE	5000	—	0.7669	0.8727	0.9803	0.8458	0.9610	0.9867
					2500	—	0.8718	0.8968	0.9329	0.9207	0.9104	0.9329
				RT	5000	—	4.6595	3.7783	3.8863	9.9177	9.2547	5.8765
					2500	—	2.6342	2.1862	2.5178	5.1783	5.1975	3.8798
Child10	200	257	3, 58, 83, 103, 123, 138, 158, 163, 183, 198	AUC	5000	0.9027	0.8297	0.8406	0.8917	0.8347	0.8819	0.9012
					2500	0.9019	0.8251	0.8107	0.8502	0.8334	0.8522	0.8601
				RE	5000	—	0.5781	0.6327	0.8121	0.6337	0.8641	0.8981
					2500	—	0.5632	0.5767	0.6625	0.6192	0.7285	0.7296
				RT	5000	—	2.3556	2.4196	2.7380	5.5186	5.5474	10.178
					2500	—	1.2655	1.4448	1.3531	2.6504	3.0767	2.5558
Gene	801	977	37, 113, 151, 232, 247, 295, 347, 385, 484, 767	AUC	5000	0.9290	0.8074	0.8686	0.9259	0.8257	0.8696	0.9266
					2500	0.9281	0.8496	0.8724	0.9195	0.8878	0.8986	0.9217
				RE	5000	—	0.5246	0.6861	0.9599	0.6136	0.7523	0.9695
					2500	—	0.6192	0.7136	0.8709	0.8550	0.8655	0.8889
				RT	5000	—	9.0206	10.099	19.906	23.737	23.476	31.571
					2500	—	5.3889	5.5577	10.683	11.578	12.061	14.357
Link	724	1125	110, 111, 155, 322, 323, 457, 522, 617, 655, 656	AUC	5000	0.9541	0.7828	0.8710	0.9372	0.8714	0.8721	0.9456
					2500	0.9574	0.8262	0.8338	0.9361	0.8849	0.8457	0.9517
				RE	5000	—	0.4688	0.6955	0.9152	0.7177	0.7361	0.9592
					2500	—	0.5968	0.6188	0.8922	0.7640	0.6388	0.9670
				RT	5000	—	8.4350	8.2773	12.182	24.329	23.518	20.709
					2500	—	4.8351	5.4237	7.5626	12.707	11.532	10.098
Lung cancer	800	1476	35, 36, 62, 174, 214, 252, 253, 382, 562, 612	AUC	5000	0.8770	0.8092	0.8002	0.8730	0.8525	0.8551	0.8730
					2500	0.8715	0.8209	0.8128	0.8627	0.8446	0.8469	0.8645
				RE	5000	—	0.6221	0.5649	0.8402	0.7690	0.7672	0.8402
					2500	—	0.6464	0.5641	0.7797	0.7912	0.7473	0.8012
				RT	5000	—	9.7682	9.5263	19.859	29.758	36.419	47.660
					2500	—	5.5976	6.1177	12.466	13.625	13.459	26.260
Pigs	441	592	62, 96, 166, 233, 251, 290, 335, 353, 404, 436	AUC	5000	0.8722	0.7308	0.7321	0.8649	0.7757	0.7988	0.8658
					2500	0.8625	0.7540	0.7933	0.8531	0.8021	0.8147	0.8531
				RE	5000	—	0.5019	0.4951	0.8863	0.6100	0.6789	0.8946
					2500	—	0.5869	0.7140	0.8507	0.7506	0.8040	0.8507
				RT	5000	—	5.1581	5.6745	13.060	14.435	13.161	16.422
					2500	—	3.2607	3.1209	8.8093	7.1789	7.3140	10.287

LRH over the other two LCMB algorithms. For the problem of (P2) that the violation of local composition may also lead to swamping and masking, we put forward the WLCMB algorithmic framework. Theoretically, WLCMB can improve LCMB, because LCMB stops searching once local composition is violated with respect to the obtained WMb in the growing phase, while WLCMB may break this abnormal exit and then continues to search those swamped true positives. The further application to the ALARM network supports this theoretical argument.

Motivated by one referee, we mention that Tables 3 and 5 also indirectly reflect the frequencies of swamping and masking with respect to IAMB or LRH or WIAMB or WLRH. Specifically, in both tables, “🟦” indicates neither swamping nor masking; “🟩” indicates swamping (but no masking); “🟧” indicates masking (but no swamping); others indicate both swamping and masking. In this sense, Table 8 counts the frequencies that swamping or masking occurs when applied to the ALARM network based on the used data set of size 5000. The results reveal that LRH and WLRH perform much better than IAMB and WIAMB in lessening swamping and resisting masking. The results also show that WIAMB (*resp.*, WLRH) improves IAMB (*resp.*, LRH) to some extent.

As a remark, we mention here that we modify $\boldsymbol{M}_2$ in the exclusion procedure of SEI if, for any $X \in \boldsymbol{M}_1$, there is some $Z \in \boldsymbol{N}_X$ such that $T \perp\!\!\!\perp X | \boldsymbol{M} \cup \{Z\}$ holds. See Sect. 4.2 for details. In fact, in the case of modifying $\boldsymbol{M}_2$, there must be κ ($\geqslant 2$) variables in $\boldsymbol{M}_1$ (without

Table 8 Frequencies of swamping and masking for LCMB and WLCMB when applied to the ALARM network

Frequency	IAMB	LRH	WIAMB	WLRH
Swamping	24/37	13/37	19/37	11/37
Masking	20/37	4/37	13/37	1/37

loss of generality, denote them by $X_1, \ldots, X_\kappa$) such that

$$T \perp\!\!\!\perp X_1 | \boldsymbol{M} \cup \{X_2\}, \quad \ldots, \quad T \perp\!\!\!\perp X_{\kappa-1} | \boldsymbol{M} \cup \{X_\kappa\}, \quad T \perp\!\!\!\perp X_\kappa | \boldsymbol{M} \cup \{X_1\} \tag{7}$$

hold simultaneously. If $\kappa = |\boldsymbol{M}_1|$, then $\boldsymbol{M}_2$ is empty before it is modified, so the search will be stopped; however, $T \not\perp\!\!\!\perp X | \boldsymbol{M}$ holds for any $X \in \boldsymbol{M}_1$, meaning $\boldsymbol{M}$ needs more variables to shield T. In this case, we modify $\boldsymbol{M}_2$ as $\{Y\}$ with $Y = \arg\max_{X \in \boldsymbol{M}_1} f_{\boldsymbol{D}}(T; X | \boldsymbol{M})$. This modification integrates the idea of IAMB such that LRH continues to search.

Here, we consider an alternative modification for $\boldsymbol{M}_2$ theoretically. Note that the CI statements given in (7) combined with $T \not\perp\!\!\!\perp X_i | \boldsymbol{M}$ $(i = 1, \ldots, \kappa)$ are similar to the definition for information equivalence (Lemeire et al. 2012). Now, we show $\mathbb{I}(T; \boldsymbol{M} \cup \{X_1\}) = \ldots = \mathbb{I}(T; \boldsymbol{M} \cup \{X_\kappa\})$, or equivalently,

$$\mathbb{I}(T; X_1 | \boldsymbol{M}) = \cdots = \mathbb{I}(T; X_\kappa | \boldsymbol{M}). \tag{8}$$

In fact, by (7) and the chain rule for CMI, we have

$$\begin{aligned}
\varrho &\triangleq \mathbb{I}(T; X_2 | \boldsymbol{M} \cup \{X_1\}) + \cdots + \mathbb{I}(T; X_\kappa | \boldsymbol{M} \cup \{X_{\kappa-1}\}) + \mathbb{I}(T; X_1 | \boldsymbol{M} \cup \{X_\kappa\}) \\
&= [\mathbb{I}(T; X_2 | \boldsymbol{M}) + \mathbb{I}(T; X_1 | \boldsymbol{M} \cup \{X_2\}) - \mathbb{I}(T; X_1 | \boldsymbol{M})] + \cdots \\
&\quad + \left[\mathbb{I}(T; X_\kappa | \boldsymbol{M}) + \mathbb{I}(T; X_{\kappa-1} | \boldsymbol{M} \cup \{X_\kappa\}) - \mathbb{I}(T; X_{\kappa-1} | \boldsymbol{M})\right] \\
&\quad + [\mathbb{I}(T; X_1 | \boldsymbol{M}) + \mathbb{I}(T; X_\kappa | \boldsymbol{M} \cup \{X_1\}) - \mathbb{I}(T; X_\kappa | \boldsymbol{M})] \\
&= \mathbb{I}(T; X_2 | \boldsymbol{M}) - \mathbb{I}(T; X_1 | \boldsymbol{M}) + \cdots + \mathbb{I}(T; X_\kappa | \boldsymbol{M}) - \mathbb{I}(T; X_{\kappa-1} | \boldsymbol{M}) + \mathbb{I}(T; X_1 | \boldsymbol{M}) \\
&\quad - \mathbb{I}(T; X_\kappa | \boldsymbol{M}) \\
&\equiv 0.
\end{aligned}$$

Combined with the nonnegativity of CMI, we obtain

$$\mathbb{I}(T; X_2 | \boldsymbol{M} \cup \{X_1\}) = \cdots = \mathbb{I}(T; X_\kappa | \boldsymbol{M} \cup \{X_{\kappa-1}\}) = \mathbb{I}(T; X_1 | \boldsymbol{M} \cup \{X_\kappa\}) = 0. \tag{9}$$

It follows from (7), (9) that

$$\begin{aligned}
\mathbb{I}(T; \{X_1, X_2\} | \boldsymbol{M}) &= \mathbb{I}(T; X_1 | \boldsymbol{M}) + \mathbb{I}(T; X_2 | \boldsymbol{M} \cup \{X_1\}) = \mathbb{I}(T; X_1 | \boldsymbol{M}) \\
&= \mathbb{I}(T; X_2 | \boldsymbol{M}) + \mathbb{I}(T; X_1 | \boldsymbol{M} \cup \{X_2\}) = \mathbb{I}(T; X_2 | \boldsymbol{M}).
\end{aligned}$$

This means $\mathbb{I}(T; X_1 | \boldsymbol{M}) = \mathbb{I}(T; X_2 | \boldsymbol{M})$. Similarly, we can show (8) holds for $\kappa \geqslant 2$.

In the sense of (8), we call $X_1, \ldots, X_\kappa$ to be *multiple information equivalent* with respect to T given $\boldsymbol{M}$ if $T \not\perp\!\!\!\perp X_i | \boldsymbol{M}$ $(i = 1, \ldots, \kappa)$ and the CI statements contained in (7) hold. As seen, in the case of $\kappa = 2$, the notion of multiple information equivalence reduces that of information equivalence proposed by Lemeire et al. (2012). Note that multiple information equivalence may exist in $\boldsymbol{M}_1$ even when $\boldsymbol{M}_2$ needs no modifications.

If multiple information equivalence exists, an alternative operation is to randomly take one variable from every such case to constitute a new $\boldsymbol{M}_2$, and other procedures of SEI remain unchanged. This idea may improve on the original SEI and thus LRH. Considering that this operation needs an extra computational complexity and that the occasions of multiple information equivalence are rare in practice, we discuss it no further.

Appendix 1: Definition of WMB

This appendix explains why we define WMB using Definition 3. For the definition of MB, it is easily verified that the following two statements are equivalent: (a) $\boldsymbol{M}$ is an Mb of T and none of its proper subsets is an Mb of T; and (b) $\boldsymbol{M}$ is an Mb of T and $T \not\perp\!\!\!\perp \boldsymbol{N}|\boldsymbol{M}\backslash\boldsymbol{N}$ holds for any nonempty $\boldsymbol{N} \subseteq \boldsymbol{M}$. However, replacing "Mb" with "WMb" in (a)(b), the resulting statements, say (a′) and (b′), are no longer equivalent: (a′) implies (b′) but not vice versa. Here is a counterexample.

Example 3 Consider a target variable T which has five potential features X, Y, Z_0, Z_1, and Z_2. Assume Z_0 carries all of (1) the unique information about T carried by X, all of (2) the unique information about T carried by Y, all of (3) the redundant information about T shared by X and Y, and some (but not all) of (4) the synergistic information about T carried jointly by X and Y; each of Z_1 and Z_2 carries some of (1), some of (2), some of (3), but neither Z_1 nor Z_2 contains (4); Z_1 and Z_2 jointly carry some (but not all) of (4), which is different from that Z_0 carries. Then, we have

$$\begin{aligned}
&T \perp\!\!\!\perp X|Z_0, \quad T \perp\!\!\!\perp Y|Z_0, \quad T \perp\!\!\!\perp X|\{Z_0, Z_1, Z_2\}, \quad T \perp\!\!\!\perp Y|\{Z_0, Z_1, Z_2\},\\
&T \perp\!\!\!\perp Z_1|Z_0, \quad T \perp\!\!\!\perp Z_2|Z_0, \quad T \not\perp\!\!\!\perp \{Z_1, Z_2\}|Z_0 \Rightarrow \begin{cases} T \not\perp\!\!\!\perp Z_1|\{Z_0, Z_2\}\\ T \not\perp\!\!\!\perp Z_2|\{Z_0, Z_1\}\end{cases}\\
&T \not\perp\!\!\!\perp Z_0|\{Z_1, Z_2\} \Rightarrow \begin{cases} T \not\perp\!\!\!\perp \{Z_0, Z_1\}|Z_2\\ T \not\perp\!\!\!\perp \{Z_0, Z_2\}|Z_1\end{cases}
\end{aligned}$$

It follows that:

- $\boldsymbol{M}_1 \triangleq \{Z_0\}$ is a WMb of T; none of its proper subsets is a WMb of T (i.e., $\varnothing$ is not a WMb of T); and $T \not\perp\!\!\!\perp \boldsymbol{N}|\boldsymbol{M}_1\backslash\boldsymbol{N}$ holds for any nonempty $\boldsymbol{N} \subseteq \boldsymbol{M}_1$.
- $\boldsymbol{M}_2 \triangleq \{Z_0, Z_1, Z_2\}$ is a WMb of T; $T \not\perp\!\!\!\perp \boldsymbol{N}|\boldsymbol{M}_2\backslash\boldsymbol{N}$ holds for any nonempty $\boldsymbol{N} \subseteq \boldsymbol{M}_2$; but its proper subset $\boldsymbol{M}_1$ is also a WMb of T, and none of the proper subsets of $\boldsymbol{M}_1$ is a WMb of T.
- $\mathbb{I}(T; \boldsymbol{M}_1) < \mathbb{I}(T; \boldsymbol{M}_2)$.

On the one hand, the implication, (a′) ⇒ (b′), means a "WMB" defined by (a′) is minimal in the sense of (b′); on the other hand, Example 3 means a WMB, $\boldsymbol{M}$, defined by (b′) may have a proper subset satisfying (a′) and this proper subset contains less information about T than $\boldsymbol{M}$. In other words, a WMB defined by (b′) can carry more information about T than a "WMB" defined by (a′). Recall that an MB is in the sense that (1) it is a refined Mb and it contains no any redundant variable, and (2) no information is lost when refining it. This motivates us to define the notion of WMB in a similar fashion: if $\boldsymbol{M}$ is a WMb and its subset, $\boldsymbol{N}$, is a WMB, then $\boldsymbol{N}$ should contain no redundant variables and should have the same mutual information with T as $\boldsymbol{M}$. By the proof of Theorem 2, we get $\mathbb{I}(T; \boldsymbol{M}\backslash\boldsymbol{N}|\boldsymbol{N}) = 0$, if using (b′) (and thus Definition 3) to define $\boldsymbol{N}$. This indicates

$$\mathbb{I}(T; \boldsymbol{M}) = \mathbb{I}(T; \boldsymbol{N}) + \mathbb{I}(T; \boldsymbol{M}\backslash\boldsymbol{N}|\boldsymbol{N}) = \mathbb{I}(T; \boldsymbol{N}),$$

so no information is lost. In comparison, Example 3 illustrates that $\boldsymbol{N}$ may lose some information if it is defined by (a′). This explains why we define the notion of WMB using Definition 3 based on (b′) instead of a definition based on (a′).

Appendix 2: Acronyms

AUC the area under ROC curve.

BN Bayesian network.

BW the backward function for the shrinking phase of LCMB algorithms.

CI conditional independence.

CMb candidate Markov blanket; see, e.g., Line 5 in the grwoing phase of IAMB in Algorithm 1.

CMI conditional mutual information.

DAG directed acyclic graph.

FW the forward function for the growing phase of LCMB algorithms.

GLL generalized local learning: an algorithmic framework for local causal discovery and feature selection (Aliferis et al. 2010a).

GS grow-shrink algorithm (Margaritis and Thrun 1999, 2000).

HITON an MB discovery algorithm, pronounced hee-tón, from the Greek Xιτών, for "cover", "cloak", or "blanket" (Aliferis et al. 2003).

IAMB incremental association Markov boundary algorithm (Tsamardinos et al. 2003a). See Algorithm 1 for details.

InterPostBW an interleaved version of PostBW. Algorithm 2 describes its pseudo code.

ImpWMB a sub-routine of WLCMB. See Algorithm 3 for details.

KIAMB a stochastic variant of IAMB (Peña et al. 2007); Algorithm 1 gives its pseudo codes.

KS Koller–Sahami algorithm (Koller and Sahami 1996).

LCMB an algorithmic framework containing those MB algorithms that are correct under local composition or Markov local composition (Definition 4).

LRH our proposed algorithm, which is used to deal with the problem of (P1). See Algorithm 1 for details. This algorithm can lessen swamping, resist masking, and highlight the true positives.

Mb Markov blanket:we call $\boldsymbol{M}$ an Mb of $\boldsymbol{T}$ if $\boldsymbol{T} \perp\!\!\!\perp \boldsymbol{V}\setminus\boldsymbol{M}\setminus\boldsymbol{T} \mid \boldsymbol{M}$ (Definition 1).

MB Markov boundary: an MB of $\boldsymbol{T}$ is any Mb such that none of its proper subsets is an Mb of $\boldsymbol{T}$ (Definition 1).

PCMB parents and children based Markov boundary algorithm (Peña et al. 2007).

PostBW a post-processing techiniue used to improve BW. See Algorithm 2 for details

RT running time: the single CPU time implemented on an Intel i7-3612QM 2.1 GHz and Windows 7 with 64 bits.

RE relative efficiency: defined by (3).

SEI the key sub-routine of LRH: selection-exclusion-inclusion.

WLCMB our LCMB-based algorithmic framework, which is used to deal with the problem of (P2). Algorithm 3 describes the pseudo code. Each WLCMB algorithm is correct under the weak Markov local composition assumption.

WIAMB an instantiation of WLCMB, with $\mathbb{A}$ taking $\langle$IAMB$\rangle$.

WKIAMB an instantiation of WLCMB, with $\mathbb{A}$ taking $\langle$KIAMB, $K\rangle$.

WLRH an instantiation of WLCMB, with $\mathbb{A}$ taking $\langle$LRH, $k\rangle$.

WMb weak Markov blanket: we call $\boldsymbol{M}$ a WMb of T if $\boldsymbol{T} \perp\!\!\!\perp X \mid \boldsymbol{M}$ holds for any $X \in \boldsymbol{V}\setminus\boldsymbol{M}\setminus\{T\}$ (Definition 3).

WMB weak Markov boundary: a WMB of T is any WMb such that $\boldsymbol{T} \not\perp\!\!\!\perp \boldsymbol{N}|\boldsymbol{M}\setminus\boldsymbol{N}$ holds for any nonempty $\boldsymbol{N} \subseteq \boldsymbol{M}$; see Definition 3 and "Appendix 1" for details.

References

Aliferis, C. F., Tsamardinos, I., & Statnikov, A. (2003). Hiton: A novel Markov blanket algorithm for optimal variable selection. In *AMIA 2003 annual symposium proceedings* (pp. 21–25). American Medical Informatics Association.

Aliferis, C. F., Statnikov, A., Tsamardinos, I., Mani, S., & Koutsoukos, X. D. (2010a). Local causal and Markov blanket induction for causal discovery and feature selection for classification part I: Algorithms and empirical evaluation. *Journal of Machine Learning Research*, *11*, 171–234.

Aliferis, C. F., Statnikov, A., Tsamardinos, I., Mani, S., & Koutsoukos, X. D. (2010b). Local causal and Markov blanket induction for causal discovery and feature selection for classification part II: Analysis and extensions. *Journal of Machine Learning Research*, *11*, 235–284.

Beinlich, I. A., Suermondt, H. J., Chavez, R. M., & Cooper, G. F. (1989). The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks. In *Second European conference on artificial intelligence in medicine* (pp. 247–256). London: Springer.

Ben-Gal, I. (2005). Outlier detection. In O. Maimon, L. Rokach (Eds.), *Data mining and knowledge discovery handbook* (pp. 131–146). New York: Springer US. ISBN 978-0-387-24435-8. doi:10.1007/0-387-25465-X_7.

Bromberg, F., & Margaritis, D. (2009). Improving the reliability of causal discovery from small data sets using argumentation. *Journal of Machine Learning Research*, *10*, 301–340.

Cheng, J., Greiner, R., Kelly, J., Bell, D., & Liu, W. (2002). Learning Bayesian networks from data: An information-theory based approach. *Artificial Intelligence*, *137*(1–2), 43–90.

Cochran, W. G. (1954). Some methods for strengthening the common χ^2 tests. *Biometrics*, *10*(4), 417–451.

Cover, T. M., & Thomas, J. A. (2006). *Elements of information theory* (2nd ed.). Hoboken: Wiley.

de Campos, L. M. (2006). A scoring function for learning Bayesian networks based on mutual information and conditional independence tests. *Journal of Machine Learning Research*, *7*, 2149–2187.

Fu, S., & Desmarais, M. (2007). Local learning algorithm for Markov blanket discovery. In *AI 2007: Advances in artificial intelligence* (pp. 68–79). Berlin Heidelberg: Springer.

Fu, S., & Desmarais, M. C. (2010). Markov blanket based feature selection: A review of past decade. In *Proceedings of the world congress on engineering*.

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, *3*, 1157–1182.

Hadi, A. S., Rahmatullah, I. A. H. M., & Mark, W. (2009). Detection of outliers. *Wiley Interdisciplinary Reviews Computational Statistics*, *1*(1), 57–70.

Koller, D., & Sahami, M. (1996). Toward optimal feature selection. In *Thirteen international conference in machine learning*. Stanford InfoLab.

Lee, C.-P., Leu, Y., & Yang, W.-N. (2012). Constructing gene regulatory networks from microarray data using GA/PSO with DTW. *Applied Soft Computing*, *12*(3), 1115–1124.

Lemeire, J. (2007). *Learning causal models of multivariate systems and the value of it for the performance modeling of computer programs*. ASP/VUBPRESS/UPA, PhD thesis.

Lemeire, J., Meganck, S., Cartella, F., & Liu, T. (2012). Conservative independence-based causal structure learning in absence of adjacency faithfulness. *International Journal of Approximate Reasoning*, *53*(9), 1305–1325.

Margaritis, D., & Thrun, S. (1999). Bayesian network induction via local neighborhoods. Technical Report CMU-CS-99-134, Carnegie Mellon University.

Margaritis, D., & Thrun, S. (2000). Bayesian network induction via local neighborhoods. In *Advances in neural information processing systems*, vol 12 (pp. 505–511). Morgan Kaufmann.

Martínez-Rodríguez, A. M., May, J. H., & Vargas, L. G. (2008). An optimization-based approach for the design of Bayesian networks. *Mathematical and Computer Modelling*, *48*(7–8), 1265–1278.

Masegosa, A. R., & Moral, S. (2012). A Bayesian stochastic search method for discovering Markov boundaries. *Knowledge-Based Systems*, *35*, 211–223.

Neapolitan, R. E. (2004). *Learning bayesian networks*. Upper Saddle River: Prentice Hall.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. San Francisco: Morgan Kaufmann.

Pellet, J.-P., & Elisseeff, A. (2008). Using Markov blankets for causal structure learning. *Journal of Machine Learning Research*, *9*, 1295–1342.

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

Rauh, J., Bertschinger, N., Olbrich, E., & Jost, J. (2014). Reconsidering unique information: Towards a multivariate information decomposition. In *IEEE international symposium on information theory (ISIT)* (pp. 2232–2236). IEEE.

Schlüter, F. (2014). A survey on independence-based Markov networks learning. *Artificial Intelligence Review*, *42*, 1069–1093.

Statnikov, A., Lytkin, N. I., Lemeire, J., & Aliferis, C. F. (2013). Algorithms for discovery of multiple Markov boundaries. *Journal of Machine Learning Research*, *14*(1), 499–566.

Tsamardinos, I., & Aliferis, C. F. (2003). Towards principled feature selection: Relevancy, filters and wrappers. In: *Proceedings of the ninth international workshop on artificial intelligence and statistics*.

Tsamardinos, I., Aliferis, C. F., & Statnikov, A. (2003a). Algorithms for large scale Markov blanket discovery. In: *Proceedings of the sixteenth international Florida artificial intelligence research society conference (FLAIRS)* (pp. 376–381).

Tsamardinos, I., Aliferis, C. F., & Statnikov, A. (2003b). Time and sample efficient discovery of Markov blankets and direct causal relations. In: *Proceedings of the ninth ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 673–678).

Tsamardinos, I., Brown, L. E., & Aliferis, C. F. (2006). The max–min hill-climbing Bayesian network structure learning algorithm. *Machine learning*, *65*(1), 31–78.

Williams, P. L., & Beer, R. D. (2010). Nonnegative decomposition of multivariate information. arXiv preprint arXiv:1004.2515.

Yaramakala, S. (2004). Fast Markov Blanket Discovery. MS thesis.

Yaramakala, S., & Margaritis, D. (2005). Speculative Markov blanket discovery for optimal feature selection. In *Proceedings of the fifth IEEE international conference on data mining*.

Zhang, L., & Guo, H. (2006). *Introduction to Bayesian networks*. Beijing: Science Press.

Zhang, Y., Zhang, Z., Liu, K., & Qian, G. (2010). An improved IAMB algorithm for Markov blanket discovery. *Journal of Computers*, *5*, 1755–1761.