

Received 24 March 2022, accepted 4 May 2022, date of publication 30 May 2022, date of current version 30 June 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3178842

Bayesian Network Structure Learning Approach Based on Searching Local Structure of Strongly Connected Components

KUNHUA ZHONG^{ID}1,2,3, YUWEN CHEN^{ID}1,2,3, JU ZHANG²,
AND XIAOLIN QIN^{ID}1,3, (Member, IEEE)

¹Chengdu Institute of Computer Applications, Chinese Academy of Sciences, Chengdu 610041, China

²Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, Chongqing 400714, China

³University of Chinese Academy of Sciences, Beijing 100049, China

Corresponding author: Xiaolin Qin (qinxl@casit.ac.cn)

This work was supported in part by the National Key Research and Development Plan of China, under Grant 2018YFC0116704; in part by the Sichuan Science and Technology Program of China, under Grant 2019ZDZX0006, Grant 2020YFG0010, and Grant 2020YFQ0056; and in part by the Science and Technology Service Network Initiative, CAS, China, under Grant KFI-ST-S-QYZD-2021-21-001.

ABSTRACT Learning the structure of Bayesian networks is a challenging problem because it is a NP-Hard problem. As an excellent search & score based method, the K2 algorithm strongly depends on the input of global order of all nodes to ensure the result is a directed acyclic graph. And the K2 algorithm searches parents for each node from the nodes before it in the global order. Incorrect node order is likely to result in a wrong structure. In this paper, we propose a new method to avoid the global order by Local structure Searching for Strongly Connected Components with hill climbing method Twice. Firstly, we search the best parent nodes for each node from all the remaining nodes except itself, and form a global directed graph by concatenating the arcs between nodes and their parents. Then, the directed acyclic structure of all the strongly connected components are determined by hill climbing algorithm twice continuously. Finally, we adopt local search method further to get the final result by taking the previous result as a start point. The proposed algorithm is evaluated on several standard benchmark networks with sampled data. Experimental results show that our algorithm outperforms the four compared algorithms in terms of structural Hamming distance, Bayesian information criterion score and their average ranking.

INDEX TERMS Bayesian network, structure learning, hill climbing search, strongly connected component.

I. INTRODUCTION

As one of the important methods of uncertainty reasoning in artificial intelligence, Bayesian network (BN) qualitatively represents the dependence between random variables through directed acyclic graph (DAG), and quantifies this dependencies with conditional probability distribution, which has strong interpretability. Bayesian networks are widely used in various fields, including knowledge representation and reasoning [1], clinical decision-making [2], financial modeling & analysis [3], risk prediction [4], gene analysis [5], industrial application [6], etc.

A Bayesian network, also known as belief network, is a graphical model that consists of two parts: network structure and network parameters. The structure $G = \{V, E\}$ is a DAG, in which nodes $V = \{X_1, X_2, \dots, X_n\}$ represent

The associate editor coordinating the review of this manuscript and approving it for publication was Bo Jin^{ID}.

random variables in the problem domain, and directed arcs correspond to the qualitative characterization of the relationship between nodes. Each node has a conditional probability distribution, namely network parameters, associated with it, which quantitatively depicts the dependence between random variable nodes and their parent nodes. The dependencies satisfy the first-order Markov property. And the following expression holds:

$$P(X_1, X_2, \dots, X_n) = \prod_{i=1}^n P(X_i | Pa(X_i)), \quad (1)$$

where $P(X_i | Pa(X_i))$ is the conditional probability distribution between node X_i and its parent node set $Pa(X_i)$.

Bayesian network structure learning is a NP hard problem [7] and remains a challenging problem, due to the exponential growth of the space of valid networks with respect to the number of nodes. This paper studies the

structure learning method of discrete BNs under complete data. The data sets involved have no missing values. The existing approximate algorithms for BN structure learning from complete data include conditional independence (CI) test based method, score & search based method and the hybrid method.

K2 algorithm [8] and hill climbing (HC) algorithm [9] are classical and efficient score & search based methods. HC algorithm is simple and easy to implement, but it is easy to fall into local optimum. K2 algorithm is an efficient algorithm based on greedy strategy, which has very excellent learning performance. However, K2 algorithm requires the maximum number of parent nodes and depends heavily on the order of nodes. Incorrect node order will result in a very poor result. Moreover, the impact of node order on result structure is much greater than the maximum number of parent nodes. A large number of literatures [10]–[12] have been devoted to improve this drawback and achieved certain effect. In 2020, Behjati and Beigy [10] proposed a competitive method with good result in structure score, which regards strongly connected component (SCC) as a contracted node, learns the local order of SCC nodes by exact learning method with recursive strategy, and then forms the global order, which is used as the input of K2 algorithm. In essence, the aim of this approach is to build a better order of nodes for K2 algorithm. However, it has the following two disadvantages. Firstly, in extreme situation, if the global directed graph contains only one SCC with all the nodes in dataset, then the recursive call of the algorithm is invalid and will be an endless loop. Secondly, the topological order of DAG is not unique, and the number of topological orders increases with the number of nodes. However, only one order of nodes is selected in the algorithm.

The main contribution of this paper is to provide a Bayesian network structure learning approach from data based on maximum weighted spanning tree, bidirectional greedy search and SCC local structure search. First, we estimate the maximum number of parent nodes roughly and construct the initial network structure with directed graph. Then, we extract all the SCCs and search the local DAG structure of each SCC with HC algorithm twice. A global DAG can be formed by concatenating all the local DAGs. Finally, this global DAG, as a starting point, is passed to a local search method to learn the structure of Bayesian network. The algorithm proposed can avoid the global order of nodes.

The rest of the paper is organized as follows. Section 2 presents a brief review of Bayesian network structure learning methods. Section 3 devotes for introducing some preliminaries about the proposed approach. Then, in Section 4, the proposed algorithm is described in detail. And Section 5 is dedicated to the experimental evaluation. Finally, we conclude in Section 6.

II. BAYESIAN NETWORK STRUCTURE LEARNING

Bayesian network structure learning refers to combining the prior knowledge as much as possible to obtain the optimal

topological structure, including exact algorithms and approximate algorithms. The existing approximate learning methods can be divided into three categories: CI test based, score & search based and hybrid methods. Exact learning algorithms can guarantee the global optimal solution in theory, but it is complex and inefficient. Approximate learning algorithms trade the loss of accuracy for computational efficiency, and aim to obtain a suboptimal network structure in an acceptable computing time.

A. EXACT METHODS

Exact methods contain integer linear programming (ILP) methods [13], [14], A* algorithm [15], [16] and dynamic programming (DP) methods [17], [18]. Exact methods have excellent learning performance, but its efficiency is very low, so it is difficult to be applied to networks with a large number of nodes.

B. APPROXIMATE METHODS

1) CI TEST BASED METHODS

The CI test based methods [19] regard BN as a network model representing the relationship of independent variables. Usually, statistical or information theory methods are used to analyze the dependency relationship between variables by calculating the mutual information and conditional independence between nodes, and finally a network structure is got in line with them. Typical representatives of such methods are Shrink-Grow-Shrink (SGS) algorithm [20] and Peter-Clark (PC) algorithm [9]. This kind of method is more intuitive and effective, but it requires a large number of samples to determine the independence, resulting in unreliable CI test results. Moreover, the number of CI test times grows rapidly in nodes count.

2) SCORE & SEARCH BASED METHODS

The score & search based methods regard structure learning as an optimization problem, whose goal is to search the network structure with highest score. A score & search based algorithm can be formulated as follows: given a dataset $D = d_1, d_2, \dots, d_m$ of cases, find a DAG G^* such that

$$G^* = \arg \max_{G \in \Omega} \text{score}(G | D), \quad (2)$$

where Ω is the space of DAGs defined on node set $V = \{X_1, X_2, \dots, X_n\}$. From the perspective of algorithm, it is necessary to determine the appropriate search strategy and the scoring function to measure the fitting degree to dataset. The number of DAGs grows super-exponentially with the number of nodes. It has been proved that this is a NP-Hard problem [7]. Heuristic search methods are usually used to solve this problem which is easy to fall into the local optimum. The search for the network structure can be carried out in the DAG space, the equivalence class space, or the ordering space.

Search on the DAG space is the most direct mode. Each candidate DAG is assigned a network score reflecting its

goodness of fit, which the algorithm then attempts to maximize. Some examples are heuristics, such as hill climbing, genetic algorithms [21], whale optimization strategy [22], differential evolution algorithm [23] and Particle Swarm Optimization (PSO) [24]. K2 algorithm is a classical one of this kind with outstanding performance but needing pre-provided order of nodes.

In the search method based on equivalence class space, the points in the search space are no longer a single BN structure, but a class of structures with equivalence relationship. Two examples of this category are Greedy Equivalent Search (GES) [25] and Iterated Local Greedy Equivalent Search (ILGES) [26].

Ordering based search methods [27], [28] search in the space of node orders and construct BN structure based on the obtained result order. The fundamental observation of this kind of methods is that, finding the highest score structure is not NP-Hard given an ordering on the variables in the network.

3) HYBRID METHODS

Hybrid methods [29], [30] integrate the advantages of CI test based methods and score & search based methods for solving the structure learning problem, which reduce the size of structure space by CI test, and then adopt score & search methods further to obtain the optimal result.

III. PRELIMINARIES

In this section, we introduce several fundamental concepts about the proposed approach.

A. MUTUAL INFORMATION

Given random variables X and Y , the mutual information between them is defined as:

$$\begin{aligned} I(X; Y) &= \sum_{x \in X} \sum_{y \in Y} p(x, y) \log \frac{p(x, y)}{p(x)p(y)} \\ &= H(X) - H(X | Y), \end{aligned} \quad (3)$$

where $p(x, y)$ is the joint probability of x and y , $p(x)$ and $p(y)$ are the marginal probability distribution of X and Y respectively, $H(X)$ is the entropy of X , and $H(X | Y)$ is the conditional entropy of X given Y . Mutual information is a metric of the degree of interdependence between two random variables.

B. MAXIMUM WEIGHTED SPANNING TREE (MWST)

The spanning tree of a connected graph with N nodes is the minimal connected subgraph of the original graph, which contains all the N nodes, and has the least edges to keep the graph connected. The weight of a tree refers to the sum of the weights of each edge in the tree. The maximum weighted spanning tree is the spanning tree with maximal weight.

C. STRONGLY CONNECTED COMPONENT

In a directed graph, two vertices X and Y are said to be strongly connected, if there exists a path from X to Y , and

exists a path from Y to X meanwhile. The directed graph is strongly connected if each pair of vertices in the directed graph is strongly connected. A strongly connected component of a directed graph $G = \{V, E\}$ is a maximal set of vertices $C \subseteq V$ such that for every pair of vertices u and v in C , vertices u and v are strongly connected.

D. HILL CLIMBING ALGORITHM

Starting from an initial network structure, the hill climbing algorithm modifies the current network structure through three search operators (add one edge, delete one edge and reverse one edge) to obtain neighborhood points, and selects the network structure from the neighborhood with the highest score as the starting point of the next iteration until the local optimal network structure is obtained. The reason why hill climbing algorithm is easy to fall into local optimum lies in the selection of initial network structure. A good initial structure will lead to a better result.

IV. THE PROPOSED APPROACH

As mentioned above in section 1, the method proposed in [10], denoted as SCC-Order-K2, has two drawbacks. Firstly, in extreme situation, if the global directed graph contains only one SCC with all the nodes in dataset, then the recursive calling of the algorithm is invalid and will be an endless loop. Secondly, the topological order of a DAG is not unique, and the number of topological orders increases with the number of nodes. However, only one order of nodes is selected in the algorithm. In this section, to overcome them, we propose a novel approach based on local structure search for strongly connected components. In the following, we describe how the method proposed in this paper works.

A. METHOD DESCRIPTION

The method proposed is mainly based on the following ideas:

- 1) If a good initial value is provided for HC algorithm, the global optimal network structure can be obtained. In this paper, we are committed to finding a good initial value of HC algorithm to obtain the (approximate) optimal solution of Bayesian network structure learning problem.
- 2) It is difficult to learn the global Bayesian network structure directly, for which the divide and conquer strategy can be adopted. We can learn the local structures first, then fuse and revise them, and finally get the global network structure. Similar to the calculation of local scores, each node and its parent nodes form a local network structure. Greedy search method is an intuitive and feasible method to search the set of parent nodes.
- 3) The fused global network structure may not meet the characteristic of directed acyclic, that is, there exists one or more SCCs. The global network structure can be revised step by step. We can first revise each SCC to a directed acyclic graph. Here, we use the HC algorithm twice to make full use of the results of the

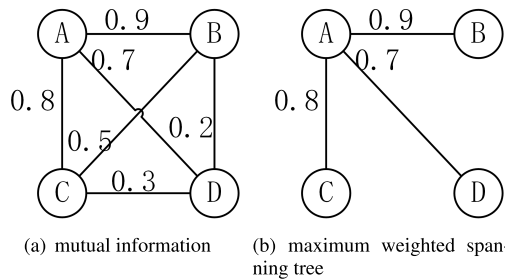


FIGURE 1. An example of estimating the maximal number of parent nodes.

previous steps. Local structure modification is also an optimization problem, but the scale of it is smaller.

- 4) The maximum number of parent nodes is an important information for many structure learning algorithms such as K2. If it is known, the amount of computation can be greatly reduced, and a more accurate structure with fewer wrong directed edges can be obtained.

MWST is an undirected graph representation of the association between nodes which has the minimum connection weight to keep the graph connected. The maximum degree of nodes in MWST can be used as a rough estimate of the maximum number of parent nodes, although this estimate is not optimal. Mutual information is a good metric for the dependent relation of random variables. Taking mutual information as the weight of edge, we can construct a maximum weighted spanning tree T for all the nodes in V , which is an undirected graph. For each node in T , there must be at least one node connected to it. The number of nodes connected to node X , is defined as the degree of it. And the maximum degree of a nodes in T can be used as the upper limit of the number of parent nodes. As shown in Fig. 1(a), we give an example of estimating the maximal number of parent nodes with a 4 nodes network. Fig. 1(b) shows the mutual information of each pair of nodes and Fig. 1(b) is its corresponding maximum weighted spanning tree. From Fig. 1(b) we can get the following information: $degree(A) = 3$, $degree(B) = degree(C) = degree(D) = 1$. Therefore, the rough estimate value of maximum number of parents is $\max(\{3, 1, 1, 1\}) = 3$.

In K2 algorithm, the parents of each node are selected from the previous nodes of it in the order through greedy search. To relax this restriction of global order, we can use greedy strategy to select the best parent node set from all the other nodes. And then, a global directed graph G_0 is formed by concatenating all the edges. However, it is likely to be no longer a DAG which, therefore, must be revised to meet the limitation of directed acyclic.

Following the approach in SCC-Order-K2 algorithm, we extract all the SCCs in the global directed graph G_0 . If each SCC is regarded as a contracted node, the global directed graph G_0 is a DAG in a broad sense, called SCC graph. A directed graph is acyclic if and only if it has no strongly connected subgraphs with more than one vertex. Therefore, we only need to revise the local structure of each

SCC to get a local directed acyclic subgraph. Instead of learning the order of nodes in each SCC, we learn the local directed acyclic structure of SCC directly by HC algorithm twice, because the edges in SCC contain some information which can be fully utilized. If we convert each directed edge in SCC to undirected one, the resulting undirected graph is a skeleton of SCC. Therefore, when using HC algorithm for the first time, this skeleton is used as the white list. And its result can be deemed as a point close to the global optimum for local structure. Then, we take it as the start DAG and adopt HC algorithm once again.

If we revise each SCC to a local DAG, and then expand the previous corresponding contracted node with the revision result, then the whole network will be a DAG. After local structures of all SCCs are determined, a global DAG \bar{G} can be obtained, which is a point close to the global optimum for global structure. Then, we employ a local search method further such as HC algorithm to get the final result of BN structure with \bar{G} as the initial point.

The proposed structure learning method for Bayesian networks, denoted as SCC-LST (SCC Local structure Searching with hill climbing method Twice), includes the following steps.

- 1) Compute the mutual information matrix.
- 2) Take mutual information as weight of edges, create the maximum weighted spanning tree, and compute the maximal number of parent nodes, denoted as μ .
- 3) Compute parents for each node with bidirectional greedy search under the limitation of μ , and form a global directed graph G_0 .
- 4) Search all the strongly connected components of G_0 and construct G^{SCC} by keeping the directed edge between SCCs and its outside.
- 5) Determine the local directed acyclic structure for each SCC by HC algorithm twice, and form the global DAG \bar{G} .
- 6) Adopt a local search method such as HC algorithm by taking \bar{G} as the initial point further to get the final result.

The differences between our method SCC-LST and the SCC-Order-K2 in [10] are as follows. (1) We employ the bidirectional greedy search to construct the initial directed graph network with the maximal number of parents obtained through maximum weighted spanning tree, while the Sparse Parent Graph Algorithm [16] is adopted in SCC-Order-K2 algorithm. (2) We learn the local directed acyclic structure of SCC by using HC algorithm twice, while the SCC-Order-K2 algorithm only learns the nodes ordering in SCC by exact learning algorithm recursively. (3) The SCC-Order-K2 algorithm regards SCC as a contracted node to form a DAG in a broad sense and obtain the global node order. However, the topological order of a DAG is not unique. SCC-Order-K2 algorithm does not select the best order among all the orders, which fails to make full use of the results of the previous steps. Our method directly constructs the local directed acyclic structure of SCC to avoid this problem. (4) In SCC-Order-K2 algorithm, the final result is obtained by K2 algorithm taking

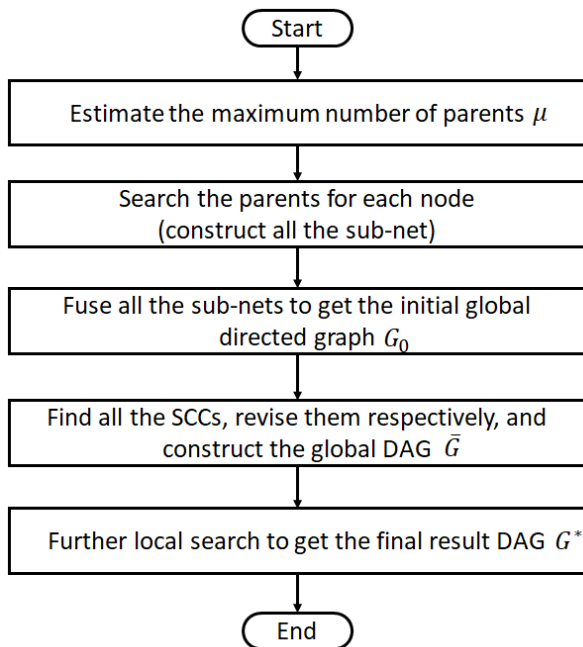


FIGURE 2. The flow chart of SCC-LST method.

the global order of nodes as an input value. While, in SCC-LST method, we get the final result by local search algorithm taking the global DAG structure as input instead of global order. (5) In extreme situation, there is only one SCC that contains all nodes. In this case, our method degenerates to a simple method using HC algorithm twice, while SCC-Order-K2 algorithm is invalid and falls into endless loop.

Fig. 2 shows the flow chart of our method. And the pseudo-code of the proposed algorithm is shown in Algorithm 1.

Algorithm 1 SCC-LST Algorithm

Input: Dataset D with m cases on nodes set $V = \{V_1, V_2, \dots, V_n\}$.

Output: a Bayesian network structure G^* .

1. compute the mutual information matrix.
2. compute the maximum weighted spanning tree.
3. $\mu \leftarrow \max_{1 \leq i \leq n} \deg(X_i)$ /* maximal number of parents */
4. initialize variable $edges$ as an empty set.
5. for $i \leftarrow 1$ to n do
6. $Pa(X_i) \leftarrow \text{bidirectional_greedy_search}(X_i, \mu)$
7. update $edges$ with directed edges (X_j, X_i) for all $X_j \in Pa(X_i)$
8. end
9. construct a global directed graph G_0 from $edges$.
10. search strongly connected components of G_0 : C_1, C_2, \dots, C_T .
11. construct G^{SCC} from G_0 .
12. for $i \leftarrow 1$ to T do
13. $ug_i \leftarrow \text{to_undirected_graph}(\text{subgraph}(C_i))$
14. $dg_i \leftarrow \text{to_directed_graph}(ug_i)$
15. $dag_scc_i \leftarrow \text{HC}(\text{white_list} = ug_i)$
16. $dag_scc_i \leftarrow \text{HC}(\text{start_dat} = dag_scc_i)$
17. end
18. $dag_global \leftarrow \text{concatenate}(\{dag_scc_i\}_{1 \leq i \leq T})$
19. $G^* \leftarrow \text{HC}(\text{start_dag} = dag_global)$
20. **return** G^* .

Based on the Bayesian network CHILD and its sampled data, we give a demonstration example of our method. For the sake of space, we put it in the Appendix.

TABLE 1. Networks used in the experiment.

Networks	Nodes	Arcs	Max Parents	Number	Network Scale
CANCER	5	4		2	Small
SURVEY	6	6		2	Small
ASIA	8	8		2	Small
CHILD	20	25		2	Medium
INSURANCE	27	52		3	Medium
ALARM	37	46		4	Medium
HAILFINDER	56	66		4	Large
HEPAR2	70	123		6	Large
WIN95PTS	76	112		7	Large

In the SCC-LST method proposed in this paper, the two steps of “finding candidate parents (sub-net construction)” and “SCC revision” are naturally suitable for parallel processing. In addition, on the one hand, due to the previous estimation of the maximum number of parent nodes, it provides very favorable information for the sub-net construction step, which can be utilized to save a lot of computing time. On the other hand, the number of nodes contained in each SCC will not exceed the number of nodes in the whole network, so its revision time is relatively less. Moreover, HC algorithm itself is a very fast method. Therefore, it can be anticipated that our method can be scaled to massive networks with hundreds and even thousands of nodes.

B. TIME AND SPACE COMPLEXITY

In this subsection, we briefly analyse the time and space complexity of our method.

The mutual information matrix is symmetric, and the computation time is $O(n)$, where n is the number of nodes. The time complexity of MWST computation is $O(n^2)$. Searching the maximum number of parent μ need $O(n)$ comparisons. Finding parents for each node with bidirectional greedy search under the limitation of μ requires $O(\mu n)$ score calculations, and all the nodes need $O(\mu n^2)$ score calculations in total. The fusion step to get G_0 requires $O(1)$ computations. Searching all the strongly connected components in a directed graph needs $O(n + e)$ calculations, where e is the number of edges in the graph. There are approximately $O(n^2)$ possible changes in HC algorithm, the score for the $O(n^2)$ initial changes are calculated firstly, after which each iteration requires $O(n)$ new calculations. And then, revision each SCC to get a local DAG by HC algorithm twice needs $O(r^2)$ score computations, where r is the maximum number of SCC nodes. So, revision all the SCCs requires approximately $O(nr)$ score computations. The time requirement for getting the global DAG \bar{G} is $O(1)$. And in the final step of further local search optimization, $O(n^2)$ score calculations are needed. Therefore, the total time complexity of our method SCC-LST is approximately $O(n^2)$ score calculations and additional $O(n^2)$ computations.

HC algorithm is a local search method, which can find a reasonable solution in a certain space with constant spatial complexity. For matrix related operations, the spatial complexity is $O(n^2)$. Therefore, the space complexity of our

TABLE 2. Results comparison with size=500.

(a) Structural Hamming Distance (SHD) with size=500						
Networks	HC	PC-HC	K2-RND	MWST-K2	OBS	SCC-LST
CANCER	3.2	3.2	6.1	6.6	3.9	3.1
SRUVEY	4.6	5.0	5.6	6.3	5.3	4.9
ASIA	4.3	6.4	11.5	9.9	4.8	5.1
CHILD	16.5	15.7	20.0	20.0	16.3	12.1
INSURANCE	42.2	41.4	48.7	54.7	46.3	38.2
ALARM	35.0	32.4	59.3	58.2	43.7	31.0
HAILFINDER	57.7	59.1	89.4	77.8	77.7	46.7
HEPAR2	108.9	110.8	141.6	139.3	134.2	107.6
WIN95PTS	104.3	101.0	145.6	152.6	141.7	110.7
(b) BIC score with size=500						
Networks	HC	PC-HC	K2-RND	MWST-K2	OBS	SCC-LST
CANCER	-1086.42	-1087.18	-1092.26	-1095.46	-1090.28	-1086.41
SRUVEY	-2002.88	-2003.33	-2014.17	-2018.68	-2011.87	-2002.85
ASIA	-1165.26	-1193.60	-1187.66	-1175.86	-1164.96	-1162.77
CHILD	-6741.08	-6925.65	-6825.71	-6934.49	-6766.34	-6710.48
INSURANCE	-7759.33	-8306.43	-8018.65	-8112.01	-7821.92	-7706.01
ALARM	-6309.14	-7234.45	-6859.51	-6769.09	-6491.17	-6313.95
HAILFINDER	-27828.92	-29383.07	-28719.6	-28784.68	-28209.65	-27825.96
HEPAR2	-16855.95	-17101.86	-17035.09	-17122.11	-16949.42	-16853.82
WIN95PTS	-5432.94	-6483.78	-5697.49	-5751.6	-5604.01	-5457.38

TABLE 3. Results comparison with size=5000.

(a) Structural Hamming Distance (SHD) with size=5000						
Networks	HC	PC-HC	K2-RND	MWST-K2	OBS	SCC-LST
CANCER	1.6	1.2	6.2	6.1	1.7	2.3
SRUVEY	4.2	5.1	5.0	5.2	4.6	4.1
ASIA	4.8	4.0	11.0	9.4	4.2	5.5
CHILD	10.4	11.0	17.7	20.0	13.3	2.5
INSURANCE	36.4	34.5	50.3	50.0	42.5	32.8
ALARM	32.2	18.9	64.1	68.9	46.9	19.7
HAILFINDER	60.9	55.9	108.5	79.6	72.6	40.6
HEPAR2	66.8	93.4	125.4	117.6	101.1	63.8
WIN95PTS	102.3	83.8	173.6	209.5	163.0	106.9
(b) BIC score with size=5000						
Networks	HC	PC-HC	K2-RND	MWST-K2	OBS	SCC-LST
CANCER	-10571.99	-10570.27	-10583.75	-10600.04	-10570.39	-10570.13
SRUVEY	-19836.11	-19874.27	-19854.15	-19862.08	-19839.27	-19835.17
ASIA	-11284.56	-11406.57	-11405.68	-11327.14	-11286.62	-11289.50
CHILD	-62281.00	-62796.54	-62821.23	-63490.94	-62464.23	-61682.27
INSURANCE	-68895.19	-75287.52	-71362.26	-71804.23	-69878.36	-68385.65
ALARM	-54965.06	-57969.55	-58334.75	-57366.09	-55955.51	-54411.92
HAILFINDER	-252302.32	-270616.34	-259694.30	-257408.61	-254026.24	-252291.79
HEPAR2	-163831.24	-165883.99	-164885.97	-165531.89	-164326.40	-163805.54
WIN95PTS	-47197.61	-58146.57	-49337.10	-50431.99	-48646.80	-47162.66

algorithm SCC-LST is a polynomial expression about the number of nodes.

V. EXPERIMENT EVALUATION

In this section, we aim to empirically evaluate the proposed method and compare it with 5 of the most popular algorithms: HC algorithm, PC-HC algorithm, K2 algorithm with random order (K2-RND), MWST-K2 algorithm and the OBS algorithm. PC-HC algorithm first obtains the skeleton of network via PC algorithm which is an undirected graph, and then orients the undirected edges with HC algorithm to obtain the final DAG structure. MWST-K2 algorithm adopts K2 algorithm with the order got from MWST. While the OBS algorithm is an ordering based search method by hill climbing search with tabu lists and random restart [27].

Our experiment is executed in the environment of PyCharm 2021.2.2 (Community Edition) with Python 3.9.7, on the platform of win 10 (64 bits) with Intel Core i9-10900 CPU @ 2.80GHz and 64G RAM.

A. DATASETS

The benchmark datasets include 9 Bayesian networks taken from the Bayesian network repository taken from the bnlearn R package [31]. We selected 3 small size networks (no more than 20 nodes), 3 medium size networks (20 – 50 nodes) and 3 large size networks (50 – 100 nodes). Table 1 shows the detailed information upon each network including the number of nodes, the number of directed edges (i.e. arcs), the number of maximal parent nodes, and the type of network. And for each network, we independently sample 10 datasets with 500, 5000 and 10000 cases respectively.

B. PERFORMANCE MEASURES

We evaluate the performance of our SCC-LST method in terms of structural Hamming distance (SHD) and the Bayesian information criterion (BIC) score of the result structure on the dataset. The SHD is sum of number of the correct edges, the additional edges, the missing edges and the

TABLE 4. Results comparison with size=10000.

(a) Structural Hamming Distance (SHD) with size=10000						
Networks	HC	PC-HC	K2-RND	MWST-K2	OBS	SCC-LST
CANCER	3.7	3.0	6.7	6.0	1.4	2.7
SRUVEY	3.0	5.3	4.9	5.0	3.3	3.2
ASIA	6.6	3.7	11.0	8.6	2.9	5.2
CHILD	10.7	10.7	18.0	19.7	13.8	5.4
INSURANCE	32.6	29.5	49.8	54.2	44.3	32.3
ALARM	30.2	17.3	66.1	76.0	47.5	23.0
HAILFINDER	57.3	49.3	114.7	80.3	77.2	37.7
HEPAR2	58.3	87.7	125.6	119.0	101.8	54.9
WIN95PTS	96.9	72.9	183.2	230.5	155.5	93.0
(b) BIC score with size=10000						
Networks	HC	PC-HC	K2-RND	MWST-K2	OBS	SCC-LST
CANCER	-21022.85	-21025.86	-21040.78	-21071.49	-21023.45	-21022.83
SRUVEY	-39619.60	-39758.49	-39636.47	-39648.70	-39620.53	-39616.94
ASIA	-22525.30	-22846.81	-22717.29	-22530.70	-22538.67	-22503.66
CHILD	-123791.31	-125410.85	-125102.09	-126050.48	-124434.92	-123162.56
INSURANCE	-135205.68	-142687.85	-139285.28	-139903.97	-136694.61	-134313.72
ALARM	-107814.92	-114248.76	-113096.17	-112082.07	-109172.74	-107300.99
HAILFINDER	-499243.51	-515302.46	-511720.01	-508042.43	-501893.34	-499340.08
HEPAR2	-326443.73	-329878.57	-328342.49	-329499.22	-327311.71	-326405.51
WIN95PTS	-92888.62	-112564.45	-96626.48	-98673.41	-94356.13	-92886.34

TABLE 5. Average ranking of algorithms.

Algorithms	Ranking of SHD			Ranking of BIC		
	500	5000	10000	500	5000	10000
HC	2.22	2.44	2.78	1.89	2.11	1.89
PC-HC	2.44	2.11	2.33	5.11	5.22	5.67
K2-RND	5.44	5.44	5.33	4.44	4.67	4.44
MWST-K2	5.44	5.44	5.44	5.22	4.89	4.78
OBS	3.67	3.56	3.22	3.11	2.89	3.11
SCC-LST	1.56	2.00	1.78	1.22	1.22	1.11

reverse edges compared with the original network structure. The smaller the SHD, the better, indicating that the obtained network structure is closer to the original network structure. On the other hand, the larger the BIC the better, which shows that the learned network structure can fit the data set better. In addition, we give the average ranking of the algorithms on SHD and BIC score.

C. EXPERIMENTAL RESULTS

In the following, we present the experimental results and compare them. Tables 2, 3 and 4 give the average SHD and BIC score of 10 datasets on each network with sample size 500, 5000 and 10000 respectively. The results highlighted in bold are the best one for the corresponding network. By a quick look at the results reported in the 3 tables, we can draw the conclusion that the proposed SCC-LST method outperforms the other algorithms, especially with respect to BIC score.

Table 2 shows the comparison result of the 6 methods with 500 samples. Of the nine networks, our algorithm has the best SHD on six of them, especially in the 3 medium networks. The SHD of our method in the three remaining networks is also very close to the best comparison algorithm. In BIC score, our algorithm has more advantages. It is only slightly inferior to HC algorithm on ALARM and WIN95PTS networks.

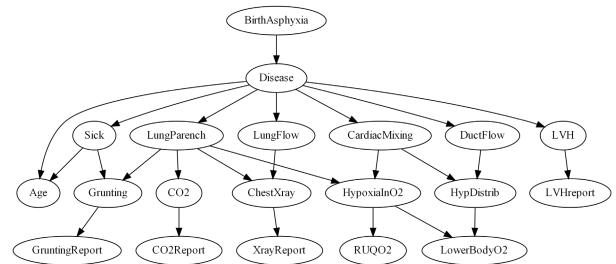


FIGURE 3. True structure of the CHILD network.

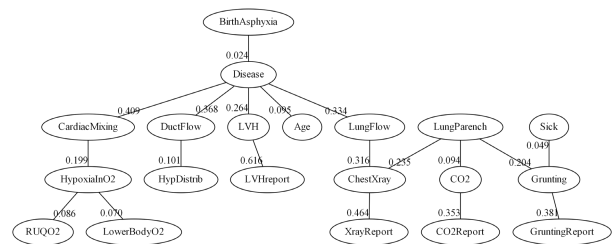


FIGURE 4. Maximum weighted spanning tree of CHILD.

Table 3 gives the comparison result about 5000 sampled data. Also, it shows the good performance of our algorithm. With respect to SHD, our algorithm is slightly better than PC-HC algorithm, but it is obviously better than the other 4 algorithms. Moreover, in terms of BIC score, our algorithm is not as good as HC algorithm only on the ASIA network. And we get the best BIC score in all the other 8 networks.

From the comparison result of 10000 samples in Table 4, we can get that our algorithm does not have much advantage in SHD, and its performance is equivalent to that of the PC-HC algorithm. However, our algorithm is still significantly better than the other 5 algorithms in BIC score.

In order to further illustrate the performance of our algorithm, we give the average ranking of six algorithms on SHD and BIC in Table 5. And the results show that the average ranking of our SCC-LST method is always the highest.

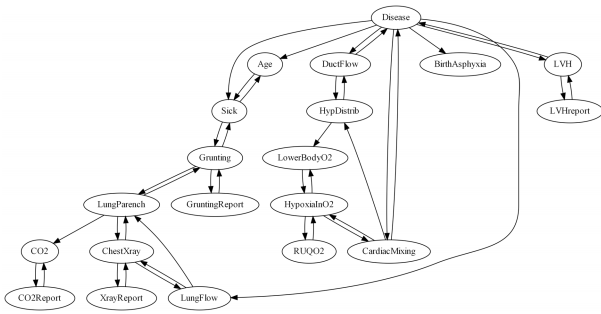


FIGURE 5. Fusion result of CHILD.

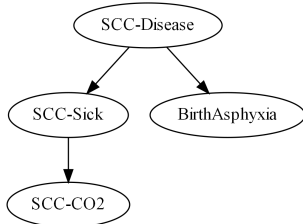


FIGURE 6. The SCC graph.

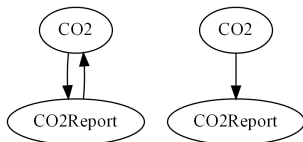


FIGURE 7. SCC-CO2 (left) and its revised result (right).

VI. CONCLUSION

In this paper, based on searching local structure of strongly connected components, we proposed a simple algorithm SCC-LST for learning Bayesian network structure from data that can avoid the global order of nodes. The proposed algorithm estimates the maximal number of parents through maximum weighted spanning tree firstly. And then the initial global directed graph is constructed by searching parents for each node from the remaining nodes except itself. After that, we search the local directed acyclic structure of strongly connected components by hill climbing search twice. Finally, a further local search process is used to get the final DAG as the structure of Bayesian network, taking the previous result as input.

We evaluated the proposed method on 9 benchmark networks by sampled datasets, in terms of SHD and BIC score and its average ranking. Experimental results showed that the proposed method outperforms the five compared algorithms.

However, two research issues remain. The quality of the parent nodes searched by bidirectional greedy search has a great impact on the performance of the proposed method. Also, when determining the local directed structure of SCC, only the internal nodes and directed edges of SCC are used, and its dependence on the outside is not fully utilized. In future, we will develop new approaches to overcome these drawbacks.

Bayesian network is one of the important methods of knowledge representation and reasoning in uncertainty

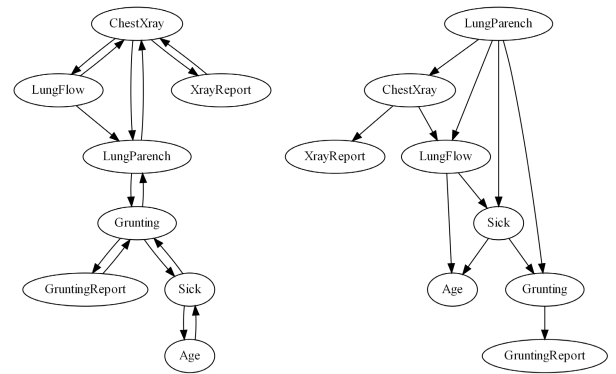


FIGURE 8. SCC-Sick (left) and its revised result (right).

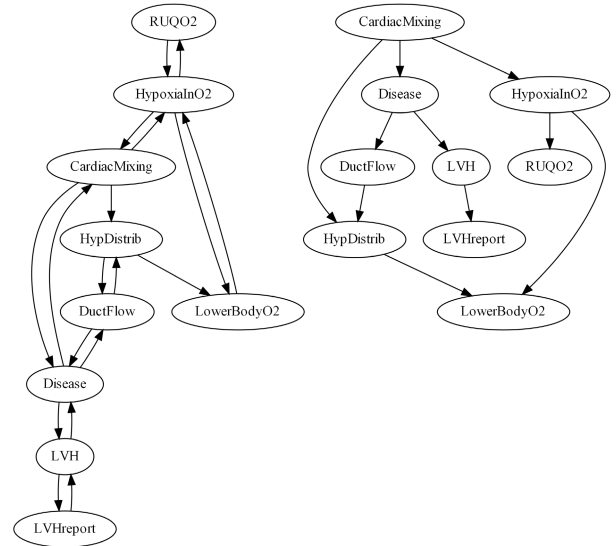


FIGURE 9. SCC-Disease (left) and its revised result (right).

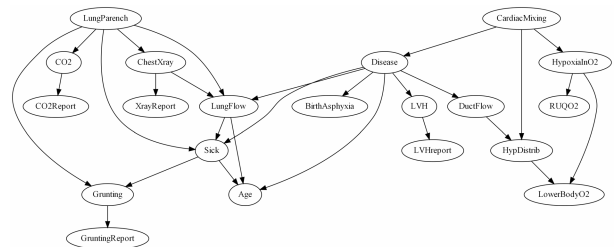


FIGURE 10. Global DAG after SCCs revision.

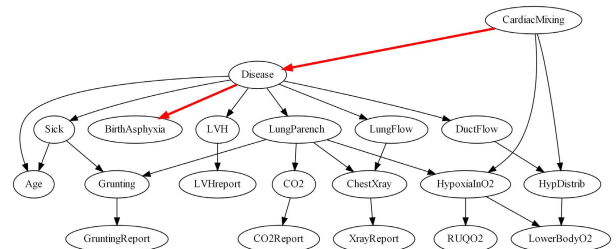


FIGURE 11. Final learned structure of CHILD.

problem, which has a wide range of applications, especially in the medical field. In the near future, we intend to apply Bayesian network to the actual scene of medical and clinical diagnosis.

APPENDIX

DEMONSTRATION EXAMPLE OF OUR METHOD

Based on the Bayesian network CHILD, we give a demonstration example to further illustrate the procedure of our method SCC-LST, as shown in Figs. 3 to 11. The CHILD network is a medium Bayesian network with 20 nodes and 25 directed edges. The maximum number of parent nodes is 2. In this demonstration example, we learn the Bayesian network structure from 5000 sampled data of the original network.

Fig. 3 shows the original true structure of the CHILD network. Fig. 4 is the maximum weighted spanning tree based on mutual information, from which we can get that the maximum number of parent nodes is 6 (pay attention to node: Disease). Fig. 5 gives the fusion result graph G_0 of each node and its parents. There exist three SCCs in Fig. 5: SCC-CO2, SCC-Sick and SCC-Disease. SCC-CO2 contains 2 node: CO2 and CO2Report. SCC-Sick contains 8 nodes: Sick, Age, Grunting, GruntingReport, LungParench, LungFlow, ChestXray and XrayReport. While SCC-Disease contains 9 nodes: RUQO2, HypoxiaInO2, CardiacMixing, HypDistrib, LowerBodyO2, DuctFlow, Disease, LVH and LVHReport. The SCC graph G^{SCC} is shown in Fig. 6. There is no intersection between any two of the three SCCs. And the node BirthAsphyxia is not in any SCC. The three SCCs and their revised directed acyclic structures are shown in Figs. 7, 8 and 9 respectively. The global DAG \bar{G} after all the SCCs are revised is presented in Fig. 10. And Fig. 11 gives the final learned structure of CHILD, in which the black arrows represent the completely correct directed edges while the bold red ones are the reversed edges. There are only two reversed directed edges: (Disease, BirthAsphyxia) and (CardiacMixing, Disease). In this demonstration example, there is no missing and additional edge.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions.

REFERENCES

- [1] B. Anderson, "Using Bayesian networks to perform reject inference," *Expert Syst. Appl.*, vol. 137, pp. 349–356, Dec. 2019.
- [2] E. Kyrimi, S. Mossadegh, N. Tai, and W. Marsh, "An incremental explanation of inference in Bayesian networks for increasing model trustworthiness and supporting clinical decision making," *Artif. Intell. Med.*, vol. 103, Mar. 2020, Art. no. 101812.
- [3] O. Chatrabgoun, A. Hosseinian-Far, V. Chang, N. G. Stocks, and A. Daneshkhan, "Approximating non-Gaussian Bayesian networks using minimum information vine model with applications in financial modelling," *J. Comput. Sci.*, vol. 24, pp. 266–276, Jan. 2018.
- [4] P. Arora, D. Boyne, J. J. Slater, A. Gupta, D. R. Brenner, and M. J. Druzdzel, "Bayesian networks for risk prediction using real-world data: A tool for precision medicine," *Value Health*, vol. 22, no. 4, pp. 439–445, Apr. 2019.
- [5] E. S. Adabor and G. K. Acquah-Mensah, "Restricted-derestricted dynamic Bayesian network inference of transcriptional regulatory relationships among genes in cancer," *Comput. Biol. Chem.*, vol. 79, pp. 155–164, Apr. 2019.
- [6] T. Adedipe, M. Shafiee, and E. Zio, "Bayesian network modelling for the wind energy industry: An overview," *Rel. Eng. Syst. Saf.*, vol. 202, Oct. 2020, Art. no. 107053.
- [7] 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, Oct. 2004.
- [8] G. F. Cooper and E. Herskovits, "A Bayesian method for the induction of probabilistic networks from data," *Mach. Learn.*, vol. 9, no. 4, pp. 309–347, Oct. 1992.
- [9] M. Scutari, C. E. Graafland, and J. M. Gutiérrez, "Who learns better Bayesian network structures: Accuracy and speed of structure learning algorithms," *Int. J. Approx. Reasoning*, vol. 115, pp. 235–253, Dec. 2019.
- [10] S. Behjati and H. Beigy, "Improved k2 algorithm for Bayesian network structure learning," *Eng. Appl. Artif. Intell.*, vol. 91, May 2020, Art. no. 103617.
- [11] X. Zheng, B. Aragam, P. Ravikumar, and E. P. Xing, "DAGs with, no. TEARS: Continuous optimization for structure learning," 2018, *arXiv:1803.01422*.
- [12] V. R. Tabar, F. Eskandari, H. Zareifard, and S. Salimi, "Finding a set of candidate parents using dependency criterion for the K2 algorithm," *Pattern Recognit. Lett.*, vol. 111, no. 1, pp. 23–29, 2018.
- [13] M. Bartlett and J. Cussens, "Integer linear programming for the Bayesian network structure learning problem," *Artif. Intell.*, vol. 244, pp. 258–271, Mar. 2017.
- [14] S. Kucukyavuz, A. Shojaie, H. Manzour, L. Wei, and H.-H. Wu, "Consistent second-order conic integer programming for learning Bayesian networks," 2020, *arXiv:2005.14346*.
- [15] C. H. Yuan, "Learning optimal Bayesian networks using A* search," in *Proc. 22nd Int. Joint Conf. Artif. Intell.*, Barcelona, Spain, Jul. 2011, pp. 2186–2191.
- [16] C. Yuan and B. Malone, "Learning optimal Bayesian networks: A shortest path perspective," *J. Artif. Intell. Res.*, vol. 48, pp. 23–65, Oct. 2013.
- [17] Y. X. Tan, X. Gao, and C. He, "Learning optimal Bayesian network structure constrained with Markov blanket," *Acta Electronica Sinica*, vol. 47, no. 9, pp. 1898–1904, Sep. 2019.
- [18] X. G. Gao, C. F. Wang, and R. H. Di, "A block learning algorithm with improved K-means algorithm for learning sparse BN optimal structure," *Acta Automatica Sinica*, vol. 46, no. 5, pp. 923–933, May 2020.
- [19] Z. G. Lv, "Overview of Bayesian network structure learning," *J. Xi'an Technological Univ.*, vol. 41, no. 1, pp. 1–17, Feb. 2021.
- [20] P. Spirtes, C. N. Glymour, R. Scheines, and D. Heckerman, *Causation, Prediction, and Search*, 2nd ed. Cambridge, U.K.: MIT Press, 2000.
- [21] Z. Yan, Z. Peng, and X. Chuan, "Research on Bayesian network structure learning algorithm based on Jaya," *Comput. Eng. Appl.*, vol. 55, no. 19, pp. 173–177, 2019.
- [22] H. R. Liu, Z. Liyue, F. A. N. Ruixing, W. Haiyu, and Z. Chunlan, "Bayesian network structure learning based on improved whale optimization strategy," *J. Electron. Inf. Technol.*, vol. 41, no. 6, pp. 1434–1441, Jun. 2019.
- [23] B. Liu, F. A. N. Ruixing, L. I. U. Haoran, H. Wang, and L. Zhang, "Bayesian network structure learning algorithm based on hybrid binary salp swarm-differential evolution algorithm," *J. Commun.*, vol. 40, no. 7, pp. 151–161, Jul. 2019.
- [24] J. W. Li, L. I. Guo-Ning, and Z. Ding, "Application of CS-PSO algorithm in Bayesian network structure learning," *J. Meas. Sci. Instrum.*, vol. 11, no. 1, pp. 94–102, 2020.
- [25] D. M. Chickering, "Optimal structure identification with greedy search," *J. Mach. Learn. Res.*, vol. 3, pp. 507–552, Nov. 2002.
- [26] J. I. Alonso, L. de la Ossa, J. A. Gámez, and J. M. Puerta, "On the use of local search heuristics to improve GES-based Bayesian network learning," *Appl. Soft Comput.*, vol. 64, pp. 366–376, Mar. 2018.
- [27] M. Teyssier and D. Koller, "Ordering-based search: A simple and effective algorithm for learning Bayesian networks," 2012, *arXiv:1207.1429*.
- [28] B. Liu, H. Wang, M. Sun, H. Liu, Y. Yu, and C. Hang, "Learning Bayesian network structure from node ordering searching optimal," *J. Electron. Inf. Technol.*, vol. 40, no. 5, pp. 1234–1241, May 2018.
- [29] 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.
- [30] J. G. Dai, "BN structure adaptive learning algorithm based on dual-scale constraint model," *Acta Automatica Sinica*, vol. 47, no. 8, pp. 1988–2001, Aug. 2021.
- [31] *BNLearn—An R Package for Bayesian Network Learning and Inference*. Accessed: Jan. 16, 2022. [Online]. Available: <https://www.bnlearn.com/bnrepository/>

• • •