# Reliability Analysis of Cold-standby Systems with Subsystems Using Conditional Binary Decision Diagrams

*Siwei Zhou, Yinghuai Yu, Xiaohong Peng*[*]

*College of Mathematics and Computer, Guangdong Ocean University, China*
*siweizhou@gdou.edu.cn, yuyinghuai@126.com, lgdpxh@126.com*

## Abstract

Cold-standby systems have been widely used for conditions with limited power, which achieve fault tolerance and high-reliability systems. The cold spare (CSP) gate is a common dynamic gate in the dynamic fault tree (DFT). DFT with CSP gates is typically used to model a cold-standby system for reliability analysis. In general, inputs of the CSP gate are considered to be basic events. However, with the requirement of the current system design, the inputs of the CSP gate may be either basic events or top events of subtrees. Hence, the sequence-dependency among basic events in CSP gates becomes much more complex. However, the early conditional binary decision diagram (CBDD) used for the reliability analysis of spare gates does not consider it well. To address this problem, the conditioning event *rep* is improved to describe the replacement behavior in CSP gates with subtrees inputs, and the related formulae are derived. Further, a combinatorial method based on the CBDD is demonstrated to evaluate the reliability of cold-standby systems modeled by CSP gates with subtrees inputs. The case study is presented to show the advantage of using our method.

**Keywords:** Dynamic fault tree, Conditional binary decision diagram, Conditioning event, Reliability, Dependable computing

## 1 Introduction

Assumptions:
1. The system is not repairable.
2. Switching between the primary and spare components is perfect.

Acronyms, abbreviations, and notations are shown in Table 1 and Table 2.

**Table 1.** Acronyms and abbreviations

| | |
|---|---|
| SFT | Static fault tree |
| DFT | Dynamic fault tree |
| CFT | Conditional fault tree |
| SP | Spare |
| CSP | Cold spare |
| HSP | Hot spare |
| WSP | Warm spare |
| MCS | Minimal cut set |
| MCQ | Minimal cut sequence |
| SDP | Sum of disjoint product |
| PDF | Probability density function |
| PIE | Principle of inclusion and exclusion |

**Table 2.** Notation

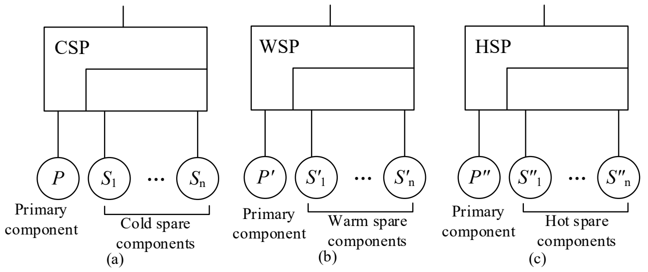
| | |
|---|---|
| $\theta_x$ | Component $X$ |
| $X$ | Failure of $\theta_x$ |
| $f_X(\tau_X)$ | Time-to-failure PDF of $\theta_X$ |
| $\theta_{\mathbb{T}}$ | Subsystem $\mathbb{T}$ |
| $\mathbb{T}$ | Failure of $\theta_{\mathbb{T}}$ |
| $\mathbb{T}\_C$ | Failure of $\theta_C$ in the $\theta_{\mathbb{T}}$ |
| $\Diamond$ | Boolean logic OR or AND |
| $+, \cdot, \neg$ | Boolean logic OR, AND, Negation |
| $\lhd$ | Temporal non-inclusive BEFORE |
| $rep(S, P)$ | Conditioning event, simplification of $rep(\theta_S, \theta_P)$, $\theta_S$ replaces $\theta_P$ in the SP gate. |
| $MCS_j[\mathbb{T}]$ | The $j^{\text{th}}$ MCS of fault tree of $\theta_{\mathbb{T}}$ |
| $\lvert MCS_j[\mathbb{T}]\rvert$ | The number of basic events in the $MCS_j[\mathbb{T}]$ |
| $MCS_j[\mathbb{T}]\_C_k$ | The $k^{\text{th}}$ basic event in $MCS_j[\mathbb{T}]$ |

Standby replacement is a widely used design technique for fault-tolerant systems, which can keep the system operational by supplying required functions even in the presence of hardware failures or software errors [1]. The standby sparing system is composed of one primary (online) component and one or more components that serve as standby spares. There are three types of spares: hot, warm, and cold. The hot spare is always in a working state, and it can replace a failed primary component immediately to keep the system operational. However, the cost of a hot spare is too high. It is not suitable for some places that suffer a shortage of power. The cold spare is a low-cost solution for standby sparing systems since it is powered off when it is in an inactive state, but it requires a long response time to replace a failed primary component. The warm spare is a compromised solution between hot and cold spares. A warm spare is in a degraded operational state before it is ready to replace a failed primary component. Hence, it has a lower cost than the hot spare and a quicker response time than the cold spare. A standby system with cold spares is called a

[*]Corresponding Author: Xiaohong Peng; E-mail: lgdpxh@126.com

DOI: 10.53106/160792642023072404011

cold standby system. Some real-world systems with power resource limitations such as satellite supply systems typically apply cold-standby systems.



(a) CSP gate (b) WSP gate (c) Hot gate

**Figure 1.** Three types of the SP gate

The dynamic fault tree (DFT) is an extension of the static fault tree (SFT), which introduced some dynamic gates such as the spare (SP) gate [2]. The DFT is used for reliability analysis for dynamic systems. The SP gate has three types: hot spare (HSP) gate, cold spare (CSP) gate, and warm spare (WSP) gate, which respectively matches HSP, CSP, and WSP, as shown in Figure 1. The DFT with CSP gates can model the cold standby system for reliability analysis. In essence, the calculation of the minimal cut sequence (MCQ) in the DFT is a permutation problem that is an exponential complexity. The calculation of the minimal cut set (MCS) in the SFT is a combination problem. In general, a combination problem has less average space complexity than a permutation problem. Hence, a conditional decision diagram (CBDD)-based method was proposed to reduce the complexity of qualitative analysis for the DFT with SP gates by replacing MCQs with MCSs in [3]. However, the analysis of the SP gate in the CBDD mainly focuses on whose inputs are all basic event and not top events of subtrees. To address this problem, the conditioning event is improved to describe the CSP gate whose inputs (primary or cold spare components) are top events of subtrees (outputs of other gates). Finally, an improved CBDD-based method is demonstrated to analyze the cold standby system modeled by the DFT with CSP gates whose inputs can be top events of subtrees.

The rest of the paper is organized as follows: Related works regarding SP gates, BDD, and TDD are introduced in Section 2. Section 3 presents the reliability analysis of CSP gates with subtrees inputs by the proposed formulae based on the extended conditioning event. Section 4 presents the reliability analysis of CSP gates with subtrees inputs based on the CBDD by complementing related operation rules. In Section 5, a case study is used to illustrate the reliability assessment based on the CBDD-based method. The conclusion is provided at the end.

## 2 Related Work

The DFT with SP gates has been used to model the failure behaviors of the standby system for reliability analysis. An algebraic structure-function based on temporal Boolean logic was proposed to do qualitative and quantitative analysis for DFTs including SP gates in [4]. To address the reliability analysis of a large cold-standby system, a fast approximation method based on the central limit theorem was proposed in [5]. An approach for reliability analysis of a standby system with multi-state elements subject to constant transition rates was proposed in [6]. In [7], the DFT with SP gates was analyzed for reliability by proposed stochastic computational models considering probabilistic common cause failures. In [8], the dynamic reliability characteristics of dormant systems with WSP were investigated by utilizing discrete-time Bayesian networks. In [9], two-unit cold standby systems were modeled by considering a periodic switching approach for the reliability analysis. In [10], a part of the hypothetical cardiac assist system modeled by a DFT with both a PAND gate and CSP gates was evaluated by algebraic structure functions considering the irrelevance coverage model. However, the methods mentioned above do not focus on the relationship between sequence-dependency of component failure and states. The MCQ remains to be used in these methods during qualitative analysis. Hence, this paper analyzes the DFT with CSP gates by a conditional fault tree (CFT), rendering the MCQ converted into MCS for qualitative analysis.

The binary decision diagram (BDD) has been widely used for the reliability analysis of the SFT. In [11], an algorithm was proposed to construct a worst-case reduced-ordered BDD, which may help projects decide if the BDD is the appropriate data structure. A variable order of the BDD was proposed in [12], which was used to analyze the reliability of the k-terminal network with imperfect vertices. In [13], a BDD was proposed to efficiently store data in the memory of a computer in the library based on a Boolean structure. A new heuristic ordering for BDD variables based on special types of fan-in 2 read-once formulas was presented in [14]. In [15], the authors presented an effective scheme for transforming the BDD representation of a Boolean function into a reversible circuit composed through reversible logic elements. In [16], BDDs were set to be elements of a newly lifted domain that were applied to analyze program families, the internal nodes were labeled with Boolean features and leaf nodes belong to an existing single-program analysis domain. An improved technique related to mapping the nodes of the BDD for any input Boolean function to the crossbar slices is proposed in [17]. In [18], the BDD was used for a genetic algorithm reordering optimizer, and it can iteratively process a large population with a randomized mixing of low destructive crossover/mutation operators. However, traditional BDD cannot be used for the DFT since basic events are no long independent. For DFTs, a sequential BDD (SBDD) was first proposed for the reliability analysis of a non-repairable cold-standby system in [19]. In [20], the SBDD was extended to analyze the reliability of warm-standby systems. An improved SBDD-based method was proposed to analyze the reliability of a DFT with the arbitrary tree structure in [21]. In [22], an algebraic binary decision diagram (ABDD) was proposed to do the reliability analysis of the DFT by introducing algebraic structure functions. An improved method based on component connection was proposed to construct SBDD encoding a DFT in [23]. In [24], a BDD-reordering optimization engine was used to drive a fast reversible circuit synthesis methodology, which was achieved by meta-heuristic optimization algorithms. In [25], the authors proposed a BDD model for the general structure systems having combinations of series, parallel, and standby structures by using a single node to denote a multistate component. In [26], a partial-order BDD-based method was proposed by introducing a new Boolean operator based on partial-order, which was used for the reliability analysis of DFT with PAND gates. However, the SBDDs and ABDDs remain to keep the sequence-dependency between component failure in their nodes. Also, SBDD does not give strict temporal operation rules. The ABDD does not eliminate inconsistencies during building. Hence, the path of the final ABDD may contain inconsistency, but it will be deleted when picking the MCSs or MCQs. The partial-order BDD-based method cannot use for DFTs with SP gates. Also, the CBDD did not present the reliability analysis of a CSP with the subtree structure in detail.

In this paper, cold-standby systems are modeled by DFTs with CSP gates. An improved CBDD-based method is presented to analyze the reliability of the CSP gate with subtrees inputs by extending the conditioning event *rep* to describe the replacement between subtrees.

# 3 CFT of the Cold-standby System with Sub-systems

## 3.1 Conditioning Event *rep*

A conditioning event of a fault tree is a normal event but not a fault event, which refers to some specific conditions or restrictions that apply to any logic gate [27].

*rep*($\theta_S$, $\theta_P$) is a conditioning event, which denotes a spare $\theta_S$ replaces a replaceable $\theta_P$ [3]. In the SP gate, a replaceable component is either an initial primary component or a working spare component. The working spare component is the initial spare component that has been activated. According to the description of the CSP gate in [2], when the replacement happens, it means that a replaceable component fails and the corresponding spare component does not fail at this time. The time diagram of *rep*($\theta_S$, $\theta_P$) is shown in Figure 2. In Figure 2, $t_P$ and $t_S$ respectively denote that the failure time of $\theta_P$ and $\theta_S$. Also, *rep*($\theta_S$, $\theta_P$) implies $\theta_P$ is failed.

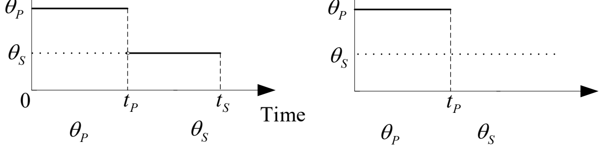


**Figure 2.** The time diagram of *rep*($\theta_S$, $\theta_P$)

The conditioning event can intuitively present the replacement behavior in the SP gates. Also, it can also show the failure sequence of the events. However, unless all the failure sequence-dependence of the primary and spare components are identified, the detailed replacement behavior cannot be confirmed. For example, in the CSP gates, two primary events $P_1$ and $P_2$ shared one spare component $S$, *rep*($\theta_S$, $\theta_P$) shows $P_1$ fails first and $S$ replaces $P_1$. However, we cannot confirm $S$ replaces $P_1$ if we only know $P_1$ fails since whether $P_2$ fails before $P_1$ is not confirmed.

For simplification, *rep*($S$, $P$) is instead of *rep*($\theta_S$, $\theta_P$). The $S$ and $P$ in rep respectively mean $\theta_S$ and $\theta_P$. A simple CSP gate can be converted into an SFT with conditioning events. To distinguish the traditional SFT, our SFT with conditioning events is called a conditional fault tree (CFT) [3]. The CFT is shown in Figure 3, and its top event can be expressed as $TE = P \cdot S \cdot rep(S, P) = S \cdot rep(S, P)$. $S \cdot rep(S, P)$ denotes that the output event of the CSP gate will occur if $\theta_S$ fails after it replaces $\theta_P$. Also, the case of shared spare components was described in [3].

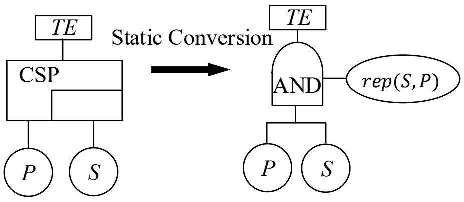


**Figure 3.** A simple CSP gate converting into CFT

To easily understand *rep*($S$, $P$), the algebraic structure-function with "◁" operation symbol [4] is borrowed to explain it. "◁" is a temporal operation symbol, which denotes a sequential relationship "Before". For example, $A \lhd B$ denotes event A occurs before event B, and B can either never occur or occur later. For the Figure 3, $S \cdot rep(S, P)$ can derive the algebraic structure-function as follows:

$$rep(S,P) \to P \lhd S$$
$$S \cdot rep(S,P) \to S \cdot (P \lhd S)$$
$$\Pr\{S \cdot rep(S,P)\} = \Pr\{S \cdot (P \lhd S)\}$$

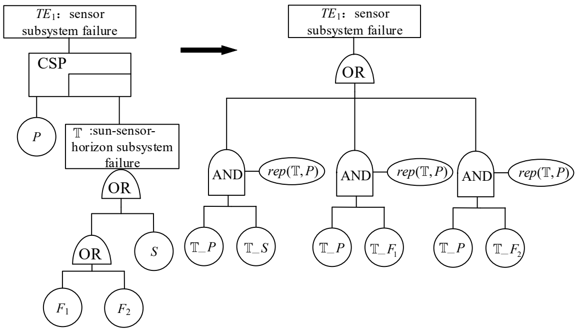
Pr{$S$·*rep*($S$, $P$)} and Pr{$S$·($P \lhd S$))denote the occurrence probability of $S \cdot rep(S, P)$ and $S \cdot (P \lhd S)$, respectively. Set $f_P(\tau_P)$ and $f_S(\tau_S)$ are time-to-failure probability density functions (PDFs) of $\theta_P$ and $\theta_S$, respectively. According to Figure 2(a), the probability of $S$·*rep*($S$, $P$) occurrence can be calculated by the following integral expression ($t$ is a mission time):

$$\Pr\{S \cdot rep(S,P)\} = \Pr\{S \cdot (P \lhd S)\}$$
$$= \int_0^t \int_0^{t-\tau_P} f_P(\tau_P) f_S(\tau_S) d_{\tau_P} d_{\tau_S}$$

## 3.2 CSP Gate with Subtrees Inputs and its CFT

### 3.2.1 Case of a Spare Component Being a Subtree

The previous conditioning event *rep* cannot directly describe the replacement between subtrees being primary or spare components. For example, there is a sensor subsystem of the satellite control system [28]. The sensor subsystem is a cold-standby system whose primary component is a star sensor and the spare component is a sun-sensor-horizon subsystem that consists of two infrared-horizon sensors and one sun sensor. The DFT of the sensor subsystem is shown in Figure 4(a). In Figure 4(a), P and S are two basic events that respectively denote star sensor and sun sensor failure. $F_1$ and $F_2$ are two basic events that respectively denote two infrared-horizon sensor failures.



(a) DFT of the sensor subsystem (b) CFT of the sensor subsystem

**Figure 4.** A sensor subsystem of the satellite control system

To improve the *rep* event to be used for the CSP with subtrees, $\theta_{\mathbb{T}}$ is set to denote that a subtree in the CSP gate. $rep(\mathbb{T}, P)$ denotes a subtree $\theta_{\mathbb{T}}$ being a spare component replaces a primary $\theta_P$ when $\theta_P$ is failed.

To distinguish a spare component corresponding to a basic event, the subtree (subsystem) being a spare component is called the spare subtree (subsystem) for short. The Boolean function ($TE_1$) with variables corresponding to condition events can be derived as follows:

$$\begin{aligned}
&\mathbb{T} = S + F_1 + F_2 \\
&TE_1 = \mathbb{T} \cdot rep(\mathbb{T}, P) \\
&TE_1 = (S + F_1 + F_2) \cdot rep(\mathbb{T}, P) \\
&TE_1 = S \cdot rep(\mathbb{T}, P) + F_1 \cdot rep(\mathbb{T}, P) + F_2 \cdot rep(\mathbb{T}, P). \qquad \textbf{(1)}
\end{aligned}$$

Note that T outside of the conditioning event denotes an internal event of a fault tree (a top event of a subtree). $\mathbb{T}$ inside of the conditioning event is a simplification of $\theta_{\mathbb{T}}$. To distinguish the basic event of a subtree in the CPS, a prefix symbol related to the subtree is complemented to the event in the Boolean expression such as $\mathbb{T}\_P$. $\mathbb{T}\_P$ denotes a basic event $S$ of the spare subtree $\mathbb{T}$ in the CSP gate. Hence, equation (1) is revised as follows:

$$\begin{aligned}
&\mathbb{T} = \mathbb{T}\_S + \mathbb{T}\_F_1 + \mathbb{T}\_F_2 \\
&TE_1 = \mathbb{T}\_S \cdot rep(\mathbb{T}, P) + \mathbb{T}\_F_1 \cdot rep(\mathbb{T}, P) \\
&+ \mathbb{T}\_F_2 \cdot rep(\mathbb{T}, P). \qquad \textbf{(2)}
\end{aligned}$$

In equation (2), the occurrence of $\mathbb{T}\_S \cdot rep(\mathbb{T}, P)$ denotes that $\theta_S$ (in $\theta_{\mathbb{T}}$) failure causes $\theta_{\mathbb{T}}$ to fail after subtree $\theta_{\mathbb{T}}$ replaces the primary $\theta_P$. In this case, the sensor subsystem fails. Similarly, the occurrence of either $\mathbb{T}\_F_1 \cdot rep(\mathbb{T}, P)$ or $\mathbb{T}\_F_2 \cdot rep(\mathbb{T}, P)$ can also cause the sensor subsystem to fail. The CFT of the sensor subsystem is shown in Figure 4(b). In the CSP gate, $\theta_{\mathbb{T}}$ is a spare subsystem and $\theta_P$ is a primary component. The time diagram of $rep(\mathbb{T}, P)$ is shown in Figure 5.

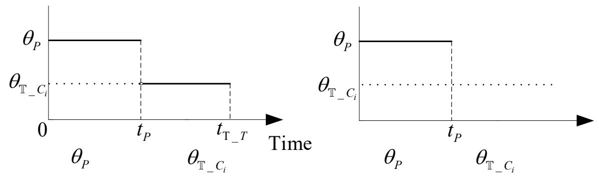

$\theta_P$
$\theta_{\mathbb{T}\_C_i}$
0
$t_P$
$t_{\mathbb{T}\_T}$
Time
$\theta_P$
$\theta_{\mathbb{T}\_C_i}$
$\theta_P$
$\theta_{\mathbb{T}\_C_i}$
$t_P$
$\theta_P$
$\theta_{\mathbb{T}\_C_i}$

**Figure 5.** Time diagram of $rep(\mathbb{T}, P)$

In Figure 5, $\theta_{\mathbb{T}\_C_i}$ denotes any component in $\theta_{\mathbb{T}}$. $t_{\mathbb{T}\_C_i}$ denotes a failure time of $\theta_{\mathbb{T}\_C_i}$. $t_P$ is a failure time of $\theta_P$. In the system modeled by the CSP gate, any component of a spare subsystem cannot fail before the spare subsystem replaces the primary component since the subsystem is un-power when it is on standby (inactivated). Thus, Event $\mathbb{T}\_C_i \lhd P$ cannot occur in the CSP gate. Hence, $rep(\mathbb{T}, P)$ is derived as follows:

$$rep(\mathbb{T}, P) \to \prod_{i=1}^{n} (P \lhd \mathbb{T}\_C_i). \qquad \textbf{(3)}$$

According to theorems related to "$\lhd$", $\mathbb{T}\_C_i \lhd P$ can be derived as follows:

$$\begin{aligned}
&P = P \lhd \mathbb{T}\_C_i + (\mathbb{T}\_C_i \lhd P) \cdot P \\
&(\mathbb{T}\_C_i \lhd P) \cdot P = 0 \\
&P = P \lhd \mathbb{T}\_C_i. \qquad \textbf{(4)}
\end{aligned}$$

Hence, for the CSP gate, according to equation (4), equation (3) is revised to be as below:

$$rep(\mathbb{T}, P) \to \prod_{i=1}^{n} (P \lhd \mathbb{T}\_C_i) = P. \qquad \textbf{(5)}$$

Eq. (5) denotes that once the primary component fails, the replacement will occur if there are enough spare components (subsystems) and switching is perfect. In other words, the failure occurrence of the primary component is no restriction in the CSP gate. Then, $\Pr\{\mathbb{T}\_C_i \cdot rep(\mathbb{T}, P)\}$ can be calculated as follows:

$$\begin{aligned}
&\mathbb{T}\_C_i \cdot rep(\mathbb{T}, P) \to \mathbb{T}\_C_i \cdot \prod_{j=1}^{n} (P \lhd \mathbb{T}\_C_j) \\
&P \lhd \mathbb{T}\_C_j = P, \quad when \quad j \neq i \\
&\mathbb{T}\_C_i \cdot \prod_{j=1}^{n} (P \lhd \mathbb{T}\_C_j) = \mathbb{T}\_C_i \cdot (P \lhd \mathbb{T}\_C_i) \cdot P \\
&\qquad = \mathbb{T}\_C_i \cdot (P \lhd \mathbb{T}\_C_i) \\
&\Pr\{\mathbb{T}\_C_i \cdot rep(\mathbb{T}, P)\} = \Pr\{\mathbb{T}\_C_i \cdot (P \lhd \mathbb{T}\_C_i)\}. \qquad \textbf{(6)}
\end{aligned}$$

In the CSP gate, the failure occurrence of the component in the spare subsystem is dependent on the failure of the primary component. Hence, in equation (6), sequence-dependency needs to be added between the $\theta_P$ and $\theta_{\mathbb{T}\_C_i}$ since $\theta_{\mathbb{T}\_C_i}$ starts to work at the time of $\theta_P$ failure.

Set $MCS_j[\mathbb{T}]$ to be the $j^{\text{th}}$ MCS of the SFT of $\theta_{\mathbb{T}}$, $|MCS_j[\mathbb{T}]|$ be the number of basic events in $MCS_j[\mathbb{T}]$, and

$MCS_j[\mathbb{T}]\_C_k$ be the k^th basic event in $MCS_j[\mathbb{T}]$. Assume that $\mathbb{T}$ has $n$ basic events and $m$ MCSs, then $\mathbb{T}\cdot rep(\mathbb{T}, P)$ can be calculated as follows:

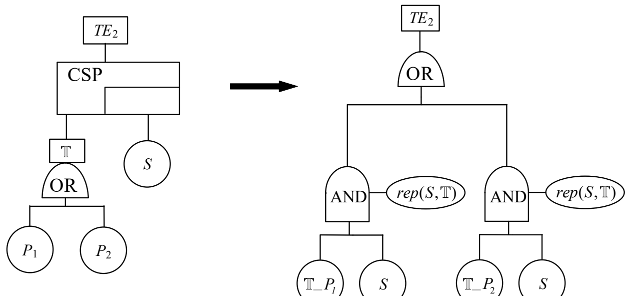
$$\begin{aligned}
&\mathbb{T}\cdot rep(\mathbb{T},P)\rightarrow \mathbb{T}\cdot\prod_{i=1}^{n}(P\lhd \mathbb{T}\_C_i)\\
&\mathbb{T}\cdot\prod_{i=1}^{n}(P\lhd \mathbb{T}\_C_i)=\\
&\sum_{j=1}^{m}(MCS_j[\mathbb{T}]\cdot\prod_{k=1}^{|MCS_j[\mathbb{T}]|}(P\lhd MCS_j[\mathbb{T}]\_C_k))\\
&\Pr\{\mathbb{T}\cdot rep(\mathbb{T},P)\}\\
&=\Pr\{\sum_{j=1}^{m}(MCS_j[\mathbb{T}]\cdot\prod_{k=1}^{|MCS_j[\mathbb{T}]|}(P\lhd MCS_j[\mathbb{T}]\_C_k))\}.
\end{aligned}\tag{7}$$

According to equation (7), the occurrence probability of $TE_1$ in Figure 4 can be calculated as follows:

$$\begin{aligned}
&MCS_1[\mathbb{T}]=S,\ MCS_2[\mathbb{T}]=F_1,\ MCS_3[\mathbb{T}]=F_2\\
&MCS_1[\mathbb{T}]\_C_1=S, MCS_2[\mathbb{T}]\_C_1=F_1,\ MCS_3[\mathbb{T}]\_C_1=F_2\\
&TE_1=MCS_1[\mathbb{T}]\cdot(P\lhd S)+MCS_2[\mathbb{T}]\cdot(P\lhd F_1)\\
&+MCS_3[\mathbb{T}]\cdot(P\lhd F_2)\\
&\Pr\{TE_1\}=\Pr\{(P\lhd S)\cdot S+(P\lhd F_1)\cdot F_1+(P\lhd F_2)\cdot F_2\}
\end{aligned}$$

### 3.2.2 Case of a Primary Component Being a Subtree

In the CSP gate, the primary component can also be a subtree. Similarly, the subsystem (subtree) being a primary component is called the primary subsystem (subtree) for short. In the CSP, any component in the spare subsystem is impossible to fail before the primary component fails. However, not all components in the primary subsystem fail before the spare component or subsystem fails. It is decided by the structure of the primary subsystem. For example, a CSP gate with a primary subtree is shown in Figure 6(a).



(a) CSP gate with a primary subsystem of logic OR structure

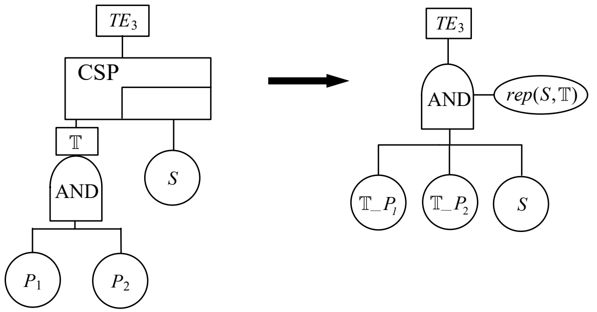
(b) CFT of the left CSP gate

**Figure 6.** A CSP gate with a primary subsystem of logic OR structure and its CFT.

In Figure 6(a), the failure of $\theta_{P_1}$ or $\theta_{P_2}$ will cause the primary $\theta_{\mathbb{T}}$ to fail. Then, the spare $\theta_S$ replaces the $\theta_{\mathbb{T}}$. Hence, the probability occurrence of $TE_2$ can be derived as follows:

$$\begin{aligned}
&\mathbb{T}=P_1+P_2\\
&rep(S,\mathbb{T})\rightarrow\mathbb{T}\lhd S=(P_1+P_2)\lhd S=P_1\lhd S+P_2\lhd S\\
&\mathbb{T}\cdot S\cdot rep(S,\mathbb{T})=\mathbb{T}\_P_1\cdot S\cdot rep(S,\mathbb{T})+\mathbb{T}\_P_2\cdot S\cdot rep(S,\mathbb{T})\\
&=\neg\mathbb{T}\_P_2\cdot\mathbb{T}\_P_1\cdot S\cdot rep(S,\mathbb{T})+\neg\mathbb{T}\_P_1\cdot\mathbb{T}\_P_2\cdot S\cdot rep(S,\mathbb{T})\\
&\quad+\mathbb{T}\_P_1\cdot\mathbb{T}\_P_2\cdot S\cdot rep(S,\mathbb{T})\\
&\neg\mathbb{T}\_P_2\cdot\mathbb{T}\_P_1\cdot S\cdot rep(S,\mathbb{T})\rightarrow((P_1+P_2)\lhd S)\cdot\neg P_2\cdot P_1\cdot S\\
&((P_1+P_2)\lhd S)\cdot\neg P_2\cdot P_1\cdot S=(P_1\lhd S)\cdot\neg P_2\cdot S\\
&\Pr\{TE_2\}=\Pr\{S\cdot rep(S,\mathbb{T})\}=\Pr\{(P_1\lhd S)\cdot S+(P_2\lhd S)\cdot S\}.
\end{aligned}\tag{8}$$

In equation (8), $\neg\mathbb{T}\_P_2\cdot\mathbb{T}\_P_1\cdot S\cdot rep(S,\mathbb{T})$ denotes that only the failure of $\theta_{P_1}$ causes the $\theta_{\mathbb{T}}$ failure, which renders a spare $\theta_S$ to replace $\theta_{\mathbb{T}}$, then $\theta_S$ fails later. According to equation (7), the CFT of Figure 6(a) is shown in Figure 6(b). However, it will be different if the primary subsystem has a logic AND structure, as shown in Figure 7.



(a) CSP gate with a primary subsystem of logic AND structure

(b) CFT of the left CSP gate

**Figure 7.** A CSP gate with a primary subsystem of logic AND structure and its CFT

In Figure 7, the failure of $\theta_{P_1}$ and $\theta_{P_2}$ will cause the primary $\theta_{\mathbb{T}}$ to fail. Then, the spare $\theta_S$ replaces the $\theta_{\mathbb{T}}$. Hence, the probability occurrence of $TE_3$ can be derived as follows:

$$\begin{aligned}
&\mathbb{T}=P_1\cdot P_2\\
&rep(S,\mathbb{T})\rightarrow\mathbb{T}\lhd S=(P_1\cdot P_2)\lhd S=(P_1\lhd S)\cdot(P_2\lhd S)\\
&\mathbb{T}\cdot S\cdot rep(S,\mathbb{T})=\mathbb{T}\_P_1\cdot\mathbb{T}\_P_2\cdot S\cdot rep(S,\mathbb{T})\\
&\mathbb{T}\_P_1\cdot\mathbb{T}\_P_2\cdot S\cdot rep(S,\mathbb{T})=(P_1\lhd S)\cdot(P_2\lhd S)\cdot S\\
&\Pr\{TE_3\}=\Pr\{S\cdot rep(S,\mathbb{T})\}=\Pr\{(P_1\lhd S)\cdot(P_2\lhd S)\cdot S\}.
\end{aligned}\tag{9}$$

In equation (9), $\mathbb{T}\_P_1\cdot\mathbb{T}\_P_2\cdot S\cdot rep(S,\mathbb{T})$ denotes that the failure of $\theta_{P_1}$ and $\theta_{P_2}$ causes the $\theta_{\mathbb{T}}$ failure, which renders a spare $\theta_S$ to replace $\theta_{\mathbb{T}}$, then $\theta_S$ fails later. According to equation (9), the CFT of Figure 7(a) is shown in Figure 7(b).

In the CSP gate, $\theta_{\mathbb{T}}$ is a primary subsystem and $\theta_S$ is a spare component. The time diagram of $rep(S, \mathbb{T})$ is shown in Figure 8.

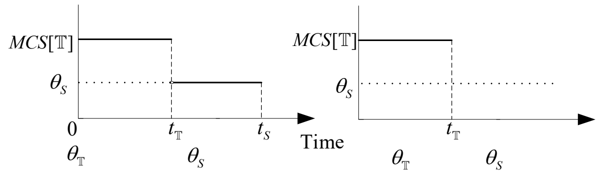


**Figure 8.** The time diagram of *rep*(*S*, 𝕋)

In Figure 8, *MCS*[𝕋] denotes any MCS of the SFT of $\theta_{\mathbb{T}}$. $t_{\mathbb{T}}$ and $t_S$ respectively denotes the failure time of $\theta_{\mathbb{T}}$ and $\theta_S$. Figure 8(a) presents that the occurrence of any MCS of the SFT of $\theta_{\mathbb{T}}$ can cause $\theta_{\mathbb{T}}$ to fail, which leads the *MCS*[𝕋]·*rep*(*S*, 𝕋) to happen, then $\theta_S$ fails later. Figure 8(b) presents that the primary $\theta_{\mathbb{T}}$ failure caused by its MCS leads the replacement to happen and $\theta_S$ never fails. Hence, *S* ·*rep*(*S*, 𝕋) can be derived as follows:

$$rep(S,\mathbb{T}) \to \mathbb{T} \lhd S = \sum_{j=1}^{m}(MCS_j[\mathbb{T}] \lhd S) = \sum_{j=1}^{m} MCS_j[\mathbb{T}]$$

$$S \cdot rep(S,\mathbb{T}) \to S \cdot \sum_{j=1}^{m}(MCS_j[\mathbb{T}] \lhd S). \quad \textbf{(10)}$$

### 3.2.3 Case of All Inputs Being Subtrees

Assume that a CSP gate with a primary $\mathbb{T}_P$ and a spare $\mathbb{T}_S$. According to equations (7) and (10), it can be obtained as follows:

$$\mathbb{T}_S \cdot rep(\mathbb{T}_S,\mathbb{T}_P) = \sum_{i=1}^{n} MCS_i[\mathbb{T}_P] \cdot \sum_{j=1}^{m} MCS_j[\mathbb{T}_S] \cdot rep(\mathbb{T}_S,\mathbb{T}_P)$$

$$\to \sum_{i=1}^{n}(\sum_{j=1}^{m} MCS_j[\mathbb{T}_S] \cdot \prod_{k=1}^{|MCS_j[\mathbb{T}_S]|}(MCS_i[\mathbb{T}_P] \lhd MCS_j[\mathbb{T}_S]\_C_k)). \quad \textbf{(11)}$$

Figure 9 shows a DFT of solar wing deployment in a satellite power system [29]. $\mathbb{T}_P$ and $\mathbb{T}_S$ denote primary and spare actuating mechanism failures, respectively. $P_1$ and $P_2$ respectively denote that the drive mechanism is stuck and mechanical fracture. $S_1$ and $S_2$ denote the same failures in the spare actuating mechanism, respectively.

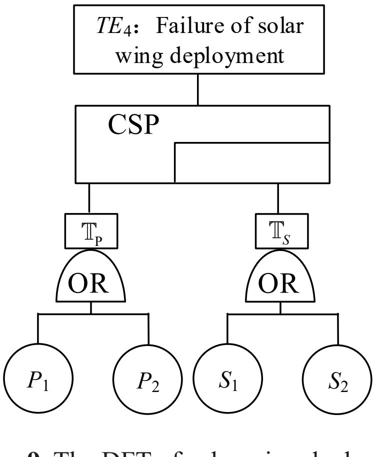


**Figure 9.** The DFT of solar wing deployment

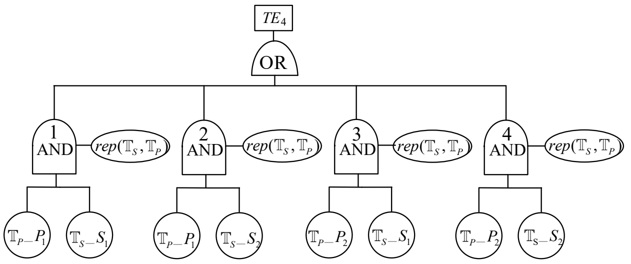


**Figure 10.** The CFT of solar wing deployment

According to equation (11), the CFT of solar wing deployment is shown in Figure 10, and the Pr{$TE_4$} can be calculated as follows:

$$\begin{aligned}
&\mathbb{T}_P = P_1 + P_2,\ \ \mathbb{T}_S = S_1 + S_2\\
&MCS_1[\mathbb{T}_P] = P_1, MCS_2[\mathbb{T}_P] = P_2\\
&MCS_1[\mathbb{T}_S] = S_1, MCS_2[\mathbb{T}_S] = S_2\\
&MCS_1[\mathbb{T}_P]\_C_1 = P_1, MCS_2[\mathbb{T}_P]\_C_1 = P_2\\
&MCS_1[\mathbb{T}_S]\_C_1 = S_1, MCS_2[\mathbb{T}_S]\_C_1 = S_2\\
&\Pr\{TE_4\} = \Pr\{\mathbb{T}_S \cdot rep(\mathbb{T}_S,\mathbb{T}_P)\}\\
&= \Pr\{(P_1 \lhd S_1)\cdot S_1 + (P_1 \lhd S_2)\cdot\\
&S_2 + (P_2 \lhd S_1)\cdot S_1 + (P_2 \lhd S_2)\cdot S_2\}.
\end{aligned} \quad \textbf{(12)}$$

### 3.2.4 Case of a Shared Spare Component Being a Subtree

Figure 11 shows a sensor subsystem in the medium accuracy attitude determination system of a satellite. It consists of an infrared horizon sensor and a sun sensor, which respectively is a cold standby system. These two cold standby systems share a spare magnetometer subsystem. The primary subsystem of the infrared horizon sensor has two infrared sensors, and a spare magnetometer subsystem will replace the primary subsystem if any of them fails and the spare subsystem is available. The sensor subsystem fails if any of them fails.

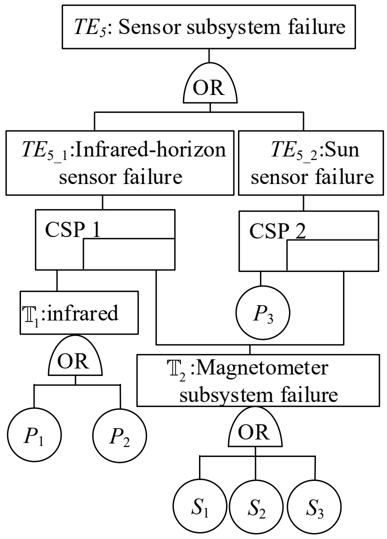

$TE_5$: Sensor subsystem failure
OR
$TE_{5\_1}$:Infrared-horizon sensor failure
$TE_{5\_2}$:Sun sensor failure
CSP 1
CSP 2
$\mathbb{T}_1$:infrared
$P_3$
OR
$\mathbb{T}_2$:Magnetometer subsystem failure
$P_1$
$P_2$
OR
$S_1$
$S_2$
$S_3$

**Figure 11.** The DFT of a sensor subsystem

In the CSP gate with shared spare components (subsystems), the current spare component (subsystem) will replace the first failure of the primary component (subsystem). According to equation (11), while considering the spare competition, $\Pr\{TE_{5\_1}\}$ can be derived as follows:

$$\begin{aligned}
&\mathbb{T}_1 = P_1 + P_2,\ \ \mathbb{T}_2 = S_1 + S_2 + S_3\\
&MCS_1[\mathbb{T}_1] = P_1, MCS_2[\mathbb{T}_1] = P_2\\
&MCS_1[\mathbb{T}_2] = S_1, MCS_2[\mathbb{T}_2] = S_2, MCS_3[\mathbb{T}_2] = S_3\\
&\Pr\{TE_{5\_1}\} = \Pr\{\mathbb{T}_2 \cdot rep(\mathbb{T}_2, \mathbb{T}_1) + \mathbb{T}_1 \cdot rep(\mathbb{T}_2, P_3)\}\\
&= \Pr\{MCS_1[\mathbb{T}_2]\cdot\\
&(MCS_1[\mathbb{T}_1] \lhd MCS_1[\mathbb{T}_2]) \cdot (MCS_1[\mathbb{T}_1] \lhd P_3)\\
&+MCS_2[\mathbb{T}_2] \cdot (MCS_1[\mathbb{T}_1] \lhd MCS_2[\mathbb{T}_2]) \cdot (MCS_1[\mathbb{T}_1] \lhd P_3)\\
&+MCS_3[\mathbb{T}_2] \cdot (MCS_1[\mathbb{T}_1] \lhd MCS_3[\mathbb{T}_2]) \cdot (MCS_1[\mathbb{T}_1] \lhd P_3)\\
&+MCS_1[\mathbb{T}_2] \cdot (MCS_2[\mathbb{T}_1] \lhd MCS_1[\mathbb{T}_2]) \cdot (MCS_2[\mathbb{T}_1] \lhd P_3)\\
&+MCS_2[\mathbb{T}_2] \cdot (MCS_2[\mathbb{T}_1] \lhd MCS_2[\mathbb{T}_2]) \cdot (MCS_2[\mathbb{T}_1] \lhd P_3)\\
&+MCS_3[\mathbb{T}_2] \cdot (MCS_2[\mathbb{T}_1] \lhd MCS_3[\mathbb{T}_2]) \cdot (MCS_2[\mathbb{T}_1] \lhd P_3)\\
&+MCS_1[\mathbb{T}_1] \cdot (P_3 \lhd (MCS_1[\mathbb{T}_1] + MCS_2[\mathbb{T}_1]))\\
&+MCS_2[\mathbb{T}_1] \cdot (P_3 \lhd (MCS_1[\mathbb{T}_1] + MCS_2[\mathbb{T}_1]))\}. \qquad \mathbf{(13)}
\end{aligned}$$

Similarly, the $\Pr\{TE_{5\_2}\}$ can be derived as follows:

$$\begin{aligned}
&\Pr\{TE_{5\_2}\} = \Pr\{\mathbb{T}_2 \cdot rep(\mathbb{T}_2, P_3) + P_3 \cdot rep(\mathbb{T}_2, \mathbb{T}_1)\}\\
&= \Pr\{(P_3 \lhd (MCS_1[\mathbb{T}_1] + MCS_2[\mathbb{T}_1]))\cdot\\
&(P_3 \lhd MCS_1[\mathbb{T}_2]) \cdot MCS_1[\mathbb{T}_2]\\
&+(P_3 \lhd (MCS_1[\mathbb{T}_1] + MCS_2[\mathbb{T}_1])) \cdot (P_3 \lhd MCS_2[\mathbb{T}_2]) \cdot MCS_2[\mathbb{T}_2]\\
&+(P_3 \lhd (MCS_1[\mathbb{T}_1] + MCS_2[\mathbb{T}_1])) \cdot (P_3 \lhd MCS_3[\mathbb{T}_2]) \cdot MCS_3[\mathbb{T}_2]\\
&+P_3 \cdot (MCS_1[\mathbb{T}_1] \lhd P_3) + P_3 \cdot (MCS_2[\mathbb{T}_1] \lhd P_3)\\
&+P_3 \cdot (MCS_3[\mathbb{T}_1] \lhd P_3)\}. \qquad \mathbf{(14)}
\end{aligned}$$

Hence, according to equations (13) and (14), the CFT of the sensor subsystem is shown as Figure 12 and the $\Pr\{TE_5\}$ can be derived as follows:

$$\Pr\{TE_5\} = \Pr\{TE_{5\_1} + TE_{5\_2}\}$$

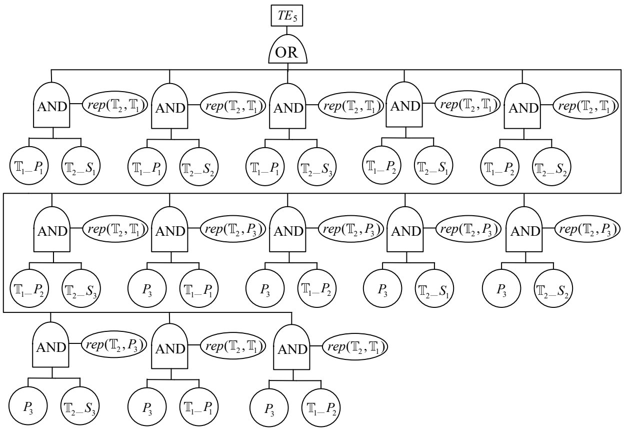


**Figure 12.** The CFT of A sensor subsystem

# 4 Improved Conditional BDD

In sub-section 3.2, to compute the occurrence probability of the top event, the principle of inclusion and exclusion (PIE) needs to be used for equations while considering repeated and dependent events such as equations (8) and (12). In general, PIE needs to compute $2^n$-1 items if there are n products in a Boolean expression. However, it may cause a combinatorial explosion if the products in the equations are too high. In general, a sum-of-disjoint-products (SDPs)-based method is a well known solution that can avoid a combinatorial explosion. The BDD can naturally generate the SDPs since it is a rooted acyclic graph based on the Shannon decomposition, which is as follows:

$$f = x \cdot f_{x=1} + \neg x \cdot f_{x=0} = ite(x, F_1, F_0). \qquad \mathbf{(15)}$$

In equation (15), $f$ denotes a Boolean expression of a fault tree. $x$ is a variable in $f$. $f_{x=1}$ and $f_{x=0}$ ($F_1$ and $F_0$) respectively denote $f$ evaluated as $x$ being 1 and 0. ite represents the concise *if−then−else* format. The *ite* format of BDD can be expressed by $F$ as follows:

$$F = x_0 \cdot F_{x=0} + x_1 \cdot F_{x=1} = ite(x, F_0, F_1)$$

CBDD is an extension of a BDD, which contains s-dependent nodes. However, it is assumed that all nodes in the CBDD are $s$-independence since the CBDD is only considered to obtain the formula which is in the form of SDP. The internal node of CBDD can be either a basic event or a conditioning event. CBDD has two edges: 0-edge and 1-edge, which presents the event occurrence and non-occurrence. The terminal CBDD is Boolean 0 and 1. The prime CBDD and general CBDD is shown in Figure 13.

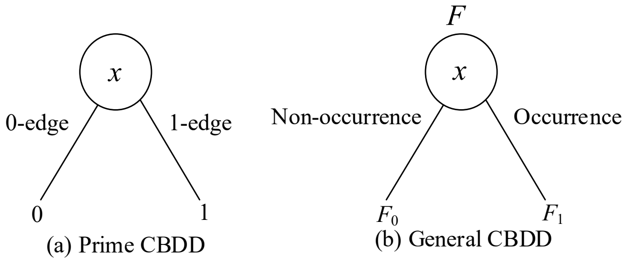


**Figure 13.** Prime CBDD and General CBDD

Two CBDDs can be operated by Boolean operation rules. Set G and H to be two CBDDs, respectively. $G = ite(x, G_0, G_1)$ and $H = ite(y, H_0, H_1)$. Let ◊ represent any Boolean logic operation (AND/OR), then we have $G \Diamond H = ite(x, G_0, G_1) \Diamond ite(y, H_0, H_1) =$

$$\begin{cases} ite(x, G_0 \Diamond H_0, G_1 \Diamond H_1), Index(x) = Index(y);\\ ite(x, G_0 \Diamond H, G_1 \Diamond H), Index(x) < Index(y);\\ ite(y, G \Diamond H_0, G \Diamond H_1), Index(x) > Index(y); \end{cases} \qquad \mathbf{(16)}$$

Here, $Index(x)$ and $Index(y)$ respectively denote the variable order of $x$ and $y$ in the CBDD. Same with the BDD, the order of input variables is also heavily relevant to the size of the CBDD. However, it is not the point focused in this paper.

According to equation (16), the recursive operation can be used between two sub-CBDD until one of them becomes a terminal CBDD. However, some inconsistency and redundancy issues need to solve since some internal nodes of the CBDD are *s*-dependent. The operation rules for improved *rep* conditioning events are set based on the rules related to *rep* in [3] while only considering the case of the CSP gate with subtrees inputs, as follows:

$$\mathbb{T}_2 \cdot rep(\mathbb{T}_1, \mathbb{T}_2) = rep(\mathbb{T}_1, \mathbb{T}_2). \tag{17}$$

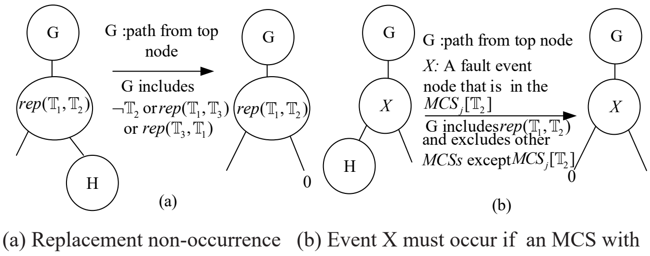
$$\neg\mathbb{T}_2 \cdot rep(\mathbb{T}_1, \mathbb{T}_2) = 0. \tag{18}$$

$$rep(\mathbb{T}_1, \mathbb{T}_2) \cdot rep(\mathbb{T}_1, \mathbb{T}_3) = 0. \tag{19}$$

$$rep(\mathbb{T}_1, \mathbb{T}_2) \cdot rep(\mathbb{T}_3, \mathbb{T}_2) = 0. \tag{20}$$

$$\neg rep(\mathbb{T}_1, \mathbb{T}_2) = \neg\mathbb{T}_2 \ . \tag{21}$$

Equation (17) denotes that the replacement can only happen when the replaceable $\theta_{\mathbb{T}_2}$ (an operational primary subsystem, or an operational spare subsystem after replacement occurrence) fails. Equation (18) denotes that if a replaceable $\theta_{\mathbb{T}_2}$ is operational, then it is impossible to be replaced. A replaceable $\theta_{\mathbb{T}_2}$ cannot be replaced twice and a spare $\theta_{\mathbb{T}_1}$ cannot replace two replaceable subsystems, which refers to Equations (19) and (20), respectively. Equation (21) denotes that if the replacement between $\theta_{\mathbb{T}_1}$ and $\theta_{\mathbb{T}_2}$ does not happen in the CSP gate when there is an available spare $\theta_{\mathbb{T}_1}$, the replaceable $\theta_{\mathbb{T}_2}$ is operational. Inconsistent elimination rules related to the CBDD based on equations (18) to (20) are shown in Figure 14.


G
G :path from top node
G includes
$\neg\mathbb{T}_2$ or $rep(\mathbb{T}_1,\mathbb{T}_3)$
or $rep(\mathbb{T}_3,\mathbb{T}_1)$
$rep(\mathbb{T}_1,\mathbb{T}_2)$
H
0
(a)
X
G :path from top node
X: A fault event node that is in the $MCS_j[\mathbb{T}_2]$
G includes $rep(\mathbb{T}_1,\mathbb{T}_2)$ and excludes other MCSs except $MCS_j[\mathbb{T}_2]$
(b)

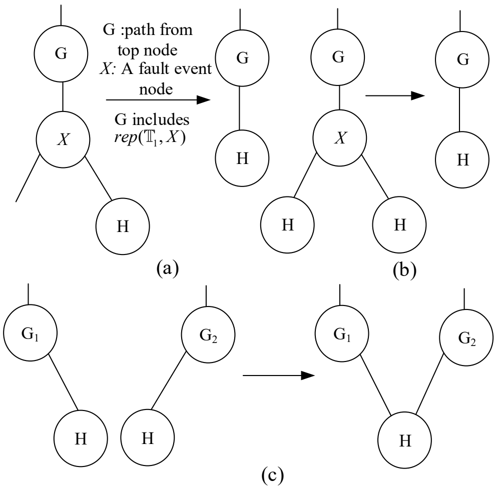
(a) Replacement non-occurrence (b) Event X must occur if an MCS with the event occurs

**Figure 14.** Inconsistent elimination rules of the CBDD

Simplification rules of CBDD considering CSP gate with subtrees inputs are shown in Figure 15.

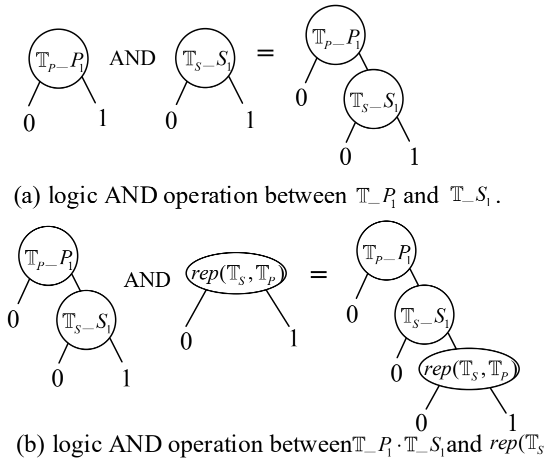
According to the order of variables $\mathbb{T}_P\_P_1 < \mathbb{T}_P\_P_2 < \mathbb{T}_S\_S_1 < \mathbb{T}_S\_S_2 < rep(\mathbb{T}_S, \mathbb{T}_P)$, the following steps are used to build a CBDD based on the CFT shown in Figure 10. Subtree 1 is created as shown in Figure 16.

Similar to the steps in Figure 16, subtrees 2, 3, and 4 can be built. The sub-CBDD-1 is generated by logic OR operation between subtrees 1 and 2, and the simplification rules, as shown in Figure 17.



(a) Replacement means the primary component fails (b) Child Nodes with the same parent merging (c) Child Nodes with different parents merging

**Figure 15.** Simplification rules of the CBDD



(a) logic AND operation between $\mathbb{T}\_P_1$ and $\mathbb{T}\_S_1$ .

(b) logic AND operation between$\mathbb{T}_P_1 \cdot \mathbb{T}_S_1$and $rep(\mathbb{T}_S, \mathbb{T}_P)$ .

**Figure 16.** The generation of subtree 1 in Figure 10

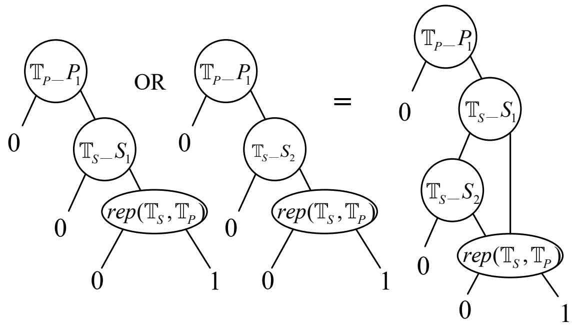


**Figure 17.** The sub-CBDD constructed by logic OR operation between subtrees 1 and 2

The other sub-CBDD-2 can be constructed by logic OR operation between subtrees 3 and 4. The final CBDD of the CFT is generated by logic OR operation between sub-CBDD-1 and sub-CBDD-2, as shown in Figure 18.

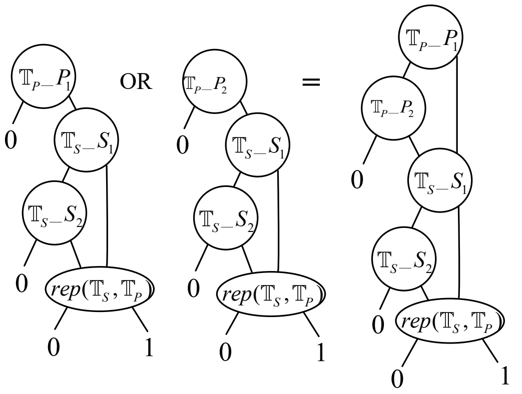


**Figure 18.** The final CBDD of the CFT is shown in Figure 10

In Figure 18, there are 4 paths from the top node to terminal node 1, as follows:

①$\mathbb{T}_P_P_1 \rightarrow \mathbb{T}_S_S_1 \rightarrow rep(\mathbb{T}_S,\mathbb{T}_P)$

②$\mathbb{T}_P_P_1 \rightarrow \neg\mathbb{T}_S_S_1 \rightarrow \mathbb{T}_S_S_2 \rightarrow rep(\mathbb{T}_S,\mathbb{T}_P)$

③$\neg\mathbb{T}_P_P_1 \rightarrow \mathbb{T}_P_P_2 \rightarrow \mathbb{T}_S_S_1 \rightarrow rep(\mathbb{T}_S,\mathbb{T}_P)$

④$\neg\mathbb{T}_P_P_1 \rightarrow \mathbb{T}_P_P_2 \rightarrow \neg\mathbb{T}_S_S_1 \rightarrow \mathbb{T}_S_S_2 \rightarrow rep(\mathbb{T}_S,\mathbb{T}_P)$

The MCS can be obtained by excluding negation CBDD nodes from these paths. There are four MCSs, as follows:

①$\mathbb{T}_P_P_1 \cdot \mathbb{T}_S_S_1 \cdot rep(\mathbb{T}_S,\mathbb{T}_P)$

②$\mathbb{T}_P_P_1 \cdot \mathbb{T}_S_S_2 \cdot rep(\mathbb{T}_S,\mathbb{T}_P)$

③$\mathbb{T}_P_P_2 \cdot \mathbb{T}_S_S_1 \cdot rep(\mathbb{T}_S,\mathbb{T}_P)$

④$\mathbb{T}_P_P_2 \cdot \mathbb{T}_S_S_2 \cdot rep(\mathbb{T}_S,\mathbb{T}_P)$

Since these paths are mutually exclusive, the occurrence probability of $TE_4$ can be computed by the following functions

$$
\begin{aligned}
&\Pr\{TE_4\} = \Pr\{\mathbb{T}_P\_P_1 \cdot \mathbb{T}_S\_S_1 \cdot rep(\mathbb{T}_S,\mathbb{T}_P)\}\\
&+(1-\Pr\{\mathbb{T}_S\_S_1\})\cdot \Pr\{\mathbb{T}_P\_P_1 \cdot \mathbb{T}_S\_S_2 \cdot rep(\mathbb{T}_S,\mathbb{T}_P)\}\\
&+(1-\Pr\{\mathbb{T}_P\_P_1\})\cdot \Pr\{\mathbb{T}_P\_P_2 \cdot \mathbb{T}_S\_S_1 \cdot rep(\mathbb{T}_S,\mathbb{T}_P)\}\\
&+(1-\Pr\{\mathbb{T}_P\_P_1\})\cdot (1-\Pr\{\mathbb{T}_S\_S_1\})\\
&\cdot \Pr\{\mathbb{T}_P\_P_2 \cdot \mathbb{T}_S\_S_2 \cdot rep(\mathbb{T}_S,\mathbb{T}_P)\}\\
&=\int_0^t \int_0^{t-\tau_{p_1}} f_{P_1}(\tau_{p_1}) f_{S_1}(\tau_{S_1}) d_{\tau_{S_1}} d_{\tau_{p_1}}\\
&+(1-\int_0^t f_{S_1}(\tau_{S_1}) d_{\tau_{S_1}})\cdot \int_0^t \int_0^{t-\tau_{p_1}} f_{P_1}(\tau_{p_1}) f_{S_2}(\tau_{S_2}) d_{\tau_{S_2}} d_{\tau_{p_1}}\\
&+(1-\int_0^t f_{P_1}(\tau_{P_1}) d_{\tau_{P_1}})\cdot \int_0^t \int_0^{t-\tau_{p_2}} f_{P_2}(\tau_{p_2}) f_{S_1}(\tau_{S_1}) d_{\tau_{S_1}} d_{\tau_{p_2}}\\
&+(1-\int_0^t f_{P_1}(\tau_{P_1}) d_{\tau_{P_1}})\cdot (1-\int_0^t f_{S_1}(\tau_{S_1}) d_{\tau_{S_1}})\\
&\cdot \int_0^t \int_0^{t-\tau_{p_2}} f_{P_2}(\tau_{p_2}) f_{S_2}(\tau_{S_2}) d_{\tau_{S_2}} d_{\tau_{p_2}}
\end{aligned}
$$

## 5 Case Study

No. 1 Hualong auxiliary water supply is a safety-critical system in the nuclear power plant, which supplies water to the secondary side of the steam generator for exportation of core waste heat if the main water supply system fails.

Figure 19 shows a DFT of the auxiliary feedwater pump subsystem in the No. 1 Hualong auxiliary feedwater system [30]. The subsystem fails if the primary electric pump fails and the spare pneumatic pump fails later. The primary electric pump fails if the pump function fails or the power subsystem fails. The power subsystem failure fails if the power fails and the diesel generator fails. The spare pneumatic pump failure fails if the pump function fails or the steam turbine fails.

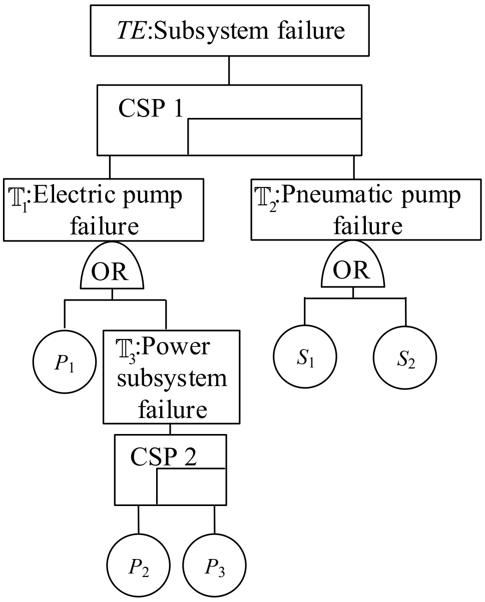

*TE*:Subsystem failure
CSP 1
$\mathbb{T}_1$:Electric pump failure
$\mathbb{T}_2$:Pneumatic pump failure
OR
OR
$P_1$
$\mathbb{T}_3$:Power subsystem failure
$S_1$
$S_2$
CSP 2
$P_2$
$P_3$

**Figure 19.** DFT of the auxiliary feedwater pump subsystem

The DFT includes two cascade CSP gates and two logic OR gates. The CSP 1 gate with two subtrees inputs $\mathbb{T}_1$ and $\mathbb{T}_2$ which are primary and spare, respectively. $\mathbb{T}_1$ and $\mathbb{T}_2$ denote the failure of the primary electric pump and the failure of the spare pneumatic pump, respectively. $\mathbb{T}_3$ denotes the failure of the power subsystem. $P_1$ denotes the functional failure of the pump. $P_2$ denotes power failure. $P_3$ denotes the failure of the diesel generator. $S_1$ and $S_2$ respectively denote the functional failure of the pump and steam turbine failure.

The system fails if $\mathbb{T}_2$ replaces $\mathbb{T}_1$ then $\mathbb{T}_2$ occurs. $\mathbb{T}_2$ occurs if $S_2$ replaces $S_1$ then $S_2$ fails. The conditioning event expression is

$$
\begin{aligned}
&\mathbb{T}_1 = P_1 + \mathbb{T}_3, \mathbb{T}_2 = S_1 + S_2, \mathbb{T}_3 = P_3 \cdot rep(P_3, P_2)\\
&MCS_1[\mathbb{T}_1] = P_1, MCS_2[\mathbb{T}_1] = \mathbb{T}_3\\
&MCS_1[\mathbb{T}_2] = S_1, MCS_2[\mathbb{T}_2] = S_2\\
&MCS_1[\mathbb{T}_3] = P_3 \cdot rep(P_3, P_2)
\end{aligned}
$$

$$\begin{aligned}
TE &= \mathbb{T}_2 \cdot rep(\mathbb{T}_2, \mathbb{T}_1) \\
&\overset{euqation\ (11)}{=} \sum_{i=1}^{2} MCS_i[\mathbb{T}_1] \cdot \sum_{j=1}^{2} MCS_j[\mathbb{T}_2] \cdot rep(\mathbb{T}_2, \mathbb{T}_1) \\
&= (\mathbb{T}_1\_P_1 + \mathbb{T}_1\_P_3 \cdot rep(P_3, P_2)) \\
&\cdot(\mathbb{T}_2\_S_1 + \mathbb{T}_2\_S_2 \cdot rep(\mathbb{T}_2, \mathbb{T}_1)) \\
&+\mathbb{T}_1\_P_1 \cdot \mathbb{T}_2\_S_1 \cdot rep(\mathbb{T}_2, \mathbb{T}_1) + \mathbb{T}_1\_P_1 \cdot \mathbb{T}_2\_S_2 \cdot rep(\mathbb{T}_2, \mathbb{T}_1) \\
&+\mathbb{T}_1\_\mathbb{T}_3\_P_1 \cdot \mathbb{T}_2\_S_1 \cdot rep(P_3, P_2) \cdot rep(\mathbb{T}_2, \mathbb{T}_1) \\
&+\mathbb{T}_1\_\mathbb{T}_3\_P_3 \cdot \mathbb{T}_2\_S_2 \cdot rep(P_3, P_2) \cdot rep(\mathbb{T}_2, \mathbb{T}_1) \qquad \textbf{(22)}
\end{aligned}$$

According to equation (22), the CFT of the auxiliary feedwater pump subsystem is shown in Figure 20.

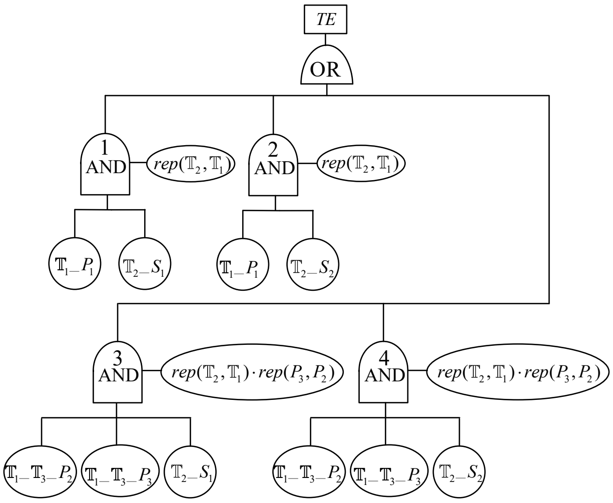


**Figure 20.** The CFT of the auxiliary feedwater pump subsystem

According to the CFT and order of variables $\mathbb{T}_P\_P_1 < \mathbb{T}_2\_S_1 < \mathbb{T}_2\_S_2 < \mathbb{T}_1\_\mathbb{T}_3\_P_2 < \mathbb{T}_1\_\mathbb{T}_3\_P_3 < rep(\mathbb{T}_2, \mathbb{T}_1) < rep(P_3, P_2)$, the CBDD can be constructed by recursively using equation (16) with operations related to inconsistent elimination and simplification. For example, sub-CBDD-5 can be obtained by the logic OR operation between sub-CBDD-3 based on subtree 3 and sub-CBDD-4 based on subtree 4, as shown in Figure 21.

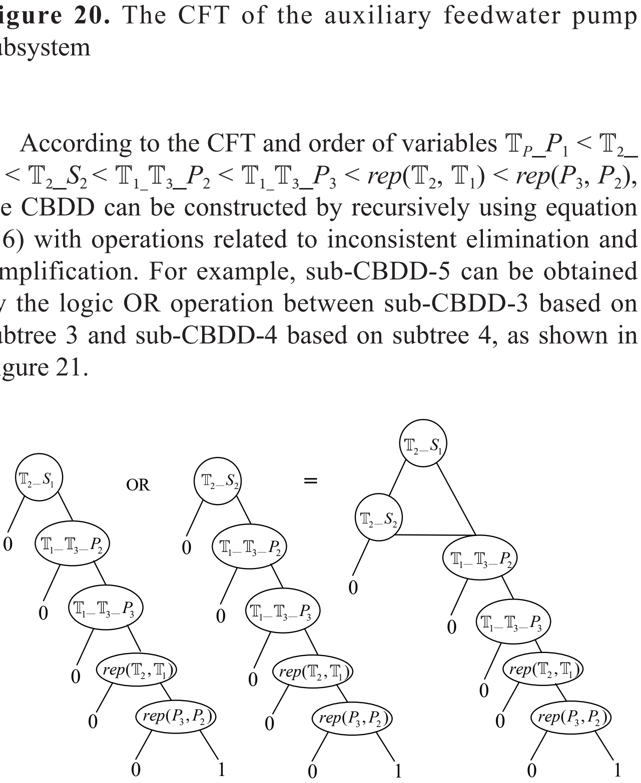


**Figure 21.** The logic OR operation between sub-CBDD-3 based on subtree 3 and sub-CBDD-4 based on subtree 4

Similarly, sub-CBDD-6 can be generated based on subtrees 1 and 2. The final CBDD is constructed by logic OR operation with sub-CBDD-5 and sub-CBDD-6, as shown in Figure 22.

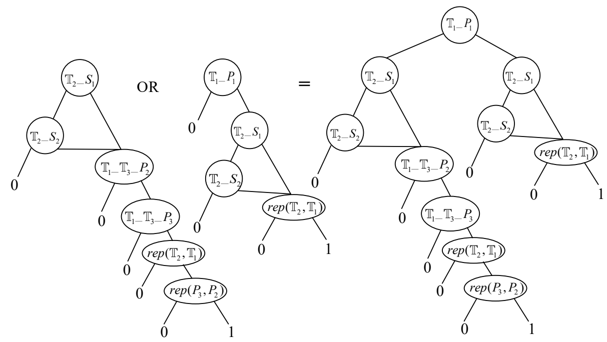


**Figure 22.** The final CBDD obtained by logic OR operation sub-CBDD-5 with sub-CBDD-6

According to the final CBDD, there are 4 paths from the top node to terminal node 1, as follows:

①$\mathbb{T}_1_P_1 \rightarrow \mathbb{T}_2_S_1 \rightarrow rep(\mathbb{T}_2, \mathbb{T}_1)$

②$\mathbb{T}_1_P_1 \rightarrow \neg\mathbb{T}_S_S_1 \rightarrow \mathbb{T}_S_S_2 \rightarrow rep(\mathbb{T}_2, \mathbb{T}_1)$

③$\neg\mathbb{T}_1_P_1 \rightarrow \mathbb{T}_2_S_1 \rightarrow \mathbb{T}_1_\mathbb{T}_3_P_2 \rightarrow \mathbb{T}_1_\mathbb{T}_3_P_3$
$\rightarrow rep(\mathbb{T}_2, \mathbb{T}_1) \rightarrow rep(P_3, P_2)$

④$\neg\mathbb{T}_1_P_1 \rightarrow \neg\mathbb{T}_2_S_1 \rightarrow \mathbb{T}_2_S_2 \rightarrow \mathbb{T}_1_\mathbb{T}_3_P_2$
$\rightarrow \mathbb{T}_1\_\mathbb{T}_3\_P_3 \rightarrow rep(\mathbb{T}_2, \mathbb{T}_1) \rightarrow rep(P_3, P_2)$

In path ②, the calculation of $\neg\mathbb{T}_S\_S_1$ probability also depends on the occurrence time of $\mathbb{T}_1\_P_1$. Similarly, in path ④, the calculation of $\neg\mathbb{T}_S\_S_1$ probability also depends on the occurrence time of $\mathbb{T}_3\_P_3 \cdot rep(P_3, P_2)$. Fox example, $\Pr\{\neg\mathbb{T}_2\_S_1 \cdot \mathbb{T}_1\_P_1\} \int_0^t f_{P_1}(\tau_{P_1})(1 - \int_0^{t-t_{P_1}} f_{S_1}(\tau_{S_1}) d_{\tau_{S_1}}) d_{\tau_{P_1}}$.

Set the time-to-failure distribution of all components to follow an exponential distribution such as $f_{P_1}(\tau_{P_1}) = \lambda_{P_1} e^{-\lambda_{P_1} t}$. The hazard rate of components is shown in Table 3.

**Table 3.** The hazard rate of components in [22]

| Component | Hazard rate λ (/hour) |
|---|---|
| $\theta_{P_1}$ | $3.48 \times 10^{-6}$ |
| $\theta_{P_2}$ | $1.4 \times 10^{-6}$ |
| $\theta_{P_3}$ | $1.03 \times 10^{-3}$ |
| $\theta_{S_1}$ | $3.19 \times 10^{-4}$ |
| $\theta_{S_2}$ | $2.1 \times 10^{-4}$ |

According to these paths, the failure probability of the auxiliary feedwater pump subsystem can be computed by equation (23).

$$\Pr\{TE\} = \int_0^t \int_0^{t-\tau_{P_1}} f_{P_1}(\tau_{p_1}) f_{S_1}(\tau_{S_1}) d_{\tau_{S_1}} d_{\tau_{P_1}}$$

$$
\begin{aligned}
&+\int_0^t \int_0^{t-\tau_{P_1}} f_{P_1}(\tau_{p_1}) f_{S_2}(\tau_{S_2})(1-\int_0^{t-\tau_{P_1}} f_{S_1}(\tau_{S_1}) d_{\tau_{S_1}}) d_{\tau_{S_2}} d_{\tau_{P_1}} \\
&\quad +(1-\int_0^t f_{P_1}(\tau_{P_1}) d_{\tau_{P_1}}) \\
&\cdot \int_0^t \int_0^{t-\tau_{P_2}} \int_0^{t-\tau_{P_2}-\tau_{P_3}} f_{P_2}(\tau_{P_2}) f_{P_3}(\tau_{P_3}) f_{S_1}(\tau_{S_1}) d_{\tau_{S_1}} d_{\tau_{P_3}} d_{\tau_{P_2}} \\
&+(1-\int_0^t f_{P_1}(\tau_{P_1}) d_{\tau_{P_1}}) \\
&\cdot \int_0^t \int_0^{t-\tau_{P_2}} \int_0^{t-\tau_{P_2}-\tau_{P_3}} f_{P_2}(\tau_{P_2}) f_{P_3}(\tau_{P_3}) f_{S_2}(\tau_{S_2}) \\
&\quad \cdot (1-\int_0^{t-\tau_{P_2}-\tau_{P_3}} f_{S_1}(\tau_{S_1}) d_{\tau_{S_1}}) d_{\tau_{S_2}} d_{\tau_{P_3}} d_{\tau_{P_2}}. \qquad \textbf{(23)}
\end{aligned}
$$

To verify the result computed by equation (23), the calculation formula based on the algebraic structure-function of the DFT is shown as follows:

$$
\begin{aligned}
TE &= ((P_1 + (P_2 \lhd P_3) \cdot P_3) \lhd S_1) \cdot ((P_1 + (P_2 \lhd P_3) \cdot P_3) \lhd S_2) \\
&\quad \cdot (S_1 + S_2) \\
\Pr\{TE\} &= \Pr\{(P_1 \lhd S_1) \cdot S_1 + (P_1 \lhd S_2) \cdot S_2 \\
&\quad + (P_2 \lhd P_3 \lhd S_1) \cdot S_1 + (P_2 \lhd P_3 \lhd S_2) \cdot S_2. \qquad \textbf{(24)}
\end{aligned}
$$

The subsystem failure probability computed by equation (23) with different mission times (*t* = 100, 300, 500, and 900 hours) is shown in Table 4. Also, the result computed by equation (24) combined with PIE-method is also in Table 4, which exactly matches the probability obtained by the CBDD-based method. Also, the same results as shown in Table 4 can be obtained by using the ABDD-based method, the final ABDD of the auxiliary feedwater pump subsystem is shown in Figure 24.

**Table 4.** The subsystem failure probability with different mission times (*t*)

| *t*(hours) | CBDD-based method | Algebraic structure-function-based method |
|---|---|---|
| | System failure probability | |
| 100 | $9.1656 \times 10^{-6}$ | $9.1656 \times 10^{-6}$ |
| 300 | $8.1659 \times 10^{-5}$ | $8.1659 \times 10^{-5}$ |
| 500 | $2.2413 \times 10^{-4}$ | $2.2413 \times 10^{-4}$ |
| 900 | $7.0602 \times 10^{-4}$ | $7.0602 \times 10^{-4}$ |

The algebraic structure-function-based method needs to calculate ($2^4$-1) multiple integral items after using the PIE while considering the impact of repeated and dependent events. Compared to the algebraic structure-function-based method, the CBDD-based method only needs to calculate 4 multiple integral items. The time-to-failure probability distribution of the auxiliary feedwater pump subsystem within 1000 hours is shown in Figure 23.

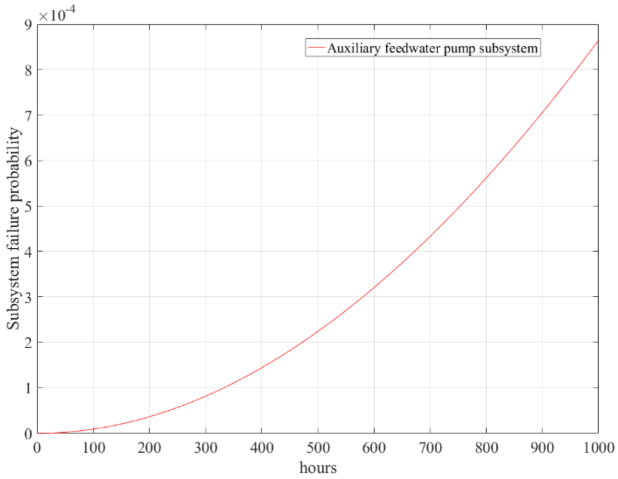


**Figure 23.** The time-to-failure probability distribution of subsystem within 1000 hours

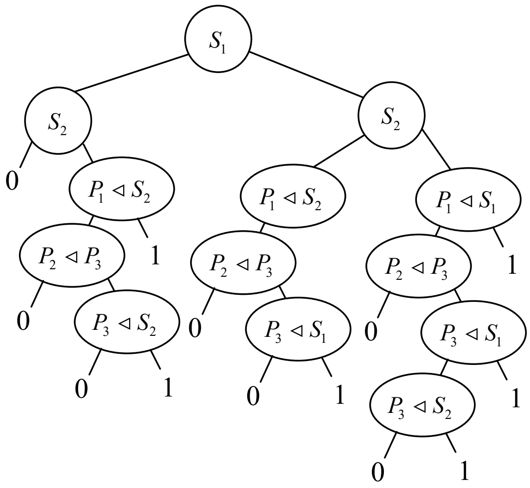


**Figure 24.** The final ABDD of the auxiliary feedwater pump subsystem (the variable ordering is $S_1 < S_2 < P_1 \lhd S_1 < P_1 \lhd S_2 < P_2 \lhd P_3 < P_3 \lhd S_1 < P_3 \lhd S_2$ )

## 6 Conclusion

To analyze the reliability of the cold-standby system with subsystems, this paper demonstrates the reliability analysis of the CPS gate with subtree inputs by a formulae based on the improved *rep* conditioning event. The improved *rep* conditioning can describe the replacement behavior between subtrees in the CSP gate. Also, new operation rules used for the improved conditioning event are proposed. Moreover, the corresponding inconsistent elimination and simplification rules used for the CBDD are presented. Based on the case study, compared to the algebraic structure-function-based method, the CBDD-based method has lower computational complexity since it can directly generate the SDPs.

In future work, we will consider the reliability analysis of the WSP gate with subtree inputs by using *rep* event while considering more complex cases.

## Acknowledgements

This work was partially supported by Guangdong Intelligence Platform of Prawn Modern Seed Industry (Grant No. 2022GCZX001), the Fund of Southern Marine Science and Engineering Guangdong Laboratory (Zhanjiang) (Grant No. ZJW-2019-06), the Guangdong Provincial Key Laboratory of Cyber-Physical Systems (Grant No. 2020B1212060069), the National&Local Joint Engineering Research Center of Intelligent Manufacturing Cyber-Physical Systems (Grant No. 2020B1212060069), and program for scientific research start-up funds of Guangdong Ocean University.

## References

[1] B. Johnson, *Design and analysis of fault-tolerant digital systems*, Massachusetts: Addison-Wesley Publishing Company, 1989.

[2] J. B. Dugan, S. J. Bavuso, M. A. Boyd, Dynamic fault-tree models for fault-tolerant computer systems, *IEEE Transactions on Reliability*, Vol. 41, No. 3, pp. 363-377, September, 1992.

[3] S. Zhou, J. Xiang, W. E. Wong, Reliability analysis of dynamic fault trees with spare gates using conditional binary decision diagrams, *Journal of Systems and Software*, Vol. 170, Article No. 110766, December, 2020.

[4] G. Merle, J.-M. Roussel, J.-J. Lesage, Algebraic determination of the structure function of dynamic fault trees, *Reliability Engineering & System Safety*, Vol. 96, No. 2, pp. 267-277, February, 2011.

[5] C. Wang, L. Xing, S. V. Amari, A fast approximation method for reliability analysis of cold-standby systems, *Reliability Engineering & System Safety*, Vol. 106, pp. 119-126, October, 2012.

[6] H. Jia, G. Levitin, Y. Ding, Y. Song, Reliability analysis of standby systems with multi-state elements subject to constant transition rates, *Quality and Reliability Engineering International*, Vol. 35, No. 1, pp. 318-328, February, 2019.

[7] P. Zhu, J. Han, L. Liu, F. Lombardi, A stochastic approach for the analysis of dynamic fault trees with spare gates under probabilistic common cause failures, *IEEE Transactions on Reliability*, Vol. 64, No. 3, pp. 878-892, September, 2015.

[8] X. Li, Y.-F. Li, H. Li, H.-Z. Huang, An algorithm of discrete-time bayesian network for reliability analysis of multilevel system with warm spare gate, *Quality and Reliability Engineering International*, Vol. 37, No. 3, pp. 1116-1134, April, 2021.

[9] Z. Behboudi, G. M. Borzadaran, M. Asadi, Reliability modeling of two-unit cold standby systems: a periodic switching approach, *Applied Mathematical Modelling*, Vol. 92, pp. 176-195, April, 2021.

[10] S. Zhou, L. Ye, S. Xiong, J. Xiang, Reliability analysis of dynamic fault trees with priority-and gates based on irrelevance coverage model, *Reliability Engineering & System Safety*, Vol. 224, Article No. 108553, August, 2022.

[11] J. Newton, D. Verna, A theoretical and numerical analysis of the worst-case size of reduced ordered binary decision diagrams, *ACM Transactions on Computational Logic*, Vol. 20, No. 1, pp. 1-36, January, 2019.

[12] J. Kawahara, K. Sonoda, T. Inoue, S. Kasahara, Efficient construction of binary decision diagrams for network reliability with imperfect vertices, *Reliability Engineering & System Safety*, Vol. 188, pp. 142-154, August, 2019.

[13] M. Mrena, M. Kvassay, R. S. Stankovic, Software Library for Teaching Applications of Binary Decision Diagrams in Reliability Analysis, *2020 18th International Conference on Emerging eLearning Technologies and Applications (ICETA)*, Košice, Slovenia, 2020, pp. 487-497.

[14] R. Banov, Z. Šimić, D. Grgić, A new heuristics for the event ordering in binary decision diagram applied in fault tree analysis, *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, Vol. 234, No. 2, pp. 397-406, April, 2020.

[15] J. Lee, Y H. Ye, X. Huang, R. Yang, Binary-decision-diagram-based decomposition of Boolean functions into reversible logic elements, *Theoretical Computer Science*, Vol. 814, pp. 120-134, April, 2020

[16] A. S. Dimovski, A binary decision diagram lifted domain for analyzing program families, *Journal of Computer Languages*, Vol. 63, Article No. 101032, April, 2021.

[17] A. Chakraborty, V. Maurya, S. Prasad, S. Gupta, R. Chakraborty, H. Rahaman, Binary decision diagram-based synthesis technique for improved mapping of Boolean functions inside memristive crossbar-slices, *IET Computers & Digital Techniques*, Vol. 15, No. 2, pp. 112-124, March, 2021.

[18] A. Awad, A. Hawash, B. Abdalhaq, A Genetic Algorithm (GA) and Swarm Based Binary Decision Diagram (BDD) Reordering Optimizer Reinforced with Recent Operators, *IEEE Transactions on Evolutionary Computation*, Vol. 27, No. 3, pp. 535-549, June, 2023.

[19] L. Xing, O. Tannous, J. B. Dugan, Reliability analysis of nonrepairable cold-standby systems using sequential binary decision diagrams, *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, Vol. 42, No. 3, pp. 715-726, May, 2012.

[20] O. Tannous, L. Xing, J. B. Dugan, Reliability analysis of warm standby systems using sequential bdd, 2011 *Proceedings-Annual Reliability and Maintainability Symposium*, Lake Buena Vista, FL, USA, 2011, pp. 1-7.

[21] H. Yu, X. Wu, A method for transformation from dynamic fault tree to binary decision diagram, *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, Vol. 235, No. 3, pp. 416-430, June, 2021.

[22] W. Jiang, S. Zhou, L. Ye, D. Zhao, J. Tian, W. E. Wong, J. Xiang, An algebraic binary decision diagram for analysis of dynamic fault tree, *2018 5th International Conference on Dependable Systems and Their Applications (DSA)*, Dalian, China, 2018, pp. 44-51.

[23] D. Guo, J. Wang, J. Lin, B. Zhang, N. Yong, D. Xia, D. Ge, An adapted component-connection method for building SBDD encoding a dynamic fault tree, *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, Article No. 1748006X221117929, August, 2022.

[24] B. Abdalhaq, A. Awad, A. Hawash, A fast Binary Decision Diagram (BDD)-based reversible logic optimization engine driven by recent meta-heuristic reordering algorithms, *Microelectronics Reliability*, Vol. 123, Article No. 114168, August, 2021.

[25] A. Alkaff, State space and binary decision diagram models for discrete standby systems with multistate components, *Applied Mathematical Modelling*, Vol. 110, pp. 298-319, October, 2022.

[26] P. Liu, S. Zhou, L. Ye, D. Zhao, J. Xiang, A combinatorial reliability analysis of dynamic fault trees with priority-and gates, *2021 IEEE International Symposium on Software Reliability Engineering Workshops*, Wuhan, China, 2021, pp. 182-188.

[27] M. Stamatelatos, W. Vesely, J. Dugan, J. Fragola, J. Minarick III, J. Railsback, Fault tree handbook with aerospace applications, *NASA Office of Safety and Mission Assurance*, August, 2002.

[28] J. Tian, *Research on prediction of residual use life for attitude control system of satellite based on dynamic fault tree*, Master thesis, Nanjing University of Aeronautics and Astronautics, Nanjing, China, 2016.

[29] K. Zhang, H. Yang, F. Qian, Reliability analysis of satellite power system based on dynamic fault tree, *Computer and Digital Engineering*, Vol. 44, No. 3, pp. 400-404, March, 2016.

[30] R. Liu, X. Ye, Y. Liu, S. Zhou, Reliability analysis of auxiliary water supply system based on dynamic fault tree, *Application of Electronic Technique*, Vol. 47, No. S1, pp. 177-184, November, 2021.

# Appendix

$A$, $B$, and $C$ are basic events of the DFT. Some parts of theorems used for the “$\lhd$” operation symbol in [4] are as follows:

$$A \lhd (B+C) = (A \lhd B)\cdot(A \lhd C)$$
$$A \lhd (B\cdot C) = (A \lhd B)+(A \lhd C)$$
$$(A+B) \lhd C = (A \lhd C)+(B \lhd C)$$
$$(A\cdot B) \lhd C = (A \lhd C)\cdot(B \lhd C)$$
$$(A \lhd B) \lhd C = (A \lhd B)\cdot(A \lhd C)$$
$$A+(A \lhd B) = A$$
$$(A \lhd B)+B = A+B$$

# Biographies

**Siwei Zhou** received his Ph.D. degree in Computer Science and Technology from Wuhan University of Technology. He is currently a teacher with the Faculty of Computer Science and Technology, Guangdong Ocean University. His research is focused on system reliability engineering and dependable computing.

**Yinghuai Yu** received the Master’s degree in Computer Application Technology from Guizhou University. He is an associate professor in Computer Science and Technology, Guangdong Ocean University. His research focuses on image processing, computer vision, and geographic information system.

**Xiaohong Peng** received Master’s degree in Computer Science and Technology from South China University of Technology. She is a professor in Computer Science and Technology, Guangdong Ocean University. Her research focuses on intelligent control and intelligent system, intelligent computing and application, and intelligent underwater robot.