Zhang Q. Dynamic uncertain causality graph for knowledge representation and reasoning: Discrete DAG cases. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 27(1): 1-23 Jan. 2012. DOI 10.1007/s11390-012-1202-7

Dynamic Uncertain Causality Graph for Knowledge Representation and Reasoning: Discrete DAG Cases

Qin Zhang (张　勤)

School of Computer Science and Technology, Beihang University, Beijing 100191, China

E-mail: zhangqin@buaa.edu.cn

Received August 22, 2011; revised December 20, 2011.

Abstract Developed from the dynamic causality diagram (DCD) model, a new approach for knowledge representation and reasoning named as dynamic uncertain causality graph (DUCG) is presented, which focuses on the compact representation of complex uncertain causalities and efficient probabilistic inference. It is pointed out that the existing models of compact representation and inference in Bayesian Network (BN) is applicable in single-valued cases, but may not be suitable to be applied in multi-valued cases. DUCG overcomes this problem and beyond. The main features of DUCG are: 1) compactly and graphically representing complex conditional probability distributions (CPDs), regardless of whether the cases are single-valued or multi-valued; 2) able to perform exact reasoning in the case of the incomplete knowledge representation; 3) simplifying the graphical knowledge base conditional on observations before other calculations, so that the scale and complexity of problem can be reduced exponentially; 4) the efficient two-step inference algorithm consisting of (a) logic operation to find all possible hypotheses in concern for given observations and (b) the probability calculation for these hypotheses; and 5) much less relying on the parameter accuracy. An alarm system example is provided to illustrate the DUCG methodology.

Keywords causality, uncertainty, knowledge representation, probabilistic reasoning

1 Introduction

Knowledge representation and reasoning deal with uncertain causalities crucial for intelligent systems. Many frameworks have been developed such as Certainty Factors[1], Evidence Reasoning[2], PROSPECTOR[3], Fuzzy Logic[4], Bayesian Network (BN)[5-30]. Among them, BN is in wide-spread use. The compact knowledge representations and efficient inference algorithms are usually the core issues of BN.

The typical representation of conditional probability distributions (CPDs) in BN is conditional probability tables (CPTs). But too many parameters are needed to specify a CPT[23]. For the example of one child variable and five parent variables with five states each, the number of conditional probabilities in the CPT is $5^6 = 15\,625$. On the other hand, the logic relations among variables are mixed and hidden in the CPT parameters, resulting in that BN relies much on the parameter accuracy. To get these parameters, a large number of statistic samples/data are needed. However, in many cases such as fault diagnoses of nuclear power plants, the fault samples are rare, resulting in the difficulty in obtaining CPTs. In fact, one of the bottlenecks of applying artificial intelligence technology is the lack of data, particularly in the area of engineering systems such as power plants, chemical engineering systems, electricity networks. Moreover, the computation amount of inference with CPTs is an NP hard problem, which means that the computation amount is exponential to the scale of problem. The compact representation models directly represent the uncertain causalities among variables, which may easily use the domain engineer's experience/knowledge/belief, rely less on the parameter accuracy and have less inference computation amount.

In order to provide the compact representation, many efforts have been made, such as noisy-OR[6], context-specific independence (CSI)[14], independence of causal influence (ICI)[15], dynamic causality diagram (DCD)[31]. However, many of them are presented for or illustrated with only binary variables, while actually these cases are single-valued but not multi-valued. This paper points out that the single-valued cases are essentially different from the multi-valued cases. This means that the compact representations and the

Regular Paper

This work is supported by Guangdong Nuclear Power Group of China under Contract No. CNPRI-ST10P005 and the National Natural Science Foundation of China under Grant No. 60643006.

corresponding inference algorithms applicable in single-valued cases may not be suitable to be applied in multi-valued cases.

The so called single-valued case or multi-valued case means that the child variable is single-valued or multi-valued. The definitions of the single-valued and multi-valued variables are given below.

Definitions 1. *The single-valued variable is such a child variable for which only the causes of its one state* (*denoted as the true state*) *are specified. The multi-valued variable is such a child variable for which the causes of its more than one states are specified separately.*

For the well-known example of the burglary and earthquake alarm system given in [6], the alarm can be caused by either burglary or earthquake. The alarm variable has two states: "on" (true) and "off" (false). Usually, only the causes of "on" are specified and the causes of "off" are not specified, because "off" has been specified as the complement of "on". According to the definition, the alarm variable is single-valued.

For the example of the temperature in a refrigerator, the temperature variable may have three states: "normal", "high" and "low". The state "high" may be caused by a failure of the compressor. The state "low" may be caused by a failure of the temperature sensor. The causes of more than one state of the temperature variable are specified separately. According to the definition, the temperature variable is multi-valued.

It is important to note that *the meaning of multi-valued variables is different from the meaning of multi-state variables.* A *multi-state* variable means that the states of the variable are more than two. Otherwise, it is binary. A binary variable has two states, e.g., on/off, male/female. A *binary variable can be either single-valued or multi-valued.* The above alarm variable is binary and single-valued. A sex variable is also binary but usually multi-valued, because the biological causes of its two states are usually specified separately.

A multi-state variable is usually multi-valued. The above temperature variable is a multi-state and multi-valued variable. This is because a single-valued multi-state variable is meaningless. As a single-valued variable, all states except the true state can be combined as one state: false, which is the complement of the true state. In other words, the single-valued multi-state variable is actually a single-valued binary variable.

It seems that, in many papers, the single-valued variables are misunderstood as binary variables, while the multi-valued variables are misunderstood as multi-state variables. As a consequence, the compact representations and inference algorithms applicable in single-valued cases are improperly extended to multi-valued cases by simply applying an imposed normalization. This paper points out that such an extension is mathematically improper, and is unclear in representing knowledge, may be self-inconsistent, inexact and even impractical. To overcome these problems and others, the dynamic uncertain causality graph (DUCG) model is presented.

In Section 2, we begin with the detailed discussion on the essential difference between the single-valued and multi-valued cases. In Section 3, the dynamic causality diagram (DCD) presented in [31] is briefly introduced in the way compared with the well-known noisy-OR and CSI, where the cases are actually single-valued only. This section also extends DCD as the single-valued DUCG (S-DUCG). Section 4 presents the compact representation model applicable in multi-valued cases (M-DUCG). Section 5 combines S-DUCG and M-DUCG as DUCG that actually achieves the sufficiency and separability desired for compact representations[19] in both single-valued and multi-valued cases. Meanwhile, an important property of DUCG, i.e., the exact inference with incomplete representation of CPDs, is discussed. Section 6 presents the method to simplify DUCG based on the observed evidence including the occurrence order of events regardless of any query, by which the qualitative solution of the inference may be found before numerical calculation. Section 7 presents the inference algorithm based on the simplified DUCG. Section 8 concludes this paper and outlines the future work briefly.

Due to the length, only the discrete, certain evidence and directed acyclic graph (DAG) are addressed.

2 Essential Difference Between Single-Valued and Multi-Valued Cases

In many cases, people only specify the causes of the true state of a binary variable X_n, where n indexes the variable. Suppose state 1 represents the true state and state 2 represents the false state. For example, the burglary (X_1), earthquake (X_2) and alarm (X_3) mentioned above, $X_3 = x_{31}$ (alarm on) can be caused by either $X_1 = x_{11}$ (burglary appears) or $X_2 = x_{21}$ (earthquake occurs) independently. It is easy for the domain engineers to give the individual conditional probabilities (will be explained later) of $X_3 = x_{31}$ caused by $X_1 = x_{11}$ and $X_2 = x_{21}$ respectively, while it is not easy for them to give the CPT directly. This is because the burglary and earthquake are different domains and their combination samples are difficult to be obtained. However, it should be noted that in this example, only the causes of $X_3 = x_{31}$ are specified, while the causes of $X_3 = x_{32}$ (alarm off) *must not* be specified, because $X_3 = x_{32}$ has been specified as the complement of $X_3 = x_{31}$.

Fig.1 illustrates this binary single-valued case, in which $\longrightarrow$ written as $P_{nk;ij}$ in text represents the state level causal link, where subscripts "*nk*; *ij*" indicate that $X_n = x_{nk}$ is caused by $X_i = x_{ij}$. The subscript before ";" is for the child variable and the subscript after ";" is for the parent variable. The special arrow shape and the green color of the directed arc $\longrightarrow$ indicate that this causal link is different from the CPT type causal link drawn as $\longrightarrow$ in BN.

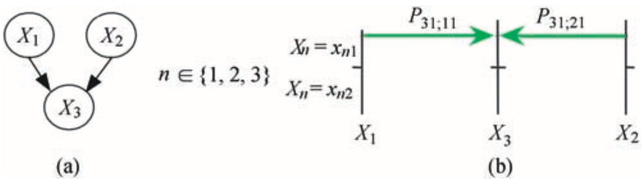


Fig.1. Illustration for the binary single-valued case.

However, the real world is not always so simple. For the example of a simple digital memory circuit, it has two identical states/outputs: "01" and "10". This is a typical binary variable. Similar to the alarm variable, we may denote this variable as X_3 with $X_3 = x_{31}$ representing state "01" and $X_3 = x_{32}$ representing state "10"; but differently, both $X_3 = x_{31}$ and $X_3 = x_{32}$ can be caused by different events. For example, $X_3 = x_{31}$ may be caused by $X_1 = x_{11}$, and $X_3 = x_{32}$ may be caused by $X_2 = x_{21}$, with independently given individual conditional probabilities $p_{31;11}$ and $p_{32;21}$ respectively. The reason why the word "individual" is put in front of "conditional probabilities" is because usually $p_{nk;ij} \neq \Pr\{X_n = x_{nk}|X_i = x_{ij}\}$. In fact, $p_{nk;ij}$ is the probability of the linkage event $P_{nk;ij}$ in DCD, i.e., $p_{nk;ij} \equiv \Pr\{P_{nk;ij}\}$. $p_{nk;ij}$ is also the probability of the complement of the inhibitor in noisy-OR (see [6] and Subsection 3.1 for details). Similar notations are also used in [13] in which $p_{31;11}$ and $p_{32;21}$ are denoted as $c_{X11}(X_{31})$ and $c_{X21}(X_{32})$, so that the two types of conditional probabilities are distinctive.

As mentioned in Section 1, another typical example of binary multi-valued case is sex variable (X_3) that has two identical valued states: "male" ($X_3 = x_{31}$) and "female" ($X_3 = x_{32}$). The biological causes (e.g., $X_1 = x_{11}$ and $X_2 = x_{21}$) of the two states are usually different and specified separately. This simple binary multi-valued case can be illustrated in Fig.2.

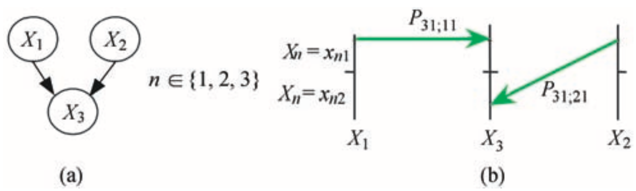


Fig.2. Illustration for the binary multi-valued case.

The essential difference between Fig.1(b) and Fig.2(b) is that in Fig.2(b), the causes of $X_3 = x_{31}$ and $X_3 = x_{32}$ are specified separately. Note that Figs. 1(a) and 2(a) are the same. This means that the essential difference between the two cases is hidden at the variable/node level. This concealment does not make sense in CPT representations, because the difference has been included in the parameters of CPTs; but it does make sense in compact representations, which will be explained later.

It is well known that the probabilities of all states of a variable must sum up to 1 in any case, because the states of a variable are exclusive and exhaustive. This probability law can be called *normalization*. The single-valued cases always satisfy the normalization, because the false state is just the complement of the true state. In multi-valued cases, however, the normalization is usually not satisfied because the individual conditional probabilities are given separately. As illustrated in Fig.2, suppose the probability of $X_1 = x_{11}$ causing $X_3 = x_{31}$ is given as $p_{31;11} \equiv \Pr\{P_{31;11}\} = 0.6$, and the probability of $X_2 = x_{21}$ causing $X_3 = x_{32}$ is given as $p_{32;21} \equiv \Pr\{P_{32;21}\} = 0.8$. If we simply treat every valued state of a multi-valued variable as a single-valued state, we have: $X_3 = x_{31}$ is irrelevant to X_2 and $X_3 = x_{32}$ is irrelevant to X_1, because the causes of $X_3 = x_{31}$ and $X_3 = x_{32}$ are specified separately. In other words, conditioned on $E = (X_1 = x_{11}) \cap (X_2 = x_{21})$, we have $\Pr\{X_3 = x_{31}|E\} = \Pr\{X_3 = x_{31}|X_1 = x_{11}\} = p_{31;11}$ and $\Pr\{X_3 = x_{32}|E\} = \Pr\{X_3 = x_{32}|X_2 = x_{21}\} = p_{32;21}$ separately for the two single-valued states. As $X_3 = x_{31}$ and $X_3 = x_{32}$ are exclusive, we further have

$$\begin{aligned}\Pr\{(X_3 = x_{31}) \cup (X_3 = x_{32})|E\} \\ &= \Pr\{X_3 = x_{31}|E\} + \Pr\{X_3 = x_{32}|E\} \\ &= p_{31;11} + p_{32;21} = 0.6 + 0.8 = 1.4 > 1,\end{aligned}$$

i.e., the normalization of X_3 is not satisfied. This is because $X_3 = x_{31}$ and $X_3 = x_{32}$ are correlated by the exclusion between them, while their causes are specified separately as if they were separately single-valued. The existing methods to solve this problem are usually to apply (1)[22]:

$$\begin{aligned}\Pr\{X_n = x_{nk}|E\} &= \frac{\Pr\{(X_n = x_{nk}) \cap E\}}{\sum_k \Pr\{(X_n = x_{nk}) \cap E\}} \\ &= \frac{\Pr\{X_n = x_{nk}|E\}}{\sum_k \Pr\{X_n = x_{nk}|E\}},\end{aligned} \tag{1}$$

in which, E represents any evidence or condition. However, (1) is valid only when $\sum_k \Pr\{X_n = x_{nk}|E\} = 1$ (the precondition), while it is the consequence to be achieved as shown in (2),

$$\begin{aligned}\Pr\{X_n = x_{nk}|E\} &= \frac{\Pr\{(X_n = x_{nk}) \cap E\}}{\Pr\{E\}} \\ &= \frac{\Pr\{(X_n = x_{nk}) \cap E\}}{\Pr\{E\}\sum_k \Pr\{X_n = x_{nk}|E\}} \\ &= \frac{\Pr\{E\}\Pr\{X_n = x_{nk}|E\}}{\Pr\{E\}\sum_k \Pr\{X_n = x_{nk}|E\}} \\ &= \frac{\Pr\{X_n = x_{nk}|E\}}{\sum_k \Pr\{X_n = x_{nk}|E\}}. \qquad (2)\end{aligned}$$

If we use the individual conditional probabilities to calculate $\Pr\{X_n = x_{nk}|E\}$ separately as in a single-valued case, $\sum_k \Pr\{X_n = x_{nk}|E\} = 1$ is usually not satisfied. The above example has shown this. Note that this problem does not exist in CPT representation, because $\sum_k \Pr\{X_m = x_{nk}|E\} = 1$ is always satisfied, where E represents a state combination of parent variables. Only in compact representations, will this imposed normalization problem exist.

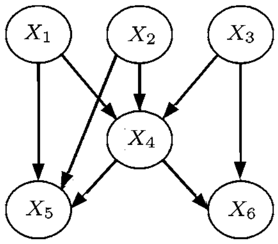
The underlining difficulty in multi-valued cases is that the compact representations have to solve the conflict between (a) satisfying the normalization of the exclusive (correlated) states of a child variable, and (b) specifying the causes of the multi-valued states (not necessarily all states) of a child variable separately. It is obvious that (a) has to be satisfied and (b) is required for compact representations. To solve this conflict, the present methods use (1). But mathematically, it is improper because the consequence of $\sum_k \Pr\{X_n = x_{nk}|E\} = 1$ is used as the precondition.

Following examples further illustrate four practical problems of applying (1) in multi-valued cases.

2.1 Unclearness

For the example shown in Fig.3, which is Fig.1 in [14], all variables are binary.

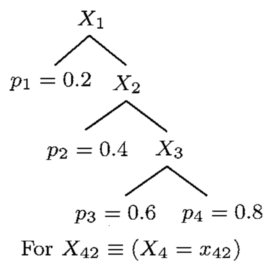
With the CSI representation, the left branch indicates true and the right branch indicates false. For simplicity, event $X_n = x_{nk}$ is briefly denoted as X_{nk}, e.g., $X_1 = x_{11}$ is denoted as X_{11}; the j-th state combination of the parent variables of X_4 is denoted as $SCPV_{4;j}$, e.g., $SCPV_{4;1} = X_{11}X_{21}X_{31}$, where the multiplication of events means logic AND. In this example, if only the causes of X_{41} are specified, while the causes of X_{42} are not specified separately, it is a single-valued case, because X_{42} is implicitly specified as the complement of X_{41}. Suppose we separately specify the causes of X_{42} as shown in Fig.4, not as the complement of X_{41}, the case becomes multi-valued. Note that X_1 is not a parent variable of X_{42}, while X_1 is a parent variable of X_{41}. In general, different states of a child variable may have different parent variables in multi-valued cases.



(a)

j	$SCPV_{4;j}$	$\Pr\{X_{41}\|SCPV_{4;j}\}$
1	$X_{11}X_{21}X_{31}$	$p_1 = 0.2$
2	$X_{11}X_{21}X_{32}$	$p_1 = 0.2$
3	$X_{11}X_{22}X_{31}$	$p_1 = 0.2$
4	$X_{11}X_{22}X_{32}$	$p_1 = 0.2$
5	$X_{12}X_{21}X_{31}$	$p_2 = 0.4$
6	$X_{12}X_{21}X_{32}$	$p_2 = 0.4$
7	$X_{12}X_{22}X_{31}$	$p_3 = 0.6$
8	$X_{12}X_{22}X_{32}$	$p_4 = 0.8$

(b)



(c)

Fig.3. CSI specification for the causes of $X_{41} \equiv (X_4 = x_{41})$.

It is obvious that $\Pr\{X_{41}|SCPV_{4;j}\} + \Pr\{X_{42}|SCPV_{4;j}\} \neq 1$ by combining Figs. 3 and 4. Now we look at what happens when we apply (1). Denote $E_j = SCPV_{4;j}$, (1) can be further written as:

$$\Pr\{X_{nk}|E_j\} = \frac{\Pr\{X_{nk}|E_j\}}{\sum_k \Pr\{X_{nk}|E_j\}} = \alpha_{n;j}\Pr\{X_{nk}|E_j\}, \qquad (3)$$

$$\alpha_{n;j} \equiv 1\Big/\sum_k \Pr\{X_{nk}|E_j\}, \qquad (4)$$

in which, $\alpha_{n;j}$ is called *normalization factor*. In both (3) and (4), the $\Pr\{X_{nk}|E_j\}$ on the right side is the separately calculated conditional probability and the $\Pr\{X_{nk}|E_j\}$ on the left side of (3) is the normalized conditional probability. Note that in (4), $\alpha_{n;j}$ is not a constant but a variable depending on $E_j = SCPV_{n;j}$. According to (3) and (4), the CPT and $\alpha_{n;j}$ can be calculated as shown in Table 1.

It is seen that the calculated CPT shown in Table 1 is based on so many different $\alpha_{4;j}$, $j \in \{1, \ldots, 8\}$. In

j	$SCPV_{4;j}$	$\Pr\{X_{42}\|SCPV_{4;j}\}$
1	$X_{11}X_{21}X_{31}$	$p_5 = 0.4$
2	$X_{11}X_{21}X_{32}$	$p_6 = 0.2$
3	$X_{11}X_{22}X_{31}$	$p_7 = 0.7$
4	$X_{11}X_{22}X_{32}$	$p_7 = 0.7$
5	$X_{12}X_{21}X_{31}$	$p_5 = 0.4$
6	$X_{12}X_{21}X_{32}$	$p_6 = 0.2$
7	$X_{12}X_{22}X_{31}$	$p_7 = 0.7$
8	$X_{12}X_{22}X_{32}$	$p_7 = 0.7$

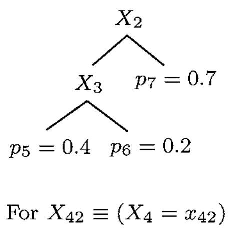


Fig.4. CSI specification for the causes of $X_{42} \equiv (X_4 = x_{42})$.

Table 1. CPT and $\alpha_{n;j}$ Calculated from (3) and (4)

j	$E_j = SCPV_{4;j}$	$\Pr\{X_{41}\|E_j\}$	$\Pr\{X_{42}\|E_j\}$	$\alpha_{4;j}$
1	$X_{11}X_{21}X_{31}$	1/3	2/3	1/0.6
2	$X_{11}X_{21}X_{32}$	1/2	1/2	1/0.4
3	$X_{11}X_{22}X_{31}$	2/9	7/9	1/0.9
4	$X_{11}X_{22}X_{32}$	2/9	7/9	1/0.9
5	$X_{12}X_{21}X_{31}$	1/2	1/2	1/0.8
6	$X_{12}X_{21}X_{32}$	2/3	1/3	1/0.6
7	$X_{12}X_{22}X_{31}$	6/13	7/13	1/1.3
8	$X_{12}X_{22}X_{32}$	8/15	7/15	1/1.5

general, the number of $\alpha_{n;j}$ equals the number of $E_j = SCPV_{n;j}$, which can be huge. For the example of five states and five parent variables, the number of $SCPV_{n;j}$ is $5^5 = 3\,125$. This is too many for domain engineers to realize when they specify the causes and parameters of the states of child variable X_n separately. The questions are: Why do we need so many implicit and different normalization factors? Are these different normalization factors realized by domain engineers when they specify the causes and parameters for the multi-valued states separately? In other words, are these different normalization factors what the domain engineers want? Do these implicit factors represent the knowledge of domain engineers? It seems that these questions have not been clearly realized and answered when people apply (1) or (3) and (4). Therefore, simply treating every valued state of a multi-valued variable as a single-valued state is questionable (as an approximation may be acceptable but is not concerned in this paper).

2.2 Inconsistency

If we change the values of p_i in Figs. 3 and 4 (the old set of p_i in Table 2) as the new set of p_i in Table 2, the calculated CPT remains the same as in Table 1, while $\alpha_{n;j}$ changes.

This is another problem that domain engineers may not realize. In fact, although the two sets of p_i correspond to a same CPT, they have different influences in the probability propagation through causality chains when we apply the chaining inference algorithms. To illustrate this, consider the BN for the refrigerator temperature shown in Fig.5, in which the events are defined as follows:

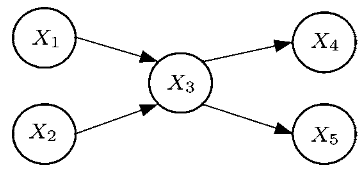


Fig.5. BN for a refrigerator temperature.

$X_{11} = \{\text{coolant leakage}\}$;
$X_{12} = \{\text{no coolant leakage}\}$;
$X_{21} = \{\text{temperature sensor failure}\}$;
$X_{22} = \{\text{no temperature sensor failure}\}$;
$X_{31} = \{\text{high temperature}\}$;
$X_{32} = \{\text{low temperature}\}$;
$X_{33} = \{\text{normal temperature}\}$;
$X_{41} = \{\text{food spoil}\}$;
$X_{42} = \{\text{no food spoil}\}$;
$X_{51} = \{\text{high power consumption}\}$;
$X_{52} = \{\text{no high power consumption}\}$.

In terms of CSI, the causes of X_{31}, X_{32}, X_{33} and X_{41} can be specified as shown in Fig.6, in which all variables split from left to right according to the state index sequence $(1, 2, \ldots)$.

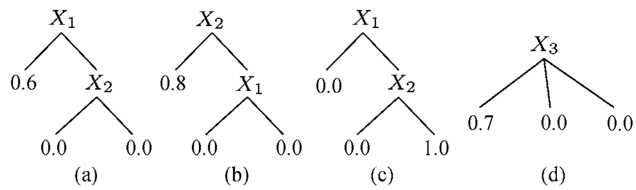


Fig.6. CSI representations for X_{31}, X_{32}, X_{33} and X_{41}. (a) For X_{31}. (b) For X_{32}. (c) For X_{33}. (d) For X_{41}.

As shown in Fig.6, X_{31} (abnormally high temperature) can be caused by X_{11} (coolant leakage) with probability 0.6; X_{32} (abnormally low temperature) can be caused by X_{21} (temperature sensor failure) with probability 0.8; X_{41} (food spoil) can be caused by X_{31} with probability 0.7. Moreover, X_{51} (abnormally high power consumption) can be caused by X_{32} with probability 0.5; X_{42} (no food spoil) is the complement of X_{41} (food spoil), i.e., $X_{42} = \overline{X}_{41}$; X_{52} (no high power consumption) is the complement of X_{51}, i.e., $X_{52} = \overline{X}_{51}$. It is obvious that X_3 is multi-valued, while X_4 and X_5 are single-valued, because *more than one* states of X_3 are specified separately, while *only one* state of X_4 and X_5

Table 2. Comparison Between Two Sets of p_i

		Old Set of p_i			New Set of p_i		
j	$E_j = SCPV_{4;j}$	$\Pr\{X_{41}\|E_j\}$	$\Pr\{X_{42}\|E_j\}$	$\alpha_{4;j}$	$\Pr\{X_{41}\|E_j\}$	$\Pr\{X_{42}\|E_j\}$	$\alpha_{4;j}$
1	$X_{11}X_{21}X_{31}$	$p_1 = 0.2$	$p_5 = 0.4$	1/0.6	$p_1 = 0.1$	$p_5 = 0.20$	1/0.30
2	$X_{11}X_{21}X_{32}$	$p_1 = 0.2$	$p_6 = 0.2$	1/0.4	$p_1 = 0.1$	$p_6 = 0.10$	1/0.20
3	$X_{11}X_{22}X_{31}$	$p_1 = 0.2$	$p_7 = 0.7$	1/0.9	$p_1 = 0.1$	$p_7 = 0.35$	1/0.45
4	$X_{11}X_{22}X_{32}$	$p_1 = 0.2$	$p_7 = 0.7$	1/0.9	$p_1 = 0.1$	$p_7 = 0.35$	1/0.45
5	$X_{12}X_{21}X_{31}$	$p_2 = 0.4$	$p_5 = 0.4$	1/0.8	$p_2 = 0.2$	$p_5 = 0.20$	1/0.40
6	$X_{12}X_{21}X_{32}$	$p_2 = 0.4$	$p_6 = 0.2$	1/0.6	$p_2 = 0.2$	$p_6 = 0.10$	1/0.30
7	$X_{11}X_{22}X_{31}$	$p_3 = 0.6$	$p_7 = 0.7$	1/1.3	$p_3 = 0.3$	$p_7 = 0.35$	1/0.65
8	$X_{11}X_{22}X_{32}$	$p_4 = 0.8$	$p_7 = 0.7$	1/1.5	$p_4 = 0.4$	$p_7 = 0.35$	1/0.75

are specified respectively. Given $E_j = X_{11}X_{21}$, if we treat X_{31}, X_{32} and X_{33} as three single-valued states separately, according to Fig.6, we have

$$\Pr\{X_{31}|X_{11}X_{21}\} = 0.6, \quad (5)$$
$$\Pr\{X_{32}|X_{11}X_{21}\} = 0.8, \quad (6)$$
$$\Pr\{X_{33}|X_{11}X_{21}\} = 0.0. \quad (7)$$

By applying (1), the CPT of X_3 conditioned on $X_{11}X_{21}$ can be calculated as follows:

$$\Pr\{X_{31}|X_{11}X_{21}/\sum_k \Pr\{X_{3k}|X_{11}X_{21}\} = 0.6/(0.6+0.8+0) = 0.428\,6, \quad (8)$$

$$\Pr\{X_{32}|X_{11}X_{21}\}/\sum_k \Pr\{X_{3k}|X_{11}X_{21}\} = 0.8/(0.6+0.8+0) = 0.571\,4, \quad (9)$$

$$\Pr\{X_{33}|X_{11}X_{21}\}/\sum_k \Pr\{X_{3k}|X_{11}X_{21}\} = 0/(0.6+0.8+0) = 0.0. \quad (10)$$

Suppose we change the parameters in Fig.6 or (5)∼(7) as

$$\Pr\{X_{31}|X_{11}X_{21}\} = 0.3, \quad (11)$$
$$\Pr\{X_{32}|X_{11}X_{21}\} = 0.4, \quad (12)$$
$$\Pr\{X_{33}|X_{11}X_{21}\} = 0.0. \quad (13)$$

By applying (1), we still have

$$\Pr\{X_{31}|X_{11}X_{21}\}/\sum_k \Pr\{X_{3k}|X_{11}X_{21}\} = 0.3/(0.3+0.4+0) = 0.428\,6, \quad (14)$$

$$\Pr\{X_{32}|X_{11}X_{21}\}/\sum_k \Pr\{X_{3k}|X_{11}X_{21}\} = 0.4/(0.3+0.4+0) = 0.571\,4, \quad (15)$$

$$\Pr\{X_{33}|X_{11}X_{21}\}/\sum_k \Pr\{X_{3k}|X_{11}X_{21}\} = 0/(0.3+0.4+0) = 0.0. \quad (16)$$

That is, the calculated CPT of X_3 remains unchanged.

Now, look at the calculations of $\Pr\{X_{41}|X_{11}X_{21}\}$ and $\Pr\{X_{42}|X_{11}X_{21}\}$. Note that X_4 is single-valued and (1) is not needed for satisfying the normalization. According to Fig.5, we have

$$\begin{aligned}\Pr\{X_{41}|X_{11}X_{21}\} = &\Pr\{X_{41}|X_{31}\}\Pr\{X_{31}|X_{11}X_{21}\} + \\ &\Pr\{X_{41}|X_{32}\}\Pr\{X_{32}|X_{11}X_{21}\} + \\ &\Pr\{X_{41}|X_{33}\}\Pr\{X_{33}|X_{11}X_{21}\}.\end{aligned} \quad (17)$$

According to Fig.6(d), we have $\Pr\{X_{41}|X_{31}\} = 0.7$, $\Pr\{X_{41}|X_{32}\} = 0.0$ and $\Pr\{X_{41}|X_{33}\} = 0.0$. Thus, (17) becomes

$$\Pr\{X_{41}|X_{11}X_{21}\} = 0.7\Pr\{X_{31}|X_{11}X_{21}\}. \quad (18)$$

Now we have two choices to apply the value of $\Pr\{X_{31}|X_{11}X_{21}\}$. One is to use the CPT value as shown in (8) or (14). Then we have

$$\Pr\{X_{41}|X_{11}X_{21}\} = 0.7 \times 0.428\,6 = 0.3, \quad (19)$$
$$\Pr\{X_{42}|X_{11}X_{21}\} = 1 - \Pr\{X_{41}|X_{11}X_{21}\} = 1 - 0.3 = 0.7. \quad (20)$$

(20) is because X_4 is single-valued. This choice means to base our inference only on the calculated CPT and give up the chaining inference algorithm associated with the compact representation. Obviously, this giving up is not desired.

Another choice is to use the value shown in (5) or (11), which means to propagate the probability calculated from the compact representation directly through the causality chain before applying (1). By using (5), (19) and (20) become

$$\Pr\{X_{41}|X_{11}X_{21}\} = 0.7 \times 0.6 = 0.42, \quad (21)$$
$$\Pr\{X_{41}|X_{11}X_{21}\} = 1 - \Pr\{X_{41}|X_{11}X_{21}\} = 1 - 0.42 = 0.58. \quad (22)$$

By using (11), (19) and (20) become

$$\Pr\{X_{41}|X_{11}X_{21}\} = 0.7 \times 0.3 = 0.21, \quad (23)$$
$$\Pr\{X_{41}|X_{11}X_{21}\} = 1 - \Pr\{X_{41}|X_{11}X_{21}\} = 1 - 0.21 = 0.79. \quad (24)$$

It is seen that the results of (19)∼ (20), (21)∼(22) and (23)∼(24) are different. In other words, although the two sets of parameters shown in (5)∼(7) and (11)∼(13) correspond to a same CPT, the inference results are different. This means that the two sets of parameters have different influence on the probability propagation through the causality chains. This different influence may be what domain engineers really want when they specify different set of parameters. However, the different results are inconsistent with each other.

2.3 Inexactness

To apply (1), it is necessary to calculate the conditional probabilities of all states of a multi-valued variable separately, where every valued state of the multi-valued variable must be treated as a single-valued state. And then, all states of a multi-valued variable must be specified with their causes separately. For example, the causes of X_{31}, X_{32} and X_{33} must be specified in Figs. 6(a), 6(b) and 6(c) separately.

However, the separate specification shown in Fig.6(c) for the cause of X_{33} is inexact, because $X_{33} \neq X_{12}X_{22}$. For example, even though both X_{11} and X_{21} are true, X_{33} may still be true, because $X_{11}X_{21}$ is not enough to cause X_{31} or X_{32}. The exact representation should be $X_{33} = \overline{X}_{31}\overline{X}_{32}$. In other words, X_{33} is the complement of $X_{31} + X_{32}$, where "+" means XOR. That is, $\Pr\{X_{33}|X_{11}X_{21}\} = 1 - \Pr\{X_{31}|X_{11}X_{21}\} - \Pr\{X_{32}|X_{11}X_{21}\}$.

Note that the complement is not a separate specification and cannot be applied in (1). Otherwise, a conflict will appear. For the above example, to calculate $\Pr\{X_{33}|X_{11}X_{21}\}$, we have to know the normalized $\Pr\{X_{31}|X_{11}X_{21}\}$ and $\Pr\{X_{32}|X_{11}X_{21}\}$; but to calculate the normalized $\Pr\{X_{31}|X_{11}X_{21}\}$ and $\Pr\{X_{32}|X_{11}X_{21}\}$, we have to know $\Pr\{X_{33}|X_{11}X_{21}\}$. We cannot use the values of $\Pr\{X_{31}|X_{11}X_{21}\}$ and $\Pr\{X_{32}|X_{11}X_{21}\}$ before normalization to calculate $\Pr\{X_{33}|X_{11}X_{21}\}$, because according to (5) and (6),

$$\begin{aligned}\Pr\{X_{33}|X_{11}X_{21}\} = 1 - \Pr\{X_{31}|X_{11}X_{21}\} - \\ \Pr\{X_{32}|X_{11}X_{21}\} = 1 - 0.6 - 0.8 = -0.4.\end{aligned} \quad (25)$$

This value is unreasonable. Hence, if we insist on applying (1), the inexact representation may be unavoidable.

2.4. Impracticalness

By using (11) and (12), (25) is changed as

$$\begin{aligned}\Pr\{X_{33}|X_{11}X_{21}\} = 1 - \Pr\{X_{31}|X_{11}X_{21}\} - \\ \Pr\{X_{32}|X_{11}X_{21}\} = 1 - 0.3 - 0.4 = 0.3.\end{aligned} \quad (26)$$

This is an acceptable value. In spite of this, however, in addition to the risk of (25), this approach is still impractical when more than one states are specified as the complement of other states. For the above example, if X_{32} and X_{33} are both specified as the complement of $X_{11} + X_{33}$ and $X_{11} + X_{32}$ respectively, the approach shown in (26) is impractical, because

$$\begin{aligned}&\Pr\{X_{32}|X_{11}X_{21}\} = 1 - \Pr\{X_{31}|X_{11}X_{21}\} - \\ &\quad \Pr\{X_{33}|X_{11}X_{21}\} = 1 - 0.3 - \Pr\{X_{33}|X_{11}X_{21}\}, \\ &\Pr\{X_{33}|X_{11}X_{21}\} = 1 - \Pr\{X_{31}|X_{11}X_{21}\} - \\ &\quad \Pr\{X_{32}|X_{11}X_{21}\} = 1 - 0.3 - \Pr\{X_{32}|X_{11}X_{21}\}.\end{aligned}$$

It should be pointed out that the domain engineers usually pay attention to only the causes of the states in concern (e.g., X_{31} as the only meaningful cause of X_{41} expressed in (18)), but not those not in concern. This results in that the separate specifications for the causes of the states not in concern may be impractical, not only because of the unnecessary and difficult work, but also because domain engineers may not know how to specify these causes separately, not as the complement of other states.

3 S-DUCG Model Applicable in Single-Valued Cases

In this section, the dynamic causality diagram (DCD)[31] is introduced, which provides the basis of the dynamic uncertain causality graph (DUCG). Before introducing DCD and presenting DUCG, a new set of notations are defined as follows.

In DCD/DUCG, the uppercase letters denote variables or events, the lowercase letters denote the probabilities of the corresponding events, the first subscript of a variable/event indexes the variable, and the second subscript indexes the state of the variable. Obviously, a state of a variable is an event.

For example, X_n denotes a variable indexed by n, X_{nk} denotes the k-th state of variable X_n or the event that X_n is in its state k. Correspondingly, $x_{nk} \equiv \Pr\{X_{nk}\}$. The difference between variable X_n and event X_{nk} is that X_{nk} has two subscripts. They can be separated by ",". But in this paper, "," is ignored for simplicity without confusion.

Since this section discusses only the single-valued cases, all variables except logic gates are binary. Readers should note that some expressions in this section may not be valid in multi-valued cases.

3.1 Introduction to DCD Model

Although DCD[31] was presented two years earlier than CSI[14] and ICI[15], DCD is not well known in the community, while noisy-OR, CSI and ICI are well known. For simplicity, this subsection will introduce DCD by briefly comparing DCD with noisy-OR and CSI, so as to help readers understand DCD. Note that DCD was originally presented in [31], not in this paper. Therefore, the systematical comparison between DCD and other models is not the purpose of this paper.

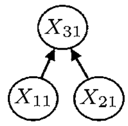
The well known noisy-OR can be illustrated with the example shown in Fig.7(c) that is similar to Fig.4.20 in [6], in which state 1 denotes true and state 2 denotes false. In this example, $\Pr\{X_{31}|X_{11}X_{22}\} = 0.3$ and $\Pr\{X_{31}|X_{12}X_{21}\} = 0.6$ (the two expressions are valid only in the single-valued case).

According to noisy-OR, the CPT of Fig.7(b) is calculated as shown in Fig.7(a).

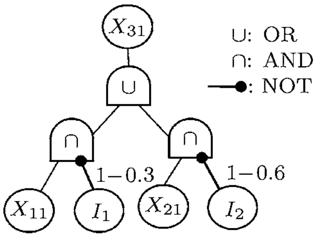
It is seen that only two parameters 0.3 and 0.6 are needed in noisy-OR to represent the eight parameters in the CPT. However, for the CPT shown in Fig.8(a), noisy-OR is not convenient, because X_{11} and X_{21} are

	X_{31}	X_{32}
$X_{11}X_{21}$	0.72	0.28
$X_{11}X_{22}$	0.30	0.70
$X_{12}X_{21}$	0.60	0.40
$X_{12}X_{22}$	0.00	1.00

(a)



(b)

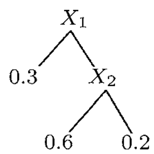


(c)

Fig.7. Illustration for noisy-OR.

	X_{31}	X_{32}
$X_{11}X_{21}$	0.3	0.7
$X_{11}X_{22}$	0.3	0.7
$X_{12}X_{21}$	0.6	0.4
$X_{12}X_{22}$	0.2	0.8

(a)



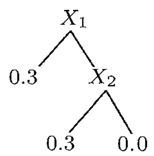
(b)

Fig.8. Illustration for CSI.

not in OR relation. CSI provides a solution as shown in Fig.8(b) in which the left branch represents state 1 (true) and the right branch represents state 2 (false). It is seen that the eight parameters in the CPT are reduced to three: 0.3, 0.6 and 0.2. Nevertheless, for the CPT shown in Fig.9(a), although CSI is applicable and the result is shown in Fig.9(b), the representation is not compact enough, because the real meaningful parameter is only one: 0.3. In DCD, this case can be represented as shown in Fig.10.

	X_{31}	X_{32}
$X_{11}X_{21}$	0.3	0.7
$X_{11}X_{22}$	0.3	0.7
$X_{12}X_{21}$	0.3	0.7
$X_{12}X_{22}$	0.0	1.0

(a)



(b)

Fig.9. Another example of CSI.

Fig.10 explicitly represents that any one of X_{11} and X_{21} or they together may cause X_{31} with probability 0.3. That is what the CPT in Fig.9(a) really tells us. In fact, the cases shown in Figs. 7 and 8 can also be represented by DCD as shown in Figs. 11(a) and 11(c) respectively. In Fig.11(a), the default logic relation between X_{11} and X_{21} is defined as OR (see [31] for details).

A logic gate variable denoted as G_4 is used in Fig.11 (b) to specify the complex logic relation between X_1 and X_2. In this example, G_4 is a special parent

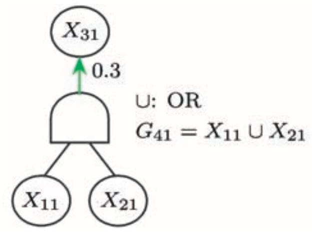


Fig.10. DCD representation of Fig.9.

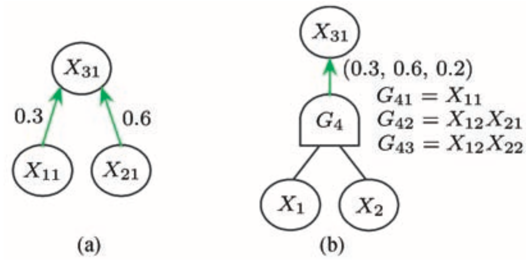


Fig.11. DCD representation of Figs. 7 and 8 respectively.

variable of X_3 and has three exclusive states denoted as G_{41}, G_{42} and G_{43} respectively. The three states of G_4 can cause X_{31} with probabilities 0.3, 0.6 and 0.2 respectively. As an extension to DCD, this paper presents that any logic gate in DCD can be specified as shown in Table 3. This table is called the logic gate specification (LGS).

Table 3. Logic Gate Specification (LGS_i)

j	G_{ij}
1	Event expression 1
2	Event expression 2
$\vdots$	$\vdots$
m	Event expression m

For the example of Fig.10, LGS_4 is specified as $G_{41} = X_{11} \cup X_{21}$, where G_4 represents the OR gate variable and has only one *active* state, while G_4 in Fig.11(b) has three *active* states. It is seen that the graphical symbols of the logic gates for different logic relations do not have to be different. We only need to draw a logic gate as G_i and specify LGS_i as illustrated in Table 3, in which i is the index of the logic gate and is different from the indexes of other variables. Thus, we can use a same symbol to represent all type logic gates.

Obviously, Fig.11(a) is equivalent to Fig.7(c). What Fig.11(a) tells us is that X_{11} and X_{21} can cause X_{31} independently and the relation between X_{11} and X_{21} is OR (the default relation defined in DCD).

In DCD, the conditional probabilities $\Pr\{X_{31}|X_{11}X_{22}\}$ and $\Pr\{X_{31}|X_{12}X_{21}\}$ in Fig.11(a) are viewed as the probabilities of the *independent* linkage events $P_{31;11}$ and $P_{31;21}$ respectively, i.e., $\Pr\{X_{31}|X_{11}X_{22}\} =$

$\Pr\{P_{31;11}\} = p_{31;11}$ and $\Pr\{X_{31}|X_{12}X_{21}\} = \Pr\{P_{31;21}\} = p_{31;21}$. In fact, when people give $\Pr\{X_{31}|X_{11}X_{22}\}$, they think only X_{11} but not X_{22}. They are even not aware of the existence of variable X_2. The situation for $\Pr\{X_{31}|X_{12}X_{21}\}$ is the same. This is because X_1 and X_2 are usually in different domains (e.g., the burglary and earthquake are different domains, while both can cause the vibration invoking alarm). This can be further interpreted as shown in Fig.12.

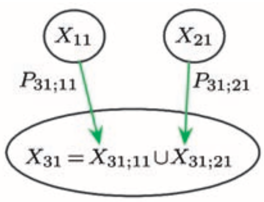


Fig.12. Interpretation for Fig.11(a).

In Fig.12, X_{31} can be caused by either $X_{31;11}$ or $X_{31;21}$. $X_{31;11}$ denotes the event that X_{31} is caused by X_{11} only; $X_{31;21}$ denotes the event that X_{31} is caused by X_{21} only. There is an uncertain physical mechanism between $X_{31;i1}$ and X_{i1}, $i \in \{1, 2\}$. Given X_{i1}, when the corresponding mechanism functions, $X_{31;i1}$ occurs; otherwise, $X_{31;i1}$ does not occur. This independent random uncertain physical mechanism is represented by an independent random event $P_{31;i1}$. Thus, $X_{31;i1} = P_{31;i1}X_{i1}$. Only when both $P_{31;i1}$ and X_{i1} occur, will $X_{31;i1}$ occur.

Definition 2 (Some Variable Types in DCD and DUCG). *"X" represents the consequence or effect variable drawn as circle. It can also be a cause variable. "B" represents the basic or root variable drawn as square and can only be an independent cause. "G" represents the logic gate variable. "P" represents the linkage event, and* $\boldsymbol{P}_{n1;i}$ *represents the event vector* $(P_{n1;i1}, P_{n1;i2}, \ldots, P_{n1;im})$.

Fig.13 is an illustration for the use of these variables.

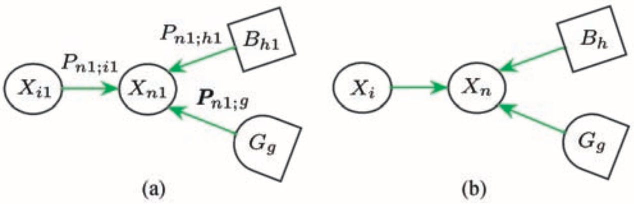


Fig.13. Illustration for some type of variables/events.

As illustrated in Fig.13(a), $P_{n1;i1}$, $P_{n1;h1}$ and $\boldsymbol{P}_{n1;g}$ represent the directed arc from parent events/variable X_{i1}, B_{h1} and G_g to child event X_{n1} respectively. As a single-valued variable, X_n has only one valued state: X_{n1}. As the parent variables of X_{n1}, X_i has one active state X_{i1}, B_h has one active state B_{h1}, and G_g has more than one active state. That is why events $P_{n1;i1}$ and $P_{n1;h1}$ and event vector $\boldsymbol{P}_{n1;g}$ are used respectively. For simplicity, $P_{n1;i1}$, $P_{n1;h1}$, $\boldsymbol{P}_{n1;g}$ and the states of variables are usually ignored in the graph as shown in Fig.13(b). The detailed information is hidden in the directed arcs and the logic gate. It should be noted that in single-valued cases, although parent variables may have multiple *active* states, the child variable has only one *valued* state (X_{n1} in this example). Only the causes of the single-valued state of a child variable should be specified by directed arcs. Otherwise, the case is multi-valued.

With the P type events, we can express the uncertain causalities between X_{31} and its parents X_{11} and X_{21} in Fig.11(a) as an event expression in the form of sum-of-products:

$$X_{31} = X_{31;11} \cup X_{31;21} = P_{31;11}X_{11} \cup P_{31;21}X_{21}. \quad (27)$$

Conditioned on $X_{11}X_{21}$, we have $X_{31}|X_{11}X_{21} = P_{31;11} \cup P_{31;21}$. By applying the well known inclusive-exclusive principle or De Morgan's laws of probabilities, we have

$$\begin{aligned}
&\Pr\{X_{31}|X_{11}X_{21}\} = \Pr\{P_{31;11} \cup P_{31;21}\} \\
&\quad = \Pr\{P_{31;11}\} + \Pr\{P_{31;21}\} - \Pr\{P_{31;11}\}\Pr\{P_{31;21}\} \\
&\quad = p_{31;11} + p_{31;21} - p_{31;11}p_{31;21} \\
&\quad = 0.3 + 0.6 - 0.3 \times 0.6 = 0.72,
\end{aligned}$$

or

$$\begin{aligned}
&\Pr\{X_{31}|X_{11}X_{21}\} = \Pr\{\overline{\overline{X}}_{31}|X_{11}X_{21}\} \\
&\quad = 1 - \Pr\{\overline{X}_{31}|X_{11}X_{21}\} \\
&\quad = 1 - \Pr\{\overline{P_{31;11}X_{11} \cup P_{31;21}X_{21}}|X_{11}X_{21}\} \\
&\quad = 1 - \Pr\{\overline{P_{31;11} \cup P_{31;21}}\} \\
&\quad = 1 - \Pr\{\overline{P}_{31;11}\overline{P}_{31;21}\} \\
&\quad = 1 - \Pr\{\overline{P}_{31;11}\}\Pr\{\overline{P}_{31;21}\} \\
&\quad = 1 - (1 - 0.3)(1 - 0.6) = 0.72.
\end{aligned}$$

The latter is what noisy-OR tells us. In noisy-OR, $1 - p_{nk;ij} \equiv \Pr\{\overline{P}_{nk;ij}\}$ is viewed as the probability of the inhibitor I_i as shown in Fig.7(c).

We can treat X_{11} and X_{21} as new child events and work out their event expressions as we have done for X_{31} in (27). This process can continue until the B type variables are reached, given that the graph is a DAG. This process is called event outspread. Any event or event group in any logic relation can be outspreaded as such event expressions. During the outspread, the various event algorithms, such as AND, OR, XOR, NOT, absorption, exclusion, complement, can be applied. For example, suppose our query is $\Pr\{H_{kj}|E\} = ?$, where H_{kj} denotes a hypothesis event or event expression in concern (e.g., $H_{11} = B_{12}$, $H_{21} = X_{21}B_{13}$, $H_{22} = X_{22}B_{12}$, etc.), and E denotes a group of events in the AND relation. We have

$$\Pr\{H_{kj}|E\} = \frac{\Pr\{H_{kj}E\}}{\Pr\{E\}}. \tag{28}$$

We can outspread $H_{kj}E$ and E as two event expressions composed of only P and B type events by applying the various event algorithms, and then calculate the probabilities of the two event expressions by simply replacing these P and B type events with their prior probabilities (lowercase letters p and b respectively). Of course, the event expressions must be in the form of disjoint/exclusive sum-of-products.

To get the disjoint sum-of-products, we can apply the following algorithm (see [31] for details):

$$C_1 \cup C_2 \cup \cdots \cup C_n = C_1 + \overline{C}_1C_2 + \overline{C}_1\overline{C}_2C_3 + \cdots + \overline{C}_1\overline{C}_2\cdots\overline{C}_{n-1}C_n, \tag{29}$$

$$\overline{C} = \overline{V_{1j_1}V_{2j_2}\cdots V_{mj_m}} = \overline{V}_{1j_1} + V_{1j_1}\overline{V}_{2j_2} + V_{1j_1}V_{2j_2}\overline{V}_{3j_3} + \cdots + V_{1j_1}\cdots V_{m-1j_{m-1}}\overline{V}_{mj_m}, \tag{30}$$

where "+" denotes XOR, $V \in \{X, P, G, B\}$, $C = V_{1j_1}V_{2j_2}\cdots V_{mj_m}$, and j_i is the second subscript of variable V_i. C is usually called cutset that is an event product at any event outspread level. By repeatedly applying (29) and (30), meanwhile applying the event absorption and exclusion, etc., we can get the disjoint sum-of-products composed of only P and B type events. In this way, we divide the computation as two steps: 1) event outspread; 2) numerical calculation. Sometimes, only the first step is needed. For some diagnostic case, we may find that only $H_{kj}E \neq 0$ while all $H_{gy}E = 0$, where H_{gy} represents all the other hypothesis events or event expressions in concern. Then we can conclude that H_{kj} is the only possible hypothesis event in concern without any numerical calculation, which means that the probability parameters are not needed. In fact, this two step approach can bring us a lot of benefits in knowledge representation and inference.

Before ending the introduction to DCD, it should be noted that DCD is capable of representing various complex uncertain causalities such as but not limited to those shown in Figs. 14∼21.

	X_{31}	X_{32}
$X_{11}X_{21}$	0.7	0.3
$X_{11}X_{22}$	0.0	1.0
$X_{12}X_{21}$	0.0	1.0
$X_{12}X_{22}$	0.0	1.0

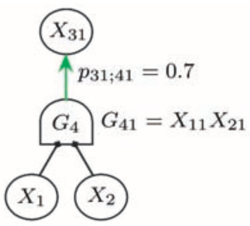


Fig.14. AND logic gate.

	X_{31}	X_{32}
$X_{11}X_{21}$	0.0	1.0
$X_{11}X_{22}$	0.0	1.0
$X_{12}X_{21}$	0.0	1.0
$X_{12}X_{22}$	0.7	0.3

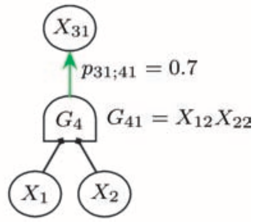


Fig.15. NOT logic gate.

	X_{31}	X_{32}
$X_{11}X_{21}$	0.0	1.0
$X_{11}X_{22}$	0.7	0.3
$X_{12}X_{21}$	0.7	0.3
$X_{12}X_{22}$	0.0	1.0

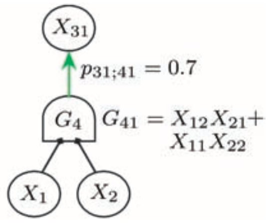


Fig.16. XOR logic gate.

	X_{31}	X_{32}
$X_{11}X_{21}$	0.0	1.0
$X_{11}X_{22}$	0.7	0.3
$X_{12}X_{21}$	0.7	0.3
$X_{12}X_{22}$	0.7	0.3

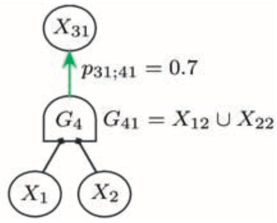


Fig.17. NOT-AND logic gate.

	X_{31}	X_{32}
$X_{11}X_{21}$	0.7	0.3
$X_{11}X_{22}$	0.0	1.0
$X_{12}X_{21}$	0.0	1.0
$X_{12}X_{22}$	0.7	0.3

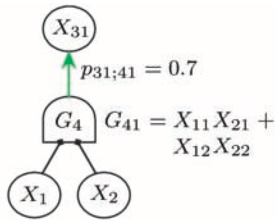


Fig.18. NOT-XOR logic gate.

	X_{31}	X_{32}
$X_{11}X_{21}X_{31}$	0.7	0.3
$X_{11}X_{21}X_{32}$	0.7	0.3
$X_{12}X_{22}X_{31}$	0.7	0.3
$X_{12}X_{21}X_{32}$	0.7	0.3
Remnant	0.0	1.0

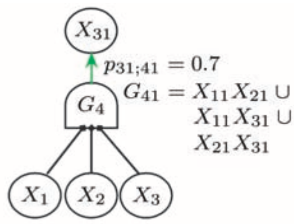


Fig.19. 2/3 logic gate.

	X_{31}	X_{32}
$X_{11}X_{21}$	0.72	0.28
$X_{11}X_{22}$	0.30	0.70
$X_{12}X_{21}$	0.00	1.00
$X_{12}X_{22}$	0.00	1.00

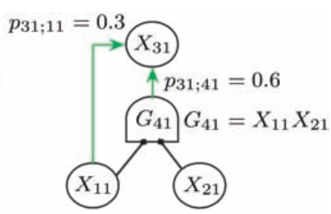


Fig.20. Special log combination.

j		X_{31}	X_{32}
1	$X_{11}X_{21}$	0.3	0.7
2	$X_{11}X_{22}$	0.6	0.4
3	$X_{12}X_{21}$	0.2	0.8
4	$X_{12}X_{22}$	0.1	0.9

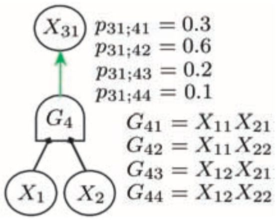


Fig.21. Completely combined logic gate.

Theoretically, any CPT can be represented by DCD with a completely combined logic gate as illustrated in Fig.21, in which j indexes the state combinations of parent variables. A completely combined logic gate is equivalent to a CPT, which is the worst case and no compactness is achieved. However, the completely combined logic gate does show the ability of DCD to represent complex uncertain causalities in the way as compact as possible.

Note that a logic gate can be the input of other logic gates, more than one logic gate can be the parents of a same child variable, and the logic gates can be partial parents of a child variable (e.g., Fig.20). Therefore, logic gate is a flexible tool for the compact representation of complex logic relation among variables.

3.2 S-DUCG Extended From DCD

The S-DUCG model is developed from DCD by adding additional properties: the conditional linkage events and the default events.

3.2.1 Conditional Linkage Events

It is interesting to note that in Fig.20, the simultaneous occurrence of X_{11} and X_{21} causes X_{31} with the probability 0.72 rather than 0.6, while 0.6 might be intuitively conceived. The intuitive idea of Fig.20 might be: when only X_{11} occurs, X_{31} may occur with probability 0.3; when $X_{11}X_{21}$ occurs, X_{31} may occur with probability 0.6. However, as the relation between $P_{31;11}$ and $P_{31;41}$ is OR, the probability of X_{31} caused by $X_{11}X_{21}$ is increased from 0.6 to 0.72, which may not be what people want to represent. To avoid this increase, Fig.20 can be modified as Fig.22, in which $\dashrightarrow$ is defined as a conditional linkage event. The condition of $P_{31;11}$ is denoted as $Z_{31;11} = \overline{X}_{21} = X_{22}$. That is, when X_{21} does not exist, $P_{31;11}$ exists; otherwise, $P_{31;11}$ does not exist. Here, $P_{31;11}$ is associated with $Z_{31;11}$ and can be expressed as $P_{31;11}Z_{31;11} = P_{31;11}X_{22}$.

	X_{31}	X_{32}
$X_{11}X_{21}$	0.6	0.4
$X_{11}X_{22}$	0.3	0.7
$X_{12}X_{21}$	0.0	1.0
$X_{12}X_{22}$	0.0	1.0

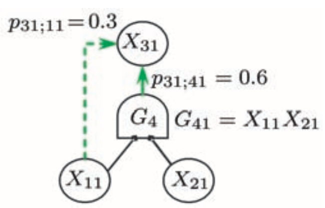


Fig.22. Conditional linkage event.

In general, the condition of the conditional linkage event $P_{nk;ij}$ or event vector $\boldsymbol{P}_{nk;i}$ is denoted as $Z_{nk;ij}$ or $\boldsymbol{Z}_{nk;i}$ respectively, and the conditional $P_{nk;ij}$ or $\boldsymbol{P}_{nk;i}$ is expressed as $P_{nk;ij}Z_{nk;ij}$ or $\boldsymbol{P}_{nk;i}\boldsymbol{Z}_{nk;i}$ respectively, where, in the single-valued cases, k indexes the valued state of the child variable. For the example of Fig.22, in terms of event expressions, we have

$$\begin{aligned}
X_{31} &= P_{31;11}Z_{31;11}X_{11} \cup P_{31;41}G_{41} \\
&= P_{31;11}X_{22}X_{11} + P_{31;41}X_{11}X_{21}, \\
\Pr\{X_{31}\} &= \Pr\{P_{31;11}X_{22}X_{11} + P_{31;41}X_{11}X_{21}\} \\
&= \Pr\{P_{31;11}\}\Pr\{X_{11}X_{22}\} + \\
&\quad \Pr\{P_{31;41}\}\Pr\{X_{11}X_{21}\} \\
&= 0.3\Pr\{X_{11}X_{22}\} + 0.6\Pr\{X_{11}X_{21}\}, \\
\Pr\{X_{31}|X_{11}X_{22}\} &= \Pr\{(P_{31;11}X_{22}X_{11} + \\
&\quad P_{31;41}X_{11}X_{21})|X_{11}X_{22}\} = \Pr\{P_{31;11}\} = 0.3, \\
\Pr\{X_{31}|X_{11}X_{21}\} &= \Pr\{(P_{31;11}X_{22}X_{11} + \\
&\quad P_{31;41}X_{11}X_{21})|X_{11}X_{21}\} = \Pr\{P_{31;41}\} = 0.6.
\end{aligned}$$

It is obvious that the conditional linkage events can be applied in many other cases, and the representation capability of DCD is significantly extended. For the burglary example in [6], if the burglary (X_{11}) and earthquake (X_{21}) share the same mechanism: vibration, in causing the alarm (X_{31}), we can use Fig.23 rather than Fig.18(b) to represent the uncertain causalities, where $Z_{31;11} = \overline{X}_{21} = X_{22}$. The situation in Fig.23 is that only when there is no earthquake, will the burglary's vibration make sense. Otherwise, the burglary cannot enhance the vibration, because the earthquake vibration exceeds the upper bound of the vibration sensor. What is still uncertain is the state of the alarm device: normal or failed. Therefore, the simultaneous occurrence of X_{11} and X_{21} has just the probability 0.9 in causing the alarm, instead of $0.9+0.8-0.9\times0.8 = 0.98$.

	X_{31}	X_{32}
$X_{11}X_{21}$	0.9	0.1
$X_{11}X_{22}$	0.8	0.2
$X_{12}X_{21}$	0.9	0.1
$X_{12}X_{22}$	0.0	1.0

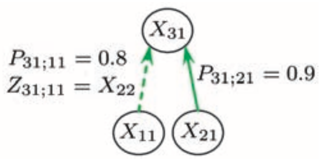


Fig.23. Burglary and earthquake example in conditional causality.

Furthermore, suppose the rat (X_{41}) can also cause the alarm with probability 0.6. That is, rat, burglary and earthquake share the same mechanism (vibration) in causing the alarm, while rat does not enhance the burglary vibration, and rat and burglary do not enhance the earthquake vibration. Then the uncertain causalities are represented in Fig.24, in which $Z_{31;41} = X_{12}X_{22}$ and $Z_{31;11} = X_{22}$.

	X_{31}	X_{32}
$X_{11}X_{21}X_{41}$	0.9	0.1
$X_{11}X_{21}X_{42}$	0.9	0.1
$X_{11}X_{22}X_{41}$	0.8	0.2
$X_{11}X_{22}X_{42}$	0.8	0.2
$X_{12}X_{21}X_{41}$	0.9	0.1
$X_{12}X_{21}X_{42}$	0.9	0.1
$X_{12}X_{22}X_{41}$	0.6	0.4
$X_{12}X_{22}X_{42}$	0.0	1.0

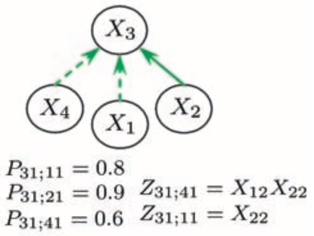


Fig.24. Rat, burglary and earthquake example.

Based on the observed evidence or the result of the event outspread, $Z_{nk;ij}$ may be met or not, i.e., $Z_{nk;ij}$ may equal to 1 (true or complete set) or 0 (false or null set). When $Z_{nk;ij}$ is not given in the evidence received, the user should be prompted to get the information, e.g., to do some experiment or physical check to determine the state of $Z_{nk;ij}$. Otherwise $(Z_{nk;ij} + \overline{Z}_{nk;ij})$ should be multiplied with the event expression and the prior probability distribution of $Z_{nk;ij}$ should be given (this case is unusual, because Z type event is defined as observable). The multiplication algorithm will be illustrated later in Subsection 4.3.

The condition $Z_{nk;ij}$ associated with the conditional linkage event $P_{nk;ij}$ can be very flexible. In fact, $Z_{nk;ij}$ can be any event observable, not only the states of the parent variables of X_n, but also the occurrence order of events, the states of other variables anywhere in the graph, and even the event not related to the state of any variable in the graph. For example, $Z_{nk;ij} = \overline{P}_{hy;gm}X_{gm}$, $Z_{nk;ij} = \{|\lambda - \beta| \leqslant \sigma\}$, $Z_{nk;ij} = \{\lambda \geqslant \beta\}$, $Z_{nk;ij} = \{\sqrt{\lambda^2 + \beta^2} \geqslant \sigma\}$ and $Z_{nk;ij} = \{E_1$ appears earlier than $E_2\}$, in which X_{gm} and $P_{hy;gm}$ can be anywhere in the graph; λ, β and σ can be any physical parameters not drawn in the graph; E_1 and E_2 can be any events included or not included in the graph. Therefore, the conditional linkage events presented in this paper can represent more complex situation than CSI and Contingent Bayesian Network (CBN)[27]. Only when $Z_{nk;ij}$ represents the events indicating the states of parent variables of X_n, will the conditional linkage event representation be similar to CSI or CBN, but the inference algorithms are different. In the case of being limited to the parent variable states, the conditional linkage event representation can be replaced by the logic gate of DCD. However, even in such a simple case, the conditional linkage event representation can be more intuitive and easier to be treated.

The conditional linkage event representation cannot be simply viewed as a compact representation of the ordinary CPT. Actually, in a CPT, once the state combination of the parent variables are given, the conditional probability distribution of the child variable is given. However, in the case of the conditional linkage event including non-parent event of the child variable or including the occurrence order of events, the CPT depends on not only the state combination of the parent variables, but also other events. Therefore, the conditional linkage event representation presented in this paper is beyond the CPT representation in BN.

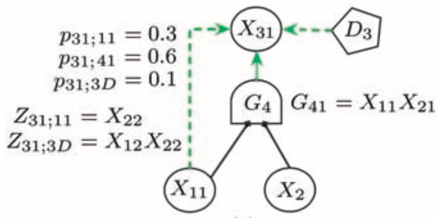
3.2.2 Default Events

It is pointed out in [7, 9-10] that in many cases, the causes of a child variable may not be modeled completely. In other words, even all parent variables are in the false state, the child variable may still have its default probability distribution different from (0, 1). This probability distribution is caused by some unknown or inexplicitly expressed causes. These causes can be represented by a leak[7,10] or dummy[9] variable. The state of the leak/dummy variable is only one: "true", i.e., it is an inevitable event with the occurrence probability always equal to 1. However, for convenience, it is still called a variable, although its state never changes.

In S-DUCG, such unknown or inexplicit cause of X_n is defined as the default variable D_n and is explicitly drawn as D_n. Similar to other parent variables, there is a linkage event between X_{nk} and D_n, i.e., $P_{nk;nD}$. The only difference between D_n and the other parent variables is that D_n has only one inevitable state, i.e., $\Pr\{D_n\} \equiv 1$.

In some cases, only when all the explicit parent variables are in the false state, will D_n functions to explain the default probability distribution of X_n. For the example shown in Fig.25(c), suppose the condition

	(a) X_{31}	(a) X_{32}	(b) X_{31}	(b) X_{32}
$X_{11}X_{21}D_3 = X_{11}X_{21}$	0.6	0.4	0.6	0.4
$X_{11}X_{22}D_3 = X_{11}X_{22}$	0.3	0.7	0.3	0.7
$X_{12}X_{21}D_3 = X_{12}X_{21}$	0.0	1.0	0.1	0.9
$X_{12}X_{22}D_3 = X_{12}X_{22}$	0.1	0.9	0.1	0.9



(a) (b) (c)

Fig.25. Default event in S-DUCG.

of $P_{31;11}$ is $Z_{31;11} = \overline{X}_{21} = X_{22}$, and the condition of $P_{31;3D}$ is $Z_{31;3D} = \overline{X}_{11}\overline{X}_{21} = X_{12}X_{22}$, we have the CPT as shown in Fig.25(a). Compared with Fig.22, $\Pr\{X_{31}|X_{12}X_{22}\} = 0.1$ rather than 0. This is because of the contribution of D_n. Of course, D_n can also be used in various other ways. For the example above, we may define $Z_{31;3D} = \overline{X}_{11} = X_{12}$, the CPT becomes Fig.25(b). For another example, D_n can be an ordinary parent variable and $P_{nk;nD}$ becomes an ordinary linkage event, where D_n is a background of other parent variables.

4 M-DUCG Model Applicable in Multi-Valued Cases

Based on the generalization of noisy-OR[7], two similar models dealing with multi-valued cases are presented in [9] and [10] respectively and can be called as noisy-MAX[17]. But this model is limited to the child variables with graded states[30]. The M-DUCG model presented in this paper does not have this limitation.

4.1 Basic Concept of M-DUCG

The M-DUCG model is based on the following assumption.

Assumption 1. Suppose V_i, $V \in \{X, B, G, D\}$, are the parent variables of X_n,

$$X_{nk} = \sum_i (r_{n;i}/r_n) \sum_{j_i} A_{nk;ij_i} V_{ij_i}. \tag{31}$$

And then

$$x_{nk} = \sum_i (r_{n;i}/r_n) \sum_{j_i} a_{nk;ij_i} v_{ij_i}, \tag{32}$$

where j_i indexes the state of parent variable V_i; $r_{n;i}$ is defined as the causal relationship intensity between X_n and V_i; $r_n \equiv \sum_i r_{n;i}$; "/" means divided by; the lowercase letters represent the probabilities of the corresponding events represented by the uppercase letters. Similar to $P_{nk;ij}$, $A_{nk;ij_i}$ is defined as the random event that V_{ij_i} does cause X_{nk} given that V_{ij_i} is true, regardless of other parent variables. To be distinguished from the linkage event in S-DUCG, $A_{nk;ij_i}$ is called the functional event from V_{ij_i} to X_{nk}.

The illustration for this assumption is shown in Fig.26, in which, for simplicity, $i \in \{1, 2, \ldots, m\}$ and $n \notin \{1, 2, \ldots, m\}$. Note that the arrow shape and color of the directed arc is $\textcolor{red}{\longrightarrow}$, instead of $\longrightarrow$, nor $\textcolor{green}{\longrightarrow}$. In DUCG, $\longrightarrow$ indicates the member of parent variables in a CPT; $\textcolor{green}{\longrightarrow}$ indicates the linkage event variable; and $\textcolor{red}{\longrightarrow}$ indicates the weighted functional event variable $\boldsymbol{F}_{n;i} \equiv (r_{n;i}/r_n)\boldsymbol{A}_{n;i}$, where $\boldsymbol{A}_{n;i}$ is an event matrix with $A_{nk;ij_i}$ as its elements in which k indexes the row and j_i indexes the column, and $\boldsymbol{F}_{n;i}$ is the brief notation of $(r_{n;i}/r_n)\boldsymbol{A}_{n;i}$ named as the weighted functional event variable that is a matrix composed of the elements: $F_{nk;ij_i} \equiv (r_n/r_{n;i})A_{nk;ij_i}$. For simplicity, j_i can be simply written as j in the case without confusion.

The interpretation for $A_{nk;ij}$ is similar to that for $P_{nk;ij}$ defined in S-DUCG applicable in single-valued cases and illustrated in Fig.12. There are two significant differences between Figs. 12 and 26: 1) in Fig.12, the relation between $X_{31;11}$ and $X_{31;21}$ is OR, while in Fig.26, the relation between $X_{nk;ij}$ and $X_{nk;ij'}$, $j \neq j'$, is XOR in effect, which means that the probabilities of $X_{nk;ij}$ can be simply summed up as shown in (32); and 2) in Fig.26, there is a weighting factor $(r_{n;i}/r_n)$ attached with $A_{nk;ij}$, while Fig.12 does not have similar weighting factors.

Similar to Fig.12, $X_{nk;ij} = (r_{n;i}/r_n)A_{nk;ij}V_{ij}$, in which $A_{nk;ij}$ represents the uncertain physical mechanism that $V_{nk;ij}$ does cause $X_{nk;ij}$ resulting in X_{nk}, given V_{ij} is true. Although the logic among $X_{nk;ij}$ in Fig.26 is XOR in effect, $A_{nk;ij}$ are not exclusive with different parent variables indexed by i, because they represent *independent* uncertain physical mechanisms and are *independent* random events. Note that $A_{nk;ij}$ is exclusive with $A_{nk';i'j'}$ given $k \neq k'$, because X_{nk} is exclusive with $X_{nk'}$. These features of the weighted events are newly defined in DUCG, which is different from the ordinary set theory and some of its special algorithms will be presented later in Section 7. This newly defined set theory may be called as the weighted set theory.

In M-DUCG, $a_{nk;ij_i} \equiv \Pr\{A_{nk;ij_i}\}$, or simply $a_{nk;ij} \equiv \Pr\{A_{nk;ij}\}$, are the original parameters given by domain engineers independently for different i. Normally, they satisfy the following constraint:

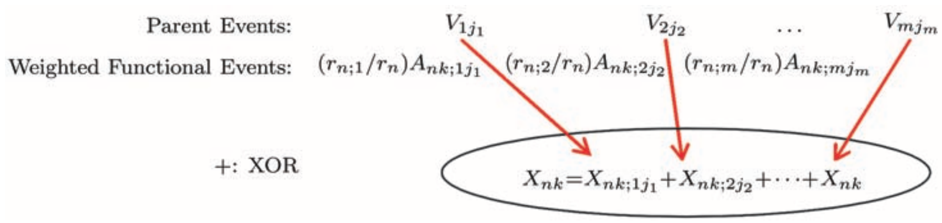


Fig.26. Illustration for the M-DUCG model.

$$\sum_k a_{nk;ij_i} = 1 \quad \text{or simply} \quad \sum_k a_{nk;ij} = 1. \tag{33}$$

This corresponds to

$$\sum_k A_{nk;ij_i} = 1 \quad \text{or simply} \quad \sum_k A_{nk;ij} = 1. \tag{34}$$

The meaning of $r_{n;i}$ is as follows. In some cases, the domain engineer is not sure whether or not there exists causal relationship between X_n and V_i. This type of uncertainty is quantified by $r_{n;i}$. That is, 1) when the domain engineer is sure that the causal relationship *exists*, $r_{n;i} = 1$; 2) when the domain engineer is sure that the causal relationship *does not exist*, $r_{n;i} = 0$; 3) the situation between 1) and 2) is represented by $1 > r_{n;i} > 0$. Since 2) cannot be reached, $r_{n;i} \neq 0$, because otherwise V_i is not a parent variable of X_n. Then we have $1 \geqslant r_{n;i} > 0$. Since $r_{n;i}$ always appears in the form of $(r_{n;i}/r_n)$, it does not matter whether or not $r_{n;i} \leqslant 1$. Sometimes, $r_{n;i} > 1$ is allowed to emphasize the importance of the causal relationship between X_n and V_i over other parent variables. Then the constraint $1 \geqslant r_{n;i} > 0$ can be loosed as $r_{n;i} > 0$. $(r_{n;i}/r_n)$ is then the normalization/weighting factor and is the weight of the probability distribution contributed from V_i to the probability distribution of X_n. With the weighting factor $(r_{n;i}/r_n)$, although $A_{nk;ij}$ can cause $X_{nk;ij}$ and then X_{nk} independently, the intensity is reduced to a degree of $(r_{n;i}/r_n)$ and the influence of $A_{nk;ij}$ to $X_{nk;ij}$ is balanced by $A_{nk;i'j'}$, $i' \neq i$, which means that every $A_{nk;ij}$ for different k and j but same n and i has the same weight $(r_{n;i}/r_n)$ in causing $X_{nk;ij}$ and then X_{nk}.

In nature, Assumption 1 is based on the following cognition:

Every parent variable 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. This cognition is actually the intuitive understanding of the domain engineers to the real world. It is also very simple. Therefore, M-DUCG can be easily applied, in particular in the case when parameters are the subjective beliefs given by domain engineers in the case without enough statistic data.

In M-DUCG, the parent variables are correlated by the weighting factors $(r_{n;i}/r_n)$, while all the parameters including $a_{nk;ij}$ and $r_{n;i}$ are independently given for individual parent variables. Therefore, M-DUCG provides a solution to the conflict between the correlation of the exclusive states of a child variable and the independence of the causal links from different parent variables, which enables the compact representation of CPTs in multi-valued cases. For the example of five states and six variables, the number of parameters in the CPT is $5^6 = 15\,625$, while the number of parameters in M-DUCG is only $5^3 + 5 = 130$.

It is seen that the causes of different states of a child variable are specified separately by (31) and (32). Therefore, (31) and (32) are applicable in multi-valued cases without limitation.

Theorem 1.
$\sum_k X_{nk} = \sum_k \sum_i (r_{n;i}/r_n) \sum_{j_i} A_{nk;ij_i}\ V_{ij_i} = 1$.

Proof. By applying $\sum_k A_{nk;ij_i} = 1$, $\sum_{j_i} V_{ij_i} = 1$ and $r_n \equiv \sum_i r_{n;i}$, we have

$$\begin{aligned}\sum_k X_{nk} &= \sum_k \sum_i (r_{n;i}/r_n) \sum_{j_i} A_{nk;ij_i} V_{ij_i} \\ &= \sum_i (r_{n;i}/r_n) \sum_{j_i} V_{ij_i} \sum_k A_{nk;ij_i} = 1. \quad \square\end{aligned}$$

Theorem 1 indicates that (31) and (32) satisfy the normalization automatically. Thus, we can use (31) and (32) to calculate the probability of the state in concern only, without considering the other states. In other words, even though the parameters needed to specify a CPT are not given completely, we can still calculate the exact probability of the state in concern, given that the causes of the state in concern are specified. This means that DUCG is able to perform the exact inference with the incomplete knowledge representation, which brings us a great convenience in knowledge base construction and probabilistic reasoning.

According to (31) and by applying the event algorithms (e.g., $V_{11}|V_{11}V_{22} = 1$ and $V_{11}|V_{12}V_{22} = 0$), we can easily get the following results:

$$\Pr\{X_{nk}| \cap_i V_{ij_i}\} = \sum_i (r_{n;i}/r_n) a_{nk;ij_i}, \tag{35}$$

$$\begin{aligned}x_{nk} \equiv \Pr\{X_{nk}\} &= \sum_i (r_{n;i}/r_n) \sum_{j_i} a_{nk;ij_i} \Pr\{V_{ij_i}\} \\ &= \sum_i (r_{n;i}/r_n) \sum_{j_i} a_{nk;ij_i} v_{ij_i}. \end{aligned} \tag{36}$$

In the same way, readers can find solutions to the cases conditional on partial parent variable states.

It should be pointed out that (35) and (36) look similar to those in [19]. The equations in [19] similar to (35) and (36) can be expressed as

$$\Pr\{X_{nk}| \cap_i V_{ij_i}\} = \sum_i (r_{n;i}/r_n) \Pr\{X_{nk}|V_{ij_i}\}, \tag{37}$$

$$\Pr\{X_{nk}\} = \sum_i (r_{n;i}/r_n) \sum_{j_i} \Pr\{X_{nk}|V_{ij_i}\} \Pr\{V_{ij_i}\}, \tag{38}$$

in which (37) is an assumption. However, (37) and (38) are different from (35) and (36) in nature, because $a_{nk;ij_i} \equiv \Pr\{A_{nk;ij_i}\} \neq \Pr\{X_{nk}|V_{ij_i}\}$.

Proof. According to (31),

$$\begin{aligned}\Pr\{X_{nk}|V_{ij_i}\} = &(r_{n;i}/r_n)\Pr\{A_{nk;ij_i}\} + \\ &\sum_{i'\neq i}(r_{n;i'}/r_n)\sum_{j_{i'}}\Pr\{A_{nk;i'j_{i'}}\}\Pr\{V_{i'j_{i'}}\} \\ &\neq \Pr\{A_{nk;ij_i}\}.\end{aligned}$$

□

The average model presented in [28] is a special case ($r_{n;i} = 1$) of M-DUCG, but its knowledge representation and inference algorithm are different. Moreover, the denominators in the weighing factors of the average model and then the weighing factors themselves are fixed, not dynamically changeable as in M-DUCG. Finally, [19] points out that the sufficiency and separability are desired for compact representations. M-DUCG does achieve them.

4.2 Logic Gate in M-DUCG

Similar to the logic gate in S-DUCG (see Subsection 3.1), the logic gate G_i in M-DUCG can also be specified with LGS_i as shown in Table 3. For the example shown in Fig.27, suppose B_1, X_2, X_4 and B_5 are binary variables and LGS_3 is as shown in Fig.28. X_4 has two direct parent variables: G_3 and B_5. Note that the real parent variables of X_4 are B_1, X_2 and B_5. G_3 is a virtual but direct parent variable of X_4.

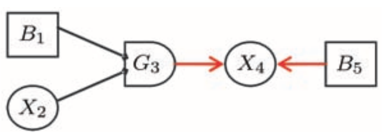


Fig.27. Illustration for the logic gate in M-DUCG.

i	G_{3j}
1	$B_{11} \cup X_{21} = B_{11} + B_{12}X_{21}$
2	$B_{12}X_{22}$

Fig.28. LGS_3 in Fig.27.

By applying (31) and LGS_3 shown in Fig.28, we have

$$\begin{aligned}X_{4k} &= (r_{4;3}/r_4)\sum_{j=1}^{2}A_{4k;3j}G_{3j} + (r_{4;5}/r_4)\sum_{j=1}^{2}A_{4k;5j}B_{5j} \\ &= (r_{4;3}/r_4)(A_{4k;31}(B_{11} \cup X_{21}) + A_{4k;32}B_{12}X_{22}) + \\ &\quad (r_{4;5}/r_4)\sum_{j=1}^{2}A_{4k;5j}B_{5j} \\ &= (r_{4;3}/r_4)(A_{4k;31}B_{11} + A_{4k;31}B_{12}X_{21} + \\ &\quad A_{4k;32}B_{12}X_{22}) + (r_{4;5}/r_4)\sum_{j=1}^{2}A_{4k;5j}B_{5j}.\end{aligned}$$

As in S-DUCG, the logic gate in M-DUCG can be as compact as possible, and the most complex logic relation can be expressed with the completely combined logic gate. For the example above, suppose G_3 is a completely combined logic gate, which is equivalent to a CPT, we have

$$\begin{aligned}X_{4k} = &(r_{4;3}/r_4)(A_{4k;31}B_{11}X_{21} + A_{4k;32}B_{12}X_{21} + \\ &A_{4k;33}B_{11}X_{22} + A_{4k;34}B_{12}X_{22}) + \\ &(r_{4;5}/r_4)\sum_{j=1}^{2}A_{4k;5j}B_{5j}.\end{aligned}$$

4.3 Conditional Functional Event in M-DUCG

An example is shown in Fig.29, in which $A_{4k;1j}$ are the conditional functional events. Suppose the condition is $\boldsymbol{Z}_{4;1} = \overline{X}_{21}$, i.e., only when X_{21} does not exist, will $A_{4k;1j}$ be possible.

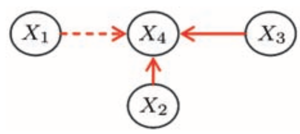


Fig.29. Illustration for the conditional functional events in M-DUCG.

In general, $\boldsymbol{Z}_{n;i}$ denotes the event matrix with elements $Z_{nk;ij}$, or denotes a single condition event associated with all $A_{nk;ij}$. Under this condition, when X_{21} is observed or given ($\boldsymbol{Z}_{4;1} = 0$), $r_{4;1} = 0$ (causal link between X_4 and X_1 does not exist); otherwise ($\boldsymbol{Z}_{4;1} = 1$), the condition is met and $A_{4k;1j}$ become ordinary functional events. Then the dashed directed arc between X_1 and X_4 becomes solid.

In this example, by applying (31), we might intuitively have

$$\begin{aligned}X_{4k} = &(r_{4;1}/r_4)Z_{4;1}\sum_{j_1}A_{4k;1j_1}X_{1j_1} + \\ &(r_{4;2}/r_4)\sum_{j_2}A_{4k;2j_2}X_{2j_2} + \\ &(r_{4;3}/r_4)\sum_{j_3}A_{4k;3j_3}X_{3j_3} \\ = &(r_{4;1}/r_4)\overline{X}_{2,1}\sum_{j_1}A_{4k;1j_1}X_{1j_1} + \\ &(r_{4;2}/r_4)\sum_{j_2}A_{4k;2j_2}X_{2j_2} + \\ &(r_{4;3}/r_4)\sum_{j_3}A_{4k;3j_3}X_{3j_3}.\end{aligned}$$

However, this expression is not precise. Moreover, it should be noted that r_4 is uncertain depending on whether or not $\boldsymbol{Z}_{4;1}$ is satisfied. Usually, based on the received evidence, the state of $\boldsymbol{Z}_{4;1}$ can be determined, because $Z_{nk;ij}$ should be observable as defined. If the state of $\boldsymbol{Z}_{4;1}$ is not observed, the prior probability of $\boldsymbol{Z}_{4;1}$ has to be given or calculated. In this example, $\Pr\{\boldsymbol{Z}_{4;1}\} = \Pr\{\overline{X}_{21}\} = 1 - \Pr\{X_{21}\}$, and $\Pr\{X_{21}\}$ can be either given or calculated from the event outspread of X_{21}. As same as in S-DUCG, when $\boldsymbol{Z}_{4;1}$ is not determined, the precise expression of X_{4k} should be multiplied with $(\boldsymbol{Z}_{4;1} + \overline{\boldsymbol{Z}}_{4;1})$:

$$\begin{aligned}X_{4k} &= (\boldsymbol{Z}_{4;1} + \overline{\boldsymbol{Z}}_{4;1})X_{4k} = (X_{21} + \overline{X}_{21})X_{4k} \\ &= (X_{21} + \overline{X}_{21})\Big((r_{4;1}/r_4)\overline{X}_{21}\sum_{j_1} A_{4k;1j_1}X_{1j_1} + \\ &\quad (r_{4;2}/r_4)\sum_{j_2} A_{4k;2j_2}X_{2j_2} + \\ &\quad (r_{4;3}/r_4)\sum_{j_3} A_{4k;3j_3}X_{3j_3}\Big) \\ &= (r_{4;2}/(r_{4;2}+r_{4;3}))A_{4k;21}X_{21} + (r_{4;3}/(r_{4;2} + \\ &\quad r_{4;3}))X_{21}\sum_{j_3} A_{4k;3j_3}X_{3j_3} + \\ &\quad (r_{4;1}/(r_{4;1}+r_{4;2}+r_{4;3}))\overline{X}_{21}\sum_{j_1} A_{4k;1j_1}X_{1j_1} + \\ &\quad (r_{4;2}/(r_{4;1}+r_{4;2}+r_{4;3}))\sum_{j_2\neq 1} A_{4k;2j_2}X_{2j_2} + \\ &\quad (r_{4;3}/(r_{4;1}+r_{4;2}+r_{4;3}))\overline{X}_{21}\sum_{j_3} A_{4k;3j_3}X_{3j_3},\end{aligned} \tag{39}$$

in which, $\overline{X}_{21} = \sum_{j_2\neq 1} X_{2j_2}$, and therefore $\sum_k X_{4k} = 1$ can be satisfied. Consequently,

$$\begin{aligned}&X_{4k}|X_{1j_1}X_{21}X_{3j_3} = (r_{4;2}/(r_{4;2}+r_{4;3}))A_{4k;21} + \\ &\quad (r_{4;3}/(r_{4;2}+r_{4;3}))A_{4k;3j_3},\ j_2 = 1, \\ &X_{4k}|X_{1j_1}X_{2j_2}X_{3j_3} \\ &\quad = (r_{4;1}/(r_{4;1}+r_{4;2}+r_{4;3}))A_{4k;1j_1} + \\ &\quad (r_{4;2}/(r_{4;1}+r_{4;2}+r_{4;3}))A_{4k;2j_2} + \\ &\quad (r_{4;3}/(r_{4;1}+r_{4;2}+r_{4;3}))A_{4k;3j_3},\ j_2 \neq 1.\end{aligned} \tag{40}$$

By replacing the uppercase letters in (39) and (40) with their corresponding lowercase letters, we can calculate $\Pr\{X_{4k}\}$ and $\Pr\{X_{4k}|X_{1j_1}X_{2j_2}X_{3j_3}\}$ easily. In the same way as illustrated above, readers can find solutions to more complex cases of the conditional functional events. Similar to the condition event in S-DUCG, the condition $\boldsymbol{Z}_{n;i}$ of $\boldsymbol{A}_{n;i}$ can be flexible, i.e., not limited to the events of the states of the parent variables of X_n, and is therefore beyond the ordinary CPT representation in BN.

4.4 Default Event in M-DUCG

The default variable D_n in M-DUCG is defined as same as in S-DUCG. However, the linkage event $P_{nk;nD}$ in S-DUCG is changed to the weighted functional event $(r_{n;D}/r_n)A_{nk;nD}$, where $A_{nk;nD}$ denotes the functional event between X_{nk} and D_n; $r_{n;D}$ denotes the causal relationship intensity between X_{nk} and D_n. Compared with $A_{nk;ij}$, $A_{nk;nD}$ has only one parent variable state, because D_n has only one state. For convenience, $A_{nk;nD}$ can also be represented by $A_{nk;ij}$, where $i = n$ and j represents D.

In multi-valued cases, although the states of a child variable are identical, there is sometimes a special state called normal state. This state is indexed by η and usually η is assigned as 0. For example, suppose X_i represents temperature. We may define $X_{i0} =$"normal", $X_{i1} =$"low", $X_{i2} =$"high", $X_{i3} =$"very low" and $X_{i4} =$"very high", where $X_{i\eta} = X_{i0}$ is the normal state and X_{ij}, $j \neq 0$, are the abnormal states.

In practice, the normal state is usually not in concern and its causes and consequences are usually not specified. In other words, $a_{i0;mg}$ and $a_{nk;i0}$ may not be given by domain engineers. This is benefited from the property of the incompleteness of DUCG (see Section 5 for details). In such a case, when all other parent variables are observed in normal states, the probability distribution of the child variable X_n will be caused by only D_n, i.e., $(r_{n;n}/r_n)\Pr\{A_{nk;nD}\}\Pr\{D_n\} = \Pr\{A_{nk;nD}\} = a_{nk;nD}$. This is because in such a case, all other $A_{nk;ij}$, $i \neq n$, do not exist and $r_n = r_{n;D}$. Usually, $A_{nk;nD}$ is a conditional functional event with the condition $Z_{nk;nD} = \{$All other parent events do not function to affect $X_n\}$. In terms of matrix, $\boldsymbol{A}_{n;n}$ is conditioned on $\boldsymbol{Z}_{n;n} = Z_{nk;nD}$, where $A_{nk;nD}$ are the elements of $\boldsymbol{A}_{n;n}$, $Z_{nk;nD}$ are the elements of $\boldsymbol{Z}_{n;n}$, and all $Z_{nk;nD}$ are equal. In the graph, $\boldsymbol{A}_{n;n}$ is drawn as the dashed directed arc from D_n to X_n. Of course, D_n can also be treated as an ordinary parent variable serving as a background of the other parent variables and the dashed directed arc becomes solid.

5 DUCG and Its Property of Incompleteness

DUCG is composed of S-DUCG or M-DUCG or the combination of them. The selection of S-DUCG or M-DUCG depends on the specific *module that is composed of a child variable and its parent variables including those linked by only logic gates*. For a specific module, when the child variable is single-valued, either S-DUCG or M-DUCG can be used; when the child variable is multi-valued, only M-DUCG can be used. The mixed use of S-DUCG and M-DUCG for different modules simply connected together in one graph is called DUCG. The common thought of S-DUCG and M-DUCG is to use the independent events $P_{nk;ji}$ or $A_{nk;ij}$ along with weighting factors $r_{n;i}$, combined with logic gates G_i, conditions $\boldsymbol{Z}_{n;i}$ and default events D_n, to compactly represent the uncertain causalities between a child variable (X type) and its real parent variables (X, B and D types). In S-DUCG, the linkage events $P_{nk;ij}$ are involved. In M-DUCG, the functional events $A_{nk;ij}$ attached with $(r_{n;i}/r_n)$ are involved. Both $P_{nk;ij}$ and $A_{nk;ij}$ represent the mechanism: a parent event does cause a child event. An illustrative example of DUCG is shown in Figs. 30 and 31.

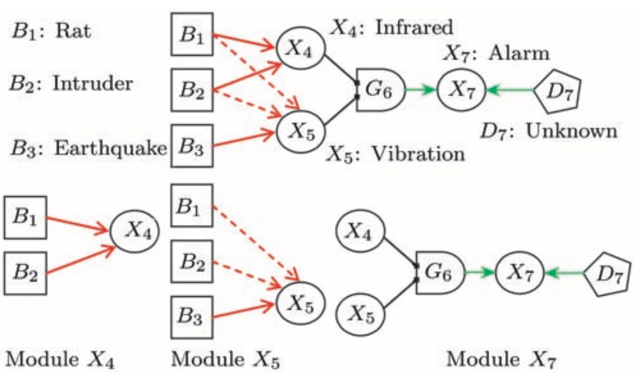


Fig.30. DUCG of an alarm system detecting intruder with its modules.

j	G_{6j}
1	$(X_{42} + X_{43})X_{52}$
2	$X_{43}X_{53}$
3	*Remnant*, i.e., $\overline{G}_{61}\overline{G}_{62} = X_{41} \cup X_{51} + X_{42}X_{53}$

Fig.31. LGS_6 of G_6 in Fig.30.

In Fig.30, $\boldsymbol{Z}_{5;1} = \boldsymbol{Z}_{5;2} = B_{32}$. X_4 and X_5 have three states each and are multi-valued. B_1, B_2 and X_7 are binary, in which X_7 is single-valued, because only the causes of X_{71} are specified. The definitions of $\{B, X, D\}$ type events are follows:

$B_{11} \equiv$ {Rat appears}; $B_{12} \equiv$ {No rat};
$B_{21} \equiv$ {Intruder appears}; $B_{22} \equiv$ {No intruder};
$B_{31} \equiv$ {Earthquake occurs};
$B_{32} \equiv$ {No earthquake};
$X_{41} \equiv$ {No infrared}; $X_{42} \equiv$ {Slight infrared};
$X_{43} \equiv$ {Strong infrared};
$X_{51} \equiv$ {No vibration}; $X_{52} \equiv$ {Slight vibration};
$X_{53} \equiv$ {Strong vibration};
$X_{71} \equiv$ {Alarm on}; $X_{72} \equiv$ {No alarm};
$D_7 \equiv$ {Unknown cause of alarm on}.

This alarm system has two sensors: the infrared sensor and vibration sensor. The signals (X_4 and X_5) of the two sensors have three states each. Some states may invoke the alarm. The alarm responses to the signals according to the logic specified in LGS_6 shown in Fig.31. The signal state combination represented by G_{63} cannot invoke the alarm, i.e., $P_{71;63} = 0$. However, even G_{63} is true, the alarm may still be invoked by some unknown cause (e.g., malfunction of the alarm), which is represented by D_7. The causes of X_{4j}, $j \in \{1,2,3\}$, are two: B_{11} and B_{21}. The causes of X_{5j}, $j \in \{1,2,3\}$, are three: B_{11}, B_{21} and B_{31}. B_{11} and B_{21} function only when there is no earthquake, i.e., $A_{5j;11}$ and $A_{5j;21}$ are conditioned on $\boldsymbol{Z}_{5;1} = \boldsymbol{Z}_{5;2} = B_{32}$. Moreover, B_{12}, B_{22} and B_{32} have no causal relation to X_4 and X_5. The weights from B_1, B_2 and B_3 to X_4 and X_5 respectively are equal, i.e., $r_{4;1} = r_{4;2} = r_{5;1} = r_{5;2} = r_{5;3} = 1$. The other parameters in concern in this example are given below:

$$\boldsymbol{a}_{4;11} = (\,0.7 \quad 0.3 \quad 0\,)^{\mathrm{T}}; \quad \boldsymbol{a}_{4;21} = (\,0 \quad 0.3 \quad 0.7\,)^{\mathrm{T}};$$
$$\boldsymbol{a}_{5;11} = (\,0.6 \quad 0.4 \quad 0\,)^{\mathrm{T}}; \quad \boldsymbol{a}_{5;21} = (\,0 \quad 0.6 \quad 0.4\,)^{\mathrm{T}};$$
$$\boldsymbol{a}_{5;31} = (\,0 \quad 0.1 \quad 0.9\,)^{\mathrm{T}};$$
$$\boldsymbol{p}_{71;6} = (\,0.9 \quad 0.7 \quad 0\,); \quad p_{71;7D} = 0.005;$$
$$\boldsymbol{b}_1 = (\,0.1 \quad 0.9\,)^{\mathrm{T}}; \quad \boldsymbol{b}_2 = (\,0.1 \quad 0.9\,)^{\mathrm{T}};$$
$$\boldsymbol{b}_3 = (\,0.01 \quad 0.99\,)^{\mathrm{T}},$$

in which, $\boldsymbol{a}_{n;i1} \equiv (a_{n1;i1}\ a_{n2;i1}\ a_{n3;i1})^{\mathrm{T}}$, $\boldsymbol{p}_{71;6} \equiv (p_{71;61}\ p_{71;62}\ p_{71;63})$ and $\boldsymbol{b}_i \equiv (b_{i1}\ b_{i2})^{\mathrm{T}}$.

Note that the parameters of $\boldsymbol{a}_{n;i2} \equiv (a_{n1;i2}\ a_{n2;i2}\ a_{n3;i2})^{\mathrm{T}}$ are not given, which means that B_{12}, B_{22} and B_{32} are not related to X_4 nor X_5. Therefore, this DUCG is incomplete. In fact, the CPTs in Fig.30 cannot be calculated from the incomplete parameters given above, unless we further give $\boldsymbol{a}_{n;i2}$, $n \in \{4,5\}$ and $i \in \{1,2,3\}$. In other words, the incomplete DUCG does not include all CPTs. People need only to give the parameters in concern, but not the parameters not in concern. This property of DUCG results in that DUCG is just the representation of the state-of-knowledge of people to the real world, but not necessarily the joint probability distribution over a set of variables, although DUCG is able to represent. Therefore, DUCG is not only a new representation of BN (the worst case is that all logic gates are completely combined) but also beyond BN.

The reason why DUCG can be incomplete is because the chaining inference of DUCG is self-relied, which is resulted from Theorem 1. The calculation of $\Pr\{X_{nk}\}$ in DUCG has nothing to do with $\Pr\{X_{nk'}\}$, nor $a_{nk';ij}$, given $k \neq k'$. When we calculate $\Pr\{X_{nk}\}$, we do not have to know $a_{nk';ij}$, $k \neq k'$. This means that some of the parameters in DUCG can be absent, without affecting the *exact* calculation in concern.

For example, suppose B_1, B_2 and X_3 are binary variables and $X_{32} = F_{32;11}B_{11} + F_{32;22}B_{22}$. When only X_{32} is in concern, the domain engineers need only to give $a_{32;11}$ and $a_{32;22}$ but not $a_{32;12}$, $a_{32;21}$, $a_{31;11}$, $a_{31;12}$, $a_{31;21}$ and $a_{31;22}$. In terms of probability expression,

$$\begin{aligned}\Pr\{X_{32}\} &= \Pr\{F_{32;11}\}\Pr\{B_{11}\} + \Pr\{F_{32;22}\}\Pr\{B_{22}\}\\ &= f_{32;11}b_{11} + f_{32;22}b_{22}\\ &= (r_{3;1}/r_3)a_{32;11}b_{11} + (r_{3;2}/r_3)a_{32;22}b_{22},\end{aligned}$$

in which, $f_{nk;ij} \equiv \Pr\{F_{nk;ij}\} = \Pr\{(r_{n;i}/r_n)A_{nk;ij}\} \equiv (r_{n;i}/r_n)\Pr\{A_{nk;ij}\} = (r_{n;i}/r_n)a_{nk;ij}$.

In effect, not in concern or not being given is equivalent to being given as 0 (not a cause). Hence, the constraint $\sum_k a_{nk;ij} = 1$ in (33) can be loosed as $\sum_k a_{nk;ij} \leqslant 1$. Similarly, the constraint $\sum_j b_{ij} = 1$ can be loosed as $\sum_j b_{ij} \leqslant 1$.

6 Simplify DUCG Conditioned on Evidence

Once evidence E is received, DUCG can be initially simplified by fixing the states of the observed variables in E. For example, suppose $E = E_1E_2 = B_{32}X_{71}$, where $E_1 = B_{32}$ and $E_2 = X_{71}$, the DUCG in Fig.30 is initially simplified as shown in Fig.32, in which B_{32} and X_{71} are fixed. This initial simplification has been presented in [26], in which the power of compiling BN with evidence is proved. In Fig.32, the observed state normal variable is filled with green color, the observed state abnormal binary variable is filled with gray color, and the state unknown (including D type) variable is not filled with any color.

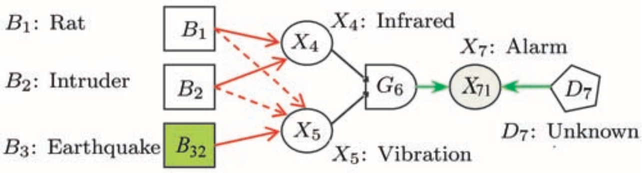


Fig.32. Initially simplified DUCG based on Fig.30, conditioned on E.

In this paper, further simplifications applicable for DUCG are presented. The basic ideas are the same as in [12, 26], i.e., to eliminate the variables contradicting with E or irrelevant to any query in concern.

Consider the example in Fig.32. As $\boldsymbol{Z}_{5;1}$ and $\boldsymbol{Z}_{5;2}$ are satisfied, the conditions of $\boldsymbol{A}_{5;1}$ and $\boldsymbol{A}_{5;2}$ are met and the arcs from B_1 and B_2 to X_5 become solid. Moreover, as $\boldsymbol{a}_{5;32}$ is not given, B_{32} cannot be a cause of X_5. Therefore, $\boldsymbol{A}_{5;3}$ is eliminated, which results in the simplified DUCG as shown in Fig.33.

In general, we can apply the following rules to further simplify the initially simplified DUCG.

Rule 1. *If E shows that $\boldsymbol{Z}_{n;i}$ is not met, $\boldsymbol{F}_{n;i}$ or $\boldsymbol{P}_{n;i}$ is eliminated from the DUCG. If E shows that $\boldsymbol{Z}_{n;i}$ is met, the conditional $\boldsymbol{F}_{n;i}$ or $\boldsymbol{P}_{n;i}$ becomes the ordinary $\boldsymbol{F}_{n;i}$ or $\boldsymbol{P}_{n;i}$.*

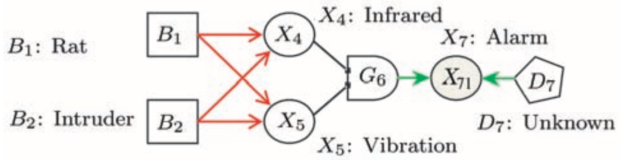


Fig.33. Simplified DUCG based on Fig.32, conditioned on E.

Rule 2. *If E shows that V_{ij}, $V \in \{B, X\}$, is true while V_{ij} is not a parent event of X_n, $\boldsymbol{F}_{n;i}$ or $\boldsymbol{P}_{n;i}$ is eliminated from the DUCG.*

For example, suppose X_{32} is not a parent event of X_5. When E shows that V_{32} is true, $\boldsymbol{F}_{5;3}$ is eliminated from the DUCG.

Rule 3. *If E shows that X_{nk} is true while X_{nk} cannot be caused by any states of V_i, $V \in \{B, X, G\}$, $\boldsymbol{F}_{n;i}$ or $\boldsymbol{P}_{n;i}$ is eliminated from the DUCG, except that V_i is included in a hypothesis, or is a descendant of an event included in a hypothesis and the causality chain between them is not blocked by any known event.*

For example, suppose X_{53} cannot be caused by any state of X_2, X_2 is not included in a hypothesis, and X_2 is not a descendant of an event included in a hypothesis, or the causality chain between X_2 and a hypothesis is blocked by known events. When E shows that X_{53} is true, $\boldsymbol{F}_{5;2}$ is eliminated. The exception in Rule 3 is because when X_{nk} is not caused by any state of V_i, while $X_{nk'}$, $k \neq k'$, is expected to be caused through a causality chain including V_i by V_{hg} included in a hypothesis, the evidence X_{nk} is a negative evidence that reduces the probability of V_{hg}, resulting in that X_{nk} is correlated to the probability of the hypothesis. If the causality chain between V_i and V_{hg} is blocked by known events, X_{nk} cannot reduce the probability of V_{hg} through V_i. Then, the causal link between X_n and V_i is irrelative to the probability of the hypothesis, so that $\boldsymbol{F}_{n;i}$ or $\boldsymbol{P}_{n;i}$ can be eliminated.

Rule 4. *If E shows that X_{nk} and V_{ij}, $V \in \{B, X\}$, are true while X_{nk} cannot be caused by V_{ij}, $\boldsymbol{F}_{n;i}$ or $\boldsymbol{P}_{n;i}$ is eliminated from the DUCG.*

For example, when E shows that X_{nk} and B_{ij} are true, and $a_{nk;ij} = 0$ or is not given, $\boldsymbol{F}_{n;i}$ is eliminated.

Rule 5. *If the state unknown X_n without input variable or G_n without input variable is encountered, X_n or G_n and its output directed arcs are eliminated from the DUCG.*

This is because such X_n or G_n is meaningless and is out of concern. By definitions, they have to have at least one input, otherwise they are meaningless. As X_n or G_n is eliminated, its output directed arcs are also meaningless and should be eliminated. It should be noted that D_n is an input of X_n. When D_n exists, X_n should not be eliminated.

Rule 6. *If G_i without any output is encountered for any reason, G_i is eliminated from the DUCG.*

For example, suppose G_i has two child variables X_1 and X_2, if E shows X_{11} and X_{21}, while X_{11} and X_{21} cannot be caused by any state of G_i, $\boldsymbol{F}_{1;i}$ and $\boldsymbol{F}_{2;i}$ are eliminated according to Rule 3, resulting in that G_i has no output, G_i is then eliminated. This is because a logic gate without any output is meaningless.

Rule 7. *If* 1) *the state of X_n is unknown,* 2) *X_n does not have any output, and* 3) *X_n is not predetermined in concern, X_n and all its input directed arcs are eliminated from the DUCG.*

For example, suppose the state of X_3 is unknown, it has no output and it is not predetermined in concern, then X_3 and $\boldsymbol{F}_{3;i}$ are eliminated. This is because given E, X_3 and $\boldsymbol{F}_{3;i}$ do not have any influence in finding the possible hypotheses and updating the probabilities of these hypotheses.

Rule 8. *If E shows that X_{nk} and V_{ij}, $V \in \{B, X\}$, are true and X_{nk} appears earlier than V_{ij}, which means that V_{ij} cannot be the cause of X_{nk}, the F or P type variables* (*they are the members of the causality chain from V_{ij} to X_{nk} and are not related to any other upstream causality chain of X_{nk}*) *are eliminated from the DUCG.*

For example, suppose X_{43} appears earlier than its ground parent event B_{22}, and between them is the parent variable X_1, $\boldsymbol{F}_{4;1}$ and $\boldsymbol{F}_{1;2}$ can all be eliminated, provided that no other ancestor of X_{43} has causality connection with X_{43} through X_1. However, if B_3 is also a parent variable of X_1, which means that B_3 can cause X_{43} through $\boldsymbol{F}_{43;1}$, only $\boldsymbol{F}_{1;2}$ can be eliminated while $\boldsymbol{F}_{4;1}$ cannot be eliminated, because $\boldsymbol{F}_{43;1}\boldsymbol{F}_{1;3}B_3$ is a possible causality chain.

It should be noted that this rule is about a specific type of evidence: the occurrence order of events. This type of evidence has been presented in [31]. Rule 8 is only an extension of the result in [31] from DCD to DUCG.

Rule 9. *If there is such a group of variables* (*named as the independent group*) *that have no causal connection with those variables related to E, and no variable in this group is predetermined in concern, this independent group of variables can be eliminated from the DUCG.*

For example, suppose B_1 and X_2 along with $\boldsymbol{F}_{2;1}$ become independent of (without any causal connection with) the other variables related to E, meanwhile B_1 and X_2 along with $\boldsymbol{F}_{2;1}$ are not predetermined in concern, B_1, X_2 and $\boldsymbol{F}_{2;1}$ are eliminated, because they have nothing to do with the inference for the hypotheses in concern, given E.

Rule 10. *If E shows X_{nk} is true while X_{nk} does not have any input due to any reason, add a virtual parent event D_n to X_{nk} with $a_{nk;nD} = 1$ and $a_{nk';nD} = 0$, $k \neq k'$. $r_{n;D}$ can be any value. The added virtual D_n can be drawn as D_n in the simplified graph.*

For example, if E shows that X_{51} is true and all its input directed arcs are broken due to the simplification, X_{51} is then without any input and should be eliminated according to Rule 5. However, suppose X_{51} is the parent of X_{61}. Then, X_{51} should not be eliminated. The problem is that X_{51} should have an input in the given DUCG, otherwise it cannot be observed. But by some mistake or other reason, this input is not given in the DUCG. Therefore, there must be an unknown cause ($a_{51;5D} = 1$) for X_{51}. This unknown cause is represented by the virtual event D_5, so that X_{51} is not eliminated. This is another property of DUCG, i.e., DUCG is able to point out the absence of meaningful events in the DUCG graph. Of course, the detailed contents of such events can only be explained by the domain experts after they are informed the existence of such events.

Rules 1$\sim$10 can be applied in any order, at any time and repeatedly, except that Rule 10 has the priority over Rule 5.

It should be pointed out that the simplification by Rules 1$\sim$10 is different from the variable elimination (VE) presented in [12], because VE is based on a given query $\Pr\{X_{nk}|E\}$, while Rules 1$\sim$10 are based on only E. The simplified DUCG can be applied for any query remaining in concern after the simplification. Of course, based on the simplified DUCG, for a given query, VE can be applied to further simplify the DUCG as a query-specific DUCG.

It is noted that the $\{X, B\}$ type variables in the original DUCG are divided, by applying the above rules, into two groups indexed by the index sets S_{in} and S_{out} respectively. The variables indexed by S_{in} are those included in the simplified DUCG. The variables indexed by S_{out} are those eliminated by Rules 1$\sim$10, and are no longer in concern given E. Therefore, the $\{B, X\}$ type variables in concern have been reduced from $S_{in} + S_{out}$ to S_{in} conditioned on E. For the example shown in Figs. 30 and 33, $S_{in} = \{1, 2, 4, 5, 7\}$ and $S_{out} = \{3\}$. Usually, $S_{in} \ll S_{in} + S_{out}$. Therefore, Rules 1$\sim$10 can dynamically reduce the scale of problem greatly.

In what follows, we will focus on the variables indexed by S_{in} only. In other words, the following discussion is based on only the simplified DUCG, and only those hypotheses included in the simplified DUCG will be considered. For the application of diagnoses, this means that the root causes to be found are reduced to only the state unknown B type variables indexed in S_{in}. Sometimes, there is only one state unknown B type variable in S_{in}. Then the diagnosis is finished, because the root cause has been exactly found. Note that such a qualitative solution can be found before any numerical calculation. Therefore, the probability parameter accuracy in DUCG is not as important as in BN. This is another benefit of DUCG.

7 Probabilistic Reasoning Based on the Simplified DUCG

Suppose H_{kj} is the hypothesis in concern conditioned on evidence $E = \prod_h E_h = \prod_h V_{hy_h}$, where H_{kj} is composed of $\{X, B, P, A\}$ type events (P and A type events can be included in a hypothesis or query is another benefit of DUCG), k indexes the variables in H_{kj} (e.g., $H_k = B_1X_2$) and j indexes the state combination of these variables (e.g., $H_{kj} = B_{11}X_{23}$); $E_h = V_{hy_h}$ and $V \in \{X, B\}$. The probability updating can be given as

$$h^s_{kj} \equiv \Pr\{H_{kj}|E\} = \frac{\Pr\{H_{kj}E\}}{\Pr\{E\}} = \frac{\Pr\{H_{kj}\prod_h V_{hy_h}\}}{\Pr\{\prod_h V_{hy_h}\}}, \tag{41}$$

where h^s_{kj} *is called the state probability* of H_{kj}. Based on the simplified DUCG, there are two alternative approaches to calculate h^s_{kj}.

The first is to calculate the CPTs still included in the simplified DUCG, provided that the simplified DUCG is not incomplete. By applying the existing algorithms of BN, the inference can be done. However, for an incomplete DUCG, not all CPTs can be calculated. Then, only the second approach presented in this paper is applicable.

The second is the event outspread approach originally presented in [31] for DCD and is extended for DUCG in this paper. It is seen that (41) needs to outspread $\prod_h V_{hy_h}$ and $H_{kj}\prod_h V_{hy_h}$ respectively into the form of the sum-of-products composed of only the $\{B, A, P, D\}$ type events, so that the probability can be calculated by simply replacing these events with their probabilities. During the event outspread, the following rules are applied.

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

Proof. V_{ij} is an event. Therefore, $(V_{ij})^y = V_{ij}$ is obvious. By definition, the different states of a variable are exclusive. Therefore, $V_{ij}V_{ij'} = 0$. □

There are many ways to apply Rule 11. For example, suppose $E_1 = X_{nk} = F_{nk;ij}X_{ij}$ and $E_2 = X_{ij}$, we have $E_1E_2 = F_{nk;ij}X_{ij}X_{ij} = F_{nk;ij}X_{ij} = E_1$. For another example, $F_{nk;ij}X_{ij}X_{ij'} = 0$.

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

Proof. By definition, $(F_{nk;ij})^y = ((r_{n;i}/r_n)A_{nk;ij})^y$. Since $A_{nk;ij}$ is an event, $(A_{nk;ij})^y = A_{nk;ij}$. Therefore, $(F_{nk;ij})^y = ((r_{n;i}/r_n)A_{nk;ij})^y = (r_{n;i}/r_n)^y A_{nk;ij}$. According to Rule 11, when $k \neq k'$, $X_{nk}X_{nk'} = 0$, which means that $A_{nk;ij}$ cannot appear simultaneously with $A_{nk';ij}$. Thus $A_{nk;ij}A_{nk';ij} = 0$; Similarly, when $j \neq j'$, $V_{ij}V_{ij'} = 0$, $V \in \{X, B, G\}$, which means that $A_{nk;ij}$ cannot appear simultaneously with $A_{nk;ij'}$. Thus $A_{nk;ij}A_{nk;ij'} = 0$. Therefore, we have $F_{nk;ij}F_{nk';ij} = (r_{n;i}/r_n)^2 A_{nk;ij}A_{nk';ij} = 0$, $F_{nk;ij}F_{nk;ij'} = (r_{n;i}/r_n)^2 A_{nk;ij}A_{nk;ij'} = 0$ and $F_{nk;ij}F_{nk';ij'} = (r_{n;i}/r_n)^2 A_{nk;ij}A_{nk';ij'} = 0$. □

For example, suppose $E_1 = F_{nk;ij}X_{ij} + F_{nk;gy}B_{gy}$, $E_2 = F_{nk;gy}B_{gy}$ and $E_3 = F_{nk';ij}X_{ij}$, where $k \neq k'$. We have

$$\begin{aligned} E_1E_2 &= F_{nk;ij}X_{ij}F_{nk;gy}B_{gy} + (F_{nk;gy}B_{gy})^2 \\ &= (r_{n;i}r_{n;g}/r_n^2)A_{nk;ij}X_{ij}A_{nk;gy}B_{gy} + \\ &\quad (r_{n;g}/r_n)^2 A_{nk;gy}B_{gy}, \\ E_1E_3 &= F_{nk;ij}X_{ij}F_{nk';ij}X_{ij} + F_{nk;gy}B_{gy}F_{nk';ij}X_{ij} = 0. \end{aligned}$$

Rule 13. *Let S_m denote the variable index set m, $m \in \{1, 2, \ldots, M\}$, and $S_1 \subseteq S_2$, $S_1 \subseteq S_3, \ldots, S_1 \subseteq S_M$. Then*

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

Proof. Suppose E_1 and E_2 are two events. From the set theory, $E_1 \cup E_1E_2 = E_1 = E_1 \cup E_1$, i.e., once E_1E_2 is true, E_1 is true, and once E_1 is true, the whole equation is true, which is equivalent to E_1E_2 is true. Thus, we can use E_1 to replace E_1E_2 in this equation. Similarly,

$$\begin{aligned} &\sum_{m=1}^{M}\prod_{i\in S_m} F_{nk;ij_i}V_{ij_i} \\ &\quad= \sum_{m=1}^{M}\prod_{i\in S_m}(r_{n;i}/r_n)A_{nk;ij_i}V_{ij_i} \\ &\quad= \sum_{m=1}^{M}\Big(\prod_{i\in S_m}(r_{n;i}/r_n)\prod_{i\in S_m}A_{nk;ij_i}V_{ij_i}\Big) \\ &\quad= \sum_{m=1}^{M}\Big(\prod_{i\in S_m}(r_{n;i}/r_n)\prod_{i\in S_1}A_{nk;ij_i}V_{ij_i}\Big) \\ &\quad= \Big(\sum_{m=1}^{M}\prod_{i\in S_m}(r_{n;i}/r_n)\Big)\prod_{i\in S_1}A_{nk;ij_i}V_{ij_i}. \end{aligned}$$

The third "=" is because once $\prod_{i\in S_m} A_{nk;ij_i}V_{ij_i}$ is true, $\prod_{i\in S_1} A_{nk;ij_i}V_{ij_i}$ is true, and once $\prod_{i\in S_1} A_{nk;ij_i}V_{ij_i}$ is true, the whole equation is true, which is equivalent to $\prod_{i\in S_m} A_{nk;ij_i}V_{ij_i}$ is true. Thus, we can use $\prod_{i\in S_1} A_{nk;ij_i}V_{ij_i}$ to replace $\prod_{i\in S_m} A_{nk;ij_i}V_{ij_i}$, conditioned on $S_1 \subseteq S_2$, $S_1 \subseteq S_3, \ldots, S_1 \subseteq S_M$. □

For example,

$$\begin{aligned}
&F_{31;11}V_{11} + F_{31;11}V_{11}F_{31;22}V_{22}\\
=&(r_{3;1}/r_3)A_{31;11}V_{11}+\\
&(r_{3;1}/r_3)A_{31;11}V_{11}(r_{3;2}/r_3)A_{31;22}V_{22}\\
=&(r_{3;1}/r_3 + (r_{3;1}/r_3)(r_{3;2}/r_3))A_{31;11}V_{11},
\end{aligned}$$

where $S_1 = \{1\}$, $S_2 = \{1, 2\}$ and $M = 2$.

It should be noted that Rule 13 actually defines a new algorithm different from the ordinary set theory. This is because in M-DUCG, the A type events are always attached with the weighting factors $(r_{n;i}/r_n)$. Rule 13 says that the event absorption of set theory is applicable, but the weighting factors cannot disappear due to the event absorption. For the example above, suppose $E_1 = (r_{3;1}/r_3)A_{31;11}V_{11}$ and $E_2 = (r_{3;2}/r_3)A_{32;22}V_{22}$. According to the set theory, E_1E_2 should be absorbed by E_1; but the weighting factors $(r_{3;1}/r_3)(r_{3;2}/r_3)$ attached with E_1E_2 should not be absorbed but be added to $(r_{3;1}/r_3)$ that is attached with E_1. In other words, the event operation and the weighting factor operation should both be done simultaneously. That is why we need to write $E_1 \cup E_1E_2 = E_1 \cup E_1$.

Rule 14. *Let* $j = j_i$, $F_{nk;ij}V_{ij}(\sum_{i'} F_{nk;i'j_{i'}}V_{i'j_{i'}}) = F_{nk;ij}V_{ij}$.

Proof. i is one of i'. With the same concept of Rule 13, we have

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

□

Rule 14 may be viewed as if $F_{nk;ij}$ from different parent variables were exclusive with each other, i.e., given $i \neq i'$, $F_{nk;ij}F_{nk;i'j'} = 0$ while $(F_{nk;ij})^2 = F_{nk;ij}$. However, *this is incorrect*, because 1) $A_{nk;ij}$ and $A_{nk;i'j'}$ are actually independent of each other (being independently given) and 2) $(F_{nk;ij})^2 = (r_{n;i}/r_n)F_{nk;ij}$ (Rule 12) instead of $(F_{nk;ij})^2 = F_{nk;ij}$.

As an application, for the example shown in Fig.32, after simplifying the DUCG as shown in Fig.33, the hypotheses in concern become $H_{11} \equiv B_{11}$, $H_{21} \equiv B_{21}$ and $H_{71} \equiv P_{71;7D}$. In other words, *denote* S_H as the possible hypothesis space conditioned on E, we have $S_H = \{H_{11}, H_{21}, H_{71}\}$ that is the qualitative solution to this diagnostic problem. Since the influence of evidence $E_1 = B_{32}$ has been included in Fig.33 and B_{32} is irrelevant to the simplified DUCG, we know that $\Pr\{B_{21}|B_{32}X_{71}\}$ is equivalent to $\Pr\{B_{21}|X_{71}\}$. According to (41), we have

$$\begin{aligned}
h^s_{21} \equiv&\Pr\{B_{21}|B_{32}X_{71}\}\\
=&\Pr\{B_{21}|X_{71}\} = \frac{\Pr\{B_{21}X_{71}\}}{\Pr\{X_{71}\}}.
\end{aligned}$$

By outspreading X_{71} and $B_{21}X_{71}$ respectively and noting $D_7 = 1$, we have

$$\begin{aligned}
X_{71} =& (P_{71;61}G_{61} + P_{71;62}G_{62}) \cup P_{71;7D}D_7 = (P_{71;61}G_{61} + P_{71;62}G_{62})\overline{P}_{71;7D} + P_{71;7D} = (P_{71;61}(X_{42} + X_{43})X_{52}+\\
&P_{71;62}X_{43}X_{53})\overline{P}_{71;7D} + P_{71;7D} = P_{71;61}\overline{P}_{71;7}X_{42}X_{52} + P_{71;61}\overline{P}_{71;7D}X_{43}X_{52} + P_{71;62}\overline{P}_{71;7D}X_{43}X_{53} + P_{71;7D}\\
=& P_{71;61}\overline{P}_{71;7D}(F_{42;11}B_{11} + F_{42;21}B_{21})(F_{52;11}B_{11} + F_{52;21}B_{21}) + P_{71;61}\overline{P}_{71;7D}(F_{43;11}B_{11} + F_{43;21}B_{21})\cdot\\
&(F_{52;11}B_{11} + F_{52;21}B_{21}) + P_{71;62}\overline{P}_{71;7D}(F_{43;11}B_{11} + F_{43;21}B_{21})(F_{53;11}B_{11} + F_{53;21}B_{21}) + P_{71;7D}\\
=& \overline{P}_{71;7D}\begin{pmatrix} P_{71;61}(F_{42;11}F_{52;11}B_{11} + (F_{42;11}F_{52;21} + F_{52;11}F_{42;21})B_{11}B_{21} + F_{42;21}F_{52;21}B_{21}+\\ F_{43;11}F_{52;11}B_{11} + (F_{43;11}F_{52;21} + F_{52;11}F_{43;21})B_{11}B_{21} + F_{43;21}F_{52;21}B_{21})+\\ P_{71;62}(F_{43;11}F_{53;11}B_{11} + (F_{43;11}F_{53;21} + F_{53;11}F_{43;21})B_{11}B_{21} + F_{43;21}F_{53;21}B_{21}) \end{pmatrix} + P_{71;7D},
\end{aligned} \tag{42}$$

$$\begin{aligned}
B_{21}X_{71} =& B_{21}\left(\overline{P}_{71;7D}\begin{pmatrix} P_{71;61}(F_{42;11}F_{52;11}B_{11} + (F_{42;11}F_{52;21} + F_{52;11}F_{42;21})B_{11}B_{21} + F_{42;21}F_{52;21}B_{21}+\\ F_{43;11}F_{52;11}B_{11} + (F_{43;11}F_{52;21} + F_{52;11}F_{43;21})B_{11}B_{21} + F_{43;21}F_{52;21}B_{21})+\\ P_{71;62}(F_{43;11}F_{53;11}B_{11} + (F_{43;11}F_{53;21} + F_{53;11}F_{43;21})B_{11}B_{21} + F_{43;21}F_{53;21}B_{21}) \end{pmatrix} + P_{71;7D}\right)\\
=& B_{21}\left(\overline{P}_{71;7D}\begin{pmatrix} P_{71;61}(F_{42;11}F_{52;11}B_{11} + (F_{42;11}F_{52;21} + F_{52;11}F_{42;21})B_{11} + F_{42;21}F_{52;21}+\\ F_{43;11}F_{52;11}B_{11} + (F_{43;11}F_{52;21} + F_{52;11}F_{43;21})B_{11} + F_{43;21}F_{52;21})+\\ P_{71;62}(F_{43;11}F_{53;11}B_{11} + (F_{43;11}F_{53;21} + F_{53;11}F_{43;21})B_{11} + F_{43;21}F_{53;21}) \end{pmatrix} + P_{71;7D}\right).
\end{aligned} \tag{43}$$

The $\{b, p, a, r\}$ type parameters have been given in Section 5. Remember $\Pr\{F_{nk;ij}\} = f_{nk;ij} = (r_{n;i}/r_n)\ a_{nk;ij}$. By replacing the uppercase letters with their corresponding lowercase letters in the above event expressions, we can easily calculate out $\Pr\{X_{71}\} = 0.027\,29$ and $\Pr\{B_{21}X_{71}\} = 0.020\,38$. Finally, $h^s_{21} = 0.746\,5$ is calculated from (41). Similarly, we can calculate out $h^s_{11} = 0.231\,4$ and $h^s_{71} = \Pr\{H_{71}|E\} = \Pr\{P_{71;7D}|X_{71}\} = 0.183\,2$. Based on that B_{11}, B_{21} and $P_{71;7D}$ are the only hypotheses given E, the conditional rank probabilities of B_{11}, B_{21} and $P_{71;7D}$ are calculated as $h^r_{11} = 0.199\,3$, $h^r_{21} = 0.642\,9$ and $h^r_{71} = 0.157\,8$ respectively, where *the conditional rank probability is defined as*

$$h^r_{kj} \equiv \frac{h^s_{kj}}{\sum_{H_{kj}\in S_H} h^s_{kj}} = \frac{\Pr\{H_{kj}E\}}{\sum_{H_{kj}\in S_H} \Pr\{H_{kj}E\}}. \tag{44}$$

It satisfies

$$\sum_{H_{kj}\in S_H} h^r_{kj} = 1. \tag{45}$$

Note that if there is only $h^r_{kj} = 1$ one hypothesis H_{kj} in S_H, according to (44), we know $h^r_{kj} = 1$ without calculating h^s_{kj}. In the diagnostic inference, when only one hypothesis H_{kj} is found possible after simplifying DUCG based on E, even though $H_{kj}E \neq E$, the diagnostic inference is finished without numerical calculation, because we know $h^r_{kj} = 1$ for sure. Meanwhile, the parameter accuracy is less important in DUCG, because 1) the qualitative solution S_H has been found before numerical calculation, and 2) the numerical calculation is limited to the possible hypotheses in S_H such that the data accuracy has less impact on the calculation.

It is noted that Fig.33 is a multiply connected graph. In this example, it is shown that the inference of DUCG in the case of the multiply connected graph does not rely on the application of the clustering or cutset conditioning algorithm as in BN. The correlations of the multiple connections in DUCG are automatically broken through the event outspread without any special computation, no matter whether the DUCG is singly or multiply connected. This is another benefit of applying DUCG.

8 Conclusions and Future Work

In this paper, it is pointed out that the compact uncertain causality representations applicable in single-valued cases may not be suitable to be applied in multi-valued cases, because the imposed normalization is improper. As a solution, DUCG is presented, which is applicable in both single-valued case (S-DUCG) and multi-valued case (M-DUCG), while the M-DUCG model can also be applied in the single-valued case. The sufficiency and separability mentioned in [19] for the compact knowledge representation and efficient inference algorithm are actually achieved by DUCG. Moreover, based on the simplified DUCG conditioned on the observed evidence including the occurrence order of events, DUCG provides a new tool, i.e., the event outspread algorithm to deal with the probabilistic reasoning, regardless of whether the simplified DUCG is singly or multiply connected. A set of rules of simplifying DUCG and the event outspread are presented. Sometimes, the simplified DUCG can provide the qualitative or even final solution to the problem without any numerical calculation. Moreover, benefited from Theorem 1 that enables the self-relied chaining inference algorithm, DUCG can be incomplete in representing CPTs, i.e., it is not necessary for DUCG to represent the causal knowledge not in concern, although the complete representation is necessary for representing CPTs. Mathematically, DUCG may not represent a joint probability distribution over a set of variables, although DUCG is able to. This, along with the capacities of representing complex conditional uncertain causalities and utilizing the occurrence order of events, makes DUCG a new framework including and beyond BN. Finally, a new event algorithm beyond the ordinary set theory is presented to deal with the logic operation of the weighted events newly defined in M-DUCG.

Limited to the length, only the discrete, certain evidence and directed acyclic graph (DAG) are addressed in this paper. It will be shown that DUCG is able to deal with directed cyclic graph (DCG) in a future paper; otherwise, the presented modularized construction of DUCG is inapplicable. The more efficient inference algorithm in terms of matrixes is also to be addressed in a future paper. Moreover, DUCG also aims at dealing with the dynamic change of online received evidence, dynamically changed causality functions, cases involving the initiating and non-initiating events in process systems, overlap of causality functions from continuous past time with varying weight, freely mixed certain and uncertain causalities in a same DUCG, uncertain/fuzzy evidence, freely mixed continuous and discrete variables in compact representations, etc. Many of these methodologies are being applied in a project for the online fault forecast, diagnosis and prediction of the nuclear power plants of China Guangdong Nuclear Power Group. All these issues are planned to be addressed in the future papers.

References

[1] Shortliffe E H, Buchanan B G. A model of inexact reason in medicine. *Mathematical Bioscience*, 1975, 23(3/4): 351-379.

[2] Shafer G. A Mathematical Theory of Evidence. Princeton, NJ: Princeton University Press, 1976.

[3] Duda R O *et al.* Development of the Prospector consultation system for mineral exploration. Final report, SRI Project 5821 and 6415, SRI International, 1978.

[4] Zadeh L A. The role of fuzzy logic in the management of uncertainty in expert systems. *Fuzzy Sets and Systems*, 1983, 11: 199-227.

[5] Pearl J. Fusion, propagation, and structuring in belief networks. *Artificial Intelligence*, 1986, 29(3): 241-288.

[6] Pearl J. Probabilistic Reasoning in Intelligent Systems. San Mateo: Morgan Kaufmann, 1988. ISBN 0-934613-73-7.

[7] Henrion M. Practical issues in constructing a Bayes' belief network. In *Proc. the 3rd Conf. Uncertainty in Artificial Intelligence*, July 1987, pp.132-139.

[8] Srinivas S. A generalization of the noisy-OR model. In *Proc. the 9th Conf. Uncertainty in Artificial Intelligence*, San Francisco, July 1993, pp.208-215.

[9] Diez F J. Parameter adjustment in Bayes networks: The generalized noisy-OR gate. In *Proc. the 9th Conf. Uncertainty in Artificial Intelligence*, 1993, pp.99-105.

[10] Pradhan M, Provan G, Middleton B, Henrion M. Knowledge engineering for large belief networks. In *Proc. the 10th*

Conference on Uncertainty in Artificial Intelligence, July 1994, pp.484-490.

[11] Gilio A, Scozzafava R. Conditional events in probability assessment and revision. *IEEE Trans. Systems, Man and Cybernetics*, 1994, 24(12): 1741-1746.

[12] Zhang N L, Poole D. A simple approach to Bayesian Network computation. In *Proc. the 10th Biennial Canadian Artificial Intelligence Conference*, 1994, pp.171-178.

[13] D'Ambrosio B. Local expression languages for probabilistic dependence. *Int. J. Approximate Reasoning*, 1995, 13(1): 61-68.

[14] Boutilier C, Friedman N, Goldszmidt M, Koller D. Context-specific independence in Bayesian network. In *Proc. the 12th Conf. Uncertainty in Artificial Intelligence*, Aug. 1996, pp.115-123.

[15] Heckerman D, Breese J S. Causal independence for probability assessment and inference using Bayesian networks. *IEEE Trans. Systems, Man and Cybernetics*, 1996, 26(6): 826-831.

[16] Cowell R G, Dawid A P, Lauritzen S L, Spiegelhalter D J. Probabilistic Networks and Expert Systems. New York: Springer, 1999.

[17] Takikawa M, D'Ambrosio B. Multiplicative factorization of noisy-MAX. In *Proc. the 15th Conf. Uncertainty in Artificial Intelligence*, July 30-Aug. 1, 1999, pp.622-630.

[18] Jensen F V. Bayesian graphical models. In *Encyclopedia of Environmetrics*, Sussex: Wiley, 2000.

[19] Pfeffer A. Sufficiency, separability and temporal probabilistic models. In *Proc. the 17th Conf. Uncertainty in Artificial Intelligence*, Aug. 2001, pp.421-428.

[20] Russell S J, Norvig P. Artificial Intelligence: A Modern Approach, 2nd edition. Prentice Hall, 2003.

[21] Pearl J, Russell S. Bayesian networks. In *The Handbook of Brain Theory and Neural Networks*, 2nd edition, Arbib M A (ed.), MIT Press, 2003.

[22] Poole D, Zhang N L. Exploiting contextual independence in probabilistic inference. *Journal of Artificial Intelligence Research*, 2003, 18(1): 263-313.

[23] Zagorecki A, Druzdzel M. An empirical study of probability elicitation under noisy-OR assumption. In *Proc. the 17th FLAIRS*, May 2004, pp.880-885.

[24] Schubert L K. A new characterization of probabilities in Bayesian networks. In *Proc. the 20th Conf. Uncertainty in Artificial Intelligence*, Banff, Canada, July 7-11, 2004, pp.495-503.

[25] Lucas P J F. Bayesian network modeling through qualitative patterns. *Artificial Intelligence*, 2005, 163(2): 233-263.

[26] Chavira M, Allen D, Darwiche A. Exploiting evidence in probabilistic inference. In *Proc. the 21st Conf. Uncertainty in Artificial Intelligence*, July 2005, pp.112-119.

[27] Milch B, Marthi B, Sontag D, Russell S, Ong D L, Kolobov. Approximate inference for infinite contingent Bayesian networks. In *Proc. the 10th International Workshop on Artificial Intelligence and Statistics*, Barbados, Jan. 2005.

[28] Zagorecki A, Voortman M, Druzdzel M J. Decomposing local probability distributions in Bayesian networks for improved inference and parameter learning. In *Proc. the 19th FLAIRS Conference*, Melbourne Beach, USA, May 2006, pp.860-865.

[29] van Gerven M A J, Lucas P J F, van der Weide Th P. A generic qualitative characterization of independence of causal influence. *International Journal of Approximate Reasoning*, 2008, 48(1): 214-236.

[30] Li W, Poupart P, van Beek P. Exploiting causal independence using weighted model counting. In *Proc. the 23rd AAAI Conference on Artificial Intelligence*, Chicago, USA, July 13-17, 2008, pp.337-343.

[31] Zhang Q. Probabilistic reasoning based on dynamic causality trees/diagrams. *Reliability Engineering and System Safety*, 1994, 46(3): 209-220.

Qin Zhang is executive secretary of China Association for Science and Technology, professor of Institute of Nuclear and New Energy Technology, Tsinghua University, and School of Computer Science and Technology, Beihang University, Beijing, and visiting scholar of Univeristy of California, Los Angeles and University of Tennessee, USA. He received the B.S. M.S. and Ph.D. degrees in nuclear engineering from Tsinghua University. He was a post-doctor of School of Economics and Management, Tsinghua University. His research interests include reliability analysis, expert system and intellectual property right, etc. He has published about 60 papers in IEEE Trans. Reliability, Reliability Engineering and System Safety, and other natural scientific journals.