Nuclear Engineering and Technology 54 (2022) 110–116

Contents lists available at ScienceDirect

Nuclear Engineering and Technology

journal homepage: www.elsevier.com/locate/net

Original article

How to incorporate human failure event recovery into minimal cut set generation stage for efficient probabilistic safety assessments of nuclear power plants

Woo Sik Jung [a, *], Seong Kyu Park [a], John E. Weglian [b], Jeff Riley [c]

[a] Sejong University, 209 Neungdong-ro, Gwangjin-gu, Seoul, 05006, South Korea
[b] Electric Power Research Institute, 1300 W. WT Harris Blvd., Charlotte, NC, 28262, USA
[c] Electric Power Research Institute, 3420 Hillview Ave, Palo Alto, CA, 94304, USA

ARTICLE INFO

Article history:
Received 5 November 2020
Received in revised form
22 March 2021
Accepted 23 April 2021
Available online 5 May 2021

Keywords:
Probabilistic safety assessment (PSA)
Human reliability analysis (HRA)
Human failure event (HFE)
Human error probability (HEP)
Dependent HEP

ABSTRACT

Human failure event (HFE) dependency analysis is a part of human reliability analysis (HRA). For efficient HFE dependency analysis, a maximum number of minimal cut sets (MCSs) that have HFE combinations are generated from the fault trees for the probabilistic safety assessment (PSA) of nuclear power plants (NPPs). After collecting potential HFE combinations, dependency levels of subsequent HFEs on the preceding HFEs in each MCS are analyzed and assigned as conditional probabilities. Then, HFE recovery is performed to reflect these conditional probabilities in MCSs by modifying MCSs.

Inappropriate HFE dependency analysis and HFE recovery might lead to an inaccurate core damage frequency (CDF). Using the above process, HFE recovery is performed on MCSs that are generated with a non-zero truncation limit, where many MCSs that have HFE combinations are truncated. As a result, the resultant CDF might be underestimated.

In this paper, a new method is suggested to incorporate HFE recovery into the MCS generation stage. Compared to the current approach with a separate HFE recovery after MCS generation, this new method can (1) reduce the total time and burden for MCS generation and HFE recovery, (2) prevent the truncation of MCSs that have dependent HFEs, and (3) avoid CDF underestimation. This new method is a simple but very effective means of performing MCS generation and HFE recovery simultaneously and improving CDF accuracy. The effectiveness and strength of the new method are clearly demonstrated and discussed with fault trees and HFE combinations that have joint probabilities.

1. Introduction

1.1. PSA quantification procedure

Probabilistic safety assessment (PSA) is performed to calculate various risks of nuclear power plants (NPPs). The PSA quantification procedure is depicted in Fig. 1. First, minimal cut sets (MCSs) are generated by solving a fault tree [1,2]. Second, MCS recovery is performed to delete nonsense MCSs that have impossible failure combinations and to perform human failure event (HFE) recovery [3,4]. Third, the core damage frequency (CDF) is calculated by the min-cut-upper-bound (MCUB) from recovered MCSs [5]. Alternatively, an accurate CDF for seismic PSA is calculated by converting MCSs into a binary decision diagram (BDD) [5].

1.2. HFE dependency analysis

HFE dependency analysis is a part of human reliability analysis (HRA), with the objective of determining the joint probabilities of HFEs. Dependency levels of subsequent HFEs on the preceding HFEs in each MCS are analyzed, and dependent human error probabilities (HEPs) are determined for HFE recovery. This HFE dependency analysis procedure is depicted in Fig. 2. HFE dependency analysis consists of four activities: (1) collect HFE combinations, (2) analyze dependent HFEs to determine dependency levels between subsequent and preceding HFEs, (3) regenerate MCSs, and (4) perform HFE recovery. In this paper, HFE recovery is defined as MCS post-processing for reflecting the dependent probabilities of HFEs in

* Corresponding author.
E-mail addresses: woosjung@sejong.ac.kr (W.S. Jung), sparkpsa@gmail.com (S.K. Park), jweglian@epri.com (J.E. Weglian), jriley@epri.com (J. Riley).

https://doi.org/10.1016/j.net.2021.04.026

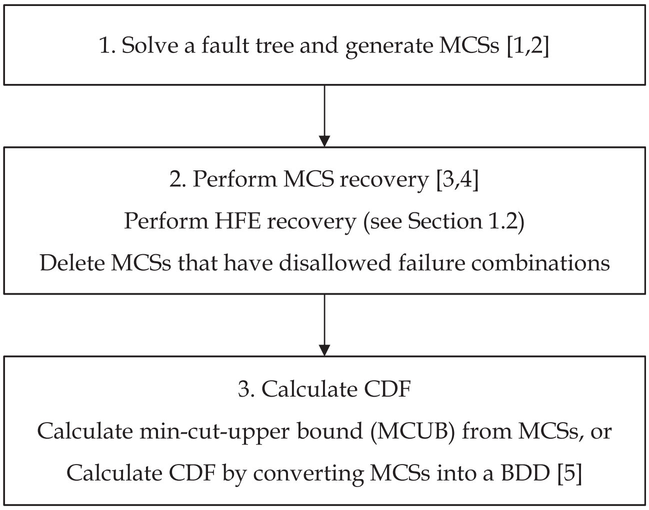

Fig. 1. PSA quantification procedure.

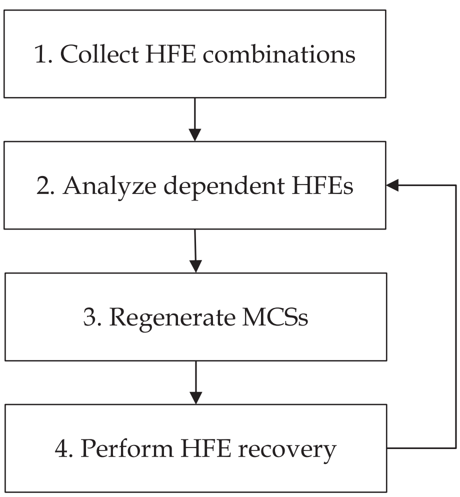

Fig. 2. HFE dependency analysis.

MCS probabilities.

MCS is a minimal combination of initiating events, component failures, and HFEs that leads to core damage of NPPs. HFEs in a single MCS can be arranged in chronological order according to the corresponding accident sequence. HFEs usually have a positive dependency on their preceding HFEs. It is well known that CDF can be underestimated if HFE dependency is ignored. On the other hand, CDF is overestimated if complete HFE dependency is assumed.

The accurate joint probability assignment of dependent HFEs is a main concern of HFE dependency analysis. An illustration of the joint probability of HFEs is given in Eq. (1).

$$p(H1H2) = p(H1)p(H2|H1). \quad (1)$$

Here, H1 and H2 are HFEs in chronological order. H1 is a preceding HFE, and H2 is a subsequent HFE. The subsequent HFE has positive dependency on the preceding HFE, as illustrated in Eq. (2).

$$P(H2|H1) \geq P(H2). \quad (2)$$

As shown in Eq. (3), if the nominal probabilities of p (H1) and p (H2) are used, many MCSs that have HFE combinations are truncated during the MCS generation stage. In order to avoid this truncation of valid MCSs, the nominal probabilities of p (H1) and p (H2) are intentionally elevated to a certain value (often called "seed value") at the MCS generation stage, and the correct p(H1H2) is reflected in the MCS probability during the HFE recovery stage. Here, p(X) and p(Y) are probabilities of regular basic events X and Y for component failures.

$$p(X)p(Y)p(H1)p(H2) < \text{MCS truncation limit} < p(X)p(Y)p(H1H2). \quad (3)$$

1.3. Difficulties in HFE dependency analysis

The difficulties of HFE dependency analysis in Fig. 2 are summarized as follows:

1. Difficulty in collecting HFE combinations: The combinations of HFEs may be collected by assigning very high HEPs to all HFEs (such as 0.9 or 1.0), lowering the truncation limit as much as possible, or using a combination of the two. In each of these cases, it takes a very long time to solve the fault tree and generate the MCSs. At the initial stage, HFE probabilities are intentionally elevated to a high value, such as 0.9 or 1.0.
2. Difficulty in analyzing dependent HFEs: The number of HFE combinations in calculated MCSs sometimes exceeds 10,000, and the number of HFEs in a single MCS ranges from 1 to 10 in a typical PSA. Since the number of HFE combinations is so large, it is also a very complex task to analyze the dependencies of the subsequent HFEs on the preceding HFEs.
3. Difficulty in regenerating MCSs: After assigning dependency levels to the dependent HFEs, MCSs are recalculated with elevated HEPs to avoid truncating MCSs that have HFEs that would remain above truncation after the application of the dependency rules.
4. Difficulty in performing HFE recovery: HFE recovery is repeatedly performed whenever MCSs are recalculated. HFE recovery frequently takes longer than calculating the MCSs.

1.4. Objectives of this study

Generally, HFE recovery rules account for 50–90% of MCS recovery rules (see Table 1). In many cases, MCS recovery has so many recovery rules that it takes much longer than MCS generation. There is a significant opportunity for improvement in the quantification speed, if the HFE recoveries can be moved into the MCS generation stage. Thus, there is a great need to incorporate HFE recovery into the MCS generation stage. However, there has been no breakthrough toward accomplishing this task, since the creation of every HFE combination in MCSs with new basic events inside the fault tree unimaginably increases the size and complexity of the fault tree. This paper introduces the first logical method to incorporate HFE recovery into the MCS generation stage.

In order to minimize the third and fourth difficulties in Section 1.3, the first objective of this study is to incorporate HFE recovery into the MCS generation stage. The second objective is to generate MCSs without elevating HEPs and/or lowering the MCS truncation limit. The third objective is to calculate a more accurate CDF by accomplishing the first and second objectives.

Table 1
Various recovery rules for HFE recovery.

PSA tools	HFE recovery rule to replace H1H2H3 with H123

SAPHIRE [4]	if H1 *H2 *H3 then
	CopyRoot;
	DeleteEvent = H1;
	DeleteEvent = H2;
	DeleteEvent = H3;
	AddEvent = H123;
	endif

PHOENIX/QRECOVER [11]	**REPLACE EVENTS**H123 0.2 T H1 H2 H3
AIMS [12]	Main fault tree has p (H123) H1 H2 H3/H123
SAREX [13]	H123 = 0.2 H1 H2 H3 = H123

2. Typical HFE dependency analysis

2.1. Analyze dependent HFEs

The goal of HFE dependency analysis is to determine the level of dependency in each HFE combination. The level of dependency can be determined using a very sophisticated process considering various human factors and performance shaping factors for HFEs [6].

In a typical method [6], one of five dependency levels is assigned to the subsequent HFE that is dependent on the preceding HFE: zero dependency (ZD), low dependency (LD), moderate dependency (MD), high dependency (HD), and complete dependency (CD). After determining the dependency level, the dependent HEP of the subsequent HFE is calculated by Eq. (4) [6–10].

$$\begin{aligned} P_{ZD} &= P_0 \\ P_{LD} &= (1 + 19 \times P_0)/20 \\ P_{MD} &= (1 + 6 \times P_0)/7 \\ P_{HD} &= (1 + P_0)/2 \\ P_{CD} &= 1.0. \end{aligned} \tag{4}$$

Here, P_0 is a nominal HEP, and the probabilities on the left-hand side in Eq. (4) denote dependent HEPs according to the dependency level from ZD to CD.

MCSs that have HFEs are imported into a dedicated tool, such as the EPRI HRA Calculator [9]. This tool arranges the HFE combinations in MCSs in chronological order and assists the user in assigning dependency levels to the subsequent HFEs. Generally, dependency between chronological HFEs is analyzed with a HFE dependency decision tree, as shown in Fig. 3. Using Eq. (4) and the HFE dependency decision tree in Fig. 3, a conditional HEP, such as p(H2|H1), is assigned, and p(H1H2) is calculated by p(H1)p(H1|H2).

Then, in the current approach, HFE recovery rules are written for the PSA tools. PSA tools have their own specific recovery rules or grammar for MCS recovery. Please note that HFE recovery is a major part of MCS recovery. The HFE recovery rule formats of some PSA tools are listed in Table 1. For one of the current commercial nuclear power plants, PSA has more than 30,000 MCS recovery rules. Because of this huge number of MCS recovery rules, MCS recovery frequently takes longer than MCS generation.

2.2. Perform HFE recovery

Once the dependency levels among HFEs are determined, it is necessary to replace dependent HFEs in a single MCS with new HFEs that have dependent HEPs or replace the HFEs with a new HFE that has joint probability of HFE combination. This is generally performed by a specific tool [3].

As shown in Eqs. (5) and (6), the first step of HFE recovery is to replace dependent HFEs (H2 and H3) with new HFEs (H2′ and H3’) that have conditional probabilities in Eq. (7) or to replace the whole HFE combination (H1H2H3) with a single HFE (H123) that has the product of conditional probabilities in Eq. (8).

$$H1H2H3 \rightarrow H1H2'H3' \tag{5}$$

$$H1H2H3 \rightarrow H123, \tag{6}$$

where

$$p(H2') = p(H2|H1) \text{ and } p(H3') = p(H3|H1H2) \approx p(H3|H2) \tag{7}$$

$$\begin{aligned} p(H123) &= p(H1)p(H2|H1)p(H3|H1H2) \\ &\approx p(H1)p(H2|H1)p(H3|H2). \end{aligned} \tag{8}$$

In order to avoid the underestimation of CDF, an unanalyzed HFE combinations are treated conservatively. In the case where some of the HFEs match a combination (H1H2H3), the probabilities of any HFEs that are not part of the combination (H4H5) are set to 1.0.

$$H1H2H3H4H5 \rightarrow H123*H4H5 \tag{9}$$

$$p(H4) = p(H5) = 1. \tag{10}$$

If no combination is matched, the first HFE has its nominal HEP, and the others are set to 1.0. This example is shown in Eq. (11).

$$H1H4H5 \text{ where } p(H1) < 1 \text{ and } p(H4) = p(H5) = 1. \tag{11}$$

3. New HFE quantification method

The new HFE dependency analysis process suggested in this paper is focused on (1) collecting a maximum number of HFE combinations without lowering the MCS truncation limit and (2) performing MCS generation and HFE post-processing simultaneously. Fig. 4 describes the details of the new method proposed in this study. For a clear explanation of this new method, illustrations are provided in Sections 4.1 and 4.2.

Fault tree analysis in PSA has two stages. First, MCSs are generated from the fault tree. Second, MCS post-processing is performed for reflecting joint HEPs in MCSs. The problem is that many MCSs that have combinations of HFEs are truncated at the first stage even with a low truncation limit. The procedure in Fig. 4 minimizes the truncation of MCSs that have combinations of HFEs.

The procedure in Fig. 4 can be summarized as follows. In Step 1, HFE combinations and their joint HEPs that were generated by HFE dependency analysis are inputs to the new procedure. In Step 2, each HFE combination is defined as a new HFE combination event that has a joint HEP. In Step 3, each HFE event in the fault tree is replaced with Boolean OR combination of HFE combination events. In Step 4, the fault tree that has regular basic events and HFE combination events are solved and MCSs are generated. Detailed illustrations and applications are provided in Section 4.

4. Application of new method

4.1. Application to simple fault tree A

The new method explained in Section 3 has been implemented in Fault Tree Reliability Evaluation eXpert (FTREX) [1–3]. FTREX generates a new fault tree f (**X**, **C**) by combining a given fault tree f (**X**,**H**) and HFE combinations and generates MCSs without

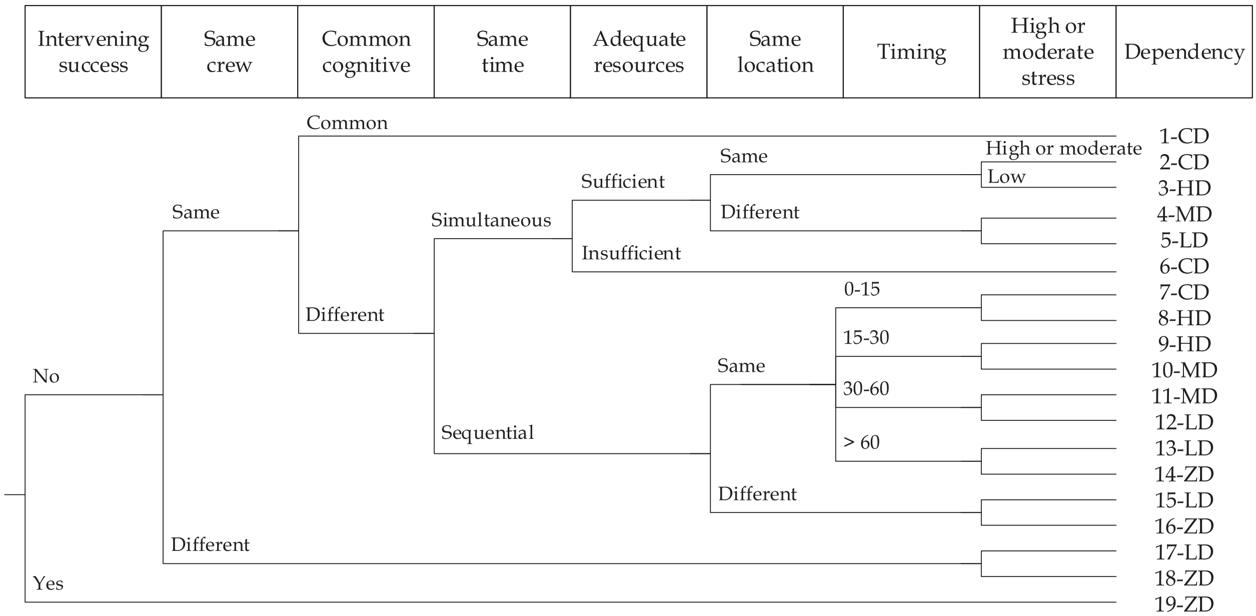

Fig. 3. Dependency decision tree [9].

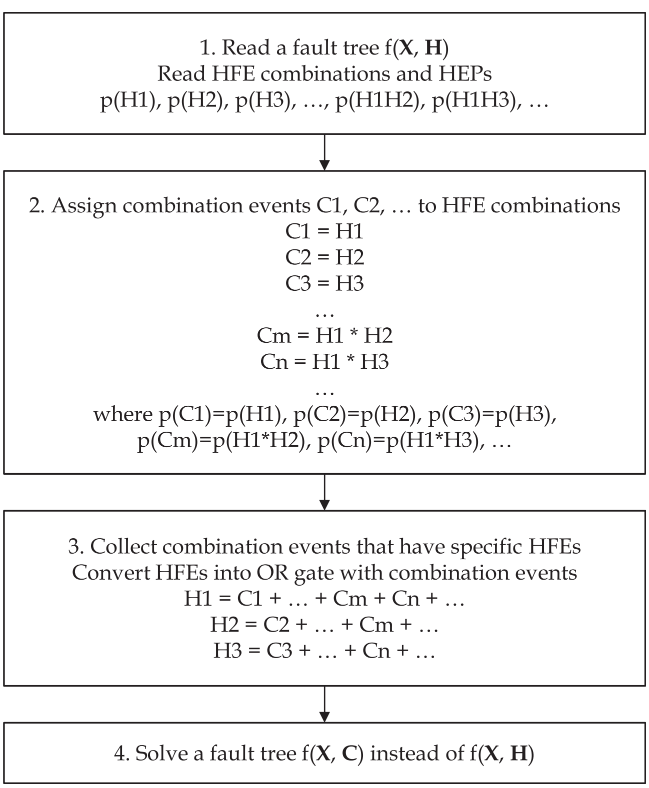

Fig. 4. New method to incorporate HFE recovery into MCS generation stage.

employing HFE recovery using the new method in Section 3. The new method with a fault tree is explained with an example in Eq. (12):

$$CD = G1 + G2 + G3 \tag{12a}$$

$$G1 = A * (B + H1) \tag{12b}$$

$$G2 = B * (C + H2) \tag{12c}$$

$$G3 = C * D * H1 * H2. \tag{12d}$$

(Step 1) The results of HFE dependency analysis such as joint HEPs such as p (H1 *H2) in Eq. (13) are inputs to this procedure. FTREX reads HFE combinations and their probabilities in Eq. (13). Please note that the joint probability p (H1H2) is much higher than p (H1) *p (H2).

$$p(H1) = 0.2 \tag{13a}$$

$$p(H2) = 0.3 \tag{13b}$$

$$p(H1 * H2) = 0.1 \tag{13c}$$

(Step 2) FTREX assigns combination events C1–C3 to the HFE combinations in Eq. (14).

$$C1 = H1 \text{ and } p(C1) = 0.2 \tag{14a}$$

$$C2 = H2 \text{ and } p(C2) = 0.3 \tag{14b}$$

$$C3 = H1 * H2 \text{ and } p(C3) = 0.1 \tag{14c}$$

(Step 3) A special mapping between combination events C1–C3 and HFE combinations is drawn in Eq. (15). H1 is in combination events C1 and C3, and H2 is in combination events C2 and C3. Using this mapping information, FTREX converts H1 and H2 events into logical OR gates in Eq. (15).

$$H1 = C1 + C3 \tag{15a}$$

$$H2 = C2 + C3. \quad (15b)$$

(Step 4) FTREX combines the given fault tree in Eq. (12) and the mapping information in Eq. (15) and solves the new fault tree in Eq. (16). Please note that H1 and H2 are not events but logical OR gates that have combination events C1–C3.

$$CD = G1 + G2 + G3 \quad (16a)$$

$$G1 = A * (B + H1) \quad (16b)$$

$$G2 = B * (C + H2) \quad (16c)$$

$$G3 = C * D * H1 * H2 \quad (16d)$$

$$H1 = C1 + C3 \quad (16e)$$

$$H2 = C2 + C3, \quad (16f)$$

where

$$P(C1) = 0.2 \quad (17a)$$

$$P(C2) = 0.3 \quad (17b)$$

$$P(C3) = 0.1. \quad (17c)$$

The MCSs that are calculated from the fault tree in Eq. (16) are in Eq. (18). It should be noted that the dependency between H1 and H2 is inherently reflected in Eq. (18) using the combination event C3 by assigning p (C3) = p (H1*H2) in Eq. (14). The joint HEP of p (H1*H2) is an input to this procedure. Without this new method, MCSs that have H1*H2 might be truncated with a given truncation limit. However, since p(C3) is larger than p(H1)*p(H2), MCSs that have C3 are not truncated with the same truncation limit. Therefore, there is no need to elevate H1 and H2 probabilities in the new method. This gives a great advantage by saving calculation time for generating MCSs and for performing HFE recovery.

$$CD = A * B + B * C + A * (C1 + C3) + B * (C2 + C3) + C * D * (C3 + C1 * C2). \quad (18)$$

Many HFE combinations, such as H1H2, are truncated in a usual PSA with a given truncation limit. However, as shown in Eq. (18), all the intended HFE combinations {H1, H2, H1H2} are produced using combination events C1, C2, and C3.

Multiple combination events in each MCS, such as C1 *C2 in Eq. (19), can be optionally created or deleted during MCS generation by the dedicated PSA tools. If a PSA analyst has confidence that all HFE combinations are found and their joint probabilities are properly assigned with a target truncation limit, there is no need to create multiple combination events. On the other hand, these multiple combination events can be optionally created to check if there are missing HFE combinations in Eq. (13). It is one of the main strengths of the new method.

$$CD \quad (19a)$$

$$= A * B + B * C + A * (C1 + C3) + B * (C2 + C3) + C * D * (C3 + C1 * C2) \quad (19b)$$

$$= A * B + B * C + A * (H1 + H1 * H2) + B * (H2 + H1 * H2) + C * D * H1 * H2 \quad (19c)$$

$$= A * B + B * C + A * H1 + B * H2 + C * D * H1 * H2. \quad (19d)$$

4.2. Application to simple fault tree B

FTREX combines a given fault tree and HFE combinations into a new fault tree and generates MCSs without truncating HFE combinations. This is explained with another fault tree in Eq. (20).

$$CD = \%I * G1 * G2 * G3 \quad (20a)$$

$$G1 = A + H1 \quad (20b)$$

$$G2 = B + H2 \quad (20c)$$

$$G3 = (C + H3)*(D + H4). \quad (20d)$$

(Step 1) The results of HFE dependency analysis such as joint HEPs are inputs to this procedure. FTREX reads HFE probabilities and combinations in the first column of Table 2.

(Step 2) FTREX assigns combination events C1–C15 to the HFE combinations in the second column of Table 2.

(Step 3) Since H1 is in combination events C1, C5, C6, C7, C11, C12, C13, and C15, FTREX converts H1 into logical OR gates, as in Eq. (21). Similarly, H2, H3, and H4 are converted into logical OR gates in Eq. (21).

$$H1 = C1 + C5 + C6 + C7 + C11 + C12 + C13 + C15 \quad (21a)$$

$$H2 = C2 + C5 + C8 + C9 + C11 + C12 + C14 + C15 \quad (21b)$$

$$H3 = C3 + C6 + C8 + C10 + C11 + C13 + C14 + C15 \quad (21c)$$

$$H4 = C4 + C7 + C9 + C10 + C12 + C13 + C14 + C15. \quad (21d)$$

(Step 4) FTREX solves a fault tree in Eq. (22) that is a combination of Eqs. (20) and (21). Please note that H1–H4 are not events but logical OR gates in Eq. (22) that have combination events C1–C15.

$$CD = \%I * G1 * G2 * G3 \quad (22a)$$

$$G1 = A + H1 \quad (22b)$$

$$G2 = B + H2 \quad (22c)$$

$$G3 = (C + H3)*(D + H4) \quad (22d)$$

$$H1 = C1 + C5 + C6 + C7 + C11 + C12 + C13 + C15 \quad (22e)$$

$$H2 = C2 + C5 + C8 + C9 + C11 + C12 + C14 + C15 \quad (22f)$$

$$H3 = C3 + C6 + C8 + C10 + C11 + C13 + C14 + C15 \quad (22g)$$

$$H4 = C4 + C7 + C9 + C10 + C12 + C13 + C14 + C15 \quad (22h)$$

The MCSs of the fault tree in Eq. (20) are listed in the first column of Table 3. The MCSs of the fault tree in Eq. (22) are listed in the second column of Table 3. The first 16 MCSs in the first and second columns in Table 3 show that the new method properly generates

Table 2
HFE combinations p (%I) = p(H1) = p(H2) = p(H3) = p(H4) = 0.1.

HFE combinations	HFE combinations mapped to combination events C1–C15
0.1H1	C1 0.1H1
0.1H2	C2 0.1H2
0.1H3	C3 0.1H3
0.1H4	C4 0.1H4
0.05H1 H2	C5 0.05H1 H2
0.05H1 H3	C6 0.05H1 H3
0.05H1 H4	C7 0.05H1 H4
0.05H2 H3	C8 0.05H2 H3
0.05H2 H4	C9 0.05H2 H4
0.05H3 H4	C10 0.05H3 H4
0.025H1 H2 H3	C11 0.025H1 H2 H3
0.025H1 H2 H4	C12 0.025H1 H2 H4
0.025H1 H3 H4	C13 0.025H1 H3 H4
0.025H2 H3 H4	C14 0.025H2 H3 H4
0.0125H1 H2 H3 H4	C15 0.0125H1 H2 H3 H4

(a) p(H1 H2 H3 H4) = 0.0125.
(b) C15 = H1 H2 H3 H4 and p(C15) = 0.0125.

MCSs and accurately performs HFE recovery simultaneously. It should be noted that all the MCSs in the first column are truncated if the truncation limit is larger than 1.0E-5. However, in the new method, none of the MCSs that have HFEs is truncated with the truncation limit 1.0E-5. This is a great strength of the new method.

As mentioned in Section 4.1, multiple combination events in each MCS can be optionally created or deleted during MCS generation by the dedicated PSA tools. If a PSA analyst has confidence that all HFE combinations are found and their joint probabilities are properly assigned with a target truncation limit, there is no need to create multiple combination events.

The additional MCSs in Table 4 are optionally created by allowing multiple combination events in each MCS. In many cases, missing HFE combinations could be obtained from the additional MCSs. This is an unexpected positive byproduct of the new method. After finding missing HFE combinations, the MCSs in Table 4 are subsumed and deleted by the MCSs in the second column of Table 3.

If the last HFE combination of {H1 *H2 *H3 *H4} is missing when collecting HFE combinations, this missing HFE combination can be recovered from the 17th MCS in Table 4, since {C1 *C14} is equal to {H1 *H2 *H3 *H4}.

4.3. Application to PSA fault tree

The new method is compared with current calculation methods with a fault tree and HFE combinations in Table 5, which are from a real PSA. The calculation results are summarized in Table 6. These comparison results are summarized as follows. They show that the new method is the most economical way to calculate an accurate CDF.

1. (Case 1) MCSs are generated using the nominal HEP at the normal truncation. Since nominal HEPs are used, many MCSs that have HFE combinations are truncated. A significantly underestimated CDF will be calculated after HFE recovery.
2. (Case 2) MCSs are generated using the HEP of Max(HEP, 0.1) and then HFE recovery is performed. Many MCSs that have 2 to 5 HFEs are calculated. However, there is no confidence that the resultant MCSs and HFE combinations will be identical to those

Table 4
Additional MCSs that have multiple combination events.

No.	Additional MCSs from fault tree f (X,C) in Eq. (22)
17	2.50E-04 %I C1 C14
18	2.50E-04 %I C10C5
19	2.50E-04 %I C11C4
20	2.50E-04 %I C12C3
…	…
84	1.25E-05 %I C5 C8 C9
85	1.00E-05 %I A B C3 C4
86	1.00E-05 %I A C C2 C4
…	…
98	1.00E-05 %I C1 C2 C3 C4

Table 3
Comparison of MCSs.

No.	MCSs from fault tree f (X,H) in Eq. (20)	MCSs from fault tree f (X,C) in Eq. (22)
1	1.00E-05 %I H1 H2 H3 H4	1.25E-03 %I C15
2	1.00E-05 %I A H2 H3 H4	2.50E-04 %I A C14
3	1.00E-05 %I B H1 H3 H4	2.50E-04 %I B C13
4	1.00E-05 %I C H1 H2 H4	2.50E-04 %I C C12
5	1.00E-05 %I D H1 H2 H3	2.50E-04 %I D C11
6	1.00E-05 %I A B H3 H4	5.00E-05 %I A B C10
7	1.00E-05 %I A C H2 H4	5.00E-05 %I A C C9
8	1.00E-05 %I A D H2 H3	5.00E-05 %I A D C8
9	1.00E-05 %I B C H1 H4	5.00E-05 %I B C C7
10	1.00E-05 %I B D H1 H3	5.00E-05 %I B D C6
11	1.00E-05 %I C D H1 H2	5.00E-05 %I C D C5
12	1.00E-05 %I A B C H4	1.00E-05 %I A B C C4
13	1.00E-05 %I A B D H3	1.00E-05 %I A B D C3
14	1.00E-05 %I A C D H2	1.00E-05 %I A C D C2
15	1.00E-05 %I B C D H1	1.00E-05 %I B C D C1
16	1.00E-05 %I A B C D	1.00E-05 %I A B C D

Table 5
Fault tree and HFE combinations.

Gates	3602
Events	6659
Complemented gates	130
Complemented events	1
Initiating events	105
HFEs	134
HFE combinations[a]	1118

[a] HFE combinations in HFE recovery rules.

Table 6
Comparison of HFE combination calculations (truncation limit = 1.0E-10).

Cases	1	2	3	4
HEPs	Nominal	Max (HEP, 0.1)	Max (HEP, 0.5)	New method
Run time	23 s.	84 s.	1125 s.	86 s.
MCSs	32,400	1,495,524	15,381,642	1,378,326
2 HFEs[a]	54	303	538	114
3 HFEs[a]	29	611	1752	140
4 HFEs[a]	1	398	1921	101
5 HFEs[a]	0	31	415	11
6 HFEs[a]	0	0	22	2
Total HFEs[a]	84	1343	4648	368

[a] HFE combinations in MCSs.

of Case 4 after HFE recovery. Furthermore, HFE combinations that have 6 HFEs are not calculated.

3. (Case 3) MCSs are generated using the HEP of Max (HEP, 0.5) and then HFE recovery is performed. Many MCSs that have 2 to 6 HFEs are calculated. However, many unnecessary MCSs are calculated, and most of them will be truncated after HFE recovery, and this greatly increases the calculation time. The total number of HFE combinations is 4648 in Table 6, which is bigger than 1118 in Table 5. However, unnecessary MCSs that have 4648 HFE combinations will be truncated after HFE recovery.
4. (Case 4) MCSs are generated by the new method that is implemented in FTREX. All HFE combinations are calculated, and there is no need to perform additional HFE recovery, since MCS generation and HFE recovery are performed simultaneously.

5. Conclusions

With current HFE recovery techniques, there is no certainty that all possible HFE combinations are generated and dependencies among HFEs are properly reflected. Furthermore, it takes a long time to generate many MCSs with elevated HEPs and perform HFE recovery in two steps. Accordingly, there has been a great need to minimize the burden of HFE recovery by incorporating it into the MCS generation stage.

By merging HFE recovery into MCS generation in this study, there is no need to lower the truncation limit or to elevate HEPs after developing HFE recovery rules. Therefore, this new method drastically reduces the burden of MCS generation and HFE recovery. Furthermore, this new method minimizes the possibility of losing HFE combinations when generating MCSs from the fault tree. Thus, this new method improves CDF accuracy. The effectiveness and strength of the new method were clearly demonstrated and discussed with simple fault trees.

This new method is simple but very effective for performing MCS generation and HFE recovery simultaneously and improving CDF accuracy. It is recommended that the new method be employed for various PSAs and their applications, such as risk monitors, for the fast and accurate calculation of CDF. Furthermore, it can be implemented in various PSA tools, as it was implemented in the fault tree solver FTREX [1–3].

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the Korea Foundation of Nuclear Safety (KoFONS) grant funded by the Nuclear Safety and Security Commission (NSSC), Republic of Korea (No. 2101062), and also supported by the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT, Republic of Korea (No. 1711126864).

References

[1] W.S. Jung, S.H. Han, J.J. Ha, A fast BDD algorithm for the reliability analysis of large coherent systems, Reliab. Eng. Syst. Saf. 83 (3) (2004) 369–374.
[2] W.S. Jung, S.H. Han, J.J. Ha, An overview of the fault tree solver FTREX, in: 13th International Conference on Nuclear Engineering, Beijing, China, May 16–20, 2005.
[3] W.S. Jung, ZBDD algorithm features for an efficient probabilistic safety assessment, Nucl. Eng. Des. 239 (10) (2009) 2085–2092.
[4] C.L. Smith, R.D. Fowler, SAPHIRE 'Recover Cut Sets' editor: a technique for automating the manipulation of PRA cut sets, Reliab. Eng. Syst. Saf. 54 (1996) 47–52.
[5] W.S. Jung, A method to improve cutset probability calculation in probabilistic safety assessment of nuclear power plants, Reliab. Eng. Syst. Saf. 134 (2015) 134–142.
[6] Snl, Nureg/Cr-1278, Handbook of Human Reliability Analysis with Emphasis on Nuclear Power Plant Applications, 1983.
[7] M. Cepin, DEPEND-HRA—a method for consideration of dependency in human reliability analysis, Reliab. Eng. Syst. Saf. 9 (2008) 1452–1460.
[8] Epri, A Process for HFE Dependency Analysis and Consideration on Use of Minimum Values for Joint Human Error Probabilities, Product ID 3002003150, 2016.
[9] Epri, HRA Calculator Ver. 5.2, Product ID 3002010680, 2017.
[10] A.M. Arigi, G. Park, J. Kim, Dependency Analysis Method for Human Failure Events in Multi-Unit Probabilistic Safety Assessments, vol. 203, 2020.
[11] Epri, Phoenix Architect Module Version 1.0b, Software Manual, Software Product ID 3002020050, 2020.
[12] S.H. Han, K.M. Oh, H.G. Lim, J.E. Yang, AIMS-MUPSA Software Package for Multi-Unit PSA, vol. 50, 2018, pp. 1255–1265.
[13] H. Seok, D.K. Kim, H.K. Lim, S.K. Kang, Development of an integrated software for reliability analysis and PSA, SAREX, in: Proceedings of the European Safety and Reliability Conference, ESREL, 2005.