Research Article

Upper-Lower Bounds Candidate Sets Searching Algorithm for Bayesian Network Structure Learning

Guangyi Liu, Ou Li, Dalong Zhang, and Tao Song

Zhengzhou Information Science and Technology Institute, Zhengzhou 450000, China

Correspondence should be addressed to Guangyi Liu; liu.guangyi@outlook.com

Received 25 September 2014; Accepted 30 October 2014; Published 25 November 2014

Academic Editor: Yoshinori Hayafuji

Copyright © 2014 Guangyi Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Bayesian network is an important theoretical model in artificial intelligence field and also a powerful tool for processing uncertainty issues. Considering the slow convergence speed of current Bayesian network structure learning algorithms, a fast hybrid learning method is proposed in this paper. We start with further analysis of information provided by low-order conditional independence testing, and then two methods are given for constructing graph model of network, which is theoretically proved to be upper and lower bounds of the structure space of target network, so that candidate sets are given as a result; after that a search and scoring algorithm is operated based on the candidate sets to find the final structure of the network. Simulation results show that the algorithm proposed in this paper is more efficient than similar algorithms with the same learning precision.

1. Introduction

Bayesian network (BN), as a graphic model handling uncertainty issues, has been discussed by many researchers through these years. It has been applied successfully in many areas such as fault detection, medical diagnosis, and traffic management [1–3]. It had been years that people focused on finding a data structure to compress the storage of joint probability density and developing inference algorithms based on that data structure, and then BN was brought up. After that, when BN had been a successful tool in this area, researchers began to follow with interest structure learning algorithms of BN based on sample data. Essentially, the problem of structure learning of BN is part of combinatorial optimization issues, and it is proved theoretically that learning structure from data was NP hard [4]. Nonetheless, some heuristic methods have been proposed and performed well in several areas [5, 6].

Currently, there are two approaches for BN structure learning. One is CI-test method [7, 8] and the other is scored-searching method [9, 10]. The first one uses conditional independence tests (CI test) to determine the conditional independence relationships among all the variables and build networks based on these relationships. The scored-searching methods attempt to find the network by maximizing a scoring function which indicates how well the network fits the data.

Both methods above have their own advantages and disadvantages. CI-test algorithms are simple and easy to operate. Because low-order CI test is quite computational effective and has high precision, they are very helpful to build a hyper graph of the target (it will be discussed in the following sections). The main drawback about these methods is about performing high-order CI test, which needs large sample sizes and has low accuracy along with the orders of getting higher [11, 12]. The scored-searching methods may have higher precision in structure learning than the CI-test methods. But they are relatively slow, especially when the scales of networks become large, as the structures space increases super-exponentially with the number of nodes.

It is obvious that if it is possible to combine the learning efficiency of CI test and prediction accuracy of scored-searching algorithm, we will get a better algorithm to deal with BN learning issues. In view of the above reasons, some hybrid methods have been proposed [13–17]. These methods may use CI-test algorithms to learn a network structure pattern at first and then use some scored-searching algorithms to find the final BN structures based on the previous pattern. These hybrid methods may perform better in some applications, but there are still some problems unsolved, as fusion in algorithm level does not always mean promotion in performance. Take MMHC (max-min hill climbing) as an

example. It includes two steps: the first one called MMPC (max-min parents and children), which constructs parents and children sets of each node via CI-test method, is to provide a partial skeleton frame. While in the second step a hill climbing algorithm is operated to refine every edge in the network. To ensure the precision of partial skeleton frame given by MMPC, high-order CI test must be involved, which unfortunately is unstable [11, 12]. So in the searching phase, it is not based on the prior structure given by MMPC strictly but operates in a relatively open space. This manner seems somewhat wasteful for computational resources.

The upper-lower bounds candidate sets searching algorithm (UBCS) which is proposed in this paper can provide a more instructive set of candidate networks through constructing the upper-lower bounds of the structure space by low-order CI test. In this framework, we get the final network structure by using the greedy search algorithm. Simulation shows that it could guarantee precision and reduce the time complexity at the same time.

2. Background

2.1. Definitions

Definition 1 (Bayesian network definition). Given a joint distribution P of random variables $x_1, x_2, \dots, x_n$ and DAG (directed acyclic graph), $BN = (G, P)$ is a Bayesian network, if vertices set $V = \{v_1, v_2, \dots, v_n\}$ is in one-to-one correspondence with random variables $x_1, x_2, \dots, x_n$ and (G, P) sacrifice Markov condition [18].

Because nodes in BN have no difference with random variables, they will not be distinguished in this paper, and they will be both called as node. In addition, let $\vec{e}_{ij}$ denote directed edge $v_i \rightarrow v_j$, and let e_{ij} denote undirected edge $v_i - v_j$.

Definition 2 (V-structure). Let $BN = (G, P)$, where $G = (V, E)$, $\forall v_i, v_j, v_k \in V$ construct a V-structure, if $\vec{e}_{ik} \in E$, $\vec{e}_{jk} \in E$, $\vec{e}_{ij} \notin E$, and $\vec{e}_{ji} \notin E$.

Definition 3 (conditional mutual information $MI(X, Y | Z)$). Conditional mutual information $MI(X, Y | Z)$ of random variables sets X, Y, Z is defined as

$$\begin{aligned} MI(X, Y | Z) &= H(X | Z) - H(X | Z, Y) \\ &= \sum_{x_i \in X, y_j \in Y} P(x_i, y_j | Z) \log \frac{P(x_i, y_j | Z_k)}{P(x_i | Z) P(y_j | Z)}, \end{aligned} \quad (1)$$

where

$$\begin{aligned} H(X | Z) &= \sum_{x_i \in X, z_k \in Z} P(x_i, z_k) \log \frac{1}{P(x_i | z_k)} \\ H(X | Z, Y) &= \sum_{x_i \in X, y_j \in Y, z_k \in Z} P(x_i, y_j, z_k) \log \frac{1}{P(x_i | y_j, z_k)}. \end{aligned} \quad (2)$$

$MI(X, Y | Z) = 0$ means random variables sets X, Y are conditional independence given Z , which can be expressed as $\text{Ind}(X, Y | Z)$ too. Therefore, it usually uses $MI(X, Y | Z)$ as CI test among random variables and calls cardinal number of Z as orders of CI test. Furthermore, its zero order CI test, if $Z = \Phi$.

Definition 4 (Markov equivalence). Two DAGs are graph equivalent if and only if (1) both of them have the same skeleton frame and (2) they have the same V structure.

Two BN $BN_1 = (G_1, P_1)$ and $BN_2 = (G_2, P_2)$ are Markov equivalent, if G_1 and G_2 are graph equivalent.

The characteristics of Markov equivalence have been given by Frydenberg [19], while Verma and Pearl expanded these into DAG [20]. Based on the Markov equivalence, all the DAGs composed by the same nodes set can be divided into different equivalent classes, which are called Markov equivalent classes. Each equivalent class indicates a unique statics model, and it can be represented by a PDAG (partial directed acyclic graph), which is called complete PDAG.

3. Method

Given data sets D , BN structure learning methods are devoted to find the best network structure of $BN = (G, P)$. The reference [21] proved the structure quantity of the BN which contains n nodes is

$$f(n) = \sum_{i=1}^n (-1)^{i+1} C_n^i 2^{i(n-i)} f(n-1). \quad (3)$$

From the formula above it can be seen that the potential network structure space rises exponentially with the node increasing. So searching for the candidate sets of network structure is a good approach to reduce dimensions effectively. Based on it, we provide an algorithm named upper-lower bounds candidate sets searching algorithm (short for UBCS), which can get the ultimate network model by constructing the upper-lower bounds of the target network pattern to find candidate sets of network and using search and scoring method. In the following section we will give the first part of the UBCS which is called upper-bound of graph learning algorithm (UGLA), prove the output G^+ is the upper bound of moral graph of the target network, and then bring in principle of nonincreasing for 0-order mutual information to reach the second section of the UBCS which is called lower bound of graph learning algorithm (LGLA). After that the searching algorithm will be discussed.

3.1. UGLA and LGLA. We will first give the algorithm description in Algorithm 1.

UGLA processing indicates that G^+ is a triangulating graph, and for triangulating graph, we have the following theorem being tenable.

Theorem 5. Any undirected graph G is complete PDAG, if and only if G is a triangulating graph.

- (1) Input: Data set D ; Variable set $V = \{v_1, v_2, \dots, v_n\}$;
- (2) Initialization: undirected graph $G^+ = (V, E)$, where $E = \Phi$;
- (3) Order-0 CI test: for each pair variables $v_i, v_j \in V, i = 1, 2, \dots, n$, if $\text{Ind}(v_i, v_j)$ is hold, then $E = E \cup \{e_{ij}\}$, where $e_{ij} = (v_i, v_j)$ is an undirected edge;
- (4) For each triplet variables $v_i, v_j, v_k \in V$
- (5) For each pair variables $v_j, v_k \in V, v_j, v_k \in V$ and $v_j \neq v_k \neq v_i$, if $e_{ij} \in E, e_{ik} \in E, e_{jk} \notin E$, and $\neg \text{Ind}(v_j, v_k \mid v_i)$ is hold, then $E = E \cup \{e_{jk}\}$;
- (6) End for;
- (7) Triangulate undirected graph G^+ , what results with G_T^+
- (8) Call the junction tree algorithm to obtain maximal prime sub-graphs $G_{\text{sub}}^1, G_{\text{sub}}^2, \dots, G_{\text{sub}}^k$ from G_T^+ ;
- (9) Output G^+ and $G_{\text{sub}}^1, G_{\text{sub}}^2, \dots, G_{\text{sub}}^k$ as final result.

ALGORITHM 1: UGLA.

The theorem which has been proved in [22] shows that G^+ is a complete PDAG; that is, in the best situation, G^+ obtained by UGLA is the PDAG of the target BN. Certainly, this condition is too strong, and we will give a theorem below which has more generality.

Theorem 6. Given sample dataset D , let the optimal structure of $BN = (G, P)$ to be learned be G_D , moral graph of the BN is $G_{D,m}$, and then the undirected graph G^+ obtained by UGLA is the upper bound of partially ordered set $GM = (G_{D,m}, \subseteq)$.

Proof. It only needs to prove that $G_{D,m}^i \subseteq G^+$ holds for each $G_{D,m}^i \in GM$, where $G_{D,m}^i = (V, E_{D,m}^i)$ and $G^+ = (V, E^+)$. Theorem 5 tells that if the complete PDAG of G_D is triangulating graph, $G_{D,m}^i = G^+$ is tenable. So the following task is to get $G_{D,m}^i \subset G^+$ proved. As all the graphs have the same nodes set, it only needs to prove that there is $e_{jk} \in E^+$ in any case that undirected edge $e_{jk} \in E_{D,m}^i$. For all the undirected edges in $E_{D,m}^i$, it is clear that it can be divided into two classes. One is composed by the undirected edges transformed by directed edges in G_D ; let it be $\bar{E}_{D,m}^i$. The other one is being constructed by the moral edges adding between the nodes which have the same parent; let it be $\tilde{E}_{D,m}^i$. It is obvious that, for $\forall e_{jk} \in \bar{E}_{D,m}^i$, 0-order CI test ensures that $e_{jk} \in E^+$ must be tenable; for $\forall e_{jk} \in \tilde{E}_{D,m}^i$, the fifth step of UGLA is the assurance of $e_{jk} \in E^+$. Proof is completed. $\square$

Theorem 7. Any V-structure in $BN = (G, P)$ exists in a subgraph decomposed from G^+ by the method MPD (maximal prime subgraph decomposition) [23].

Theorem 7 was proved in reference [17]. This theorem guarantees that the $G_{\text{sub}}^1, G_{\text{sub}}^2, \dots, G_{\text{sub}}^k$ obtained by UGLA covers all the V-structure in the target graph.

The above section discussed the upper bound of $GM = (G_{D,m}, \subseteq)$, from what we can get the candidate sets for searching the structure. In the following part, the lower bound of BN structure space will be debated for choosing a relatively precise initial value.

We will start with a lemma as blow.

Lemma 8. For any two random variables $v_i, v_j \in V$ and subset $S \subseteq V / \{v_i, v_j\}$ of V , $H(v_i \mid v_j, S) \leq H(v_i \mid v_j)$ is tenable, and the equity holds if and only if $\text{Ind}(v_i, v_j \mid S)$.

The proof is omitted.

Theorem 9. Given $BN = (G, P)$, where $G = (V, E)$, $MI(v_i, v_k) = \min\{\bigcup_{v_l, v_n \in V', v_l \neq v_n} MI(v_l, v_n)\}$ holds for any $v_i, v_j, v_k \in V$, where $V' = \{v_i, v_j, v_k\} \subseteq V$. If the connecting relationship among $v_i, v_j, v_k \in V$ is $\vec{e}_{ij} \in E, \vec{e}_{kj} \in E, \vec{e}_{ik} \notin E$, and $\vec{e}_{ki} \notin E$.

Proof. As $\vec{e}_{ij} \in E$, without loss of generality, let $MI(v_i, v_j) = \min\{MI(v_i, v_j), MI(v_k, v_j)\}$, and it is only necessary to prove that $MI(v_i, v_k) > MI(v_i, v_j)$. According to the definition of mutual information, $MI(v_i, v_k) = H(v_i) - H(v_i \mid v_k)$ and $MI(v_i, v_j) = H(v_i) - H(v_i \mid v_j)$.

Then

$$\begin{aligned} MI(v_i, v_k) - MI(v_i, v_j) &= H(v_i) - H(v_i \mid v_k) - H(v_i) + H(v_i \mid v_j) \\ &= H(v_i \mid v_k) - H(v_i \mid v_j). \end{aligned} \quad (4)$$

It can be seen that $v_j \in S$ from the relationship among v_i, v_j, v_k , where S scarifies $\text{Ind}(v_i, v_k \mid S)$, so the equation above can be expressed as

$$\begin{aligned} MI(v_i, v_k) - MI(v_i, v_j) &= H\left(v_i \mid v_j, \{v_k\} \cup \frac{S}{v_j}\right) - H(v_i \mid v_j). \end{aligned} \quad (5)$$

Lemma 8 shows $MI(v_i, v_k) \leq MI(v_i, v_j)$, while the equity holds for $S/v_j = \Phi$.

Proof is completed. $\square$

We named Theorem 9 as principle of nonincreasing for 0-order mutual information (principle NZMI). The condition of the theorem indicates that it is not suitable for the situation when there is V-structure. For the BN structure shown

(1) Input: Data set D ; Variable set $V = \{v_1, v_2, \dots, v_n\}$; $G_{\text{sub}}^1, G_{\text{sub}}^2, \dots, G_{\text{sub}}^k$;
 (2) Initialization: $G^- = (V, E^-)$, where $E^- = \Phi$;
 (3) For each G_{sub}^i , If there is a cycle, then call VSTA;
 (4) For each $v_i \in V$, Find $v_j \in V$ makes $\text{MI}(v_i, v_j)$ the highest, $E^- = E^- \cup \{e_{ij}\}$;
 (5) If there is another $v_k \in V$ makes $\text{MI}(v_i, v_j) = \text{MI}(v_i, v_k)$ is the highest
 (6) If $\text{Ind}(v_i, v_k | v_j)$ is hold, then $E^- = E^- \cup \{e_{ik}\} / \{e_{ij}\}$,
 (7) End if;
 (8) Output $G^- = (V, E^-)$ as final result.

ALGORITHM 2: LGLA.

(1) Input: Data set D ; Variable set $V = \{v_1, v_2, \dots, v_n\}$; $G^- = (V, E^-)$; G_{sub}^i , where $G_{\text{sub}}^i[V] = \{v_i, v_j, v_k\}$;
 (2) If $\text{Ind}(v_i, v_j)$ and $\neg \text{Ind}(v_i, v_j | v_k)$
 (3) Orient $v_i \rightarrow v_k \leftarrow v_j$ and delete $v_i - v_j$ in G^- ,
 (4) Else if $\neg \text{Ind}(v_i, v_j | v_k)$ and $\text{Ind}(v_i, v_j | v_l)$, $v_l \in V / \{v_i, v_j, v_k\}$, then
 (5) Orient $v_i \rightarrow v_k \leftarrow v_j$ and delete $v_i - v_j$ in G^- ,
 (6) End if;
 (7) Output G^- .

ALGORITHM 3: VSTA.

in Figure 1, it cannot tell whether $\text{MI}(a, d)$ is bigger than $\text{MI}(a, b)$ or not, only from the Theorem 9. But, if we can eliminate all the V-structure first and then come to consider the connected relationship between node a and nodes cbe , we will notice that $\text{MI}(a, c) = \text{MI}(a, b) > \text{MI}(a, e)$ must hold when $\text{MI}(a, c)$ is the biggest. So it only needs 1-order CI test to rule out the possibility of ac connection, so it turns out that ab are connected. As a matter of fact, principle NZMI provides a new approach for ascertaining whether there is an undirected edge between two nodes without using the V-structure methods.

Algorithm 2 gives the G^- learning algorithm (LGLA) based on the discussion above.

The VSTA (V-structure test algorithm) is involved in LGLA list as in Algorithm 3.

For V-structure test algorithm (VSTA) see Algorithm 3.

VSTA is a testing method which only provides “best effort” services. It only involves 0-order and 1-order CI tests, the high accuracy of which guarantees the existence of V-structure detected. For the situation that there is more than one edge between two father nodes in a V-structure, the detecting will not be operated. This approach avoids bringing in high computation and additional interference edges.

These is a theorem that holds for the output from LGLA.

Theorem 10. Let all the complete PDAGs from $\text{BN} = (G, P)$ and inclusion relationship compose a partially ordered set $\text{PG} = (G_{\text{PDAG}}, \subseteq)$, and then the G^- generated from LGLA is a PDAG.

Proof. For the undirected edges included in G^- , the principle NZMI guarantees that $e_{ij} \in G[E]$ holds for any $e_{ij} \in G^-[E^-]$. VSTA gives the directed edges of G^- , and according to the Theorem 7, any V-structure in $\text{BN} = (G, P)$ is surely contained in a G_{sub}^i . The characteristic of VSTA makes sure

FIGURE 1: V-structure in BN.

that a V-structure must exist in G if it is contained by G^- , which does not hold water certainly conversely. It is obvious that an acyclic graph will be still acyclic no matter any directed edges are deleted. So G^- is a PDAG.

Proof is completed. $\square$

Theorem 11. G^- is the lower bound of PG if output from VSTA is entirely accurate.

Proof is omitted.

The condition of Theorem 11 is relatively strong. As a matter of fact, it can be considered that G^- is the lower bound of PG in many cases. Take network Asia as an example; Figure 2 shows that all the edges in G^- exist in the PDAG of original network.

3.2. Searching Method. Hill-climbing algorithm which is based on search and scoring method is one of the greedy searching algorithms for BN structure learning. It contains three searching operators: adding edge, subtractive edge, and reversing edge. The hill-climbing method is also involved in the UDCS algorithm, but the searching processing is

FIGURE 2: Three subpictures from left to right are structure of Asia network, PDAG of Asia network, and G^- of Asia network.

restricted by the upper and lower bounds given by UGLA and LGLA, which means abandoning the new structure got from searching operators if the new one beyond the bounds is given by UGLA nor LGLA. In case of trapping in local optimum too fast, bring in suboptimal competitive mechanism and retain the top N structures which got higher scores each round to the next iteration. The N is decided by the scale of network in principle, which means the greater the scale of network is, the bigger the N is. But it should be noticed that the oversize candidate sets will lead to an increase in the time complexity of algorithm. For the BN network of which scale is as big as alarm, recommend empirical value is $N = 10$. In order to make a comparison with the BENA algorithm mentioned in [17], the BDeu score function is used as the objective function of searching.

4. Experiment

We test the performances of UBCS with BNEA and MMHC together in Alarm network. The comparison of scores is shown in Table 1. For ease of observation, we present a normalization to deal with the results.

Table 1 shows the results averaged over 10 runs, where SS represents sample size. As can be seen from Table 1, the performance of UBCS is the best among three methods when SS is small, and the scores from all three methods tend to be very close with the increase of SS. Although the VSTA which is involved by LGLA cannot be adequate to assure the facticity of the detection, it has little impact on the learning performance according to the simulation result. This phenomenon is caused by two reasons: on one side, the upper bound given by UGLA is very stable, and on the other side, the effect is reduced by the process of search and scoring. It should be noticed that the BENA shows “over learning” when SS becomes larger ($SS > 5000$). The “over learning” is considered as a phenomenon only occurs in small sample size typically, while, as the combinatorial optimization with high dimensions (such as BN), it is hard to get plenty of samples, and the time cost is also unacceptable when the current algorithms operate on extremely large datasets. So it is reasonable to find the algorithms that could get a balance between precision and generalization in dataset with appropriate size, which is the intention of UBCS, as the restriction of the upper-lower bounds.

Figure 3 shows that UBCS has an obvious advantage over the other two algorithms in time complexity. The experiment was operated on a typical desktop computer. Comparing

TABLE 1: Comparison of scores based on the data set from UBCS, BENA, and MMHC.

Score	SS = 500	SS = 1000	SS = 2000	SS = 5000
UBCS	1.0000	1.0000	1.0000	1.0000
BENA	1.0801	1.0309	1.0286	0.9991
MMHC	1.2000	1.0981	1.0848	1.0029

FIGURE 3: Comparison of time complexity in different algorithms.

with MMHC, both BNEA and UBCS perform better in time complexity because of using MDP to reduce the dimensions of searching space. On the other hand, because MMHC which is involved in MMHC is used in BNEA, MMHC should have the same time complexity with BNEA, in the worst situation. While BNEA only involves 0-order and 1-order CI test, therefore it has better performance in time complexity.

5. Conclusion

We propose a hybrid method for Bayesian network structural learning (UBCS). In this method, two constructional algorithms are given to build the upper and lower bound of the BN structure and theoretical proof is completed as well. UGLA which is the first part of UBCS outputs upper bound of the moral graph of the target structure, while the following part named LGLA offers lower bound of the target structure's PDAG. Principle NZMI is also proved in this paper, which indicates the hidden information in 0-order CI test that could be used for reducing the reaching space. As only involving low order CI test, UBCS has an advantage in time complexity comparing with other hybrid learning methods, which is also supported by simulation results.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] M. Pourali and A. Mosleh, "A functional sensor placement optimization method for power systems health monitoring," *IEEE Transactions on Industry Applications*, vol. 49, no. 4, pp. 1711–1719, 2013.
- [2] C. T. C. Arsene and P. J. Lisboa, "Bayesian neural network applied in medical survival analysis of primary biliary cirrhosis," in *Proceedings of the 14th International Conference on Computer Modelling and Simulation (UKSim '12)*, pp. 81–85, IEEE, March 2012.
- [3] A. Hofleitner, R. Herring, P. Abbeel, and A. Bayen, "Learning the dynamics of arterial traffic from probe data using a dynamic bayesian network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 4, pp. 1679–1693, 2012.
- [4] D. M. Chickering, D. Geiger, and D. Heckerman, "Learning Bayesian networks is NP-hard," Tech. Rep. MSR-TR-94-17, Microsoft Research, 1994.
- [5] A. Carvalho, "A cooperative coevolutionary genetic algorithm for learning bayesian network structures," in *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation (GECCO '11)*, pp. 1131–1138, 2011.
- [6] A. Basak, I. Brinster, X. Ma et al., "Accelerating Bayesian network parameter learning using Hadoop and MapReduce," in *Proceedings of the 1st International Workshop on Big Data, Streams and Heterogeneous Source Mining: Algorithms, Systems, Programming Models and Applications*, pp. 101–108, ACM, 2012.
- [7] X. W. Chen, G. Anantha, and X. Wang, "An effective structure learning method for constructing gene networks," *Bioinformatics*, vol. 22, no. 11, pp. 1367–1374, 2006.
- [8] M. Kalisch and P. Bühlmann, "Estimating high-dimensional directed acyclic graphs with the PC-algorithm," *Journal of Machine Learning Research*, vol. 8, pp. 613–636, 2007.
- [9] D. Heckerman, D. Geiger, and D. M. Chickering, "Learning Bayesian networks: the combination of knowledge and statistical data," *Machine Learning*, vol. 20, no. 3, pp. 197–243, 1995.
- [10] G. F. Cooper and E. Herskovits, "A Bayesian method for the induction of probabilistic networks from data," *Machine Learning*, vol. 9, no. 4, pp. 309–347, 1992.
- [11] P. Spirtes, C. Glymour, and R. Scheines, *Causation, Prediction, and Search*, MIT Press, 2000.
- [12] L. M. de Campos and J. F. Huete, "Approximating causal orderings for Bayesian networks using genetic algorithms and simulated annealing," in *Proceedings of the 8th International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems (IPMU '00)*, vol. 1, pp. 333–340, 2000.
- [13] H.-Y. Jia, J. Chen, Y.-G. Zhu, and D.-Y. Liu, "A hybrid method for orienting edges of bayesian network," *Acta Electronica Sinica*, vol. 37, no. 8, pp. 1842–1847, 2009.
- [14] L. E. Brown, I. Tsamardinos, and C. F. Aliferis, "A novel algorithm for scalable and accurate Bayesian network learning," *Studies in Health Technology and Informatics*, vol. 11, pp. 711–715, 2004.
- [15] P. Spirtes and C. Meek, "Learning Bayesian networks with discrete variables from data," in *Proceedings of the 1st International Conference on Knowledge Discovery and Data Mining*, pp. 294–299, AAAI Press, Menlo Park, Calif, USA, 1995.
- [16] I. Tsamardinos, L. E. Brown, and C. F. Aliferis, "The max-min hill-climbing Bayesian network structure learning algorithm," *Machine Learning*, vol. 65, no. 1, pp. 31–78, 2006.
- [17] M.-M. Zhu, S.-Y. Liu, and Y.-L. Yang, "Structural learning Bayesian network equivalence classes based on a hybrid method," *Acta Electronica Sinica*, vol. 41, no. 1, pp. 98–104, 2013.
- [18] R. E. Neapolian, *Learning Bayesian Networks*, Prentice Hall, Englewood Cliffs, NJ, USA, 2004.
- [19] M. Frydenberg, "The chain graph Markov property," *Scandinavian Journal of Statistics*, vol. 17, no. 4, pp. 333–353, 1990.
- [20] T. Verma and J. Pearl, "An algorithm for deciding if a set of observed independencies has a causal explanation," in *Proceedings of the 8th Annual Conference on Uncertainty in Artificial Intelligence*, Stanford, Calif, USA, 1992.
- [21] R. W. Robinson, "Counting unlabeled acyclic digraphs," in *Combinatorial Mathematics V*, pp. 28–43, Springer, Berlin, Germany, 1977.
- [22] S. A. Andersson, D. Madigan, and M. D. Perlman, "A characterization of Markov equivalence classes for acyclic digraphs," *The Annals of Statistics*, vol. 25, no. 2, pp. 505–541, 1997.
- [23] D. Wu, "Maximal prime subgraph decomposition of Bayesian networks: a relational database perspective," *International Journal of Approximate Reasoning*, vol. 46, no. 2, pp. 334–345, 2007.