Applied Artificial Intelligence

An International Journal

ISSN: 0883-9514 (Print) 1087-6545 (Online) Journal homepage: https://www.tandfonline.com/loi/uaai20

A Novel Method for Aggregation of Bayesian Networks without Considering an Ancestral Ordering

Vahid Rezaei Tabar & Fatemeh Elahi

To cite this article: Vahid Rezaei Tabar & Fatemeh Elahi (2018) A Novel Method for Aggregation of Bayesian Networks without Considering an Ancestral Ordering, Applied Artificial Intelligence, 32:2, 214-227, DOI: 10.1080/08839514.2018.1451134

To link to this article: https://doi.org/10.1080/08839514.2018.1451134

Published online: 21 Mar 2018.

APPLIED ARTIFICIAL INTELLIGENCE
2018, VOL. 32, NO. 2, 214–227
https://doi.org/10.1080/08839514.2018.1451134

A Novel Method for Aggregation of Bayesian Networks without Considering an Ancestral Ordering

Vahid Rezaei Tabar[a] and Fatemeh Elahi[b]

[a]Department of Statistics, Faculty of Mathematics and Computer Sciences, Allameh Tabataba'i University, Tehran, Iran; [b]Department of Information Technology Management, Faculty of Management, Kharazmi University, Tehran, Iran

ABSTRACT
A good method of combining Bayesian networks (BNs) should be a generic one that ensures a combined BN meets three important criteria of avoiding cycles, preserving conditional independencies, and preserving the characteristics of individual BN parameters. All combination methods assumed that there is an ancestral ordering shared by individual BNs. If this assumption is violated, then avoiding cycles may be inefficient.

In this paper, without considering an ancestral ordering, we introduce a novel method for aggregation of BNs. For this purpose, we first combine the BNs using the modification of the method introduced by Feng et al. We then use the simulated annealing algorithm for getting an acyclic graph in which the minimum arcs have been removed. Using this method, most of the conditional independencies are preserved. We compare the results of the proposed method with the two classical BNs combination methods; union and intersection, and hence to demonstrate the distinctive advantages of the proposed BNs combination method.

Introduction

The combination of Bayesian networks (BNs) has been shown to be an effective approach for combining knowledge structures to some extent (Pennock and Wellman 1999; Chickering 2002; Del Sagrado and Moral 2003). A good method of combining BNs should be a generic one that ensures a combined BN meets qualitative (avoiding cycles and preserving conditional independencies) and quantitative (preserving the characteristics of individual BN parameters) aggregations (Feng et al. 2014). All existing methods of combining BNs assume that the original BNs are equally important and there exist an ancestral ordering. Thus, most of these methods mainly focus on preserving conditional independencies and preserving the characteristics of individual BN parameters.

There are two different perspectives in preserving conditional independencies; the first aims at preserving all conditional independencies (intersection)

CONTACT Vahid Rezaei Tabar vhrezaei@gmail.com Department of Statistics, Faculty of Mathematics and Computer Sciences, Allameh Tabataba'i University, Tehran, Iran.

Color versions of one or more of the figures in the article can be found online at www.tandfonline.com/UAAI.

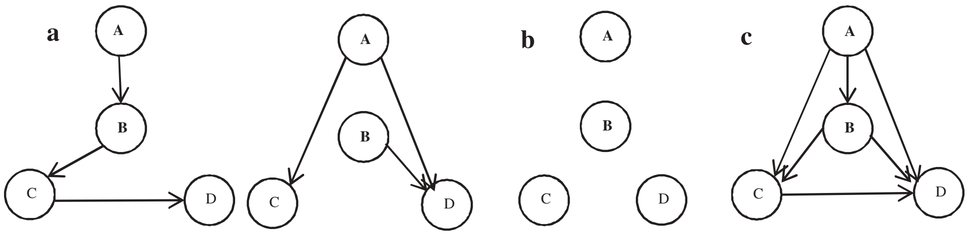


Figure 1. Two BNs and their intersection and union results: (a) two original BNs; (b) intersection; and (c) union (Feng et al. 2014).

while the second preserves only their common conditional independencies (union) (Figure 1).

Del Sagrado and Moral (2003) prove that the intersection and union of two BNs satisfy the two perspectives of preservations when the two BNs meet certain condition: no new v-structure is generated in the process of combining two BNs. The results of the intersection or the union of two BNs are incomplete because these methods combine only the structures. Therefore, Li, Liu, and Yue (2008) propose a method that deals with preservation of both conditional independencies and characteristics of individual parameters with a considerably rigid constraint. Feng et al. (2014) develop a novel method for combining BNs to address the weakness of existing methods. Their method is the only one that is generic and combines both structures and parameters of BNs. Following other existing BN combination methods, Feng et al. (2014) also assume that there is an ancestral ordering shared by individual BNs that helps avoid cycles. But this assignation is difficult, and the problem is actually a critical issue in combination of BN.

In this paper, without considering an ancestral ordering, we introduce a novel method for aggregation of BNs. For this purpose, we first combine the BNs with some modifications on the method introduced by Feng et al. (2014). We then use the simulated annealing (SA) algorithm for getting an acyclic graph in which the minimum arcs have been removed. We also compare the results of the proposed method with the other combination methods, and hence to demonstrate the distinctive advantages of the proposed BNs combination method.

The remainder of this paper is organized as follows: in section "Basic concepts," some definitions are introduced. In section "Feng et al. (2014) method," the combination method by Feng et al. is presented. Our proposed method is introduced in section "Proposed method." We finally compare our method with the other methods in the task of combining BN.

Basic concepts

Definition 1 (Conditional independency; Neufeld 1993). Let $U = \{\alpha,\ \beta, \ldots\}$ be a finite set of variables with discrete values. Let $P\ (.)$ be a joint probability distribution over the variables in U, and let X, Y, and Z stand for any three subsets of variables in U. X and Y are said to be conditionally independent given Z, denoted $I(X, Z, Y)$, if $P(X|Z, Y) = P(X|Y)$, whenever $P(Y, Z) > 0$.

Definition 2 (Bayesian networks; Neufeld 1993). A BN defined over a set of variables $X = \{X_1, \ldots, X_m\}$ is a pair $(G\langle X, E\rangle, P)$, where G is a directed acyclic graph (DAG) describing the structure of the BN and consisting of the set E of edges between variables. Then statistical meaning of this BN can be described completely and concisely by the conditional independencies set $Q = \{I(x, pa(x), nde(x))|x\epsilon X\}$, where $pa(x)$ the set of all parents of node x, $de(x)$ is the set of all descendants of node x, and $nde(x) = V - \{x\} - pa(x) - de(x)$ is the set of all non-descendants of node x. As well $P = \{P(X_1|pa(X_1), \ldots, P(X_m|pa(X_m)\}$ is a set of conditional probability distribution (CPD), i.e., parameters. Then its joint probability distribution over X can be defined as follows:

$$P(X) = \prod_{i=1}^{m} P(X_i|pa(X_i)$$

Definition 3 (Intersection of BNs; Del Sagrado and Moral 2003): Given two BNs $BN_1 = (G_1\langle X, E_1\rangle, P_1)$ and $BN_2 = (G_2\langle Y, E_2\rangle, P_2)$, $BN\cap = (G_\cap\langle X \cup Y, E_\cap\rangle, P_\cap)$ is called the intersection of BN_1 and BN_2, if each $P(Z|pa_{BN_\cap}(Z)) \in P_\cap$ satisfies the following three conditions: (1) when $Z \in X - Y$, $pa_{BN_\cap}(Z) = pa_{BN_1}(Z)$; (2) when $Z \in Y - X$, $pa_{BN_\cap}(Z) = pa_{BN_2}(Z)$; and (3) when $Z \in X\cap\, Y$, $pa_{BN_\cap}(Z) = [pa_{BN_1}(Z) \cup pa_{BN_2}(Z) - X\cap Y] \cup [pa_{BN_1}(Z) \cap pa_{BN_2}(Z)]$.

Definition 4 (Union of BNs; Del Sagrado and Moral 2003): Given two BNs $BN_1 = (G_1\langle X, E_1\rangle, P_1)$ and $BN_2 = (G_2\langle Y, E_2\rangle, P_2)$, $BN_\cup = (G_\cup\langle X \cup Y, E_\cup\rangle, P_\cup)$ is called the union of BN_1 and BN_2, if each $P(Z|pa_{BN_\cup}(Z)) \in P_\cup$ satisfies the following three conditions: (1) when $Z \in X - Y$, $pa_{BN_\cup}(Z) = pa_{BN_1}(Z)$; (2) when $Z \in Y - X$, $pa_{BN_\cup}(Z) = pa_{BN_2}(Z)$; and (3) when $Z \in X\cap\, Y$, $pa_{BN_\cup}(Z) = pa_{BN_1}(Z) \cup pa_{BN_2}(Z)$.

Definition 5 (Interior and exterior nodes; Feng et al. 2014): For two BNs $BN_1 = (G_1\langle X, E_1\rangle, P_1)$ and $BN_2 = (G_2\langle Y, E_2\rangle, P_2)$, $Z \in X\cap\, Y$ is an interior node in the two BNs, if $pa_{BN_1}(Z) \subseteq X\cap\, Y$ or $pa_{BN_2}(Z) \subseteq X\cap\, Y$. Otherwise, $Z \in X\cap\, Y$ is an exterior node. In other words, for an interior node, all its parents in at least one BN belong to the common variables, while the two sets of parents in two BNs of an exterior node contain at least one non-common variable, respectively.

For example, according to Figure 2, we have $X\cap\, Y = \{B, D\}$. Since $pa_{BN_2}(B) = \emptyset \subseteq \{B, D\}$, B is an interior node in two graphs, but $pa_{BN_1}(D) = \{A\} \subseteq \{B, D\}$ and $pa_{BN_1}(D) = \{C\} \subseteq \{B, D\}$, so D is an exterior node.

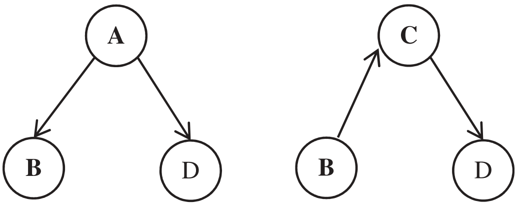


Figure 2. Two BNs with interior and exterior nodes (Feng et al. 2014).

Feng et al. (2014) method

Feng et al. (2014), without considering how many data samples the original BNs are generated from, introduced a novel method for combining BNs. Further, an ancestral ordering is assigned in advance by experts to avoid cycles. According to their method, for two BNs $BN_1 = (G_1\langle X, E_1\rangle, P_1)$ and $BN_2 = (G_2\langle Y, E_2\rangle, P_2)$, where $P_1 = \{P(X_1|pa_{BN_1}(X_1)), \ldots, P(X_m|pa_{BN_1}(X_m)\}$ and $P_2 = \{P(Y_1|pa_{B2_1}(Y_1)), \ldots, P(Y_n|pa_{BN_2}(Y_n))\}$, the result of combining BN_1 and BN_2 certainly has the formal $BN_* = (G_*\langle X\cup\ Y, E_*\rangle, P_*)$. Their combination method is proposed for combining $P_1$with P_2 in P_* as follows (Algorithm 1).

Algorithm 1. The main logics of the Feng et al. method

```
if $Z \in X - Y$
    $P(Z|pa_{BN_*}(Z)) = P(Z|pa_{BN_1}(Z))$
else if $Z \in Y - X$
    $P(Z|pa_{BN_*}(Z)) = P(Z|pa_{BN_2}(Z))$
else if $Z \in X\cap\ Y$ is an interior node
    $P(Z|pa_{BN_*}(Z))$ is determined by the deleting rule (Algorithm 2)
else $Z \in X\cap\ Y$ is an exterior node
    $P(Z|pa_{BN_*}(Z))$ is determined by the combining rule (Algorithm 3)
end if
```

Algorithm 2. The deleting rule for an interior variable

```
if $pa_{BN_k}(Z) \subseteq X\cap^{Y}(k = 1, 2)$
    Preserve $P(Z|pa_{BN_k}(Z))$
    Delete the other one
else
    Delete one of the $pa_{BN_1}(Z)$ and $pa_{BN_2}(Z)$ randomly
end if
```

Algorithm 2 includes two steps. First, if any one set of their parent variables meets the condition $pa_{BN_k}(Z) \subseteq X\cap^{Y}$, the corresponding CPD $P(Z|pa_{BN_k}(Z))$ will be preserved and the other will be removed. The reason to do this is that they intend to preserve the connectivity of the resulting BN. Second, if the condition in the first step is not met, it means $pa_{BN_1}(Z) \subseteq X \cap Y$ and $pa_{BN_2}(Z) \subseteq X \cap Y$, one of the two CPDs will be randomly deleted. Let us now consider two BNs in Figure 3(a), and then the combined BN is as shown in Figure 3(b).

Algorithm 3. The combining rule for an exterior variable

```
(1) Incorporate all parents: $pa(Z) = pa_{BN_1}(Z) \cup pa_{BN_2}(Z)$
(2) **for** each state $z_i$ of Z
(3)     Compute based on Eq.1
(4) **end for**
(5) **for** each state $z_i$ of Z
(6)     Normalize based on Eq.2
(7) **end for**
```

According to Algorithm 3, the combination of $P(Z|pa_{BN_1}(Z))$ and $P(Z|\, pa_{BN_2}(Z))$ into $P(Z|pa_{BN_1}(Z) \cup P(Z|pa_{BN_2}Z)))$ is given by

$$P(Z|\, pa_{BN_1}(Z) \cup P(Z|\, pa_{BN_2}(Z))) = \; P(Z|\, pa_{BN_1}(Z)) \oplus P(Z|\, pa_{BN_2}(Z)) \tag{1}$$

where

$$\begin{aligned} P(Z|pa_{BN_1}(Z)) \oplus P(Z|pa_{BN_2}(Z)) = P(Z|pa_{BN_1}(Z)) + P(Z|pa_{BN_2}(Z)) \\ - P(Z|pa_{BN_1}(Z).P(Z|pa_{BN_2}(Z) \end{aligned}$$

is the fuzzy fusion of parameters based on association degree superposition Liu and Song (2001). Note that the direct results of Eq. 1 may not satisfy

$$\sum\nolimits_{i=1}^{n} P(z_i|pa_{BN_1}(Z) \cup P(Z|pa_{BN_2}(Z))) = 1,$$

where $\{z_1, \ldots, z_n\}$ are n states of Z. Therefore by normalizing, we have

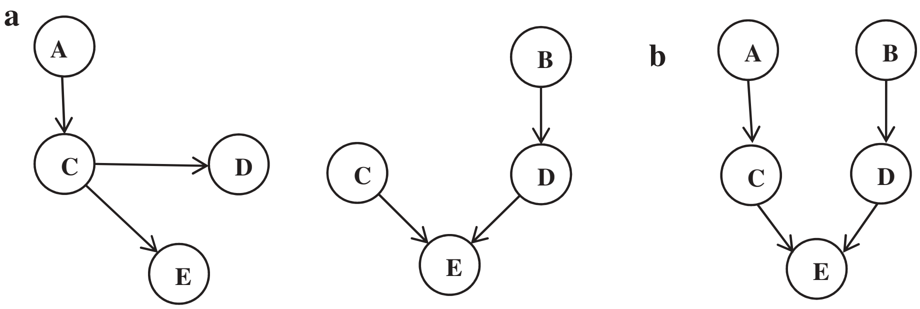
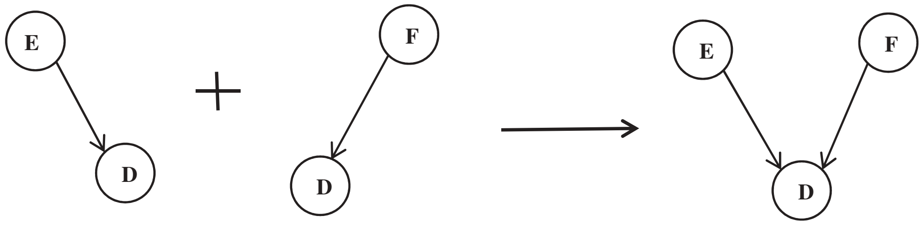


Figure 3. Two BNs and the combination results: (a) two BNs and (b) the combination (Feng, Zhang, and Liao 2014).



P(D/E)	e_1	e_2
d_1	0.3	0.6
d_2	0.7	0.4

+

P(D/F)	f_1	f_2
d_1	0.8	0.6
d_2	0.2	0.4

=

P(D/EF)	e_1f_1	e_1f_2	e_2f_1	e_2f_2
d_1	0.53	...	...	...
d_2	0.47	...	...	...

Figure 4. Combination of parameters (Feng, Zhang, and Liao 2014).

$$P'(z_i|pa_{BN_1}(Z) \cup P(Z|pa_{BN_2}(Z)) = \frac{P(z_i|pa_{BN_1}(Z) \cup P(Z|pa_{BN_2}(Z))}{\sum^{i=1} nP(z_i|pa_{BN_1}(Z) \cup P(Z|pa_{BN_2}(Z))) = 1} \tag{2}$$

Details of combinations of BN parameters are illustrated in Figure 4.

Proposed method

Without considering data

Regarding the approach adopted by existing BNs combination methods, Feng, Zhang, and Liao (2014) also assume that original BNs are equally important, i.e., without considering how many data samples the original BNs are generated from. Further, in their method, there is an ancestral ordering shared by individual BNs that helps avoid cycles. If this assumption is violated, then the avoiding cycles may be inefficient.

In this paper, we also assume that the original BNs are equally important. In addition, we assume that there is no ancestral ordering for two BNs. We first modify the method introduced by Feng, Zhang, and Liao (2014) with a change in Algorithm 2 as follows.

Algorithm 4. The deleting rule for an interior variable with a change

```
if $pa_{BN_k}(Z) \subseteq X \cap^Y (k = 1, 2)$
    Preserve $P(Z|pa_{BN_k}(Z))$
    Delete the other one
else if
 $pa_{BN_k}(Z) \subseteq X \cap Y\ (\forall k = 1, 2)$ and one parent is $\varnothing$
Preserve the other one
else
Delete one of the $pa_{BN_1}(Z)$ and $pa_{BN_2}(Z)$ randomly
end if
```

The reason of adding Step 5 in Algorithm 2 is that we intend to preserve the connectivity of the resulting BN. After applying the method introduced by Feng, Zhang, and Liao (2014) with this change, we then deal with a problem of having a cyclic graph. In mathematical language, this corresponds to a problem of making a graph acyclic by removing as few links as possible and thus altering the original graph in the least possible way. The exact solution of this problem requires enumeration of all cycles and combinations of removed links, which, as an NP-hard problem, is computationally prohibitive even for modest-size networks. For solving this problem, we use the approximate numerical algorithms based on the SA of the hierarchical layout of the network, which minimizes the number of "backward" links going from lower to higher hierarchical levels (Ispolatov and Maslov 2008). SA is a method for solving unconstrained and bound-constrained optimization problems. The method models the physical process of heating a material and then slowly lowering the temperature to decrease defects, thus minimizing the system energy. At each iteration of the SA algorithm, a new point is randomly generated. The distance of the new point from the current point, or the extent of the search, is based on a probability distribution with a scale proportional to the temperature. The algorithm accepts all new points that lower the objective, but also, with a certain probability, points that raise the objective. By accepting points that raise the objective, the algorithm avoids being trapped in local minima and is able to explore globally for more possible solutions. An annealing schedule is selected to systematically decrease the temperature as the algorithm proceeds. As the temperature decreases, the algorithm reduces the extent of its search to converge to a minimum. This algorithm outperforms the other algorithms in terms of speed, memory requirement, and the actual number of removed links (Ispolatov and Maslov 2008). Taken together, our proposed algorithm will be as follows (Algorithm 5).

Algorithm 5. Proposed algorithm (without considering data)

```
(1)  if $Z \in X - Y$
(2)      $P(Z|pa_{BN_*}(Z)) = P(Z|pa_{BN_1}(Z))$
(3)  else if $Z \in Y - X$
(4)      $P(Z|pa_{BN_*}(Z)) = P(Z|pa_{BN_2}(Z))$
(5)  else if $Z \in X \cap Y$ is an interior node
(6)  $P(Z|pa_{BN_*}(Z))$ is determined by the deleting rule (Algorithm 4)
(7)  else $Z \in X \cap Y$ is an exterior node
(8)      $P(Z|pa_{BN_*}(Z))$ is determined by the combining rule (Algorithm 3)
(9)  end if
(10) Make cyclic graph acyclic using SA algorithm (subsection 4.3)
```

With considering data

Here, we assume that two different methods are used to learn different BNs. In this case, we also assume that original BNs are equally important (i.e., two equivalent BNs). Therefore, the deleting rule for an interior variable will be as follows (Algorithm 6).

Algorithm 6. The deleting rule for an interior variable using Bayesian information criterion (BIC) score

```
if $pa_{BN_k}(Z) \subseteq X \cap^{Y} (k = 1, 2)$
    Preserve $P(Z|pa_{BN_k}(Z))$
    Delete the other one
else if
 $pa_{BN_k}(Z) \subseteq X \cap Y \ (\forall k = 1, 2)$ and one parent is $\varnothing$
Preserve the other one
else
Delete one of the $pa_{BN_1}(Z)$ and $pa_{BN_2}(Z)$ according to BIC score
end if
```

It means that we preserve that $pa_{BN_k}(Z)$, which has the higher BIC score between node Z and its parents. We call this score the partial score.

Make cyclic graph acyclic

After combining some BNs, if the resulting network is a cyclic graph, we must make it acyclic. We should consider two significant points to remove cycles:

(1) maintaining conditional independencies
(2) removing minimum number of edges

The problem "minimum feedback arc set (MFAS), or dually a maximum acyclic subgraph" is one of the 21 famous problems that Karp (1972) introduced as an NP-hard problem. A feedback arc set (FAS) or feedback edge set is a set of edges that, when removed from the graph, leave a DAG.

An exact solution for removing cycle contains two steps: (1) enumerating all cycles and (2) finding different combination of edges for getting the acyclic graph. This method not only has a high computational cost, but also we are not certain about enumeration of all possible combinations of edges. We implemented an algorithm for making acyclic graph. Our main idea is based on the method introduced by Ispolatov and Maslov (2008) in the field of bioinformatics.

Although there are different nondeterministic and greedy algorithms to solve the "MFAS" problem, according to what we will say farther about MFAS, using meta-heuristic methods seem more appropriate. Therefore, we chose SA because

SA compared to other methods has some outstanding advantages. For instance, it is able to escape from being stuck in local optimum point.

Methodology

Consider a directed graph that has **N** vertices and **L** directed edges. In the process of eliminating cycle, we distributed nodes in **M** levels hierarchically. In fact our goal is to arrange vertices in a manner that the number of anti-hierarchical[1] edges is minimal.

The edges specified as anti-hierarchical are the same as FAS. Deleting these edges leads to remove cycle from graph. The number of levels is at least equal to the graph's diameter[2] and less than or equal to N. When $M = N$ we say that every node is inserted in one level. By considering the above points and SA's parameters, the steps of implementation are as follows:

(1) Determine the initial temperature (T_0): the right choice and high enough temperature let us explore well in the process of SA.

SA manages risk to some extent, and it is another reason of its superiority over other methods. In fact in high temperature it accepts worse solutions probabilistically. However, SA approximates the real answer by decreasing temperature and exploiting. There are different ways of determining T_0. In this paper, we choose T_0 as a coefficient of (L/N).

(2) Generate a random rudimentary answer; this answer depicts sample of decorating nodes in some levels.

At each temperature, the objective function is defined based on the current structure, as follows:

$$\mathbf{E} = \mathbf{L}_{\text{opposite}} + \textbf{penalty value} \tag{3}$$

We consider E as energy, L_{opposite} as the number of anti-hierarchical edges, and penalty value is considered to make a better layout with minimum total length of hierarchical structure.

(3) Make a new answer in the current temperature. In any temperature, the best answer is saved.

In this step, the main question is "how to make the best answer in each temperature?" Based on the current answer, we make a new answer. It contains a new order of some nodes. So, the amount of target will be changed. We compare the new organizing of nodes with the current answer at this temperature. For this purpose, we have

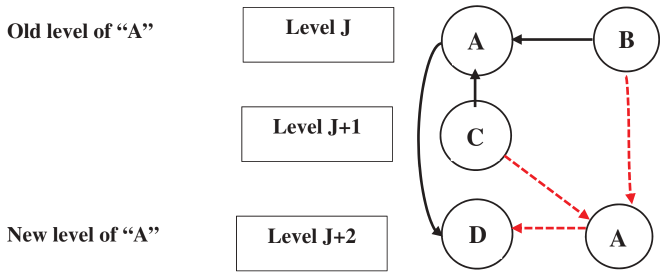


Figure 5. Links (B →A) and (C→A) change from anti-hierarchical to hierarchical but link (A→D) changes from hierarchical to anti-hierarchical; so ΔE = (−1)+(−1)+(1) = −1.

$$\text{If } E_{\text{new}} \leq E_{\text{old}} \rightarrow \text{solution} = \text{new solution}$$

Figure 5 shows that when the node "A" is transferred from level J to level J + 2, how much the amount of E is modified?

We can normalize and omit the effect of target's scale on T_0 as follows:

$$\Delta\mathrm{E}' = \frac{E_{\text{new}} - E_{\text{old}}}{E_{\text{old}}} \tag{4}$$

The acceptance probability of the new solution is computed as follows:

$$P = -\Delta E/T \tag{5}$$

As it is obvious if $\Delta E' \leq 0$ we accept new solution completely.

(4) Reduce the temperature.

SA interprets slow cooling as a slow decrease in the probability of accepting worse solutions as it explores the solution space.

The gradual reduction of temperature is done in a loop. The number of cycles in the loop is dependent on the process of temperature reduction. In this work, temperature is decreased exponentially such as follows:

$$T_k = \alpha T_{\mathrm{k}-1} = \alpha^2 T_{k-2} = \ldots == \alpha^k T_0 \rightarrow \; T_k = \mathrm{T}_0 \mathrm{e}^{\mathrm{kln}\alpha} \tag{6}$$

The amount of α is dependent on the size of problem, we set α = 0.9 in our study. Finally, we get the best result among all results of different temperatures.

Therefore, our program has two main loops in SA procedure:

- First loop: for decreasing temperature.
- Second and internal loop: It is a small loop that is set 10–20 times in each temperature. We start with a solution and move from one answer to neighboring one through this internal loop. In this way, every cycle of loop is implemented so that we get a better or worse answer.

We discussed the four main steps in SA. But determining the number of levels is also an important problem in SA. The total procedure of cycle elimination included a loop that its counter is started from $M = d$ (the amount of diameter) and is increased up to $M = N$, i.e., this loop is executed $(N - d + 1)$ times.

Note that in the case “without considering data” we determine the final acyclic graph according to the minimum number of removable links. If two different numbers of levels result in two acyclic graphs with the same number of removable links, we randomly choose one of them. In the case “with considering data,” making graph acyclic algorithm will be a dual purpose decision: (1) removing as few links as much as possible and (2) getting acyclic graph with a higher BIC score. It means that if two different number of levels result in two acyclic graphs with the same number of removing links, we choose the one that has a higher BIC score. This score is called the general BIC score. Whatever we did to remove cycle is presented in Figure 6.

Experimental results

In this section, we present the empirical results obtained. We focus on the case “with considering data.” For this purpose, we use four well-known networks: Asia, A Logical Alarm Reduction Mechanism (ALARM), insurance, and hailfinder, which have 8, 37, 27, and 56 variables, respectively. In addition, their actual networks have 8, 46, 52, and 66 edges.

The Asia network is a small synthetic from (Lauritzen and Spiegelhalter 1988) about lung diseases (tuberculosis, lung cancer, or bronchitis) and visits to Asia. The ALARM provides an alarm message system for patient

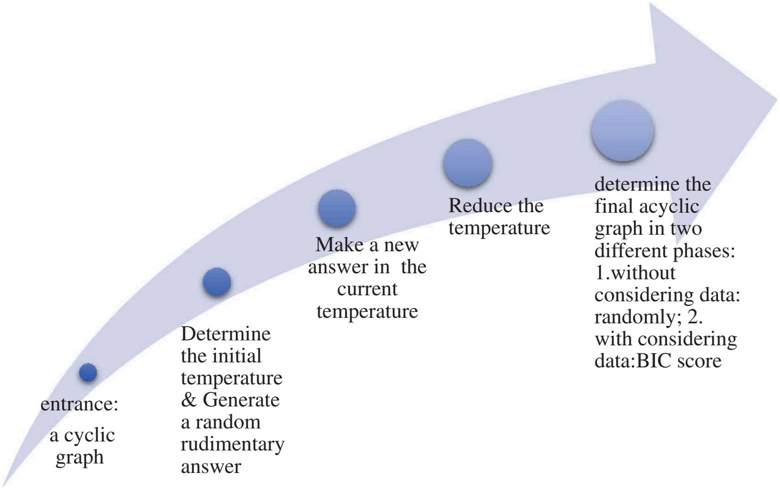

entrance:
a cyclic graph
Determine the initial temperature & Generate a random rudimentary answer
Make a new answer in the current temperature
Reduce the temperature
determine the final acyclic graph in two different phases: 1.without considering data: randomly; 2. with considering data:BIC score

Figure 6. The process of removing cycle.

monitoring (Beinlich et al. 1989). Insurance is a network for evaluating car insurance risks (Binder et al. 1997). Hailfinder is a BN designed to forecast severe summer hail in northeastern Colorado (Abramson et al. 1996).

We conducted learning exercises using 2000, 5000, 10,000, and 20,000 cases, and evaluated the performance of the proposed method. Note that for the Asia network we only focus on the 10,000 and 20,000 cases.

For combining two BNs, we use two hybrid learning algorithms: max–min hill-climbing (MMHC) and restricted maximization (RSMAX2) (Tsamardinos, Brown, and Aliferis 2006). The MMHC combines the max–min parents and children algorithm to restrict the search space and the hill-climbing algorithm to find the optimal network structure in the restricted space. The RSMAX2 is a more general implementation of the max–min hill climbing, which can use any combination of constraint-based and score-based algorithms.

We compare the results of our method with the union and intersection methods for combining two BNs. Note that making acyclic graph in union and intersection has also been done by the SA algorithm. The BIC scores for aggregated BNs are shown in Table 1. Results show that the proposed method has a higher BIC score in the task of combining BNs for all data sets.

We also use the edge scores by computing the number of edges that are correct, missing, reverse, and additional edges. The edge scores make it possible to define the important terms, which indicate the performance of the method. For this purpose, the true positive (TP), false positive (FP), true negative (TN), and false negative (FN) values are computed. In addition, known measures such as accuracy (ACC) and *F*-score measure (*F*-measure) are considered (Baesens et al. 2002). These measures are defined as follows:

Table 1. Result of different combining methods for different data sets using 'BIC' score.

Data set	Sample size	Intersection	Union	Proposed method	MMHC	RSMAX2
Asia	10,000	−4935.6	−4861.2	−4861.2	−4507	−4943.6
	20,000	−12277.5	−12074.8	−12074.8	−11109.7	−12286.9
	Mean	−8606.5	−8468	−8468	−7808.3	−8615.2
ALARM	2000	−38618.2	−28586.6	−27196.9	−30090	−33701.8
	5000	−94421.3	−68893.6	−67540.5	−71204.7	−83888.4
	10,000	−183365.4	−141931.5	−136929.4	−141196.1	−164266.6
	20,000	−389466.8	−244893.1	−236921.9	−254684.3	−336979.7
	Mean	−176467.9	−121076.2	−117147.1	−124293.7	−154709.1
Hail	2000	−123609.4	−114968.8	−114788.2	−106881.2	−122896.4
	5000	−306227.1	−280481.8	−280481.8	−255317.1	−303870.8
	10,000	−611046.95	−554314.9	−554227.5	−501569.4	−605882.5
	20,000	−1227829.3	−1109499.3	−1109499.3	−993435.6	−1218087.9
	Mean	−567178.1	−514816.2	−514749.2	−464300.8	−562684.4
Insurance	2000	−37200.7	−30829.7	−29961.7	−29673.96	−34207.2
	5000	−89753.9	−74451.7	−73733.1	−69474.8	−80619.5
	10,000	−167507.7	−142219	−140313.5	−134843.7	−162019.6
	20,000	−328180.1	−280097	−280097	−265858.5	−323361.1
	Mean	−155660.6	−131899	−131026.3	−124962.7	−150051.8

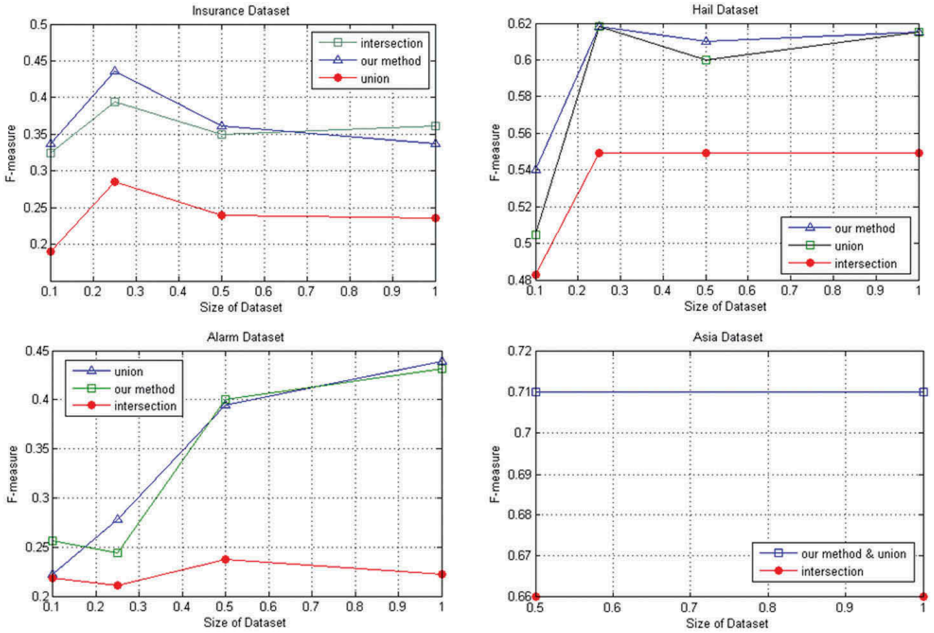


Figure 7. Comparison of the accuracy of the three methods.

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{FDR} = \frac{\text{FP}}{\text{FP} + \text{TP}}, \quad \text{ACC} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}$$

$$\text{PPV} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad F - \text{measure} = 2\frac{\text{PPV}.\ \text{TPR}}{\text{PPV} + \text{TPR}},$$

F-measure is a quantity used to compare learned and actual networks. The method with larger values of *F*-measure is more efficient in learning the skeleton of the network. We compute the *F*-measures for only the 20,000 cases and compare the results of intersection, union, and the proposed method. Results are shown in Figure 7.

Note that it is easy to extend the results of this paper to *n* networks instead of two.

Notes

1. Anti-hierarchical edge is an edge where its source and destination are in the same level or its destination's level is in a higher level than the source's level.
2. Graph's diameter is the maximum distance in a graph between all nodes.

References

Abramson, B., J. Brown, W. Edwards, A. Murphy, and R. L. Winkler. 1996. Hailfinder: A Bayesian system for forecasting severe weather. *International Journal of Forecasting* 12 (1):57–71. doi:10.1016/0169-2070(95)00664-8.

Baesens, B., M. Egmont-Petersen, R. Castelo, and J. Vanthienen (2002). Learning bayesian network classifiers for credit scoring using markov chain monte carlo search. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on* (Vol. 3, pp. 49-52). IEEE.

Beinlich, I. A., H. J. Suermondt, R. M. Chavez, and G. F. Cooper. 1989. *The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks*, In Second European Conference on Artificial Intelligence in Medicine, pp. 247–56. Springer Berlin Heidelberg.

Binder, J., D. Koller, S. Russell, and K. Kanazawa. 1997. Adaptive probabilistic networks with hidden variables. *Machine Learning* 29 (2–3):213–44. doi:10.1023/A:1007421730016.

Chickering, D. M. 2002. Learning equivalence classes of Bayesian-network structures. *Journal of Machine Learning Research* 2 (Feb):445–98.

Del Sagrado, J., and S. Moral. 2003. Qualitative combination of Bayesian networks. *International Journal of Intelligent Systems* 18 (2):237–49. doi:10.1002/(ISSN)1098-111X.

Feng, G., J. D. Zhang, and S. S. Liao. 2014. A novel method for combining Bayesian networks, theoretical analysis, and its applications. *Pattern Recognition* 47 (5):2057–69. doi:10.1016/j.patcog.2013.12.005.

Ispolatov, I., and S. Maslov. 2008. Detection of the dominant direction of information flow and feedback links in densely interconnected regulatory networks. *BMC Bioinformatics* 9.1:1.

Karp, R. M. 1972. Reducibility among combinatorial problems. In Complexity of Computer Computations. The IBM Research Symposia Series. Eds., Miller R.E., Thatcher J.W., Bohlinger J.D., 85–103. Boston, MA: Springer.

Lauritzen, S. L., and D. J. Spiegelhalter. 1988. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society. Series B (Methodological)* 157–224.

Li, W., W. Liu, and K. Yue. 2008. Recovering the global structure from multiple local Bayesian networks. *International Journal on Artificial Intelligence Tools* 17 (06):1067–88. doi:10.1142/S0218213008004308.

Liu, W. Y., and N. Song. 2001. The fuzzy association degree in semantic data models. *Fuzzy Sets and Systems* 117 (2):203–08. doi:10.1016/S0165-0114(98)00334-0.

Neufeld, E. 1993. Pearl Judea. probabilistic reasoning in intelligent systems: Networks of plausible inference. Series in representation and reasoning. Morgan Kaufmann, San Mateo 1988, xix+ 552 pp. *The Journal of Symbolic Logic* 58 (02):721–721. doi:10.2307/2275238.

Pennock, D. M., and M. P. Wellman (1999). Graphical representations of consensus belief. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence* (pp. 531–40). Morgan Kaufmann Publishers Inc.

Tsamardinos, I., L. E. Brown, and C. F. Aliferis. 2006. The max-min hill-climbing Bayesian network structure learning algorithm. *Machine Learning* 65 (1):31–78. doi:10.1007/s10994-006-6889-7.