13th International Conference on Applications of Statistics and Probability in Civil Engineering, ICASP13
Seoul, South Korea, May 26-30, 2019

Algorithms for Complex Systems Reliability Analysis Based on Bayesian Network

Xiaohu Zheng
Graduate Student, College of Aerospace Science and Engineering, National University of Defense Technology, Changsha, China

Wen Yao
Professor, National Innovation Institute of Defense Technology, Chinese Academy of Military Science, Beijing, China

Yingchun Xu
Graduate Student, College of Aerospace Science and Engineering, National University of Defense Technology, Changsha, China

Yong Zhao
Professor, College of Aerospace Science and Engineering, National University of Defense Technology, Changsha, China

ABSTRACT: With the increase of complex systems' functions, the number of its components will rise. This will lead to the amount of state combinations of components increasing exponentially. In order to solve this problem, a new compression algorithm and a new inference algorithm are developed to analyze the reliability of complex systems based on Bayesian Network in this paper. A satellite transmission system reliability is used to validate the proposed algorithms.

KEY WORDS: Bayesian Network, Compression, Reliability, complex systems

1. INTRODUCTION

Reliability analysis is one of the most important parts of system safety. Once some systems, such as aerospace systems, power systems, are failed, it will cause serious economic losses and affect people's normal life. With the increasing complexity of modern engineering systems, reliability analyses of systems are becoming more challenging. Therefore, it is essential for analyzing the reliability of systems by a suitable method.

The Bayesian Network (BN) (1988) is a useful probabilistic tool for system reliability analysis. There are some researches about reliability analysis based on BN. In the reliability model of BN, Sturlaugson and Sheppard (2015) studied the method of analyzing sensitivity in continuous time BN, which considers continuous time BN as a decomposition Markov process that decomposes the system into interrelated subsystems. Su and Fu (2014) proposed a causal logic method based on BN reliability model for analyzing the reliability of wind turbine considering environmental factors and uncertainties, and using the expected adjustment method to achieve the final quantitative analysis. Considering the effect of common cause failure to system reliability and the widespread presence of multistate systems in engineering practices, Mi et al. (2016) proposed a method for multistate systems reliability analysis by taking the advantage of graphic representation and uncertainty reasoning of BN. Besides, Mi et al. (2018) proposed a multi-state evidential networks method for reliability analysis of complex system with common cause failure groups based on evidence theory and BN.

For complex systems, with the increase of satellite system functions, the number of its components will rise. This will lead to the amount of state combinations of components increasing exponentially. For the above problems, Tien and Kiureghian (2016) proposed compression algorithm and inference algorithm for reliability assessment of infrastructure systems by compressing system column of node probability table. For the compression algorithm, it integrates two encoding compression technique, i.e. run-length encoding compression technique (Hauck 1986) and Lempel-Ziv encoding compression technique (Ziv and Lempel 1977). By compressing system column of the NPT, the compression algorithm can reduce the memory storage requirements of system node's NPT. After that, the inference algorithm can perform inference based on the compressed NPT. Besides, to construct the BN model of multistate systems, Tong and Tien (2017) proposed new compression algorithm which compressed the NPT to be several bundles representing repeated patterns of fixed length in the system state of NPT. In Tong and Tien's algorithms, the number of states of components and system must be equal. For example, if the studied system has five states, its subsystems or components will also have five states. However, with the complexity of logical relationship between system and its components, the existing compression algorithms cannot completely compress the NPT.

In this paper, in order to completely compress the NPT, a new compression algorithm is developed based on the researches of Tien and Kiureghian (2016). Based on the compressed NPT, a new inference algorithm is developed for performing the inference of BN model at the same time. The rest of this paper is structured as follows. In section 2, several variables are defined in this section, wherein the format of node probability table is also presented. Then the new compression algorithm is developed. In section 3, the new inference algorithm is developed by variable elimination algorithm, and the rules of constructing intermediate factor are also developed in this section. Finally, a numerical example is studied to validate the proposed algorithms in section 4.

2. COMPRESSION ALGORITHM

2.1. Format of NPT

NPT reflects the relationship between child node and parent nodes. The state of each binary parent node in NPT's each row is calculated by equation (1) proposed by Tien and Kiureghian (2016).

$$s_i^k = \begin{cases} 0 \quad \text{if } ceil\left(\dfrac{k}{2^{n-i}}\right) \in \text{odd} \\ 1 \quad \text{if } ceil\left(\dfrac{k}{2^{n-i}}\right) \in \text{even} \end{cases} \tag{1}$$

In equation (1), k is the row number of NPT and n is the binary parent nodes number of binary child node. $ceil(x)$ is the function that calculated the value of x rounded up to the nearest integer. s_i^k means the state of the ith binary parent node in the kth row of NPT. After each row's state combination of binary parent nodes is decided, the conditional probability of child node can be gotten by the logical relationship between child node and its parent nodes. For example, a binary child node S has two parallel binary parent nodes C_1 and C_2. According to equation (1), the state combination of two parent nodes is shown as in Table 1. For the first row of Table 1, because of both the state of two parallel parent nodes is 0, the state of S is 0. Therefore, the conditional probability $P(S=0 \mid C_1, C_2)=1$ and $P(S=1 \mid C_1, C_2)=0$ as shown in Table 1.

Table 1: NPT of child node S

C_1	C_2	$P(S \mid C_1, C_2)$	
		$S=0$	$S=1$
0	0	1	0
0	1	0	1
1	0	0	1
1	1	0	1

2.2. Compression algorithm

Supposed that a binary child node S has n binary parent nodes $\{C_i \mid i=1,2,\cdots,n\}$. Therefore, the NPT's conditional probability column of child node S is $P(S|C_1,C_2,\cdots,C_n)$. The new compression algorithm is developed to compress an object sequence, denoted as OS, which can be any one of the two conditional probability columns, i.e. $S=1$ or $S=0$.

2.2.1. Run and phrase

According to Tien and Kiureghian (2016), the definition of run and phrase are shown as follow respectively.

- Run is a sequence like $00\cdots0$ or $11\cdots1$, which is composed by a same numerical value.
- Phrase is a sequence like $01\cdots1$ or $10\cdots0$, which the second numerical value is different from the first numerical value but is the same with the numerical value from the third to the last. Particularly, the length of a phrase is greater than or equal to 2.

The compressed objective sequence, denoted as cOS, has m rows which is composed of runs and phrases. cOS has a run accompany dictionary and a phrase accompany dictionary, denoted as d_r and d_p, respectively. For d_r, the q^j th row is $\{q^j,r^j,L_r^j\}$, where q^j is the row number, r^j is the numerical value that composes the run in the q^j th row of d_r, and L_r^j is the length of the run in the q^j th row of d_r. For d_p, the p^j th row is $\{p^j,v_1^j,v_2^j,L_p^j\}$, where p^j is the row number, v_1^j and v_2^j are respectively the first and the second numerical value that compose the phrase in the p^j th row of d_p, and L_p^j is the length of the phrase in the p^j th row of d_p. For the j th row of cOS, if this row is a run, it is $\{run,q^j,n_r^j\}$, where $j=1,2,\cdots,m$, n_r^j is the repeated numbers of this run in OS. Conversely, if the j th row is a phrase, this row is $\{phrase,p^j,n_p^j\}$, where n_p^j is the repeated numbers of this phrase in OS. For the j th row's run or phrase of cOS, RP_j is the set of row start numbers that are row numbers of the repeated runs' or phrases' first numerical values in OS.

2.2.2. New compression algorithm

To solve the problem of state combinations of components increasing exponentially, the new compression algorithm is developed to compress the NPT of child nodes. The flowchart of new compression algorithm is shown in Figure 1.

In Figure 1, the new compression algorithm has five outputs, i.e. cOS, d_r, d_p, RP and S^{all}, where RP is the set of RP_j and S^{all} is the set of all the start row numbers of runs and phrases. $rem(x,y)$ is the function that calculates the remainder of x/y. S^{now} is the current query run's or phrase's row start number. S^{exist} is the row start number of the existed run or phrase that is the same as the current query run or phrase. $OS_{S^{now}-1}$ and $OS_{S^{exist}-1}$ are the numerical value of the $(S^{now}-1)$ th row and the $(S^{exist}-1)$ th row of OS, respectively.

In the compression process, the parent nodes' state combination of NPT's each row is calculated by equation (1) at first. Then, the value of $P(S=1|C_1,C_2,\cdots,C_n)$ can be gotten by according to the logical relationship between child node and its parent nodes. After that, querying the next value of $P(S=1|C_1,C_2,\cdots,C_n)$ that corresponds to the parent nodes' state combination of NPT's next row. In the process of querying each row value of the A column of the NPT, it is judged whether the sequence of the current query is run or phrase. The judgment process is shown in the following.

Firstly, if the value of the $(k+1)$ th row of $P(S=1|C_1,C_2,\cdots,C_n)$ column is different from the value of the k th row, it means that the k th row is the start row of a phrase. Let S^{now} equal to k and k is stored in the set S^{all}. Then, continuing to query the next row until a phrase is identified. Once the first phrase is identified, the phrase accompany dictionary d_p is constructed. According to the filter condition in Figure 1, it is judged whether the current querying phrase is a new phrase or already exists in d_p. If the current querying phrase is a new phrase, it will be added into cOS and d_p in the form of $\{phrase, p^j, n_p^j\}$ and $\{p^j, v_1^j, v_2^j, L_p^j\}$, respectively. If the current querying phrase already exists in d_p, the value of n_p^j, which corresponds to already existed phrase, is updated only.

Secondly, if the value of the $(k+1)$th row of $P(S=1|C_1,C_2,\cdots,C_n)$ column is the same as the value of the kth row, it means that the kth row is the start row of a run. Let S^{now} equal to k and k is stored in the set S^{all}. Then, continuing to query the next row until a run is identified. Once the first run is identified, the run accompany dictionary d_r is constructed. According to the filter condition in Figure 1, it is judged whether the current querying run is a new run or already exists in d_r. If the current querying run is a new run, it will be added into cOS and d_r in the form of $\{run, q^j, n_r^j\}$ and $\{q^j, r^j, L_r^j\}$, respectively. If the current querying run already exists in d_r, the value of n_r^j, which corresponds to already existed run, is updated only.

Finally, after the current querying run or phrase is finished, repeating the above process until the last row of OS.

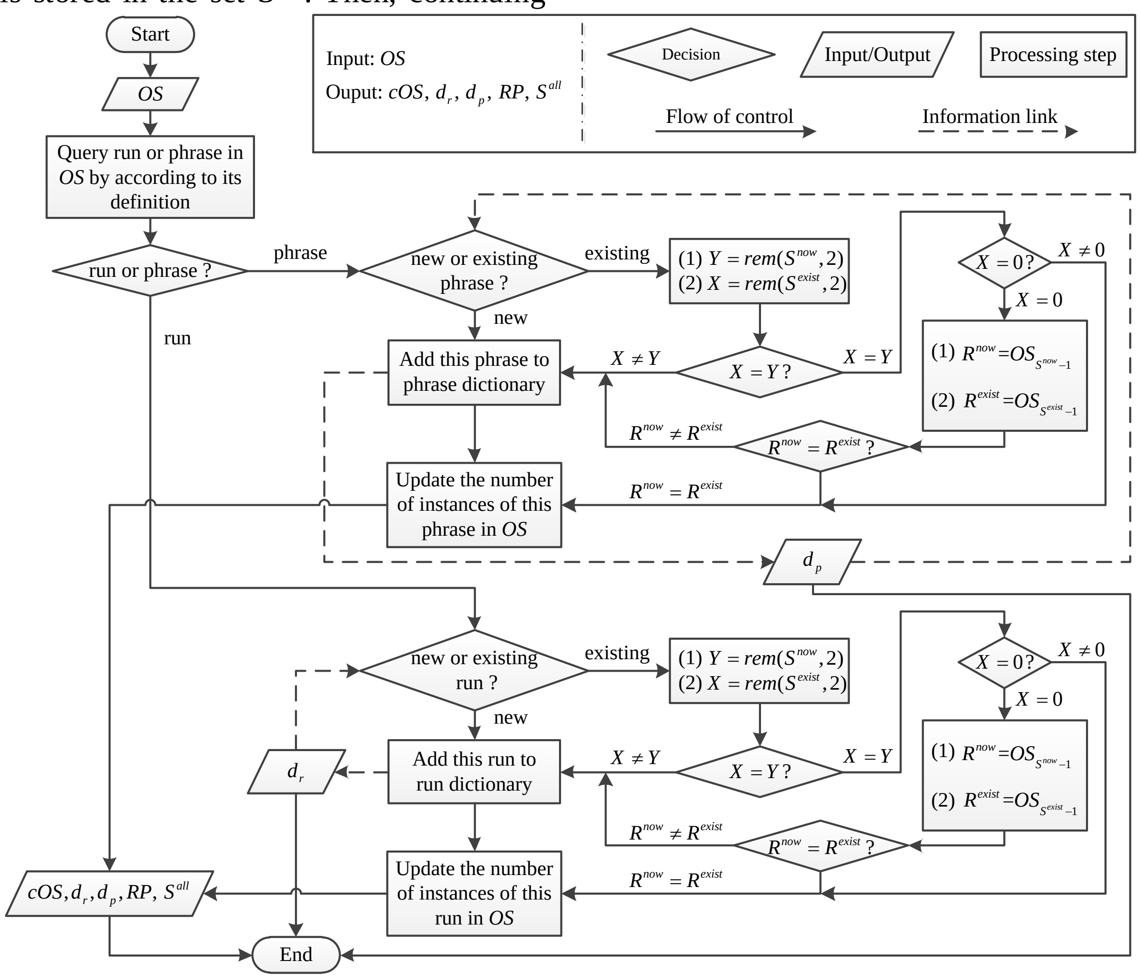

Figure 1: Flowchart of new compression algorithm

3. INFERENCE ALGORITHM

The inference algorithm is developed by variable elimination algorithm (VE) (1998). For example, for the node S in Table 1, $\Pr(S \mid C_1)$ is calculated as shown in equation (2), where λ_2 is the intermediate factor after eliminating node C_2.

$$\begin{aligned}\Pr(S \mid \mathrm{C}_1) &= \sum_{C_2} \Pr(C_2)\Pr(S \mid C_1, C_2) \\ &= \Pr(S \mid C_1) \\ &= \lambda_2\end{aligned} \tag{2}$$

3.1. Rules for constructing intermediate factor

As shown in Table 2, Table 3 and Table 4, rules for constructing intermediate factor $c\lambda_i^j$ is developed for the inference of BN based on the compressed NPT.

Table 2: Rules for constructing $c\lambda_i^j$ starting in row j of $c\lambda_{i+1}$

switch	*conditions*			J_{i+1}^j *and* I	r_i^j *or* p_i^j	$L_{r_i}^j$ *or* $n_{p_i}^j$	R_{i+1}^j	R^{all}
run	$S_{i+1}^j \in \text{odd}$	$L_{r_{i+1}}^j \in \text{odd}$		$J_{i+1}^j = ismb(RP_{i+1}^j, S_{i+1}^{all})$ $I = J_{i+1}^j(1)$	$q_i^j = q_i^{j-1} + 1$	$n_{r_{i+1}}^j$	$r_{i+1}^j \times \Pr(C_i = 0)$	$L_{RP} = length(RP_{i+1}^j)$ $\begin{cases}\text{for } i_{RP} = 1 : L_{RP}, \text{do} \\ \quad R_{J_{i+1}^j(1, i_{RP})}^{all} = R_{i+1}^j \\ \text{end}\end{cases}$
		$L_{r_{i+1}}^j \in \text{even}$			$q_i^j = q_i^{j-1} + 1$	$n_{r_{i+1}}^j$	*0*	
	$S_{i+1}^j \in \text{even}$	$L_{r_{i+1}}^j \in \text{odd}$	$L_{r_{i+1}}^j = 1$		$q_i^j = q_i^{j-1} + 1$	$n_{r_{i+1}}^j$	*0*	
			$L_{r_{i+1}}^j > 1$		$q_i^j = q_i^{j-1} + 1$ (*also* $q_i^{j+1} = q_i^{j-1} + 2$)	$n_{r_i}^j = n_{r_{i+1}}^j$ (*also* $n_{r_i}^{j+1} = n_{r_{i+1}}^j$)	*0*	
		$L_{r_{i+1}}^j \in \text{even}$	$L_{r_{i+1}}^j = 2$		$q_i^j = q_i^{j-1} + 1$	$n_{r_{i+1}}^j$	$r_{i+1}^j \times \Pr(C_i = 0)$	
			$L_{r_{i+1}}^j > 2$		$q_i^j = q_i^{j-1} + 1$ (*also* $q_i^{j+1} = q_i^{j-1} + 2$)	$n_{r_i}^j = n_{r_{i+1}}^j$ (*also* $n_{r_i}^{j+1} = n_{r_{i+1}}^j$)	$r_{i+1}^j \times \Pr(C_i = 0)$	
phrase	$S_{i+1}^j \in \text{odd}$	$L_{p_{i+1}}^j \in \text{odd}$	$L_{p_{i+1}}^j = 3$		p_{i+1}^j	$n_{p_{i+1}}^j$	$v_{2_{i+1}}^j \times \Pr(C_i = 0)$	
			$L_{p_{i+1}}^j > 3$		p_{i+1}^j	$n_{p_{i+1}}^j$	$v_{2_{i+1}}^j \times \Pr(C_i = 0)$	
		$L_{p_{i+1}}^j \in \text{even}$	$L_{p_{i+1}}^j = 2$		p_{i+1}^j	$n_{p_{i+1}}^j$	*0*	
			$L_{p_{i+1}}^j > 2$		p_{i+1}^j	$n_{p_{i+1}}^j$	*0*	
	$S_{i+1}^j \in \text{even}$	$L_{p_{i+1}}^j \in \text{odd}$			p_{i+1}^j	$n_{p_{i+1}}^j$	*0*	
		$L_{p_{i+1}}^j \in \text{even}$	$L_{p_{i+1}}^j = 2$		p_{i+1}^j	$n_{p_{i+1}}^j$	$v_{2_{i+1}}^j \times \Pr(C_i = 0)$	
			$L_{p_{i+1}}^j > 2$		p_{i+1}^j	$n_{p_{i+1}}^j$	$v_{2_{i+1}}^j \times \Pr(C_i = 0)$	

Table 3: Rules for constructing $d_{r_i}^j$ starting in row j of $c\lambda_{i+1}$

switch	*conditions*			r_i^j	$L_{r_i}^j$
run	$S_{i+1}^j \in \text{odd}$	$L_{r_{i+1}}^j \in \text{odd}$		r_{i+1}^j	$(L_{r_{i+1}}^j - 1)/2$
		$L_{r_{i+1}}^j \in \text{even}$		r_{i+1}^j	$L_{r_{i+1}}^j/2$
	$S_{i+1}^j \in \text{even}$	$L_{r_{i+1}}^j \in \text{odd}$	$L_{r_{i+1}}^j = 1$	$r_{i+1}^j \times \Pr(C_i = 1) + R_{I-1}^{all}$	*1*
			$L_{r_{i+1}}^j > 1$	$r_{i+1}^j \times \Pr(C_i = 1) + R_{I-1}^{all}$ (*also* $r_i^{j+1} = r_{i+1}^j$)	*1* (*also* $L_{r_i}^{j+1} = (L_{r_{i+1}}^j - 1)/2$)
		$L_{r_{i+1}}^j \in \text{even}$	$L_{r_{i+1}}^j = 2$	$r_{i+1}^j \times \Pr(C_i = 1) + R_{I-1}^{all}$	*1*
			$L_{r_{i+1}}^j > 2$	$r_{i+1}^j \times \Pr(C_i = 1) + R_{I-1}^{all}$ (*also* $r_i^{j+1} = r_{i+1}^j$)	*1* (*also* $L_{r_i}^{j+1} = (L_{r_{i+1}}^j - 2)/2$)

Table 4: Rules for constructing $d_{p_i}^j$ starting in row j of $c\lambda_{i+1}$

switch	condition			$v_{1_i}^j$	$v_{2_i}^j$	$L_{p_i}^j$
phrase	$S_{i+1}^j \in \text{odd}$	$L_{p_{i+1}}^j \in \text{odd}$	$L_{p_{i+1}}^j = 3$	$[v_{1_{i+1}}^j \times \Pr(C_i = 0)] + [v_{2_{i+1}}^j \times \Pr(C_i = 1)]$	-3	1
			$L_{p_{i+1}}^j > 3$	$[v_{1_{i+1}}^j \times \Pr(C_i = 0)] + [v_{2_{i+1}}^j \times \Pr(C_i = 1)]$	$v_{2_{i+1}}^j$	$[(L_{p_{i+1}}^j - 3)/2] + 1$
		$L_{p_{i+1}}^j \in \text{even}$	$L_{p_{i+1}}^j = 2$	$[v_{1_{i+1}}^j \times \Pr(C_i = 0)] + [v_{2_{i+1}}^j \times \Pr(C_i = 1)]$	-3	1
			$L_{p_{i+1}}^j > 2$	$[v_{1_{i+1}}^j \times \Pr(C_i = 0)] + [v_{2_{i+1}}^j \times \Pr(C_i = 1)]$	$v_{2_{i+1}}^j$	$[(L_{p_{i+1}}^j - 2)/2] + 1$
	$S_{i+1}^j \in \text{even}$	$L_{p_{i+1}}^j \in \text{odd}$		$R_{I-1}^{all} + [v_{1_{i+1}}^j \times \Pr(C_i = 1)]$	$v_{2_{i+1}}^j$	$[(L_{p_{i+1}}^j - 1)/2] + 1$
		$L_{p_{i+1}}^j \in \text{even}$	$L_{p_{i+1}}^j = 2$	$R_{I-1}^{all} + [v_{1_{i+1}}^j \times \Pr(C_i = 1)]$	-3	1
			$L_{p_{i+1}}^j > 2$	$R_{I-1}^{all} + [v_{1_{i+1}}^j \times \Pr(C_i = 1)]$	$v_{2_{i+1}}^j$	$[(L_{p_{i+1}}^j - 2)/2] + 1$

$c\lambda_i$ is the intermediate factor after eliminating node C_i. Similar to cOS, the jth row of $c\lambda_i$ is $\{run, q_i^j, n_{r_i}^j\}$, the q_i^jth row of d_{r_i} is $\{q_i^j, r_i^j, L_{r_i}^j\}$,and the p_i^jth row of d_{p_i} is $\{p_i^j, v_{1_i}^j, v_{2_i}^j, L_{p_i}^j\}$. In Table 2, Table 3 and Table 4, $ismb(\mathbf{x}, \mathbf{y})$ is the function that finds the position of each element of $\mathbf{x}$ in $\mathbf{y}$, and $length(\mathbf{x})$ is the function that calculate the length of $\mathbf{x}$. *-3* means that the value of corresponding variable is non-existent. S_{i+1}^j is the run or phrase row start number of the jth row in $c\lambda_{i+1}$. $\Pr(C_i)$ is the marginal probability distribution of node C_i. J_{i+1}^j is a set and $J_{i+1}^j(1)$ is the first element of J_{i+1}^j. L_{RP} is the elements' numbers of RP_{i+1}^j and $i_{RP} = 1, 2, \cdots, L_{RP}$. $J_{i+1}^j(1, i_{RP})$ means the i_{RP}th element of J_{i+1}^j. R_{i+1}^j is reminder after finishing the calculation of the jth row in $c\lambda_{i+1}$, and R^{all} is the set of all R_{i+1}^j. $R_{J_{i+1}^j(1,i_{RP})}^{all}$ is the $J_{i+1}^j(1, i_{RP})$th element of R^{all}. I means the position of S_{i+1}^j in S_{i+1}^{all}. R_{I-1}^{all} is the $(I-1)$th element of R^{all}.

Especially, if the jth row of $c\lambda_{i+1}$ is a run and S_{i+1}^j is an even and $L_{r_{i+1}}^j > 2$, the jth row and $(j+1)$th row will be constructed at the same time. For this particular situation, RP_{i+1} and S_{i+1}^{all} should also be updated synchronously. The detail process is shown as follow:

The jth row's run of $c\lambda_{i+1}$ is divided into two equivalent runs. Supposed that $R1_{i+1}^j$ and $R2_{i+1}^j$ means the start row number of the first and the second equivalent run, respectively. Therefore,

$$R1_{i+1}^j = \{k_l \mid l = 1, 2, \cdots, n_{r_i}^j\} \tag{3}$$

$$R2_{i+1}^j = \{k_l + 1 \mid l = 1, 2, \cdots, n_{r_i}^j\} \tag{4}$$

After eliminating node C_i, RP_{i+1} and S_{i+1}^{all} should be updated to be RP_{i+1}^{update} and S_{i+1}^{update} as shown in equation (5) and equation (6), respectively.

$$\begin{aligned} RP_{i+1}^{update} &= \{RP_{i+1}^1, RP_{i+1}^2, \cdots, RP_{i+1}^j, \cdots, RP_{i+1}^{m_{i+1}}\} \\ &= \{RP_{i+1}^1, \cdots, RP_{i+1}^{j-1}, R1_{i+1}^j, R2_{i+1}^j, RP_{i+1}^{j+1} \cdots, RP_{i+1}^{m_{i+1}}\} \end{aligned} \tag{5}$$

$$\begin{aligned} S_{i+1}^{update} &= \{S_{i+1}^1, S_{i+1}^2, \cdots, S_{i+1}^{j-1}, S_{i+1}^j, S_{i+1}^{j+1}, \cdots, S_{i+1}^{m_{i+1}}\} \\ &= \{S_{i+1}^1, S_{i+1}^2, \cdots, S_{i+1}^{j-1}, S_{i+1}^j, (S_{i+1}^j + 1), S_{i+1}^{j+1}, \cdots, S_{i+1}^{m_{i+1}}\} \end{aligned} \tag{6}$$

3.2. New inference algorithm

Based on VE, a new inference algorithm is developed as shown in Table 5. Given that the marginal probability distribution $\Pr(C_i)$, query node set Q and evidence set E, $\Pr(S \mid Q, E)$ is calculated by the follow steps:

① According to E, if $C_i \in E$, $\Pr(C_i)$ is updated.

② If $C_i \in Q$, record C_i to the extreme left of NPT's parent nodes, i.e. numbered 1. The recorded NPT is denoted as $NPT^{reorder}$.

Table 5: New inference algorithm

Input	$\Pr(C_i)$, Q, E
Output	$\Pr(Ch \mid Q, E)$
Step 1	For $i \leftarrow 1$ to n, do If $C_i \in E$, update $\Pr(C_i)$. End and output updated $\Pr(C_i = 0)$
Step 2	For $i \leftarrow n$ to 1, do If $C_i \in Q$, record C_i to the extreme left of NPT's parent nodes, i.e. numbered 1. End and output $NPT^{reorder}$.
Step 3	Compress $P(S=1 \mid C_1, C_2, \cdots, C_n)$ column by the developed new compression algorithm as shown in Figure 1 to get $c\lambda_{n+1}$ and calculate $N_Q = length(Q)$.
Step 4	For $i \leftarrow n$ to (N_Q+1), do (a)-(d) (a) Calculate row numbers m_{i+1} of $c\lambda_{i+1}$. (b) Eliminate C_i. (c) For $j_1 \leftarrow 1$ to m_{i+1}, do Construct $c\lambda_i^{new-j_1}$, $d_{r_i}^{new-j_1}$ and $d_{p_i}^{new-j_1}$ by the rules in Table 2, Table 3 and Table 4. Update RP_{i+1} and S_{i+1}^{all} to get RP_{i+1}^{update} and S_{i+1}^{update}, respectively. End and output $c\lambda_i^{new}$, $d_{r_i}^{new}$, $d_{p_i}^{new}$, RP_{i+1}^{update} and S_{i+1}^{update}. (d) Row by row, decompress one row of $c\lambda_i^{new}$ by basing on RP_{i+1}^{update} and S_{i+1}^{update}, then compress this row. After process all rows of $c\lambda_i^{new}$, the final output are $c\lambda_i$, d_{r_i}, d_{p_i}, RP_i and S_i^{all}. End and output $c\lambda_{N_Q+1}$, $d_{r_{N_Q+1}}$, $d_{p_{N_Q+1}}$, RP_{N_Q+2} and $S_{N_Q+2}^{all}$.
Step 5	Decompress $c\lambda_{N_Q+1}$ by basing on RP_{N_Q+2} and $S_{N_Q+2}^{all}$ to get $\Pr(Ch = j_3 \mid Q, E)$.

③ The $P(S=1 \mid C_1, C_2, \cdots, C_n)$ column of NPT is compressed by the developed new compression algorithm as shown in Figure 1 to get $c\lambda_{n+1}$.

④ Based on VE, the nodes that are not included in Q are eliminated one by one.

4. NUMERICAL EXAMPLE

4.1. Background and BN modeling

The satellite antenna transmission system is composed of a pitch axis subsystem and an azimuth axis subsystem in parallel. The two subsystems are composed of a stepping motor (C_1, C_4), a drive shaft (C_2, C_5) and a harmonic reducer (C_3, C_6) in series as shown in Figure 2, respectively.

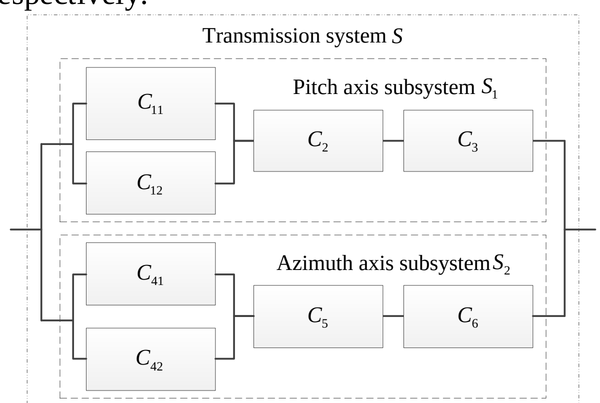

Figure 2: Satellite transmission system

The marginal probability distribution of components is shown in Table 6. According to the logical relationship between transmission system and its components, the BN model can be constructed as shown in Figure 3.

Table 6: Marginal probability distribution

C_i	C_{11}	C_{12}	C_2	C_3	C_{41}	C_{42}	C_5	C_6
$\Pr(C_i=0)$	0.09	0.09	0.12	0.05	0.09	0.09	0.12	0.05
$\Pr(C_i=1)$	0.91	0.91	0.88	0.95	0.91	0.91	0.88	0.95

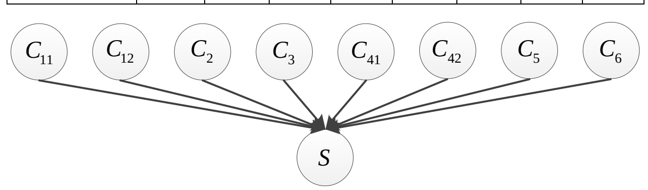

Figure 3: BN of satellite transmission system

4.2. Reliability inference

Based on the developed new compressed algorithm and new inference algorithm, the inference of BN in Figure 3 can be performed and the results are shown in Table 7.

Table 7: Probability distribution of S

State	$S=0$	$S=1$
Probability	0.0292	0.9708

As shown in Figure 2, satellite antenna transmission system has two subsystems. If the number of subsystems is gradually increased, the maximum memory storage requirement of NPT is shown in Figure 4. When the number of subsystems is 20, the maximum memory storage requirement of NPT is out of the maximum memory storage of the platform in using BNT method. However, the proposed algorithms can still perform the inference of BN. Besides, the maximum memory storage requirement of the proposed algorithms is lower than the original algorithms.

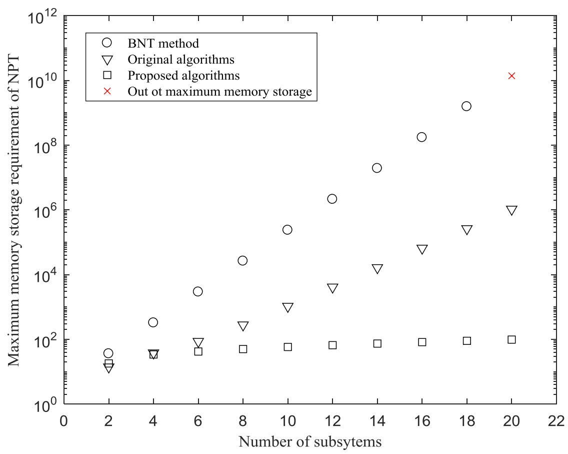

Figure 4: Maximum memory storage requirements

5. CONCLUSIONS

To completely compress the NPT, new compression algorithm is developed in this paper. Based on the compressed NPT, new inference algorithm is developed for performing the inference of BN model at the same time. The proposed algorithms are validated by the reliability analysis of satellite antenna transmission system.

6. ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (No.51675525 and 11725211).

7. REFERENCES

Dechter, R. (1998). *Bucket Elimination: A Unifying Framework for Probabilistic Inference*, Springer Netherlands.

Hauck, E. L. (1986). "Data compression using run length encoding and statistical encoding." US.

Mi, J., Li, Y. F., Peng, W., and Huang, H. Z. (2018). "Reliability Analysis of Complex Multi-State System with Common Cause Failure Based on Evidential Networks." *Reliability Engineering & System Safety*, 174.

Mi, J., Li, Y. F., Yang, Y. J., Peng, W., and Huang, H. Z. (2016). "Reliability assessment of complex electromechanical systems under epistemic uncertainty." *Reliability Engineering & System Safety*, 152, 1-15.

Pearl, J. (1988). *Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann Publishers.

Sturlaugson, L., and Sheppard, J. W. (2015). "Sensitivity Analysis of Continuous Time Bayesian Network Reliability Models." *SIAM/ASA Journal on Uncertainty Quantification*, 3(1), 346-369.

Su, C., and Fu, Y. Q. (2014). "Reliability Assessment for Wind Turbines Considering the Influence of Wind Speed Using Bayesian Network." *Eksploatacja i Niezawodnosc - Maintenance and Reliability*, 16(1), 1-8.

Tien, I., and Kiureghian, A. D. (2016). "Algorithms for Bayesian network modeling and reliability assessment of infrastructure systems." *Reliability Engineering & System Safety*, 156, 134-147.

Tong, Y., and Tien, I. (2017). "Algorithms for Bayesian Network Modeling, Inference, and Reliability Assessment for Multistate Flow Networks." *Journal of Computing in Civil Engineering*, 31(5), 04017051.

Ziv, J., and Lempel, A. (1977). "Lempel, A.: A Universal Algorithm for Sequential Data Compression. IEEE Transactions on Information Theory 23(3), 337-343." *IEEE Transactions on Information Theory*, 23(3), 337-343.