Nuclear Engineering and Technology 56 (2024) 4289–4295

Contents lists available at ScienceDirect

Nuclear Engineering and Technology

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

Technical Note

A hybrid approach of partially applying BDD for seismic PSA quantification

Sang Hoon Han

Risk Assessment Research Division, Korea Atomic Energy Research Institute, 1045 Daedeokdaero, Yuseong-gu, Daejeon, 305-353, Republic of Korea

ARTICLE INFO

Keywords:
Seismic PSA
Partial BDD
Hybrid approach
PSA quantification

ABSTRACT

The binary decision diagram (BDD) method provides a means to calculate the exact probability of a fault tree, but cannot solve large fault trees for nuclear power plant PSAs. However, the BDD method can be used as a supplementary means to increase the accuracy of PSA quantification, especially for a seismic PSA which includes many non-rare events.

Since it is difficult to solve a large PSA model using BDD method, several approaches have been developed that partially apply the BDD technique to small-sized branches of a large PSA model. This paper proposes an approach of partially applying BDD technique to large-sized branches. The results from the pilot application to a seismic PSA model shows that the partial BDD approach of this paper provides a relatively accurate way to quantify each sequence as well as the overall core damage in a seismic PSA model.

1. Introduction

A typical probabilistic safety assessment (PSA) quantification is performed as follows:

1. Calculate minimal cut sets (MCS) of the fault tree for the PSA model (where the negate corresponding to a success branch is processed by 'delete term approximation', and the unimportant cut sets are truncated)
2. Calculate the probability using 'rare event approximation' (REA) or 'minimal cut upper bound' (MCUB) method from the obtained minimal cut sets

This PSA quantification method produces relatively accurate results for internal event PSAs, but the quantification error becomes large for seismic PSAs with many non-rare events (probabilities greater than approximately 0.1).

The sum-of-disjoint-product (SDP) technique is a useful tool for calculating the exact probability of a fault tree [1,2]. The binary decision diagram (BDD) technique can be used to transform a fault tree into SDP form (which is called as BDD logic in this paper).

The previous traditional seismic PSA uses an approach to quantify a small-sized seismic initiator event tree (SIET), that models important seismic failures, using the BDD method [3,4] and to pass only the resulting values to the next scenario, secondary seismic event trees (SSET). Recently, seismic PSA methodology has shifted toward integrating and analyzing both SIET and SSET. To increase the quantification accuracy for the integrated seismic PSA model, techniques have been developed to partially introduce BDD method.

However, since it is difficult to solve a large PSA model using BDD technique, several approaches to partially apply the BDD technique to a large PSA model have been used. Kim and Kim [5] presents insights for an approach that reduces quantification errors by converting a success branch into SDP form, which is introduced as a negate-down approach in FTREX [6]. Lim [7] describes a technique for converting important parts of a seismic PSA into BDD logic, combining them with the remaining parts, and then solving them using the typical PSA quantification method. The basic idea of these two approaches is to solve small-sized branches using BDD technique, and to solve the rest using the typical PSA quantification method. In some cases, a technique of converting the calculated cut sets into BDD logic can be used to reduce quantification errors [8]. Monte Carlo methods [9,10] are good tools to quantify high-probability PSA models, such as seismic PSA, with reasonable accuracy. But, it does not produce minimal cut sets required for PSA, so it is used as a supplementary means to verify the quantification results of seismic PSA.

In this paper we extend approaches of Lim, Kim and Kim and proposes an approach of utilizing the BDD technique even for large-sized branches in a seismic PSA. Section 2 presents the proposed approach, Section 3 shows the pilot application results of the proposed method, and Section 4 provides a summary.

E-mail address: shhan2@kaeri.re.kr.

https://doi.org/10.1016/j.net.2024.05.034
Received 13 December 2023; Received in revised form 19 April 2024; Accepted 25 May 2024
Available online 26 May 2024

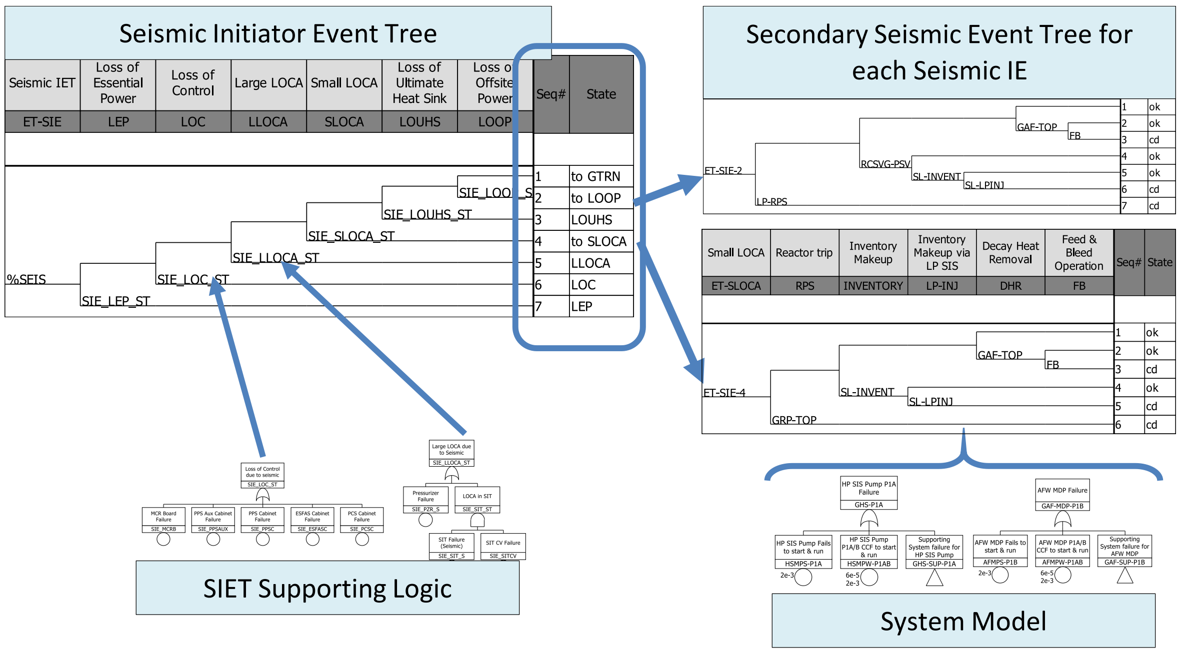

Fig. 1. Typical plant response model for seismic PSA.

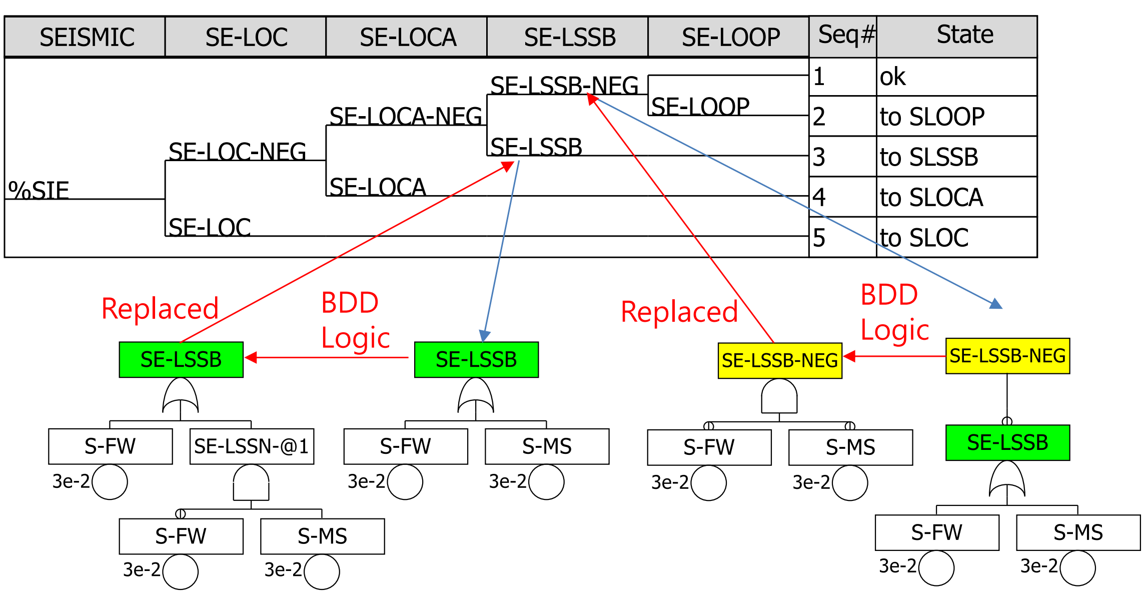

Fig. 2. An example of BDD conversion for a small-sized branch.

2. Proposed hybrid approach to incorporate partial BDD

A typical seismic PSA model consists of a SIET, which models the initial scenario after an earthquake occurs, and SSETs, which model the operation of the safety systems, as shown in Fig. 1. Each branch of an event tree is modeled as a fault tree. Important seismic events are mainly modeled in the SIET. Therefore, the small-sized SIET is solved by BDD method, and large-sized SSET is quantified by the typical PSA quantification method [5].

This paper presents a hybrid approach of applying the BDD technique to such large-sized branches. The hybrid approach proposed in this paper are as follows, and detailed information is given in sections 2.1 and 2.2 below.

- Step A. Small-sized branches in SIET are converted into BDD logic (section 2.1).
- Step B. BDD technique is partially applied to large-sized branches in SSET (section 2.2).
- Step C. After replacing the branches with fault trees corresponding to BDD logic, the typical PSA quantification method is used.

2.1. Approach for a small-sized branch

A small-sized branch is converted to BDD logic. All failure branches and success branches in SIET are replaced with fault trees corresponding to BDD logic. Fig. 2 illustrates an example of replacing a failure branch SE-LSSB and a success branch SE-LSSB-NEG (=$\overline{\text{SE–LSSB}}$, the negate of

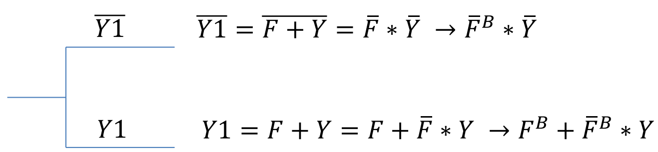

Fig. 3. Partial BDD approach for a large-sized branch.

the failure branch) with BDD logic.

2.2. *Approach for a large-sized branch*

2.2.1. *Basic approach for a large-sized branch*

For a large-sized branch, only important parts are converted to BDD logic. If the important part in the failure branch *Y1* is *F*, *Y1* is separated into *F* and *Y*:

$$Y1 \rightarrow F + Y \tag{1}$$

where

- *F* is an important part for the branch from a quantification perspective.
- *Y* is the rest of *Y1* with *F* removed.

If *Y1* is converted into the BDD logic focusing on a key part *F*, it becomes as follows:

$$Y1 = F + Y \rightarrow F + \bar{F} * Y \tag{2}$$

To increase the quantification accuracy, we convert *F* and $\bar{F}$ into BDD logic F^B and $\bar{F}^B$, respectively, which are important parts from a quantification perspective. Then, the failure branch *Y1* and success branch $\overline{Y1}$ can be converted as shown in Fig. 3. (We simply call this approach the 'partial BDD approach' in this paper.)

F^B and $\bar{F}^B$ represents the BDD logic in the SDP form. For the converted model, a typical PSA quantification method is used. For example, the negate $\bar{Y}$ is processed using 'delete term approximation'.

2.2.2. *Finding the important part for a branch*

We need to select an important part *F* for a large-sized branch carefully:

- The analyst may prepare it based on the analyst's judgment, or

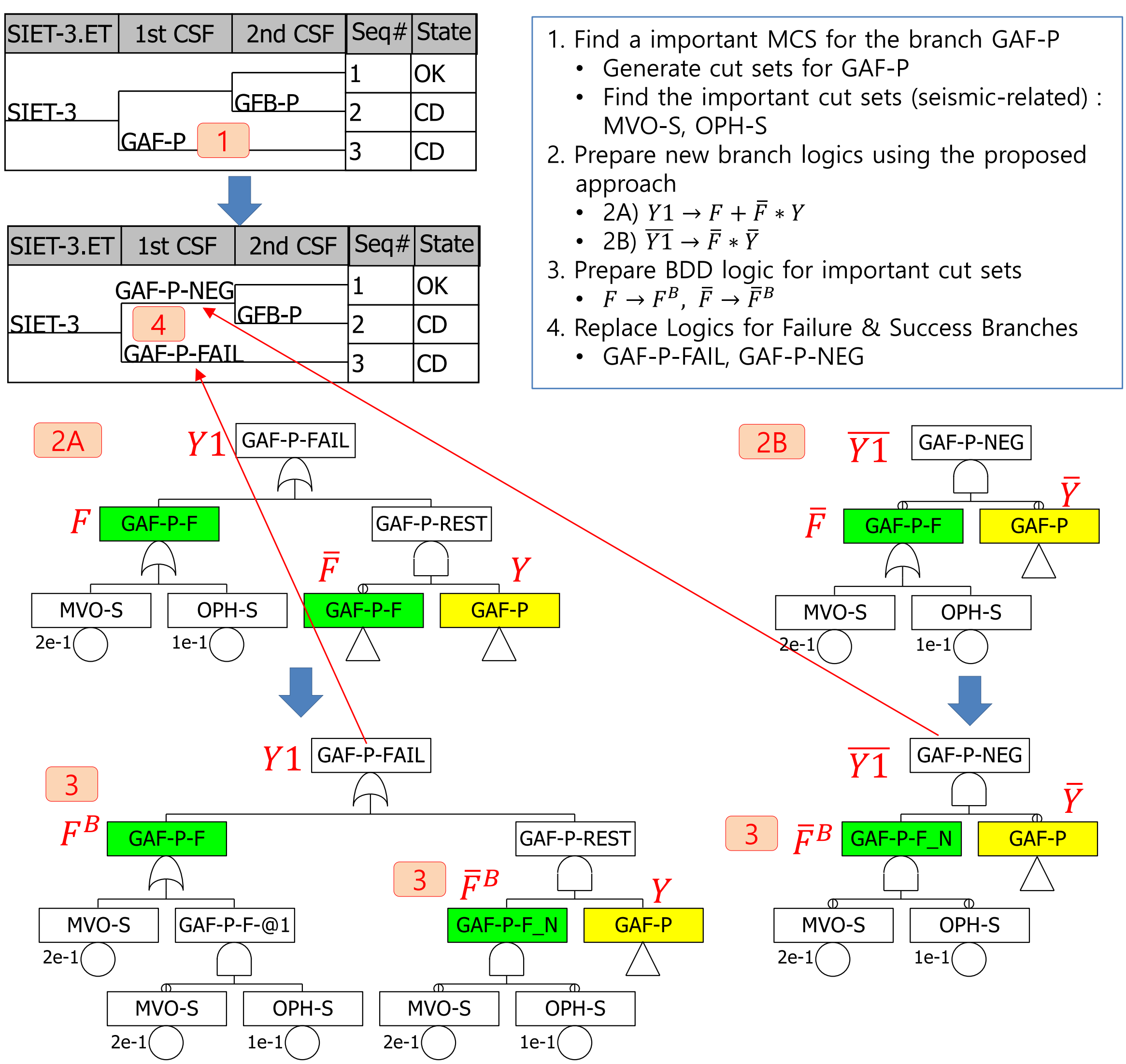

Fig. 4. An example of Partial BDD conversion for a large-sized branch.

Table 1
Pilot application of partial BDD approach to a seismic PSA (high PGA of 1.2247g).

Sequence	MC(n = 1e9[(1)])	SD Ratio[(2)]	SIET-bdd	Ratio[(3)]	SSET-pbdd	Ratio[(4)]
SLEP-1!	2.34E-01	0.02 %	2.34E-01	1	2.34E-01	1
SSLOCA-1!	3.57E-02	0.05 %	3.57E-02	1	3.57E-02	1
SSTRUC-1!	6.83E-01	0.01 %	6.83E-01	1	6.83E-01	1
SLOOP-2!	3.25E-07	23.62 %	4.69E-07	1.443	3.28E-07	1.010
SLOOP-4!	3.30E-08	60.70 %	1.57E-07	4.767	3.41E-08	1.034
SLOOP-5!	3.88E-03	0.08 %	1.04E-02	2.678	3.92E-03	1.011
SLOOP-6!	1.06E-03	0.33 %	2.01E-03	1.905	1.07E-03	1.012
SLOOP-7!	3.46E-03	0.22 %	5.21E-03	1.507	3.50E-03	1.012
SLOOP-8!	2.19E-03	0.18 %	2.54E-03	1.158	2.22E-03	1.012
SLOOP-9!	5.49E-04	0.54 %	5.49E-04	0.999	5.49E-04	0.999
SSSLB-3!	2.10E-04	0.70 %	5.11E-04	2.433	2.14E-04	1.019
SSSLB-4!	5.70E-05	1.52 %	9.89E-05	1.734	5.82E-05	1.021
SSSLB-5!	1.87E-04	0.78 %	2.56E-04	1.369	1.91E-04	1.019
SSSLB-6!	1.18E-04	0.84 %	1.25E-04	1.054	1.21E-04	1.021
SSSLB-7!	6.25E-06	3.82 %	6.21E-06	0.994	6.21E-06	0.994
SSSLB-8!	9.01E-06	4.32 %	8.99E-06	0.998	8.99E-06	0.998
Sum	9.64E-01	0.00 %	9.74E-01	1.01	9.64E-01	1

Note.
[1])1.0e-3 represents 1.0×10^{-3}.
[2])SD Ratio is calculated by dividing the standard deviation of MC(n = 1e9) by MC(n = 1e9).
[3])The ratio is calculated by dividing SIET-bdd by MC(n = 1e9).
[4])The ratio is calculated by dividing SSET-pbdd by MC(n = 1e9).

- The best way is to select the top cut sets after calculating the minimal cut sets of the branch (maybe 95–99 % cut sets, or those with a value of 0.1 or higher).
- The size of F should be small to be easily solved by BDD method.

2.2.3. Removing the important part from the original branch

If we want to remove F from the failure branch *Y1* in order to get Y, we need to modify the branch logic manually. This can be a time-consuming task.

Fortunately, instead of using Y where F is removed from *Y1*, we can get the same result by using the original branch *Y1* as is. If we use *Y1* instead of Y, we get an extra $\overline{F}^B * F$ that becomes null. The proof and example for $\overline{F}^B * F = 0$ are given in Appendix.

$$Y1 \rightarrow F^B + \overline{F}^B * Y \rightarrow F^B + \overline{F}^B * Y1 = F^B + \overline{F}^B * (F + Y)$$

$$= F^B + \overline{F}^B * Y + \overline{F}^B * F = F^B + \overline{F}^B * Y \tag{3}$$

Y1 can be also used instead of Y for the success branch $\overline{Y1}$, where $\overline{F}^B * \overline{F}$ becomes $\overline{F}^B$. The proof and example for $\overline{F}^B * \overline{F} = \overline{F}^B$ are given in Appendix.

$$\overline{Y1} = \overline{F}^B * \overline{Y} \rightarrow \overline{F}^B * \overline{Y1} = \overline{F}^B * \overline{(F + Y)} = \overline{F}^B * \overline{F} * \overline{Y} = \overline{F}^B * \overline{Y} \tag{4}$$

Thus, we can select any one from two approaches:

Table 2
Pilot application of partial BDD approach to a seismic PSA (moderate PGA of 0.6124g).

Sequence	MC(n = 1e9[(1)])	SD Ratio[(2)]	SIET-bdd	Ratio[(3)]	SSET-pbdd	Ratio[(4)]
SLEP-1!	5.54E-02	0.01 %	5.54E-02	1	5.54E-02	1
SSLOCA-1!	1.72E-02	0.02 %	1.72E-02	1	1.72E-02	1
SSTRUC-1!	3.72E-02	0.01 %	3.72E-02	1	3.72E-02	1
SLOOP-2!	8.00E-06	1.16 %	8.08E-06	1.01	7.97E-06	0.996
SLOOP-4!	1.02E-07	36.63 %	1.20E-07	1.179	1.04E-07	1.024
SLOOP-5!	5.91E-04	0.31 %	6.60E-04	1.117	5.94E-04	1.006
SLOOP-6!	1.17E-04	0.83 %	1.27E-04	1.085	1.17E-04	0.998
SLOOP-7!	2.73E-04	0.45 %	2.91E-04	1.067	2.72E-04	0.999
SLOOP-8!	6.16E-04	0.39 %	6.25E-04	1.015	6.17E-04	1.001
SLOOP-9!	1.43E-04	0.70 %	1.44E-04	1.002	1.44E-04	1.002
SSSLB-3!	2.44E-06	7.56 %	2.63E-06	1.08	2.38E-06	0.978
SSSLB-4!	4.55E-07	14.70 %	5.06E-07	1.112	4.68E-07	1.029
SSSLB-5!	1.14E-06	6.71 %	1.16E-06	1.021	1.09E-06	0.960
SSSLB-6!	2.38E-06	6.26 %	2.50E-06	1.049	2.47E-06	1.040
SSSLB-7!	3.94E-07	20.45 %	3.85E-07	0.977	3.85E-07	0.977
SSSLB-8!	8.00E-09	129.13 %	8.47E-09	1.059	8.47E-09	1.059
Sum	1.12E-01	0.00 %	1.12E-01	1.001	1.12E-01	1

- Approach 1) use the branch model with selected part F removed
  - ■ $Y1 \rightarrow F^B + \overline{F}^B * Y$, $\overline{Y1} \rightarrow \overline{F}^B * \overline{Y}$ (5)
- Approach 2) use the branch model as is
  - ■ $Y1 \rightarrow F^B + \overline{F}^B * Y1$, $\overline{Y1} \rightarrow \overline{F}^B * \overline{Y1}$ (6)

2.2.4. An example for a large-sized branch

Fig. 4 shows an example of preparing and replacing the logic of the failure branch GAF-P-FAIL (*Y1*) and the success branch GAF-P-NEG ($\overline{Y1}$) by applying the partial BDD approach. MVO-S and OPH-S are important cut sets in the original failure branch called GAF-P. The original logic of the failure branch GAF-P is used for Y as is.

3. Pilot application to a seismic PSA

The partial BDD approach proposed in this paper is tested for a pilot seismic PSA model. The seismic PSA model consists of one SIET and two SSETs, and contains 3807 gates and 3165 basic events. Major seismic failures are modeled in the SIET, which has small branch sizes and is easily converted into BDD logic. The branch of SSET contains seismic failures and is large in size, making it difficult to solve with BDD method. Therefore, it is processed using the partial BDD approach proposed in this paper.

With this seismic PSA model, calculations are performed for two cases: PGA = 1.2247g (high probability of seismic failure) and PGA = 0.6124g (moderate probability of seismic failure). The conditional core damage probability (CCDP) of each sequence is calculated using the proposed approach. Since there is no way to calculate the exact value of each sequence, the CCDP of each sequence is estimated using the Monte Carlo method using a sufficiently large sample number of 10^9 and compared with the approach proposed in this paper.

The results are given in Tables 1 and 2. Among the sequences, SLEP-1, SSLOCA-1, and SSTRUCT-1 are SIET sequences that are solved using the BDD method, and the rest are SSET sequences that are difficult to solve with BDD, so the partial BDD approach is applied.

The results are summarized as follows, and it can be seen that the SSET-pbdd case, corresponding to the approach proposed in this paper, provides fairly accurate results.

- MC (n = 1e9): Calculate using Monte Carlo method. The number of samples is 1e9. SD Ratio (the standard deviation of Monte Carlo method) shows that CCDP of 1e-5 or higher is calculated fairly accurately by Monte Carlo method.
- SIET-bdd: Converts the SIET to BDD, and calculates the rest using the typical PSA quantification method. The errors for SSET sequences are very large. For high PGA case, the failure probabilities of several components are beyond 0.1. CCDP is calculated 4 times larger for the SLOOP-4! sequence. For moderate PGA case, the failure probabilities of components are around 0.04 or less (which is much smaller than that for high PGA case). The maximum error is about 1.18 times for SLOOP-4! sequence.
- SSET-pbdd: Convert the SIET to BDD, and process SSET using partial BDD approach. The difference of SSET sequences to Monte Carlo method are less than 3.4 % for high PGA case, and 6 % for moderate PGA case. The results of the Monte Carlo method have large deviations for sequences with small CCDPs, so the difference between the SSET-pbdd and the Monte Carlo method does not necessarily mean an error in the SSET-pbdd approach.

Note that the REA method should be applied to the calculated minimal cut sets. Since the cut sets contain many negate events due to the use of the SDP, using the MCUB method may lead to underestimated results.

4. Summary

The BDD method provides a means to reduce quantification errors for seismic PSAs which have many non-rare events. Recently, methods to improve seismic PSA accuracy by converting small-sized branches to BDD have been developed and used.

This paper proposes an approach of partially applying BDD technique to large-sized branches. The partial BDD approach of this paper provides a relatively accurate way to quantify seismic PSA. The pilot application to a seismic PSA shows that the quantification error can be greatly reduced.

After replacing the branches with fault trees corresponding to BDD logic, the typical PSA quantification method is used. Therefore, other than BDD conversion, there is also the advantage that existing PSA analysis software can be used as is.

CRediT authorship contribution statement

Sang Hoon Han: Methodology, Writing – original draft, Writing – review & editing.

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.

Acknowledgement

This work was supported by the Nuclear Safety Research Program through the Korea Foundation of Nuclear Safety (KoFONS) using financial resources granted by the Nuclear Safety and Security Commission (NSSC) of Korea (No. RS-2023-00240452).

Appendix. Proof of $\overline{F}^B * F = 0$ and $\overline{F} * \overline{F}^B = \overline{F}^B$

A1. Proof of $\overline{F}^B * F = 0$

Before verifying $\overline{F}^B * F = 0$, we need to verify $C_x\overline{C_x}^B = 0$. Suppose x-th cut set C_x consists of the multiplication of M events.

$$C_x = E_1\ E_2\ E_3 \ldots\ E_M = \prod\nolimits_{j=1}^{M} E_j, \text{ where } E_j \text{ is the j} - \text{th event in the cut set } C_x \tag{A1}$$

$\overline{C_x}$, the negate of C_x, is expressed as follows:

$$\overline{C_x} = \overline{E_1} + \overline{E_2} + \overline{E_3} + \ldots + \overline{E_M} \tag{A2}$$

$\overline{C_x}^B$, the SDP form of $\overline{C_x}$, becomes:

$$\overline{C_x}^B = \overline{E_1} + E_1\overline{E_2} + E_1E_2\overline{E_3} + \ldots + E_1E_2E_3\ldots\overline{E_M} \tag{A3}$$

The k-th term of $\overline{C_x}^B$ and C_x are expressed:

$$\text{k} - \text{th term of } \overline{C_x}^B = \overline{E_k}\prod\nolimits_{j<k} E_j \tag{A4}$$

$$C_x = E_k\prod\nolimits_{j\neq k} E_k \tag{A5}$$

The multiplication of C_x and k-th term of $\overline{C_x}^B$ becomes 0 because it includes the multiplication of E_k and $\overline{E_k}$ as follows:

$$\left(E_k\prod\nolimits_{j\neq k} E_k\right)\left(\overline{E_k}\prod\nolimits_{j<k} E_j\right) = 0 \tag{A6}$$

Thus, $C_x\overline{C_x}^B$ becomes 0:

$$C_x\overline{C_x}^B = (E_1\ E_2\ E_3\ldots\ E_M)(\overline{E_1} + E_1\overline{E_2} + E_1E_2\overline{E_3} + \ldots + E_1E_2E_3\ldots\overline{E_M}) = 0 \tag{A7}$$

Suppose F consists of N minimal cut sets:

$$F = C_1 + C_2 + C_3 + \dots + C_N = \sum_{i=1}^{N} C_i, \text{where } C_i \text{ is the i} - \text{th cut set} \tag{A8}$$

$\overline{F}$, the negate of F, is expressed as follows:

$$\overline{F} = \overline{C_1}\ \overline{C_2}\ \overline{C_3} \dots \overline{C_N} \tag{A9}$$

$\overline{F}^B$, the SDP form of $\overline{F}$, becomes:

$$\overline{F}^B = \overline{C_1}^B\ \overline{C_2}^B\ \overline{C_3}^B \dots \overline{C_N}^B \tag{A10}$$

The multiplication of k-th term of F and $\overline{F}^B$ becomes 0 because $C_x \overline{C_x}^B = 0$ as follows:

$$C_k \left(\overline{C_k}^B \prod_{i \neq k} \overline{C_i}^B \right) = 0 \tag{A11}$$

Thus, $F * \overline{F}^B$ becomes 0:

$$F * \overline{F}^B = (C_1 + C_2 + C_3 + \dots + C_N)\left(\overline{C_1}^B \overline{C_2}^B\ \overline{C_3}^B \dots \overline{C_N}^B \right) = 0 \tag{A12}$$

Let me show an example:

$$F = AB + AC$$

$$\overline{F} = \overline{(AB + AC)} = \overline{AB}\ \overline{AC} = (\overline{A} + \overline{B})(\overline{A} + \overline{C}) = \overline{A} + \overline{BC}$$

$$\overline{F}^B = \overline{A} + A\overline{BC}$$

$$F * \overline{F}^B = (AB + AC)(\overline{A} + A\overline{BC}) = 0 \tag{A13}$$

A2. Proof of $\overline{F} * \overline{F}^B = \overline{F}^B$

Before verifying $\overline{F} * \overline{F}^B = \overline{F}^B$, we need to verify $\overline{C_x} * \overline{C_x}^B = \overline{C_x}^B$.

$\overline{C_x}$ and $\overline{C_x}^B$ are as follows as in Eqs. A2 and A3:

$$\overline{C_x} = \overline{E_1} + \overline{E_2} + \overline{E_3} + \dots + \overline{E_M} \tag{A14}$$

$$\overline{C_x}^B = \overline{E_1} + E_1\overline{E_2} + E_1E_2\overline{E_3} + \dots + E_1E_2E_3 \dots \overline{E_M} \tag{A15}$$

Suppose we multiply each term of $\overline{C_x}$ to $\overline{C_x}^B$.

$$\overline{E_1}\ \overline{C_x}^B = \overline{E_1}(\overline{E_1} + E_1\overline{E_2} + E_1E_2\overline{E_3} + \dots + E_1E_2E_3 \dots \overline{E_M}) = \overline{E_1} \tag{A16}$$

$$\overline{E_2}\ \overline{C_x}^B = \overline{E_2}\ (\overline{E_1} + E_1\overline{E_2} + E_1E_2\overline{E_3} + \dots + E_1E_2E_3 \dots \overline{E_M}) = \overline{E_2}\ \overline{E_1} + E_1\overline{E_2} \rightarrow E_1\overline{E_2},$$

where $\overline{E_2}\ \overline{E_1}$ is the superset of $\overline{E_1}$ in Eq. A16, and is subsumed. (A17)

$$\overline{E_3}\ \overline{C_x}^B = \overline{E_3}(\overline{E_1} + E_1\overline{E_2} + E_1E_2\overline{E_3} + \dots + E_1E_2E_3 \dots \overline{E_M})$$

$$= \overline{E_3}\ \overline{E_1} + \overline{E_3}E_1\overline{E_2} + E_1E_2\overline{E_3} \rightarrow E_1E_2\overline{E_3}$$

Where $\overline{E_3}\ \overline{E_1}$ and $\overline{E_3}E_1\overline{E_2}$ are subsumed by $\overline{E_1}$ and $E_1\overline{E_2}$, respectively. (A18)

$$\overline{E_M}\ \overline{C_x}^B = \overline{E_M}(\overline{E_1} + E_1\overline{E_2} + E_1E_2\overline{E_3} + \dots + E_1E_2E_3 \dots \overline{E_M})$$

$$= \overline{E_M}\ \overline{E_1} + \overline{E_M}\ E_1\overline{E_2} + \overline{E_M}\ E_1E_2\overline{E_3} + \dots + E_1E_2E_3 \dots \overline{E_M} \rightarrow E_1E_2E_3 \dots \overline{E_M} \tag{A19}$$

The k-th term of $\overline{C_x}$ multiplied by the preceding terms than k-th term in $\overline{C_x}^B$ are superset of others and are subsumed. Terms multiplied by the next terms than k-th term in $\overline{C_x}^B$ becomes 0. Only k-th term of $\overline{C_x}^B$ remains. The multiplication of k-th term of $\overline{C_x}$ and $\overline{C_x}^B$ becomes k-th term of $\overline{C_x}^B$. Thus, the multiplication of $\overline{C_x}$ and $\overline{C_x}^B$ becomes $\overline{C_x}^B$:

$$\overline{C_x} * \overline{C_x}^B = (\overline{E_1} + \overline{E_2} + \overline{E_3} + \dots + \overline{E_M})(\overline{E_1} + E_1\overline{E_2} + E_1E_2\overline{E_3} + \dots + E_1E_2E_3 \dots \overline{E_M})$$

$$= \overline{E_1} + E_1\overline{E_2} + E_1E_2\overline{E_3} + \dots + E_1E_2E_3 \dots \overline{E_M} = \overline{C_x}^B \tag{A20}$$

$\overline{F}$ and $\overline{F}^B$ are as follows:.

$$\overline{F} = \overline{C_1}\ \overline{C_2} \dots \overline{C_N} = \prod(\overline{C_i}) \tag{A21}$$

$$\overline{F}^B = \overline{C_1}^B\,\overline{C_2}^B \ldots \overline{C_N}^B = \prod\left(\overline{C_i}^B\right) \tag{A22}$$

The multiplication of $\overline{F}$ and $\overline{F}^B$ becomes $\overline{F}^B$ as follows:.

$$\overline{F} * \overline{F}^B = \left(\overline{C_1}\,\overline{C_2}\,\ldots\,\overline{C_N}\right)\left(\overline{C_1}^B\,\overline{C_2}^B\,\ldots\overline{C_N}^B\right) = \prod\left(\overline{C_i C_i}^B\right) = \prod\left(\overline{C_i}^B\right) = \overline{F}^B,$$

where $\overline{C_i C_i}^B = \overline{C_i}^B$ (A23)

Let me show an example:

$$F = AB + AC$$

$$\overline{F} = \overline{(AB + AC)} = \overline{AB}\,\overline{AC} = (\overline{A} + \overline{B})(\overline{A} + \overline{C}) = \overline{A} + \overline{BC}$$

$$\overline{F}^B = \overline{A} + A\overline{BC}$$

$$F * \overline{F}^B = (AB + AC)(\overline{A} + A\overline{BC}) = 0 \tag{A24}$$

$$\overline{F} * \overline{F}^B = (\overline{A} + \overline{BC})(\overline{A} + A\overline{BC}) = (\overline{A} + \overline{A}A\overline{BC}) + (\overline{A}\overline{BC} + A\overline{BC}) = \overline{A} + A\overline{BC} = \overline{F}^B \tag{A25}$$

References

[1] R.E. Bryant, Graph-based algorithms for boolean function manipulation, IEEE Trans. Comput. C-35 (No.8) (1986).
[2] Antoine Rauzy, New algorithms for fault trees analysis, Reliab. Eng. Syst. Saf. 40 (1993) 203–211.
[3] Jung Han Kim, et al., Overview of development of PRASSE: probabilistic risk assessment of systems for seismic events, in: Transact Kor. Nuclear Soc. Autumn Meet, Gyeongju, Korea, 2011 Oct., pp. 27–28.
[4] EQE International Inc, EQESRA Reference Document Version 3, 1995.
[5] Ji Suk Kim, Man Cheol Kim, Insights gained from applying negate-down during quantification for seismic probabilistic safety assessment, Nucl. Eng. Technol. 54 (2022) 2933–2940.
[6] FTREX 2, 0 Software Manual, EPRI 3002018234, 2020.
[7] Hak Kyu Lim, A Simple approach to calculate CDF with non-rare events in seismic PSA model of Korean nuclear power plants, J. Korean Surg. Soc. 36 (5) (2021) 86–91.
[8] Woo Sik Jung, A method to improve cutset probability calculation in probabilistic safety assessment of nuclear power plants, Reliab. Eng. Syst. Saf. 134 (2015) 134–142.
[9] Sang Hoon Han, Enhancement of the FTeMC Software for Fault Tree Top Event Probability Evaluation Using Monte Carlo Approach, Transactions of the Korean Nuclear Society Spring Meeting, Jeju, Korea, 2018. May. 17-18.
[10] Tetsukuni Oikawa, et al., Development of systems reliability analysis code SECOM-2 for seismic PSA, Reliab. Eng. Syst. Saf. 62 (1998) 251–271.