Sequential Bayesian Network Structure Learning

Sachini Piyoni Ekanayake*, Daphney–Stavroula Zois*
*Department of Electrical and Computer Engineering
University at Albany, SUNY, Albany, NY, USA
Emails: {sekanayake, dzois}@albany.edu

2022 56th Asilomar Conference on Signals, Systems, and Computers | 978-1-6654-5906-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/IEEECONF56349.2022.10052016

***Abstract*—In many real–world applications, e.g., medical diagnosis, behavioral analysis, Bayesian networks are used to describe relationships between variables. In this context, a very important task is learning the underlying structure of such networks. However, this constitutes an NP–hard problem. In this paper, we propose an approach to speed–up the structure learning process without compromising accuracy, assuming a given set of candidate network structures. Specifically, the proposed method sequentially evaluates variable relationships until it reaches a specific decision regarding the underlying Bayesian network. The performance of the proposed approach is illustrated on two standard Bayesian networks and compared with existing methods.**

***Index Terms*—Bayesian networks, structure learning, sequential evaluation, hypothesis space, mutual information**

I. Introduction

Bayesian networks are typically used to describe relationships between variables of interest in a specific domain [1], [2]. They are described by directed acyclic graphs (DAGs), where variables are represented as nodes and relationships between them are denoted by directed edges. However, the exact structure of the underlying DAG is not known in many application domains (e.g., medical diagnosis [3]–[6], behavioral analysis [7], speech recognition [8]). Thus, an increased interest in Bayesian network structure learning [2], [9], [10], where the goal is to identify the structure of the Bayesian network that *best* describes the relationships between variables, has emerged. In this context, domain knowledge and/or existing data can be used to learn the structure of a Bayesian network [2], [9], [11]–[13]. Nevertheless, using expert knowledge can be a time–consuming task [11], [12], [14]. On the other hand, the total number of possible DAGs grows exponentially with the number of variables of interest [15]. Therefore, the task of Bayesian network structure learning is NP–hard [10], [16].

Various heuristic methods [10], [17], [18] have been proposed and typically used due to the complexity of the structure learning problem. For instance, Hill–Climbing [9] constitutes a greedy algorithm that learns a structure by maximizing a given score (e.g., the Bayesian Information Criterion). At the same time, various methods have been proposed to speed up the process of structure learning from data within a restricted search space [10], [17]–[19]. For example, the Chow–Liu algorithm [13] learns a tree–structured Bayesian network, where the result is sub–optimal. Nonetheless, it is widely used in many applications due to its low complexity [20]. In [18] – [19], the search space is reduced to the ordering of variables, while in [12], expert knowledge is employed. In many real–world applications (e.g., monitoring patients in the medical domain [21]), expert knowledge is typically available, but expert elicitation is expensive and time–consuming [11].

In this paper, we propose an approach to speed–up structure learning without compromising accuracy assuming a given set of candidate network structures. Specifically, we consider the set of candidate network structures as our hypothesis space and devise a method that sequentially evaluates possible variable relationships. The goal is to accurately learn the underlying Bayesian network structure from existing data in less time by evaluating the most prominent relationships early on. The performance of the proposed approach is validated on two standard Bayesian networks and compared with that of existing structure learning methods. Experimental results indicate that the proposed approach outperforms existing methods with respect to various accuracy metrics and graph learning time.

II. Problem Description

We consider a set $\mathcal{X} \triangleq \{X_1, X_2, \ldots, X_n\}$ of n discrete random variables with a joint probability distribution P^*. Here, $X_i, i \in \{1, 2, \ldots, n\}$, are categorical random variables, with the simplest case being binary–valued. We assume there is a gold standard Bayesian network $G^* = (\mathcal{V}, \mathcal{E})$ induced by the joint distribution P^*. G^* is a Directed Acyclic Graph (DAG) having discrete random variables $\mathcal{V}$ as nodes. $\mathcal{E}$ is the set of directed edges that represent relationships between variables in G^*. We assume that we have access to a dataset D consisting of S instances of observations of $\mathcal{X}$ generated from distribution P^*. We also assume that the dataset D is complete, i.e., there is no missing data. We consider a finite number M of hypotheses $G_i, i \in \{1, 2, \ldots, M\}$, each of which corresponds to a single DAG. We denote as $E \triangleq \{E_1, \ldots, E_K\}$, where $K = n(n-1)/2$, the set of distinct relationships[1] between any two random variables in $\mathcal{X}$. We propose to sequentially select variable relationships from the set E based on their significance (see Section IV) and assess their suitability as part of the underlying Bayesian network structure given dataset D. To this end, we consider two types of costs. Specifically, we assume cost coefficient $e_k > 0$, $k = 1, \ldots, K$, that represents the value of time and effort spent in evaluating variable

This material is based upon work supported by the National Science Foundation under Grants ECCS–1737443 & CNS–1942330. We would like to thank Dr. Yasitha Warahena Liyanage for his suggestions.

[1]Directionality is not considered in the definition of E. For instance, $X_1 \rightarrow X_2$ and $X_2 \rightarrow X_1$ are considered as a single distinct relationship $X_2 - X_1$.

relationships. We also consider misclassification costs $C_{ij} > 0, i, j = 1, 2, \ldots, M$, where C_{ij} denotes the cost of selecting hypothesis G_j when the true hypothesis is G_i. Our objective is to identify which of the graphs in $\mathcal{G} \triangleq \{G_1, \ldots, G_M\}$ *best* describes the underlying distribution of the dataset D, while balancing between accuracy and graph learning time.

A. Optimization Problem

First, we provide some important definitions. Here, $G_i, i \in \{1, \ldots, M\}$, is considered as a collection of information $I_D^{G_i}$ distributed among the edges that represents the relationships between variables $E_k, k \in \{1, \ldots, K\}$. Consider $I_D^{E_k}$ as the information content in E_k given the dataset D. Therefore, we define $I_D^{G_i} \triangleq \sum_{\forall E_i \in E^{G_i}} I_D^{E_i}$, where $E^{G_i} \subseteq E$. The probability of the relationship $E_k, k \in \{1, \ldots, K\}$, given the graph G_i, is denoted as $P(E_k|G_i)$. This is proportional to the relative strength of E_k within G_i and defined as $P(E_k|G_i) \triangleq \frac{I_D^{E_k}}{I_D^{G_i}}$ if $E_k \in E^{G_i}$ and 0 otherwise. First, we prove that $P(E_k|G_i) \in [0, 1]$. Specifically, for $E_k \in E^{G_i}$, $I_D^{E_k} > 0$, and thus, $\sum_{\forall E_k \in E^{G_i}} I_D^{E_k} > 0$. On the other hand, for $E_k \notin E^{G_i}$, $P(E_k|G_i) = 0$ by definition. Combining these two results shows that $P(E_k|G_i) \geq 0$. Furthermore, when $E_k \in E^{G_i}$, $I_D^{E_k} \leq \sum_{\forall E_i \in E^{G_i}} I_D^{E_k}$. Here, the equality holds when G_i has only one relationship and it is E_k. The last inequality suggests that $P(E_k|G_i) \leq 1$. Therefore, $P(E_k|G_i) \in [0, 1]$ holds. Next we prove that $\sum_{k=0}^{K} P(E_k|G_i) = 1$ as follows:

$$\begin{aligned}\sum_{k=0}^{K} P(E_k|G_i) &= \sum_{\forall E_k \in E^{G_i}} \left(\frac{I_D^{E_i}}{I_D^{G_i}}\right) + \sum_{\forall E_k \notin E^{G_i}} (0), \\ &= \frac{\sum_{\forall E_k \in E^{G_i}} I_D^{E_k}}{I_D^{G_i}}, \\ &= \frac{I_D^{G_i}}{I_D^{G_i}} = 1. \end{aligned} \tag{1}$$

Finally, we assume that $E_k, k \in \{1, \ldots, K\}$, are conditionally independent given the total information content of $G_i, i \in \{1, \ldots, M\}$, as each E_k represents a distinct relationship between two variables in G_i.

Consider a pair (R, D_R) of random variables associated with the sequential evaluation process of $E_k, k \in \{1, \ldots, K\}$. Random variable $R \in \{0, \ldots, K\}$ represents the last variable relationship selected from the set E. Random variable $D_R \in \{1, \ldots, M\}$ denotes the decision of selecting DAG $\tilde{G}_{D_R}$ among the M possible choices. Given our previously stated objective, our goal is to minimize the following optimization function with respect to R and D_R:

$$J(R, D_R) = \mathbb{E}\left[\sum_{k=1}^{R} e_k\right] + \sum_{j=1}^{M}\sum_{i=1}^{M} C_{ij} P(D_R = j, G_i), \tag{2}$$

where $P(D_R = j, G_i)$ represents the joint probability of selecting graph G_j while the true graph is G_i. In Eq. (2), the first term denotes the total cost of evaluating R variable relationships in the sequential process, while the second term penalizes DAG decisions.

III. Optimum Solution

To minimize the cost function in Eq. (2), we first find the optimum decision D_R^* for a given R. Then, the reduced cost function $J(R)$ depends only on R. Finally, we find the optimum R^* by minimizing $J(R)$. We refer to R^* and D_R^* as optimum stopping and decision strategies, respectively.

Consider the posterior probability $\pi_k \triangleq [\pi_k^1, \ldots, \pi_k^M]^T$ after evaluating k out of K distinct variable relationships. The probability $\pi_k^i \triangleq P(G_i|E_1, \ldots, E_k)$ denotes the posterior probability of the hypothesis $G_i, i = \{1, \ldots, M\}$. At stage $k = 0, \pi_0 \triangleq [p_1, \ldots, p_M]^T$, where $p_i \triangleq P(G_i), i = 1, 2, \ldots, M$. From Bayes' rule, as more variable relationships are evaluated, the posterior probability π_k^i is recursively updated as follows:

$$\pi_k^i = \frac{P(E_k|G_i)\pi_{k-1}^i}{P(E_k|G_1)\pi_{k-1}^1 + \ldots + P(E_k|G_M)\pi_{k-1}^M}. \tag{3}$$

Eq. (2) can be rewritten in terms of the posterior probability and the indicator function $\mathbb{1}_A$ (i.e., $\mathbb{1}_A \triangleq 1$ when event A occurs, and 0 otherwise) as follows:

$$J(R, D_R) = \mathbb{E}\left[\sum_{k=1}^{R} e_k + \sum_{j=1}^{M}\sum_{i=1}^{M} C_{ij}\pi_R^i \mathbb{1}_{D_R=j}\right]. \tag{4}$$

The optimum decision strategy D_R^* for any R can be found by minimizing the expected DAG decision cost as:

$$D_R^* = \underset{1 \leq j \leq M}{\operatorname{argmin}} \left[\mathbf{C}_j^T \pi_R\right], \tag{5}$$

where $\mathbf{C}_j \triangleq [C_{1j}, C_{2j}, \ldots, C_{Mj}]^T$. As a result, the cost function in Eq. (4) can be written as:

$$J(R) = \mathbb{E}\left[\sum_{k=1}^{R} e_k + g(\pi_R)\right], \tag{6}$$

where $g(\pi_R) \triangleq \min_{1 \leq j \leq M}[\mathbf{C}_j^T \pi_R]$.

Finally, the optimum stopping strategy R^* can be found by minimizing the cost function in Eq. (6) via dynamic programming [22]. Specifically, since there are K variable relationships, there are maximum $K+1$ stages for the associated dynamic programming equations:

$$L_k(\pi_k) = \min\left[g(\pi_k), \tilde{L}_k(\pi_k)\right], k = 0, \ldots, K-1, \tag{7}$$

where

$$\tilde{L}_k(\pi_k) = e_{k+1} + \sum_{E_{k+1}} L_{k+1}(\pi_{k+1})\Delta_{k+1}^T(E_{k+1})\pi_k, \tag{8}$$

with $\Delta_k(E_k) \triangleq [P(E_k|G_1), \ldots, P(E_k|G_M)]^T$ and $L_K(\pi_K) = g(\pi_K)$.

IV. Proposed Approach

In this section, we outline an approach that exploits the results of Section III to identify the graph $\tilde{G}$ out of the given hypothesis space $\mathcal{G}$ that *best* describes a dataset D. Our proposed approach consists of a preprocessing phase and a graph learning phase. During the preprocessing phase, the optimum decision and stopping strategies described by Eqs. (5)

and (7) are solved offline. Specifically, quantizing the interval $[0, 1]$ such that $\sum_{i=1}^{M} \pi_k^i = 1$, a $K \times d$ matrix is generated, where d is the number of possible π_k vectors, and used to numerically solve Eqs. (5) and (7). We use mutual information (MI) to estimate the importance of variable relationships in E and appropriately rank them. First, let us recall the definition of MI [23]. Specifically, consider a variable relationship $E_k, k \in \{1, \ldots, K\}$, between two variables $\{X_u, X_v\} \in \mathcal{X}$, where u, $v \in \{1, \ldots, n\}$. $I_D^{E_k}$ is calculated using MI as follows:

$$I_D^{E_k} \triangleq \sum_{X_u, X_v} P_D(X_u, X_v) \log \frac{P_D(X_u, X_v)}{P_D(X_u) P_D(X_v)}. \quad (9)$$

Then, the variable relationships in E are ordered in descending order of MI. Further, MI is used to estimate $P(E_k|G_i)$ (see Section II-A). Finally, during the preprocessing phase, we assume all hypotheses are equally likely, i.e., $P(G_i) = \frac{1}{M}, i = 1, \ldots, M$.

During the graph learning phase, the numerical solutions found during preprocessing are employed to select and assess the suitability of variable relationships, and identify the graph $\tilde{G}$ that best describes a dataset D. Specifically, our proposed approach begins by initializing the posterior probability $\pi_0 \triangleq [p_1, \ldots, p_M]^T, p_i = \frac{1}{M}, i = 1, \ldots, M$. Next, during the evaluation of the kth relationship, the two terms inside the minimization operator of Eq. (7) are compared. If the cost of selecting and assessing the suitability of a particular relationship ($\tilde{L}_k(\pi_k)$) is less than the cost of making a decision ($g(\pi_k)$), an appropriate relationship is selected and the posterior probability π_{k+1} is updated through Eq. (3). This process is repeated until a decision is reached by selecting a subset of the available relationships or the suitability of all variable relationships in E is assessed. The estimated $\tilde{G}$ is identified using the final posterior probability in conjunction with the optimum decision strategy of Eq. (5). Specifically, at each step k, relationships E_k are evaluated in descending order of information content. If a hypothesis G_i contains E_k, $P(E_k|G_i) > 0$ and $P(E_k|G_i) = 0$ otherwise (see Section II-A). Then, the hypotheses $G_i, i \in \{1, \ldots, M\}$, that contain already selected variables relationships are assigned a higher posterior probability π_{k+1} compared to the rest. This process continues until the optimum stopping stage, where the optimum decision $\tilde{G}$ is obtained such that the expected DAG decision cost is minimum. Note that the proposed approach assesses if a relationship (irrespective of orientation) belongs to G^* or not. Edge orientation is considered during hypothesis space definition, where only DAGs are included. When our approach stops evaluating relationships, it selects the hypothesis with the minimum DAG decision cost (Eq. (5)). Since hypotheses that contain the same set of relationships, but with different orientations, have the same DAG decision cost, the proposed approach randomly selects between them.

V. Experimental Results

In this section, we present experimental results to illustrate the performance of our proposed approach. Specifically, to evaluate its performance, we consider two standard Bayesian networks (see Fig. 1), the *Student* network [9] and the *Sprinkler* network [24], and use the associated probability distributions to randomly generate datasets with $1,000$ samples each. Since hypothesis space selection is out of the scope of this work, we consider a reduced hypothesis space $\mathcal{G}$ consisting of graphs with: (i) $\mathcal{N}(G_i) < \mathcal{N}(G^*)$, (ii) $\mathcal{N}(G_i) > \mathcal{N}(G^*)$, (iii) $\mathcal{N}(G_i) = \mathcal{N}(G^*)$, and (iv) the gold standard network, G^*, where $\mathcal{N}(G_i)$ denotes the number of edges in graph G_i. In our experiments, we set $M = 6$, assume misclassification costs $C_{ij} = 1, \forall i \neq j$, and $C_{ii} = 0, \forall i, j \in \{1, \ldots, M\}$, and same variable relationship evaluation costs, i.e., $e_k = e$. We assess performance with respect to the following metrics: (i) *Correct Edges* (CE): number of edges present in both $\tilde{G}$ and G^* with correct orientation (higher the better), (ii) *Missing Edges* (ME): number of edges in G^* that are not present in $\tilde{G}$ irrespective of the orientation (lower the better), (iii) *Wrong Orientation* (WO): number of edges present in both $\tilde{G}$ and G^* but with opposite orientation (lower the better), (iv) *Wrong Connection* (WC): number of edges present in $\tilde{G}$ but not in G^* excluding wrong orientation edges (lower the better), (v) *Graph Error* (GE): ME + WO + WC (lower the better), (vi) Best Result: the $\tilde{G}$ associated with the minimum GE estimated by our proposed algorithm considering the randomly generated datasets, (vii) *Preprocessing Time* (PT): total time required for the preprocessing stage described in Section IV, and (viii) *Graph Learning Time* (LT). All experiments are conducted on a PC with Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz with 16 GB memory.

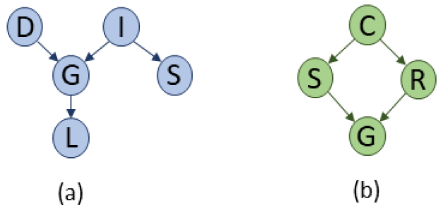

Fig. 1. Standard Bayesian networks: (a) Student (D: Difficulty, I: Intelligence, G: Grade, S: SAT, L: Letter) and, (b) Sprinkler (C: Cloudy, S: Sprinkler, R: Rain, G: Wet grass).

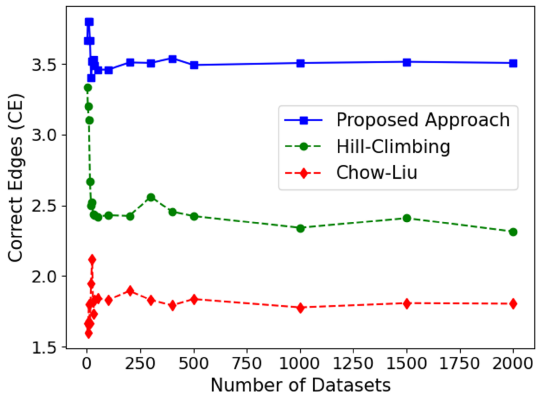

Fig. 2. Variation of *Correct Edges* (CE) metric as a function of the number of datasets generated from the *Sprinkler* network.

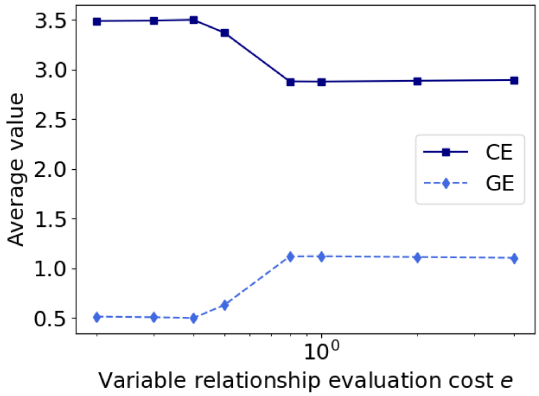

Fig. 3. Variation of average value of the *Correct Edges* (CE) and *Graph* Error (GE) metrics as a function of the variable relationship evaluation cost $e \in \{0.20, 0.30, 0.40, 0.50, 0.80, 1.00, 2.00, 4.00\}$ for the *Sprinkler* network.

Fig. 4. Learned Bayesian networks by (a) Hill–Climbing, and (b) Chow–Liu algorithms for *Sprinkler* network, when our approach correctly identifies the gold standard network G^*. Graph errors are shown in red while missing edges are shown in dashed lines.

Initially, we generate different number of datasets in the interval $[3, 2000]$, each of which includes $1,000$ samples, and evaluate the performance of the proposed approach and two widely used methods, Hill–Climbing [9] and the Chow–Liu [13] algorithm. Fig. 2 shows the variation of the CE metric as we increase the number of datasets generated from the *Sprinkler* network. We observe that the proposed approach succeeds in identifying on average higher number of edges compared to the two other algorithms. Furthermore, the value of the CE metric stabilizes around 500 datasets. We observe similar trends for the rest of the metrics and the *Student* network, but the relevant results are not included herein due to space limitations. For the rest of the experiments, we generate 500 datasets and report average results hereafter.

Fig. 3 illustrates the variation of CE and GE as a function of the variable relationship evaluation cost e for the *Sprinkler* network. As expected, different variable relationship evaluation costs lead to different CE and GE values, while evaluating less variable relationships leads to lower CE and thus higher GE values. We underscore that the LT metric is also reduced when the variable relationship evaluation cost increases, since less relationships are evaluated in this case. Thus, an optimum decision is reached in a shorter learning time. From here onwards, we report average results for $e_k = e = 0.3, 0.5$, for the *Sprinkler* and *Student* networks, since our experiments indicate that these values lead to $\tilde{G}$ close to G^*.

TABLE I
COMPARATIVE ANALYSIS FOR *Student* AND *Sprinkler* NETWORKS.

	Metrics	Proposed Approach	Hill–Climbing	Chow–Liu
Student	CE	**3.0240 ± 1.0007**	1.3300 ± 1.3932	2.2400 ± 0.7665
	ME	**0.0000 ± 0.0000**	**0.0000 ± 0.0000**	**0.0000 ± 0.0000**
	WO	**0.9760 ± 1.0007**	2.6700 ± 1.3932	1.7600 ± 0.7665
	WC	**0.0000 ± 0.0000**	1.1420 ± 0.9008	**0.0000 ± 0.0000**
	GE	**0.9760 ± 1.0007**	3.8120± 2.2678	1.7600± 0.7665
	LT (s)	**0.0015 ± 0.0034**	0.1331 ± 0.0212	0.0406± 0.0123
Sprinkler	CE	**3.4920 ± 0.5004**	2.4240 ± 1.1845	1.8360 ± 0.8139
	ME	**0.0000 ± 0.0000**	**0.0000 ± 0.0000**	1.0000 ± 0.0000
	WO	**0.5080 ± 0.5004**	1.5760 ± 1.1845	1.1640 ± 0.8139
	WC	**0.0000 ± 0.0000**	0.4100± 0.6409	**0.0000 ± 0.0000**
	GE	**0.5080 ± 0.5004**	1.9860 ± 1.7040	2.1640 ± 0.8139
	LT (s)	**0.0028 ± 0.0043**	0.0768 ± 0.0126	0.0341 ± 0.0057

Table I reports average statistics (± standard deviation) for all evaluation metrics (except PT and Best Result that are separately discussed in the text) on the *Sprinkler* and *Student* networks for the proposed approach, Hill–Climbing and the Chow–Liu algorithm. We compare with such methods due to their wide use in many applications [20] and their low time complexity. We observe that the LT of our proposed approach is nearly 96% less than the LT of Hill–Climbing and the Chow–Liu algorithms for both the *Student* and *Sprinkler* networks. This is due to the fact that our proposed approach is able to identify the most prominent relationships between variables early on. In this way, it avoids spending resources in non–informative relationships and results in sparser graphs but with more appropriate information content. In contrary, Hill–Climbing starts with a random network, manipulating one edge at a time, until the local maximum of a given score is found. However, in our approach we do not necessary evaluate all variable relationships, thus, saving considerable amount of time. In addition, if our hypothesis space includes the gold standard network G^*, our proposed approach tries to find the global optimum solution. This is in sharp contrast to Hill–Climbing, which ends up in a local maximum. Similar to our approach, the Chow–Liu algorithm [13] uses the mutual information to guide relationship selection. In contrast, our proposed approach tries to balance between information content and the sparsity of the graph. Furthermore, the estimated graph $\tilde{G}$ can be of arbitrary DAG structure, contrary to the tree–structured nature imposed by the Chow–Liu algorithm. We underscore that our approach is able to prioritize edges with higher information content. For instance, the priority of a wrong connection edge is less than a correct one given the way we define the information content of an edge with respect to a particular dataset (see Section II-A).

As seen in Table I, our proposed approach is able to identify 76% and 87% of correct edges in the *Student* and *Sprinkler* Bayesian networks, respectively. In addition, only 24% and 13% edges have wrong orientation. Thus, the proposed approach, irrespective of its simplicity, achieves competitive performance against well–known structure learning algorithms considered in this paper. Among the randomly generated datasets for each network, there exist instances of best results, which identify all the correct edges present in G^* resulting in GE of 0. This suggests that our proposed approach is

capable of identifying the gold standard network G^* from the given set of candidate graphs. In contrast, both Hill–Climbing and Chow–Liu algorithms result in *non–zero* graph errors (see also Fig. 4). Moreover, we observe that our approach outperforms Hill–Climbing and the Chow–Liu algorithms. Specifically, for both networks, the number of correct edges is higher than that identified by the Hill–Climbing and Chow–Liu algorithms. Even though our hypothesis space consists of both denser and sparser graphs, our proposed approach is able to balance between maximizing information content and sparsity to favor the graphs that are most related to G^*. Note that the average number of missing edges is low for both our approach, Hill–Climbing, and Chow–Liu. One caveat of our approach is its PT, which amounts to 2.1590 ± 0.1836 (s) and 2.1574 ± 0.1599 (s) for the *Student* and *Sprinkler* networks, respectively. To address this challenge, we plan to explore existing methods [25], [26] to improve the speed of solving the dynamic programming approach discussed in Section III. This is expected to considerably shorten the PT.

VI. Conclusions and Future Work

In this paper, we proposed a method to speed–up the graph learning time without compromising accuracy given a finite set of candidate graphs. The proposed method sequentially evaluates variable relationships until it reaches a specific decision regarding the underlying Bayesian network. Overall, the proposed approach is shown to outperform well–known existing structure learning methods with respect to various accuracy metrics and graph learning time. Nonetheless, the preprocessing stage overhead is proportional to the size of the network. Thus, we plan to explore methods to decrease the preprocessing time and extend the proposed approach for larger Bayesian networks. Since the number of all possible hypotheses increases with respect to the number of random variables, we also plan to address the problem of hypothesis space selection by exploring hierarchical hypothesis classification approaches. Finally, we plan to use an asymmetric score to account for edge orientation, while explicitly deciding the orientation of relationships as they are added to the graph.

References

[1] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Elsevier, 2014.

[2] 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.

[3] D. Nikovski, "Constructing Bayesian Networks for Medical Diagnosis from Incomplete and Partially Correct Statistics," *IEEE Transactions on Knowledge and Data Engineering*, vol. 12, no. 4, pp. 509–516, 2000.

[4] K. B. Korb and A. E. Nicholson, *Bayesian Artificial Intelligence*. CRC press, 2010.

[5] S. L. Lauritzen and D. J. Spiegelhalter, "Local Computations with Probabilities on Graphical Structures and Their Application to Expert Systems," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 50, no. 2, pp. 157–194, 1988.

[6] S. Andreassen, R. Hovorka, J. Benn, K. G. Olesen, and E. R. Carson, "A Model-based Approach to Insulin Adjustment," in *AIME 91*. Springer, 1991, pp. 239–248.

[7] E. Nazerfard and D. J. Cook, "Using Bayesian Networks for Daily Activity Prediction." in *AAAI workshop: plan, activity, and intent recognition*. Citeseer, 2013.

[8] V. Mitra, H. Nam, C. Y. Espy-Wilson, E. Saltzman, and L. Goldstein, "Gesture-based Dynamic Bayesian Network for Noise Robust Speech Recognition," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2011, pp. 5172–5175.

[9] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. MIT press, 2009.

[10] M. Scanagatta, A. Salmerón, and F. Stella, "A Survey on Bayesian Network Structure Learning from Data," *Progress in Artificial Intelligence*, vol. 8, no. 4, pp. 425–439, 2019.

[11] M. J. Flores, A. E. Nicholson, A. Brunskill, K. B. Korb, and S. Mascaro, "Incorporating Expert Knowledge when Learning Bayesian Network Structure: A Medical Case Study," *Artificial intelligence in medicine*, vol. 53, no. 3, pp. 181–204, 2011.

[12] H. Amirkhani, M. Rahmati, P. J. Lucas, and A. Hommersom, "Exploiting Experts' Knowledge for Structure Learning of Bayesian Networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 11, pp. 2154–2170, 2016.

[13] C. Chow and C. Liu, "Approximating Discrete Probability Distributions with Dependence Trees," *IEEE transactions on Information Theory*, vol. 14, no. 3, pp. 462–467, 1968.

[14] X.-W. Chen, G. Anantha, and X. Lin, "Improving Bayesian Network Structure Learning with Mutual Information-based Node Ordering in the K2 Algorithm," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 5, pp. 628–640, 2008.

[15] R. R. W., "Counting Labeled Acyclic Digraphs," in *New Directions in the Theory of Graphs*, F. Harary, Ed. Academic Press, NY, 1973, pp. 239–273.

[16] M. Chickering, D. Heckerman, and C. Meek, "Large-sample Learning of Bayesian Networks is NP-hard," *Journal of Machine Learning Research*, vol. 5, 2004.

[17] R. Daly, Q. Shen, and S. Aitken, "Learning Bayesian Networks: Approaches and Issues," *The knowledge engineering review*, vol. 26, no. 2, pp. 99–157, 2011.

[18] M. Teyssier and D. Koller, "Ordering-based Search: A Simple and Effective Algorithm for Learning Bayesian Networks," *arXiv preprint arXiv:1207.1429*, 2012.

[19] S. Behjati and H. Beigy, "Improved K2 Algorithm for Bayesian Network Structure Learning," *Engineering Applications of Artificial Intelligence*, vol. 91, p. 103617, 2020.

[20] J. Jiao, Y. Han, and T. Weissman, "Beyond Maximum Likelihood: Boosting the Chow-Liu Algorithm for Large Alphabets," in *2016 50th Asilomar Conference on Signals, Systems and Computers*. IEEE, 2016, pp. 321–325.

[21] I. A. Beinlich, H. J. Suermondt, R. M. Chavez, and G. F. Cooper, "The ALARM Monitoring System: A Case Study with Two Probabilistic Inference Techniques for Belief Networks," in *AIME 89*. Springer, 1989, pp. 247–256.

[22] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 2nd ed. Athena Scientific, 2000.

[23] A. Darwiche, *Modeling and reasoning with Bayesian networks*. Cambridge university press, 2009.

[24] S. Russell and P. Norvig, "Artificial Intelligence: A Modern Approach," 2002.

[25] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and Acting in Partially Observable Stochastic Domains," *Artificial intelligence*, vol. 101, no. 1-2, pp. 99–134, 1998.

[26] M. T. Spaan and N. Vlassis, "Perseus: Randomized Point-based Value Iteration for POMDPs," *Journal of artificial intelligence research*, vol. 24, pp. 195–220, 2005.