Learning Bayesian Networks Structures with an Effective Knowledge-driven GA

Weijian Zhang, Wei Fang*, Jun Sun, Qidong Chen
Jiangsu Provincial Engineering Laboratory of Pattern Recognition and Computational Intelligence
Department of Computer Science and Technology
Jiangnan University
Wuxi, China
Email: fangwei@jiangnan.edu.cn

***Abstract*—Bayesian networks (BNs) are probabilistic graphical models, which are regarded as one of the most effective theoretical models in the field of representing and reasoning under uncertainty. Learning BNs structure is an NP-hard problem since the search space of structure grows super-exponentially as the increasing of the number of variables. Evolutionary algorithms (EAs) are widely used to learn BNs structure while single-solution searching methods may trap into local optima. This work aims to propose an efficient knowledge-driven Genetic algorithm (EKGA-BN) to solve the BN structure learning problem. The proposed EKGA-BN uses a novel selection operator to keep population diversity in order to learn a BN structure with higher accuracy. The idea of Hill climbing algorithm (HC) is combined in the selection operator so as to accelerate the convergence rate. A novel knowledge-driven mutation procedure is proposed to enhance the local search ability of EKGA-BN. Experimental results on four well-known benchmark networks show that the proposed method outperforms state-of-the-art algorithms in both convergence rate and the accuracy of BNs structure.**

***Index Terms*—Bayesian networks, Structure learning, Genetic algorithm, Selection operator, Hill climbing algorithm**

I. Introduction

Bayesian networks (BNs), as a method of reasoning under uncertainty, is commonly considered to be one of the best approaches to represent causal knowledge and is very popular in the field of probability [1]. Directed acyclic graph (DAG), in which nodes represent random variables and the existence of arcs denote the dependence relationships between variables, is usually used to represent BNs. These relationships are quantified by a set of conditional probability distributions (CPD), which is determined by its parent nodes for each variable. Since the advantages of BNs in inference and learning ability, it becomes increasingly popular in various research areas such as bioinformatics research [2], medical problem [3], image processing [4], etc. The structure of BNs can be provided by experts manually but the accuracy cannot be guaranteed and it is also time-consuming. Therefore learning a BN structure from data is an important task and has been studied extensively during last two decades.

Learning a completely correct BN structure from data is an NP-hard problem when the number of variables increases rapidly [5]. Many methods have been proposed to solve BNs structure learning problem. Learning a BN structure equals to learning the topology of the network. There are three commonly used methods to approximately learn BNs structure from data, which are constraint-based approaches, scored-based approaches, and hybrid approaches.

Constraint-based approaches first identify conditional independence relations between variables through statistical methods such as Pearson's χ^2 test [6]. Then a BN structure that best fit those relations is constructed. Some widely known algorithms are PC algorithm [7], Grow-Shrink algorithm (GS) [8], etc.

The score-based algorithms evaluate the quality of candidate network structure with a scoring metric. These approaches regard BNs structure learning problem as a combinatorial optimization problem. Traditional single-solution search algorithms such as Hill climbing algorithm (HC) [9] and simulated annealing may trap into local optima [1]. Some population-based search algorithms are introduced to indentify optimal structure, such as genetic algorithm (GA) [10], particle swarm optimization (PSO) [11], and ant colony optimization (ACO) [12]. Some of the best known score-based algorithms include K2GA [13], Chain-model GA [14], etc.

The hybrid algorithms integrate score-based approaches and constraint-based approaches together to searching the BNs structure in a large search space [1]. One of the commonly used strategy is to use constraint-based approaches to construct a graph's skeleton and then use a score-based approach to search a optimal DAG with the highest score. Max-Min Hill Climbing (MMHC) [15] and Sparse-Candidate (SC) algorithm [16] are the representative hybrid algorithms. EAs-based algorithms are also usually used in the search procedure of hybrid algorithms such as Canonical GA [10], in which GA is carried out after reducing the search space through conditional independence test. And Contaldi [17] proposed AESL-GA which relies on individuals with higher fitness for deleting redundant parent nodes and adaptively adjusting the maximum fan-in for each nodes.

In this paper, an efficient knowledge-driven Genetic algorithm (EKGA-BN), which contains a novel diversity-guided HC selection and an additional knowledge-driven mutation procedure, is proposed to solve the BN structure learning problem. The main contributions are summarized as follows.

1) A novel selection operator based on the population diversity is proposed and the idea of HC algorithm is

978-1-7281-6929-3/20/$31.00 ©2020 IEEE

incorporated. Keeping the population diversity makes the identified BNs structure with higher accuracy. The convergence rate is significantly accelerated by introducing the idea of HC algorithm.

2) A knowledge-driven mutation procedure is designed to improve the local search ablity. A common BN structure is identified by individuals with better performance, which contains the information of correct BN structure. The common BN structure therefore is used to reduce the redundant search space which makes searching more effectively and enhance exploitation.

The remainder of this paper is organized as follows: Section II introduces BNs and commonly used hybrid approaches to learn BNs stricture with GA. In Section III, a detailed description of the proposed EKGA-BN is provided. And the benchmark problems, the compared algorithms, the experimental results, and analysis are presented in Section IV. Finally, conclusion is drawn in Section V.

II. Problem statement

Let $G = (X, E)$ be a DAG where $X = \{x_1, x_2, ..., x_n\}$ is a node set representing variables and $E = \{e_{ij}\}$ is an directed edges set representing independence relationship between these variables. The relationships are represented by E, which is combined with directed arcs $e_{i,j}$ from parent node X_i to child node X_j. And $Pa(X_i)$ is defined as the parent nodes set of X_i. The dependence relationship between X_i and X_j can be quantified with conditional probability distribution $P(X_i \| Pa(X_i))$. If (G, P) satisfies the Markov condition, (G, P) can be called as a BN [11], where joint probability distribution P is combined by a product of local conditional probability distributions according to (1)

$$P(x_1, x_2, ..., x_n) = \prod_{i=1}^{n} P(x_i | Pa(x_i)) \tag{1}$$

Score-based approaches defined a score metric to evaluate the matching rate between the network and the observed data, then get a BN strcture which has a highest score during the procedure.

The score metric can be categorized into two classes called Bayesian and Information-theoretic scoring functions. The Akaike's Information Criterion (AIC) [18], Bayesian Information Criterion (BIC) [19], and Minimum Description Length (MDL) [20] are several well-known examples of Information-theoretic scoring metrics. For Bayesian scoring metric, given a training dataset D, its general idea is to compute the posterior probability distribution of a graph G and penalize this score by the complexity of the BN according to (2)

$$p(G|D) = \frac{p(D|G)p(G)}{p(D)} \propto p(D|G)p(G) \tag{2}$$

where $p(G)$ is the prior probability on different graph structures, and $p(D|G)$ is the parameter prior that puts a probability on different parameters Θ given a graph G. According to Bayes theorem, $p(D|G) = \int_\theta p(D|G, \theta)p(\theta, G)d\theta$, which is the marginal likelihood that averages the probability of the data D over all possible parameter assignments to G [11].

A. Learning BNs structure based on GA

Using scoring metric to learn BN structure can be regarded as a combinatorial optimization problem. GAs are effective methods to learn BN structure. The normally used procedure can be summarized as follows:

1) Encoding for BN structure learning problem like commonly applied adjacency matrix or a set of edges existing in DAG. A BN structure with n variables can be represented by a $n \times n$ adjacency matrix. An individual therefore is able to be represented by string (3)

$$x_i = a_{11}a_{12}...a_{1n}a_{21}a_{22}...a_{ij}...a_{nn} \tag{3}$$

where a_{ij} denotes whether node i is the parent of node j and it can be calculated using (4)

$$a_{ij} = \begin{cases} 1 & if\ i\ is\ a\ parent\ of\ j \\ 0 & otherwise \end{cases} \tag{4}$$

2) Constructing the initial population randomly and using predefined N as population size to limit the number of initialized individuals. Randomly initialization may generate invalid individuals with cycles. Then it is necessary to remove cycles with little changes in DAGs. Each individual is encoded as (3) and the fitness can be calculated according to the predefined scoring metric.
3) Using selection operator to keep individuals with higher fitness alive. For instance, roulette wheel selection and tournament selection are two widely used selection operator. The selection operator is to insert an individual with higher score into the new population ,which will be transferred to next generation
4) The crossover operator and mutation operator is carried out for each individual to generate new individuals, which may also introduce cycles into DAGs. Removing cycles procedure therefore is also needed after crossover and mutation operators to make individuals valid.
5) For each individual in next generation, repeat step 3 and step 4 until the termination condition is satisfied and output the individual with highest score as final solution.

III. The proposed EKGA-BN

In this section, inspired by the idea of the HC algorithm [9], the proposed selection operator keeps population diversity so as to increase the accuracy of final identified BN structure, while combining HC algorithm to accelerate the convergence rate. And a novel knowledge-driven mutation procedure, which is carried out after mutation operator, is designed to enhance the local search ability and the effectiveness of search. The proposed algorithm which is detailed in this section, can be regarded as a extended version of the SiRG method [21] denote as EKGA-BN.

According to (3), using logic operators to reproduce each individual is available. The truth table of logic operators is show in Table I and in this paper we use the signs $\bigotimes, \bigoplus, +$ to denote the **AND**, **XOR**, and **OR** logic operator, respectively.

TABLE I
LOGIC OPERATOR

X	Y	XOR(X,Y)	AND(X,Y)	OR(X,Y)
0	0	0	0	0
0	1	1	0	1
1	0	1	0	1
1	1	0	1	1

A. Overview of EKGA-BN

The procedure of EKGA-BN is shown in Fig. 1. Main steps are summarized as follows.

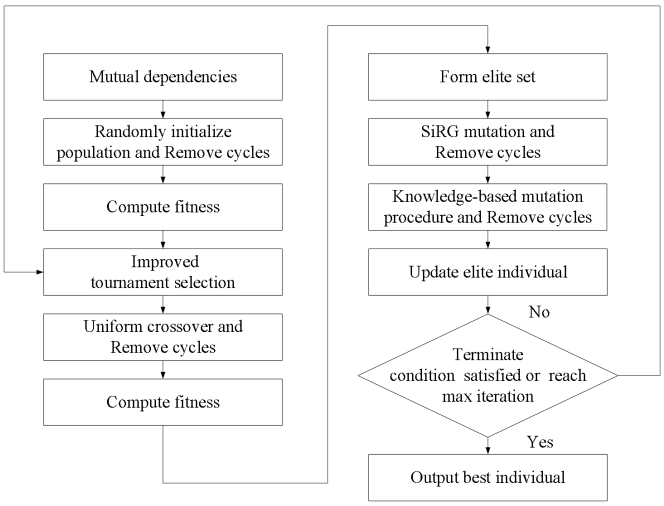


Fig. 1. Procedure of EKGA-BN.

- Mutual Dependencies. It uses statistical tests to build an undirected graph structure, which is referred to a super-structure (SS) and is help to reduce search space [10]. If node i and node j is conditional independent, undirected edge e_{ij} is added to $SS = \{e_{ij}\}$. Each edge in SS has 3 states which are $A \leftarrow B$, $A \rightarrow B$, and $A \nleftrightarrow B$.
- Randomly initialization. For each undirected edges in SS, one of the state is randomly chosen to initialize population.
- Remove cycles. During initialization, crossover, and mutation procedure, cycles may be introduced. And the nodes with the number of parent nodes larger than the predefined $N_m p$ might be produced, where $N_m p$ denote the maximum number of parent nodes for each node. Consequently a procedure to remove cycles in the invalid individuals is necessary. The removing cycles procedure randomly drops redundant parent nodes at first and then solves the minimum cycles set problem with an improved GR algorithm [10]. The removing cycles algorithm prefers to remove edges between nodes with more children and fewer parents for the sake of further minimizing the number of changes.
- Compute fitness. BDeu score (with equivalent sample size of 1) is used to evaluate the fitness of each individual after carrying out initializing, crossover, and mutation procedure at each generation.
- Selection. The proposed tournament selection operator, which can maintain the population diversity and introduce the idea of HC algorithm to speed up convergence rate, is applied to select individuals to next generation with better performance.
- Crossover. Uniform crossover is adopted for this algorithm. N new individuals are constructed with the strategy of uniform crossover, each bit in the offspring is randomly chosen from its two parents.
- Form elite set. Using predefined elite eligibility threshold α to produce elite set. Appending individuals whose score higher than $\alpha * f_{max}$ into elite set, where f_{max} denotes the highest fitness among the population.
- Mutation. An adaptive mutation scheme from Site-specific Rate Genetic (SiRG) algorithm [21], which retain the balance of exploration and exploitation of the search, is adopted. The mutation rate of each bit is determined by their fitness and the distribution of edges across the elite set.
- Knowledge-driven mutation procedure. After the mutation operator, the proposed mutation procedure is carried out to identify a common structure determined by elite set. The knowledge of the correct BN structure is contained among the individuals in elite set. Differences between each individual and the common structure can be identified. Single-point mutation is used so as to search the difference area therefore enhance the local search ability.

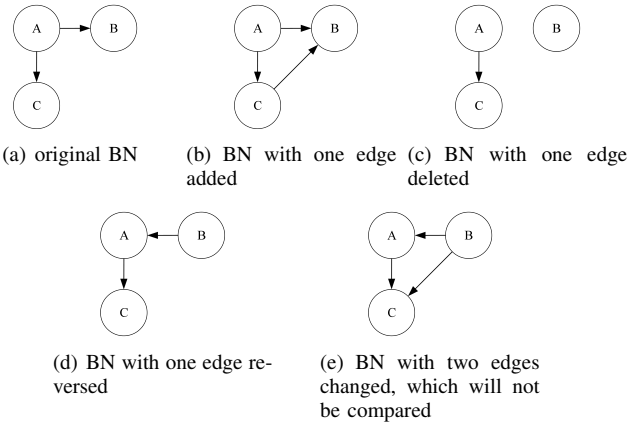
Repeating these steps until the termination condition is satisfied, and then the final solution with highest BDeu score is selected as the identified BN structure.

B. The diversity-guided HC selection

The improved tournament selection operator is proposed to keep population diversity and speed up convergence rate by HC algorithm. In order to avoid making all individuals be the same, the maximum number of same individuals is predefined to prevent this situation and therefore maintaining the population diversity. We also combine the idea of HC algorithm with selection operator to accelerate convergence rate. In HC algorithm, only the edge that increases the score mostly will be changed each time. Therefore in the selection operator, each individual is replaced with those having the best score among all individuals who have only one different edge comparing to itself.

Generally speaking, the next generation obtained by combining HC algorithm with the selection operator in BNs structure learning problem maintains the optimal solution in each niche, and keeps the population diversity. In the evolution process, it has higher probability to identify better structure and speeds up the convergence rate. The proposed selection operator can be summarized as follows:

1) Individuals are randomly selected from the population and grouped into groups $X_1, X_2, ..., X_t$, and select the individual X_{win} with the highest score.
2) If the number of existing individuals in the next generation P_{next}, which is as same as X_{win}, exceeds the predefined population size by $1/5$, X_{win} cannot be transferred to the next generation and repeat Step 1. Otherwise adding X_{win} to P_{next} and repeat the Step 1-2 until the population size reaches N.
3) Each individual X_i in the population is compared with the remaining individuals X_j. If the edge number of structure differences between X_i and X_j is less than or equal to one and the Bayesian score of X_i is higher than X_j, adding X_i to the set P_{new}. As shown in Fig. 2, assuming Fig. 2(a) is the original BN, and the Bayesian scores of BNs in Fig. 2(b), Fig. 2(c), and Fig. 2(d) are all higher than Fig. 2(a). Selecting a BN with highest score to take place of Fig. 2(a). Whether the score of Fig. 2(e) is higher than Fig. 2(a), the BN in Fig. 2(a) will not be replaced for there are two different edges comparing to the original BN. Individuals are selected through the procedure given in Algorithm 1.



(a) original BN (b) BN with one edge added (c) BN with one edge deleted

(d) BN with one edge reversed (e) BN with two edges changed, which will not be compared

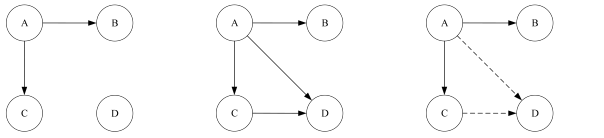
Fig. 2. The idea of HC in selection operator

Algorithm 1 Diversity-guided HC Tournament Selection

Require: Population, P_t;
Population number, N;
Tournament size, T;
Ensure: Population after selection, P_{new};
Initialize $i = 1$ and P_{new} to be an empty list;
repeat
  Get T individuals from P_t and select the best one as P_i^{new} which has highest bayesian score;
  Get number of individuals N_{same} which are as same as P_i^{new} in P_{new};
  if $N_{same} < N/5$ **then**
    append P_i^{new} to P_{new};
    $i++$;
  else
    continue;
  end if
until $(i \geq N)$
for $i = 1$ to N **do**
  initial L_{better} as an empty list;
  for $j = 1$ to N **do**
    Calculate the number of different edges between X_i and X_j accroding to $Dis = Sum(X_i \bigoplus X_j) - Sum((X_i \bigoplus X_j) \bigotimes (X_i \bigoplus X_j)^T)/2$;
    if $Dis < 2$ **then**
      Append P_j to L_{better};
    end if
  end for
  Get an new individual whose fitness score is highest in the list L_{better} and replace the original P_i in P_{new};
end for

C. Knowledge-driven mutation procedure

The proposed knowledge-driven mutation procedure is used to strengthen the local search ability which is executed after the mutation operator taken from SiRG algorithm [21] based on the information of correct BN structure contained on the elite set. We notice that individuals will quickly converge to the best individual when the individuals in elite set are stagnant. To tackle this problem, Searching the area between the individual and the common BN structure, which is constructed by elite set, to ensure the diversity of each niche. The proposed procedure makes search more effective and enhances the local search ability. First of all, a BN common structure is defined by voting at each gene bits with each individual in elite set. Second, single-point mutation is performed at every different gene bits, which is identified by the difference between each individual and the constructed common structure. This limited search space contains the knowledge of individuals with better performance and lead the mutation direction, therefore makes local search more efficient. As shown in Fig. 3(a), it represents the common structure of the elite set. Fig. 3(b) is the original BN, there are 2 edges identical in both Fig. 3(a) and Fig. 3(b). Accordingly the dotted line in Fig.3(c) shows the edges that might mutate. Algorithm 2 illustrates the process of the knowledge-driven mutation procedure.



(a) Common BN structure (b) original individual (c) Edges which might be mutated

Fig. 3. Knowledge-driven mutation procedure, mutate gene bits

Algorithm 2 Knowledge-driven mutation Procedure

Require: Population before knowledge-driven mutation procedure, P_t;
Population number, N;
Elite set, E;
Elite set number, N_E;

Ensure: Population after knowledge-driven mutation procedure, P_{new};

Initialize $i = 1$ and P_{skel} as same as first individual in E;
for $i = 1$ to N_E **do**
  $P_{skel} = P_{skel} \bigotimes E_i$;
end for
for $i = 1$ to N **do**
  P_i is the ith individual in P_t;
  Focused edge set $M_{diff} = P_i \bigoplus P_{skel}$;
  Mutation rate $m = \frac{1}{sum(M_{diff}+\epsilon)}$;
  for e in M_{diff} **do**
    if $(P_i(e_1, e_2) = 1$ or $P_i(e_2, e_1) = 1)$ and rand<m **then**
      Randomly change edge type of $P_i(e)$, for instance $e_1 \leftarrow e_2$ change to $e_1 \rightarrow e_2$ or $e_1; \nleftrightarrow e_2$;
    end if
  end for
  Append P_i to P_{new};
  Get an new individual whose fitness score is highest in the list L_{better} and replace the original P_i in P_{new};
end for
P_{new}=Remove-arc(P_{new});

IV. Experimental results and analysis

A. Datasets

Four BNs with different sizes are chosen for evaluation from Bayesian Network Repository [22]. The four BNs are listed in Table II and include a small BN with 8 nodes and 8 edges (ASIA) [23], a medium BN with 37 nodes and 46 edges (ALARM) [24], a large BN with 70 nodes and 128 edges (HEPAR II) [25] containing 500 cases, and a very large network with 223 nodes and 338 edges (ANDE) [26] with 300 cases, respectively.

TABLE II
Database used in experiments

Databases	Original network	Number of cases	Node number	Arc number	Score
Asia-500	Asia	500	8	8	-1147.34
Alarm-500	Alarm	500	37	46	-6111.6
HeparII-500	HeparII	500	70	123	-17215.9
Ande-300	Ande	300	223	338	-30127.5

B. Experimental settings

We compare the proposed EKGA-BN with six BNs structures learning algorithms, which are Maximum weight spanning tree (MWST) [27], Tree Augmented Naive Bayes (TAN) [28], K2 algorithm [29], HC algorithm [9], Max-min hill climbing algorithm (MMHC) [15], and GA-based algorithm AESL-GA [17]. MMHC is implemented in the Causal Explorer system [30] and the others are implemented by BNT Structure Learning Package (BNT-SLP) [31], which is developed on the Bayes Net Toolbox for Matlab [32].

The parameter settings of EKGA-BN and AESL-GA(according to Contaldi) are given in Table III. GA-based algorithms use the same population size $n = 100$, the maximum number of generation $m = 100$, and the CI test threshold for CB phase $\varepsilon = 0.5$. The maximum parent node number is 4 as ut is the average number of parent nodes by the model we used. HC, MWST, TAN and MMHC are implemented in the BNT-SLP and Causal Explorer system and default parameters are used. The order of node for K2 is generated randomly. BDeu score is used in experiments as described in Section II. Each algorithm is executed 20 times independently for each dataset and the average values are recorded for final result.

For the propose of fair comparison, the performance of the algorithm is compared by the score and structure differences between the final output network and the original network structure. F1-score, sensitivity, and specificity [33] are taken as the measurements.

The proposed algorithm is implemented with BNT-SLP. The first experiment is to test the effectiveness of proposed operators. The second experiment is to compare the proposed EKGA-BN with GA based hybrid structure learning algorithm and a series of well-known and widely used structure learning algorithms. All the algorithms run on Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz with 31GB RAM.

C. Effects of the proposed selection operator and knowledge-driven mutation procedure

SiRG is taken as the baseline algorithm. First, adding the knowledge-driven mutation procedure in SiRG, and then replacing tournament selection with proposed Diversity-guided HC tournament selection, which forms the proposed EKGA-BN. The experimental results with mean values and standard deviations are presented in Table IV.

From the results in Table IV, average Bayesian scores are same as each other among three methods on the ASIA500, which denotes that both of them are able to get global optimal in small dataset. As for larger datasets, evaluation metrics improve obviously in the order of SiRG, SiRG with knowledge-driven mutation procedure, and EKGA-BN, especially in ALARM500 dataset. Therefore both knowledge-driven mutation procedure and Diversity-guided HC tournament selection are effective and EKGA-BN is easier to identify the BN structure with highest Bayesian score and F1 score.

The standard deviations of the means for SiRG, SiRG with knowledge-driven mutation procedure, and EKGA-BN in both 3 datasets is in a decreasing manner. The lower standard deviations of the means are, the more stable output is. The result shows EKGA-BN has the most stable final solution comparing to other baselines.

D. Comparison EKGA-BN and the other learning algorithms

The experimental results by seven algorithms are shown in Table V and the bold evaluate metrics are those with the highest values. According to the result in Table V, the proposed algorithm achieves the best scores on three datasets which are ASIA500, HEPARII500, and ANDE300. Although

TABLE III
PARAMETER SETTINGS FOR EKGA-BN AND AESL-GA

Experiment	Population size	Maximum number of generation	Tournament size	CI test threshold for CB phase	Elite eligibility threshold	Maximum parent node number
EKGA-BN	100	100	4	0.01	0.5	4
AESL-GA	100	100	N/A	0.01	0.9	12

TABLE IV
EFFECTS OF THE PROPOSED OPERATORS (BEST RESULTS IN BOLD)

Methods	Asia-500		Alarm-500		HeparII-500	
	F1 Score	Bayesian score	F1 Score	Bayesian score	F1 Score	Bayesian score
SiRG	0.7875 (0.057)	-1146.521 (0)	0.6460 (0.096)	-6192.384 (61.107)	0.3004 (0.020)	-16428.873 (6.972)
SiRG with knowledge-driven mutation procedure	0.8063 (0.062)	-1146.521 (0)	0.7048 (0.072)	-6166.432 (46.648)	0.3119 (0.025)	-16426.336 (5.476)
EKGA-BN	**0.825** (0.0645)	-1146.521 (0)	**0.7761** (0.061)	**-6142.239** (46.571)	**0.3242** (0.017)	**-16423.024** (4.388)

TABLE V
COMPARING EKGA-BN WITH OTHER ALGORITHMS

Method		Asia-500	Alarm-500	HeparII-500	ANDE-300
EKGA-BN	F1 Score	**0.825** (0.0645)	**0.7761** (0.0610)	**0.3230** (0.01733)	**0.5232** (0.018)
	Sensitivity	**0.825** (0.06454)	**0.7478** (0.0534)	**0.2293** (0.01259)	**0.4722** (0.019)
	Specificity	0.9642 (0)	**0.9945** (0.004)	0.9945 (0.001)	0.9960 (0.0002)
	Score	**-1146.521** (0)	-6142.028 (47.8710)	**-16423.024** (4.389)	**-30094.437** (71.464)
AESL-GA	F1 Score	0.775 (0.053)	0.6659 (0.031)	0.3040 (0.029)	0.3213 (0.022)
	Sensitivity	0.775 (0.053)	0.6674 (0.040)	0.2154 (0.020)	0.2970 (0.019)
	Specificity	0.9643 (0)	0.9854 (0.004)	0.9934 (0.001)	0.9941 (0.0003)
	Score	**-1146.521)** (0)	-6278.734 (53.379)	-16434.180 (4.914)	-31713.014 (153.050)
MWST	F1 Score	0.2267 (0.167)	0.3732 (0.040)	0.2552 (0.022)	0.2636 (0.013)
	Sensitivity	0.2125 (0.156)	0.3326 (0.036)	0.1991 (0.017)	0.2183 (0.011)
	Specificity	0.9286 (0)	0.9924 (0)	0.9873 (0)	0.9972 (0)
	Score	-1172.951 (0)	-6602.342 (0)	-16457.470 (0)	-31673.124 (0)
TAN	F1 Score	0.2762 (0.1305)	0.2803 (0.045)	0.1331 (0.025)	0.1918 (0.009)
	Sensitivity	0.3625 (0.1713)	0.3565 (0.0572)	0.1406 (0.027)	0.2216 (0.011)
	Specificity	0.7321 (0.035)	0.9361 (0.003)	0.9568 (0.001)	0.9883 (0.0001)
	Score	-1180.902 (10.658)	-7012.498 (173.336)	-17018.265 (339.869)	-32527.118 (263.307)
K2	F1 Score	0.3482 (0.150)	0.3501 (0.0517)	0.1719 (0.041)	0.2568 (0.017)
	Sensitivity	0.3625 (0.150)	0.4087 (0.054)	0.1333 (0.032)	0.2973 (0.017)
	Specificity	0.8643 (0.050)	0.9619 (0.007)	0.9850 (0.002)	0.9889 (0.001)
	Score	-1157.091 (6.187)	-6464.332 (102.595)	-16510.977 (31.371)	-30269.058 (112.3962)
HC	F1 Score	0 (0)	0.4842 (0)	running out of time	running out of time
	Sensitivity	0 (0)	0.5 (0)		
	Specificity	0.8929 (0)	0.9863 (0)		
	Score	-1151.431 (0)	**-6130.914** (0)		
MMHC	F1 Score	0.7143 (0)	0.6667 (0)	0.2530 (0)	0.4595 (0)
	Sensitivity	0.625 (0)	0.6522 (0)	0.1707 (0)	0.4024 (0)
	Specificity	**1** (0)	0.9894 (0)	**0.9958** (0)	**0.9968** (0)
	Score	-1164.500 (0)	-6248.543 (0)	-16492.764 (0)	-30232.257 (0)

the score of HC algorithm in ALARM500 is highest, the F1 score, sensitivity, and specificity of EKGA-BN in ALARM500 outperforms HC algorithm. As F1 score and sensitivity are concerned, EKGA-BN obtains the best performance for all of the eight measures. As for Aisa network, EKGA-BN is able to find a Bayesian score equal to other GA based algorithm. On ALARM500 and ANDE300 datasets, EKGA-BN is able to find a BN which performs much better than the second best result in the field of structure differences. In other words, EKGA-BN is the best-performing algorithms in maximizing F1 score and the identified BN structure is able to reflect the reality most, while it can also achieves the highest Bayesian score consistently comparing to other algorithms.

Due to the limit of data, the F1 score and Bayesian score of TAN and MWST is poor, which denotes that constraint based algorithms may perform bad when the training data is insufficient. K2 and HC are both efficient single-solution searching methods. K2 optimize Bayesian score effectively, but the final solution does not match the original BN well and therefore has low F1 score. HC is able to identify BN structure with lower Bayesian score comparing to K2 in ASIA500 and ALARM500, but runs out of time in larger datasets.

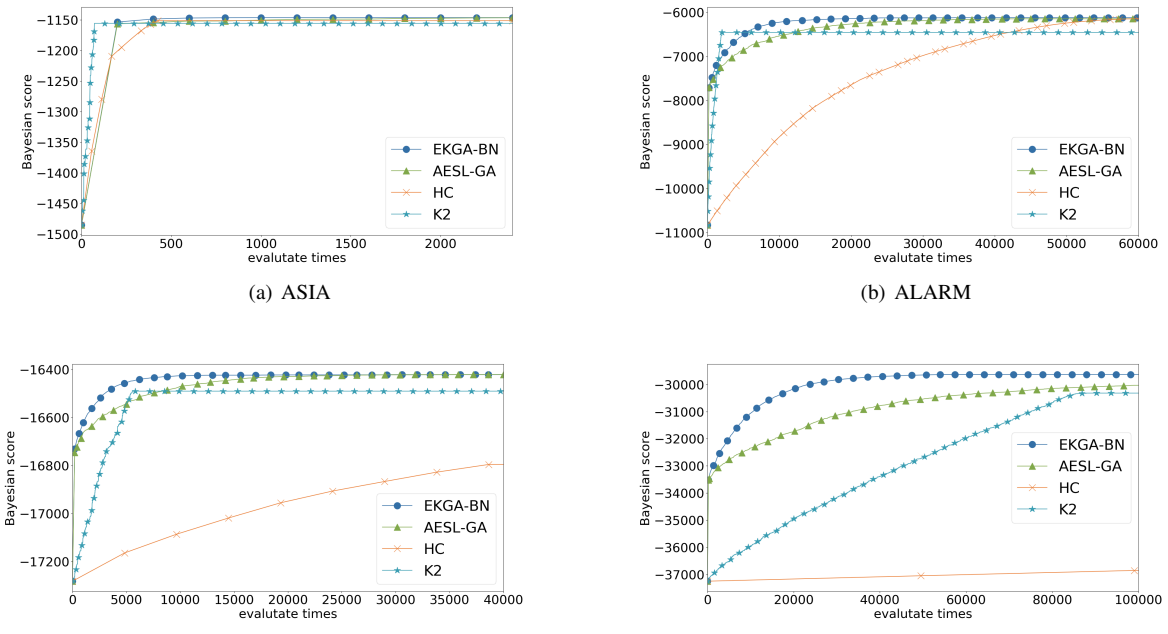
The F1 score of MMHC performs best among K2, HC, MWST, and TAN, which means the BN structure identified by MMHC is closer to the representation of reality and thus the accuracy is higher. But its F1 score performs worse than EA based algorithm in large datasets.

Fig. 4 illustrates the convergence rate of the algorithms with four datasets. The convergence graphs present the average scores obtained by EKGA-BN, AESL-GA, K2 and HC using databases ASIA-500, ALARM-500 and HEPARII-500. MWST and TAN are constraint-based algorithms, which therefore are not shown in Fig. 4. As the number of variables increases, the convergence speed of K2 algorithm and HC algorithm decrease obviously. Although K2 algorithm convergence faster than EKGA-BN in ASIA500, ALARM500, and HEPARII500, the score is much lower than EKGA-BN and AESL-GA in four datasets. And the HC algorithm is forced to stop after 36 hours spent on a single run in large datasets such as HeparII500 and ANDE300, while in small and medium problems it performs well and can search a BN with high score. The convergence rate for EKGA-BN is obviously faster than AESL-GA and HC algorithm in all four datasets and is more likely to get highest Bayesian score.

The experimental results show that the proposed algorithm is superior to other compared algorithms on the quality and performance measure. The result of MMHC algorithm is not shown in the result since the Causal Explorer system only give the executable file rather than source code. Generally speaking, the convergence rate of EKGA-BN is much faster while the score of the constructed networks is also improved comparing with other widely used BNs structure learning methods such as MWST, TAN, K2, HC, MMHC, and AESL-GA.

V. Conclusion

This paper has proposed a series of new operators into a novel efficient knowledge-driven GA for BNs structure learning problem. We fist predefined a maximum number of same individuals to maintains the population diversity. The higher population diversity makes EKGA-BN easier to overstep the local extreme, and therefore the identified BN



(a) ASIA

(b) ALARM

(c) HEPARII

(d) HEPARII

Fig. 4. Convergence rate of different algorithms in benchmark networks

structures will have a better performance. Then, the idea of HC algorithm is introduced to speed up searching each niche, which accelerate the convergence rate. Moreover, the elite set, which composed of well-performed individuals, is analysed to construct a common BN structure. The common BN structure is used to lead the search direction and enhance the local search ability. The experimental results reported in Section IV shows that the performance of EA based algorithm performs generally better than single solution score based search algorithms and constraint based algorithms. Comparing to other EA based algorithms, EKGA-BN is capable of constructing a near optimal networks with higher BDeu score and has faster convergence speed over four datasets. And EKGA-BN also shows robust and significant results especially in datasets which have higher number of variables.

However it is still a challenge to find the optimal structure in large data size. For future research, we would like to study the optimization mechanism of other state-of-the-art EA algorithms, and extend our study for BN learning problem with large data size.

ACKNOWLEDGMENT

This work was supported in part by the National Key R&D Program of China under Grant 2017YFC1601800 and 2017YFC1601000, in part by the National Natural Science foundation of China, under Grant 61673194, and Grant 61672263, in part by the Key Research and Development Program of Jiangsu Province, China, under Grant BE2017630, in part by "Blue Project" in Jiangsu Universities, in part by the Postdoctoral Science Foundation of China under Grant 2014M560390.

REFERENCES

[1] S. Gheisari and M. R. Meybodi, "Bnc-pso: structure learning of bayesian networks by particle swarm optimization," *Information Sciences*, vol. 348, pp. 272–289, 2016.

[2] J. Xuan, J. Lu, G. Zhang, R. Y. Da Xu, and X. Luo, "A bayesian nonparametric model for multi-label learning," *Machine Learning*, vol. 106, no. 11, pp. 1787–1815, 2017.

[3] F. L. Seixas, B. Zadrozny, J. Laks, A. Conci, and D. C. M. Saade, "A bayesian network decision model for supporting the diagnosis of dementia, alzheimer's disease and mild cognitive impairment," *Computers in biology and medicine*, vol. 51, pp. 140–158, 2014.

[4] S. Nikolopoulos, G. T. Papadopoulos, I. Kompatsiaris, and I. Patras, "Evidence-driven image interpretation by combining implicit and explicit knowledge in a bayesian network," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 41, no. 5, pp. 1366–1381, 2011.

[5] P. LarrañAga, H. Karshenas, C. Bielza, and R. Santana, "A review on evolutionary algorithms in bayesian network learning and inference tasks," *Information Sciences*, vol. 233, pp. 109–125, 2013.

[6] K. Pearson, "X. on the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 50, no. 302, pp. 157–175, 1900.

[7] P. Spirtes, C. N. Glymour, R. Scheines, and D. Heckerman, *Causation, prediction, and search*. MIT press, 2000.

[8] J.-P. Pellet and A. Elisseeff, "Using markov blankets for causal structure learning," *Journal of Machine Learning Research*, vol. 9, no. Jul, pp. 1295–1342, 2008.

[9] W. L. Buntine, "Operations for learning with graphical models," *Journal of artificial intelligence research*, vol. 2, pp. 159–225, 1994.

[10] F. Vafaee, "Learning the structure of large-scale bayesian networks using genetic algorithm," in *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*. ACM, 2014, pp. 855–862.

[11] J. Wang and S. Liu, "Novel binary encoding water cycle algorithm for solving bayesian network structures learning problem," *Knowledge-Based Systems*, vol. 150, pp. 95–110, 2018.

[12] L. M. De Campos, J. M. Fernandez-Luna, J. A. Gámez, and J. M. Puerta, "Ant colony optimization for learning bayesian networks," *International Journal of Approximate Reasoning*, vol. 31, no. 3, pp. 291–311, 2002.

[13] P. Larrañaga, M. Poza, Y. Yurramendi, R. H. Murga, and C. M. H. Kuijpers, "Structure learning of bayesian networks by genetic algorithms: A performance analysis of control parameters," *IEEE transactions on pattern analysis and machine intelligence*, vol. 18, no. 9, pp. 912–926, 1996.

[14] R. Kabli, F. Herrmann, and J. McCall, "A chain-model genetic algorithm for bayesian network structure learning," in *Proceedings of the 9th annual conference on Genetic and evolutionary computation*. ACM, 2007, pp. 1264–1271.

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

[16] N. Friedman, I. Nachman, and D. Peér, "Learning bayesian network structure from massive datasets: the «sparse candidate «algorithm," in *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 1999, pp. 206–215.

[17] C. Contaldi, F. Vafaee, and P. C. Nelson, "Bayesian network hybrid learning using an elite-guided genetic algorithm," *Artificial Intelligence Review*, vol. 52, no. 1, pp. 245–272, 2019.

[18] H. Akaike, "A new look at the statistical model identification," in *Selected Papers of Hirotugu Akaike*. Springer, 1974, pp. 215–222.

[19] Y. M. Shtar'kov, "Universal sequential coding of single messages," *Problemy Peredachi Informatsii*, vol. 23, no. 3, pp. 3–17, 1987.

[20] J. Suzuki, "A construction of bayesian networks from databases based on an mdl principle," in *Uncertainty in Artificial Intelligence*. Elsevier, 1993, pp. 266–273.

[21] F. Vafaee, G. Turan, P. C. Nelson, and T. Y. Berger-Wolf, "Among-site rate variation: adaptation of genetic algorithm mutation rates at each single site," in *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*. ACM, 2014, pp. 863–870.

[22] M. Scutari, "Learning bayesian networks with the bnlearn r package," *arXiv preprint arXiv:0908.3817*, 2009.

[23] 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.

[24] 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.

[25] A. Onisko, "Probabilistic causal models in medicine: Application to diagnosis of liver disorders," in *Ph. D. dissertation, Inst. Biocybern. Biomed. Eng., Polish Academy Sci., Warsaw, Poland*, 2003.

[26] C. Conati, A. S. Gertner, K. VanLehn, and M. J. Druzdzel, "On-line student modeling for coached problem solving using bayesian networks," in *User Modeling*. Springer, 1997, pp. 231–242.

[27] 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.

[28] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian network classifiers," *Machine learning*, vol. 29, no. 2-3, pp. 131–163, 1997.

[29] G. F. Cooper and E. Herskovits, "A bayesian method for the induction of probabilistic networks from data," *Machine learning*, vol. 9, no. 4, pp. 309–347, 1992.

[30] C. F. Aliferis, I. Tsamardinos, A. R. Statnikov, and L. E. Brown, "Causal explorer: A causal probabilistic network learning toolkit for biomedical discovery." in *METMBS*, vol. 3, 2003, pp. 371–376.

[31] P. Leray and O. Francois, "Bnt structure learning package: Documentation and experiments," *Laboratoire PSI, Universitè et INSA de Rouen, Tech. Rep*, 2004.

[32] K. Murphy *et al.*, "The bayes net toolbox for matlab," *Computing science and statistics*, vol. 33, no. 2, pp. 1024–1034, 2001.

[33] C. J. Van Rijsbergen, "Information retrieval," 1979.