Efficient Markov Blanket Discovery and Its Application

Tian Gao, *Student Member, IEEE*, and Qiang Ji, *Fellow, IEEE*

***Abstract*—In a Bayesian network (BN), a target node is independent of all other nodes given its Markov blanket (MB), and finding the MB has many applications, including feature selection and BN structure learning. We propose a new MB discovery algorithm, simultaneous MB (STMB), to improve the efficiency of the existing topology-based MB discovery algorithms. The proposed method removes the necessity of enforcing the symmetry constraint that is prevalent in existing algorithms, by exploiting the coexisting property between spouses and descendants of the target node. Since STMB mainly reduces the number of independence tests needed to complete the MB set after finding the parents-and-children set, it is applicable to all previous topology-based methods. STMB is both sound and complete. Experiments show that STMB has a comparable accuracy but much better efficiency than state-of-the-art methods. An application on benchmark feature selection datasets further demonstrates the excellent performance of STMB.**

***Index Terms*—Bayesian network (BN), feature selection, local structure learning, Markov blanket (MB).**

I. Introduction

THE MARKOV blanket (MB) was first termed by Pearl [1], and represents a crucial concept in a Bayesian network (BN). From the graph-theoretic point of view, the MB of any node in a BN consists of the node's parents, children, and spouses (i.e., the other parents of their common children). Collectively, the MB in a BN has a unique and valuable property: given the MB of a target node, all other nodes are independent of the target node. This means that the conditional probabilistic distribution of the target node given all other variables is equal to the conditional probabilistic distribution of the target node given only its MB nodes. The smallest MB that consists of the above property is called the minimal MB. MB discovery is the process used to find the minimal MB. We will use MB to represent the minimal MB, since only the minimal MB interests us.

Many principled solutions have been proposed to discover the MB. They can be roughly divided into two main types of approaches: 1) nontopology-based[1] and 2) topology-based. Overall, both approaches are heuristic-based and approximate the globally optimal MB set using a satisfactory set, due to the lack of a both time-efficient and data-efficient algorithm. In this paper, we propose a new topology-based MB discovery algorithm, simultaneous MB (STMB), built on the same assumptions and framework as previous topology-based methods. Compared to existing topology methods, STMB avoids the costly step of the symmetry enforcement that requires finding the PC sets of all target node's parents and children nodes, and thus achieves far better efficiency. We also study the performance of the STMB under assumption violations compared to other MB methods. We empirically demonstrate STMB's effectiveness on synthetic datasets, standard MB discovery datasets, and feature selection datasets.

II. Related Work

Nontopology-based MB discovery methods finds the MB by greedily testing independence relationships between each variable and the target variable. The very first work [2] that aimed to directly discover the MB proposed a nontopology-based method, later coined as Koller–Sahami algorithm (KS). KS minimizes the cross-entropy loss using a backward variable elimination process and requires two predefined parameters that sacrifice accuracy for a lower complexity: the predicted MB size and the maximum allowable size of the conditioned set. Since then, many other nontopology-based methods have been proposed to improve on the KS algorithm. Margaritis and Thrun [3] introduced the growth and shrink algorithm (GS) to use independence tests (ITs) and mutual information as criteria to select variables. GS algorithm first orders all the random variables univariately in an ascending order of the mutual information with the target variable, and then follows this order to sequentially test and add variables to the MB set during the growth stage. The shrinking stage then removes false positive nodes from the obtained MB. Incremental association MB (IAMB) [4] improves GS by reordering the variables each time the MB set changes. This reduces the number of false positives and improves the accuracy considerably. Since then, many variations of IAMB have been proposed such as inter-IAMB, IAMBnPC, Fast-IAMB [5], and KIAMB [6]. However, IAMB and its variants are not data-efficient [6]. If the sample size is small compared to the variable size, the performance of IAMB may suffer.

Manuscript received March 22, 2015; revised July 1, 2015; accepted February 22, 2016. Date of publication March 24, 2016; date of current version April 13, 2017. This work was supported by the National Science Foundation (NSF) under Grant IIS 1145152. This paper was recommended by Associate Editor Y. Xiang.

The authors are with the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY 12180-3590 USA (e-mail: gaot@rpi.edu; jiq@rpi.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2016.2539338

[1] Also called greedy methods in the literature.

Discovery methods with a restriction on MB parameterization were also proposed [7].

Due to the large amount of data required in nontopology-based methods and the limited data availability for many real world applications such as biological data, topology-based methods aim to tackle the data efficiency while maintaining a reasonable time complexity. Min–max MB (MMMB) [8] improves the data efficiency by making the sample requirement dependent on the structure topology instead of the size of the conditioned set. MMMB proposes to find the parents-and-children (PC) set first and then complete the MB by finding the spouses. Although MMMB was later found unsound [6], it still introduced a solid direction to pursue, which followed later methods. Recent topology-based methods typically include a symmetry check step [9] to correct the faulty PC set. HITON-MB[2] directly uses the MMMB framework and tries to remove false positives in the PC set as early as possible by interweaving the addition and removal process, and hence, reduces the number of ITs needed. Parents children-based MB algorithm (PCMB) [6] is the first proven sound topology algorithm and checks for collider nodes only in the PC set of the target during the MB completion step. Experiments on small datasets have shown the superiority of PCMB over nontopology-based methods. Iterative parent–child-based search of MB (IPCMB) [12] is based on PCMB and it uses a more efficient method to search for the PC set but still utilizes the symmetry check. A unified framework of topology-based methods is summarized [10], [13] and through extensive experimental studies it confirms the superior performance of topology-based methods in various applications.

By finding the MB, we can solve many problems directly or indirectly, for example, feature selection [14] and BN structure learning. Feature selection is a dimension-reduction technique widely used in all kinds of machine learning problems. To reduce the intractability and prune out some irrelevant or redundant features, feature selection is often used to select a few most "useful" ones while minimizing some "information loss." Many existing algorithms seek to use different criteria to represent such loss, such as the statistics of features (filtered methods, see [15], [16]), accuracy rates of some classifiers (wrapper methods), and some classifier specific objective functions (embedded methods, see [17]–[19]). However, most of these work lack a theoretical justification on the optimality of their feature selection criteria, while feature selection using MB has been shown to be theoretically optimal [2], [10]. Experiment studies [10], [13] showed the superior performance of MB methods over other methods for feature selection. Moreover, MB algorithms can be used for causal feature selection and casual discovery. Different from traditional feature selection, causal feature selection can explain the underlying causal mechanisms of the selected features, distinguish between actual relevant features and experimental artifacts, and lead to prediction of actions by external agents [20]. Studies [21] also show the superior performance of the MB feature set over other feature sets in the application of casual versus noncausal feature selection.

BN structure learning can also benefit from the MB discovery. Because BN structure learning is generally NP-hard [22], many methods seek to reduce the complexity by restricting its parameterization [23], structure [24], or both [25]. Other methods use prior knowledge and approximate learning methods to reduce the complexity [26]. As MB represents local structures in a BN, MB discovery can be seen as a subproblem of BN structure learning as we are interested in the local structure of the network with respect to one node instead of the entire network. It is practical because often only the local structure is desired in tasks like feature selection and causal discovery. If one is interested in the global structure, several works [3], [9], [27] have proposed algorithms to first identify each node's MBs, and then connect them in a maximally consistent way to infer the global BN structure. By doing such a local-to-global approach, the general BN structure can be learned exactly without any abovementioned restriction [23]–[25] on its parameterization or structure. In addition, this exact local-to-global approach can learn a much larger network than some of the existing exact global BN structure learning algorithms [28], [29]. In summary, finding the MB could solve many realistic and important applications.

The rest of this paper is organized as follows. Section III reviews some concepts related to the MB discovery. Section IV presents and analyzes our new algorithm. We show the experimental performance of the proposed method in Section V. Section VI concludes this paper with a discussion of potential future research directions.

III. Background

Let $\mathbf{V}$ denote a set of random variables. A BN for $\mathbf{V}$ is represented by a pair (G, θ). The network structure G is a directed acyclic graph (DAG) with nodes corresponding to the random variables in $\mathbf{V}$ and edges capturing the dependencies between the connected nodes. If a directed edge exists from node X to node Y, X is a parent of Y and Y is a child of X. If X is either a parent or a child of Y, X and Y are neighbors and adjacent to each other [1]. The parameters θ indicate the conditional probability distribution of each node $X \in \mathbf{V}$ given its parents. Moreover, let a path between two nodes X and Y in G be any sequence of nodes between them such that any successive nodes are connected by a directed edge, and no node appears in the sequence twice. A directed path of a DAG is a path with nodes $(V_1, \ldots, V_n)$ such that, for $1 \leq i < n$, V_i is a parent of V_{i+1}. If there is a directed path from X to Y, then X is an ancestor of Y and Y is a descendant of X. If X and Y have a common child and they are not adjacent, X and Y are spouses of each other. For the rest of this paper, we use capital letters (such as X, Y) to represent variables, small letters (such as x, y) to represent values of variables, bold letters (such as $\mathbf{V}, \mathbf{MB}$) to represent variable sets, and use $|\mathbf{V}|$ to represent the size of a set $\mathbf{V}$. We also use $X \perp\!\!\!\perp Y$ and $X \not\perp\!\!\!\perp Y$ to represent independence and dependence between X and Y, respectively. X is independent of Y if $P_{XY}(x, y) = P_X(x)P_Y(y)$,

[2] HITON comes from a Greek word meaning "cloak" [10].

and X is conditionally independent of Y given some set Z if $P_{XY|Z}(x, y|z) = P_{X|Z}(x|z)P_{Y|Z}(y|z)$.

Definition 1 (Markov Condition [1]): A node in a BN is independent of its nondescendant nodes, given its parents.

Markov condition enables the efficient representation and parameterization of random variables in a BN.

Definition 2 (Faithfulness Condition [30]): A BN G and a joint distribution P are faithful to each other if and only if all and only the conditional independencies true in P are entailed by G.

Definition 3 (V-Structure [1]): Three nodes X, Y, and Z form a V-structure if node Y has two incoming edges from X and Z, forming $X \rightarrow Y \leftarrow Z$, and X is not adjacent to Z.

Y is a collider if Y has two incoming edges from X and Z in a path, whether X and Z are adjacent or not. Y with nonadjacent parents X and Z is an unshielded collider for the path X to Z.

Definition 4 (Blocked Path [1]): A path J from node X and Y is blocked by a set of nodes $\mathbf{Z}$, if any of the following holds true: 1) there is a noncollider node in J belonging to $\mathbf{Z}$ and 2) there is a collider node C on J such that neither C nor any of its descendants belong to Z. Otherwise, J from X and Y is unblocked or active.

Definition 5 (d-Separation [1]): Two nodes X and Y are d-separated by a set of nodes $\mathbf{Z}$ if and only if every path from X to Y is blocked by $\mathbf{Z}$.

Such a set $\mathbf{Z}$ would be called a sepset of X from Y, denoted as $\text{Sep}_Y\{X\}$. Therefore, by performing ITs and identifying d-separation relationships among random variables, the entire graph structure can be inferred. Theorem 1 justifies the soundness of common independence-test-based MB discovery methods.

Theorem 1 [30]: If a BN G is faithful to a joint probability distribution P, then: 1) node X and Y are adjacent in G if and only if X and Y are dependent given every set of nodes that does not include X and Y and 2) for nodes X, Y, and Z in G, if X and Y are adjacent, Y is adjacent to Z, and Z is not adjacent to X, they form a V-structure with Y as a collider node if and only if $X \not\perp\!\!\!\perp Z|\boldsymbol{S}$, $\forall \boldsymbol{S}$ such that $X, Z \notin \boldsymbol{S}$ and $Y \in \boldsymbol{S}$.

Next, we introduce specific concepts related to the MB discovery.

Definition 6 (Markov Blanket [1]): An MB of a target variable T, $\mathbf{MB}_T$, is a set of nodes conditioned on which all other nodes are independent of T, denoted as $X \perp\!\!\!\perp T|\mathbf{MB}_T, \forall X \subseteq \mathbf{V}\backslash\{T\}\backslash\mathbf{MB}_T$. $\mathbf{MB}_T$ is minimal if none of its proper subsets satisfies the above property.

We will refer to the MB as the minimal MB[3] here since only the minimal MB is of interest.

Given an unknown distribution P that satisfies the Markov condition with respect to an unknown DAG G, MB discovery is the process used to estimate the MB of a target node from independently and identically distributed data samples D of P. Assuming the faithfulness condition holds and ITs correctly reflect independence, the MB of a target node is uniquely identifiable.

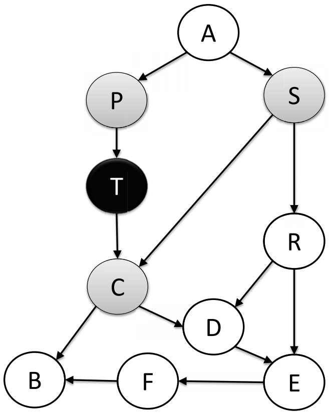

Fig. 1. Sample BN. Black node T is the target node and the shaded nodes are the MB of T.

Theorem 2 (MB Uniqueness [1]): If a BN G and a distribution P are faithful to each other, then $\mathbf{MB}_T$, $T \in \boldsymbol{V}$, is unique and is the set of parents, children, and spouses of T. In addition, the parents and children set of T, $\mathbf{PC}_T$, is also unique.

For example, in Fig. 1, nodes P and C form $\mathbf{PC}_T$, adjacent to T. $\mathbf{MB}_T$ consists of its parent node P, its child node C, and its spouse S. All other nodes A, B, R, D, E, and F are independent of T, given $\mathbf{MB}_T$, due to blocked paths. C, D, and E are collider nodes for path $T - C - S$, $C - D - R$, and $D - E - R$, respectively.

Violating the faithfulness assumption would potentially invalidate Theorem 2 and introduce multiple sets of $\mathbf{MB_T}$, which is not in the scope of current topology-based methods. Nevertheless, through empirical study, we offer some informal discussion on the effects of assumption violations in Section IV-E.

Last, one of the main concepts in the topology-based MB algorithms is the symmetry constraint.

Proposition 1 (Symmetry Constraint): For a node X to be a parent or child of T, both of the following statements must hold true: X must be in the PC set of T and T must be in the PC set of X, i.e., $X \in \mathbf{PC}_T$ and $T \in \mathbf{PC}_X$.

All of the existing topology-based algorithms employ the symmetry constraint to remove those false positive PC nodes in the returned PC set. In Fig. 1, for example, using topology-based methods, node D would be in the returned PC set of T due to $D \not\perp\!\!\!\perp T|\mathbf{Z}, \forall \mathbf{Z} \subseteq \{P, C\}$. D would be removed from the returned PC set using the symmetry constraint. In some MB related works [31], [32], a variant of the symmetry constraint uses the OR-rule instead of the AND-rule: if $X \in \mathbf{PC_T}$ or $T \in \mathbf{PC_X}$, then X is a PC node of T.

IV. MB Discovery Algorithm

Before introducing the proposed method, we would like to briefly review the existing MB discovery algorithms.

A. Existing Algorithms

In topology-based MB algorithms, methods such as PCMB and IPCMB would first find the PC set of a target node, and

[3] The minimal MB is also called Markov boundary in some literature.

Algorithm 1 General Framework of IPCMB and PCMB

```
Input: Data, $D$; target node, $T$
{step 1: find the PC set }
$\mathbf{PC}_T \leftarrow \mathbf{V} \setminus \{T\}$;
for $i = 0$ to $|\mathbf{PC}_T|$ do
    for all $\mathbf{Z} \subseteq \mathbf{PC}_T$ with $|\mathbf{Z}| = i$ do
        if $X \perp\!\!\!\perp T|\mathbf{Z}$, $\forall X \in \mathbf{PC}_T$, then
            $\mathbf{PC}_T \leftarrow \mathbf{PC}_T \setminus \{X\}$;
            $\mathbf{Sep}_T\{X\} \leftarrow \mathbf{Z}$;
        end if
    end for
end for
{step 2: enforce the symmetry constraint}
Find the PC sets of $X$, $\mathbf{PC}_X$, $\forall X \in \mathbf{PC}_T$;
for each $X \in \mathbf{PC}_T$ do
    if $T \notin \mathbf{PC}_X$ then
        $\mathbf{PC}_T \leftarrow \mathbf{PC}_T \setminus \{X\}$
    end if
end for
{step 3: find the spouses}
find $\mathbf{H}'$: the PC nodes of every node in $\mathbf{PC}_T$
$\mathbf{S}_T \leftarrow \emptyset$;
for each $X \in \mathbf{H}'$ do
    if $\exists Y \in \mathbf{PC}_T$ s.t. $X \not\perp\!\!\!\perp T|\mathbf{Sep}_T\{X\} \cup \{Y\}$ then
        $\mathbf{S}_T \leftarrow \mathbf{S}_T \cup \{X\}$;
    end if
end for
Return: $MB \leftarrow \mathbf{PC}_T \cup \mathbf{S}_T$
```

then enforce the symmetry constraint to remove false positives in the PC set. Given the PC, existing algorithms then look for spouses to complete MB. A generalized framework of PCMB and IPCMB is shown in Algorithm 1, with three steps: 1) find the PC set using exhaustive search; 2) enforce the symmetry constraint to remove false positive nodes in the PC set; and 3) then find the spouses to complete the MB. In the first step, starting with the entire variable set as the potential PC set for T, if there exists a set $\mathbf{Z}$ such that $X \perp\!\!\!\perp T|\mathbf{Z}$, X would be removed from the PC set of T. The search of the set $\mathbf{Z}$ will increase from $|\mathbf{Z}| = 0$ to the largest possible size. In the second step, enforcing the symmetry constraint is necessary to remove false positives in the found PC set [6].

B. Proposed Improvement

Finding the PC set in the MB discovery algorithms is the most computationally expensive step due to the exhaustive search, and the procedure to enforce the symmetry constraint would be $|\mathbf{PC}|$ times more costly. Unfortunately, currently there is no alternative to the symmetry constraint. Motivated to reduce such a performance bottleneck, we propose a method to avoid the expensive symmetry check step and yet can still remove false positive PCs by combining the last two steps of Algorithm 1. The insight comes from the composition of the returned PC set in step 1 of the existing algorithms.

Proposition 2 (False Positives in the PC Set Search): In the existing topology-based methods,[4] under the faithfulness assumption, the found parents and children set, $\mathbf{PC}^f$, is a union between the true parents and children set, $\mathbf{PC}^t$, and some false positives $\boldsymbol{F}$, i.e., $\mathbf{PC}^f = \mathbf{PC}^t \cup \boldsymbol{F}$, where $\boldsymbol{F}$ might be nonempty.

Proof: First, we show that $\mathbf{PC}^f$ contains all the true positives and no false negatives, i.e., $\mathbf{PC}^t \subseteq \mathbf{PC}^f$. Let X be some true positives in the entire search set $\mathbf{V}$, then there exists no set $\mathbf{Z} \subseteq \mathbf{V}$ that d-separates X from T, since X is a true PC of T and they are adjacent in the graph. Thus, according to the first part of Theorem 1, $X \not\perp\!\!\!\perp T|\mathbf{Z}$ will always hold true, given the faithfulness assumption. As a result, X will never be removed. Therefore, $\mathbf{PC}^t \subseteq \mathbf{PC}^f$. Second, we prove by contradiction that there may be some false positives entering $\mathbf{PC}^f$. Let us assume that false positives were never in $\mathbf{PC}^f$. Consider the case in Fig. 1, given or not given node C, node D is dependent on T. Because S and R both have a sepset A of size 1, and the smallest sepset of D is either $\{C\} \cup \{S\}$ or $\{C\} \cup \{R\}$ of size 2, node S and R will be removed from $\mathbf{PC}_T$ earlier than D as shown in Section IV-A, and as a result D will stay in $\mathbf{PC}^f$ as no sepsets of D exist anymore in $\mathbf{PC}^f$. However, D is neither a parent or child of T, contradicting the assumption. Thus, there will be some false positives in $\mathbf{PC}^f$. ■

Proposition 2 formalizes a previous speculation [6] that descendants may exist as false positives and precisely defines the domain of the PC set. Building on Propositions 2 and 3 aims to identify the false positives.

Proposition 3 (PC False Positive Identity): False positives $\boldsymbol{F} \in \mathbf{PC}^f$ consist of only descendants of the target T, denoted as $\mathbf{Des}_T$.

Proof: Using Proposition 2, $\mathbf{PC}^f$ consists of the entire $\mathbf{PC}^t$ and some false positives $\mathbf{F}$. We show $\mathbf{F} \subseteq \mathbf{Des}_T$. Since $\mathbf{PC}^f$ is a super set of all true positive PCs, $\mathbf{PC}^f$ must contain the entire parent set of T, $\mathbf{Pa}_T$, due to exhaustive search for the PC set. By the Markov condition, all the nondescendant nodes are independent of T given $\mathbf{Pa}_T$. If $F \in \mathbf{F}$ is any nondescendant node, then $F \perp\!\!\!\perp T|\mathbf{Pa}_T$. Thus, simply by Markov condition, F would be removed from $\mathbf{PC}^f$. By contradiction, $\mathbf{F} \subseteq \mathbf{Des}_T$. ■

Given the identity of such false positives, if they exist, we have the following insight on how to remove false positives and find the spouse set at the same time.

Theorem 3 (Coexistence Between Spouses and Descendants): In the existing topology-based MB discovery algorithms, the only false positives in $\mathbf{PC}^f$ belong to the descendants of T, $\mathbf{Des}_T$, due to an unblocked path between T and its descendants with a V-structure $T \rightarrow$ child $\leftarrow$ spouse.

Proof: Propositions 2 and 3 show that only false positives in $\mathbf{PC}^f$, if they exist, are the descendants of T. Now we need to show the second part of the theorem is true. In the PC set search step, starting with $\mathbf{PC}^f = \mathbf{V}\backslash\{T\}$, if $\exists \mathbf{Z} \subseteq \mathbf{PC}^f$ such that $X \perp\!\!\!\perp T|\mathbf{Z}$, $X \in \{\mathbf{PC}^f \backslash \mathbf{Z}\}$, then X will be removed from $\mathbf{PC}^f$. Assuming false positives exist, let $F \in \mathbf{Des_T}$, F passes ITs and stays in $\mathbf{PC}^f$ because $F \not\perp\!\!\!\perp T|\mathbf{Z}, \forall \mathbf{Z} \subseteq \mathbf{PC}^f$.

[4] We define the existing topology-based methods to be MMMB, HITON, IPCMB, and PCMB exclusively.

Consider $\mathbf{Q} = \mathbf{PC}^f \backslash \{F\}$, for F to exist in $\mathbf{PC}^f$, $F \not\perp\!\!\!\perp T|\mathbf{Q}$ must be true. Since random variables in $\mathbf{PC}^t \subseteq \mathbf{Q}$ must be present in all paths from T to F by the definition of PC nodes, the dependence between T and F occurs only if $\mathbf{Q}$ unblocks some path from T to F. This can only happen when there is a collider node in $\mathbf{Q}$. Hence, the only way F can exist in $\mathbf{PC}^f$ is through an unblocked path that contains a V-structure $T \rightarrow$ child $\leftarrow$ spouse. ■

Theorem 3 shows that if there is a false positive node in $\mathbf{PC}^f$, then it must be due to a child collider node of T. The resulting V-structure implies T must have at least one spouse, which shows that there is a coexistent relationship between false positive nodes in $\mathbf{PC}^f$ and the spouses of T. In topology-based MB algorithms, given the PC set, the task left is to remove the false positives and then adding spouses back to MB. Existing methods separate them into two steps. Theorem 3 shows the possibility that we can accomplish both steps simultaneously, which would reduce the complexity.

C. Simultaneous Markov Blanket Discovery

We propose STMB to find the MB of a target node. Our algorithm, shown in Algorithm 2, has two steps: 1) the first step of STMB identifies the PC sets using the same step 1 as PCMB or IPCMB. We use RecogPC to represent step 1 of IPCMB (lines 2–10 of Algorithm 1) and 2) in step 2, STMB finds the spouse and removes the non-MB descendants from the PC set at the same time. Specifically, in step 2 of Algorithm 2, STMB looks for a node $Y \in \mathbf{PC}^f$ that unblocks a path from T to some node $X \in \mathbf{V} \backslash \mathbf{PC}^f$ (i.e., a candidate spouse set). If found such Y, X could be a spouse (line 13) and Y could a child or non-child descendant node. Notation-wise, $\mathbf{spouse}_T\{Y\}$ represents a subset of spouse nodes of T, $\mathbf{spouse}_T$, with corresponding child node Y. After checking at line 9, line 10 removes the found non-MB descendants that have one V-structure path to the target. Then starting at line 19, STMB tests for false positive spouses X (such as spouses' parents) by conditioning on other nodes that unblocked by each Y. If X and T are independent, X is removed from spouse candidate set (line 23). STMB then tests for other non-MB descendants X in the PC set that may have multiple paths to the target. If X and T are independent, X is removed from the PC set (line 30).

The soundness and completeness of STMB can be derived from the algorithm procedure.

Theorem 4 (Soundness and Completeness of STMB): Under the faithfulness assumption, Algorithm 2 finds all and only the MB nodes of the target node.

Algorithm 2 STMB Algorithm

```
1:  Input: Data, $D$; target node, $T$
2:  $\mathbf{PC}_T \leftarrow \mathbf{V} \setminus \{T\}$;
    {step 1: find the PC set }
3:  $[\mathbf{PC}_T, \mathbf{Sep}_T] \leftarrow$ RecogPC($T$, $\mathbf{PC}_T$, $D$);
    {step 2: find spouses and remove non-child descendants}
4:  $\mathbf{spouse}_T \leftarrow \emptyset$;
5:  $\mathbf{remove} \leftarrow \emptyset$;
6:  for each $Y \in \mathbf{PC}_T$ do
7:      for each $X \in \{\mathbf{V} \setminus \{T\} \setminus \mathbf{PC}_T\}$ do
8:          if $X \not\perp\!\!\!\perp T|\mathbf{Sep}_T\{X\} \cup \{Y\}$ then
9:              if $Y \perp\!\!\!\perp T|\mathbf{Z}$, $\exists \mathbf{Z} \subseteq \mathbf{PC}_T \cup \{X\} \setminus \{Y\}$ then
10:                 $\mathbf{remove} \leftarrow \mathbf{remove} \cup \{Y\}$;
11:                 break;
12:             else
13:                 $\mathbf{spouse}_T\{Y\} \leftarrow \mathbf{spouse}_T\{Y\} \cup \{X\}$;
14:             end if
15:         end if
16:     end for
17: end for
18: $\mathbf{PC}_T \leftarrow \mathbf{PC}_T \setminus \mathbf{remove}$;
19: for each $Y$ in $\mathbf{spouse}_T$ do
20:     for each $X$ in nonempty $\mathbf{spouse}_T\{Y\}$ do
21:         $\mathbf{testSet} \leftarrow \mathbf{PC}_T \cup \mathbf{spouse}_T\{Y\} \setminus \{X\}$;
22:         if $X \perp\!\!\!\perp T|\mathbf{testSet}$ then
23:             remove $X$ from $\mathbf{spouse}_T\{Y\}$;
24:         end if
25:     end for
26: end for
27: $\mathbf{M} \leftarrow \mathbf{PC}_T$;
28: for each $X \in \mathbf{M}$ do
29:     if $X \perp\!\!\!\perp T|\mathbf{PC}_T \cup \mathbf{spouse}_T \setminus \{X\}$ then
30:         $\mathbf{PC}_T \leftarrow \mathbf{PC}_T \setminus \{X\}$;
31:     end if
32: end for
33: $\mathbf{MB} \leftarrow \mathbf{spouse}_T \cup \mathbf{PC}_T$;
```

Proof: First, we show STMB is complete, i.e., it finds all the true positive MB nodes. Starting with the entire variable set, by using the same procedure as IPCMB, step 1 of Algorithm 2 returns $\mathbf{PC}_T$ containing all the true positive PCs and some descendants of the target node [12]. Since the true positive PCs are always dependent of T, $\mathbf{PC}_T$ always contains all the true positive PCs. After the detection of $\mathbf{PC}_T$, STMB looks for spouse candidate X with a collider node Y (lines 6–8). Because a node may be one of multiple identities, upon finding X, STMB uses Theorem 3 to check if Y is a corresponding child or non-child descendant node. If Y becomes conditionally independent of T (line 9), STMB removes Y from $\mathbf{PC}_T$ (line 10); if no non-child descendants are found, then X is a candidate spouse via a child node Y (line 12), since any newly-found conditional dependence would indicate the existence of a V-structure. Due to exhaustive search, we would not miss any true positive spouse $X \in \mathbf{V} \backslash \{T\} \backslash \mathbf{PC}_T$. On the other hand, $\mathbf{PC}_T$ may contain true positives spouses, if a non-child descendant node in $\mathbf{PC}_T$ is also a spouse candidate, such as in Fig. 1, to discover the MB of node C in Fig. 1, node E has multiple paths to node C unblocked by the PC set T, B, and D. Since line 8 checks only one conditioned variable X at a time, false positive nodes like E that has multiple unblocked paths would not be removed. Therefore, at line 18 $\mathbf{PC}_T$ and $\mathbf{spouse}_T$ sets together contain all the true positive spouses and PCs, thus all the true positive MB nodes. True positives PCs are always in $\mathbf{PC}_T$ by Theorem 1 and true positive spouses will not be removed at later steps (lines 23 and 30) because they are always dependent of T given the true positive PCs. Therefore, STMB is complete.

Second, we show STMB's soundness, i.e., STMB will remove false positive MB nodes only. False positives exist in two forms: 1) some of the unblocked nodes in $\mathbf{spouse}_T$ may be false positives, such as spouses' parents and 2) there are also non-MB descendants in $\mathbf{PC}_T$. Line 23 of STMB removes false positive spouses $\in \mathbf{spouse}_T$, since independence relationships with T given some PCs and spouses (i.e., a candidate set of MB) would indicate false MB nodes, directly using Definition 6. After line 26, we have only the true spouses in $\mathbf{spouse}_T$ as the exhaustive test ensures no false positive spouse left. We can use $\mathbf{spouse}_T$ to remove the non-MB descendants in $\mathbf{PC}_T$ using Theorem 3. By conditioning on all of the true positive spouses in addition to the true positive PCs in the joint set $\mathbf{PC}_T$ and $\mathbf{spouse}_T$, non-MB descendants in $\mathbf{PC}_T$ will be removed (line 30). Therefore, in the end $\mathbf{PC}_T$ and $\mathbf{spouse}_T$ sets contain all and only the true positive PCs and spouses. Their union forms the true MB. Hence, STMB is sound. ■

D. STMB Computational Complexity

The STMB complexity is determined by the first step of finding the PC set, and the second step of removing false positives in the PC set and finding the true spouses. The computational cost of finding the PC set varies among different algorithms but they are all very expensive. In the worst case, PCMB finds the PC set in $O(P2^{N+1})$ ITs and IPCMB takes $O(P2^N)$ IT, where P is the largest size of conditioned sets during the PC set search and N is the total number of variables. In the second step of STMB, the complexity of finding candidate spouses (lines 6–17) is $O(C(N - C)$ IT and the rest of steps (lines 18–33) take $O(SK + C)$ IT, where C is the largest size of the PC sets of all the nodes, S is the number of spouses, and K is the largest size of all $\mathbf{spouse}_T\{y\}$. Overall, in our implementation using the PC set search of IPCMB, STMB takes $O(P2^N + C(N - C) + SK + C) = O(P2^N)$ IT in the worst case, predominated by the cost of the first step. The existing algorithm IPCMB shares the same first step cost, i.e., $O(P2^N)$ IT, but in their second step, enforcing the symmetry constraint repeats the first step C times and takes $O(CP2^N)$ IT. IPCMB then finds spouses in $O(CP2^{N-2})$ IT. Overall, IPCMB takes $O((C + 1)P \cdot 2^N + CP2^{N-2}) = O(CP2^N)$ IT. Therefore, IPCMB would cost C times more than STMB. In practice, the computational time depends on the structure of each dataset and both methods should be much faster than the worst case complexity. For larger and more densely connected networks, STMB can achieve more speedups.

E. Assumption Violation

Faithfulness is a standard assumption in BN learning algorithms. It is well known that the Lebesgue measure of the unfaithfulness distributions is zero [44], which indicates the probability of unfaithful parameterization is very low. Even when unfaithfulness distributions occur, the unfaithfulness relationships only consist of a small percent of the total independence and dependence relationships in the graph. As a result, the impact of unfaithful relationships on the overall performance should be minimal. Nevertheless, the faithfulness assumption may still be violated in practice, and we evaluate the effects of the potential faithfulness violation on both the proposed STMB and the traditional symmetry enforcement methods such as IPCMB. Some existing works focus on finding multiple Markov boundaries under the violations [33], but this is the first attempt to discuss the effects of faithfulness violation on MB discovery algorithms. Note that there are many existing works that provide formal and theoretical analysis of faithfulness violation in a BN [34]. Some BN structure learning algorithms are specifically designed to operate under faithfulness violation [35]–[38]. While we realize the importance of formally studying the performance of STMB under faithfulness violation using these unfaithful BN structure learning framework [34]–[38], such a study is, however, beyond the scope of the present work. We will investigate this issue in the future, and here we offer some informal discussion of STMB's performance under faithfulness violation.

We consider the cases only when data can be represented by a BN; otherwise the independence relationships cannot be captured by a BN fully, such as in the cases of multiple XOR gates [39]. Under the faithfulness violation where G does not have the same independence relationships with P, we can classify the differences into several categories, considering the discrepancies between the independence relationships of a distribution $\mathbf{P}$ embedded in the data $\mathbf{D}$ and the independence relationships of a structure $\mathbf{G}$ learned from the data $\mathbf{D}$.

1) Case 1 (Fewer Independence in $\mathbf{G}$ *Than* $\mathbf{P}$*):* We can further divide this case based on whether the adjacent nodes (which mainly affect the PC set in MB) or nonadjacent nodes (which mainly affect the spouse set in MB) in $\mathbf{G}$ contain fewer independence relationships.

1) *Fewer Independence Among Adjacent Nodes in* $\mathbf{G}$*:* $\mathbf{G}$ contains fewer independence, or more edges, among the adjacent nodes than those indicated in $\mathbf{P}$. This case cannot happen in STMB and IPCMB because every adjacent independence in $\mathbf{G}$ is learned directly from $\mathbf{P}$. The exhaustive search on the PC set ensures that no marginal independence in $\mathbf{P}$ is missed between adjacent nodes in $\mathbf{G}$.
2) *Fewer Independence Among Nonadjacent Nodes in* $\mathbf{G}$*:* $\mathbf{G}$ indicates dependence between 2 nonadjacent variables but $\mathbf{P}$ indicates their independence. For example, $\mathbf{G}$ is the structure in Fig. 1 as $P \not\!\perp\!\!\!\perp A$, $S \not\!\perp\!\!\!\perp A$, but $\mathbf{P}$ indicates $P \perp\!\!\!\perp S$ or $P \perp\!\!\!\perp S|R$ in addition to other independence.

   Effects: This case is possible, as the parameters in G can capture the additional independence in P. Nonadjacent variables affect the PC set search of STMB and IPCMB only when extra marginal dependence introduce non-child descendants in the PC set; both IPCMB and STMB could remove them during either the symmetry enforcement step or the 2nd step in Algorithm 2, respectively. Therefore, this case affects the discovery of spouses more. Since both methods test for the true spouses, fewer independence would result in more spouses than desired. These extra spouses would happen in both algorithms, but STMB could remove these false spouses with line 22 with a large conditioned set, while IPCMB cannot recover.

2) Case 2 (More Independence in **G** *Than* **P***):* We again divide this case based on whether the adjacent nodes or nonadjacent nodes in **G** contain more independence relationships.

1) *More Independence Among Adjacent Nodes in* **G***:* **G** contains no direct edges between two adjacent variables even though they are marginally dependent in **P**. For example, in Fig. 1, if $P \not\perp\!\!\!\perp A$, $P \perp\!\!\!\perp A|S$, and $P \perp\!\!\!\perp S|A$ in **P**, then the edge between P and A is removed following the MB algorithm procedures, thus $P \perp\!\!\!\perp A$ in **G**.

   Effects: **G** could contain more independence due to the greedy nature of the PC set search in both STMB and IPCMB algorithms. If they find one independence/conditional independence relationship between two variables, edges between them would be removed from **G**. This case of violation affects both algorithms as they share the same step 1 PC set search, but IPCMB could be more affected because, if the adjacent nodes of the target are found to be conditionally independent of the target in the symmetry enforcement step (which could happen due to the greedy nature), a correct edge would be removed despite the PC set search step contains it.

2) *More Independence Among Nonadjacent Nodes in* **G***:* **G** captures more independence among nonadjacent nodes than **P**. For example, in Fig. 1, $P \perp\!\!\!\perp S$ in **G** but not in **P**.

   Effects: This case is possible to happen if case 2(1) also happens. Extra independence among nonadjacent variables do not affect the PC set search result in any way. It may affect the non-child descendants in the PC set but it is actually beneficial to do so. In the spouse search step, this case may remove true spouses, which is likely to happen in both methods but STMB could be more affected due to extra function calls to remove false positive spouses.

3) Case 3 (Latent Nodes): When some variables are totally hidden, **G** can only be learned on a subset of variables with partial independence relationships from **P**. With missing variables, bidirected edges could emerge in order to capture the full independence relationship [22], [40].

Effects: With these bidirected edges, STMB and IPCMB could both find extra spouses in the DAG setting. Since these spouses are needed to reflect the correct independence relationships, the error will be preserved and thus STMB and IPCMB are both affected equally.

The above analysis seems to suggest that the violation of the faithfulness condition makes IPCMB more susceptible to mistakes than STMB. Also, since the size of a PC set is always bigger or equal to the size of a spouse set, the chance of errors occurring is actually higher for IPCMB using the symmetry constraint (which tends to make more mistakes in the PC set search), and hence STMB could possibly outperform IPCMB in term of the overall accuracy. We also empirically evaluate the performance of STMB against faithfulness violation in Section V-B. Further theoretical analysis in this direction would be an interesting future research direction.

In addition, violating the correct ITs, such as setting a wrong p-value threshold in ITs, could violate any of the above cases, changing the learned MB in both methods. Since the thresholds may vary from datasets to datasets, it is unavoidable to happen in practice. Better statistical ITs can always improve the results on both STMB and IPCMB.

V. Experiments

First, we evaluate the effectiveness and efficiency of the proposed STMB with other MB discovery algorithms on synthetic datasets and standard MB discovery datasets. We then evaluate STMB on real feature selection datasets. We use existing implementations as much as possible, namely Causal Explorer in MATLAB [41] and FEAST [42] while implementing the rest of algorithms ourselves. Since the IPCMB PC set searching is more efficient, we use the IPCMB version in our implementation of STMB. All codes are implemented in MATLAB with some C++ libraries to speed up the process. The experiments are conducted on a computer with 2.66 GHz CPU.

We also follow the standard experiment protocols on parameters and thresholds. Due to the page limit, we only use mutual information based ITs with the significance level of 0.02. All of the algorithms can freely use other ITs. For the MB discovery evaluation, we use the same MB discovery error metric as previous papers, namely the distance between the true MB and found MB: $d = \sqrt{(1 - \text{Precision})^2 + (1 - \text{Recall})^2}$. Precision is the number of true positives in the detected MB divided by the total size of the detected MB set. Recall is the number of true positives in the detected MB divided by the size of the ground truth MB. Thus, the lower d is better. We run the MB discovery for each node in each BN with ten different samples and report the average distance, along with standard deviation, for all the nodes. We report both time and the number of ITs conducted as the efficiency measure of different algorithms.

A. Synthetic Datasets

For synthetic datasets, we compare STMB with three topology-based algorithms: 1) HITON; 2) PCMB; and 3) IPCMB.

We use subnetworks of Fig. 1 to build two synthetic BNs with five nodes and seven nodes, respectively, shown in Fig. 2. These two networks include complex spouse-child relationships to verify that the detected MB results are sound. The parameters are randomly generated with some minor tuning to ensure that the faithfulness assumption holds.

Table I summarizes the experimental results under different sample sizes. It confirms all MB discovery algorithms, except for HITON, are sound and can find the true MB, when the faithfulness condition upholds. Since the existing implementation of HITON does not report the number of ITs used, we report time to compare different algorithms. Comparing the efficiency, we can see that STMB is two to three times faster than the current state of art IPCMB, which is exactly the average MB size in the testing networks, and much faster than all other topology-based algorithms.

TABLE I

DISTANCE, *d* VALUE, AND THE AVERAGE CPU TIME, IN SECONDS, ON SYNTHETIC DATASETS OF DIFFERENT DATA SIZE

		Distance				Time			
Dataset	Size	HITON	PCMB	IPCMB	STMB	HITON	PCMB	IPCMB	STMB
5BN	1000	0.133 ±0.0	0±0.0	0±0.0	0±0.0	0.0465±0.03	0.1150±0.03	0.0356±0.01	0.0104 ±0.01
5BN	5000	0.133±0.0	0±0.0	0±0.0	0±0.0	0.0862±0.03	0.1861±0.04	0.0348±0.02	0.0175±0.02
5BN	10000	0.133±0.0	0±0.0	0±0.0	0±0.0	0.1154±0.05	0.2809±0.06	0.0493±0.04	0.0232±0.02
5BN	50000	0.133±0.0	0±0.0	0±0.0	0±0.0	0.5223±0.07	1.2431±0.09	0.2025±0.02	0.0949±0.01
MEAN						**0.1926**	**0.4563**	**0.0806**	**0.0365**
7BN	1000	0.167±0.0	0±0.0	0±0.0	0±0.0	0.1415±0.05	0.2246±0.07	0.0570±0.03	0.0194±0.02
7BN	5000	0.167±0.0	0±0.0	0±0.0	0±0.0	0.2880±0.06	0.4588±0.09	0.0681±0.04	0.0292±0.02
7BN	10000	0.167±0.0	0±0.0	0±0.0	0±0.0	0.4408±0.11	0.8258±0.13	0.1146±0.08	0.0473±0.04
7BN	50000	0.167±0.0	0±0.0	0±0.0	0±0.0	2.0377±0.13	3.6914±0.16	0.5542±0.08	0.2507 ±0.04
MEAN						**0.7270**	**1.3002**	**0.1985**	**0.0867**

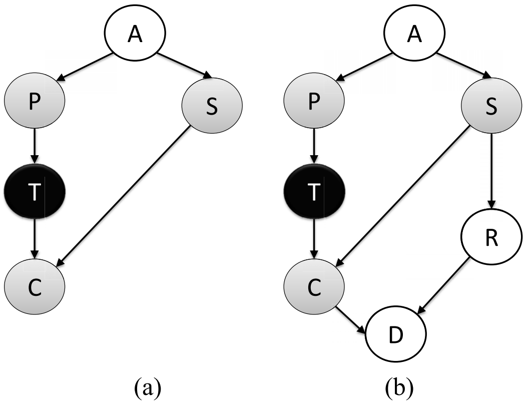

Fig. 2. Synthetic experiments: two sub-BNs from Fig. 1. (a) 5 variable BN, or 5BN. (b) 7 variable BN, or 7BN.

B. Empirical Study of Faithfulness Violation on Synthetic Datasets

We also evaluate the robustness of different MB discovery algorithms against faithfulness violation on synthetic datasets. Specifically, we use the popular Erdös−Rënyi model to randomly generate a DAG G with a fixed variable size N and a predefined probability q for edge generation, following the existing work [43]. We use $N = 10$ and $q = 0.5$. To produce an unfaithful distribution, we randomly generate the parameters from an uniform distribution for each node in the BN with structure G. It is well known that the Lebesgue measure of the set of unfaithful parameterizations for a graph is zero [44], as the number of unfaithful parameterizations is finite compared to the infinite continuous parameter space. As a result, the probability of unfaithful parameterizations is very low. Since the entire parameter space is equal to the parameter space of each parameter times the number of independence parameters, we have taken two measures in our experiments to increase the chance of producing unfaithful parameterizations: first, we discretize the continuous parameter space between 0 and 1 into 10^5 uniformly distributed discrete values, effectively making the parameter space finite. Second, for each node, we randomly make a subset of its parameters equal, therefore, reducing the number of independent parameters for each node. With these two measures, the parameter space is significantly reduced and therefore increases the probability of generating unfaithful parameterizations.

Given the randomly generated parameters for G, we directly use the definition of the faithfulness condition to find the unfaithful parameters. Specifically, we first use the generated conditional probabilistic tables of each node and the exact variable elimination inference method [1] to compute the conditional probabilistic distributions $P(X, Y|\mathbf{S})$, $P(X|\mathbf{S})$, and $P(Y|\mathbf{S})$, for every two variables $\{X, Y\}$ in the entire variable space $\mathbf{V}$, $\{X, Y\} \in \mathbf{V}$, and all possible conditioned set $\mathbf{S} \subseteq V\backslash\{X, Y\}$. We then check if $P(X, Y|\mathbf{S})$ is equal to $P(X|\mathbf{S}) \cdot P(Y|\mathbf{S})$ for all instantiations of the variables to determine the independence relationships of X and Y given $\mathbf{S}$ implied by the joint distribution P of the BN. Then by comparing the independence relationships between every pair of X and Y, conditioned on all possible subsets $\mathbf{S}$, implied by P with those by G, we can find the number of the inconsistency between independence relationships implied by structure G and those by P of the BN. The experiment can be conducted with a larger variable size N, but the exhaustive search for the faithfulness assumption has an exponential time complexity and it could easily become intractable. We generate a large amount of data (with the sample size of 10^6) and apply MB discovery algorithms to each node in the network, with the significance level of 0.02 for ITs. We repeat this procedure to generate 1000 BNs with the same G but different Ps, and plot the average distance of different MB discovery algorithms on these generated BNs versus the number of faithfulness violation, in order to compare the robustness of different MB discovery algorithms to faithfulness violation.

Fig. 3 shows that under varying numbers of faithfulness violation in the randomly generated BNs, STMB has slightly better accuracy than IPCMB. The results also show both methods have perform relatively stable performance under varying amounts of faithfulness violation in these synthetic networks, because the percentage of unfaithful relationships is relatively small, less than 0.1% of the total numbers of independence relationships.

C. Standard MB Discovery Data

We also evaluate STMB on standard MB datasets, available from online BN repository.[5] We compare STMB with only the state of the art algorithm IPCMB. Four networks are tested: 1) ALARM network has total 37 nodes, 46 edges, and

[5] http://www.bnlearn.com/bnrepository/

TABLE II
DISTANCE, *d* VALUE, AND THE NUMBER OF ITS CONDUCTED ON STANDARD DATASETS OF DIFFERENT DATA SIZES

		Distance		Independence Test	
Dataset	Size	IPCMB	STMB	IPCMB	STMB
Alarm	500	0.51±0.05	0.56±0.04	809.0±91.1	126.8±4.3
Alarm	1000	0.43±0.05	0.48±0.04	694.0±28.9	163.6±6.3
Alarm	5000	0.37±0.04	0.36±0.04	488.9±8.6	174.8±2.5
MEAN		**0.44**	**0.47**	**664.0**	**155.1**
Hail	500	1.08±0.03	1.24±0.02	2078.6±156.1	264.9±11.6
Hail	1000	1.06±0.02	1.00±0.01	1492.7±143.4	250.4±6.7
Hail	5000	1.10±0.01	1.04±0.01	307.6±36.8	159.1±3.6
MEAN		**1.08**	**1.09**	**1293.0**	**224.8**
Child	500	0.57±0.06	0.63±0.06	434.2±50.1	82.6±7.2
Child	1000	0.39±0.04	0.42±0.05	387.3±41.8	89.3±3.8
Child	5000	0.24±0.05	0.18±0.03	283.4±21.0	83.5±2.2
MEAN		**0.40**	**0.41**	**368.3**	**85.1**
Child10	1000	0.67±0.01	0.53±0.01	1629.0±56.7	838.7±16.5
Child10	5000	0.29±0.00	0.25±0.01	1239.4±24.3	705.4±3.4
MEAN		**0.48**	**0.39**	**1434.2**	**772.1**

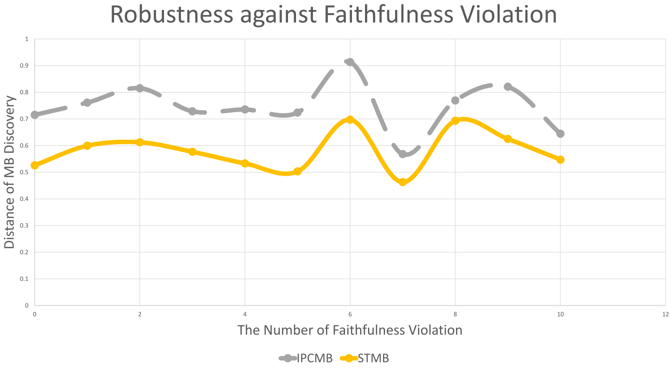

Fig. 3. Synthetic experiments on the robustness of IPCMB and STMB algorithms against faithfulness violation. The figure is best viewed in color.

509 parameters. The average PC set size is 2.48 ± 1.34 and the average MB size is 3.51 ± 2.07; 2) HAILFINDER has 56 nodes, 66 edges, and 2656 parameters. Its average PC set size is 2.36 ± 2.40 and the average MB size is 3.54 ± 2.70; 3) CHILD network has 20 nodes, with the average PC set size of 2.50 ± 1.70 and average MB size of 3.00 ± 2.15; and 4) CHILD10 is the CHILD network tiled ten times and has 200 nodes. Its average PC set size is 2.57 ± 1.63 and the MB size is 3.08 ± 2.06. Note that all these standard BN structure learning datasets have varying degree of unfaithfulness. It can be easily demonstrated by conducting simple tests to check the consistency between independence relationships implied by the graph structure and those by the BN distribution. For example, nodes 22 and 36 in ALARM are marginally dependent implied by the DAG with an unblocked path $\mathbf{22 - 9 - 16 - 17 - 26 - 27 - 36}$, but they are independent by the BN distribution, such as $P(V_{36} = 1, V_{22} = 1) = P(V_{36} = 1) \cdot P(V_{22} = 1) = 0.1548$ and for other variable values as well. The same case happens between Nodes 3 and 42 in HAILFINDER, and Nodes 1 and 16 in CHILD. Note that we did not include all the faithfulness violations here, due to the exhaustive nature of enumeration. The performance of STMB versus IPCMB on these datasets reflect the property of these algorithms when applied to unfaithful datasets.

We directly use the available data online for these datasets,[6] which contains ten different partitions for each data size. Table II shows the distance results. On the ALARM dataset, STMB shows the comparable accuracy with IPCMB. Speedwise, STMB is 4.3 times faster than IPCMB on average, and about seven times faster with 500 data. On the HAILFINDER dataset, the accuracy again is comparable between IPCMB and STMB, and STMB is again faster than IPCMB, with 5.8 times faster on average and about an order of magnitude faster with 500 data. Similar patterns are also observed in CHILD and the large CHILD10 datasets, with STMB showing relatively more speedup on lower sample sizes. On the largest dataset CHILD10, STMB also seems to outperform IPCMB in term of accuracy.

As discussed by Section IV-D, the expected speedup of STMB is the PC set size of a target variable. According to networks structures, the expected speedup of STMB should be 2.46 on ALARM, 2.36 times on HAILFINDER, 2.50 times on CHILD, and 2.57 times on CHILD10. The empirical speedup with the largest sample size of 5000 is 2.80 times on ALARM, 1.93 times on HAILFINDER, 3.39 times on CHILD, and 1.78 times on CHILD10. There results are mostly consistent with the complexity analysis. Significant speedup is also observed with smaller sizes as discussed above. The efficiency differences among different data sizes can be contributed to the fact that small data sizes could introduce more erroneous dependencies (introducing erroneous independence is much harder as the measure of correlation is harder to change to zero than to any other number). More dependence mean more variables left to search, and the search procedure could take exponentially longer time, due to the greedy nature with the gradually increasing conditioned set sizes. IPCMB is more prone to this error due to the usage of the symmetry constraint, which conducts PC set search for each PC node of the target. More

[6] http://www.cs.mtu.edu/lebrown/supplements/mmhc_paper/mmhc_index.html

TABLE III
ERROR RATE ON REAL FEATURE SELECTION DATASETS USING KNN

DATASET	JMI	IPCMB	STMB
CONGRESS	10.14%	7.83%	**6.91%**
HEART	21.48%	**19.26%**	21.48%
KRVSKP	7.88%	16.46%	**4.50%**
LUNGCANCER	**62.5%%**	FAIL	**62.5%**
LANDSAT	13.46%	13.27%	**13.15%**
PARKINSONS	18.37%	FAIL	**17.35%**
SPECT	23.15%	22.56%	**19.40%**
SPLICE	27.91%	30.37%	**27.41%**
WAVEFORM	21.12%	21.12%	**20.72%**
WINE	**8.99%**	**8.99%**	**8.99%**
MEAN	21.27%	N/A	**20.23%**
AVERAGE TIME	5.0 SEC	233.06 SEC	14.97 SEC

erroneous tests result longer time for IPCMB with smaller sample sizes. With large sample sizes, the empirical efficiency improvements of STMB over existing methods are consistent with the average PC set sizes of these networks.

D. Real Feature Selection Data

We use ten datasets from [42] to show the effectiveness of STMB on feature selection applications. Since MB is a filter method [2], we make a direct comparison with the state-of-the-art filtered method joint mutual information (JMI), which is shown to have the best performance [42]. We follow their experiment setups and use a simple K-Nearest Neighbors (KNN) classifier to test the accuracy. We choose the top N JMI features, where $N = \max(|\text{IPCMB}|, |\text{STMB}|)$ for each dataset. Table III shows the error rates of different feature selection algorithms with bold numbers representing the best results. IPCMB can fail the feature selection by selecting none of the features, due to no PC set overlaps during the symmetry check. STMB is very competitive in term of accuracy. Specifically, STMB reduces the error rate of JMI by 43% on KRVSKP and 32% on CONGRESS, and has the best overall performance on these ten datasets. Speed-wise, we use time to compare the efficiency for different methods (as JMI does not compute ITs). STMB is slower than JMI[7] but is about 15 times faster than IPCMB on average.

E. Performance Discussion

STMB can, through theoretical and empirical analysis, improve the efficiency of the state-of-the-art algorithm IPCMB, typically by the target's PC set size and sometimes by one order of magnitude when data size is small, with comparable accuracy. STMB is also empirically shown to be stable under different amounts of faithfulness violation and to be slightly more robust to faithfulness violation than IPCMB.

Despite its promising performance, the theoretical analysis of STMB holds only under the faithfulness assumption. Although it improves the efficiency, STMB still has an exponential complexity in the worst case. Empirically, its accuracy depends on the accuracy of conditional ITs whose accuracy depends on the sample size. The accuracy of the PC set search in step 1 of STMB also significantly affects the later steps.

VI. CONCLUSION

We present a novel topology-based algorithm of finding the MB, improving the efficiency of current state-of-the-art topology-based methods. The main contributions of this paper are the discovery of the coexistence property of spouses and false PC nodes, the introduction and the theoretical analysis of the STMB algorithm to simultaneously find them. STMB can apply to any other topology-based MB discovery methods. Experiments have shown STMB has a comparable accuracy to other topology-based methods but takes much less time for both MB discovery and feature selection. Future studies could focus on improving the accuracy of the PC set search such that it would improve the robustness of STMB against faulty PC sets. Another future research direction would be to systematically analyze the impact of faithfulness violation in MB discovery algorithms, following the existing framework in [34]–[38]. It would be also interesting to study how to minimize the effect of assumption violations on the performance of STMB and other MB algorithms in general.

REFERENCES

[1] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, 2nd ed., M. B. Morgan, Ed. San Mateo, CA, USA: Morgan Kaufmann, 1988.

[2] D. Koller and M. Sahami, "Toward optimal feature selection," in *Proc. ICML*, Bari, Italy, 1996, pp. 284–292.

[3] D. Margaritis and S. Thrun, "Bayesian network induction via local neighborhoods," in *Proc. Adv. Neural Inf. Proc. Syst.*, Denver, CO, USA, 1999, pp. 505–511.

[4] I. Tsamardinos, C. F. Aliferis, and A. R. Statnikov, "Algorithms for large scale Markov blanket discovery," in *Proc. 16th Int. FLAIRS Conf.*, St. Augustine, FL, USA, 2003, pp. 376–380.

[5] S. Yaramakala and D. Margaritis, "Speculative Markov blanket discovery for optimal feature selection," in *Proc. 5th IEEE Int. Conf. Data Min.*, Houston, TX, USA, 2005, pp. 809–812.

[6] J. M. Peña, R. Nilsson, J. Björkegren, and J. Tegnér, "Towards scalable and data efficient learning of Markov boundaries," *Int. J. Approx. Reason.*, vol. 45, no. 2, pp. 211–232, Jul. 2007.

[7] A. Klein and S. Shimony, "Discovery of context-specific Markov blankets," in *Proc. IEEE Int. Conf. Syst. Man Cybern.*, vol. 4, Oct. 2004, pp. 3833–3838.

[8] I. Tsamardinos, C. F. Aliferis, and A. Statnikov, "Time and sample efficient discovery of Markov blankets and direct causal relations," in *Proc. 9th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, Washington, DC, USA, 2003, pp. 673–678.

[9] I. Tsamardinos, L. E. Brown, and C. F. Aliferis, "The max-min hill-climbing Bayesian network structure learning algorithm," *Mach. Learn.*, vol. 65, no. 1, pp. 31–78, 2006.

[10] C. F. Aliferis, A. Statnikov, I. Tsamardinos, S. Mani, and X. D. Koutsoukos, "Local causal and Markov blanket induction for causal discovery and feature selection for classification part I: Algorithms and empirical evaluation," *J. Mach. Learn. Res.*, vol. 11, pp. 171–234, Jan. 2010.

[11] C. F. Aliferis, I. Tsamardinos, and A. Statnikov, "HITON: A novel Markov blanket algorithm for optimal variable selection," in *Proc. AMIA Annu. Symp. Proc.*, 2003, pp. 21–25.

[12] S. Fu and M. C. Desmarais, "Fast Markov blanket discovery algorithm via local learning within single pass," in *Proc. 21st Conf. Adv. Artif. Intell. Can. Soc. Comput. Stud. Intell.*, Windsor, ON, Canada, 2008, pp. 96–107.

[13] C. F. Aliferis, A. Statnikov, I. Tsamardinos, S. Mani, and X. D. Koutsoukos, "Local causal and Markov blanket induction for causal discovery and feature selection for classification part II: Analysis and extensions," *J. Mach. Learn. Res.*, vol. 11, pp. 235–284,, Jan. 2010.

[14] J. Yu, S. S. R. Abidi, and P. H. Artes, "A hybrid feature selection strategy for image defining features: Towards interpretation of optic nerve images," in *Proc. Int. Conf. Mach. Learn. Cybern.*, vol. 8. Guangzhou, China, 2005, pp. 5127–5132.

[7] JMI time is an estimated MATLAB time.

[15] H. H. Yang and J. Moody, "Feature selection based on joint mutual information," in *Proc. Int. ICSC Symp. Adv. Intell. Data Anal.*, New York, NY, USA, 1999, pp. 22–25.

[16] Y. Liu, F. Tang, and Z. Zeng, "Feature selection based on dependency margin," *IEEE Trans. Cybern.*, vol. 45, no. 6, pp. 1209–1221, Jun. 2015.

[17] Y. Mohsenzadeh, H. Sheikhzadeh, A. M. Reza, N. Bathaee, and M. M. Kalayeh, "The relevance sample-feature machine: A sparse Bayesian learning approach to joint feature-sample selection," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 2241–2254, Dec. 2013.

[18] R. Hong *et al.*, "Image annotation by multiple-instance learning with discriminative feature mapping and selection," *IEEE Trans. Cybern.*, vol. 44, no. 5, pp. 669–680, May 2014.

[19] M. Liang and X. Hu, "Feature selection in supervised saliency prediction," *IEEE Trans. Cybern.*, vol. 45, no. 5, pp. 900–912, May 2015.

[20] I. Guyon, C. Aliferis, and A. Elisseeff, "Computational methods of feature selection," in *Causal Feature Selection*, Boca Raton, FL, USA: CRC Press, 2007, ch. 4, pp. 63–86.

[21] G. C. Cawley, "Causal and non-causal feature selection for ridge regression," in *Proc. WCCI Causation Predict. Challenge*, 2008, pp. 107–128.

[22] D. M. Chickering, D. Heckerman, and C. Meek, "Large-sample learning of Bayesian networks is NP-hard," *J. Mach. Learn. Res.*, vol. 5, pp. 1287–1330, Jan. 2004.

[23] D. Vidaurre, C. Bielza, and P. Larrañaga, "Learning an L1-regularized Gaussian Bayesian network in the equivalence class space," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 40, no. 5, pp. 1231–1242, Oct. 2010.

[24] Y. Sun, A. K. Wong, and Y. Wang, "Generative and discriminative learning by CL-net," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 37, no. 4, pp. 1022–1029, Aug. 2007.

[25] Y. Xiang and M. Truong, "Acquisition of causal models for local distributions in Bayesian networks," *IEEE Trans. Cybern.*, vol. 44, no. 9, pp. 1591–1604, Sep. 2014.

[26] A. Cano, A. R. Masegosa, and S. Moral, "A method for integrating expert knowledge when learning Bayesian networks from data," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 41, no. 5, pp. 1382–1394, Oct. 2011.

[27] J.-P. Pellet and A. Ellisseeff, "Using Markov blankets for causal structure learning," *J. Mach. Learn.*, vol. 9, pp. 1295–1342, Jul. 2008.

[28] T. Silander and P. Myllymaki, "A simple approach for finding the globally optimal Bayesian network structure," in *Proc. 22nd Annu. Conf. Uncertainty Artif. Intell. (UAI)*, 2006, pp. 445–452.

[29] C. P. de Campos, Z. Zeng, and Q. Ji, "Structure learning of Bayesian networks using constraints," in *Proc. 26th Annu. Int. Conf. Mach. Learn.*, Montreal, QC, Canada, 2009, pp. 113–120.

[30] P. Spirtes, C. N. Glymour, and R. Scheines, *Causation, Prediction, and Search*. New York, NY, USA: Springer-Verlag, 1993.

[31] S. R. De Morais and A. Aussem, "A novel scalable and data efficient feature subset selection algorithm," in *Machine Learning and Knowledge Discovery in Databases*. Heidelberg, Germany: Springer, 2008, pp. 298–312.

[32] T. Niinimaki and P. Parviainen, "Local structure discovery in Bayesian network," in *Proc. Uncertainty Artif. Intell. Workshop Causal Struct. Learn.*, 2012, pp. 634–643.

[33] P. Spirtes, C. N. Glymour, and R. Scheines, *Causation, Prediction, and Search*, vol. 81, 2nd ed. Cambridge, MA, USA: MIT Press, 2000.

[34] A. Statnikov, N. I. Lytkin, J. Lemeir, and C. F. Aliferis, "Algorithms for discovery of multiple Markov boundaries," *J. Mach. Learn. Res.*, vol. 14, no. 1, pp. 499–566, 2013.

[35] Y. Xiang, S. M. Wong, and N. Cercone, "Critical remarks on single link search in learning belief networks," in *Proc. 12th Int. Conf. Uncertainty Artif. Intell.*, Portland, OR, USA, 1996, pp. 564–571.

[36] T. Chu and Y. Xiang, "Exploring parallelism in learning belief networks," in *Proc. 13th Conf. Uncertainty Artif. Intell.*, Providence, RI, USA, 1997, pp. 90–98.

[37] Y. Xiang, J. Lee, and N. Cercone, "Parameterization of pseudo-independent models." in *Proc. FLAIRS Conf.*, St. Augustine, FL, USA, 2003, pp. 521–525.

[38] Y. Xiang and J. Lee, "Learning decomposable Markov networks in pseudo-independent domains with local evaluation," *Mach. Learn.*, vol. 65, no. 1, pp. 199–227, 2006.

[39] X. Yang, "Pseudo-independent models and decision theoretic knowledge discovery," in *Encyclopedia of Data Warehousing and Mining*, 2nd ed. Hershey, PA, USA: IGI Global, 2009, ch. 249, pp. 1632–1638.

[40] C. X. Ling and H. Zhang, "The representational power of discrete Bayesian networks," *J. Mach. Learn. Res.*, vol. 3, pp. 709–721, Mar. 2003.

[41] T. Claassen, J. Mooij, and T. Heskes, "Learning sparse causal models is not NP-hard," in *Proc. 27th Conf. Annu. Conf. Uncertainty Artif. Intell.*, Bellevue, WA, USA, 2013, pp. 172–181.

[42] C. F. Aliferis, I. Tsamardinos, A. Statnikov, and L. E. Brown, "Causal explorer: A probabilistic network learning toolkit for biomedical discovery," in *Proc. Int. Conf. Math. Eng. Tech. Med. Biol. Sci. (METMBS)*, Las Vegas, NV, USA, Jun. 2003. pp. 371–376.

[43] G. Brown, A. Pocock, M.-J. Zhao, and M. Lujan, "Conditional likelihood maximisation: A unifying framework for information theoretic feature selection," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 27–66, Jan. 2012.

[44] H. Hu, Z. Li, and A. R. Vetta, "Randomized experimental design for causal graph discovery," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2339–2347.

Tian Gao (S'11) received the B.S. degree from the Department of Electrical, Computer, and System Engineering, Rensselaer Polytechnic Institute, Troy, NY, USA, in 2009, where he is currently pursuing the Ph.D. degree in electrical engineering.

From 2010 to 2012, he was a National Science Foundation Triple Helix Program Fellow, and has broad experiences with machine learning, pattern recognition, computer vision, and affective computing. His current research interests include probabilistic graphical models, feature selection, and causal discovery.

Qiang Ji (F'15) received the Ph.D. degree in electrical engineering from the University of Washington, Seattle, WA, USA.

He is currently a Professor with the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute (RPI), Troy, NY, USA. From 2009 to 2010, he served as a Program Director with the National Science Foundation (NSF), Arlington, VA, USA, where he managed NSF's computer vision and machine learning programs. He also held teaching and research positions with the Beckman Institute, University of Illinois at Urbana–Champaign, Champaign, IL, USA, the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA, the Department of Computer Science, University of Nevada, Reno, NV, USA, and the U.S. Air Force Research Laboratory, Rome, NY, USA. He currently serves as the Director of the Intelligent Systems Laboratory with RPI. He has published over 200 papers in peer-reviewed journals and conferences. His current research interests include computer vision, probabilistic graphical models, pattern recognition, and their applications in various fields.

Prof. Ji was a recipient of multiple awards for his work. He is an Editor of several related IEEE and international journals. He has served as the General Chair, the Program Chair, the Technical Area Chair, and the Program Committee Member in numerous international conferences/workshops. He is a fellow of the International Association of Pattern Recognition.