Transactions of the Korean Nuclear Society Spring Meeting
Chuncheon, Korea, May 25-26 2006

Exact Minimal Cutset Quantification Using Equivalent Sum of Disjoint Products

Jong Soo Choi[a], Nam Zin Cho[b]
[a]*Korea Institute of Nuclear Safety, 19 Guseong-dong Yuseong-gu Daejeon 305-338; k209cjs@kins.re.kr*
[b]*Department. of Nuclear and Quantum Engineering, Korea Advanced Institute of Science and Technology, 373-1 Guseong-dong, Yuseong-gu, Daejeon, 305-701; nzcho@kaist.ac.kr*

1. Introduction

Fault tree analysis is extensively applied in probabilistic safety assessments (PSA). The majority of fault tree analysis methods are based on the minimal cut set (MCS) approach. In PSAs, the risk and importance measures are computed from a cutset equation mainly by using an approximation. The conservatism of the approximations is also a source of quantification uncertainty.

In this paper, a new MCS quantification method, which is based on the concept of 'sum of disjoint products (SDP)', is proposed and its applicability is described.

2. Method and Results

This section presents the concept and algorithm for generating an equivalent SDP to a set of MCSs.

2.1 MCS Quantification

In PSAs for nuclear power plants (NPPs), the risk measures (e.g., core damage frequency (CDF) and large early release frequency (LERF)) and importance measures (e.g., risk achievement worth (RAW), risk reduction worth (RRW), Fussell-Vesely (FV)) are computed from a cutset equation mainly by using "rare event" approximation or "min cut upper bound" approximation. It is well known that these approximations always provide conservative probabilities [1].

In order to calculate the risk and importance measures, some unavailability functions are computed from a set of developed MCSs, $\{\mathbf{K}_i \mid i = 1, \ldots, m\}$ as follows:

$$h^m(\mathbf{p}) = \Pr\{\bigcup_{i=1}^{m} \mathbf{K}_i\} \tag{1}$$

$$h^m(0_i, \mathbf{p}) = \Pr\{\bigcup_{i=1}^{m} \mathbf{K}_i \mid p_i = 0\} \tag{2}$$

$$h^m(1_i, \mathbf{p}) = \Pr\{\bigcup_{i=1}^{m} \mathbf{K}_i \mid p_i = 1\}, \tag{3}$$

where $\mathbf{K}_i$ is the i-th MCS. However, the unavailability functions calculated with "rare event" approximation are greater than exact unavailability functions:

$$h^m(\mathbf{p})_{RE} = \sum_{i=1}^{m} \Pr\{K_i\} > h^m(\mathbf{p}) \tag{4}$$

2.2 Concept of SDPs

The basic foundation of SDP algorithms [2-4] is to transform the set of cut sets into another set of mutually exclusive events (a set of disjoint products (*DP*)) and then reduce the probability evaluation to a simple summation given as:

$$\begin{aligned} h^m(\mathbf{p}) &= \Pr\{\bigcup_{i=1}^{m} \mathbf{K}_i\} \\ &= \Pr\{\mathbf{K}_1\} + \Pr\{\overline{\mathbf{K}_1}\mathbf{K}_2\} + \Pr\{\overline{\mathbf{K}_1}\,\overline{\mathbf{K}_2}\mathbf{K}_3\} + \cdots + \Pr\{\overline{\mathbf{K}_1}\,\overline{\mathbf{K}_2}\,\overline{\mathbf{K}_3} \cdots \mathbf{K}_m\} \\ &= \sum_{i=1}^{d} \Pr\{DP_i\}. \end{aligned} \tag{5}$$

Consequently, SDP algorithms compute the exact MCS probabilities, $h^m(\mathbf{p})$, $h^m(0_i,\mathbf{p})$ and $h^m(1_i,\mathbf{p})$ $\forall i$.

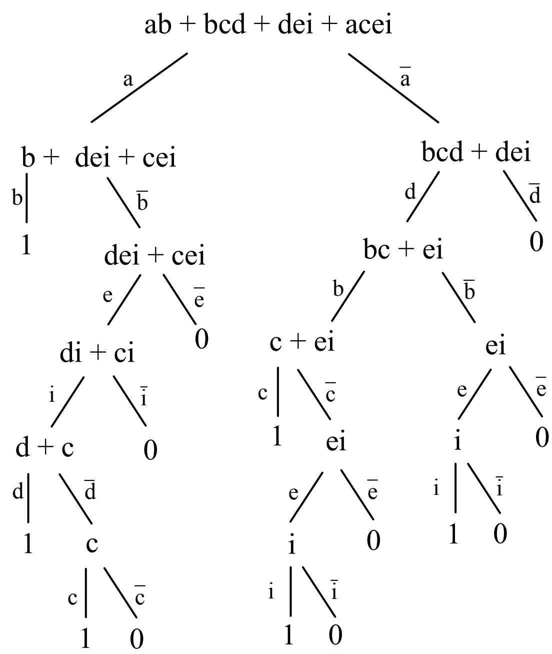
2.3 Proposed Algorithm for Generating Equivalent SDP

The proposed SDP algorithm is based on the factorization method [2-4]. This method works according to the following recursive principle. Given a formula F, either F is reduced to a constant or it is possible to select a pivot variable x and to study recursively the two formulae $F_{\bar{x}}$ and F_x , i.e., the formula F in which the constants 0 and 1, respectively, are substituted for the variable x. In other words, the method builds, at least implicitly, a tree. Leaves of this tree encode constants. Internal nodes encode formulae of the form $F = xF_x + \bar{x}F_{\bar{x}}$. Branches of the tree that lead to a 1-leaf are labeled with wanted disjoint products. This tree-like presentation of the algorithm makes clear its exponential complexity.

As an illustration, let us consider the set of MCSs, $S = ab + bcd + dei + acei$, taken from Ref. 2. The binary tree traversed by the SDP algorithm for S is shown in Figure 1. The sum of disjoint products developed by the proposed SDP algorithm is (from left to right): $ab + a\bar{b}eid + a\bar{b}ei\bar{d}c + \bar{a}dbc + \bar{a}db\bar{c}ei + \bar{a}d\bar{b}ei$. All the produced products are mutually disjoint and their sum is equivalent to the original formula S. Note that the proposed SDP algorithm does not provide necessarily the optimal sum of disjoint products equivalent to the original set of MCSs.

The efficiency of the SDP algorithm relies on two issues. First, the data structure used to encode sum-of-products $F_{\bar{x}}$ and F_x from F and x. Secondly, the branching heuristic. In order to be efficient, the proposed SDP algorithm uses the following simple branching heuristic. The selected pivot variable is the variable in single-event products or the most frequent variable in formula *F*. Furthermore, the proposed algorithm uses optimally-reduced formulae of F_x to make the data structure reduced.

The proposed SDP algorithm is similar to the ESOP method [2]. These methods are different from each other in the data structure and the branching heuristic. From some numerical tests, we can see that the proposed algorithm is faster than the ESOP method.



$$
\begin{aligned}
&\text{SDP}(ab + bcd + dei + acei)\\
&= ab + \overline{ab}\,bcd + \overline{ab}\,\overline{bcd}\,dei + \overline{ab}\,\overline{bcd}\,\overline{dei}\,acei\\
&= ab + a\bar{b}eid + a\bar{b}ei\bar{d}c + \bar{a}dbc + \bar{a}db\bar{c}ei + \bar{a}d\bar{b}ei\\
&= ab + \bar{a}bcd + (a\bar{b} + \bar{a}b\bar{c} + \bar{a}\bar{b})dei + \bar{b}\bar{d}acei\\
&= ab + \bar{a}bcd + (\bar{b} + \bar{a}b\bar{c})dei + \bar{b}\bar{d}acei
\end{aligned}
$$

Figure 1. An example of binary tree built by the proposed SDP algorithm

2.4 Numerical Example

Example 1 is the fault tree for the AFWS of Kori unit 3&4, which contains 613 logic gates and 677 independent basic events.

Example 2 is the fault tree 'European 1' (given in Ref. 5), which contains 84 logic gates and 61 independent basic events.

All runs were performed on a 2 GHz Pentium IV using the FORTRAN program based on the proposed SDP algorithm. The proposed algorithm provides $h^m(\mathbf{p})$, $h^m(0_i,\mathbf{p})$ and $h^m(1_i,\mathbf{p})$ $\forall i$ as output.

Table 1. MCS quantification results for example 1

Cut off value	# of MCS	$h^m(\mathbf{p})_{RE}$	$h^m(\mathbf{p})_{SDP}$	# of DPs	CPU time(s)
10^{-8}	230	2.416E-4	2.359E-4	1285	0.05
10^{-9}	806	2.436E-4	2.374E-4	23044	0.19
10^{-10}	2245	2.441E-4	2.377E-4	544199	1.58
10^{-11}	5423	2.442E-4	2.378E-4	50901357	127.11

$h^m(\mathbf{p})_{RE}$: calculated with "rare event" approximation
$h^m(\mathbf{p})_{SDP}$: calculated by the proposed algorithm

Table 2. MCS quantification results for example 2

Cut off value	# of MCS	$h^m(\mathbf{p})_{RE}$	$h^m(\mathbf{p})_{SDP}$	# of DPs	CPU time(s)
10^{-10}	1225	1.143E-6	9.189E-7	455953	0.79
10^{-11}	7613	1.370E-6	1.064E-6	7268250	13.20
0	46188	1.393E-6	1.078E-6	~ 2.8E9	5533.2

3. Conclusions

This paper presents an exact MCS quantification method using the equivalent SDP to a set of developed minimal cut sets. This method can be easily applied to most problems in PSAs. This method will be useful in reducing uncertainty in the field of PSA quantification.

REFERENCES

[1] J. S. Choi, N. Z. Cho, "Truncation Error Evaluation Method for Minimal Cut Set-Based Fault Tree analysis," *Journal of Nuclear Science and Technology*, Vol. 42, No. 10, pp. 854-860, 2005.

[2] A. Rauzy, E. Chatelet, Y. Dutuit, *et al*, "A Practical Comparison of Methods to Assess Sum-of-Products," *Reliability Engineering and System Safety*, Vol.79, p.33-42, 2003.

[3] A. S. Heger, J. K. Bhat, D. W. Stack, *et al*, "Calculating Exact Top-event Probabilities Using ∑∏-Patrec," *Reliability Engineering and System Safety*, Vol.50, p.253-259, 1995.

[4] M. Locks, "Recent Developments in Computing of System-Reliability," *IEEE Trans. on Reliability*, Vol.R-34, p.425-436, 1985.

[5] A. Rauzy, "New Algorithms for fault trees analysis," *Reliability Engineering and System Safety*, Vol.40, p.203-211, 1993.