ICIC Express Letters
Volume **10**, Number **8**, August **2016**

 ISSN 1881-803X
pp. **1801–1806**

PERFORMANCE ANALYSIS WITH TOPOLOGICAL NODE ORDERING TO BAYESIAN NETWORK STRUCTURES LEARNING PROBLEM

Chen Li, Guoyang Li and Yanghui Wu*

School of Science
Northwest A&F University
No. 3, Taicheng Road, Yangling 712100, P. R. China
*Corresponding author: sxxawyh08@nwsuaf.edu.cn

Received December 2015; accepted March 2016

Abstract. *Search and score algorithm is one of the main algorithms for Bayesian network structure learning. Previous researches with three benchmark problems have shown the performance of this algorithm depends on the size and the complexity of each problem. In this paper, a deep investigation is carried out with eight new problems to thoroughly study the behavior of search and score algorithm. We explore the potential relationships between topological node ordering and the different performance to each of the problems. The results indicate that both Tarjan and Kahn node orderings give positive effect to chain model and K2 model respectively. The results in this paper give some implications for further research to construct a hyper-heuristic approach in BN structure learning problem.*

Keywords: Bayesian network, Search and score algorithm, Topology node ordering, K2 algorithm, Tarjan and Kahn methods

1. **Introduction.** Bayesian Networks (BN) are probabilistic graphical models which are used to represent knowledge about uncertain domains. A BN for a set of variables $X = \{X_1, X_2, \ldots, X_n\}$ consists of a network structure and a joint probability distribution (JPD) over the random variables X. In particular, given structure S, the JDP for X is given by

$$p(X) = \prod_{i=1}^{n} p(X_i|\pi(X_i)) \tag{1}$$

Here, $\pi(X_i)$ indicates the set of parents of node X_i.

Learning BN structure is an NP-hard optimization problem. It is known that the number of possible structures grows super-exponentially with the number of nodes, and so evaluating all possible structures is infeasible in most practical domains, where the number of variables is typically large. The process of finding cheaper approaches for learning the structure of BNs from large datasets is now a very active research area. In recent years, many novel algorithms have been successfully applied to BN structure learning [1, 2]. Search and score algorithm is one of the main algorithms which use metaheuristic search combined with deterministic construction and scoring of a network. A range of well-known techniques have been applied, including Hill Climbing [3, 4], Genetic Algorithms (GA) [1], Simulated Annealing [5], and Ant Colony Optimization (ACO) [2]. Scoring functions in the search and score algorithm are used to indicate the likelihood of a particular candidate network given a set of observed data.

Typically, search and score algorithms are compared on a range of benchmark problems with known structure and generated datasets. Algorithms are compared on search efficiency, optimized score and the structural similarity between the recovered and the original networks. Results are observed to vary widely with different benchmarks. In all

these existing algorithms, the empirical experiments show some different performances both on accuracy and efficiency. In our previous work, we have investigated combinations of two metaheuristic search techniques GA and ACO with two scoring approaches, K2 and chain [6] on a range of benchmarks [7, 8]. The results suggested that the difficulty of structure learning makes the choice of suitable algorithms and scoring functions for certain problems play an important role on the design of approaches. We also proved that the chain approach is able to yield high quality solutions with significantly less computational effort than the K2 approach in problems where the true structure of the data is amenable to alignment of node juxtapositions in a single ordering. In this paper we mainly focus on studying the performance of two score functions in search and score approaches. We aim to understand in which degree the choice of different fitness function affects the effeteness of the BN structure learning algorithms. The main contribution is that we introduce two kinds of topological node orderings of each known structure, and study the potential relationships between topological node orderings and the Chain and K2 score functions. We conduct empirical experiments to explain the observed performance with eight complex structures. We will explore existing empirical performance comparisons with further potential for understanding the relative difficulty of benchmark problems and the robustness of particular algorithms.

The remainder of the paper is organized as follows. In Section 2, we provide background on the K2-CH and chain scoring metrics. In Section 3, we introduce the experiments designed in this paper. The results and discussion are described in Section 4. Finally, conclusions and a discussion of the wider implications of this study are presented in Section 5.

2. **K2 Score Function and Chain Score Function.** K2 greedy search algorithm, proposed by Cooper and Herskovits, is one of the most widely-studied algorithms for learning BN structures. K2 algorithm searches, given a database D for the BN structure G with maximal $P(G, D)$. Here $P(G, D)$ is defined as

$$P(G, D) = P(G) \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{(r_i - 1)!}{(N_{ij} + r_i - 1)!} \prod_{k=1}^{r_i} N_{ijk}! \tag{2}$$

$P(G)$ denotes the prior probability of the network structure G, n is the number of discrete variables in the dataset, and q_i denotes the number of possible different instances in the parent $\pi(X_i)$ of variable X_i can take. r_i is the number of possible values assignments X_i has. N_{ijk} denotes the number of cases in the dataset D in which X_i takes value k of its r_i possible values when its parents $\pi(X_i)$ have their j-th configuration of values. N_{ij} is the sum of all N'_{ijk} for all values X_i can take. $N'_{ijk} > 0$ $(k = 1, 2, \ldots, r_i)$ is the hyparameters of the Dirichlet distribution [9].

K2 algorithm assumes that an ordering on the variables is available and that all structures respecting that ordering have equal likelihood. K2 algorithm starts by assuming that all nodes are without parents (i.e., independent), after which in every step it adds nodes incrementally to the parent set $\pi(X_i)$ of each node X_i $(i = 1, 2, \ldots, n)$. Assuming a uniform prior for $P(G)$, the following Function (3) is used to measure the conditional probability of each node and its parents. With the new nodes added to the set of parent $\pi(X_i)$, it should maximize this function.

$$g\left(X_i, \pi(X_i)\right) = \prod_{j=1}^{q_i} \frac{(r_i - 1)!}{(N_{ij} + r_i - 1)!} \prod_{k=1}^{r_i} N_{ijk}! \tag{3}$$

The process stops adding the nodes to the set of parents when the addition of a single parent cannot increase the probability $g(X_i, \pi(X_i))$.

K2 algorithm uses a Bayesian scoring metrics (2), which measures the joint probability of a BN structure G and a dataset D. In most of the search and score algorithms, people use this scoring function as a quality measure to the structure learned. The metric adopted the name of the algorithm, and in this paper, we name it as K2 score. From (2), we know, the main factor in K2 metric is Function (3), so in experimental evaluating, the number of counting on Function (3) is regarded as a measurement to evaluate the complexity of some specific algorithms.

The chain model structure approach to BN structure learning is defined in [7] and can be thought of as a refinement of K2 based search-and-score on the space of orderings. Chain structure algorithm operates in two phases. In first phase, a hypothesis is made that an initial search phase of evaluating fixed chain structure imposed on orderings provides a sufficiently good scoring function to locate high scoring regions of the space of node orderings. A second phase then follows where K2 is applied directly to the best orderings found. Given a node ordering $X_1, X_2, \ldots, X_n$, the associated chain structure is defined by adding edges between successive nodes. As described in Figure 1, X_i is the sole parent of X_{i+1}. E_i is the edge from X_i to X_{i+1}. The pseudocode of chain-based algorithm is shown in [6].

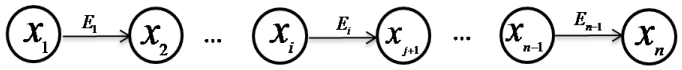

FIGURE 1. Chain structure on an ordering

Equation (2) is still used to quality evaluation of the chain structure model. From the definition of chain structure model, we know, in the chain model, the set parent $\pi(X_i)$ of node X_i in chain structure only has one node X_{i-1}. So in this case the operation of Equation (2) becomes cheaper. In this paper, we name this kind of function as chain score metric.

3. **Experimental Methodology.** The experiments designed in this paper have two stages. First of all, we generate both Tarjan and Kahn topological node orderings [10] with benchmark problems and investigate the inner link of each problem. Secondly, we conduct experiments with four approaches with chain score and K2 score respectively to analyze the potential relationships with the properties of node orderings. The four approaches in this paper are ChainACO, ChainGA, K2ACO and K2GA. More details about these approaches can be found in [6, 7]. We will take the above described experiments on eight benchmark structures. They are Asia, Car, Alarm, Credit, Tank, B, Boerlaga, and Insurance. All the data cases are sampled using probabilistic logic sampling in the Genie tool software [11]. The data size for all of these problems is $3,000$.

To measure the experimental results, in each approach, we record the following data.

- The fitness values (Avg. Score) of the learned structures: the closer the value is to zero, the closer the probability is to 1. This means the better is the network.
- The number of factor evaluations (F.E): F.E. is utilized to evaluate the efficiency of each algorithm. It is defined as being the count of times the term (3) is accessed when (2) is used. F.E indicates the complexity of each algorithm, less number of F.E., cheaper algorithm.

4. **Results and Discussions.**

4.1. **Topological node ordering with eight benchmark problems.** From analyzing in the above section, we notice that the different values of width and depth to each structure will affect the Kahn and Tarjan type orderings, and we also expect that the total number of true edges in different original structures will lead to different performance of

TABLE 1. Properties of group level on benchmark structures

Structure	N	E	W	D	W/E	D/E	True Arcs	
							Kahn	Tarjan
Asia	8	8	3	4	0.37	0.50	0.31	0.52
Credit	12	12	7	4	0.58	0.33	0.13	0.35
Tank	5	20	14	4	0.25	0.20	0.14	0.39
Car	18	18	10	5	0.55	0.27	0.14	0.27
B	18	39	5	6	0.12	0.16	0.20	0.52
Boerlaga	23	36	3	14	0.08	0.39	0.41	0.71
Insurance	27	52	5	10	0.09	0.18	0.18	0.44
Alarm	37	46	12	11	0.26	0.24	0.13	0.34

landscape to each ordering. To understand the topological node orderings in later section, we describe properties of group levels to all benchmark structures in Table 1. These include the number of width, depth, true edges, the ratio of width to true edges and depth to edges, and the percent values of true arcs produced by Kahn and Tarjan algorithms. We got these percent values through randomly generating 1000 node orderings on Kahn and Tarjan algorithms respectively, comparing the number of true edges that emerged in these orderings to the total number of edges in the original structures. These values show the difference between each algorithm on producing the true edges. In Table 1, N represents the nodes number of each structure, E is the number of original edges in the structure, W indicates the width, and D shows the depth of the specific structure.

4.2. **Results of search and score algorithms on benchmark structures.** Main results of chain and K2 based algorithms applied on benchmark structures are shown in Figure 2 and Figure 3. In these figures, the number from 1-8 in X axis indicates Asia, Credit, Tank, Car, B, Boerlaga, Insurance and Alarm respectively. Y axis in Figure 2 indicates the fitness score and this in Figure 3 indicates the number of F.E. with the four approaches.

In Figure 2, a very clear result from comparing fitness values is in all structures, chain based algorithms (ChainACO) got the best score, whether the structure is simple or complex. On the other hand, approaches with ACO achieved better score than GA based approaches. To investigate the structures learned, we carried out one way ANOVA test using the Bonferroni correction with the four algorithms to compare the averaged score in each structure. The results indicate, at the 0.05 level, there is not significant difference between the ChainACO approach and the K2 based approaches in Asia, Car, Boerlaga and Alarm structures. The F values in these structures are 9.127, 0.692, 8.078 and 26.525 respectively. There are significant difference between the ChainGA approach and the K2 based approaches in all structures; except in Car structure there is not significant difference. The F values in structures Credit, Tank, B and Insurance are 9.127, 0.692, 8.078 and 26.525 respectively. The F.E. results in Figure 3 describe the computational costs in all structures with chain and K2 based algorithms. The most notable consequence is that chain based approaches are cheaper than K2 based algorithms. The values of F.E. have a significant difference even for the simple structure Asia and Credit. With the increasing of nodes in structures, this difference becomes clearer. In all structures, ChainACO is the cheapest approach and K2ACO is the most expensive one in computing consumption.

These results indicate that there is a high degree of problem dependency both in the effectiveness and the efficiency of the approaches used. Both the choice of metaheuristic and the choice of scoring method can significantly affect performance. In all structures, the chain based algorithms are always more computationally efficient, but with a penalty on

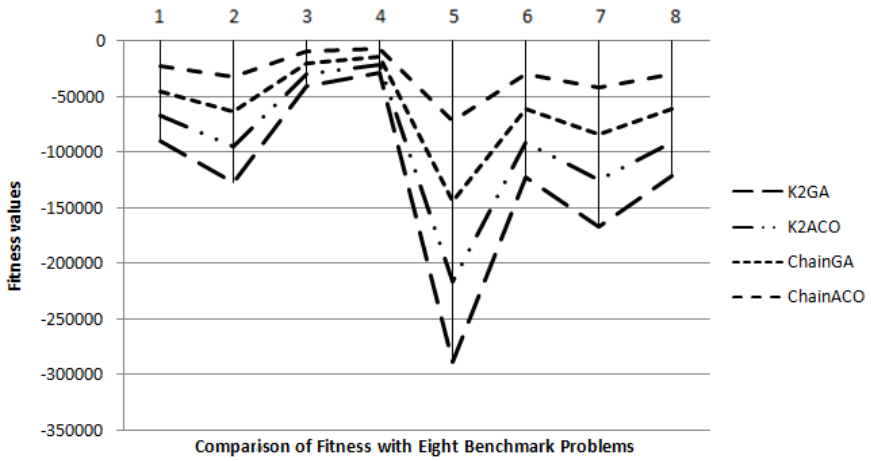

FIGURE 2. Comparison of fitness with eight benchmark problems

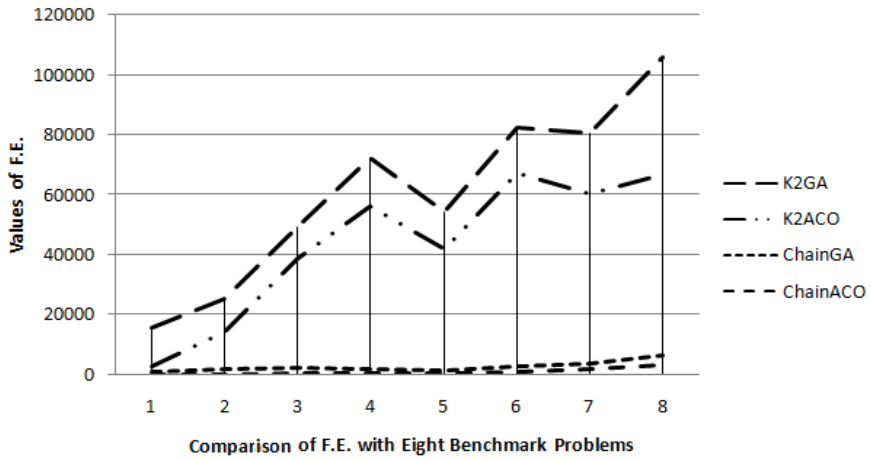

FIGURE 3. Comparison of F.E. with eight benchmark problems

the success in retrieving structure on known. The results also indicate, in some situation, the chain scoring approach is likely to be unsuccessful in producing high quality structures (for example, in Credit, Tank, B and Insurance structures, there are significant differences between the chain based approaches and the K2 based approaches to score values), so the relative benefit of reduced computational time is lost. However, experiments on the overall distribution and success of the four algorithms showed that the scoring approach, as opposed to the choice of metaheuristic' is the dominant influencing factor. This motivates analysis of the interaction between scoring metric and problem structure.

From pervious section, we know both Kahn and Tarjan algorithms generate node orderings by using their node groups and levels of the structures. The depth and width of structure affect the Kahn and Tarjan node orderings. More value of depth means more possibility on getting Tarjan node orderings, this also depending on the total edges in specific structure. The values of D/E in Table 1 show the relative results of depth to total edges of each structure. We sort these structures according to the values from high to low: Asia, Boerlaga, Credit, Car, Alarm, Tank, Insurance and B. Comparing these structure orderings to results in Figure 2, we can find chain based algorithm (ChainACO) got better results on higher value of D/E structures Asia, Boerlaga, Car and Alarm, this performance can test and verify in chain score algorithm, and the Tarjan node orderings can get better performance in some structures. Table 1 tells us that in all benchmark structures, the Tarjan orderings can get higher percent number on true edges comparing

to Kahn orderings. When random swapping operation executed on these specific node orderings, the true edge existing in Kahn orderings is destroyed easily; however, the Tarjan orderings can still keep higher percent number of the true edges, and the change to chain score correspondingly is not as dramatic as in the Kahn orderings.

5. **Conclusions.** In this paper we propose constructing topological node ordering to different benchmark problems, and we conduct experiments with chain and K2 based search and score approaches, by analyzing the performance of each approach to investigate the relationships with score function and node ordering.

Our results found that the depth and width of each structure have the effect to the performance of each algorithm. Chain score function can maintain the characteristic of Tarjan sorts when conducting random swapping, which makes the search process smoother and easier. K2 score based landscape, on the Kahn node orderings performs better sometimes; but we know, this process is expensive on most of the benchmark problems. The results gave us some useful inspiration when learning BN structure with search and score approaches. The results indicate that the structure of node orderings will affect the search and score algorithms in a large extent, and that the Tarjan type orderings are suggested to be good for both of the algorithms discussed in this paper. The mathematical theoretical analysis about these kinds of orderings is also needed to be studied in our future research.

Acknowledgement. This paper is partially supporting by programs for the Fundamental Research Funds for the Central University (Z109021561), the Scientific Research Foundation for doctorate of Shaanxi Province of China (Z111021504, Z111021306).

REFERENCES

[1] C. P. de Campos and Q. Li, Efficient structure learning of Bayesian networks using constraints, *Journal of Machine Learning Research*, vol.12, pp.663-689, 2011.

[2] R. Daly and Q. Shen, Learning Bayesian network equivalence classes with ant colony optimization, *Journal of Artificial Intelligence Research*, vol.35, pp.391-447, 2009.

[3] I. Tsamardinos, L. E. Brown and C. F. Aliferis, The max-min hill-climbing Bayesian network structure learning algorithm, *Machine Learning*, vol.65, pp.31-78, 2006.

[4] L. M. de Campos, J. M. Ferndez-luna and J. M. Puerta, An iterated local search algorithm for learning Bayesian networks with restarts based on conditional independence tests, *International Journal of Intelligent Systems*, vol.18, pp.221-235, 2003.

[5] T. Wang, J. W. Touchman and G. Xue, Applying two-level simulated annealing on Bayesian structure learning to infer genetic networks, *Proc. of the IEEE Computational Systems Bioinformatics Conference*, pp.647-648, 2004.

[6] Y. Wu, J. A. W. McCall and D. W. Corne, Two novel ant colony optimization approaches for Bayesian network structure learning, *IEEE Congress on Evolutionary Computation*, pp.1-7, 2010.

[7] R. Kabli, F. Herrmann and J. McCall, A chain-model genetic algorithm for Bayesian network structure learning, *Proc. of the 9th Annual Conference on Genetic and Evolutionary Computation*, pp.1264-1271, New York, NY, USA, 2007.

[8] Y. Wu, J. McCall and D. Corne, Comparative analysis of search and score metaheuristics for Bayesian network structure learning using node juxtaposition distributions, *Lecture Notes in Computer Science*, pp.424-433, 2010.

[9] P. Larranga, C. M. H. Kuijpers, R. H. Murga and Y. Yurramendi, Learning Bayesian network structures by searching for the best ordering with genetic algorithms, *IEEE Trans. Systems, Man and Cybernetics*, vol.26, pp.487-493, 1996.

[10] Y. Wu and L. Zheng, Analysis with topological node ordering on Bayesian structure model, *ICIC Express Letters*, vol.9, no.9, pp.2499-2504, 2015.

[11] *Genie and Smile*, http://genie.sis.pitt.edu/about.html.