OPEN ACCESS Freely available online

PLOS ONE

Using Consensus Bayesian Network to Model the Reactive Oxygen Species Regulatory Pathway

Liangdong Hu, Limin Wang*

Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun, Jilin, P. R. China

Abstract

Bayesian network is one of the most successful graph models for representing the reactive oxygen species regulatory pathway. With the increasing number of microarray measurements, it is possible to construct the Bayesian network from microarray data directly. Although large numbers of Bayesian network learning algorithms have been developed, when applying them to learn Bayesian networks from microarray data, the accuracies are low due to that the databases they used to learn Bayesian networks contain too few microarray data. In this paper, we propose a consensus Bayesian network which is constructed by combining Bayesian networks from relevant literatures and Bayesian networks learned from microarray data. It would have a higher accuracy than the Bayesian networks learned from one database. In the experiment, we validated the Bayesian network combination algorithm on several classic machine learning databases and used the consensus Bayesian network to model the *Escherichia coli*'s ROS pathway.

Citation: Hu L, Wang L (2013) Using Consensus Bayesian Network to Model the Reactive Oxygen Species Regulatory Pathway. PLoS ONE 8(2): e56832. doi:10.1371/journal.pone.0056832

Editor: Frank Emmert-Streib, Queen's University Belfast, United Kingdom

Received May 19, 2012; **Accepted** January 16, 2013; **Published** February 15, 2013

Funding: This research is supported by Special Science Foundation for the Doctoral Program of China (No. 200801831011) and Postdoctoral Science Foundation of China (No. 20100481053). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: wanglim@jlu.edu.cn

Introduction

Reactive Oxygen Species (ROS) are formed as by-products of normal metabolism of aerobic organisms, they can react with DNA and produce damage [1]. Cells protect themselves from ROS by detoxification mechanisms and repair mechanisms [2,3]. Microarray is a powerful tool for measuring a large number of genes' expressions. Given the microarray expressions, it is possible to construct the regulatory pathway that organisms response to the oxidative stress directly.

An outstanding idea is the use of Bayesian network for representing regulatory pathway [4–7]. Bayesian network is a Directed Acyclic Graph (DAG) used for representing probabilistic relationships between variables. It was first proposed by Pearl [8], and Jensen [9] gave an intuitive definition. A lot of work has been done in the automatic learning of Bayesian network from database. Consequently, large numbers of Bayesian network learning algorithms based on different methodologies have been developed [10–13] and they have high accuracies in learning Bayesian networks from classic machine learning databases. However, when applying these algorithms to learn Bayesian networks from microarray data, the accuracies are low. Careful studies show that this is because the databases they used to learn Bayesian networks contain too few microarray data. On the other hand, microarray chip is expensive, it is difficult to obtain a large number of microarray data from one laboratory or one database, and a few hundred expression data can not guarantee a high learning accuracy.

To overcome this problem, we propose a consensus Bayesian network which is constructed by combining several Bayesian networks. This consensus Bayesian network is approximately equal to the Bayesian network learned from the database obtained by merging all these combined Bayesian networks' corresponding databases, then its equivalent database may have enough data and the accuracy can be improved. The main procedure of construction of consensus Bayesian network can be described as follow: (1) Review all relevant literatures and derive the Bayesian networks. (2) Search microarray expressions which are not used in relevant literatures and download them to learn Bayesian networks. (3) Combine all these Bayesian networks to construct the consensus Bayesian network.

Combination of Bayesian networks includes combination of graph models and aggregation of probability distributions [14–17]. Utz [18] proposed a method to combine many different Bayesian networks into an undirected graph, and each edge in the graph has a weight represents the frequency with which the edge occurs in the component networks. Zhang et al. [19] proposed a method for fusing Bayesian networks. They construct an initial network based on the union and intersection of the Bayesian networks, and then search for the structure which maximizes the scoring function(CH criterion). Our Bayesian network combination algorithm is based on the properties of probability. Due to probabilistic independence, Conditional Probability Tables (CPTs) can be extended, then corresponding nodes' CPTs can be changed into a same form and the aggregation function can be applied to these CPTs. After extending every corresponding CPTs, the combination of Bayesian networks changed into the aggregations of every corresponding nodes' CPTs if these Bayesian networks' variables' prior orders are consistent with each other. Some nodes' CPTs were extended previously, so they may have bogus parents after combination, then we should find them, delete the bogus edges and simplify the CPTs. The combination algorithm can also be applied to the

combination of Bayesian networks defined over different variable sets by using the extension of Bayesian network.

*Escherichia coli MG*1655 was used in the experiment, a constructed ROS pathway was derived from the literature wrote by Hodges et al. [20] and 612 microarray expression data were downloaded from the Many Microbe Microarrays Database(M3D) [21]. 27 genes were identified from the EcoCyc [22] ROS detoxification pathway. A consensus Bayesian network using the 27 genes as variables was constructed by combining the Bayesian network from the literature and the Bayesian network learned from the 612 microarray expressions. For demonstrating the combination of Bayesian networks defined over different variable sets, we used a prediction program to find genes may be involved in the ROS pathway, learned a Bayesian network which using the 27 genes and the newly found genes as variables, and then combined this Bayesian network and the Bayesian network from the literature to construct a new consensus Bayesian network.

Results

Validation on classic machine learning databases

In order to validate whether the consensus Bayesian network BN_c constructed by combining Bayesian networks BN_1 and BN_2 is equivalent to the Bayesian network learned from the database obtained by merging the two Bayesian networks' corresponding databases DB_1 and DB_2 or not, 6 databases were downloaded from the UCI Machine Learning Repository (http://archive.ics.uci.edu/ml/), and the databases of ALARM net and Chest-clinic net were generated by the BN PowerConstructor. For each database DB, we chose n samples (about $1/3 \sim 1/2$ of the samples in DB) randomly and used them as DB_1, the rest samples in DB were used as DB_2. Two Bayesian networks BN_1 and BN_2 were learned from DB_1 and DB_2, respectively. Consensus Bayesian network BN_c was constructed by combining BN_1 and BN_2. After that, another Bayesian network BN used as a reference was learned from DB, BN_c was compared with BN and the proportion of the number of identical edges between BN_c and BN to the total number of edges in BN_c and BN (similarity S) was computed. The program was run 100 times to compute the average similarity. All results of the experiments are shown in Table 1. Table 1 shows that all the average similarities are greater than 75%. So, consensus Bayesian network BN_c is approximately equal to Bayesian network BN. Although the combination algorithm is validated with 8 different databases and the types of data in these databases are very different, it doesn't affect the results. The Bayesian network learning algorithm just compute the distributions by counting the number of samples, and determine the relationships between the variables by analyzing the distributions. The Bayesian network combination algorithm is used to combine Bayesian networks and it doesn't involve the data. So, the type of data doesn't affect the validation.

Consensus Bayesian network BN_c is approximately equal to Bayesian network BN, then we can view BN's database DB as BN_c's equivalent database, and DB have more samples than DB_1 or DB_2. So, the use of consensus Bayesian network helps to solve the problem of lack of data in partial databases and the accuracy can be improved. The true structures of ALARM net and Chest-clinic net are known. Then we compared the learned networks with the known networks, the results are shown in Table 2. Table 2 shows that BN_c has a higher accuracy than BN_1 or BN_2.

Construction of consensus Bayesian network for modeling *Escherichia coli*'s ROS pathway

The consensus Bayesian network is constructed by combining Bayesian networks derived from literatures and Bayesian networks learned from microarray data. First, relevant literatures were reviewed and a ROS pathway was derived from the literature wrote by Hodges et al.[20], denoted as BN_1. In the literature, 27 genes identified from the EcoCyc ROS detoxification pathway were chosen as variables and 305 microarray expressions were used to learn the Bayesian network. Second, microarray data was searched and a microarray expression build with 612 microarray expressions was downloaded from the M3D database. Then Bayesian network BN_2 which also uses the 27 genes as variables was learned from these microarray expressions. Finally, consensus Bayesian network BN_c was constructed by combining these two Bayesian networks, and the result is shown in Figure 1. In the combination program, we take weights $W_1 = 305$, $W_2 = 612$ and threshold $e = 0.026$.

A novel prediction algorithm based on the computation of mutual information was developed to identify genes which are strongly associated with a particular gene in the regulatory pathway. If R is a gene in the regulatory pathway, gene O is strongly associated with R, then O may work together with R and also be involved in the pathway. The main procedure of this algorithm can be described as follow: assume set $\mathbf{R}$ includes all the known genes in the regulatory pathway, and set $\mathbf{O}$ includes the rest genes of the organism. Choose one gene O_i in $\mathbf{O}$, for each gene $R_j \in \mathbf{R}$, compute the mutual information $I(O_i; R_j)$, if $I(O_i; R_j) > \varepsilon$, it means gene O_i is related to gene R_j, then O_i may be involved in the pathway too. The program is ended until every gene in $\mathbf{O}$ has been tested.

27 genes identified from the EcoCyc ROS detoxification pathway were used as set $\mathbf{R}$, while the rest genes in *Escherichia coli* were used as set $\mathbf{O}$. The program found 4 genes may be involved in the ROS pathway, and the results are shown in Table 3. A new Bayesian network BN_2' using the 31 (27+4) genes as variables was learned from the 612 microarray expressions. BN_2' contains more genes than BN_1, so BN_1 was extended into $BN_1 \oplus BN_2'$. Then a new consensus Bayesian network BN_c' was constructed by combining BN_2' and $BN_1 \oplus BN_2'$, and the result is shown in Figure 2.

Discussion

In the discussion, we address this question: does the Bayesian network learned from microarray expressions match with a known regulatory pathway?

Before answering this question, we carried out an experiment. The procedure of the experiment can be described as follow: assume that V includes all of the genes of *Escherichia coli*, and then we construct an undirected graph $G_V = (V, E)$, where $E = \{(X_i, X_j) | X_i, X_j \in V, I(X_i, X_j) > \varepsilon\}$. Let C be the largest connected subgraph of G_V. Then 99.3% of the genes in *Escherichia coli* were included in C. Mutual information $I(X_i, X_j) > \varepsilon$ means genes X_i and X_j are interacted, so this phenomenon shows that almost all genes in *Escherichia coli* are related directly or indirectly. We can infer that some genes may be involved in different regulatory pathways, simultaneously. Otherwise, if there is no gene be involved in more than one regulatory pathway, that is, the regulatory pathways in *Escherichia coli* have no intersection, then we can't observe the phenomenon that thousands of genes related directly or indirectly. On the other hand, before microarray measurements, the *Escherichia coli* was alive, so almost all of the regulatory pathways of *Escherichia coli* were at work. Then although two genes must be interacted if there

Table 1. Validation of the combination algorithm.

Database			similarity	$T(s)$	similarity	$T'(s)$
Letter Recognition	17	20000	100.0%	0.000009	100.0%	0.000010
Shuttle	10	14500	100.0%	0.000008	100.0%	0.000008
Parkinsons Telemonitoring	26	5875	79.4±2.2%	0.086804	77.9±1.2%	1437.502573
Image Segmentation	20	2310	80.6±1.7%	0.066748	78.0±1.9%	835.820385
Contraceptive Method Choice	10	1473	83.2±2.1%	0.033214	82.5±2.5%	18.325412
Solar Flare	13	1389	75.0±3.0%	0.043424	76.6±2.5%	261.702598
ALARM net	37	10000	97.8±2.2%	0.123528	95.6±2.2%	372.952340
Chest-clinic net	8	1000	93.4±0.4%	0.026708	93.4±0.4%	12.259816

Where is the number of variables in the database, is the number of samples in the database, similarity is the average proportion of the number of identical edges between and to the total number of edges in and , and is the execution time of the Bayesian network combination program. The table shows that the similarity is depend on the number of samples, this is because the algorithms are based on the computation of probabilities and the accuracy of computation of probability is sensitive to the number of samples. Specifically, there are two reasons: (a)The real distributions of variables can't be reflected if the database only have several samples; (b)The equation we used to compute the probabilities is sensitive to the number of samples. Then in the experiments on and , similarity , this is because the two databases have enough samples and can provide enough information for constructing the real Bayesian networks, then the learned Bayesian networks , and are completely the same. So, consensus Bayesian network , and and are the same. Similarity and execution time are the results of the experiments using the fusion method proposed by Zhang et al. [19] instead of our combination algorithm. and show that our algorithm works more efficiently. The time complexity of our algorithm is , where is the number of nodes in the network. However, the execution time of Zhang's fusion method grows exponentially as the size of the biggest clique in the Clique tree increases.

doi:10.1371/journal.pone.0056832.t001

is a directed edge between them in the Bayesian network, it is hard to determine the directed edge belongs to which regulatory pathway. For example, there is a directed edge between *marA* and *marR* in BN_c (Figure 1), then there must be an interaction between *marA* and *marR*. They are involved in the regulation of transcription (EcoCyc database) and this biological process was working when measuring the expressions of these genes using microarray, therefore, the existence of *marR*→*marA* maybe due to that they are regulating the transcription. However, the ROS detoxification pathway (EcoCyc database) also contains *marA* and *marR*, then the existence of *marR*→*marA* maybe due to that they are regulating the response to the oxidative stress. So, it is hard to determine the directed edge *marR*→*marA* belongs to which regulatory pathway. If there is no edge between two genes in the Bayesian network, then the two genes are not interacted directly in any regulatory pathway. So, if a known regulatory pathway contains n genes and we use these n genes as variables to learn a Bayesian network from microarray expressions. Then all of the interactions between the n genes are contained in the Bayesian network, however, some of these interactions may not contained in this regulatory pathway. This means the regulatory pathway is a subgraph of the Bayesian network. Although the Bayesian network is not equivalent to the regulatory pathway, it still has important significance. With its guidance, the number of biological experiments could be greatly reduced when modeling a regulatory pathway.

Table 2. Comparison of the accuracies.

	ALARM			Chest-clinic		
	BN_1	BN_2	BN_c	BN_1	BN_2	BN_c
$\|E_n\|$	52	49	48	10	12	8
$\|E_m\|$	0	1	0	1	0	0
$\|E_e\|$	6	4	2	3	4	0

Where $|E_n|$ is the number of edges in the Bayesian network, $|E_m|$ is the number of missing edges, $|E_e|$ is the number of extra edges. The true structures of ALARM net and Chest-clinic net contain 46 directed edges and 8 directed edges, respectively.

doi:10.1371/journal.pone.0056832.t002

Methods

Data preprocessing

The algorithms can only process discrete data in this paper. However, the 612 microarray expression data of *Escherichia coli MG*1655 downloaded from the M3D database are continuous. Then expression data for each gene was discretized using a maximum entropy approach which uses three equally-sized bins (q3 quantization). And the genes' expressions were divided into three categories: underexpressed, normal, overexpressed.

Usually, Bayesian networks derived from literatures only have a structure, then we have three ways to obtain the parameters: (1) If the program of the learning algorithm is available on the internet, then both the structure and the parameters of the Bayesian network can be obtained by run the program directly. (2) If the microarray data used in the literatures were collected in a database available on the internet, then we can download these microarray data to learn the parameters. (3) Sometimes the corresponding database is unable to be found, or the Bayesian network is not learned form database, but constructed by biological experiments directly. Then distribution for each node can be estimated by analyzing the genes' special characteristics and the relationships between genes.

Bayesian network

A Bayesian network defined over a variable set V can be represented as a pair (G,P), where G is a DAG and each directed edge in the DAG represents a dependence, P is a group of CPTs and each node in the DAG has a CPT. Usually, G is called Bayesian network's structure and can be represented as a pair (V,E), where E is the edge set; P is called Bayesian network's parameter. G is a directed acyclic graph, that is, the nodes in G have a topological order, and we call it prior order. Let BN_1 and BN_2 be two Bayesian networks and their DAGs are $G_1=(V_1,E_1)$

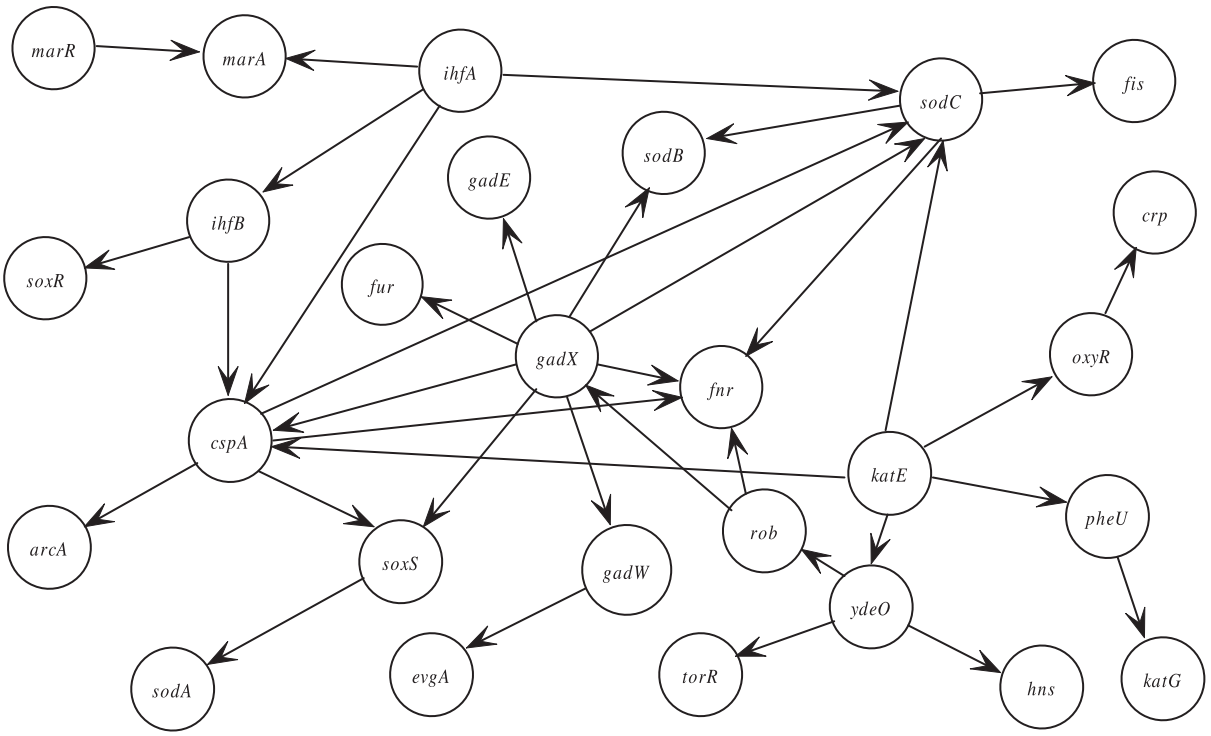

Figure 1. Consensus Bayesian network BN_c**.** 27 genes were identified from the EcoCyc ROS detoxification pathway.
doi:10.1371/journal.pone.0056832.g001

and $G_2=(V_2,E_2)$, respectively, then BN_1 and BN_2's variables' prior orders are consistent with each other if $G'=(V_1\cup V_2,E_1\cup E_2)$ is acyclic. Let A be a node in G, A's direct precursor nodes are called its parents, denoted as $Pa(A)$, then A's CPT represents the conditional probability $P(A|Pa(A))$. Suppose we have the CPT of A as shown in Figure 3(c), it shows B is a parent of A and x_{00} means $P(A=a_0|B=b_0)=x_{00}$. Assume that CPT cpt_1 represents $P_1(A|B,C)$, cpt_2 represents $P_2(A|B,C)$ and cpt_3 represents $P_3(A|C)$. Then cpt_1 and cpt_2 are two tables with the same structure and the conditional probabilities in the corresponding positions of the two tables represents the same conditional probability, so we say they have a same form. While cpt_1 and cpt_3 do not have a same form.

Bayesian network learning algorithm

Usually, Bayesian network is learned from database, it represents the probabilistic relationships between the variables in the database. So, a Bayesian network matches with a database, and we call this database Bayesian network's corresponding database. Bayesian network learning includes structure learning and parameter learning. We use an information-theory based learning algorithm proposed by Cheng et al. [11] to learn Bayesian network's structure in this paper.

Dependence between two variables can be quantitatively computed by using mutual information. Mutual information $I(X_i;X_j)$ between two variables X_i and X_j can be defined as:

$$I(X_i;X_j)=\sum_{x_i,x_j}P(x_i,x_j)log\frac{P(x_i,x_j)}{P(x_i)P(x_j)} \quad (1)$$

where x_i, x_j are the expression values of X_i and X_j, respectively. Mutual information is non-negative, it means $I(X_i;X_j)\geq 0$. $I(X_i;X_j)=0$ holds if and only if X_i and X_j are independent. Given a threshold ε ($\varepsilon>0$), X_i and X_j are related if $I(X_i;X_j)>\varepsilon$. Similarly, conditional mutual information $I(X_i,X_j|X_k)$ can be defined as:

$$I(X_i;X_j|X_k)=\sum_{x_i,x_j,x_k}P(x_i,x_j,x_k)log\frac{P(x_i,x_j|x_k)}{P(x_i|x_k)P(x_j|x_k)} \quad (2)$$

Then the main procedure of Cheng's Bayesian network structure learning algorithm can be described as follow:

Step 1. Create initial undirected graph. A Maximum Weight Span Tree (MWST) [23] is used as the initial graph. Let $L=\{(X_i,X_j)|X_i,X_j\in V,I(X_i;X_j)>\varepsilon\}$ be an undirected edge list, where V is the variable set. Sort L in descending order of mutual information. For each $(X_i,X_j)\in L$, add it into the undirected graph(and delete it from L) if it doesn't form a circle. End this loop until the graph contains $n-1$ edges.

Step 2. Add edges. Assume that set $D_{X_i}(X_i,X_j)$ contains all the nodes which are in the paths between X_i and X_j and in the neighborhood of X_i, simultaneously. $D_X(X_i,X_j)$ represents one of sets $D_{X_i}(X_i,X_j)$ and $D_{X_j}(X_i,X_j)$ which contains less nodes. For each $(X_i,X_j)\in L$, add it into the undirected graph(and delete it from L) if $I(X_i;X_j|D_X(X_i,X_j))>\varepsilon$ holds.

Step 3. Remove redundant edges. For each edge (X_i,X_j) in the undirected graph, delete it if $I(X_i;X_j|D_X(X_i,X_j))<\varepsilon$ holds.

Step 4. Determine edges' directions. For each X_i-Y-X_j, direct them $X_i\rightarrow Y\leftarrow X_j$ if

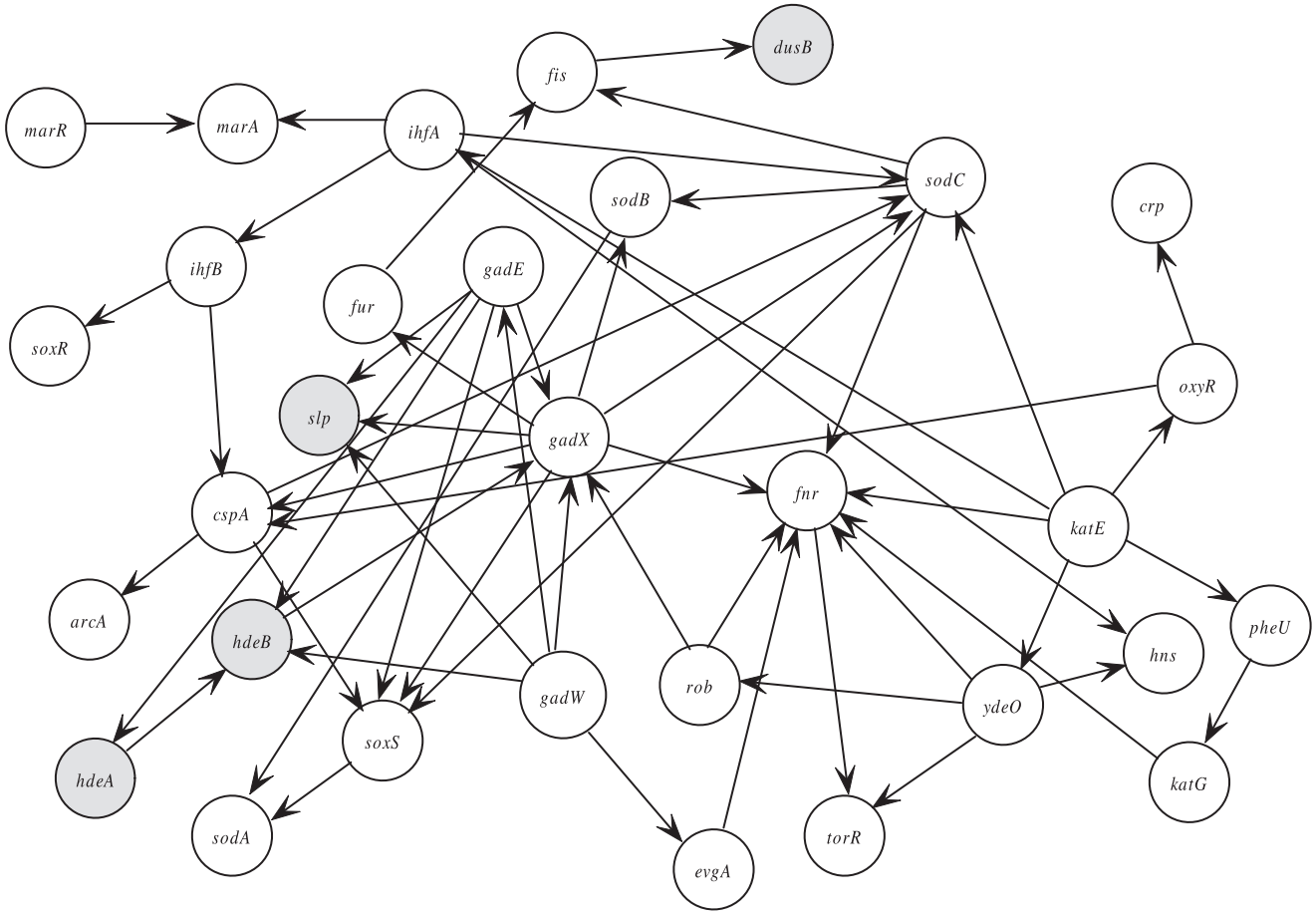

Figure 2. Consensus Bayesian network $BN_c^{'}$. 27 genes were identified from the EcoCyc ROS detoxification pathway, while genes *dusB, hdeB, slp, hdeA* were identified by the prediction program.
doi:10.1371/journal.pone.0056832.g002

$$\frac{I(X_i; X_j|Y)}{I(X_i; X_j)} > (1+\delta) \tag{3}$$

holds, where threshold $\delta > 0$. Some undirected edges' directions can be determined by using Bayesian network's acyclic property. For the rest undirected edges, use the local Minimal Description Length (MDL) score [24] to choose the direction which makes the MDL score more smaller.

Table 3. 4 genes identified by the prediction program.

Gene O	Gene R	Mutual information $I(O,R)$
dusB	*fis*	0.6599
hdeA	*gadE*	0.5559
hdeB	*gadE*	0.5811
slp	*gadE*	0.5689

Genes *fis*, *gadE* were identified from the EcoCyc ROS detoxification pathway. The interactions between gene O and gene R can also be found in EcoCyc database.
doi:10.1371/journal.pone.0056832.t003

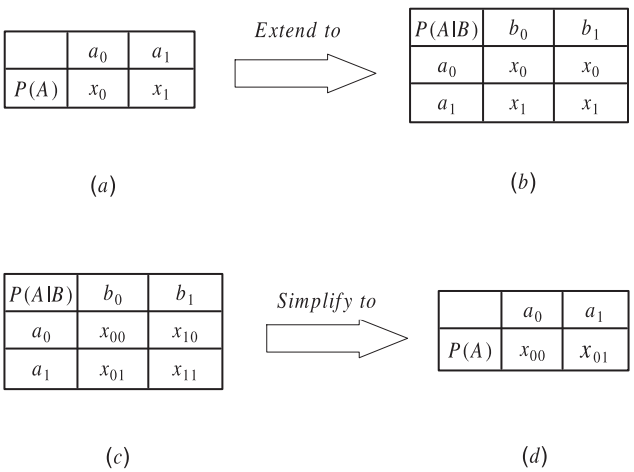

Figure 3. Extension and simplification of CPT.
doi:10.1371/journal.pone.0056832.g003

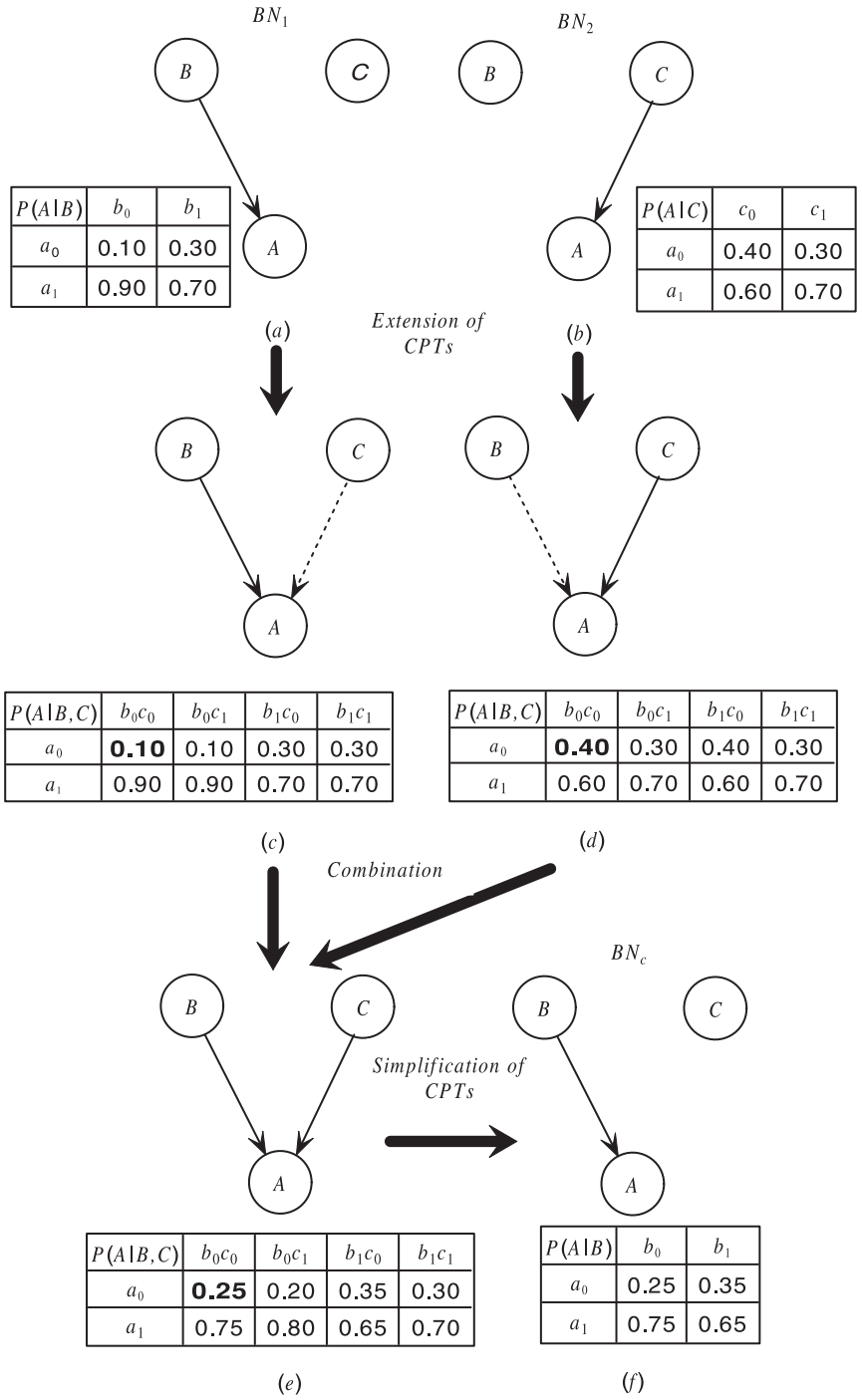

Figure 4. An example to demonstrate the combination of two Bayesian networks. Assume that weights $W_1 = W_2$, $e = 0.001$, and we have two Bayesian networks as shown in (a) and (b). The CPTs of A in the two Bayesian networks do not have a same form, so they need to be extended. After extending the CPTs, the two Bayesian networks' structures and every corresponding CPTs' forms are completely the same(as shown in (c) and (d), the dashed edges represent bogus edges), and then the aggregation function can be applied to aggregate the conditional probabilities in corresponding positions of each corresponding CPTs. For example, $P(A = a_0|B = b_0, C = c_0) = (0.10 + 0.40)/2 = 0.25$. In the combined Bayesian network as shown in (e), we need to use variance to test A's two parent nodes, $D_B = 0.0025 > e$, $D_C = 0.000625 < e$, so C is a bogus parent. Then the CPT of A need to be simplified and the bogus edge $C \rightarrow A$ should be deleted. The consensus Bayesian network is shown in (f).
doi:10.1371/journal.pone.0056832.g004

In Bayesian network parameter learning, the following equation is used to compute the conditional probabilities in each node's CPT:

$$P(A=a_i|Pa(A)=p_k)=\frac{N(A=a_i,Pa(A)=p_k)}{N(Pa(A)=p_k)} \quad (4)$$

where $N(Conditions)$ is the number of samples satisfies *Conditions* in the database.

Extension and simplification of CPT

Theorem 1. Given variables A and B, then $P(A|B)=P(A)$ $(P(B)\neq 0)$ holds if A and B are independent.

Corollary 1. Given A, B and any other variable C, then $P(A|B,C)=P(A|C)$ $(P(B|C)\neq 0)$ holds if A and B are independent given C.

Suppose we have the CPT of node A as shown in Figure 3(a), it can be extended into the form as shown in Figure 3(b) if A and B are independent of each other. Since A and B are independent, B can not affect the distribution of A, then for $\forall b_j\in B$, $P(A=a_i|B=b_j)=P(A=a_i)$ holds. According to that, two CPTs of a same node in different Bayesian networks can be extended into a same form, and then can be aggregated even if the node does not have a same parent set in these Bayesian networks. Specifically, for a node A, and its parent sets are $Pa_1(A)$ and $Pa_2(A)$ $(Pa_2(A)\neq Pa_1(A))$ in BN_1 and BN_2, respectively. Then the two CPTs of A do not have a same form, and the aggregation function can't be applied (See the CPTs of A shown in Figure 3(b) and Figure 3(c), they have a same form, then the aggregation function can be applied to aggregate the conditional probabilities in the corresponding position of the two CPTs, and the aggregation function can't be applied to aggregate the CPTs shown in Figure 3(a) and Figure 3(c)). However, we can take $Pa(A)=Pa_1(A)\cup Pa_2(A)$ as the parent set and extend both the CPTs of A in BN_1 and BN_2 into form $P(A|Pa(A))$, then the two CPTs of A have a same form and the aggregation function can be applied. This means we also view the nodes in $Pa_2(A)-Pa_1(A)$ as the parents of A in BN_1, although they are not real parents and do not affect A's conditional probability. We call these parents bogus parents and the directed edges between a node and its bogus parents bogus edges. As shown in Figure 4(c), C is a bogus parent of A, and $C\rightarrow A$ is a bogus edge.

Theorem 2. Given variables A and B, A is independent of B if the conditional probability of A does not change when B takes different values.

Proof. Assume that the number of expression values of B is n and for $\forall b_j\in B$, $P(B=b_j)\neq 0$. Then for $\forall a_i\in A$, we have:

$$\begin{aligned}
&P(A=a_i|B=b_1)=\cdots=P(A=a_i|B=b_n)\\
\Leftrightarrow\quad &\frac{P(A=a_i,B=b_1)}{P(B=b_1)}=\cdots=\frac{P(A=a_i,B=b_n)}{P(B=b_n)}\\
\Leftrightarrow\quad &\frac{P(A=a_i,B=b_j)}{P(B=b_j)}=\frac{\sum_{j=1}^{n}P(A=a_i,B=b_j)}{\sum_{j=1}^{n}P(B=b_j)}\\
\Leftrightarrow\quad &\frac{P(A=a_i,B=b_j)}{P(B=b_j)}=\frac{P(A=a_i)}{1}\\
\Leftrightarrow\quad &P(A=a_i|B=b_j)=P(A=a_i)
\end{aligned}$$

So, $P(A|B)=P(A)$, then A and B are independent.

End of the proof.

Corollary 2. Given A, B and any other variable C, A is independent of B given C if the conditional probability of A does not change when B takes different values (only B changes).

Theorem 2 and Corollary 2 can be used to determine whether two nodes are independent of each other or not. Suppose we have the CPT of node A as shown in Figure 3(c), if $x_{00}=x_{10}$, $x_{01}=x_{11}$ or they are approximately equal, it deduces that A and B are independent, and B is not the parent node of A. Then the CPT of node A can be simplified into the form as shown in Figure 3(d). Conditional probabilities in the CPT of A are discrete values, then variance can be used to determine whether the conditional probability of A changes or not when B takes different values. Assume that A's parent set is $\{B,OtherParents\}$. First, compute each variance $D_{a_ip_k}$ of the conditional probabilities satisfy $A=a_i$, $OtherParents=p_k$ and B takes different values. Second, compute the average variance D_B of all $D_{a_ip_k}$ when A and $OtherParents$ take different values. Given a threshold e $(e>0)$, if $D_B<e$, it means the conditional probability of A almost does not change when B takes different values, then A and B are independent and B is not the parent node of A. In the combination algorithm, CPTs were extended previously, then some nodes may have bogus parents after the aggregation of CPTs. However, we can use this method to find them, and then simplify the CPTs and delete the bogus edges. Threshold e can be selected by using domain knowledge. Specifically, we have $D_{X_j}=D_k>e$ if variables X_i and X_j are related, and $D_{X_j'}=D_k'<e$ if variables X_i' and X_j' are independent. And then we have $\{D_k>e|k=1,2,\cdots,n\}$ if there is n pair of variables related, and $\{D_k'<e|k=1,2,\cdots,m\}$ if there is m pair of variables independent each other. So we have $min\{D_1,D_2,D_3,\cdots,D_n\}>e>max\{D_1',D_2',D_3',\cdots,D_m'\}$.

Aggregation function

Assume that the conditional probability of node A in Bayesian networks BN_1 and BN_2 are $P_1(A=a_i|Pa(A)=p_k)=x_1$ and $P_2(A=a_i|Pa(A)=p_k)=x_2$, respectively. Then in the consensus Bayesian network, the conditional probability of A can be computed using the following equation:

$$P(A=a_i|Pa(A)=p_k)=\frac{W_1*x_1+W_2*x_2}{W_1+W_2} \quad (5)$$

where W_1 is the weight of BN_1 and W_2 is the weight of BN_2. Weight W is a positive integer representing a belief to the Bayesian network. $W_1>W_2$ means BN_1 is more reliable than BN_2; $W_1\rightarrow\infty$ means BN_1 is absolutely reliable.

Next, we would like to discuss why we choose this aggregation function. The combination of Bayesian networks must satisfies this property: the consensus Bayesian network BN_c constructed by combining Bayesian networks BN_1 and BN_2 is equivalent to the Bayesian network learned from the database obtained by merging the two Bayesian networks' corresponding databases DB_1 and DB_2. Then the aggregation function should satisfy it too. Assume that node X is not the parent of A in any Bayesian network, then in the consensus Bayesian network, X can not be the parent of A. Then CPTs of A in different Bayesian networks after extension not only have a same form, but also contains all of A's possible parent nodes. So, we needn't consider the nodes which are not included in the parent set of A when aggregating the CPTs. When computing the conditional probability of one node in Bayesian network, Equation (4) is used. The conditional probability of A in

Bayesian networks BN_1 and BN_2 are $P_1(A=a_i|Pa(A)=p_k)=x_1$ and $P_2(A=a_i|Pa(A)=p_k)=x_2$, respectively. Assume that the number of samples satisfy $Pa(A)=p_k$ in DB_1 is n_1, then the number of samples satisfy $A=a_i$ and $Pa(A)=p_k$ in DB_1 is n_1*x_1; the number of samples satisfy $Pa(A)=p_k$ in DB_2 is n_2, then the number of samples satisfy $A=a_i$ and $Pa(A)=p_k$ in DB_2 is n_2*x_2. So, the conditional probability of A in the Bayesian network learned from the database obtained by merging DB_1 and DB_2 is:

$$P(A=a_i|Pa(A)=p_k)=\frac{n_1*x_1+n_2*x_2}{n_1+n_2} \tag{6}$$

On the other hand, samples satisfy $Pa(A)=p_k$ in DB_1 and in DB_2 obey the same distribution. So, we have:

$$\frac{n_1}{N_1}\approx\frac{n_2}{N_2} \tag{7}$$

where N_1 and N_2 are the total numbers of samples in DB_1 and DB_2, respectively. Then the conditional probability of A changed to be:

$$P(A=a_i|Pa(A)=p_k)=\frac{N_1*x_1+N_2*x_2}{N_1+N_2} \tag{8}$$

Total numbers of samples in databases are unable to be known sometimes, so we use the weights of the Bayesian networks instead of them, then Equation (8) changed into Equation (5). In the experiment, we still use the total numbers of samples as they are already known.

Combination of Bayesian networks

If two Bayesian networks are defined over the same variable set and their variables' prior orders are consistent with each other, then they can be combined using the method described as follow:

Step 1. Extend every corresponding CPTs in the two Bayesian networks into same form. Then the structures of the two Bayesian networks are completely the same(although some of their edges are bogus edges).

Step 2. Use the aggregation function to aggregate the conditional probabilities in the corresponding positions of each corresponding CPTs.

Step 3. In each CPT after aggregation, compute variance D_X for each parent node X, determine whether $D_X<e$ holds or not to judge node X is a bogus parent or not, then simplify the CPT and delete the bogus edge if $D_X<e$ holds.

After simplifying the CPTs and deleting the bogus edges, the consensus Bayesian network is obtained. Figure 4 shows an example of combination of two Bayesian networks. However, Bayesian networks' variables' prior orders do not always consistent with each other, then it needs to reverse some directed edges sometimes. The principle of reversal is to ensure that the Bayesian network after reversal is equivalent to the original Bayesian network.

Extension of Bayesian network

Sometimes the Bayesian networks going to be combined may not defined over the same variable set, then they need to be extended. Specifically, given two Bayesian networks BN_1 and BN_2 with their variable sets satisfy $V_1\neq V_2$ and $V_1\cap V_2\neq\varnothing$, if their variables' prior orders are consistent with each other, BN_1 can be extended into $BN_1\oplus BN_2$ using the method described as follow:

Step 1. Extend BN_1's DAG G_1 into $G_1\oplus G_2$. Let $G_1\oplus G_2=(V_1\cup V_2,E_1)$, and then add all the directed edges satisfy

$$\{(C,A)\in E_2|C,A\in V_2-V_1 \text{ or } C\in V_2-V_1,A\in V_1 \text{ or } C\in V_1,A\in V_2-V_1\}$$

into graph $G_1\oplus G_2$. These added edges are not in BN_1 originally, so we call them extended edges.

Step 2. Compute each node's CPT. For a node $A\in G_1\oplus G_2$, if $A\in V_2-V_1$, then its CPT is the same as the CPT of A in BN_2; if $A\in V_1$ and there is no directed edge satisfies $\{(C,A)\in E_2|C\in V_2-V_1\}$, then its CPT is the same as the CPT of A in BN_1; if $A\in V_1$ and has directed edges satisfy $\{(C,A)\in E_2|C\in V_2-V_1\}$, in this case, there are three possible situations may appeared in the extended Bayesian network as shown in Figure 5. Then the conditional probabilities of A in these three situations can be computed using the following equations, respectively:

In Figure 5(a)

$$P(A=a_i|B=b_j,C=c_k)=\frac{P_1(A=a_i|B=b_j)*\frac{P_2(C=c_k|A=a_i,B=b_j)}{P_2(C=c_k|B=b_j)}}{\sum_{i=1}^{m}P_1(A=a_i|B=b_j)*\frac{P_2(C=c_k|A=a_i,B=b_j)}{P_2(C=c_k|B=b_j)}} \tag{9}$$

where m is the number of expression values of A. $P_2(C=c_k|A=a_i,B=b_j)$ and $P_2(C=c_k|B=b_j)$ can be computed using the standard Bayesian network inference algorithm [25] in BN_2, while $P_1(A=a_i|B=b_j)$ is already known in BN_1.

In Figure 5(b)

$$P(A=a_i|B=b_j,C=c_k)=\frac{P_1(A=a_i|B=b_j)*\frac{P_2(C=c_k|A=a_i)}{P_2(C=c_k)}}{\sum_{i=1}^{m}P_1(A=a_i|B=b_j)*\frac{P_2(C=c_k|A=a_i)}{P_2(C=c_k)}} \tag{10}$$

In Figure 5(c)

$$P(A=a_i|C=c_k)=\frac{P_1(A=a_i)*\frac{P_2(C=c_k|A=a_i)}{P_2(C=c_k)}}{\sum_{i=1}^{m}P_1(A=a_i)*\frac{P_2(C=c_k|A=a_i)}{P_2(C=c_k)}} \tag{11}$$

If B and A are disconnect in BN_2, it can only deduce that B and A are independent given C, however, it doesn't affect the conditional probabilities $P_2(C=c_k|B=b_j,A=a_i)$ and $P_2(C=c_k|B=b_j)$, then the conditional probability of A can be computed using Equation (9). If both B and C, B and A are disconnect in BN_2, it deduces B and C are independent, then the conditional probability of A can be computed using Equation (10). After obtaining every node's CPT in $G_1\oplus G_2$, the extension of Bayesian network BN_1 is finished.

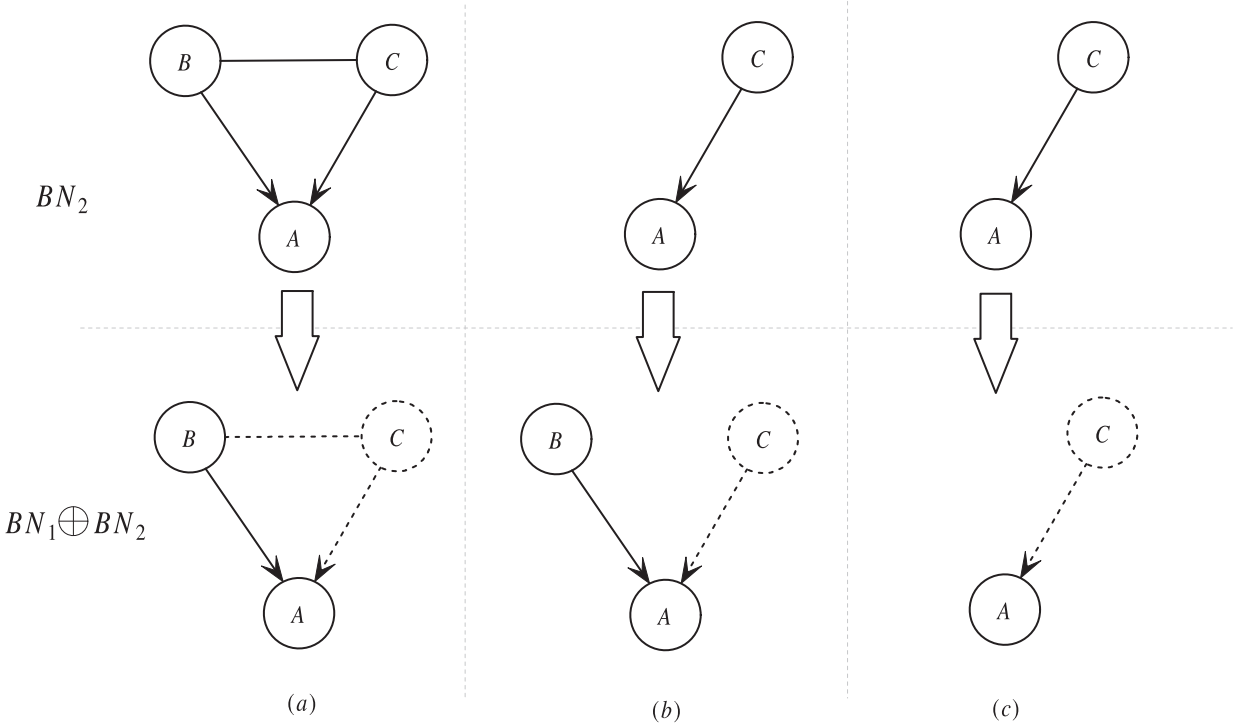

Figure 5. Three possible situations in the extended Bayesian network $BN_1 \oplus BN_2$. Where $A, B \in V_1$, $C \in V_2 - V_1$, B and C may be two nodes or two node sets with each node in them has a directed edge point to A. In $BN_1 \oplus BN_2$, solid lines represent the edges in BN_1 originally, and dashed lines represent the extended edges. The undirected edge $B - C$ represents one of these three cases: (1) directed edge $B \rightarrow C$; (2) directed edge $B \leftarrow C$; (3) B and C is disconnect.
doi:10.1371/journal.pone.0056832.g005

After extending BN_1 into $BN_1 \oplus BN_2$ and BN_2 into $BN_2 \oplus BN_1$, $BN_1 \oplus BN_2$ and $BN_2 \oplus BN_1$ are defined over the same variable set $V_1 \cup V_2$, and then they can be combined using the combination algorithm.

Author Contributions

Conceived and designed the experiments: LDH LMW. Performed the experiments: LDH LMW. Analyzed the data: LDH LMW. Wrote the paper: LDH.

References

1. Ramotar D, Popoff SC, Gralla EB, Demple B (1991) Cellular role of yeast apn1 apurinic endonuclease/3′-diesterase: repair of oxidative and alkylation dna damage and control of spontaneous mutation. Molecular and Cellular Biology 11: 4537–4544.
2. Demple B, Harrison L (1994) Repair of oxidative damage to dna: enzymology and biology. Annual Review of Biochemistry 63: 915–948.
3. Bohr VA, Dianov GL (1999) Oxidative dna damage processing in nuclear and mitochondrial dna. Biochimie 81: 155–160.
4. Friedman N, Linial M, Nachman I, Pe'er D (2000) Using bayesian networks to analyze expression data. Journal of Computational Biology 7: 601–620.
5. Irene M, Jeremy D, David P (2002) Modelling reguratory pathways in e. coli from time series expression profiles. Bioinformatics 18: S241–S248.
6. Beal M, Falciani F, Ghahramani Z, Rangel C, Wild D (2005) A bayesian approach to reconstructing genetic regulatory networks with hidden factors. Bioinformatics 21: 349–356.
7. Bansal M, Belcastro V, Ambesi-Impiombato A, di Bernardo D (2007) How to infer gene networks from expression profiles. Molecular Systems Biology 3: 78.
8. Pearl J (1986) Fusion, propagation and structuring in belief networks. Artificial Intelligence 29: 241–288.
9. Jensen F (2001) Bayesian networks and decision graphs. New York: Springer.
10. Cooper GF, Herskovits E (1992) A bayesian method for the induction of probabilistic networks from data. Machine Learning 9: 309–347.
11. Cheng J, Greiner R, Kelly J, Bell D, Liu WR (2002) Learning bayesian networks from data: an information-theory based approach. Artificial Intelligence 137: 43–90.
12. Tsamardinos I, Brown LE, Aliferis CF (2006) The max-min hill-climbing bayesian network structure learning algorithm. Machine Learning 65: 31–78.
13. Cano A, Masegosa A, Moral S (2011) A method for integrating expert knowledge when learning bayesian networks from data. IEEE transactions on systems, man, and cybernetics 41: 1382–1394.
14. Wong SM, Butz CJ (2001) Constructing the dependency structure of a multiagent probabilistic network. IEEE Transactions on Knowledge and Data Engineering 13: 395–415.
15. Yang ZQ, Wright RN (2006) Privacy-preserving computation of bayesian networks on vertically partitioned data. IEEE Transactions on Knowledge and Data Engineering 18: 1253–1264.
16. Pavlin G, de Oude P, Maris M, Nunnink J, Hood T (2010) A multi-agent systems approach to distributed bayesian information fusion. Infromation Fusion 11: 267–282.
17. Sagrado J, Moral S (2003) Qualitative combination of bayesian networks. International Journal of Intelligent Systems 18: 237–249.
18. Utz CM (2010) Learning ensembles of bayesian network structures using random forest techniques. Master Thesis of University of Oklahoma.
19. Zhang Y, Yue K, Yue M, Liu W (2011) An approach for fusing bayesian networks. Journal of Information and Computational Science 8: 194–201.
20. Hodges AP, Dai DJ, Xiang ZS, Woolf P, Xi CW, et al. (2010) Bayesian network expansion identifies new ros and biofilm regulators. PLoS ONE 5: e9513.
21. Faith JJ, Driscoll ME, Fusaro VA, Cosgrove EJ, Hayete B, et al. (2008) Many microbe microarrays database: uniformly normalized affymetrix compendia with structured experimental metadata. Nucleic Acids Research 36: D866–D870.
22. Keseler IM, Collado VJ, Gama CS, Ingraham J, Paley S (2005) Ecocyc: a comprehensive database resource for escherichia coli. Nucleic Acids Research 33: D334–D337.
23. Chow CK, Liu CN (1968) Approximating discrete probability distributions with dependence trees. IEEE Transactions on Information Theory 14: 462–467.
24. Lam W, Bacchus F (1994) Learning bayesian belief networks: an approach based on the mdl principle. Computational Intelligence 10: 269–293.
25. Zhang LW, Guo HP (2006) Introduction to Bayesian network. Peking: Science Press.