Dynamic Uncertain Causality Graph for Knowledge Representation and Probabilistic Reasoning: Statistics Base, Matrix, and Application

Qin Zhang, Chunling Dong, Yan Cui, and Zhihui Yang

***Abstract*— Graphical models for probabilistic reasoning are now in widespread use. Many approaches have been developed such as Bayesian network. A newly developed approach named as dynamic uncertain causality graph (DUCG) is initially presented in a previous paper, in which only the inference algorithm in terms of individual events and probabilities is addressed. In this paper, we first explain the statistic basis of DUCG. Then, we extend the algorithm to the form of matrices of events and probabilities. It is revealed that the representation of DUCG can be incomplete and the exact probabilistic inference may still be made. A real application of DUCG for fault diagnoses of a generator system of a nuclear power plant is demonstrated, which involves >600 variables. Most inferences take <1 s with a laptop computer. The causal logic between inference result and observations is graphically displayed to users so that they know not only the result, but also why the result obtained.**

***Index Terms*— Causality, complex system, fault diagnosis, probabilistic reasoning, uncertainty.**

I. Introduction

OVER last two decades or so, graphical models for probabilistic reasoning received a lot of attention [1]–[3]. Typical models include Bayesian network (BN) [4]–[7], hidden Markov models [8], latent tree models [9], dependency networks [10], cloud model [11], and so on. In which, BN is in widespread use [12]–[15]. However, the probabilistic inference of BN is NP hard [16], [17], while today's inference efficiency requirement is strict. Thus, many compact representation models and corresponding inference algorithms are presented (e.g., noisy-OR [4], dynamic causality diagram [18], context-specific independence [19], and independence of causal influence [20]). They use less parameters to represent the uncertain causal relationship between parent variables and child variable and achieve higher inference efficiency than using traditional conditional probability tables (CPTs). However, they have to apply the imposed normalization formula in multivalued cases [21], which is questioned in [22].

Manuscript received May 12, 2013; revised July 7, 2013; accepted August 19, 2013. Date of publication September 16, 2013; date of current version March 10, 2014. This work was supported in part by China General Nuclear Power Corporation under Contract CNPRI-ST10P005 and in part by the National Natural Science Foundation of China under Grant 61050005 and Grant 61273330.

Q. Zhang and C. Dong are with the School of Computer Science and Engineering, Beihang University, Beijing 100191, China (e-mail: zhangqin@buaa.edu.cn; dongcl@act.buaa.edu.cn).

Y. Cui and Z. Yang are with China General Nuclear Power Corporation, Shenzhen 518000, China (e-mail: kaijie_luo@163.com; yangzhihui@cgnpc.com.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2013.2279320

As defined, BN is quantified by a series of CPTs, in which the logic relationships among parent variables are hidden. In this regard, the causality representation in BN can be referred to as implicit, although the child–parent relationships (causal dependences) are explicitly represented. Because of this, the parameter accuracy of BN is important. In many applications such as pattern recognition, these parameters can be learned from statistic/historical data. But for some applications such as fault diagnoses of nuclear power plants, the statistic failure data are usually rare and unusable. This is because: 1) these large, complex, and expensive systems are usually unique, high reliable, and each is different from others even in a same type and 2) in many cases, once a failure happens, this and other similar systems are improved to prevent it from happening again, resulting in the historical failure data unusable. Furthermore, in real applications, what is expected for the intelligent system to do includes to find the root failures never happened before (the novelty identifiability, one of the eight requirements presented in [23]). In these areas, only the domain expert's knowledge is available, which is based on the understanding and experience of domain experts to each specific system and is therefore in the form of explicit uncertain causal relationships among variables.

To be practical, an intuitive and explicit uncertain causality representation tool is needed for domain experts to represent and maintain their knowledge base, while the constructed knowledge base should work well with its corresponding probabilistic inference. However, BN is based on CPTs. When the number of parent variables increases, the number of parameters in a CPT increases exponentially, and the large number of parameters is difficult to be specified by domain experts directly. Another challenge for BN is that the inference results should be explanatory; otherwise users may not accept them [24]. However, the inference of BN is mainly to update the probability distributions of variables/nodes. The inference results are then identified according to the posterior probability distributions. Therefore, the results are not easy to be intuitively explained to users, at least are hard to be explained with causal graphs.

As a newly developed framework of intelligent system, dynamic uncertain causality graph (DUCG) aims to represent uncertain causal knowledge compactly and intuitively, provide efficient probabilistic reasoning, and make the inference results explanatory. DUCG can: 1) represent complex causalities explicitly and easily with graphical symbols including logic gates representing any logic relationship among

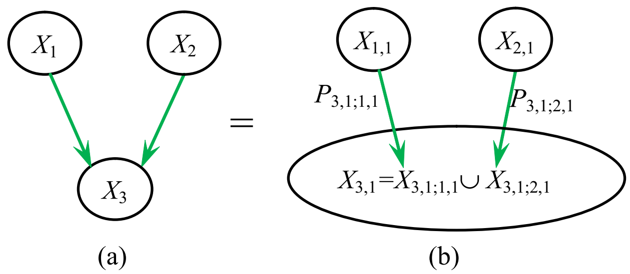


Fig. 1. (a) Example of S-DUCG. (b) Illustration for Fig. 2(a).

	$X_{3,1}$	$X_{3,0}$	Total
$X_{1,1}X_{2,0}$	30	70	100
$X_{1,0}X_{2,1}$	120	80	200
$X_{1,1}X_{2,1}$	72	28	100
$X_{1,0}X_{2,0}$	0	200	200
Total	222	378	600

(a)

	$X_{3,1}$	$X_{3,0}$
$X_{1,1}X_{2,0}$	0.3	0.7
$X_{1,0}X_{2,1}$	0.6	0.4
$X_{1,1}X_{2,1}$	0.72	0.28
$X_{1,0}X_{2,0}$	0.0	1.0
Unknown	0.37	0.63

(b)

Fig. 2. Illustrative (a) statistic samples and (b) CPT represented in Fig. 1.

variables; 2) combine separately constructed sub-DUCGs into a whole DUCG, so that the maintenance of DUCG knowledge base is easy and each sub-DUCG is simple and well understood; 3) reduce the scale of problem for observed evidence significantly, so as to find all possible hypotheses and achieve high inference efficiency; and 4) deal with not only single-valued cases, but also multivalued cases. The so-called single-valued case means that only the causes of the true state of a child variable are specified, and the false state is just the complement of the true state without specifying its causes separately. The so-called multivalued case means that the causes of all states of a child variable can be specified separately. The DUCG involving single-valued cases is called S-DUCG. The DUCG involving multivalued cases is called M-DUCG. The mixture of them is called DUCG. No imposed normalization formula is needed. The probability parameters of M-DUCG can be incomplete (specifying only those in concern), while the exact probabilistic inference can still be made, which provides people with great convenience to apply DUCG.

The rest of this paper is organised as follows. Section I explains the statistic basis of DUCG, Section II introduces M-DUCG in details and points out the need for the algorithm in terms of matrices, Section III presents the algorithm in terms of matrices, Section IV discusses how an exact probabilistic inference can be made based on incomplete DUCG, Section V reports a real application of DUCG to show some properties of DUCG: 1) the decomposed construction of knowledge base; 2) the simplification of DUCG according to the observed evidence; 3) the efficiency of inference in dealing with large and complex systems; and 4) the graphical explanation to inference results. In addition, the benefit of constructing an incomplete DUCG for real application is explored. Finally, Section VI concludes this paper and outlines the future work.

Limited by the length, this paper addresses only the static and discrete cases. The dynamic and continuous cases will be addressed in following papers.

II. Statistic Basis of DUCG

A. S-DUCG Model

The basic idea of S-DUCG is shown in Fig. 1, in which Fig. 1(b) reveals the functional mechanism inside Fig. 1(a) (Fig. 1(b) is Fig. 12 in [22]). In this example, $X_{1,1}$ and $X_{2,1}$ are in OR relationship in causing $X_{3,1}$. In DUCG, we use the first subscript to index the variable and the second subscript to index the state of the variable. A comma between them is used to divide them, e.g., $X_{2,1}$ indicates state 1 of X_2. In the case without confusion, the comma can be ignored, e.g., X_{n1}, X_{ij}, and so on.

In single-valued cases, only the causes of the true state of a child variable are specified. For simplicity, state 1 denotes the true state and state 0 denotes the false state. The false state is just the complement of the true state. This is why only the causes of $X_{3,1}$ are specified in Fig. 1(b). In this example, $X_{1,1}$ and $X_{2,1}$ are the only causes of $X_{3,1}$.

In S-DUCG, the CPT between a child variable X_n and its parent variables V_i (e.g., $V = X$ and $i\in\{1,2\}$) is replaced by the linkage events $P_{n1;ij}$ and their occurrence probabilities $p_{n1;ij}$, where $P_{n1;ij}$ is defined as independent of $P_{n1;i'j'}$, $i \neq i$'. In other words, the OR relationship implicitly represented in the CPT as shown in Fig. 2(b) is explicitly and compactly represented by events $P_{n1;ij}$ and their probabilities $p_{n1;ij} \equiv \Pr\{P_{n1;ij}\}$ encoded in the green directed arcs, where ";" is used to divide the subscripts of parent event V_{ij} and the subscripts of child event X_{n1}, and $X_{n1;ij}$ is defined as the event that $P_{n1;ij}V_{ij}$ causes X_{n1}. Thus, for the example shown in Fig. 1(b), we have $X_{3,1} = X_{3,1;1,1} \cup X_{3,1;2,1} = P_{3,1;1,1}X_{1,1} \cup P_{3,1;2,1}X_{2,1}$, and $X_{3,2} = \bar{X}_{3,1} = 1 - X_{3,1}$ ("1" also denotes the complete set).

Fig. 1 is actually the noisy-OR model in BN [4], but is more intuitive. Its statistic basis can be illustrated as follows: suppose samples are collected as shown in Fig. 2(a), the corresponding CPT is shown in Fig. 2(b).

In terms of S-DUCG, these data can be modeled as shown in Fig. 1(a) by specifying two parameters $p_{3,1;1,1} = 0.3$ and $p_{3,1;2,1} = 0.6$, which means all $P_{n1;ij} = 0$ (null) except $P_{3,1;1,1}$ and $P_{3,1;2,1}$. As the relationship between $X_{1,1}$ and $X_{2,1}$ is OR in Fig. 1 (the default relationship defined in S-DUCG), we have

$$\begin{aligned} Pr\{X_{3,1}|X_{1,1}X_{2,1}\} &= Pr\{P_{3,1;1,1} \cup P_{3,1;2,1}\} \\ &= p_{3,1;1,1} + p_{3,1;2,1} - p_{3,1;1,1}p_{3,1;2,1} \\ &= 0.3 + 0.6 - 0.3 \times 0.6 = 0.72. \end{aligned}$$

	$X_{3,1}$	$X_{3,0}$	Total
$X_{1,1}X_{2,0}$	60	40	100
$X_{1,0}X_{2,0}$	60	140	200
$X_{1,1}X_{2,1}$	20	80	100
$X_{1,0}X_{2,1}$	30	270	300
Total	170	530	700

(a)

	$X_{3,1}$	$X_{3,0}$
$X_{1,1}X_{2,0}$	0.6	0.4
$X_{1,0}X_{2,0}$	0.3	0.7
$X_{1,1}X_{2,1}$	0.2	0.8
$X_{1,0}X_{2,1}$	0.1	0.9
Unknown	0.243	0.757

(b)

Fig. 3. Statistic samples without clear logic relationship. (a) Samples. (b) CPT.

That is, when $X_{1,1}$ and $X_{2,1}$ occur simultaneously, the conditional probability of $X_{3,1}$ is enhanced from $\Pr\{X_{3,1}|X_{1,1}X_{2,0}\} = 0.3$ and $\Pr\{X_{3,1}|X_{1,0}X_{2,1}\} = 0.6$, respectively, to $\Pr\{X_{3,1}|X_{1,1}X_{2,1}\} = 0.72$. Therefore, S-DUCG is an enhancement model. Similarly

$$\begin{aligned} Pr\{X_{3,1}|X_{1,0}X_{2,0}\} &= Pr\{P_{3,1;1,0} \cup P_{3,1;2,0}\} \\ &= Pr\{0 \cup 0\} = 0 \\ Pr\{X_{3,0}|X_{1,0}X_{2,0}\} &= 1 - Pr\{X_{3,1}|X_{1,0}X_{2,0}\} = 1. \end{aligned}$$

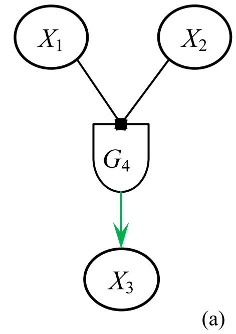
Note that "0" also denotes null set.

As explained in [22], S-DUCG can only be applied in single-valued cases, which means that the causes of X_{n0} should not be specified separately from X_{n1} and can only be specified as $X_{n0} = 1-X_{n1}$. Based on this, the other parameters of CPT in Fig. 2(a) can be calculated, e.g., $\Pr\{X_{3,0}|X_{1,1}X_{2,0}\} = 1-0.3 = 0.7$.

The original samples also show that the unconditional probability distribution of X_3 is $x_{3,1} \equiv \Pr\{X_{3,1}\} = 0.37$ and $x_{3,0} \equiv \Pr\{X_{3,0}\} = 0.63$. However, without knowing the original samples, the unconditional probability distribution cannot be calculated only from the CPT. Hence, the original samples include more information than CPT does. The following examples are the same. How to use the extra information of samples will be discussed in another paper.

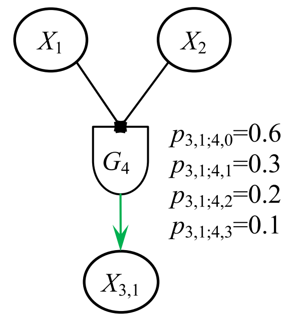
In practice, if we are sure that $X_{1,1}$ and $X_{2,1}$ are in OR relationship, we can put statistic data into this model and find the conditional probabilities (parameters) using various methods such as the regression method, and so on, even if samples do not exactly match (e.g., the samples of $X_{1,1}X_{2,1}X_{3,1}$ to $X_{1,1}X_{2,1}X_{3,0}$ in Fig. 1(a) is not exactly 72:28). This is a learning process.

When we do not have samples, we can give parameters $p_{n1;ij}$ directly according to domain expert's knowledge. In this example, if the causal relationship between $X_{1,1}$ and $X_{2,1}$ is known as OR, we can specify parameters $p_{3,1;1,1} = 0.3$ and $p_{3,1;2,1} = 0.6$ directly. The other parameters can be calculated as shown above. Of course, we may have other logic relationships, such as AND, NOT, XOR, and so on. In S-DUCG, if statistic samples cannot be put into the OR model, we may use other models as shown in Figs. 8–10 and 14–21 in [22], and so on, in which the logic gates G_i may be employed. The most complex relationship such as that shown in Fig. 3 can be modeled by a completely combined logic gate shown in Fig. 4, in which LGS_4 denotes the logic gate specification of G_4. This is the worst case and no compactness is achieved.



j	G_{4j}
0	$X_{1,0}X_{2,0}$
1	$X_{1,1}X_{2,0}$
2	$X_{1,0}X_{2,1}$
3	$X_{1,1}X_{2,1}$

(a)



(b)

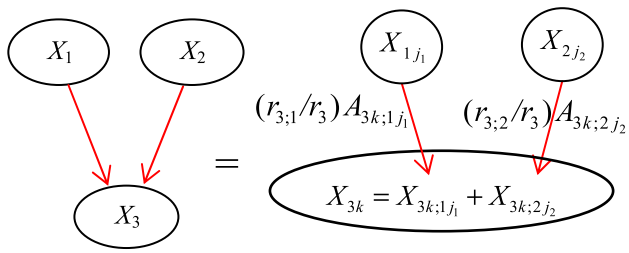
Fig. 4. S-DUCG that models the case of Fig. 3 using completely combined logic gate G_4. (a) LGS_4. (b) Parameters.

From these examples, it is easy to see that S-DUCG is able to model any single-valued case and can be as compact as possible.

Given the statistic samples shown in Fig. 3(a), the unconditional probability distribution of X_3 is $x_{3,1} \equiv \Pr\{X_{3,1}\} = 0.243$ and $x_{3,0} \equiv \Pr\{X_{3,0}\} = 0.757$. As mentioned above, the unconditional probabilities cannot be calculated directly from the CPT.

B. *M*-DUCG Model

The M-DUCG model is shown in Fig. 5 that is included in Fig. 26 in [22], in which, similar to the idea of S-DUCG, $X_{3k;ij}$ denotes the event that $(r_{3;i}/r_3)A_{3k;ij}V_{ij}$ causes X_{3k}, event $A_{3k;ij}$ represents the physical mechanism that parent event V_{ij} causes $X_{3k;ij}$, $(r_{3;i}/r_3)$ is a weighting factor associated with $A_{3k;ij}$, where $r_{n;i} > 0$ is called the causal relationship intensity between V_i and X_n, $r_n \equiv \sum_i r_{n;i}$. For simplicity, subscript j_i in Fig. 5 is abbreviated as j. Note that the directed arcs in M-DUCG are red, so as to be different from S-DUCG. In Fig. 5, suppose X_1, X_2, and X_3 have two states each indexed by 0 and 1, respectively. Different from S-DUCG, $X_{3,0}$ is not simply the complement of $X_{3,1}$ and the



(a) (b)

Fig. 5. (a) Example of M-DUCG. (b) Illustration for Fig. 5(a).

	$X_{3,0}$	$X_{3,1}$	Total
$X_{1,1}X_{2,0}$	60	40	100
$X_{1,0}X_{2,1}$	150	50	200
$X_{1,1}X_{2,1}$	70	30	100
$X_{1,0}X_{2,0}$	65	35	100
Total	345	155	500

(a)

	$X_{3,0}$	$X_{3,1}$
$X_{1,1}X_{2,0}$	0.6	0.4
$X_{1,0}X_{2,1}$	0.75	0.25
$X_{1,1}X_{2,1}$	0.7	0.3
$X_{1,0}X_{2,0}$	0.65	0.35
Unknown	0.69	0.31

(b)

Fig. 6. Illustrative (a) statistic samples and (b) corresponding CPT of Fig. 5.

causes of $X_{3,0}$ can be specified separately from $X_{3,1}$. This is an essential difference between S-DUCG and M-DUCG.

The statistic samples of Fig. 5 might be as shown in Fig. 6(a), and its corresponding CPT is shown in Fig. 6(b).

In M-DUCG, the samples in Fig. 6(a) can be modeled as Fig. 5, in which

$$a_{3;1} = \begin{pmatrix} a_{3,0;1,0} & a_{3,0;1,1} \\ a_{3,1;1,0} & a_{3,1;1,1} \end{pmatrix} = \begin{pmatrix} 0.75 & 0.6 \\ 0.25 & 0.4 \end{pmatrix}$$

$$a_{3;2} = \begin{pmatrix} a_{3,0;2,0} & a_{3,0;2,1} \\ a_{3,1;2,0} & a_{3,1;2,1} \end{pmatrix} = \begin{pmatrix} 0.6 & 0.75 \\ 0.4 & 0.25 \end{pmatrix}$$

$$r_{3;1} = 1, \qquad r_{3;2} = 2.$$

They are encoded in the red directed arcs in Fig. 5. The learning method can be similar to existing methods, e.g., EM, and is a topic of another paper. As defined in M-DUCG, $r_3 = r_{3;1} + r_{3;2} = 1 + 2 = 3$, $(r_{3;1}/r_3) = (1/3)$, $(r_{3;2}/r_3) = (2/3)$. According to the CPT expression defined in M-DUCG: $\Pr\{X_{nk} | \cap_i V_{ij}\} = \Sigma_i (r_{n;i}/r_n) a_{nk;ij}$, which is (35) in [22], we can calculate the CPT exactly as shown in Fig. 6(b). For example

$$\begin{aligned} \Pr\{X_{3,0}|X_{1,1}X_{2,0}\} &= (r_{3;1}/r_3)a_{3,0;1,1} + (r_{3;2}/r_3)a_{3,0;2,0} \\ &= (1/3)0.6 + (2/3)0.6 = 0.6 \\ \Pr\{X_{3,0}|X_{1,0}X_{2,1}\} &= (r_{3;1}/r_3)a_{3,0;1,0} + (r_{3;2}/r_3)a_{3,0;2,1} \\ &= (1/3)0.75 + (2/3)0.75 = 0.75. \end{aligned}$$

	$X_{4,0}$	$X_{4,1}$	$X_{4,2}$	Total
$X_{1,0}X_{2,0}X_{3,0}$	210	60	30	300
$X_{1,0}X_{2,0}X_{3,1}$	300	140	160	600
$X_{1,0}X_{2,1}X_{3,0}$	80	25	45	150
$X_{1,0}X_{2,1}X_{3,1}$	100	60	140	300
$X_{1,1}X_{2,0}X_{3,0}$	140	60	100	300
$X_{1,1}X_{2,0}X_{3,1}$	80	70	150	300
$X_{1,1}X_{2,1}X_{3,0}$	90	50	160	300
$X_{1,1}X_{2,1}X_{3,1}$	60	120	420	600
Total	1060	585	1205	2850

(a)

	$X_{4,0}$	$X_{4,1}$	$X_{4,2}$
$X_{1,0}X_{2,0}X_{3,0}$	0.7	0.2	0.1
$X_{1,0}X_{2,0}X_{3,1}$	0.5	0.2333	0.2667
$X_{1,0}X_{2,1}X_{3,0}$	0.5333	0.1667	0.3
$X_{1,0}X_{2,1}X_{3,1}$	0.3333	0.2	0.4667
$X_{1,1}X_{2,0}X_{3,0}$	0.4667	0.2	0.3333
$X_{1,1}X_{2,0}X_{3,1}$	0.2667	0.2333	0.5
$X_{1,1}X_{2,1}X_{3,0}$	0.3	0.1667	0.5333
$X_{1,1}X_{2,1}X_{3,1}$	0.1	0.2	0.7
Unknown	0.372	0.205	0.423

(b)

Fig. 7. (a) Statistic samples and (b) corresponding CPT in a more complex case.

It is noted that the causes of both $X_{3,0}$ and $X_{3,1}$ are specified separately. For example, the cause of $X_{3,0}|X_{1,1}X_{2,0}$ is specified as $(r_{3;1}/r_3)A_{3,0;1,1}+(r_{3;2}/r_3)A_{3,0;2,0}$ and the cause of $X_{3,1}|X_{1,1}X_{2,0}$ is specified as $(r_{3;1}/r_3)A_{3,1;1,1} + (r_{3;2}/r_3)A_{3,1;2,0}$, and so on. In addition, this example illustrates that M-DUCG is a weighted average model. For example, when $X_{1,1}$ and $X_{2,1}$ occur simultaneously, the conditional probability of $X_{3,0}$ averages from $\Pr\{X_{3,0}|X_{1,1}X_{2,0}\} = 0.6$ and $\Pr\{X_{3,0}|X_{1,0}X_{2,1}\} = 0.75$, respectively, to $\Pr\{X_{3,0}|X_{1,1}X_{2,1}\} = 0.7$, and the conditional probability of $X_{3,1}$ averages from $\Pr\{X_{3,1}|X_{1,1}X_{2,0}\} = 0.4$ and $\Pr\{X_{3,1}|X_{1,0}X_{2,1}\} = 0.25$, respectively, to $\Pr\{X_{3,1}|X_{1,1}X_{2,1}\} = 0.3$. This is another essential difference between M-DUCG and S-DUCG. Someone may argue that if $X_{1,1}$ and $X_{2,1}$ occur simultaneously, the conditional probability of $X_{3,1}$ should be enhanced as in S-DUCG, but unfortunately, we have not found such a model that achieves both the applicability in multivalued cases and the enhancement mechanism. In other words, until now, M-DUCG is the only available model theoretically consistent in dealing with multivalued cases.

It is seen that the number of parameters encoded in Fig. 5 is 10, while the number of parameters in the corresponding CPT is 8. It seems that no compactness is achieved in M-DUCG. However, even in this small case, the causal relationships of $\{X_3,\ X_1\}$ and $\{X_3,\ X_2\}$ are represented separately. To demonstrate the compactness of M-DUCG, consider a more complex case shown in Fig. 7.

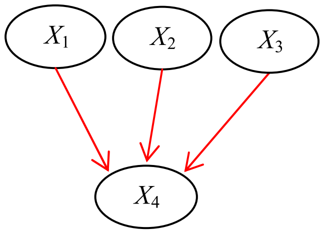


Fig. 8. M-DUCG with three parent variables.

With M-DUCG, these data can be modeled as Fig. 8. The parameters encoded in Fig. 8 are

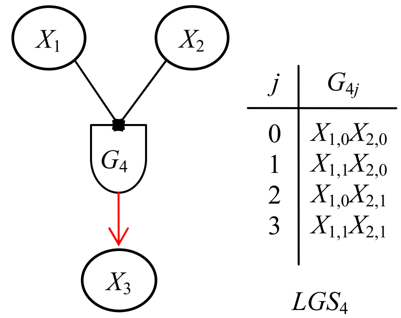
$$a_{4;1}=\begin{pmatrix}a_{4,0;1,0} & a_{4,0;1,1}\\ a_{4,1;1,0} & a_{4,1;1,1}\\ a_{4,2;1,0} & a_{4,2;1,1}\end{pmatrix}=\begin{pmatrix}0.8 & 0.1\\ 0.1 & 0.1\\ 0.1 & 0.8\end{pmatrix}$$
$$a_{4;2}=\begin{pmatrix}a_{4,0;2,0} & a_{4,0;2,1}\\ a_{4,1;2,0} & a_{4,1;2,1}\\ a_{4,2;2,0} & a_{4,2;2,1}\end{pmatrix}=\begin{pmatrix}0.6 & 0.1\\ 0.3 & 0.2\\ 0.1 & 0.7\end{pmatrix}$$
$$a_{4;3}=\begin{pmatrix}a_{4,0;3,0} & a_{4,0;3,1}\\ a_{4,1;3,0} & a_{4,1;3,1}\\ a_{4,2;3,0} & a_{4,2;3,1}\end{pmatrix}=\begin{pmatrix}0.7 & 0.1\\ 0.2 & 0.3\\ 0.1 & 0.6\end{pmatrix}.$$
$$r_{4;i}=1,\text{ then } (r_{4;i}/r_4)=(1/3),\quad i=1,2,3.$$

With the above parameters, the corresponding CPT can be calculated exactly as shown in Fig. 7(b). For example

$$\begin{aligned}\Pr\{X_{4,0}|X_{1,0}X_{2,0}X_{3,0}\} &= (1/3)a_{4,0;1,0}+(1/3)a_{4,0;2,0}\\ &\quad +(1/3)a_{4,0;3,0}\\ &= (1/3)0.8+(1/3)0.6\\ &\quad +(1/3)0.7=0.7\\ \Pr\{X_{4,1}|X_{1,0}X_{2,0}X_{3,0}\} &= (1/3)a_{4,1;1,0}+(1/3)a_{4,1;2,0}\\ &\quad +(1/3)a_{4,1;3,0}\\ &= (1/3)0.1+(1/3)0.3\\ &\quad +(1/3)0.2=0.2\\ \Pr\{X_{4,2}|X_{1,0}X_{2,0}X_{3,0}\} &= (1/3)a_{4,2;1,0}+(1/3)a_{4,2;2,0}\\ &\quad +(1/3)a_{4,2;3,0}\\ &= (1/3)0.1+(1/3)0.1\\ &\quad +(1/3)0.1=0.1.\end{aligned}$$

Therefore, the M-DUCG shown in Fig. 8 does model the data shown in Fig. 7. It is seen that the number of parameters in this CPT is 24, while the number of $\{a\text{-}, r\text{-}\}$ type parameters in this M-DUCG is 21. M-DUCG is compact. In general, if parent variables and/or variable states increase, the number of parameters in the corresponding CPT increases exponentially. For example, in the case of five parent variables and five states for all the six variables, the number of parameters in the CPT is $5^6 = 15\,625$, while the number of $\{a\text{-}, r\text{-}\}$ type parameters in M-DUCG is only $5^3 + 5 = 130$. Therefore, the compactness achieved in M-DUCG can be great. Plus the separate representation for each parent variable, DUCG is desired in case the parameters have to be specified by users directly.

Similar to the situation of S-DUCG, the most complex case can be represented by a completely combined logic gate.



(a)

$$a_{3;4}=\begin{pmatrix}a_{3,0;4,0} & a_{3,0;4,1} & a_{3,0;4,2} & a_{3,0;4,3}\\ a_{3,1;4,0} & a_{3,1;4,1} & a_{3,1;4,2} & a_{3,1;4,3}\end{pmatrix}$$
$$=\begin{pmatrix}0.6 & 0.3 & 0.2 & 0.1\\ 0.4 & 0.7 & 0.8 & 0.9\end{pmatrix}$$

(b)

Fig. 9. Illustration for the completely combined logic gate in M-DUCG. (a) Completely combined logic gate in M-DUCG. (b) α-type parameters.

For the example of Fig. 3(a), we can use a completely combined logic gate G_4 to model the samples as shown in Fig. 9(a), where the green directed arc in Fig. 4(a) is changed to red in Fig. 9(a).

The a-type parameters encoded in Fig. 9(a) are as shown in Fig. 9(b), while the r-type parameter is useless because $(r_{n;i}/r_n) = (r_{n;i}/r_{n;i}) = 1$. X_3 can then be expressed in terms of matrices of events and probabilities, respectively, as

$$X_3 = F_{3;4}G_4\{X_1, X_2\} = A_{3;4}\begin{pmatrix}X_{1,0}X_{2,0}\\ X_{1,1}X_{2,0}\\ X_{1,0}X_{2,1}\\ X_{1,1}X_{2,1}\end{pmatrix}$$
$$x_3=\begin{pmatrix}0.6 & 0.3 & 0.2 & 0.1\\ 0.4 & 0.7 & 0.8 & 0.9\end{pmatrix}\begin{pmatrix}\Pr\{X_{1,0}X_{2,0}\}\\ \Pr\{X_{1,1}X_{2,0}\}\\ \Pr\{X_{1,0}X_{2,1}\}\\ \Pr\{X_{1,1}X_{2,1}\}\end{pmatrix}.$$

The matrix expression and algorithm will be discussed in detail in the following sections.

It is easy to see that the output a-type parameters of a completely combined logic gate are just the corresponding CPT parameters. In other words, M-DUCG can also be as compact as possible. Another insight is that S-DUCG and M-DUCG are equivalent in the completely combined logic gate cases (e.g., Fig. 9 is equivalent to Fig. 4).

In summary of this section, DUCG has the similar statistic basis as other compact representation models have, and its parameters can be learned from statistic data. However, in some applications such as fault diagnoses for large and complex systems, the statistic fault samples are rare, and the compact parameters of DUCG can be easily specified by domain experts directly based on their knowledge about the specific cases.

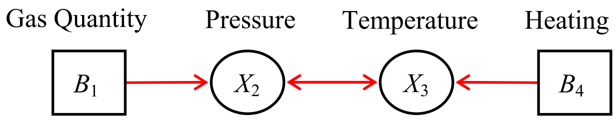


Fig. 10. DUCG of a gas container system.

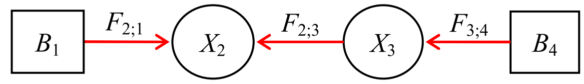


Fig. 11. Modified DUCG without DCG.

III. Brief Introduction to M-DUCG

In the following part of this paper, we will discuss only M-DUCG, because it can model both single-valued cases (with different meaning from S-DUCG) and multivalued cases, and is more flexible in applications. Fig. 10 is a simple example of M-DUCG (in what follows, M-DUCG may be abbreviated as DUCG because it is included in DUCG).

The physical meaning of the DUCG in Fig. 10 is: some chemical reaction occurs in a gas container at a certain pressure and temperature. According to physics, given the container volume, the higher the temperature (X_3) is, the higher the pressure (X_2) is; the higher the pressure is, the higher the temperature is; and vice versa. Meanwhile, the abnormal temperature can be caused by the overheating or less heating (B_4), and the abnormal pressure can be caused by the abnormal quantity of gas (B_1) pumped into or pumped out of the container. The query may be: what is the probability of overheating ($B_{4,2}$) conditioned on the evidence $E = X_{2,1}$ (low pressure) or $E = X_{2,2}X_{3,2}$ (high pressure and high temperature), i.e., $\Pr\{B_{4,2}|E\} =?$, or what is the probability of normal quantity of gas ($B_{1,0}$) and overheating ($B_{4,2}$) conditioned on the above E, i.e., $\Pr\{B_{1,0}B_{4,2}|E\} =?$, and so on. The b-type parameters $b_{ij} \equiv \Pr\{B_{ij}\}$ (prior probabilities of B-type events) are easy to be obtained, because they represent root causes in DUCG. The uncertainties of the causalities represented by the red directed arcs are quantified by the $\{a\text{-}, r\text{-}\}$ type parameters. The basic inference algorithm of DUCG is roughly composed of three steps: 1) simplify the original DUCG according to the observed evidence E by applying Rules 1–10 presented in [22], so that the possible hypotheses H_{kj} (e.g., $H_{4,2} = B_{4,2}$) are limited to only those (denoted as S_H, i.e., $H_{kj} \in S_H$) included in the simplified DUCG; 2) expand $H_{kj}E$ and E, respectively, as in the form of sum-of-products composed of independent events B_{ij} and $A_{nk;ij}$ associated with weighting factors $(r_{n;i}/r_n)$; and 3) calculate the posterior probability of H_{kj}: $\Pr\{H_{kj}|E\} = \Pr\{H_{kj}E\}/\Pr\{E\}$.

The accuracy of $\{a\text{-}, r\text{-}\}$ type parameters has only relative meaning, because: 1) the possible hypotheses $H_{kj} \in S_H$ have been found by step 1 and 2) according to the basic inference algorithms of DUCG: $\Pr\{H_{kj}|E\} = \Pr\{H_{kj}E\}/\Pr\{E\}$, the posterior probability of H_{kj} depends on the relative values of $\Pr\{H_{kj}E\}$ and $\Pr\{E\}$. It is seen that different parameters related to E influence $\Pr\{H_{kj}E\}$ and $\Pr\{E\}$ similarly. Only the parameters involved in H_{kj} can make significant difference and these parameters are usually easy to be obtained (H_{kj} is usually composed of B-type events in diagnostic problems and b-type parameters are easy to be obtained). Therefore, DUCG is robust, so that it is realistic for domain experts to specify DUCG's $\{a\text{-}, r\text{-}\}$ type parameters directly (using same criterion) based on their knowledge in cases without statistic data. In DUCG, what is important to the $\{a\text{-}, r\text{-}\}$ type parameter is zero or nonzero. Once it is nonzero, the relative value is meaningful, but not the absolute value.

It is interesting to note that there is a static logic cycle in Fig. 10: $X_2 \rightarrow X_3 \rightarrow X_2$ or $X_3 \rightarrow X_2 \rightarrow X_3$ represented by the bidirectional arc between X_2 and X_3. Actually, the bidirectional arc is composed of two directed arcs: $X_2 \rightarrow X_3$ and $X_3 \rightarrow X_2$. Such case can be called the directed cyclic graph (DCG). This DCG cannot be removed by applying dynamic BN, because the bidirectional causal functions are instant. In static cases (within a same time slice), it is well known that BN is defined on the directed acyclic graph (DAG). In contrast, DUCG does not have to be defined on DAG (the DCG structured S-DCUG case has been discussed in [18]), but for simplicity, the DUCG presented in [22] deals with only DAG cases. In this paper, we will continue to assume the DAG structure. The DCG cases are planned to be discussed in a following paper. For this sake, we revise Fig. 10 as Fig. 11 in which the DCG structure is removed.

In DUCG, the red directed arc can be written in text as $F_{n;i}$ indicating the directed arc from the parent variable indexed by i to the child variable indexed by n as shown in Fig. 11. Usually, the text $F_{n;i}$ are ignored in the graph as shown in Fig. 10. Therefore, in terms of text, the difference between Figs. 10 and 11 is that $F_{3;2}$ is removed in Fig. 11.

The inference step 1 (simplifying DUCG) has been discussed in [22] and will be demonstrated in Section VI. In what follows, we will discuss only the simplified DUCG, i.e., Fig. 11 will be taken as the simplified DUCG.

To perform inference of $\Pr\{H_{kj}|E\} = \Pr\{H_{kj}E\}/\Pr\{E\}$, we need to expand E and $H_{kj}E$ into the expressions composed of only $\{B\text{-}, A\text{-}\}$ type events and r-type parameters, respectively, which involves the expanding of X_{nk} and $X_{nk}X_{ij}$, and so on. It is noted that the DUCG methodology presented in [22] is only in terms of individual events and probabilities. For the example shown in Fig. 11, suppose B_1, X_2, X_3, and B_4 have three states each (indexed by 0, 1, and 2, respectively), X_2 can be expanded as (2) by applying (1) that is (31) in [22] in which V denotes parent variables

$$X_{nk} = \sum_i (r_{n;i}/r_n) \sum_j A_{nk;ij} V_{ij} = \sum_i \sum_j (r_{n;i}/r_n) A_{nk;ij} V_{ij} \tag{1}$$

$$\left.\begin{aligned}
X_{2,0} &= \sum_{j=0}^{2} (r_{2;1}/r_2) A_{2,0;1j} B_{1j} + \sum_{k=0}^{2} (r_{2;3}/r_2) A_{2,0;3k} X_{3k} \\
X_{2,1} &= \sum_{j=0}^{2} (r_{2;1}/r_2) A_{2,1;1j} B_{1j} + \sum_{k=0}^{2} (r_{2;3}/r_2) A_{2,1;3k} X_{3k} \\
X_{2,2} &= \sum_{j=0}^{2} (r_{2;1}/r_2) A_{2,2;1j} B_{1j} + \sum_{k=0}^{2} (r_{2;3}/r_2) A_{2,2;3k} X_{3k}
\end{aligned}\right\} \tag{2}$$

where $F_{nk;ij} \equiv (r_{n;i}/r_n)A_{nk;ij}$ is an element of $F_{n;i}$. Equation (1) is defined in M-DUCG ([22] for details) and is shown in Fig. 5. Through further applying (1) to expand X_{3k}, we have

$$\left.\begin{aligned} X_{2,0} &= \sum_{j=0}^{2}(r_{2;1}/r_2)A_{2,0;1j}B_{1j} + \sum_{k=0}^{2}(r_{2;3}/r_2)A_{2,0;3k} \\ &\quad \sum_{m=0}^{2}(r_{3;4}/r_3)A_{3k;4m}B_{4m} \\ X_{2,1} &= \sum_{j=0}^{2}(r_{2;1}/r_2)A_{2,1;1j}B_{1j} + \sum_{k=0}^{2}(r_{2;3}/r_2)A_{2,1;3k} \\ &\quad \sum_{m=0}^{2}(r_{3;4}/r_3)A_{3k;4m}B_{4m} \\ X_{2,2} &= \sum_{j=0}^{2}(r_{2;1}/r_2)A_{2,2;1j}B_{1j} + \sum_{k=0}^{2}(r_{2;3}/r_2)A_{2,2;3k} \\ &\quad \sum_{m=0}^{2}(r_{3;4}/r_3)A_{3k;4m}B_{4m} \end{aligned}\right\}. \tag{3}$$

Since the $\{B\text{-}, A\text{-}\}$ type events in any event product in (3) are independent of each other, according to (4) that is (32) in [22], we have (5) that is almost the same as (3) except that the uppercase letters become lowercase letters

$$x_{nk} = \sum_i (r_{n;i}/r_n)\sum_j a_{nk;ij}v_{ij} = \sum_i\sum_j (r_{n;i}/r_n)a_{nk;ij}v_{ij} \tag{4}$$

$$\left.\begin{aligned} x_{2,0} &= \sum_{j=0}^{2}(r_{2;1}/r_2)a_{2,0;1j}b_{1j} + \sum_{k=0}^{2}(r_{2;3}/r_2)a_{2,0;3k} \\ &\quad \sum_{m=0}^{2}(r_{3;4}/r_3)a_{3k;4m}b_{4m} \\ x_{2,1} &= \sum_{j=0}^{2}(r_{2;1}/r_2)a_{2,1;1j}b_{1j} + \sum_{k=0}^{2}(r_{2;3}/r_2)a_{2,1;3k} \\ &\quad \sum_{m=0}^{2}(r_{3;4}/r_3)a_{3k;4m}b_{4m} \\ x_{2,2} &= \sum_{j=0}^{2}(r_{2;1}/r_2)a_{2,2;1j}b_{1j} + \sum_{k=0}^{2}(r_{2;3}/r_2)a_{2,2;3k} \\ &\quad \sum_{m=0}^{2}(r_{3;4}/r_3)a_{3k;4m}b_{4m} \end{aligned}\right\}. \tag{5}$$

In DUCG, the lowercase letters denote the probabilities of the corresponding uppercase letter events, e.g., $x_{nk} \equiv \Pr\{X_{nk}\}$, $a_{nk;ij} \equiv \Pr\{A_{nk;ij}\}$, $b_{ij} \equiv \Pr\{B_{ij}\}$ and $v_{ij} \equiv \Pr\{V_{ij}\}$.

Equations (1) and (4) are the basic assumption of M-DUCG. Its physical meaning is: every parent variable (logic gate is also a parent) independently contributes a weighted probability distribution over the states of the child variable. The sum of the weighted probability distributions from all parent variables is the final probability distribution of the child variable. The state of the child variable is decided randomly according to this final probability distribution. Obviously, this assumption is easy to be understood and accepted by domain experts, because it matches the knowledge structure in the mind of domain experts.

Equations (2) and (3) can be further briefly written as (6) and (7)

$$\left.\begin{aligned} X_{2,0} &= \sum_{j=0}^{2}F_{2,0;1j}B_{1j} + \sum_{k=0}^{2}F_{2,0;3k}X_{3k} \\ X_{2,1} &= \sum_{j=0}^{2}F_{2,1;1j}B_{1j} + \sum_{k=0}^{2}F_{2,1;3k}X_{3k} \\ X_{2,2} &= \sum_{j=0}^{2}F_{2,2;1j}B_{1j} + \sum_{k=0}^{2}F_{2,2;3k}X_{3k} \end{aligned}\right\} \tag{6}$$

$$\left.\begin{aligned} X_{2,0} &= \sum_{j=0}^{2}F_{2,0;1j}B_{1j} + \sum_{k=0}^{2}F_{2,0;3k}\sum_{m=0}^{2}F_{3k;4m}B_{4m} \\ X_{2,1} &= \sum_{j=0}^{2}F_{2,1;1j}B_{1j} + \sum_{k=0}^{2}F_{2,1;3k}\sum_{m=0}^{2}F_{3k;4m}B_{4m} \\ X_{2,2} &= \sum_{j=0}^{2}F_{2,2;1j}B_{1j} + \sum_{k=0}^{2}F_{2,2;3k}\sum_{m=0}^{2}F_{3k;4m}B_{4m} \end{aligned}\right\}. \tag{7}$$

Once more, $F_{nk;ij} \equiv (r_{n;i}/r_n)A_{nk;ij}$. Similarly, X_3 can be expanded as (8)

$$\left.\begin{aligned} X_{3,0} &= \sum_{m=0}^{2}F_{3,0;4m}B_{4m} \\ X_{3,1} &= \sum_{m=0}^{2}F_{3,1;4m}B_{4m} \\ X_{3,2} &= \sum_{m=0}^{2}F_{3,2;4m}B_{4m} \end{aligned}\right\}. \tag{8}$$

Correspondingly, in terms of probabilities, we have

$$\left.\begin{aligned} x_{2,0} &= \sum_{j=0}^{2}f_{2,0;1j}b_{1j} + \sum_{k=0}^{2}f_{2,0;3k}\sum_{m=0}^{2}f_{3k;4m}b_{4m} \\ x_{2,1} &= \sum_{j=0}^{2}f_{2,1;1j}b_{1j} + \sum_{k=0}^{2}f_{2,1;3k}\sum_{m=0}^{2}f_{3k;4m}b_{4m} \\ x_{2,2} &= \sum_{j=0}^{2}f_{2,2;1j}b_{1j} + \sum_{k=0}^{2}f_{2,2;3k}\sum_{m=0}^{2}f_{3k;4m}b_{4m} \end{aligned}\right\} \tag{9}$$

$$\left.\begin{aligned} x_{3,0} &= \sum_{m=0}^{2}f_{3,0;4m}b_{4m} \\ x_{3,1} &= \sum_{m=0}^{2}f_{3,1;4m}b_{4m} \\ x_{3,2} &= \sum_{m=0}^{2}f_{3,2;4m}b_{4m} \end{aligned}\right\} \tag{10}$$

where $f_{nk;ij} \equiv (r_{n;i}/r_n)a_{nk;ij}$. Suppose the evidence is $E = E_1E_2 = X_{2,2}X_{3,2}$, where $E_1 = X_{2,2}$ and $E_2 = X_{3,2}$, and $H_{kj} = B_{4,2}$, we have

$$\Pr\{H_{kj}|E\} = \Pr\{B_{4,2}|X_{2,2}X_{3,2}\} = \frac{\Pr\{B_{4,2}X_{2,2}X_{3,2}\}}{\Pr\{X_{2,2}X_{3,2}\}}. \tag{11}$$

Then, to calculate the probabilities in (11), we need to expand $X_{2,2}X_{3,2}$ and $B_{4,2}X_{2,2}X_{3,2}$, respectively, into the sum-of-products composed of only independent B- and A-type events with r-type parameters. Through applying (7) and (8), $X_{2,2}X_{3,2}$ can be expanded as

$$\begin{aligned}
X_{2,2}X_{3,2} &= \left(\sum_{j=0}^{2} F_{2,2;1j}B_{1j} + \sum_{k=0}^{2} F_{2,2;3k}\sum_{m=0}^{2} F_{3k;4m}B_{4m}\right) \\
&\quad \times \left(\sum_{m=0}^{2} F_{3,2;4m}B_{4m}\right) \\
&= \left(\sum_{j=0}^{2} F_{2,2;1j}B_{1j}\right)\left(\sum_{m=0}^{2} F_{3,2;4m}B_{4m}\right) \\
&\quad + \left(\sum_{k=0}^{2} F_{2,2;3k}\sum_{m=0}^{2} F_{3k;4m}B_{4m}\right)\left(\sum_{m=0}^{2} F_{3,2;4m}B_{4m}\right) \\
&= \left(\sum_{j=0}^{2} F_{2,2;1j}B_{1j}\right)\left(\sum_{m=0}^{2} F_{3,2;4m}B_{4m}\right) \\
&\quad + \left(F_{2,2;3,2}\sum_{m=0}^{2} F_{3,2;4m}B_{4m}\right)\left(\sum_{m=0}^{2} F_{3,2;4m}B_{4m}\right) \\
&= \left(\sum_{j=0}^{2} F_{2,2;1j}B_{1j}\right)\left(\sum_{m=0}^{2} F_{3,2;4m}B_{4m}\right) \\
&\quad + F_{2,2;3,2}\sum_{m=0}^{2}\left(F_{3,2;4m}B_{4m}\right)^2 \\
&= \left(\sum_{j=0}^{2} (r_{2;1}/r_2)A_{2,2;1j}B_{1j}\right)\left(\sum_{m=0}^{2} (r_{3;4}/r_3)A_{3,2;4m}B_{4m}\right) \\
&\quad + \left(r_{2;3}/r_2\right)A_{2,2;3,2}\sum_{m=0}^{2}\left(r_{3;4}/r_3\right)^2 A_{3,2;4m}B_{4m} \\
&= \left(\sum_{j=0}^{2} (r_{2;1}/r_2)A_{2,2;1j}B_{1j}\right)\left(\sum_{m=0}^{2} A_{3,2;4m}B_{4m}\right) \\
&\quad + (r_{2;3}/r_2)A_{2,2;3,2}\sum_{m=0}^{2} A_{3,2;4m}B_{4m} \\
&= \left((r_{2;1}/r_2)\sum_{j=0}^{2} A_{2,2;1j}B_{1j} + (r_{2;3}/r_2)A_{2,2;3,2}\right) \\
&\quad \times \sum_{m=0}^{2} A_{3,2;4m}B_{4m}. \qquad (12)
\end{aligned}$$

In which, the third and fourth equators are because of applying Rule 12 presented in [22] ($F_{nk;ij}F_{nk';ij} = 0$ and $F_{nk;ij}F_{nk;ij'} = 0$), the fifth equator is because of applying Rule 11 presented in [22] ($(V_{ij})^y = V_{ij}$ and $V_{ij}V_{ij'} = 0$), the sixth equator is because of applying Rule 12 presented in [22] ($(F_{nk;ij})^y = (r_{n;i}/r_n)^y A_{nk;ij}$), and also because X_3 has only one parent variable leading to $(r_{3;4}/r_3) = (r_{3;4}/r_{3;4}) = 1$.

Furthermore

$$\begin{aligned}
B_{4,2}X_{2,2}X_{3,2} &= B_{4,2}\Big((r_{2;1}/r_2)\sum_{j=0}^{2} A_{2,2;1j}B_{1j} \\
&\quad + (r_{2;3}/r_2)A_{2,2;3,2}\Big)\sum_{m=0}^{2} A_{3,2;4m}B_{4m} \\
&= \Big((r_{2;1}/r_2)\sum_{j=0}^{2} A_{2,2;1j}B_{1j} \\
&\quad + (r_{2;3}/r_2)A_{2,2;3,2}\Big)A_{3,2;4,2}B_{4,2}. \qquad (13)
\end{aligned}$$

In which, the second equator is because of applying Rule 11 presented in [22] ($V_{ij}V_{ij'} = 0$, given $j \neq j'$).

Finally, we have (14) as shown at the top of the next page.

It is easy to see that to expand (12) and (13) as being composed of only independent events B_{ij} and $A_{nk;ij}$ associated with $(r_{n;i}/r_n)$ is redundant, and a lot of event absorption and exclusion operations are involved (by applying Rules 11 and 12 presented in [22]). It is obvious that as the scale and complexity of problem increase, the expanding becomes more and more difficult and the result becomes more and more nonunderstandable. Therefore, the matrix expression and algorithm is needed.

IV. Weighted Event Matrix Expression and Algorithm

In M-DUCG, it is defined that $A_{nk;ij}$ is associated with a weighting factor $(r_{n;i}/r_n)$. Another thing is that $A_{nk;ij}$ are elements of matrix $A_{n;i}$, or $F_{nk;ij} = (r_{n;i}/r_n)A_{nk;ij}$ are elements of matrix $F_{n;i}$. The expanding of E and $H_{kj}E$ for calculating $\Pr\{H_{kj}|E\}$ needs a new algorithm dealing with the operation of weighted events and event matrices. This section is aiming to provide this new algorithm. The basic ideas are: 1) apply the ordinary set theory for the event operation; 2) apply the ordinary algebra operation for the weighting factor operation; 3) combine them together; and 4) make the event matrix expression as the probability matrix expression.

To be general, we need to summarize the notations of M-DUCG as follows: the functional event $A_{nk;ij}$ represents the individual random physical mechanism that the parent event V_{ij} causes the child event X_{nk}, where n indexes the child variable X_n, k indexes the state of X_n, i indexes the parent variable V_i, j indexes the state of V_i, $V \in \{X,B,G,D\}$; X denotes the event/variable that has at least one parent event/variable and can be a parent event/variable of other events/variables; B denotes the basic event/variable that should be a parent event/variable but does not have its own parent event/variable (in other words, B represents the root cause event/variable); G denotes the logic gate event/variable specified with the logic gate specification (LGS) as shown in Fig. 12, in which the logic expressions are composed of input/parent events.

G-type event/variable must be a parent of other events/variables (including other logic gates); D denotes the default parent event of an X-type event/variable, which can also be viewed as the variable with only one state;

$$
\begin{aligned}
\Pr\{B_{4,2}|E\} &= \frac{\Pr\{B_{4,2}X_{2,2}X_{3,2}\}}{\Pr\{X_{2,2}X_{3,2}\}} \\
&= \frac{\Pr\left\{\left((r_{2;1}/r_2)\sum_{j=0}^{2} A_{2,2;1j}B_{1j} + (r_{2;3}/r_2)A_{2,2;3,2}\right)A_{3,2;4,2}B_{4,2}\right\}}{\Pr\left\{\left((r_{2;1}/r_2)\sum_{j=0}^{2} A_{2,2;1j}B_{1j} + (r_{2;3}/r_2)A_{2,2;3,2}\right)\sum_{m=0}^{2} A_{3,2;4m}B_{4m}\right\}} \\
&= \frac{\left((r_{2;1}/r_2)\sum_{j=0}^{2} a_{2,2;1j}b_{1j} + (r_{2;3}/r_2)a_{2,2;3,2}\right)a_{3,2;4,2}b_{4,2}}{\left((r_{2;1}/r_2)\sum_{j=0}^{2} a_{2,2;1j}b_{1j} + (r_{2;3}/r_2)a_{2,2;3,2}\right)\sum_{m=0}^{2} a_{3,2;4m}b_{4m}} \\
&= \frac{a_{3,2;4,2}b_{4,2}}{\sum_{m=0}^{2} a_{3,2;4m}b_{4m}}.
\end{aligned}
\tag{14}
$$

j	G_{ij}
0	Expression 0
1	Expression 1
m	Expression m

Fig. 12. Logic gate specification (LGS_i).

$F_{nk;ij} \equiv (r_{n;i}/r_n)A_{nk;ij}$ and can be viewed as a weighted event between child event X_{nk} and parent event V_{ij}.

Without losing generality, for convenience, we assume $k \in \{0,1,\ldots,K\}$ and $j \in \{0,1,\ldots,J\}$ in $F_{nk;ij}$, $f_{nk;ij}$, $A_{nk;ij}$, $a_{nk;ij}$, X_{nk}, x_{nk}, V_{ij}, v_{ij}, and so on, while keeping in mind that K and J may be different for different variables. Thus, in terms of matrices, we have

$$
\begin{aligned}
X_n &\equiv \left(X_{n0}\ X_{n1}\ \cdots\ X_{nk}\ \cdots\ X_{nK}\right)^T \\
x_n &\equiv \left(x_{n0}\ x_{n1}\ \cdots\ x_{nk}\ \cdots\ x_{nK}\right)^T \\
V_i &\equiv \left(V_{i0}\ V_{i1}\ \cdots\ V_{ij}\ \cdots\ V_{iJ}\right)^T \\
v_i &\equiv \left(v_{i0}\ v_{i1}\ \cdots\ v_{ij}\ \cdots\ v_{iJ}\right)^T \\
F_{n;i} &\equiv \begin{pmatrix}
F_{n0;i0} & F_{n10i1} & \cdots & F_{n0;ij} & \cdots & F_{n0;iJ} \\
F_{n1;i0} & F_{n1;i1} & \cdots & F_{n1;ij} & \cdots & F_{n1;iJ} \\
\vdots & \vdots & & \vdots & & \vdots \\
F_{nk;i0} & F_{nk;i1} & & F_{nk;ij} & & F_{nk;iJ} \\
\vdots & \vdots & & \vdots & & \vdots \\
F_{nK;i0} & F_{nK;i1} & \cdots & F_{nK;ij} & \cdots & F_{nK;iJ}
\end{pmatrix} \\
&= (r_{n;i}/r_n)A_{n;i} \\
A_{n;i} &\equiv \begin{pmatrix}
A_{n0;i0} & A_{n10i1} & \cdots & A_{n0;ij} & \cdots & A_{n0;iJ} \\
A_{n1;i0} & A_{n1;i1} & \cdots & A_{n1;ij} & \cdots & A_{n1;iJ} \\
\vdots & \vdots & & \vdots & & \vdots \\
A_{nk;i0} & A_{nk;i1} & & A_{nk;ij} & & A_{nk;iJ} \\
\vdots & \vdots & & \vdots & & \vdots \\
A_{nK;i0} & A_{nK;i1} & \cdots & A_{nK;ij} & \cdots & A_{nK;iJ}
\end{pmatrix} \\
f_{n;i} &\equiv \begin{pmatrix}
f_{n0;i0} & f_{n10i1} & \cdots & f_{n0;ij} & \cdots & f_{n0;iJ} \\
f_{n1;i0} & f_{n1;i1} & \cdots & f_{n1;ij} & \cdots & f_{n1;iJ} \\
\vdots & \vdots & & \vdots & & \vdots \\
f_{nk;i0} & f_{nk;i1} & & f_{nk;ij} & & f_{nk;iJ} \\
\vdots & \vdots & & \vdots & & \vdots \\
f_{nK;i0} & f_{nK;i1} & \cdots & f_{nK;ij} & \cdots & f_{nK;iJ}
\end{pmatrix} \\
&= (r_{n;i}/r_n)a_{n;i} \\
a_{n;i} &\equiv \begin{pmatrix}
a_{n0;i0} & a_{n0;i1} & \cdots & a_{n0;ij} & \cdots & a_{n0;iJ} \\
a_{n1;i0} & a_{n1;i1} & \cdots & a_{n1;ij} & \cdots & a_{n1;iJ} \\
\vdots & \vdots & & \vdots & & \vdots \\
a_{nk;i0} & a_{nk;i1} & & a_{nk;ij} & & a_{nk;iJ} \\
\vdots & \vdots & & \vdots & & \vdots \\
a_{nK;i0} & a_{nK;i1} & \cdots & a_{nK;ij} & \cdots & a_{nK;iJ}
\end{pmatrix} \\
A_{nk;i} &\equiv \left(A_{nk;i0}\ A_{nk;i1}\ \cdots\ A_{nk;ij}\ \cdots\ A_{nk;iJ}\right) \\
a_{nk;i} &\equiv \left(a_{nk;i0}\ a_{nk;i1}\ \cdots\ a_{nk;ij}\ \cdots\ a_{nk;iJ}\right) \\
F_{nk;i} &\equiv \left(F_{nk;i0}\ F_{nk;i1}\ \cdots\ F_{nk;ij}\ \cdots\ F_{nk;iJ}\right) \\
&= (r_{n;i}/r_n)A_{nk;i} \\
f_{nk;i} &\equiv \left(f_{nk;i0}\ f_{nk;i1}\ \cdots\ f_{nk;ij}\ \cdots\ f_{nk;iJ}\right) \\
&= (r_{n;i}/r_n)a_{nk;i} \\
P_{nk;i} &\equiv \left(P_{nk;i0}\ P_{nk;i1}\ \cdots\ P_{nk;ij}\ \cdots\ P_{nk;iJ}\right) \\
p_{nk;i} &\equiv \left(p_{nk;i0}\ p_{nk;i1}\ \cdots\ p_{nk;ij}\ \cdots\ p_{nk;iJ}\right) \\
A_{n;ij} &\equiv \left(A_{n0;i,j}\ A_{n1;i,j}\ \cdots\ A_{nk;i,j}\ \cdots\ A_{nK;i,j}\right) \\
a_{n;ij} &\equiv \left(a_{n0;ij}\ a_{n1;ij}\ \cdots\ a_{nk;ij}\ \cdots\ a_{nK;ij}\right) \\
F_{n;ij} &\equiv \left(F_{n1;ij}\ F_{n2;ij}\ \cdots\ F_{nk;ij}\ \cdots\ F_{nK;ij}\right)^T \\
&= (r_{n;i}/r_n)A_{n;ij} \\
f_{n;ij} &\equiv \left(f_{n0;ij}\ f_{n1;ij}\ \cdots\ f_{nk;ij}\ \cdots\ f_{nK;ij}\right)^T \\
&= (r_{n;i}/r_n)a_{n;ij} \\
V_{i\bullet} &\in \{V_i, V_{ij}\} \\
A_{n\bullet;i\bullet} &\in \{A_{n;i}, A_{nk;i}, A_{n;ij}A_{nk;ij}\}
\end{aligned}
$$

$$
\begin{aligned}
A_{n\bullet;i} &\in \{A_{n;i}, A_{nk;i}\} \\
A_{n;i\bullet} &\in \{A_{n;i}, A_{n;ij}\} \\
F_{n\bullet;i\bullet} &\in \{F_{n;i}, F_{nk;i}, F_{n;ij}, F_{nk;ij}\} \\
F_{n\bullet;i} &\in \{F_{n;i}, F_{nk;i}\} \\
F_{n;i\bullet} &\in \{F_{n;i}, F_{n;ij}\}
\end{aligned}
$$

where "•" in the subscripts indicates with or without the state subscript k or j. $P_{nk;i}$ and $p_{nk;i}$ are applicable in only single-valued cases where k indexes the true state of variable X_n ($k = 1$ in this paper).

It is noted that the subscripts are used to distinguish the events, variables, and matrices. In general, for $V \in \{X, B, G\}$, when there is only one subscript i, we know that V_i represents a variable or a vector; when there are two subscripts i and j, we know that V_{ij} is an event or a value of V_i. D_n is a default event as defined in DUCG ([22] for details). For $\{A\text{-}, P\text{-}\}$ type event variables, when there are four subscripts "nk;ij", $A_{nk;ij}$, or $P_{nk;ij}$ is an event; when the subscripts are <4, $A_{n;i}$, $A_{nk;i}$, $A_{n;ij}$, and $P_{nk;i}$ represent event matrices/vectors. We use $A_{nk;nD}$ or $P_{nk;nD}$ in the case when D_n is a parent event of X_n or X_{nk}. Also, j in $A_{nk;ij}$ or $P_{nk;ij}$ may represent D.

With the above notations, (1) and (4) can be compactly expressed as

$$X_{nk} = \sum_i F_{nk;i} V_i \tag{15}$$

$$x_{nk} = \sum_i f_{nk;i} v_i \tag{16}$$

or more compactly

$$X_n = \sum_i F_{n;i} V_i \tag{17}$$

$$x_n = \sum_i f_{n;i} v_i. \tag{18}$$

Furthermore, conditioned on the state combination of parent variables $\cap_i V_{ij}$, X_{nk}, and its conditional probability can be expressed as (19) and (20), respectively

$$X_{nk} | \bigcap_i V_{ij} = \sum_i F_{nk;ij} \tag{19}$$

$$\Pr\{X_{nk} | \bigcap_i V_{ij}\} = \sum_i f_{nk;ij} \tag{20}$$

where $X_{nk} | \cap_i V_{ij}$ is defined as a special event that X_{nk} is conditioned on $\cap_i V_{ij}$. Equation (20) expresses the CPT of X_n in terms of M-DUCG.

It should be noted that different order of events in an event product has no difference. However, in terms of matrices, the order of matrices cannot be arbitrary. For example, in terms of events, $A_{nk;ij} V_{ij} = V_{ij} A_{nk;ij}$. But in terms of matrices, $A_{n;i} V_i \neq V_i A_{n;i}$. Sometimes, different order of matrices means same thing. For example, $A_{n;i} V_i A_{h;g} V_g = A_{h;g} V_g A_{n;i} V_i$.

In principle, the order of matrices in a same causality chain cannot be changed. The subscripts of matrices must be in the order "child n; parent i," "child i; parent h," "child h; parent g," and so on. But for different chains in a same product, the order of these chains can be any. For the above example, $A_{n;i} V_i$ and $A_{h;g} V_g$ are in two causality chains and therefore, $A_{n;i} V_i A_{h;g} V_g$ equals to $A_{h;g} V_g A_{n;i} V_i$.

In [22], Rules 11–14 for expanding $H_{kj} E$ and E have been presented and proved (for convenience, the rules applied in DUCG are indexed sequentially in the serial papers). They are in the event form. For convenience, they are listed below, and are extended in the corresponding corollaries in terms of matrices.

Rule 11: Given $V \in \{B, X, G, D\}$, $j \neq j'$, and integer $y \geq 2$, we have $(V_{ij})^y = V_{ij}$ and $V_{ij} V_{ij'} = 0$.

Corollary 11: $V_{ij} V_i = V_i V_{ij} = V_{ij}$, $(V_i)^y = V_i$.

Proof: According to Rule 11, $V_{ij} V_i = V_i V_{ij} = (0 \ldots 0\ V_{ij}\ 0 \ldots 0)^T$, where "0" denotes null. By definition, $(0 \ldots 0\ V_{ij}\ 0 \ldots 0)^T$ implies V_{ij}. Therefore, $V_{ij} V_i = V_i V_{ij} = (0 \ldots 0\ V_{ij}\ 0 \ldots 0)^T = V_{ij}$. In addition, Rule 11 implies that for a same variable V_i, only one state of it can be true at a same time, i.e., in $(V_i)^y$, only $(V_{ij})^y = V_{ij}$ and all other state combinations are null. By definition, such case is represented by V_i, i.e., $(V_i)^y = V_i$.

Rule 12: Given integer $y \geq 2$, $k \neq k'$, and $j \neq j'$, we have $(F_{nk;ij})^y = (r_{n;i}/r_n)^y A_{nk;ij}$, $F_{nk;ij} F_{nk';ij} = 0$, $F_{nk;ij} F_{nk;ij'} = 0$ and $F_{nk;ij} F_{nk';ij'} = 0$.

Corollary 12: Given $y \geq 2$, $n \neq n'$, $k \neq k'$, and $j \neq j'$, we have

$$(F_{n;i})^y = (r_{n;i}/r_n)^y A_{n;i} \tag{21}$$

$$(F_{nk;i})^y = (r_{n;i}/r_n)^y A_{nk;i} \tag{22}$$

$$(F_{n;ij})^y = (r_{n;i}/r_n)^y A_{n;ij} \tag{23}$$

$$F_{nk;ij} \prod_{h=1}^{y-1} F_{n\bullet;i\bullet} = (r_{n;i}/r_n)^y A_{nk;ij} \tag{24}$$

$$F_{nk;i} \prod_{h=1}^{y-1} F_{n\bullet;i} = (r_{n;i}/r_n)^y A_{nk;i} \tag{25}$$

$$F_{n;ij} \prod_{h=1}^{y-1} F_{n;i\bullet} = (r_{n;i}/r_n)^y A_{n;ij} \tag{26}$$

$$
\begin{aligned}
F_{n\bullet;i} F_{ij;h\bullet} &= F_{n\bullet;ij} F_{i;h\bullet} = F_{n\bullet;ij} F_{ij;h\bullet} \\
&= (r_{n;i}/r_n)(r_{i;h}/r_i) A_{n\bullet;ij} A_{ij;h\bullet}
\end{aligned} \tag{27}
$$

$$
\left.
\begin{aligned}
A_{nk;i\bullet} A_{n;i'\bullet} &= A_{nk;i\bullet} A_{nk;i'\bullet} \\
A_{n\bullet;ij} A_{n'\bullet;i} &= A_{n\bullet;ij} A_{n'\bullet;ij} \\
A_{n\bullet;i} V_i V_{ij} &= A_{n\bullet;ij} V_{ij} \\
A_{nk;i\bullet} A_{nk';i'\bullet} &= 0 \\
A_{n\bullet;ij} A_{n'\bullet;ij'} &= 0
\end{aligned}
\right\}. \tag{28}
$$

Proof: See Appendix A.

Rule 13: Suppose S_m denote the variable index set m, $m \in \{1, 2, \ldots, M\}$, and $S_1 \subseteq S_m$, then

$$\sum_{m=1}^{M} \prod_{i \in S_m} F_{nk;ij_i} V_{ij_i} = \left(\sum_{m=1}^{M} \prod_{i \in S_m} (r_{n;i}/r_n) \right) \prod_{i \in S_1} A_{nk;ij_i} V_{ij_i}.$$

Corollary 13: Suppose $m \in \{1,2,\ldots,M\}$ and

$$\prod_{i \in S_1} A_{n\bullet;i\bullet} V_{i\bullet} \supseteq \prod_{i \in S_m, m \neq 1} A_{n\bullet;i\bullet} V_{i\bullet}$$

then

$$\sum_{m=1}^{M}\prod_{i\in S_m} F_{n\bullet;i\bullet}V_{i\bullet} = \left(\sum_{m=1}^{M}\prod_{i\in S_m}(r_{n;i}/r_n)\right)\prod_{i\in S_1} A_{n\bullet;i\bullet}V_{i\bullet}. \tag{29}$$

Proof: See Appendix B.

Rule 14: Given $j = j_i$, $F_{nk;ij}V_{ij}\left(\sum_{i'} F_{nk;i'j_{i'}}V_{i'j_{i'}}\right) = F_{nk;ij}V_{ij}$.

Corollary 14: Given i is one of i'

$$F_{nk;ij}V_{ij}\sum_{i'} F_{n;i'\bullet}V_{i'\bullet} = F_{nk;ij}V_{ij} \tag{30}$$

$$F_{nk;i}V_i\sum_{i'} F_{n\bullet;i'}V_{i'} = F_{nk;i}V_{\mathrm{i}} \tag{31}$$

$$F_{n;ij}V_{ij}\sum_{i'} F_{n;i'\bullet}V_{i'\bullet} = F_{n;ij}V_{ij} \tag{32}$$

$$F_{n;i}V_i\sum_{i'} F_{n;i'}V_{i'} = F_{n;i}V_i. \tag{33}$$

Proof: See Appendix C.

By applying Corollary 14, for $y \geq 2$, we can prove $(X_{nk})^y = \left(\sum_i F_{nk;ij}V_{ij}\right)^y = \sum_i F_{nk;ij}V_{ij} = X_{nk}$.

Proof: See Appendix D.

Using the above expressions and rules/Corollaries, (3) and (7) can be written as

$$X_2 = F_{2;1}B_1 + F_{2;3}F_{3;4}B_4 \tag{34}$$

$$X_3 = F_{3;4}B_4 \tag{35}$$

and furthermore

$$\begin{aligned} X_{2,2} &= F_{2,2;1}B_1 + F_{2,2;3}F_{3;4}B_4 = (r_{2;1}/r_2)A_{2,2;1}B_1 \\ &\quad +(r_{2;3}/r_2)A_{2,2;3}(r_{3;4}/r_3)A_{3;4}B_4 \end{aligned} \tag{36}$$

$$X_{3,2} = F_{3,2;4}B_4 = (r_{3;4}/r_3)A_{3,2;4}B_4 \tag{37}$$

$$\begin{aligned} &X_{2,2}X_{3,2} \\ &\quad= \left((r_{2;1}/r_2)A_{2,2;1}B_1 + (r_{2;3}/r_2)A_{2,2;3}(r_{3;4}/r_3)A_{3;4}B_4\right) \\ &\qquad \left((r_{3;4}/r_3)A_{3,2;4}B_4\right) \\ &\quad= (r_{2;1}/r_2)A_{2,2;1}B_1(r_{3;4}/r_3)A_{3,2;4}B_4 \\ &\qquad +(r_{2;3}/r_2)A_{2,2;3}(r_{3;4}/r_3)A_{3;4}B_4(r_{3;4}/r_3)A_{3,2;4}B_4 \\ &\quad= (r_{2;1}/r_2)(r_{3;4}/r_3)A_{2,2;1}B_1A_{3,2;4}B_4 \\ &\qquad +(r_{2;3}/r_2)(r_{3;4}/r_3)(r_{3;4}/r_3)A_{2,2;3,2}A_{3,2;4}B_4 \\ &\quad= \left((r_{2;1}/r_2)A_{2,2;1}B_1 + (r_{2;3}/r_2)A_{2,2;3,2}\right)A_{3,2;4}B_4. \end{aligned} \tag{38}$$

The third equator is because of applying Corollaries 11 and 12, where $(r_{3;4}/r_3) = 1$ as mentioned in Section II. Similarly

$$\begin{aligned} &B_{4,2}X_{2,2}X_{3,2} \\ &\quad= B_{4,2}\left(\left((r_{2;1}/r_2)A_{2,2;1}B_1 + (r_{2;3}/r_2)A_{2,2;3,2}\right)A_{3,2;4}B_4\right) \\ &\quad= \left((r_{2;1}/r_2)A_{2,2;1}B_1 + (r_{2;3}/r_2)A_{2,2;3,2}\right)A_{3,2;4,2}B_{4,2}. \end{aligned} \tag{39}$$

The second equator is because of applying Corollary 11. Correspondingly

$$\begin{aligned} \Pr\{X_{2,2}X_{3,2}\} &= \left((r_{2;1}/r_2)a_{2,2;1}b_1\right. \\ &\quad \left.+(r_{2;3}/r_2)a_{2,2;3,2}\right)a_{3,2;4}b_4 \end{aligned} \tag{40}$$

$$\begin{aligned} \Pr\{B_{4,2}X_{2,2}X_{3,2}\} &= \left((r_{2;1}/r_2)a_{2,2;1}b_1\right. \\ &\quad \left.+(r_{2;3}/r_2)a_{2,2;3,2}\right)a_{3,2;4,2}b_{4,2}. \end{aligned} \tag{41}$$

Finally, we have

$$\begin{aligned} &\Pr\{B_{4,2}|X_{2,2}X_{3,2}\} \\ &\quad= \frac{\Pr\{B_{4,2}X_{2,2}X_{3,2}\}}{\Pr\{X_{2,2}X_{3,2}\}} \\ &\quad= \frac{\left((r_{2;1}/r_2)a_{2,2;1}b_1 + (r_{2;3}/r_2)a_{2,2;3,2}\right)a_{3,2;4,2}b_{4,2}}{\left((r_{2;1}/r_2)a_{2,2;1}b_1 + (r_{2;3}/r_2)a_{2,2;3,2}\right)a_{3,2;4}b_4} \\ &\quad= \frac{a_{3,2;4,2}b_{4,2}}{a_{3,2;4}b_4}. \end{aligned} \tag{42}$$

It is easy to see that to obtain (42) is much easier than to obtain (14). Suppose

$r_{2;1} = 0.5$, $r_{2;3} = 1$, $r_2 = r_{2;1} + r_{2;3} = 0.5+1=1.5$

$$b_1 = \begin{pmatrix}0.8 & 0.1 & 0.1\end{pmatrix}^T, \quad b_4 = \begin{pmatrix}0.7 & 0.1 & 0.2\end{pmatrix}^T$$

$$a_{2;1} = \begin{pmatrix}1 & 0.3 & 0.2\\ 0 & 0.7 & 0\\ 0 & 0 & 0.8\end{pmatrix}, \quad a_{2;3} = \begin{pmatrix}1 & 0.2 & 0.1\\ 0 & 0.8 & 0\\ 0 & 0 & 0.9\end{pmatrix}$$

$$a_{3;4} = \begin{pmatrix}1 & 0.2 & 0.3\\ 0 & 0.8 & 0\\ 0 & 0 & 0.7\end{pmatrix}.$$

According to (14), we have

$$\begin{aligned} \Pr\{B_{4,2}|E\} &= \frac{a_{3,2;4,2}b_{4,2}}{\sum_{m=0}^{2} a_{3,2;4m}b_{4m}} \\ &= \frac{0.7\times 0.2}{0\times 0.7 + 0\times 0.1 + 0.7\times 0.2} = 1. \end{aligned} \tag{43}$$

In comparison, according to (42), we have

$$\begin{aligned} \Pr\{B_{4,2}|E\} &= \frac{a_{3,2;4,2}b_{4,2}}{a_{3,2;4}b_4} \\ &= \frac{0.7\times 0.2}{\begin{pmatrix}0 & 0 & 0.7\end{pmatrix}\begin{pmatrix}0.7\\ 0.1\\ 0.2\end{pmatrix}} = 1. \end{aligned} \tag{44}$$

It is seen that although the expressions of (43) and (44) are different, their results are same. It is easy to understand $\Pr\{B_{4,2}|X_{2,2}X_{3,2}\} = 1$, because $B_{4,2}$ is necessary to cause $X_{2,2}$ and $X_{3,2}$ simultaneously. If our query is $\Pr\{B_{1,2}|X_{2,2}X_{3,2}\} =?$, (39) is changed as

$$\begin{aligned} &B_{1,2}X_{2,2}X_{3,2} \\ &\quad= B_{1,2}\left((r_{2;1}/r_2)A_{2,2;1}B_1 + (r_{2;3}/r_2)A_{2,2;3,2}\right)A_{3,2;4}B_4 \\ &\quad= \left((r_{2;1}/r_2)A_{2,2;1,2}B_{1,2} + B_{1,2}(r_{2;3}/r_2)A_{2,2;3,2}\right)A_{3,2;4}B_4. \end{aligned} \tag{45}$$

Corresponding to (41), we have

$$\begin{aligned} &\Pr\{B_{1,2}X_{2,2}X_{3,2}\} \\ &\quad= \left((r_{2;1}/r_2)a_{2,2;1,2}b_{1,2} + b_{1,2}(r_{2;3}/r_2)a_{2,2;3,2}\right)a_{3,2;4}b_4. \end{aligned} \tag{46}$$

Combining (40) and (46), we have

$$
\begin{aligned}
&\Pr\{B_{1,2}|X_{2,2}X_{3,2}\}\\
&\quad= \frac{\left((r_{2;1}/r_2)a_{2,2;1,2}b_{1,2} + b_{1,2}(r_{2;3}/r_2)a_{2,2;3,2}\right) a_{3,2;4}b_4}{\left((r_{2;1}/r_2)a_{2,2;1}b_1 + (r_{2;3}/r_2)a_{2,2;3,2}\right) a_{3,2;4}b_4}\\
&\quad= \frac{\left((r_{2;1}/r_2)a_{2,2;1,2} + (r_{2;3}/r_2)a_{2,2;3,2}\right) b_{1,2}}{(r_{2;1}/r_2)a_{2,2;1}b_1 + (r_{2;3}/r_2)a_{2,2;3,2}}\\
&\quad= \frac{((0.5/1.5)0.8 + (1/1.5)0.9)\,0.1}{(0.5/1.5)\left(0\ 0\ 0.8\right)\begin{pmatrix}0.8\\0.1\\0.1\end{pmatrix} + (1/1.5)0.9}\\
&\quad= 0.138.
\end{aligned}
\tag{47}
$$

This result is reasonable, because $B_{1,2}$ is not necessary to cause $X_{2,2}$ and $X_{3,2}$ simultaneously. $B_{4,2}$ must exist; otherwise $X_{3,2}$ cannot be caused. Similarly, we can calculate out $\Pr\{B_{1,2}B_{4,2}|E\} = 1$, $\Pr\{B_{1,1}|E\} = \Pr\{B_{4,1}|E\} = \Pr\{B_{1,0}|E\} = \Pr\{B_{4,0}|E\} = 0$, and so on.

During expanding of E and $H_{kj}E$, in addition to the above rules/corollaries, we may encounter the case in which an absorbed event can be absorbed by more than one absorbing events. For the example of $X_n = F_{n;1}V_1 + F_{n;2}V_2$, we have

$$
\begin{aligned}
&(X_n)^2\\
&\quad= (F_{n;1}V_1 + F_{n;2}V_2)^2\\
&\quad= (F_{n;1}V_1)^2 + 2F_{n;1}VF_{n;2}V_2 + (F_{n;2}V_2)^2\\
&\quad= (r_{n;1}/r_n)^2A_{n;1}V_1 + 2(r_{n;1}/r_n)A_{n;1}V(r_{n;2}/r_n)A_{n;2}V_2\\
&\qquad+(r_{n;2}/r_n)^2A_{n;2}V_2
\end{aligned}
$$

in which (21) in Corollary 12 is used. According to Corollary 13, $A_{n;1}V_1A_{n;2}V_2$ can be absorbed by either $A_{n;1}V_1$ or $A_{n;2}V_2$. Suppose it is absorbed by $A_{n;1}V_1$, we have

$$
\begin{aligned}
&(X_n)^2\\
&\quad= (r_{n;1}/r_n)^2A_{n;1}V_1 + 2(r_{n;1}/r_n)A_{n;1}V(r_{n;2}/r_n)A_{n;2}V_2\\
&\qquad+(r_{n;2}/r_n)^2A_{n;2}V_2\\
&\quad= \left((r_{n;1}/r_n)^2 + 2(r_{n;1}/r_n)(r_{n;2}/r_n)\right) A_{n;1}V_1\\
&\qquad+(r_{n;2}/r_n)^2A_{n;2}V_2\\
&\quad\neq X_n.
\end{aligned}
\tag{48}
$$

Suppose it is absorbed by $A_{n;2}V_2$, we have

$$
\begin{aligned}
&(X_n)^2\\
&\quad= (r_{n;1}/r_n)^2A_{n;1}V_1 + 2(r_{n;1}/r_n)A_{n;1}V(r_{n;2}/r_n)A_{n;2}V_2\\
&\qquad+(r_{n;2}/r_n)^2A_{n;2}V_2\\
&\quad= (r_{n;1}/r_n)^2A_{n;1}V_1 + \left((r_{n;2}/r_n)^2 + 2(r_{n;1}/r_n)(r_{n;2}/r_n)\right)\\
&\qquad\times A_{n;2}V_2\\
&\quad\neq X_n.
\end{aligned}
\tag{49}
$$

It is obvious that (48) and (49) have different results and therefore, the two ways of absorption are inconsistent. We cannot say which way is incorrect, because they should have the equal chance. Thus, we have the following assumption.

Assumption 2: During the weighted event absorption, when the absorbing events are more than one, they have the equal chance to absorb the absorbed event.

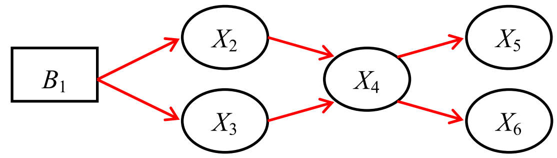


Fig. 13. Multiply connected graph.

Where the assumptions of DUCG are sequentially indexed in the serial papers. Assumption 2 is reasonable, because no absorbing event has the priority to absorb the absorbed event over the other absorbing events. To represent this situation, equal chance is the only choice. Thus, the above example becomes

$$
\begin{aligned}
(X_n)^2 &= (r_{n;1}/r_n)^2A_{n;1}V_1 + 2(r_{n;1}/r_n)A_{n;1}V(r_{n;2}/r_n)A_{n;2}V_2\\
&\quad+(r_{n;2}/r_n)^2A_{n;2}V_2\\
&= \left((r_{n;1}/r_n)^2 + (r_{n;1}/r_n)(r_{n;2}/r_n)\right) A_{n;1}V_1 + \left((r_{n;2}/r_n)^2\right.\\
&\quad\left.+(r_{n;1}/r_n)(r_{n;2}/r_n)\right) A_{n;2}V_2\\
&= \left((r_{n;1}/r_n) + (r_{n;2}/r_n)\right)(r_{n;1}/r_n)A_{n;1}V_1 + \left((r_{n;2}/r_n)\right.\\
&\quad\left.+(r_{n;1}/r_n)\right)(r_{n;2}/r_n)A_{n;2}V_2\\
&= (r_{n;1}/r_n)A_{n;1}V_1 + (r_{n;2}/r_n)A_{n;2}V_2\\
&= X_n
\end{aligned}
\tag{50}
$$

in which, $(r_{n;1}/r_n) + (r_{n;2}/r_n) = 1$ is used, because $r_n = r_{n;1} + r_{n;2}$ in this example. Of course, the result of (50) is expected.

In general, we have the following theorem.

Theorem 2: Given that integer $y \geq 1$ and V_i are all parent variables of X_n

$$\left(\sum_i F_{n;i}V_i\right)^y = \sum_i F_{n;i}V_i \tag{51}$$

$$\left(\sum_i F_{nk;i}V_i\right)^y = \sum_i F_{nk;i}V_i \tag{52}$$

$$\left(\sum_i F_{n;ij}V_{ij}\right)^y = \sum_i F_{n;ij}V_{ij} \tag{53}$$

where the theorems of DUCG are sequentially indexed in the serial papers. Theorem 2 ensures that the expanding of X_n or X_{nk} is consistent with rule/Corollary 11.

Proof: See Appendix E.

As mentioned in [22], we may encounter the multiply connected DUCG as shown in Fig. 13: Then, we have

Rule 15: $A_{nk_n;ij}V_{ij}A_{mk_m;ij}V_{ij} = A_{nk_n;ij}A_{mk_m;ij}V_{ij}$.

Corollary 15: $A_{nk_n;i}V_iA_{mk_m;i}V_i = \left(A_{nk_n;i} * A_{mk_m;i}\right)V_i$, in which

$$
\begin{aligned}
\left(A_{nk_n;i} * A_{mk_m;i}\right) \equiv \big(&A_{nk_n;i1}A_{mk_m;i1} \quad A_{nk_n;i2}A_{mk_m;i2} \quad \ldots\\
&A_{nk_n;ij}A_{mk_m;ij} \quad \ldots \quad A_{nk_n;iJ}A_{mk_m;iJ}\big).
\end{aligned}
$$

Correspondingly, we have

$$
\begin{aligned}
a_{nk_n;i} * a_{mk_m;i} \equiv \big(&a_{nk_n;i1}a_{mk_m;i1} \quad a_{nk_n;i2}a_{mk_m;i2} \quad \ldots\\
&a_{nk_n;ij}a_{mk_m;ij} \quad \ldots \quad a_{nk_n;iJ}a_{mk_m;iJ}\big)
\end{aligned}
$$

where "*" is an AND/multiplication matrix operator specially defined in DUCG.

Proof: Rule 15 is obvious. Through applying Rule 11, $V_{ij}V_{ij'} = 0$, $j \neq j'$, we have

$$\begin{aligned} &A_{nk_n;i}V_iA_{mk_m;i}V_i \\ &\quad = \left(\sum_{j=1}^{J} A_{nk_n;ij}V_{ij}\right)\left(\sum_{j=1}^{J} A_{mk_m;ij}V_{ij}\right) \\ &\quad = \sum_{j=1}^{J} A_{nk_n;ij}A_{mk_m;ij}V_{ij} = (A_{nk_n;i} * A_{mk_m;i})V_i. \end{aligned} \tag{54}$$

For the example in Fig. 13, we have

$$\begin{aligned} X_{5k} &= F_{5k;4}X_4 = F_{5k;4}(F_{4;2}F_{2;1}B_1 + F_{4;3}F_{3;1}B_1) \\ X_{6j} &= F_{6j;4}X_4 = F_{6j;4}(F_{4;2}F_{2;1}B_1 + F_{4;3}F_{3;1}B_1). \end{aligned}$$

Through applying Corollary 15, we have

$$\begin{aligned} X_{5k}X_{6j} &= F_{5k;4}X_4F_{6j;4}X_4 \\ &= (r_{5;4}/r_5)A_{5k;4}X_4(r_{6;4}/r_6)A_{6j;4}X_4 \\ &= (r_{5;4}/r_5)(r_{6;4}/r_6)(A_{5k;4} * A_{6j;4})X_4 \\ &= (F_{5k;4} * F_{6j;4})(F_{4;2}F_{2;1}B_1 + F_{4;3}F_{3;1}B_1). \end{aligned} \tag{55}$$

Since all the matrices within any event product of (55) are independent of each other, we have

$$\begin{aligned} Pr\{X_{6j}\} &= f_{6j;4}(f_{4;2}f_{2;1}b_1 + f_{4;3}f_{3;1}b_1) \\ Pr\{X_{5k}X_{6j}\} &= (f_{5k;4} * f_{6j;4})(f_{4;2}f_{2;1}b_1 + f_{4;3}f_{3;1}b_1) \end{aligned}$$

where the operator "*" for F- and f-type variables is the same as defined in Corollary 15.

As illustrated in this example, it is seen that Corollary 15 is very efficient in dealing with the multiply connected variables. DUCG does not need any special algorithm such as clustering method ([4] for details) in dealing with the multiply connected graph. Only the new operator "*" defined in Corollary 15 is needed.

Finally, rules/Corollaries 11–15 and Assumption 2 provide the algorithm for expanding $H_{kj}E$ and E in terms of both weighted events and matrices into the form of sum-of-products composed of $\{B, A, r\}$ variables/parameters. These products are not inclusive with each other, and the sum-of-products are just the probability expression by replacing the uppercase letters with lowercase letters. Following principles are used in this algorithm: perform the ordinary set operation such as exclusion and inclusion/absorption, while keeping the weighting factors remain, so as to keep the sum of all state probabilities of a variable always equal to 1. Of course, the algorithm has to be self-consistent, such as

$$(X_n)^y = \left(\sum_i F_{n;i}V_i\right)^y = \sum_i F_{n;i}V_i = X_n$$

and so on, which is ensured by the rules/corollaries and theorems of DUCG. The above algorithm is to expand E and $H_{kj}E$ such as $X_{2,2}X_{3,2}$, and $B_{1,2}X_{2,2}X_{3,2}$ and so on. Once the expanding expression becomes composed of only the $\{B$-, D-, A-, P-,$\}$ type events/variables and r-type parameters, we say that this expression is expanded. Once the exclusion and absorption in the expanding expression are done, the expanding finishes and the expanded expression is the same as a probability expression, which can be calculated by simply replacing the $\{B$-, A-, P-$\}$ type events/variables with their $\{b$-, a-, p-$\}$ type parameters, or in other words, replacing the uppercase letters with the lowercase letters. Actually, based on Assumptions 1 and 2, rules/Corollaries 11–15, and Theorems 1 and 2, a new mathematics is formed. This new mathematics may be called the weighted set theory. Of course, we still have a lot of work to do before it becomes a well-founded mathematics.

V. Discussions on Performing Exact Inference With Incomplete DUCG

From the above discussions, it is easy to see that the complete representation of M-DUCG without DCGs represents a joint probability distribution over a set of random variables. But to give the complete DUCG is not easy and sometimes unnecessary. In the example shown in Fig. 11, suppose we do not care about the states of normal and low pressures ($X_{2,0}$ and $X_{2,1}$) and normal and low temperatures ($X_{3,0}$ and $X_{3,1}$), we may give the data of $a_{2;1}$, $a_{2;3}$, and $a_{3;4}$ as

$$a_{2;1} = \begin{pmatrix} - & - & - \\ - & - & - \\ 0 & 0 & 0.8 \end{pmatrix}, \quad a_{2;3} = \begin{pmatrix} - & - & - \\ - & - & - \\ 0 & 0 & 0.9 \end{pmatrix}$$
$$a_{3;4} = \begin{pmatrix} - & - & - \\ - & - & - \\ 0 & 0 & 0.7 \end{pmatrix} \tag{56}$$

in which, "–" means not in concern or no contribution from this parent event, and its function is equivalent to 0. Then, b_{i0} and b_{i1} are also not in concern, and b_1 and b_4 can be given as

$$b_1 = \begin{pmatrix} - & - & 0.1 \end{pmatrix}^T, \quad b_4 = \begin{pmatrix} - & - & 0.2 \end{pmatrix}^T. \tag{57}$$

Even after all these changes, we see that the calculations and results of (46) and (49) remain unchanged, because the parameter values marked as "–" are not involved in the calculations. It is seen that the parameters given above are incomplete and the DUCG shown in Fig. 11 is no longer the complete representation of the joint probability distribution over random variables B_1, X_2, X_3, and B_4. Even so, we still get the exact results in (44) and (47). This property of DUCG exists in not only this example, but also other cases, because it is ensured by Theorem 1 presented in [22], which reveals that the causality chains in DUCG are self-relied.

It is well known that BN represents a joint probability distribution over a set of random variables and the probabilistic inference of BN is to update this distribution conditioned on the observed evidence. The above example shows that DUCG does not have to be a complete representation of a joint probability distribution over a set of random variables, while the exact probabilistic inference can still be made. Actually, DUCG can be a partial (incomplete) representation of the joint probability distribution, which represents the state-of-the-knowledge of people to the real world. In other words, we can represent the joint probability distribution partially and calculate the posterior probabilities of hypotheses in concern exactly, provided that the parameters in concern are given. This

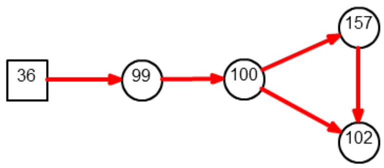


Fig. 14. Sub-DUCG no. 14.

makes DUCG more realistic and applicable. The following example demonstrates this property.

VI. Real Application: Fault Diagnoses With DUCG

A. Brief Description of the Target System

The target system is the two turbine generator sets system installed in Lingdong nuclear power plant operated by China General Nuclear Power Corporation. This generator system (rated active power 1150-MW half speed) consists of the generator excitation and voltage regulation system, generator stator cooling water system, generator hydrogen cooling system, generator hydrogen supply system, conventional island closed cooling water system (SRI), generator seal oil system, and so on. The system division rule is according to the French basic system code. Totally 633 variables are currently selected and defined for this application. In which, there are 427 X-type variables, and most of them are signals and can be received online from the supervisory information system (SIS) of the nuclear power plant once per second (the total number of signals provided by SIS once per second is ∼50 000); there are 205 B-type variables representing all the possible root causes when abnormality of the generator system appears; there are five G-type variables representing the logics among some variables; there is one D-type variable/event representing a normal situation that a high bearing vibration may happen when the generator speed goes through some special ranges; there are 2952 A/a-type matrices or red directed arcs connecting the above X-, B-, G-, and D-type variables; there are 34 input variables of the five logic gates.

B. Construction of the DUCG of the Generator System

Due to the large scale and complexity of this system, the DUCG is constructed by being decomposed as 32 sub-DUCGs listed in Table I.

As they are too many and too trivial, this paper gives details of only one of them.

1) Sub-DUCG No. 14: The sub-DUCG no. 14 is for the numerous windings short circuit fault of generator excitation transformer. It is constructed as shown in Fig. 14.

In which, for simplicity, the variable letters B and X are ignored and only their subscripts remain, because B and X can be recognized by the shapes in the graph: square indicates B and cycle indicates X. The descriptions of the variables in Fig. 14 are shown in Table II.

All the variables in Fig. 14 have only two states: 0 (normal) or 1 (abnormal). The b- and a-type parameters encoded in

TABLE I

List of Sub-DUCGs of the Generator System

No.	Description
1	Short circuit fault of generator's stator windings
2	Cooling water lost fault generator stator windings
3	Abnormally high temperature of generator upper winding (1)
4	Abnormally high temperature of generator upper winding (2)
5	Generator magnetic core fault
6	Abnormally high temperature of generator lower winding
7	Vibration fault caused by generator retaining ring break up
8	Generator tilting pad journal bearing's vibration fault
9	Vibration fault caused by the rotor short circuit/the generator rotor fans break up
10	Short circuit fault of generator excitation transformer's magnetic core
11	Partial short circuit fault of generator excitation transformer's magnetic core
12	Generator's rotor windings short circuit fault
13	Generator excitation transformer's windings short circuit fault
14	Numerous windings short circuit fault of generator excitation transformer
15	Generator excitation transformer's windings open circuit fault
16	Generator hydrogen leakage fault
17	Incoming cable insulation aging fault of generator exciter inductor
18	Fault of SRI
19	Vibration fault caused by the generator rotor magnetization
20	Inner generator cooling water leakage fault
21	Abnormally high temperature of generator tilting pad journal bearing
22	Generator thyristor breakdown fault (1)
23	Generator thyristor breakdown fault (2)
24	Generator thyristor breakdown fault (3)
25	Generator hydrogen leakage alarm because of the shaft seal fault
26	Generator lube oil hi-level alarm because of the shaft seal fault
27	Generator hydrogen cooler water isolation inlet valve fault closing fault
28	Generator hydrogen cooler's water side fouling fault (1)
29	Generator hydrogen cooler's water side fouling fault (2)
30	Generator hydrogen cooler's water side fouling fault (3)
31	Generator hydrogen cooler's water side fouling fault (4)
32	Generator hydrogen cooler's airproof ring damage fault

Fig. 14 are shown below, in which "–" indicates not in concern

$$b_{36} = \begin{pmatrix} - \\ 0.01 \end{pmatrix}, \quad a_{99;36} = \begin{pmatrix} - & - \\ - & 0.8 \end{pmatrix}$$
$$a_{100;99} = \begin{pmatrix} - & - \\ - & 0.99 \end{pmatrix}, \quad a_{157;100} = \begin{pmatrix} - & - \\ - & 0.99 \end{pmatrix}$$
$$a_{102;100} = \begin{pmatrix} - & - \\ - & 0.99 \end{pmatrix}, \quad a_{102;157} = \begin{pmatrix} - & - \\ - & 1 \end{pmatrix}$$
$$a_{102;57} = \begin{pmatrix} - & - \\ - & 1 \end{pmatrix}, \quad \text{all } r_{n;i} = 1$$

in which, $0 < a_{nk;ij} < 1$ means that the effect may appear later after the cause. $a_{nk;ij} = 1$ indicates that the effect will appear immediately after the cause. To be exact, we should give $a_{nk;ij}$ as a function of time t as in [18]. But for simplicity, we use a fixed number (e.g., $a_{99,1;36,1} = 0.8$) to roughly represent the causality delay situation. Since the $a_{nk;ij}$ parameters are not given completely, the DUCG of the generator system is incomplete. This is because the purpose of this application is

TABLE II
UNITS FOR MAGNETIC PROPERTIES

Variable	Descriptions
X_{99}	Excitation transformer over current protection triggers (GPA127SY_XG01)
X_{100}	Exciter transformer's protection triggers (GPA009XK_XG01)
X_{102}	Generator is tripped (GSY001JA_XA01)
X_{157}	Excitation regulator is tripped (GEX206KST_XG01)
B_{36}	Numerous windings short circuit fault of generator excitation transformer

to diagnose the faults or root causes of the abnormality of the generator system when abnormal signals are received. As illustrated in this section, all a-type matrices in the 32 sub-DUCGs are sparse. At least, all $a_{nk;ij}$, $k = 0$, and/or $j = 0$, are given as "–," because they are not in concern.

In general, for fault diagnoses, only the causalities among abnormal states are represented and the causalities connected to normal states are not represented, because normal states are not in concern. This brings us a great convenience to construct DUCG and to simplify the DUCG by applying Rules 1–10 presented in [22] based on the observed evidence. Usually, the more the incompleteness is, the more significant the simplification can be.

2) Combination of Sub-DUCGs: Combining the 32 sub-DUCGs listed in Table I by fusing same variables in different sub-DUCGs, we get the synthetic DUCG of the generator system as shown in Fig. 15.

It is seen that although the sub-DUCGs are relatively simple, the synthetic DUCG is quite complex. It is not easy to recognize the causalities among variables in Fig. 15. However, since the construction of the synthetic DUCG is decomposed as to construct small and simple sub-DUCGs, the DUCG construction is easy and can be done by more than one people separately. Of course, during the separate constructions of sub-DUCGs, we have to keep the definitions of all variables consistent. Where overlaps among sub-DUCGs are allowed. In addition, the maintenance/revision of the DUCG is easy, because we need only to maintain/update the sub-DUCGs individually. The simplest sub-DUCG is a module composed of a child variable and its parent variables, such as that shown in Fig. 8. If the parameters of a same directed arc are given differently in different sub-DUCGs, the combination process will prompt this difference to users automatically for them to discuss and decide what is correct. Also important is that the synthetic DUCG shown in Fig. 15 does not involve any DCG. Otherwise, we have to discuss how to deal with DCGs, which is not the topic of this paper. Actually, the decomposed construction of DUCG may cause DCGs easily, although every sub-DUCG does not have DCG.

C. Fault Diagnosis

One of the fault diagnoses for this generator system is the fault diagnosis for SRI. In the corresponding experiment, the abnormal signal states observed online from SIS are $X_{4,2}$, $X_{8,2}$, $X_{185,1}$, $X_{186,1}$, $X_{187,1}$, $X_{188,2}$, $X_{189,2}$, and $X_{191,1}$, and all the states of other X-type variables are normal.

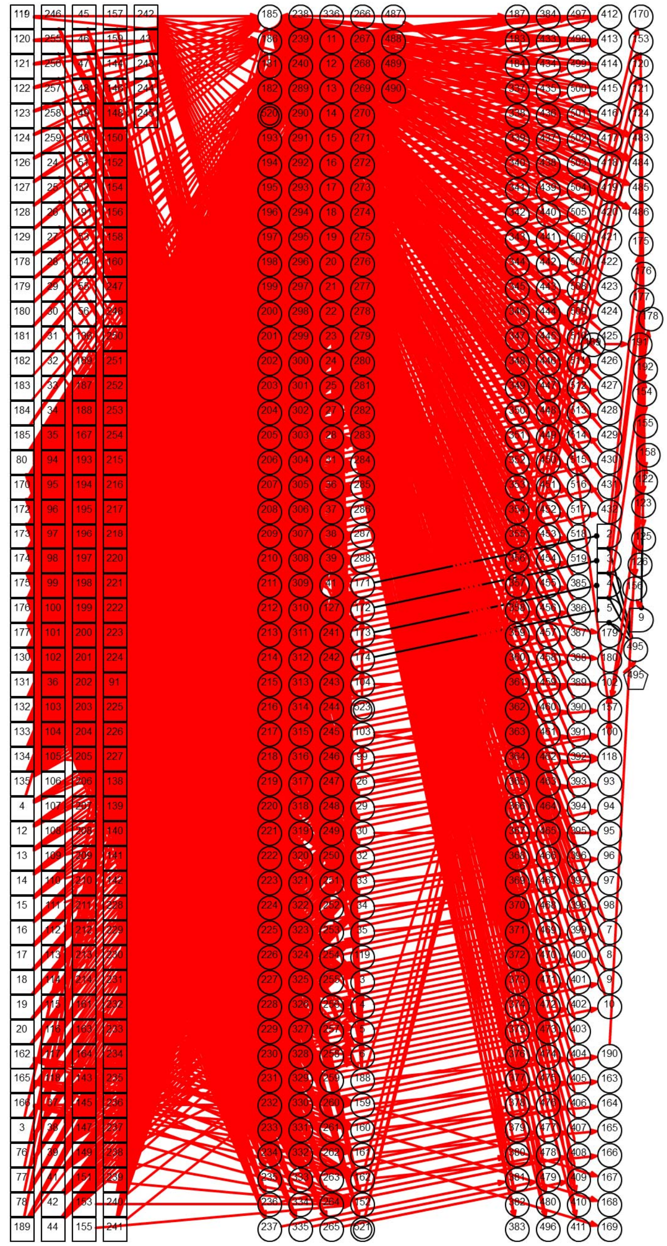


Fig. 15. Synthetic DUCG of the generator system.

When the abnormal signals are received, the inference engine is activated. According to the simplification Rules 1–10 presented in [22], all the irrelevant and meaningless causalities and variables are eliminated from the synthetic DUCG. For example, Rule 2 in [22] says: "If E shows that V_{ij}, $V \in \{B, X\}$, is true while V_{ij} is not a parent event of X_n, $F_{n;i}$, or $P_{n;i}$ is eliminated from the DUCG." Thus the outputs of all observed X_{i0} are discarded, because in Fig. 15, $a_{nk;i0} = 0$, which means that all X_{i0} are not parents of any other event. Finally, Fig. 15 is simplified as Fig. 16. The significant simplification is benefited from the incomplete representation of the DUCG encoded in Fig. 15.

Fig. 16 includes all abnormal variable states, in which, the yellow color indicates that the variable state is abnormally

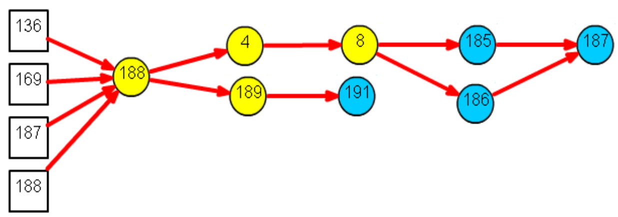


Fig. 16. Simplified DUCG based on received evidences.

high, the blue color indicates that the variable state is abnormally low. It is easy to see that Fig. 16 explains the inference results well.

As expected, the inference results are correct, because only $B_{136,1}$, $B_{169,1}$, $B_{188,1}$, and $B_{187,1}$ (where "1" indexes the failure state) can cause all the observed abnormal variable states. The rank of them is according to their rank probabilities defined in [22] and is ignored here, because we have already known that $B_{136,1}$, $B_{169,1}$, $B_{188,1}$, and $B_{187,1}$ are the only four possible causes out of the 205 B-type root cause variables and the rank depends on only their unconditional probabilities $b_{136,1}$, $b_{169,1}$, $b_{188,1}$, and $b_{187,1}$. The total inference time by our DUCG software developed with Java is ~719 ms in the computer ThinkPad T420. We have performed 35 different cases, the inference times range from 324 to 12 786 ms, in which 29 are <1 s, five are between 1 and 2 s, and one is 12 786 ms (there are 214 abnormal signals in this case).

It should be mentioned that in the example of Fig. 16, there is no negative evidence (the observed normal state of child variables of hypotheses $B_{136,1}$, $B_{169,1}$, $B_{188,1}$, and $B_{187,1}$); otherwise, the corresponding normal state variables as negative evidences of the hypotheses should also be included in Fig. 16. For simplicity, other detailed information of this real application is omitted in this paper.

VII. Conclusion

In this paper, the statistic basis of DUCG is demonstrated. It is seen that the parameters of DUCG can be learned from data. But in cases without data, the compact parameters in DUCG can be easily specified by domain experts directly based on their knowledge. A set of rules/corollaries for expanding E and $H_{kj}E$, which are a group of events multiplied, is presented. They involve the event matrix operation and the $(r_{n;i}/r_n)$-type weighting factor operation. These rules/corollaries along with Assumptions 1 and 2 and Theorems 1 and 2 form a new algorithm or mathematics that may be called the weighted set theory. It is initially proved in this paper that the algorithm/theory is self-consistent. The parameters of DUCG can be either complete or incomplete. The incomplete DUCG can represent people's knowledge in concern only, although it may not be sufficient to represent a joint probability distribution over a set of random variables. As being ensured by Theorem 1 presented in [22], the causality chain in DUCG is self-relied. Therefore, we can get the exact inference results based on an incomplete DUCG. A real application of DUCG, i.e., the generator system of a nuclear power plant, is demonstrated. It is seen that DUCG is a user friendly and powerful tool to construct and maintain large and complex knowledge base. The inference is efficient and the results are easy to be explained to users.

More complex cases are planned to be addressed in the following papers. These cases include the DCG structure, the combination of dynamical evidence, the continuous overlap of causalities over the past time at a same later time, the initiating and noninitiating events in process systems, the forecast and prediction of system fault, the uncertain or fuzzy evidence, the continuous variables freely mixed with discrete variables, the cubic DUCG dynamically generated according to the online evidence based on the original DUCG, and so on. Of course, we still have a lot of work to do to well found the weighted set theory that is only initially presented in this paper.

Appendix A

Proof for Corollary 12

By the definition of $F_{n;i}$, we know that $F_{n;i}$ must appears with X_n just before $F_{n;i}$ and V_i just after $F_{n;i}$. By definition, we have $(F_{n;i})^y = (r_{n;i}/r_n)^y (A_{n;i})^y$. It is noted that $A_{nk;ij}A_{nk';ij} = 0$, $A_{nk;ij}A_{nk;ij'} = 0$, and $A_{nk;ij}A_{nk';ij'} = 0$ when $k \neq k'$ and $j \neq j'$, because $X_{nk}X_{nk'} = 0$ and $V_{ij}V_{ij'} = 0$. Therefore, as the result of $(A_{n;i})^y$, only $(A_{nk;ij})^y = A_{nk;ij}$ remain while other multiplications are null. That is, only $A_{nk;ij}$ exist as the elements of $(A_{n;i})^y$, which means $(A_{n;i})^y = A_{n;i}$. Thus, we have $(F_{n;i})^y = (r_{n;i}/r_n)^y A_{n;i}$. Equation (21) is then proved.

Similarly, we can prove (22) and (23). To prove (24), by applying Rules 11 and 12, we have

$$\begin{aligned} &F_{nk;ij}\prod_{h=1}^{y-1}F_{n\bullet;i\bullet} \\ &= (r_{n;i}/r_n)^y A_{nk;ij}\prod_{h=1}^{y-1}A_{n\bullet;i\bullet} = (r_{n;i}/r_n)^y A_{nk;ij}\prod_{h=1}^{y-1}A_{nk;ij} \\ &= (r_{n;i}/r_n)^y A_{nk;ij}. \end{aligned} \tag{58}$$

Similarly, (25) and (26) can be proved.

According to Corollary 11, only X_{ij} can simultaneously satisfy $A_{n\bullet;i}$, $A_{n\bullet;ij}$, $A_{i;h\bullet}$, and $A_{ij;h\bullet}$. Therefore, $F_{n\bullet;i}F_{ij;h\bullet} = (r_{n;i}/r_n)A_{n\bullet;i}(r_{i;h}/r_i)A_{ij;h\bullet} = (r_{n;i}/r_n)A_{n\bullet;ij}(r_{i;h}/r_i)A_{ij;h\bullet} = F_{n\bullet;ij}F_{ij;h\bullet}$. Similarly, we can prove $F_{n\bullet;ij}F_{i;h\bullet} = F_{n\bullet;ij}F_{ij;h\bullet}$. Equation (27) is then proved.

Similarly, since $A_{nk;ij}$ implies V_{nk} and V_{ij}, according to Corollary 11, there must be $V_{nk}V_{n\bullet} = V_{nk}$ and $V_{ij}V_{i\bullet} = V_{ij}$, which means $A_{nk;i\bullet}A_{n;i'\bullet} = A_{nk;i\bullet}A_{nk;i'\bullet}$, $A_{n\bullet;ij}A_{n'\bullet;i} = A_{n\bullet;ij}A_{n'\bullet;ij}$, and $A_{n\bullet;i}V_iV_{ij} = A_{n\bullet;ij}V_{ij}$. And according to $V_{ij}V_{ij'}$ =0 in Rule 11, there must be $A_{nk;i\bullet}A_{nk';i'\bullet} = 0$, $A_{n\bullet;ij}A_{n'\bullet;ij'} = 0$. Thus, (28) is proved.

APPENDIX B

PROOF FOR COROLLARY 13

$\prod_{i\in S_1} A_{n\bullet;i\bullet}V_{i\bullet} \supseteq \prod_{i\in S_m, m\neq 1} A_{n\bullet;i\bullet}V_{i\bullet}$ means that more events are multiplied, the smaller the set is. Then, we have

$$\sum_{m=1}^{M}\prod_{i\in S_m} F_{n\bullet;i\bullet}V_{i\bullet} = \sum_{m=1}^{M}\prod_{i\in S_m}(r_{n;i}/r_n)A_{n\bullet;i\bullet}V_{i\bullet} = \left(\sum_{m=1}^{M}\prod_{i\in S_m}(r_{n;i}/r_n)\right)\prod_{i\in S_1}A_{n\bullet;i\bullet}V_{i\bullet}. \quad (59)$$

The second equator is because once $\prod_{i\in S_m} A_{n\bullet;i\bullet}V_{i\bullet}$ is true, $\prod_{i\in S_1} A_{n\bullet;i\bullet}V_{i\bullet}$ must be true. And once $\prod_{i\in S_1} A_{n\bullet;i\bullet}V_{i\bullet}$ is true, the whole equation is true, which is equivalent to $\prod_{i\in S_m} A_{n\bullet;i\bullet}V_{i\bullet}$ being true. Then, we can use $\prod_{i\in S_1} A_{n\bullet;i\bullet}V_{i\bullet}$ to replace $\prod_{i\in S_m} A_{n\bullet;i\bullet}V_{i\bullet}$ while keep its r-type factors remain the same. Where the condition $\prod_{i\in S_1} A_{n\bullet;i\bullet}V_{i\bullet} \supseteq \prod_{i\in S_m} A_{n\bullet;i\bullet}V_{i\bullet}$ is used. Note that "•" indicates either the subscript exists or not.

APPENDIX C

PROOF FOR COROLLARY 14

To prove (30), by applying rule/Corollary 11 and 12, we have

$$\begin{aligned}
&F_{nk;ij}V_{ij}\sum_{i'}F_{n\bullet;i'\bullet}V_{i'\bullet}\\
&= F_{nk;ij}V_{ij}F_{n\bullet;i\bullet}V_{i\bullet} + \sum_{i'\neq i}F_{nk;ij}V_{ij}F_{n\bullet;i'\bullet}V_{i'\bullet}\\
&= (r_{n;i}/r_n)^2 A_{nk;ij}V_{ij} + \sum_{i'\neq i}(r_{n;i}/r_n)(r_{n;i'}/r_n)\\
&\quad A_{nk;ij}V_{ij}A_{n\bullet;i'\bullet}V_{i'\bullet}\\
&= (r_{n;i}/r_n)^2 A_{nk;ij}V_{ij} + \sum_{i'\neq i}(r_{n;i}/r_n)(r_{n;i'}/r_n)A_{nk;ij}V_{ij}\\
&= \Big((r_{n;i}/r_n)^2 + \sum_{i'\neq i}(r_{n;i}/r_n)(r_{n;i'}/r_n)\Big)A_{nk;ij}V_{ij}\\
&= (r_{n;i}/r_n)\Big((r_{n;i}/r_n) + \sum_{i'\neq i}(r_{n;i'}/r_n)\Big)A_{nk;ij}V_{ij}\\
&= (r_{n;i}/r_n)A_{nk;ij}V_{ij}\\
&= F_{nk;ij}V_{ij}
\end{aligned}$$

in which, the second equator is because of applying (24), the third equator is because once $A_{nk;ij}V_{ij}A_{n\bullet;i'\bullet}V_{i'\bullet}$ is true, $A_{nk;ij}V_{ij}$ must be true. And once $A_{nk;ij}V_{ij}$ is true, the whole equation is true, which is equivalent to $A_{nk;ij}V_{ij}A_{n\bullet;i'\bullet}V_{i'\bullet}$ being true. Thus, we can use $A_{nk;ij}V_{ij}$ to replace $A_{nk;ij}V_{ij}A_{n\bullet;i'\bullet}V_{i'\bullet}$ while keeping its r-type parameters remain the same. Equation (30) is then proved.

Similarly, we can prove (31)–(33).

APPENDIX D

PROOF FOR $(X_{nk})^y = \left(\sum_i F_{nk;ij}V_{ij}\right)^y = X_{nk}$

$$\begin{aligned}
(X_{nk})^y &= \left(\sum_i F_{nk;i}V_i\right)^y\\
&= \left(\sum_i F_{nk;i}V_i\sum_{i'}F_{nk;i'}V_{i'}\right)\left(\sum_{i"}F_{nk;i"}V_{i"}\right)^{y-2}\\
&= \sum_i F_{nk;i}V_i\left(\sum_{i"}F_{nk;i"}V_{i"}\right)^{y-2}\\
&= \sum_i F_{nk;i}V_i\left(\sum_{i'}F_{nk;i'}V_{i'}\right)\left(\sum_{i"}F_{nk;i"}V_{i"}\right)^{y-3}\\
&= \sum_i F_{nk;i}V_i\left(\sum_{i"}F_{nk;i"}V_{i"}\right)^{y-3}\cdots\\
&= \sum_i F_{nk;i}V_i = X_{nk}.
\end{aligned}$$

APPENDIX E

PROOF FOR THEOREM 2

No proof is needed when $y = 1$. When $y \geq 2$, we have

$$\left(\sum_i F_{n;i}V_i\right)^y = \left(\sum_i F_{n;i}V_i\right)^2\left(\sum_i F_{n;i}V_i\right)^{y-2}. \quad (60)$$

In which, by applying (21) in Corollary 12, we have

$$\begin{aligned}
&\left(\sum_i F_{n;i}V_i\right)^2\\
&\quad= \sum_i (F_{n;i}V_i)^2 + 2\sum_{i\neq i'}F_{n;i}V_iF_{n;i'}V_{i'}\\
&\quad= \sum_i (r_{n;i}/r_i)^2 A_{n;i}V_i + 2\sum_{i\neq i'}(r_{n;i}/r_n)(r_{n;i'}/r_n)\\
&\qquad\times A_{n;i}V_iA_{n;i'}V_{i'}.
\end{aligned}$$

$A_{n;i}V_iA_{n;i'}V_{i'}$ can be absorbed by either $A_{n;i}V_i$ or $A_{n;i'}V_{i'}$, both cover $\sum_i (r_{n;i}/r_i)^2 A_{n;i}V_i$. According to Assumption 2, they have the equal chance to absorb $A_{n;i}V_iA_{n;i'}V_{i'}$, i.e., one of the two $A_{n;i}V_iA_{n;i'}V_{i'}$ in the above equation is replaced by $A_{n;i}V_i$ and another is replaced by$A_{n;i'}V_{i'}$. Similar to (50), we have

$$\left(\sum_i F_{n;i} V_i\right)^2$$
$$= \sum_i (r_{n;i}/r_n)^2 A_{n;i} V_i + 2 \sum_{i \neq i'} (r_{n;i}/r_n)(r_{n;i'}/r_n) A_{n;i} V_i A_{n;i'} V_{i'}$$
$$= \sum_{i \neq i'} \left((r_{n;i}/r_n)^2 + (r_{n;i}/r_n)(r_{n;i'}/r_n)\right) A_{n;i} V_i$$
$$= \sum_{i \neq i'} \left((r_{n;i}/r_n) + (r_{n;i'}/r_n)\right)(r_{n;i}/r_n) A_{n;i} V_i$$
$$= \sum_i \sum_{i'} (r_{n;i'}/r_n)(r_{n;i}/r_n) A_{n;i} V_i$$
$$= \sum_i (r_{n;i}/r_n) A_{n;i} V_i = \sum_i F_{n;i} V_i.$$

Continuing the decomposition expressed in (60), we can prove (51). In the same way, we can prove (52) and (53).

REFERENCES

[1] K. P. Murphy, "The bayes net toolbox for MATLAB," *Comput. Sci. Stat.*, vol. 33, no. 2, pp. 33–40, 2001.

[2] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*, 1st ed. Cambridge, MA, USA: MIT Press, 2009.

[3] P. Larrañaga and S. Moral, "Probabilistic graphical models in artificial intelligence," *Appl. Soft Comput.*, vol. 11, no. 2, pp. 1511–1528, 2011.

[4] J. Pearl, *Probabilistic Reasoning in Intelligent Systems*. San Mateo, CA, USA: Morgan Kaufmann, 1988.

[5] F. V. Jensen, *Encyclopedia of Environmetrics*. New York, NY, USA: Wiley, 2000.

[6] F. V. Jensen and T. D. Nielsen, *Bayesian Networks and Decision Graphs*, 2nd ed. New York, NY, USA: Springer-Verlag, 2007.

[7] J. Pearl, *Causality: Models, Reasoning and Inference*, 2nd ed. New York, NY, USA: Cambridge Univ. Press, 2009.

[8] H. Ocak and K. A. Loparo, "HMM-based fault detection and diagnosis scheme for rolling element bearings," *J. Vibrat. Acoust.*, vol. 127, no. 4, pp. 299–306, 2005.

[9] N. L. Zhang, "Hierarchical latent class models for cluster analysis," *J. Mach. Learn. Res.*, vol. 5, no. 6, pp. 697–723, 2004.

[10] D. Heckerman, D. M. Chickering, C. Meek, R. Rounthwaite, and C. Kadie, "Dependency networks for inference, collaborative filtering, and data visualization," *J. Mach. Learn. Res.*, vol. 1, pp. 49–75, Oct. 2000.

[11] D. Y. Li, C. Y. Liu, and W. Y. Gan, "A new cognitive model: Cloud model," *Int. J. Intell. Syst.*, vol. 24, no. 3, pp. 357–375, 2009.

[12] A. Zagorecki, M. Voortman, and M. J. Druzdzel, "Decomposing local probability distributions in Bayesian networks for improved inference and parameter learning," in *Proc. 19th Int. FLAIRS Conf.*, Melbourne Beach, FL, USA, May 2006, pp. 860–865.

[13] P. Larranaga, H. Karshenas, C. Bielza, and R. Santana, "A review on evolutionary algorithms in Bayesian network learning and inference tasks," *Inf. Sci.*, vol. 233, pp. 109–125, Jun. 2013.

[14] A. Rebai, *Bayesain Networks*. Rijeka, Croatia: Sciyo, 2010.

[15] D. Bouchaffra, "Mapping dynamic Bayesian networks to α-shapes: Application to human faces identification across ages," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 8, pp. 1229–1241, Aug. 2012.

[16] G. F. Cooper, "The computational complexity of probabilistic inference using Bayesian belief networks," *Artif. Intell.*, vol. 42, nos. 2–3, pp. 393–405, 1990.

[17] P. Dagum and M. Luby, "Approximating probabilistic inference in Bayesian belief networks is NP-hard," *Artif. Intell.*, vol. 60, no. 1, pp. 141–153, 1993.

[18] Q. Zhang, "Probabilistic reasoning based on dynamic causality trees/diagrams," *Rel. Eng. Syst. Safety*, vol. 46, no. 3, pp. 209–220, 1994.

[19] C. Boutilier, N. Friedman, M. Goldszmidt, and D. Koller, "Context-specific independence in Bayesian network," in *Proc. 12th Annu. Conf. Uncertainty Artif. Intell.*, Portland, OR, USA, Aug. 1996, pp. 115–123.

[20] D. Heckerman and J. S. Breese, "Causal independence for probability assessment and inference using Bayesian networks," *IEEE Trans. Syst., Man Cybern. A, Syst. Humans*, vol. 26, no. 6, pp. 826–831, Nov. 1996.

[21] D. Poole and N. L. Zhang, "Exploiting contextual independence in probabilistic inference," *J. Artif. Intell. Res.*, vol. 18, pp. 263–313, Apr. 2003.

[22] Q. Zhang, "Dynamic uncertain causality graph for knowledge representation and reasoning: Discrete DAG cases," *J. Comput. Sci. Technol.*, vol. 27, no. 1, pp. 1–23, 2012.

[23] S. Dash and V. Venkatasubramanian, "Challenges in the industrial applications of fault diagnostic systems," *Comput. Chem. Eng.*, vol. 24, no. 2, pp. 785–791, 2000.

[24] C. Yuan, H. Lim, and T. C. Lu, "Most relevant explanation in Bayesian networks," *J. Artif. Intell. Res.*, vol. 42, no. 1, pp. 309–352, 2011.

Qin Zhang was born in 1956 in Chongqing, China. He received the B.S., M.S., and Ph.D. degrees from Tsinghua University, Beijing, China, in 1982, 1984, and 1989, respectively, all in nuclear engineering.

He was a Visiting Scholar with University of California, Los Angeles, CA, USA, and University of Tennessee, Knoxville, TN, USA, from 1987 to 1989, on fault diagnoses and system reliability engineering. He was a Post-Doctoral Fellow with the School of Economics and Management, Tsinghua University, from 1990 to 1991, and an Associate Professor of Institute of Nuclear Energy Technology (INET), Tsinghua University, from 1991 to 1993, conducting the online expert system development supported by IAEA. He was a Department Manager of China Technology Innovation Corporation and General Manager of United Corporation, Xiamen Technology Innovation, Xiamen, China, from 1993 to 1997. He was the Head of Chongqing Science and Technology Commission from 1997 to 2003 and a Deputy Commissioner of State Intellectual Property Office of China from 2003 to 2009. Since 2009, he has been an Executive Secretary of China Association for Science and Technology. He is a Professor and Ph.D. Supervisor of INET, Tsinghua University and School of Computer Science and Engineering, Beihang University, Beijing, on the theory and system development of uncertain causal knowledge representation and probabilistic reasoning. He has published over 70 papers in international and domestic academic journals and conferences.

Chunling Dong was born in 1979 in Shandong, China. She received the B.S. and M.S. degrees in computer science and technology from Shandong Normal University, Jinan, China, in 2001 and 2005, respectively. She is currently pursuing the Ph.D. degree in computer science and technology with Beihang University, Beijing, China.

She has been a Lecturer with Shandong Normal University, Shandong, since 2005. She has published 18 papers and a monograph. Her current research interests include computational intelligence for engineering reliability, causality representation and inference under uncertainties, and the diagnosis and decision-making for complex systems.

Ms. Dong is currently a member of the Specialized Committee for Uncertain Artificial Intelligence within the Chinese Association for Artificial Intelligence. She has participated in programs with the National Natural Science Foundation of China, Major National Science and Technology Program, Research Foundation for Doctoral Program of China Ministry of Education, and Development Project of China General Nuclear Power Corporation.

Yan Chui was born in Shangdong, China, in 1979. She received the B.S. degree in computer science and technology from Xi'an Tongxin University, Xi'an, China, in 2003, and the M.S. degree in software engineering from Xi'an Jiaotong University, Xi'an, in 2011.

She has been with China General Nuclear Power Corporation, Guangdong, China, since 2011. Her current research interests include reliability engineering, system maintenance, and fault diagnosis of nuclear power plants.

Zhihui Yang was born in Shanxi, China, in 1971. He received the B.S. degree in industrial electrical automation from Dongbei University, Shenyang, China, in 1995.

He was with Shanxi Shentou First Power Plant, Shanxi, as an Engineer, from 1995 to 1999, Shanxi Yangcheng International Power Co., Ltd., Shanxi, from 1999 to 2004, Guangxi Beihai Power Plant, Guangxi, China, from 2004 to 2008, and China General Nuclear Power Corporation, Guangdong, China, from 2008 to 2013. His current research interests include reliability engineering, equipment maintenance, and intelligent operation of power plants.