

## ABSTRACT

VAISHANAV, PRAGYA. Uncertainty Quantification and Propagation for External Hazard Probabilistic Risk Assessment . (Under the direction of Dr. Abhinav Gupta.)

Uncertainty quantification in modelling and simulation helps in identifying the factors that affect mechanics based and mathematical models. In recent years, there is an increased emphasis on formal quantification of risk under external hazards while also quantifying the associated uncertainties and their propagation. Probabilistic risk assessment (PRA) is a widely used tool by the nuclear industry for safety during normal operations as well as for protection against external hazards. Computation of total risk in an external hazard PRA is dependent on hazard assessment, fragility assessment, and systems analysis. Hence, uncertainty propagation in the assessment of external hazards requires a careful investigation in the context of systems analysis.

A systems analysis for propagation of component fragilities is conducted using event and fault trees. The event and fault trees for an actual power plant can be fairly large in size, which imposes computational challenges. A fault and event tree analysis for a power plant requires modelling of hundreds of component failures, logic gates, multiple occurring events, and dependent events. Such interconnections create large networks that can lead to excessive computational demand. Hence, certain assumptions are employed for computational efficiency. Most of the traditional methods address computational demand by using assumptions or rely on high performance computing facilities that allow implementations of parallel computing. The assumptions typically represent the conditions imposed during the design basis (DB) scenarios. Traditional PRA tools based on these assumptions are also widely applied to perform risk assessment in the context of beyond design basis (BDB) scenarios. However, some of these assumptions may not be

valid for certain BDB scenarios. In addition, the probability of dependent failures also increases in BDB scenarios due to common cause failures (CCF). The errors due to the assumptions in the traditional tools and CCF, while negligible for DB scenarios, can be significantly more in a BDB scenario.

A novel framework for analyzing fault and event tree is presented to address the computational demands of PRA. The proposed solution also eliminates certain assumptions and approximations. The proposed algorithm is developed for networks with dependent nodes and multiple chains connecting one dependent node. Enhancements are provided for the proposed algorithm to consider multi hazard or multi unit accident scenarios and common cause failures.

It is shown that the computational demand of proposed algorithm increases only linearly with the size of the network in terms of the total number of basic events and intermediate events. It is also shown that the proposed algorithm gives identical results to those obtained by the traditional approaches and, for a simple case, identical to that calculated using fundamental concepts of set theory. Moreover, an extension of this study is performed to understand scaling uncertainties. The purpose is to enhance and support the scaling extrapolation and reduce the reliance on heuristics for improved uncertainty quantification. This study introduces and implements a methodology which examines experiments at multiple scales to infer the effect of scaling related uncertainty quantification.

© Copyright 2021 by Pragya Vaishnav

All Rights Reserved

Uncertainty Quantification and Propagation for External Hazard Probabilistic Risk  
Assessment

by  
Pragya Vaishanav

A dissertation submitted to the Graduate Faculty of  
North Carolina State University  
in partial fulfillment of the  
requirements for the Degree of  
Doctor of Philosophy

Civil Engineering

Raleigh, North Carolina

2021

APPROVED BY:

---

Dr. Murthy Narasimha Guddati

---

Dr. Kumar Mahinthakumar

---

Dr. Nam Dinh

---

Dr. Abhinav Gupta  
Chair of Advisory Committee

# DEDICATION

*To all my teachers...*

## BIOGRAPHY

Pragya Vaishnav was born on 15 July 1994 in a small town named Salumbar located in India. She lived in Salumbar until the completion of her primary and secondary education. She joined the undergraduate program in Civil Engineering at Indian Institute of Technology, Guwahati, India in 2010 and received the Bachelors of Technology degree in July, 2014. She then received a Masters of Technology (MTech) degree in Civil Engineering from the Indian Institute of Technology Kanpur, India in July 2017. During her masters, she worked on the stability analysis of bridges subjected to erosion. After her MTech, she joined North Carolina State University in August 2017 to pursue her doctoral degree in Structural Engineering and Mechanics. Her research interests include various fields of structural engineering: electrical cabinet response, validation of advanced simulation tools, seismic and flooding probabilistic risk assessment, uncertainty quantification, dynamic response of coupled system and machine learning.

She is a member of the engineering communities; ASCE, PENC and NICEE. She has served as a Teaching Assistant for undergraduate and graduate courses at NC State University and Indian Institute of Technology Kanpur. Apart from research, her interest lies in business development, psychology & human behaviour, jiu jitsu, and painting.

## ACKNOWLEDGEMENTS

Firstly, I would like to express my sincere gratitude to my advisor, Dr. Abhinav Gupta, for giving me the opportunity to pursue my doctoral studies at NCSU. His patience, motivation, immense knowledge and constant guidance helped me during the entire research work and in writing this dissertation. He helped me to build a new personality far better than my previous self. I consider myself extremely fortunate to get an opportunity for personal and professional development under his direction. I would also like to thank Dr. Nam Dinh, Dr. Kumar Mahinthakumar, Dr. Murthy Guddati, and Dr. Piyush Sabharwall for graciously agreeing to serve on my advisory committee and for their valuable suggestions.

My masters advisor, Dr. Durgesh Rai has played a crucial role in helping me to pursue a doctoral career. His guidance and direction enlightened me with preliminary footprints of research. I am grateful to my dear friend, Saran Srikanth for not only providing his input and feedback without which this research could not have been possible but also teaching me to take new initiatives and pursue opportunities. I am thankful to my roommates Abhinab, Sheik, SSB, Arshita, Snehal, Raj, Neha, Nupur without whom it would be impossible to imagine the life in Raleigh. A very special thanks to Harleen for innumerable fun conversations and teaching me the skill of prioritization. I received a precious support and spent wonderful time with my colleagues and friends – Ankit, Sugandha, Joomyung, Nick, Sangwoo, Parth, Alexis, Jimmy, Abraham, Guang-Yu, Shahil, Serena, Ashton, Chandramauli, Hemant, Mayank, Bhavya, Gaurav, Ankith and Arpit in Raleigh and Arshita, Shashank, Hrishikesh, Priyanka, Deepak, Anjali and Shashank (Kalra) in Los Angeles. Especial thanks to Arshita and Hrishikesh for filling

our lives with joy during the adverse time of pandemic.

I express a profound gratitude to my parents, in-laws and my brother for providing me with constant support and continuous encouragement during my PhD. My sister in law, Krishna made the last few most challenging weeks of my student life amusing and eventful. The love and support of my better half is beyond words. My accomplishments would not have been possible without the support and love of my family.

Finally, the research was supported by the Center for Nuclear Energy Facilities and Structures (CNEFS) at North Carolina State University. I am greatly indebted to the center for providing me with all the facilities and support throughout my graduate school. I would also like to thank all the staff members of the Civil Engineering department for their help and support extended over the years.

# TABLE OF CONTENTS

<b>LIST OF TABLES</b> . . . . .	<b>ix</b>
<b>LIST OF FIGURES</b> . . . . .	<b>xi</b>
<b>Part I Introduction</b> . . . . .	<b>1</b>
I.1 Introduction . . . . .	2
I.2 Background . . . . .	3
I.3 Research Objectives . . . . .	9
I.4 Proposed Research . . . . .	9
I.5 Organization . . . . .	13
<b>Part II Limitations of Traditional Tools for Beyond Design Basis</b>	
<b>External Hazard PRA</b> . . . . .	<b>15</b>
II.1 Introduction . . . . .	16
II.2 Simple Example Application . . . . .	18
II.3 Example Application to Nuclear Power Plant PRA . . . . .	43
II.4 Conclusions . . . . .	52
<b>Part III Computationally Efficient Approach for Risk-informed</b>	
<b>Decision Making</b> . . . . .	<b>55</b>
III.1 Introduction . . . . .	56
III.2 Traditional Approaches . . . . .	60
III.3 Proposed Algorithm . . . . .	62
III.4 Algorithm for Tree with Independent Nodes . . . . .	65
III.5 Tree with Two Dependent Chains . . . . .	68

III.6 Tree with Multiple Dependent Chains . . . . .	72
III.7 Simple Example Application . . . . .	75
III.8 Complex Example Application . . . . .	79
III.9 Computational Efficiency . . . . .	84
III.10Conclusions . . . . .	88
<b>Part IV    Computationally Efficient Approach for Multi hazard and</b>	
<b>          Multi Unit Probabilistic Risk Assessment . . . . .</b>	<b>90</b>
IV.1 Introduction . . . . .	91
IV.2 Summary of the Existing Framework . . . . .	92
IV.3 Enhancements to the Existing Approach . . . . .	96
IV.4 Nested Loop Structure . . . . .	97
IV.5 Consideration of Common Cause Failures (CCF) . . . . .	112
IV.6 $n$ of the $m$ Input ( $n/m$ ) Logic Gates . . . . .	123
IV.7 Computational Efficiency Compared with Traditional Fault Tree Analysis	
Approaches . . . . .	124
IV.8 Comparison of Proposed Algorithm with other Bayesian Network Algorithms	127
IV.9 Conclusions . . . . .	132
<b>Part V    Understanding Uncertainty Scaling for Flooding PRA based</b>	
<b>          on Experimental Data . . . . .</b>	<b>134</b>
V.1 Introduction . . . . .	135
V.2 Description of Application Case Study . . . . .	137
V.3 Scaling Analysis . . . . .	144
V.4 Summary and Conclusions . . . . .	163

<b>Part VI Summary, Conclusions, and Recommendations for Future</b>	
<b>Research . . . . .</b>	<b>165</b>
VI.1 Summary and Conclusions . . . . .	166
VI.2 Recommendations for Future Research . . . . .	172
<b>References . . . . .</b>	<b>173</b>
<b>Appendix . . . . .</b>	<b>186</b>
Appendix A Fault Tree Files . . . . .	187
A.1 Explanation for the input DATA format . . . . .	188
A.2 Data used in section III.9 . . . . .	190
A.3 Data used in section IV.7 . . . . .	283

## LIST OF TABLES

Table II.1	Comparison of quantification estimates for failure sequence 2 . . . . .	27
Table II.2	Comparison of quantification estimates for failure sequences with unsafe conditions . . . . .	28
Table II.3	Comparison of quantification estimates for failure sequence 2 with CCF events . . . . .	32
Table II.4	Sources for the fault tree diagrams . . . . .	46
Table II.5	Comparison between traditional event tree analysis (ETA) and Bayesian network (BN) approach for DB scenario . . . . .	51
Table II.6	Comparison between traditional event tree analysis (ETA) and Bayesian network (BN) approach for BDB scenario . . . . .	52
Table III.1	Truth Tables for logic gates with two input nodes $A$ and $B$ ( <i>The term 0 and 1 in truth table translate to success and failure in fault and event trees</i> ) . . . . .	65
Table III.2	Probability of failure and success for each leaf node . . . . .	80
Table III.3	Calculation of $k_{1_i}$ and $k_{2_i}$ for complex example with three chains . . . . .	83
Table III.4	Details of the three set of fault tree structure . . . . .	84
Table IV.1	Details of the three set of fault tree structure . . . . .	93
Table IV.2	Probability of failure and success for each basic events . . . . .	105
Table IV.3	Calculation of $k_{1_i}$ and $k_{2_i}$ for $C_3$ and $C_4$ of the inner loop . . . . .	108
Table IV.4	Calculation of $k_{1_i}$ and $k_{2_i}$ for illustrative example with two chains . . . . .	111
Table IV.5	Probability for CCF related basic event failures . . . . .	121
Table IV.6	Calculation of $k'_{1_i}$ and $k'_{2_i}$ for CCF example with three chains . . . . .	122

Table IV.7	Details of the five cases of fault tree structure . . . . .	125
Table IV.8	Summary of computational steps . . . . .	132
Table V.1	Dimension of weir outlet for all the experiments . . . . .	145
Table V.2	Mean and standard deviation of $C_d$ for each experiment . . . . .	147
Table V.3	Change in Non-linear range of $C_d$ vs $h/b$ with $b$ . . . . .	155

## LIST OF FIGURES

Figure II.1	Fault trees for failure of (a) System 1 and (b) System 2 . . . . .	19
Figure II.2	Event tree for simple example . . . . .	20
Figure II.3	Venn Diagrams for: (a) $FT_1^c$ , (b) $FT_2$ and (c) $FT_1^c \cap FT_2$ . . . . .	22
Figure II.4	Fault trees for (a) System 1 (b) Modified System 2 . . . . .	29
Figure II.5	Fault trees for (a) System 1 (b) System 2 . . . . .	30
Figure II.6	Mapped Bayesian network with node states corresponding to the event tree shown in Figure II.2 . . . . .	37
Figure II.7	BN for accident sequence 2 . . . . .	39
Figure II.8	Mapped BN for Failure sequence 2 with common cause failures . . .	43
Figure II.9	Event tree diagram for risk assessment of nuclear power plant . . .	44
Figure II.10	Fault Tree diagram for Auxiliary Feedwater System (AFWS) Failure	47
Figure II.11	Fault Tree diagram for High Pressure Coolant Injection (HPI) Failure	47
Figure II.12	Fault Tree diagram for Emergency Core Cooling System (ECCS) Failure . . . . .	48
Figure II.13	Fault Tree diagram for Residual Heat Removal (RHR) Failure . . .	48
Figure II.14	Mapped Bayesian network corresponding to the event tree shown in Figure II.9 . . . . .	50
Figure III.1	Generic Fault tree and Tree data structure . . . . .	62
Figure III.2	Logic Gates . . . . .	64
Figure III.3	A simple Tree data structure with independent nodes . . . . .	67
Figure III.4	A simple Tree data structure with one dependent node and single loop . . . . .	69

Figure III.5	Illustration of calculation of Chain probabilities . . . . .	70
Figure III.6	A generic Tree data structure with one dependent node and multiple chains . . . . .	73
Figure III.7	Event tree and its corresponding fault trees for simple example . .	75
Figure III.8	Venn diagrams . . . . .	76
Figure III.9	Tree data structure of accident sequence 2 . . . . .	78
Figure III.10	Complex Fault Tree structure . . . . .	80
Figure III.11	Tree data structure for the fault tree shown in Figure III.10 . . . .	82
Figure III.12	Calculation of $z$ for all the chains . . . . .	83
Figure III.13	Comparison of efficiency and accuracy of the proposed algorithm compared to traditional approach . . . . .	86
Figure III.14	A typical fault tree structure used for comparison of computational efficiency . . . . .	87
Figure III.15	Comparison of computational time between traditional and proposed approach for the fault tree structure in Figure III.14 . . .	87
Figure IV.1	Generic Fault tree and Tree data structure . . . . .	93
Figure IV.2	A generic Tree data structure with one dependent node and multiple chains . . . . .	95
Figure IV.3	A generic fault case of loop inside loop in a Tree data structure . .	98
Figure IV.4	Conversion of a loop to independent chain . . . . .	100
Figure IV.5	Conversion of a generic loop inside loop to a Tree data structure with one loop . . . . .	102
Figure IV.6	Conversion of inner loop to independent chain when $K_1$ is equal to 0	103
Figure IV.7	An illustrative example of nested loops in a fault tree structure . .	105

Figure IV.8	Tree data structure for the example of loop in loop . . . . .	107
Figure IV.9	Conversion of Inner loop to independent chain in illustrative example	110
Figure IV.10	Illustration for the inclusion of common cause failures: (a) Fault tree for system failure without CCF, (b) Fault tree for system failure with CCF . . . . .	113
Figure IV.11	A generic case of common cause failures in a Tree data structure .	115
Figure IV.12	A simple fault tree with many common cause failures referred from [95]: (a) Fault tree with three transfer gates, (b) Fault tree connected at transfer gate A, (c) Fault tree connected at transfer gate B and (d) Fault tree connected at transfer gate C . . . . .	120
Figure IV.13	Conversion of $n/m$ logic gate to a set of AND and OR gates. . . .	123
Figure IV.14	Computational efficiency of the proposed algorithm compared to traditional PRA approaches for nested loops . . . . .	126
Figure IV.15	Computational efficiency of the proposed algorithm compared to traditional PRA approaches for CCF . . . . .	127
Figure IV.16	Example from Bensi et al. [30] . . . . .	128
Figure IV.17	Tree data structure for example in Bensi et al. [30] . . . . .	129
Figure IV.18	Equivalent tree with independent nodes for Inner loop . . . . .	131
Figure V.1	Geometry of rectangular weir . . . . .	138
Figure V.2	Plot for $C_d$ vs $h$ for Experiment 1 . . . . .	141
Figure V.3	Plot for $C_d$ vs $h$ for Experiment 2 . . . . .	141
Figure V.4	Plot for $C_d$ vs $h$ for Experiment 3 . . . . .	142
Figure V.5	Plot for $C_d$ vs $h$ for Experiment 4 . . . . .	142
Figure V.6	Plot for $C_d$ vs $h$ for Experiment 5 . . . . .	143

Figure V.7	Plot for $C_d$ vs $h$ for Experiment 6 . . . . .	143
Figure V.8	Plot for $C_d$ vs $h$ for Experiment 7 . . . . .	144
Figure V.9	Plot between $C_d$ vs $h/b$ for all the experiments . . . . .	147
Figure V.10	Biasness observed in the Experiment 1 . . . . .	149
Figure V.11	Plot for $C_d$ vs $h/b$ : (a) $b < 0.06$ and (b) $b > 0.06$ . . . . .	150
Figure V.12	(a) Plot for $C_d$ vs $h/b$ from all the experiments with $b/B > 0.25$ and (b) Idealized shape for $C_d$ vs $h/b$ . . . . .	151
Figure V.13	Plot for parameters for $C_d$ vs $h$ plot: (a) Slope of segment 1 vs $b/B$ , (b) Slope of segment 2 vs $b/B$ , (c) $h/b$ at the lowest point vs $b/B$ and (d) Minimum values of $C_d$ vs $b/B$ . . . . .	152
Figure V.14	Plot of $C_d$ vs $h/b$ for various range of $b$ : (a) $b = 0.02\text{m} - 0.06 \text{ m}$ , (b) $b = 0.2\text{m} - 0.3 \text{ m}$ & (c) $b = 0.3\text{m} - 0.5 \text{ m}$ . . . . .	154
Figure V.15	Plot for standard deviation of $C_d$ with change in width . . . . .	156
Figure V.16	Plot of standard deviation of $C_d$ with width $b$ : (a) All available data for the experiments and (b) Plot after removal of data impacted by boundary effect . . . . .	158
Figure V.17	Mean of $C_d$ with increase in width . . . . .	159
Figure V.18	Fitting the Lognormal and Beta distribution for $C_d$ . . . . .	160
Figure V.19	Parameters for fitted Beta distribution for $C_d$ . . . . .	160
Figure V.20	Plot of mean and standard deviation of $C_d$ with (a) water height $h$ , (b) water discharge $Q$ and (c) height and width ration $h/b$ . . . . .	162
Figure A.1	Data format for Fault tree structure . . . . .	188
Figure A.2	Fault tree structure for the given Data format . . . . .	189
Figure A.3	Data format for Probabilities of the component failures . . . . .	190

---

---

# PART I

---

## Introduction

## I.1 Introduction

Uncertainty quantification is a science to increase the credibility of computational results obtained by Modeling and Simulation (M&S) tools. It is performed to identify and characterize the factors in M&S that affect the accuracy of the conceptual and mathematical models. Uncertainties are commonly addressed in three simple steps of M&S: (a) construction of the conceptual mechanics based model, (b) formulation of the mathematical model, and (c) computation of the simulation results. Some common sources of uncertainties lie in the assumptions, mathematical form of conceptual or mathematical models, initial boundary conditions of the governing equations or the parameters chosen in the mathematical model. These uncertainties are propagated from conceptual model to simulation results in a step by step process, which means they are mathematically mapped from sources to simulation results.

In recent years, there is an increased emphasis on having a formal quantification of risk due to external hazards while also accounting for uncertainties associated with these hazards a plant's response to the hazards. An enhanced understanding of the propagation of uncertainties from one state to another state, i.e. component level to system level or small scale level to large scale level is a key aspect that could help mitigate undesired events at nuclear plants. This research is intended to address the following four aspects of uncertainty quantification in external hazard probabilistic risk assessment (PRA),

- Limitations of traditional tools for beyond design basis external hazard PRA
- Computationally efficient approach for risk-informed decision making
- Computationally efficient approach for multi hazard and multi unit probabilistic risk assessment

- Understanding uncertainty scaling for flooding PRA based on experimental data

## **I.2 Background**

### **I.2.1 Limitations of Traditional Tools for Beyond Design Basis External Hazard PRA**

Probabilistic risk assessment is being used increasingly by the nuclear industry for safety during normal operations as well as for safeguards against external hazards. Computation of total risk in an external hazard PRA is dependent on hazard assessment, fragility assessment, and systems analysis [1, 2, 3]. A systems analysis for the propagation of component fragilities is conducted using fault and event trees [4, 5, 6]. The fault and event trees for an actual power plant system can be fairly large in size. Historically, this has imposed computational challenges on the propagation of basic event failure probabilities through the fault and event trees for all the possible accident sequences [7]. Such computational challenges have been addressed by considering the component failure probabilities to be small as well as statistically independent [5, 8]. These two assumptions seem appropriate for the failures of nuclear power plant (NPP) components during design basis (DB) events, since the failure probabilities of components subjected to DB accident loads are significantly small.

The Fukushima-Daiichi nuclear accident has led to several global initiatives for evaluating the risk of nuclear plant operations when subjected to beyond design basis (BDB) events. The assumptions related to small failure probability that are usually valid for DB are not necessarily valid for BDB scenarios. In fact, several studies have shown that these can be about two orders of magnitude higher in most cases [9, 10, 11]. The estimation of relative importance of accident sequences and the determination of the

critical path are dependent solely on the quantification approach used for event and fault tree analysis. Therefore, the assumptions made in the traditional PRA tools have the greatest impact on the assessment of the accident sequences and the critical path.

In addition to the calculation of total risk for core damage or early release of radioactivity, PRA is increasingly being used for decision making during normal operations as well as for the development of accident management guidelines [3]. Nuclear power plants are increasingly considering incorporation of autonomous tools as the reliance on the technological advances increases. In the long run, autonomous systems will gradually decrease a reliance on humans during operations and hence pose an increased risk of dependent failures due to common cause failure (CCF) events [12]. Common cause failures pose significant importance in PRA because their presence can lead to occurrence of multiple failure events together thereby leading to a increasing the chances of a system failure. Hence, the presence of common cause failure events in the context of BDB requires a careful reassessment of the assumption regarding statistical independence of component failures.

### **I.2.2 Computationally Efficient Approach for Risk-informed Decision Making**

The fault and event trees play a crucial role in system analysis by logically combining the component level fragilities, and convoluting the fragilities with the hazard curve [4, 5, 6]. The events in fault and event trees are modeled using binary states: failure or not a failure (success) [13]. The fault and event trees for an actual power plant unit consist of several different types of logic gates, interconnected events, and dependent events, etc. Such complex connectivities can give rise to excessive computational demand and storage

requirement for the analysis [14]. An assessment of logic trees with  $n$  events requires an analysis of a maximum of  $2^n$  scenarios emerging from the various combinations of event failures. Hence, a complete analysis of combined fault and event tree network demands computational complexity on the order of  $2^n$  in either time or space. The exponential order of complexity is required mainly to address the dependencies and dependent failure events [15].

Even with the advancements in available computational power, it is impractical to solve real-world problems using algorithms for which the computational efforts increases exponentially with the increase in network size. The exponential complexity has been a major roadblock for performing the risk assessment accurately for a very long time. There are several traditional methods that reduce the complexity but at the expense of accuracy [15]. In order to reduce the computational demands of the fault and event tree networks, several assumptions or approximations are often relied upon.

Fault and event tree analysis methods include mocus algorithm, modularization [16, 17], binary decision diagram techniques [18, 19, 20], parallel computing processes [21, 22, 23, 24, 25], and Bayesian networks [26, 27, 28, 29, 30]. However, all the available techniques suffer from some type of computational deficiencies as each technique has a trade off between storage and computational time. Therefore, there is a need to develop a new approach which does not consume excessive storage or computational time while improving overall computational efficiency and yielding accurate results.

### **I.2.3 Computationally Efficient Approach for Multi hazard and Multi Unit Probabilistic Risk Assessment**

The accident at Fukushima Daiichi has highlighted the importance of risk from external hazards associated with multiple nuclear reactor units and multiple hazards. As a result, nuclear research community has given considerable attention to multi-unit and multi-hazard probabilistic risk assessment (PRA) in the past few years [31, 32, 33]. For example, seismically induced tsunami or hurricane induced wind and flooding scenarios have been studied quite widely. A multi-unit or multi-hazard PRA computes measures of risk and identifies the contributors to risk by conducting PRA for the entire site including multiple units. In such a case, any possible unit-to-unit interactions and dependencies should be modelled and accounted for. Therefore, the assessment of entire site for multi-hazard leads to highly complex dependencies among the component failures.

Prior to such studies, PRAs for individual hazards were conducted in isolation of each other. The fault and event trees employed for a conventional probabilistic risk assessment of nuclear power plants consist of hundreds of components failures. Even for conventional single hazard PRA, the traditional fault and event tree analysis approaches rely on assumptions, approximations or high-performance computing resources that allow implementations of parallel computing. Consideration of multi-hazard scenarios results in a single dependent initiating event which links the fault and event tree networks for individual hazards thereby resulting in a very large network for combined hazards and making the traditional approaches impractical for such computations [34].

The need for improvement in computational efficiency of PRAs is also desired in the development of new technologies such as the development of Digital Twins for

autonomous control of advanced small and micro reactors. Such advanced autonomous systems have to rely on risk-informed decision making in real-time which requires running hundreds of scenarios in a relatively short amount of time to evaluate the best strategy for controlling the plant. Therefore, complex dependencies exhibited in the assessment of multi-unit and multi-hazard PRAs needs to be addressed.

#### **I.2.4 Understanding Uncertainty Scaling for Flooding PRA based on Experimental Data**

Nuclear industry has undertaken several studies to ensure safety against flooding in power plants after the tsunami induced flooding initiated the accident at Fukushima Daiichi power station in 2011 [35, 36]. A large number of such studies rely on advanced flooding simulation tools [37]. Simulation of catastrophic flooding events often suffers from a lack of confidence in the predictions due to inherent randomness, lack of knowledge about the physics of the complex interactions and the associated uncertainties. Some recent studies have focused on the development of a consistent methodology for verification and validation of these advanced simulation tools [38, 39, 40]. Yet, such validation studies are quite challenging due to a lack of real-world data.

Advanced simulation tools are validated by comparing its results with data from laboratory experiments. The data from laboratory experiments is also used for uncertainty quantification [41, 42]. However, it must be noted that a certain degree of validation based on laboratory experiments does not necessarily translate into the same degree of validation at real-world scale. It is difficult to provide an explicit proof of this comparison due to a lack of usable quantifiable data at a real-world scale. Therefore, a comprehensive knowledge of the scaling phenomena is necessary.

An improved understanding of the scaling phenomena would help to reduce the gap between a real-world application at the plant level and a test facility at a smaller scale. An implicit extrapolation of the data is performed when a simulation tool is validated through a smaller scale facility in a laboratory for studying a real-world application. This could lead to scale distortion and greater uncertainty in predictions [43, 44]. Lack of available quantifiable data at full scale, and lack of understanding about the applicability of data obtained at a reduced scale leads to a residual epistemic uncertainty in a simulation model.

Traditionally, scaling phenomena is addressed by an analysis of complex flooding processes through a dimensionless analysis of models at a smaller scale. It involves the analysis of various fluid dynamics phenomena such as mass transfer among different phases, the interfacial tension, diffusion, dispersion, absorption, etc. A physical simulation of all these quantities leads to a large number of dimensionless parameters that cannot be satisfied simultaneously between a real-world prototype and a small-scale model. Either geometric similarity or the physical similarity cannot be attained in such models. Moreover, comprehending the scale of flooding at nuclear power plants beforehand is also difficult. Presently, this inadequacy is addressed through expert opinion and professional judgment because it is impractical to design a full-scale experiment for each plant. Contemplating the scaling phenomena by considering the data at many different and yet smaller scales can increase the understanding of scaling effects. Therefore, it is necessary to characterize the scaling pattern by collecting data from small to medium scale experiments and then analyze the uncertainties propagation from smaller scale to larger scale applications.

### **I.3 Research Objectives**

The primary objective of this research is to assess the impact of uncertainties in the risk assessment of various systems, structures, and components (SSCs) of nuclear power plants during external hazards such as earthquakes and flooding. It is also directed towards achieving an improved solution to reduce uncertainties wherever possible. The proposed research aims to characterize the impact of uncertainties by identifying sources of unaccounted modelling parameters in mathematical formulation of traditional approaches, developing and suggesting improved and efficient ways for PRA modelling and propagating uncertainties at different scales.

### **I.4 Proposed Research**

The four distinct aspects of the proposed research and the specific tasks needed to achieve the objective of this study are described in detail below.

#### **I.4.1 Limitations of Traditional Tools for Beyond Design Basis External Hazard PRA**

In this research, a simple and a relatively more complex illustrative examples are used to show the limitations of the assumptions in traditional approaches for numerical quantification of risk for the case of BDB events. Case studies with CCF events across multiple fault trees are also presented to illustrate the effect of related assumptions when traditional approach is used in BDB risk assessment. It is shown that the assumptions are valid for the case of DB conditions but may lead to excessively conservative risk estimates in the case of BDB conditions. A Bayesian network based top-down algorithm is proposed as an alternative tool for accurate numerical

quantification of total risk in systems analysis. The specific tasks undertaken to conduct this research are:

- Study the implementation of the traditional PRA approach for various scenarios (i.e. DB or BDB).
- Outline the assumptions and the corresponding approximations to overcome the computational limitations of the traditional PRA tools.
- Determine the impact of simplifications on risk assessment for the BDB scenarios which have several instances of common cause failures.
- Illustrate the impact of the key assumptions on the identification of critical accident sequences.
- Determine solutions that could eliminate the uncertainties in quantification due to inapplicability of assumptions in BDB and CCF scenarios in order to improve the PRA quantification.
- Demonstrate an alternative Bayesian network-based framework to eliminate the sources of errors in traditional PRA approaches.
- Illustrate the importance of the proposed solution using a realistic but a relatively simple accident scenario at a nuclear power plant.

#### **I.4.2 Computationally Efficient Approach for Risk-informed Decision Making**

This study focuses on developing and proposing a new approach to address computational time and storage demand issues while maintaining accuracy without incorporating assumptions and approximations. More specifically, an attempt is made to reduce the complexity for the analysis of multiple occurring events (MOEs) and

dependent events in fault and event trees. In this framework, a bottom-up approach is proposed for analyzing fault tree/event tree networks with a dependent node and multiple chains connecting the dependent node. Its implementation shows computationally efficient results compared to existing implementations which makes it highly promising for additional development. The computational efficiency of the proposed approach over traditional approaches is illustrated by comparison for several different examples. The specific tasks required to achieve the objectives of this research are summarized below:

- Outline the commonly implemented methods for the analysis of fault and event trees and the complexities associated with them.
- Identify the key sources of computational complexity and provide solution to address those sources of complexities.
- Propose and develop a new approach to improve computational efficiency for analyzing fault and event trees.
- Incorporate the analysis of multiple occurring events and dependent events without causing an exponential complexity.
- Compare the computational efficiency of the proposed approach with traditional methods to highlight the importance of the proposed approach.

### **I.4.3 Computationally Efficient Approach for Multi hazard and Multi Unit Probabilistic Risk Assessment**

In this study, major enhancements to the framework proposed in Section I.4.2 are presented to address complex dependencies exhibited in the assessment of multi-unit and multi-hazard PRAs. More specifically, the proposed enhancements consider

multiple dependent nodes, common cause failure events,  $n/m$  logic gates, and nested loops that can exist when one initiating event leads to more than one failure paths. The computational efficiency of the proposed modifications are illustrated through applications examples. The specific tasks undertaken to conduct this research are:

- Identify the fault and event tree network complexities in multi-hazard and multi-unit scenario.
- Extend the proposed computationally efficient algorithm for such scenarios.
- Propose enhancements for the analysis of CCF in the multi-hazard or multi unit accident scenarios.
- Incorporate enhancements for other commonly observed logic gates in the fault and event trees.
- Compare the computational demand of the proposed and traditional approaches to illustrate the benefits of the proposed approach.

#### **I.4.4 Understanding Uncertainty Scaling for Flooding PRA based on Experimental Data**

In this research, the scaling effect is studied and illustrated by a categorization of available experimental data at small, medium and large scale. Subsequently, uncertainties in the data at different scales are characterized and quantified to allow logical extrapolation for real life applications. Such an improved understanding can reduce the reliance on heuristic approaches. An increased awareness of the scaling phenomena can help guide future experimental studies to focus on collection of data at appropriate scales and in effect enhance the performance of simulation codes. The resulting reduction in scale distortion will enhance our ability to estimate flooding risks more realistically. The

specific tasks required to achieve the objectives of this research are summarized below:

- Understand the process of verification and validation of advanced simulation tools for increasing the confidence in simulation prediction in the context of underlying uncertainties.
- Delineate the importance of real-world data in developing a consistent methodology for verification and validation.
- Characterize scaling of uncertainties by analyzing illustrative flooding models representing accident scenarios at NPPs at small to medium scale.
- Demonstrate the effect of scaling on the dimensional parameters of the illustrative example.
- Identify the factors that have a large impact on uncertainties at different scales of the experiment.
- Analyze data from 7 experiments that illustrates the flooding scenario and perform uncertainty quantification with respect to all the scaled parameters of interest.
- Illustrate the impact of uncertainties on the probabilistic distribution of the key modeling parameters in the illustrative example.

## **I.5 Organization**

This dissertation primarily consists of six sections. Section I gives an introduction to the problem being studied followed by a discussion on the objective of the research. The second part of the dissertation presents limitations of traditional tools for beyond design basis external hazard PRA. The manuscript provided in section II is published in Journal of Nuclear Engineering and Design. The third section focuses on the computationally efficient approach for risk-informed decision making. In the fourth section, the proposed

algorithm in section III is extended for developing computationally efficient approach for multi-hazard and multi-unit probabilistic risk assessment. In the fifth section, an effort is made to understand uncertainty scaling for flooding PRA based on experimental data. Finally, the sixth section of this dissertation presents a summary and conclusions of the work that has been discussed in sections II to V. It also proposes recommendations for future work. The work presented in section III to V are copies of manuscripts that will be submitted for publication to international journals.

---

---

## PART II

---

### Limitations of Traditional Tools for Beyond Design Basis External Hazard PRA

## II.1 Introduction

The nuclear industry has increased its reliance on probabilistic risk assessment tools for nuclear power plant design, operation, life extension, and regulation. Probabilistic safety assessment (PSA) or probabilistic risk assessment (PRA) evaluates the risk associated with a specific hazard by a convolution of system fragility and hazard curve. The fragility curve of structures, systems, and components (SSCs) is expressed as the conditional probability of failure for a given hazard and is a function of the uncertainties in empirical, experimental, and/or numerical data for the available physical models of the SSCs. US Nuclear Regulatory Commission (USNRC) and International Atomic Energy Agency (IAEA) have issued guidelines for conducting a full scope PRA [1, 2, 3, 45, 46], where the plant level risk is calculated by combining the component and the subsystem fragility curves through a systems analysis. Typically, fault and event trees [4, 5] are used for conducting the systems analysis by logically combining the component level fragilities and for performing the convolution with a hazard curve.

The fault and event trees for an actual power plant system can be fairly large in size. Historically, this has imposed a computational challenge on the propagation of basic event failure probabilities through the fault and event trees for all the possible accident sequences [7]. The computational challenges have been addressed by incorporating certain assumptions such as the component failure probabilities are small as well as statistically independent. These two assumptions are more likely to be correct in the context of design basis events compared to beyond design basis events. The failure probabilities of components subjected to design basis loads in nuclear plants are significantly small. Furthermore, the probabilities associated with intersection of

two events (component failures) are even smaller by several orders of magnitude which allows the intersection probabilities to be considered numerically negligible.

The Fukushima-Daiichi nuclear accident has led to several global initiatives which focus on evaluating the risk for nuclear power plant operations when subjected to beyond design basis (BDB) scenarios. The assumptions related to small failure probability that are usually valid for DB may not be valid for BDB scenarios. Several studies have also shown that component failure probabilities can be higher than two orders of magnitude for BDB events in most cases [9, 10, 11].

From a design perspective, an increase in the relative values of probabilities is not significant especially because the frequency of occurrence of a BDB hazard is lower compared to DB and the overall effect on the total risk is not significant. However, the estimation of relative importance of accident sequences and the determination of the critical path are independent of the frequency of occurrence of hazards. These are dependent solely on the quantification approach used for event and fault tree analysis. Therefore, the quantification approaches and related approximations directly impact the assessment of these accident sequences and the critical path.

In addition to the calculation of total risk for core damage or early release of radioactivity, PRA is also used for decision making during normal operation as well as for development of accident management guidelines [3]. The usual operations at nuclear power plants are incorporating automation elements as the reliance on technological advances increases. These automations will gradually decrease human involvement during operations and hence pose a greater risk of dependent failures due to common cause failure (CCF) events [12]. CCF possess significant importance in PRA because their presence

can lead to failure of multiple components together. Hence, the treatment of common cause failure events has been reassessed in the context of BDB scenario in this study.

In this study, we present an illustrative case study using a synthetic simple example to outline the two key assumptions that are made in traditional PRA tools and evaluate their impact on the determination of accident sequence probabilities and the critical accident sequence. Simplicity of the illustrative example allows the use of fundamental set theory to calculate the exact probability of failures without making any assumption. The numerical differences between the risk estimates from traditional approaches and exact quantification methods for BDB and CCF events suggest a scope for improvement in the traditional quantification approaches. A Bayesian framework is proposed as a tool in this study to overcome the limitations in the traditional PRA approaches. The event tree and fault trees are mapped into a Bayesian network (BN) to illustrate the mapping algorithm and to demonstrate the capability of BN methodology in quantifying the probabilities of failure and accident sequences.

## **II.2 Simple Example Application**

### **II.2.1 Description of Simple Example**

A simple example is taken to: (1) illustrate the approach used in traditional PRA tools, (2) outline their key assumptions, (3) describe the corresponding simplifications, (4) compare the results to those from exact computation of risk, and (5) illustrate the differences in risk assessment when the underlying assumptions are not valid in the case of BDB calculations. Figure A.3 shows fault trees of two simple systems that are connected through an event tree to evaluate the complete plant failure. As shown in Figure A.3(a), failure of System 1 is dependent upon failure of two components  $A$  and

$B$  that are connected through an OR gate. Therefore, the System 1 would fail if either of the two components fail. The System 2 is dependent upon three components  $A$ ,  $B$ , and  $C$  as shown in Figure A.3(b). In order for System 2 to fail, either component  $C$  fails or both components  $A$  and  $B$  fail. Failure of components  $A$ ,  $B$ , and  $C$  are representative of basic events in a traditional risk assessment approach. The conditional probabilities of failure (often termed as fragilities) for these basic events given an initiating event can be denoted by  $P_A$ ,  $P_B$ , and  $P_C$ , respectively. The events  $FT_1$  and  $FT_2$  that represent the failure of System 1 and System 2, respectively, are typically referred to as top events and are linked in an event tree diagram to determine the various failure sequences that can lead to the failure of entire plant. The probability of failure associated with  $FT_1$  and  $FT_2$  can be denoted by  $PF_{S1}$  and  $PF_{S2}$ , respectively.

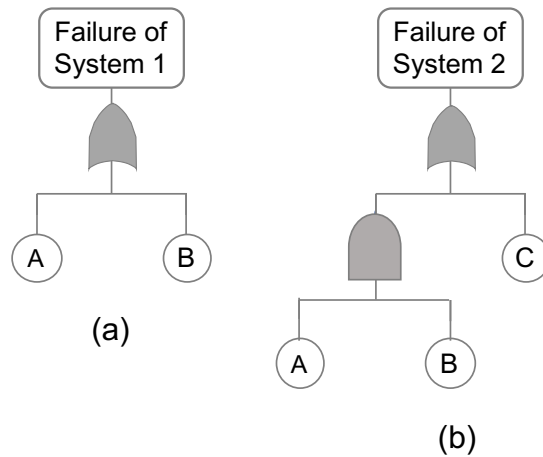


Figure II.1: Fault trees for failure of (a) System 1 and (b) System 2

The event tree for the example considered in this study is shown in Figure II.2. An event tree diagram starts with an initiating event ( $IE$ ) that induces failures of various components. The failure of components could result in failure of System 1 or System 2 or both. Together, these failures (or success of various systems) are connected in series in

the event tree diagram thereby resulting in various accident sequences. For the particular example considered, a total of 3 accident sequences exist as shown in Figure II.2. Two of these sequences (numbered 2 and 3) lead to unsafe conditions at the plant whereas it is potentially possible to maintain the plant in a safe state for sequence 1. It can be seen that the accident sequence 2 occurs when System 1 does not fail (termed as “success” event or complement of event  $FT_1$  that is denoted by  $FT_1^c$  with a probability of success being  $(1 - PF_{S1})$ ) but System 2 fails. Similarly, accident sequence 3 occurs when System 1 fails. For risk assessment, it would be important to calculate the risk associated with both the accident sequences, i.e. sequence 2 and sequence 3.

The fault and event trees described above are used to examine the traditional approach for risk assessment in order to outline the underlying assumptions and the possible limitations in applicability of traditional approach to the case of BDB studies. For this purpose, it is important to first understand the calculation of exact probabilities of failures for a given accident sequence. For brevity, accident sequence 2 is considered for a detailed illustration and discussion in the following sections.

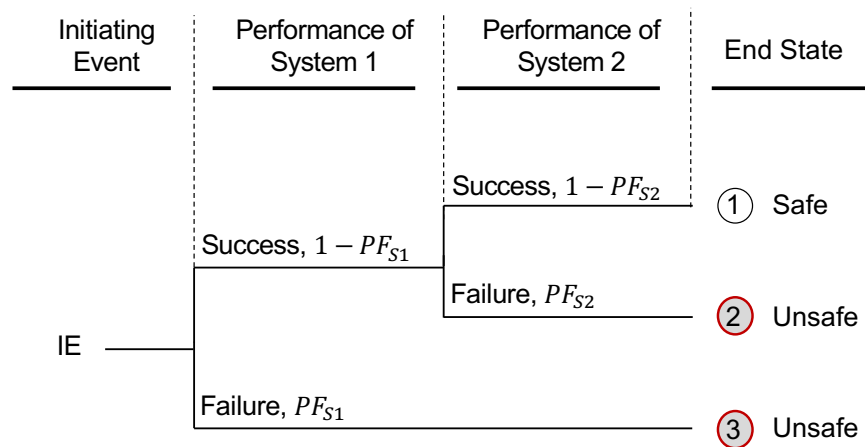


Figure II.2: Event tree for simple example

## II.2.2 Exact Probability of Failure

Exact probabilities of failures can be calculated by using the fundamental concepts of set theory. Venn diagrams can be used to illustrate the expressions for various terms. In general, a failure sequence  $k$  would contain a total of  $n_k$  top events which can be decomposed into  $m_k$  top events that exhibit “success” and  $(n_k - m_k)$  top events that exhibit “failure”. The total probability of failure for a  $k^{th}$  sequence can be expressed as follows:

$$P_{Seq\ k} = \bigcap_{i=1}^{m_k} P(FT_i)^c \bigcap_{j=m_k+1}^{n_k} P(FT_j) \quad (II.1)$$

where,  $P(FT_i)$  represents the failure of  $i^{th}$  top event and superscript  $c$  denotes the complement of an event. In the specific case of simple example described above with two fault trees, Eq. (II.1) for accident sequence 2 shown in Figure II.2 simplifies as follows:

$$P_{Seq\ 2} = P(FT_1)^c \cap P(FT_2) \quad (II.2)$$

Calculation of events  $FT_1$  and  $FT_2$  and their probabilities in terms of basic events  $A$ ,  $B$ , and  $C$  can be further illustrated through the use of Venn diagrams. Figure II.3 shows the Venn diagrams for  $FT_1^c$ ,  $FT_2$ , and  $FT_1^c \cap FT_2$ . The expression for Eq. (II.2) corresponding to accident sequence 2 can be written in terms of basic events using set theory as:

$$Seq\ 2 = (A \cup B)^c \cap ((A \cap B) \cup C) \quad (II.3)$$

If the basic events  $A$ ,  $B$ , and  $C$  are considered to be statistically independent, the intersection probabilities can be calculated simply as multiplication of individual event

probabilities. If the probabilities of failure for events  $A$ ,  $B$ , and  $C$  are denoted by  $P_A$ ,  $P_B$ , and  $P_C$ , then the total probability of failure or risk associated with accident sequence 2 can be calculated as:

$$\begin{aligned}
 P_{Seq\ 2} &= P(C) - P(C \cap A) - P(C \cap B) + P(A \cap B \cap C) \\
 &= P_C - P_A P_C - P_B P_C + P_A P_B P_C
 \end{aligned}
 \tag{II.4}$$

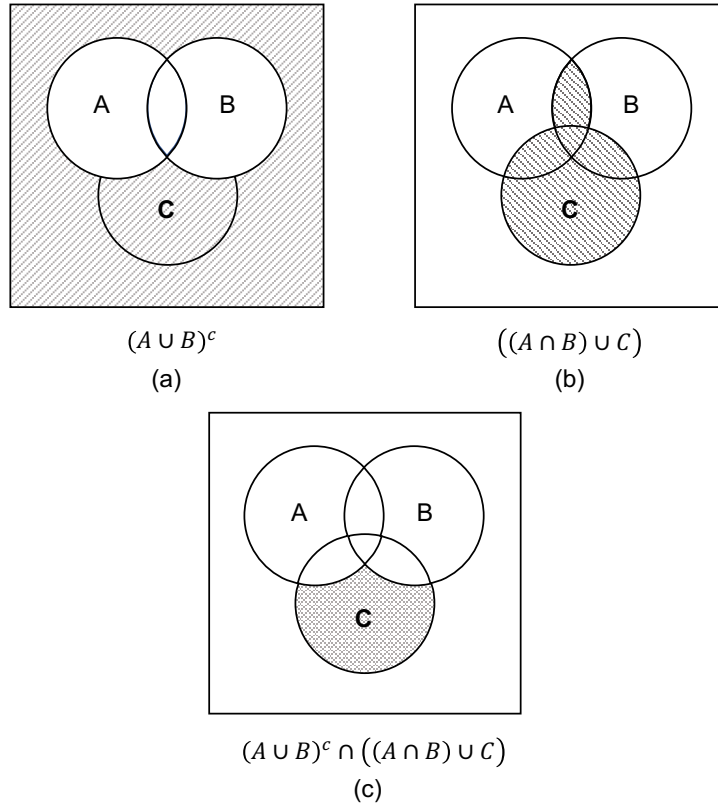


Figure II.3: Venn Diagrams for: (a)  $FT_1^c$ , (b)  $FT_2$  and (c)  $FT_1^c \cap FT_2$

### II.2.3 Traditional Approach for Risk Assessment

Initial implementations [47, 48] of event tree analysis in PRA tools relied on a very simplistic approach wherein all the top events in an accident sequence were assumed to be statistically independent. Consequently, the probability of failure for a particular accident sequence can be calculated relatively easily by a multiplication of probabilities (failure or success) for each top event in the sequence. In the case of simple example considered in this study, probability of failure for accident sequence 2 will be given by  $PF_{S2}(1 - PF_{S1})$ . However, this assumption is not appropriate particularly in the case of nuclear power plant PRA because the various top events in the event tree are usually not statistically independent. For example, the top events in accident sequence 2 of the simple example are not independent because the System 1 and System 2 both share common basic events  $A$  and  $B$  between them.

To overcome the deficiencies of the simplistic approach in initial implementations of PRA tools, a relatively more robust approach based on the concept of minimal cut sets has been used and implemented [49]. Minimal cut sets for the failure of a system is a list of sets such that the failure of even a single set in the list would lead to the failure of the corresponding system. It is important to note in this context that the list of cut sets is developed such that none of the cut set is a subset of another cut set. This is so because if  $A \subseteq B$ , then  $A \cup B = B$ . Hence, if such a subset exists, then it is eliminated from the list and not included in the final list of minimal cut sets. A Boolean reduction process is used to calculate the list of minimal cut sets for a system failure. A detailed discussion on Boolean reduction can be found in Akinode et al. [50] and McKelvin and Lee [51]. The use of minimal cut set approach for quantifying the probability of failure

for an accident sequence is more robust because this approach is directly based on the basic events or component level information. In this approach, Eq. (II.1) is re-written using De Morgan's law as follows:

$$\begin{aligned}
P_{Seq\ k} &= \bigcap_{i=1}^{m_k} P(FT_i)^c \bigcap_{j=m_k+1}^{n_k} P(FT_j) \\
&= \left( \bigcup_{i=1}^{m_k} P(FT_i) \right)^c \bigcap_{j=m_k+1}^{n_k} P(FT_j)
\end{aligned} \tag{II.5}$$

The above equation for  $k^{th}$  accident sequence in the event tree can be envisioned as fault trees of  $m_k$  "success" events connected through an OR gate and fault trees of  $(n_k - m_k)$  "failure" events connected through an AND gate. Let the success related fault trees be connected through an OR gate yield  $ns_k$  minimal cut sets. Similarly, the failure related fault trees connected through AND gate yield  $nf_k$  minimal cut sets. Hence, Eq. (II.5) can be re-written in terms of minimal cut set as,

$$P_{Seq\ k} = \left( \bigcup_{i=1}^{ns_k} P(CS_{si}) \right)^c \cap \left( \bigcup_{j=1}^{nf_k} P(CS_{fj}) \right) \tag{II.6}$$

where,  $CS$  denotes cut set. Subscript  $s$  and  $f$  are used to show *success* event cut set and *failure* event cut set, respectively. Once again, invoking De Morgan's law, one can write Eq. (II.6) as:

$$P_{Seq\ k} = \left( \bigcap_{i=1}^{ns_k} P(CS_{si})^c \right) \cap \left( \bigcup_{j=1}^{nf_k} P(CS_{fj}) \right) \tag{II.7}$$

While the above equation relies heavily on complement of cut sets for success

events, a key shortcoming of the minimal cut set approach is that the use of complemented events significantly slows the Boolean reduction process in PRA tools [52]. Hence, two specific assumptions are made to facilitate a computationally efficient implementation in PRA tools.

- First, it is assumed that the probability of failure for each component is very small.
- Second, the cut sets are assumed to be independent of each other.

Moreover, event and fault trees cannot handle non-binary relations between intermediate and basic events. These assumptions are reasonable in the context of PRA for design basis events in a nuclear facility. Based on the above assumptions, we can write  $P(CS_i) \ll 1$ . Hence  $(1 - P(CS_i)) \simeq 1$  or  $CS_i^c = \Omega$ , where  $CS_i$  represents cut set  $i$  and  $\Omega$  denotes a sample set. Hence, Eq. (II.7) can be reduced to:

$$\begin{aligned}
 P_{Seq\ k} &\simeq \left( \bigcap_{i=1}^{ns_k} P(\Omega) \right) \cap \left( \bigcup_{j=1}^{nf_k} P(CS_{fj}) \right) \\
 &\simeq \bigcup_{j=1}^{nf_k} P(CS_{fj})
 \end{aligned} \tag{II.8}$$

However, Eq. (II.8) will not be valid if some common cut sets exist between success and failure event cut sets. For example, if  $CS_{s1}$  and  $CS_{f1}$  are the same events, then  $(CS_{s1}^c \cap CS_{f1}) = \phi$ , i.e. null. In such a case, Eq. (II.7) will be reduced to:

$$\begin{aligned}
 P_{Seq\ k} &\simeq P(CS_{s1}^c \cap CS_{f1}) \cup \left( \bigcup_{j=2}^{nf_k} P(\Omega) \cap P(CS_{fj}) \right) \\
 &\simeq P(\phi) \cup \left( \bigcup_{j=2}^{nf_k} P(CS_{fj}) \right)
 \end{aligned} \tag{II.9}$$

Hence, if there are some common cut sets between *success* and *failure* events then those must be removed from the final list of cut sets for the accident sequence. In the case of simple example described in Figure A.3 and Figure II.2, the accident sequence 2 consists of the success of System 1 and failure of System 2. For the fault trees in Figure A.3, a Boolean reduction leads to the minimal cut sets for the failure of System 1 as  $A$ ,  $B$  and the minimal cut sets for System 2 as  $A \cap B$ ,  $C$ . For the success of System 1,  $A$  and  $B$  both must not fail which also indicates that  $A \cap B$  will not fail too. Hence,  $A \cap B$  is a common cut set between failure of System 2 and success of System 1 along the accident sequence 2. It will be removed from the list of the minimal cut set for the accident sequence 2. Consequently, the final cut set for accident sequence 2 is  $C$ . Hence, the probability of accident sequence 2 based on the PRA approach can be calculated simply as:

$$P_{Seq\ 2} \simeq P_C \quad (II.10)$$

Clearly, the expressions in Eq. (II.4) and Eq. (II.10) for the failure probability of sequence 2 are quite different. The effect of this difference is explored further by calculating the probabilities for design basis (DB) and beyond design basis (BDB) scenarios. For illustration purposes, the point estimate of component failure probability is considered instead of fragility curves to main the simplicity and brevity of the example. The failure probabilities of components  $A$ ,  $B$ , and  $C$  are taken based on the guidance from literature for DB and BDB related studies [10, 11] and are given in Table II.1. Here, the main focus is on the relative magnitude differences between the DB and BDB failure probabilities. The order of magnitude for BDB values is about 2 times more than that of the DB values. Therefore, in this study, we have considered the failure probabilities of BDB and DB events to be similar order of magnitudes as

mentioned in the related studies but not exactly the same numbers. The proposed methodology and the primary observations would not change if slightly different values are adopted for failure probabilities of components. The failure probability of the accident sequence 2 is quantified using Eq. (II.10) which represents the traditional approach and Eq. (II.4) which represents the exact quantification. Table II.1 compares the probabilities for both the scenarios and illustrates the differences that exist in using the traditional analysis for a BDB scenario.

Table II.1: Comparison of quantification estimates for failure sequence 2

	$P_A$	$P_B$	$P_C$	Traditional Eq. (II.10)	Exact Eq. (II.4)	Error (% $\epsilon$ )
DB	1.25E-3	8E-4	2E-3	0.0020	0.001996	0.21
BDB	0.125	0.08	0.20	0.2000	0.1610	24.34

The differences presented in Table II.1 are more significant in the context of determining the critical paths for a particular BDB scenario. Identification of an appropriate critical path is important because an incorrect estimate can result in incorrect decisions and operator actions. For the particular example discussed above, Table II.2 gives probabilities of failures for the two accident sequences of BDB scenario as calculated using the traditional and exact approaches. The failure probability for accident sequence 3 is calculated using the methodology illustrated above and is given by following equation:

$$P_{Seq\ 3} = P_A + P_B - P_A \times P_B \quad (II.11)$$

Since, the accident sequence 3 only consists of one top event, the expression from traditional and exact approach are exactly same. As shown in Table II.2, a traditional approach yields accident sequence 2 as the critical path whereas the exact approach yields accident sequence 3 to be critical. The analysis indicates that the traditional PRA approaches for risk assessment may lead to inappropriate results in BDB scenarios.

Table II.2: Comparison of quantification estimates for failure sequences with unsafe conditions

Quantification Method	Seq. 2 $P_f$	Seq. 3 $P_f$	Critical Sequence
Traditional, (Eq. (II.10), Eq. (II.11))	0.20	0.195	Seq. 2
Exact, (Eq. (II.4), Eq. (II.11))	0.161	0.195	Seq. 3

#### II.2.4 Consideration of Common Cause Failures

The discussion in the section above depicts the issues associated with numerical estimation of total risk and appropriate critical path for BDB scenario due to the assumptions that the probabilities of failure for basic events and the intersection of basic events are sufficiently small. Another quantification difficulty exists in the application of traditional PRA approach due to the assumption regarding statistical independence of basic events. In a nuclear power plant PRA, it is often the case that different components or systems may fail due to a common cause and are therefore not statistically independent. The condition wherein the basic events are not statistically independent is typically referred to by “common cause failure (CCF)”. The problem associated with CCF and its impact has been studied by many researchers [53, 54, 55]. The issues illustrated in previous section are also relevant when various fault trees in an event tree share common cause failures and therefore increases the quantification errors

in BDB scenarios.

The implementation of CCF in traditional PRA tools and their impact on risk for beyond design basis condition is illustrated by a slightly modified version of the simple example considered earlier in this study as shown in Figure II.4. The fault tree for the System 1 is unchanged whereas the fault tree of System 2 is modified by replacing common events  $A$  and  $B$  by new events  $D$  and  $E$ , i.e. there are no common events in the two systems. In the context of CCF, it is considered that events  $A$  and  $D$  fail due to a common cause, which introduces a dependency between the fault trees of System 1 and System 2. The events impacted by a common cause event form a common cause component group (CCCG). It must be noted that CCF implies that events  $A$  and  $D$  may either fail together or not. The event tree for this example remains the same as shown in Figure II.2.

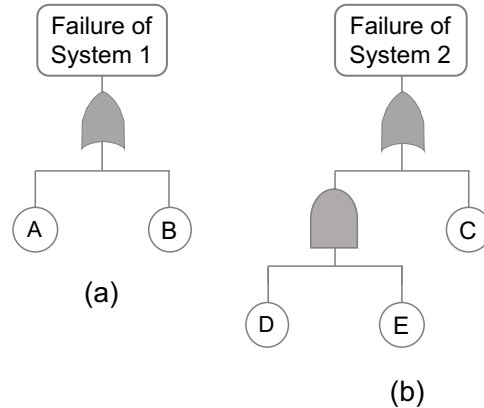


Figure II.4: Fault trees for (a) System 1 (b) Modified System 2

A basic parameter model (BPM) is used to model the dependent failures due to a common cause [56]. It states that in a common cause failure group of size  $n$ , let  $m$  failures occur simultaneously where  $m$  can range from 1 to  $n$ . A component failure

probability involves independent failure of the component and combinations of CCFs with other components in the common cause component group. The total failure of  $A$  is divided into an independent failure and a failure due to common cause event which is shared with component  $D$ , as expressed in Eq. (II.12). The updated fault trees after the expansion from BPM are shown in Figure II.5.

$$A_T = A_I \cup C_{AD} \tag{II.12}$$

$$D_T = D_I \cup C_{AD}$$

where,  $A_T$  and  $D_T$  are the total failure of  $A$  and  $D$  from all causes,  $A_I$  and  $D_I$  is the failure of  $A$  and  $D$  from independent causes and  $C_{AD}$  is the failure of  $A$  and  $D$  from common causes.

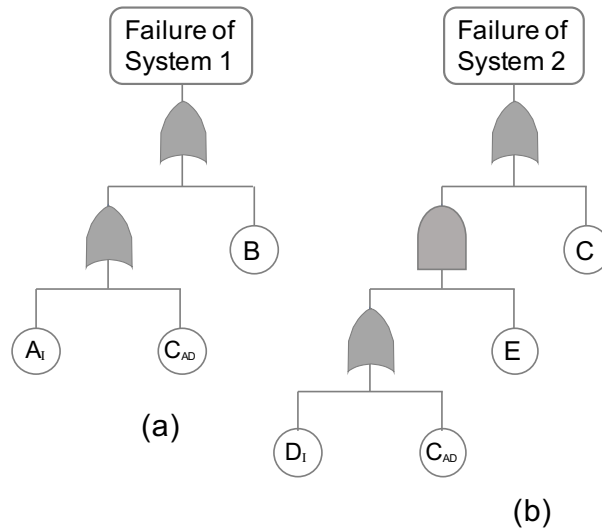


Figure II.5: Fault trees for (a) System 1 (b) System 2

There are various models which can be used to quantify the independent failure probabilities ( $A_I$ ,  $D_I$ ) and joint failure probability in common cause component group

( $C_{AD}$ ). Some of these models are the alpha factor model, the beta factor model, and multi Greek letter model [57]. These models require input related to component failures and their frequencies which are based on expert judgment and past failures at plants. The probabilities for common cause failures require continuous updating based on experience data. In United States, US Nuclear Regulatory Commission (USNRC) collects and disseminates data on such failures and provides an estimation for the parameters needed in the modeling [58, 59].

We consider alpha factor model in this study to quantify the failure probability due to CCF. The alpha factor model for CCF relies on an underlying assumption of symmetry, i.e. the probabilities of events in common cause component group involving same number of components are the equal. For example, probabilities of events  $A_I$  and  $D_I$  are considered to be equal.

The exact failure for the accident sequence 2 with updated fault trees can be expressed by rewriting Eq. (II.5) as:

$$Seq\ 2 = (A_I \cup B \cup C_{AD})^c \cap \left( C \cup (E \cap (D_I \cup C_{AD})) \right) \quad (II.13)$$

The above equation simplifies to:

$$Seq\ 2 = A_I^c \cap B^c \cap C_{AD}^c \cap (C \cup (E \cap D_I)) \quad (II.14)$$

The exact probability for the accident sequence 2 in Eq. (II.14) is given by:

$$P_{Seq\ 2, Exact} = (1 - P_{A_I})(1 - P_B)(1 - P_{C_{AD}})(P_C + P_E P_{D_I} - P_C P_{D_I} P_E) \quad (II.15)$$

In a traditional PRA implementation of the above problem, the first step lies in determining the cut sets for both the systems. The minimal cut sets for System 1 are  $A_I, B, C_{AD}$  whereas the minimal cut sets for System 2 are  $C, E \cap D_I, E \cap C_{AD}$ . After removing the common cut set  $C_{AD}$  from the System 2 cut sets, the cut sets for accident sequence 2 are  $C, E \cap D_I$ . The failure probability of accident sequence 2 is then given by:

$$P_{Seq\ 2, Trad} = (P_C + P_E P_{D_I} - P_C P_{D_I} P_E) \quad (II.16)$$

Table II.3 shows the differences that can exist for the failure probabilities during DB and BDB scenarios between the traditional PRA approach and the exact calculations. As seen in Table II.3, significant over-estimation exists when the traditional PRA approach for BDB scenario is used.

Table II.3: Comparison of quantification estimates for failure sequence 2 with CCF events

	$P_{A_I}$	$P_B$	$P_C$	$P_{D_I}$	$P_E$	$P_{C_{AD}}$	Traditional	Exact	% $\varepsilon$
<b>DB</b>	4.9E-4	1.6E-3	5.6E-3	0.074	0.0049	0.00035	0.00563	0.00562	0.21
<b>BDB</b>	0.15	0.085	0.25	0.07	0.15	0.0074	0.2578	0.1990	29.2

As the number of cut sets for an accident sequence increases, formal approaches are needed to quantify the probability of accident sequence in the traditional approach. One such method often considered to be most accurate is termed as inclusion-exclusion principle [60]. A generalized form of the failure probability as expressed by the inclusion-exclusion principle to quantify the accident sequence probability is shown in Eq. (II.17).

The expression in Eq. (II.16) can also be derived using Eq. (II.17).

$$\begin{aligned}
 P_{Seq} = P\left(\bigcup_{i=1}^m CS_i\right) &= \sum_{i=1}^m P(CS_i) - \sum_{i<j}^m P(CS_i \cap CS_j) + \\
 \sum_{i<j<k}^m P(CS_i \cap CS_j \cap CS_k) &+ \dots + (-1)^{m-1} P\left(\bigcap_{i=1}^m CS_i\right)
 \end{aligned} \tag{II.17}$$

where,  $m$  is total number of minimal cut sets and  $P(CS_i)$  is the probability of the  $i$ -th minimal cut set. For a realistic system, the total number of accident sequences can be quite large, and the computational cost of using the inclusion-exclusion principle increases exponentially with an increase in the number of cut sets. It is because the inclusion-exclusion principle takes the combination of all the minimal cut sets and results in a computational cost that is in the order of  $O(2^m)$ . To reduce the computational cost, traditional approaches often utilize simplified and approximated methods such as, rare event approximation and minimal cut set upper bound [61]. The rare event approximation is only accurate when cut set probabilities are very small and the cut sets do not share same basic events (Eq. (II.18)). When relatively large cut set probabilities are observed, the rare approximation can even exceed 1. The minimal cut set upper bound provides an upper bound for the exact quantification of failure probability (Eq. (II.19)). The minimal cut set upper bound approximation works well when the logic trees only contain AND and OR gates, otherwise it leads to conservative results. Such approximations lead to even greater errors in quantification for BDB events. At the same time, quantification of exact probabilities of failure and all the accident sequences in a realistic system is also impractical due to significant effort and

even larger computational demands compared to the traditional approaches.

$$P_{Seq} = \sum_{i=1}^m P(CS_i) \quad (\text{II.18})$$

$$P_{Seq} = 1 - \prod_{i=1}^m (1 - P(CS_i)) \quad (\text{II.19})$$

Among the different alternatives, one potential solution is to use Bayesian network. It is just one alternative which is used in this study. It is quite possible that other alternatives solutions might be relatively more efficient. However, the purpose of this study is not to assess the relative efficiencies of different options.

A Bayesian network has several advantages over traditional fault tree and event tree analysis due to its robust and mathematically coherent framework which facilitates modelling direct or indirect causes of accidents and facilitates updating quantification based on new information. Bayesian networks permit a more general statistical relationship than the logical combinations provided in event trees and associated fault trees. Bayesian networks do not make assumptions based on large or small probabilities. In addition, the modelling of dependent events is an inherent part of the Bayesian networks. Hence, Bayesian network framework can overcome the challenges faced due to conservative assumptions involved in the definition and computation of event tree sequences in the current PRA tools.

## II.2.5 Bayesian Networks

A Bayesian network (BN) is a probabilistic graphical model that provides more general form of statistical relationships among the events compared to fault and event trees.

This generality makes it a powerful tool for structure/system reliability, risk management, accident analysis, artificial intelligence, etc. A Bayesian network is a directed acyclic graph which consists of three main parts: the nodes (or vertices) that represent a discreet or continuous random event/variable, the arrows (or arcs) that represent direct relationship between nodes, and the conditional probability table (CPT) assigned to nodes that describe the quantitative relationship between interconnected nodes. The nodes which have arrows directed to other nodes are parent nodes and nodes that have arrows coming from other nodes are called as child nodes. A node that do not have any arrow coming from another node is called as root node and it does not have any parent node. An intermediate node serves as a parent node as well as a child node. The Bayesian analysis is performed based on the conditional probability of nodes and conditional independence assumption, i.e.  $P(x, y|z) = P(x|z)P(y|z)$  if and only if  $x \perp y|z$ . The joint probability distribution of the entire Bayesian network can be expressed using conditional probability as:

$$P(X_1, X_2, \dots, X_n) = \prod_{i=1}^n P(X_i | Parent(X_i)) \quad (II.20)$$

where,  $X$  represents the nodes/variables in a Bayesian network,  $n$  is total number of variables/nodes in the Bayesian network,  $Parents(X_i)$  represents all the variables on which  $(X_i)$  is dependent.  $P(X_i | Parent(X_i))$  is the CPT of node  $X_i$ .

The importance of Bayesian networks has been illustrated by many researchers [14, 62, 63], but their adoption in industry practice is quite limited. Bayesian networks have also been used to address common cause failures in fault trees but their use to model event trees has been quite limited to very simple event trees. It appears that

existing literature on Bayesian network representation of event trees does not address the impact of common cause failures that affect multiple fault trees in a single event tree diagram. Such common cause failures across multiple fault trees lead to a complex network of interconnected fault trees. Therefore, in this study, we use Bayesian networks to quantify accident sequences with common cause failures across multiple fault trees in an event tree.

While a Bayesian network overcomes the computational limitations of exact calculations in an event and fault tree analysis, its primary limitation is the requirement of a significantly large memory storage for  $2^n$  elements. This is because, if  $n$  basic events are connected to an intermediate event, then the CPT of the intermediate event will have  $2^n$  elements. However, with rapid advances in the computing and storage power available to engineers and with advances in development of new algorithms, the memory storage required to construct CPT of Bayesian networks can be significantly reduced [64, 65]. It must be realized that the new algorithms come at the expense of increased computation cost, but it is not a major increase and there is a trade-off that exists between the memory storage and the computation cost. Clearly, it appears that a Bayesian network is a logical and rational approach to evaluate accurate risk and accident sequences for BDB scenarios and needs to be researched in greater detail. An event tree and its corresponding fault trees are mapped into a Bayesian network using the following steps:

- Convert top events in the event tree (ET) into nodes in the Bayesian network.
- Convert accident sequences in the ET into node states and directional arcs between nodes in the BN.
- Develop conditional probability tables (CPTs) for the mapped top events in the

BN based on accident sequences in the ET.

- Next map the corresponding fault trees (FT) into the BN
  - The basic events in the FT are mapped into root nodes in the BN and the intermediate events in the FT are mapped into child nodes of root nodes.
  - The top event is connected as child nodes of basic and intermediate events.
  - The logic gates, AND and OR gates in the FTs are converted into CPTs of the mapped intermediate and top event nodes in the BN.

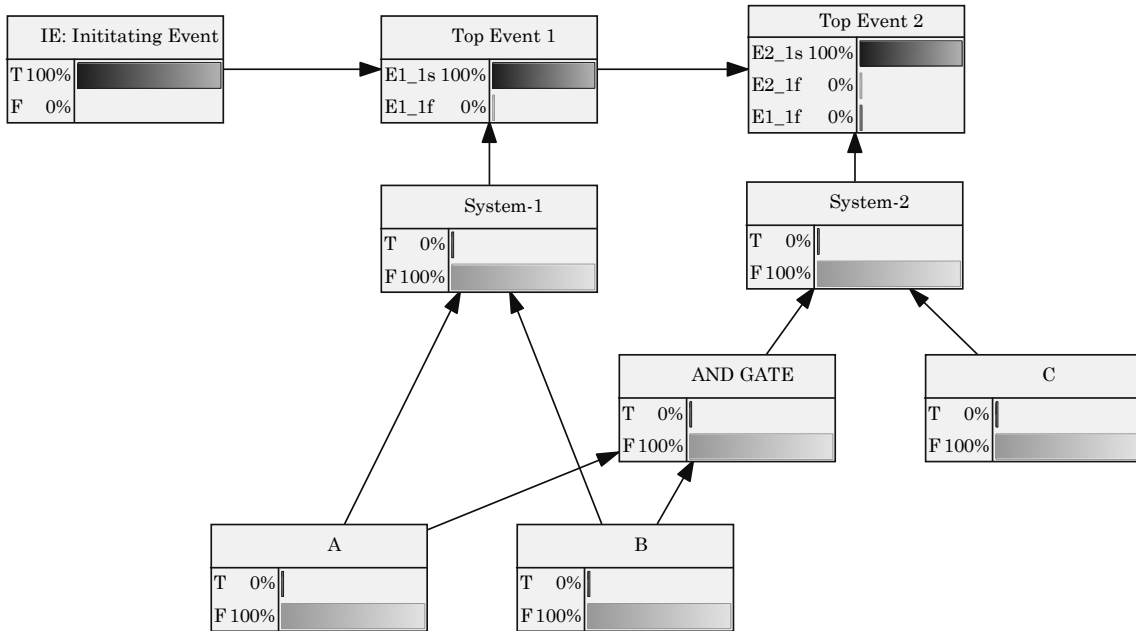


Figure II.6: Mapped Bayesian network with node states corresponding to the event tree shown in Figure II.2

The marginal probability of the top node in the Bayesian network is calculated from the joint distribution of the network. In this study, we consider only binary states (*failure* or *success*) of the nodes in the Bayesian network. The probabilities from all the possible states must be combined based on Eq. (II.20). Then, algorithms such as variable

elimination and belief propagation can be used for exact quantification of marginal and joint probabilities with better computational cost compared to the traditional approaches [66, 67, 68].

In this study, we present a simple algorithm using fundamentals of logic gate properties to demonstrate the exact computation of failure sequence probability in the Bayesian network. For brevity, accident sequence 2 is considered to compare the quantification of accident sequence failure probability between event tree analysis and the corresponding mapped Bayesian network. Figure II.7 shows the Bayesian network corresponding to accident sequence 2. The top node in the Bayesian network is dependent on success of System 1 top event ( $FT_1^c$ ) and failure of System 2 top event ( $FT_2$ ). Similarly, the fault trees of top events System 1 and System 2 are mapped into the corresponding nodes and arcs. Logical AND relation between event  $A$  and  $B$  has been mapped to a node named  $InE1$  which represents an intermediate event. The basic events  $A$ ,  $B$ ,  $C$  in the Bayesian network are assumed to be independent events, which is true unless they share common cause failure events. Such dependence between basic events to explain the effect of the common cause failures is discussed later. Figure II.7 also shows the relationship of all the child nodes with their parent nodes. Note that the successful top event must be added in the network as a complement. The complement of successful top event ( $FT_1^c$ ) for the sequence is defined as  $(A \cup B)^c$ , which is modified using De Morgan's rule in the figure. According to De-Morgan rule,  $(A \cup B)^c = A^c \cap B^c$ .

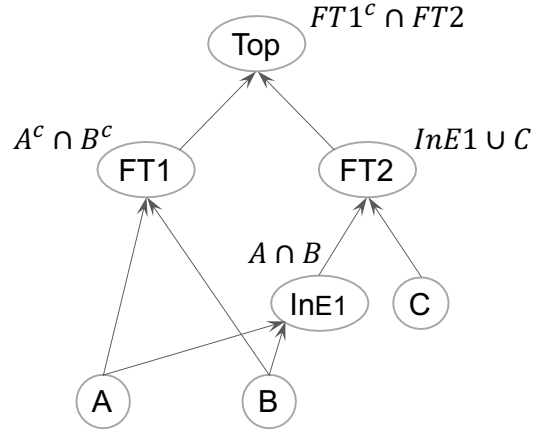


Figure II.7: BN for accident sequence 2

## II.2.6 Quantification of the Bayesian Network

The proposed algorithm is based on a top-down approach and it is implemented based on following steps:

1. Start with the *failure* state of the top node ( $TE = f$ ).
2. Identify the parent nodes of the top node, let us say  $Parent(TE) = IE1, IE2$ . Then, obtain all possible combination of the parent nodes that can lead to the *failure* state of the top node. If suppose,  $TE$  is connected to nodes  $IE1, IE2$  by AND logic, then there is only one possible combination of  $IE1, IE2$  that can lead to *failure* state of  $TE$  i.e. both  $IE1, IE2$  must fail.

$$P_{(TE=f)} = P(TE = f|IE1 = f, IE2 = f)P(IE1 = f)P(IE2 = f)$$

where,  $f$  represents a *failure* state. The *failure* state of a node  $X$  in the Bayesian network represents the failure of an event in the fault and event tree diagram i.e.  $P(X = f) = P(X)$ . If suppose,  $TE$  is connected to nodes  $IE1, IE2$  by OR logic,

then there are three combinations of  $IE1$ ,  $IE2$  states that can lead to *failure* state of TE i.e. either  $IE1$  fails,  $IE2$  fails or both  $IE1$  and  $IE2$  fail.

$$\begin{aligned}
 P_{(TE=f)} &= P(TE = f|IE1 = f, IE2 = f)P(IE1 = f)P(IE2 = f) + \\
 &\quad P(TE = f|IE1 = s, IE2 = f)P(IE1 = s)P(IE2 = f) + \\
 &\quad P(TE = f|IE1 = f, IE2 = s)P(IE1 = f)P(IE2 = s)
 \end{aligned}$$

where,  $s$  represents a *success* state. The *success* state of a node  $X$  in the Bayesian network represents the success of an event in the fault and event tree diagram i.e.

$$P(X = s) = P(X^c)$$

3. Next, identify the parent nodes of the nodes ( $IE1$ ,  $IE2$ ) obtained in step 2. Similarly, obtain all the possible combinations of the parent nodes than can lead to the state of the node obtained in step 2. If suppose,  $IE1$  is connect to nodes  $BE1$ ,  $BE2$  by AND logic, then

$$\begin{aligned}
 P_{(TE=f)} &= P(TE = f|IE1 = f, IE2 = f)P(IE1 = f)P(IE2 = f) \\
 &= P(TE = f|IE1 = f, IE2 = f)P(IE1 = f|BE1 = f, BE2 = f) \times \\
 &\quad P(BE1 = f)P(BE2 = f)P(IE2 = f)
 \end{aligned}$$

4. Repeat the same process for all the parent nodes in step 3 until the root node is reached.
5. Compute the sequence probability by substituting the probabilities for conditional nodes and root nodes. The failure probabilities for all the conditional nodes is equal to 1.

The top-down algorithm for the quantification of accident sequence 2 shown in Figure II.7 can be summarized as follows:

- The failure probability of accident sequence 2 is represented by the failure of the top event in the Bayesian network.

$$P_{Seq\ 2} = P(Seq\ 2|Top = f)P(Top = f) \quad (II.21)$$

- The node is related to nodes  $FT_1^c$ ,  $FT_2$  by an AND logic. Therefore, there is only one combination of  $FT_1^c$ ,  $FT_2$  that can lead to *failure* state of  $Top$  node.

$$P_{(Top = f)} = P(Top = f|FT_1 = s, FT_2 = f)P(FT_1 = s)P(FT_2 = f) \quad (II.22)$$

- Next,  $FT_1^c$  node is related to the nodes  $A$ ,  $B$  by AND logic.

$$P_{(FT_1 = s)} = P(FT_1 = s|A = s, B = s)P(A = s)P(B = s) \quad (II.23)$$

- As  $A$ ,  $B$  are root nodes, we move back to the node  $FT_2$ .  $FT_2$  is connected to nodes  $InE1$ ,  $C$  by OR logic. Therefore, there will be three combinations of  $InE1$ ,  $C$  that can lead to *failure* state of  $FT_2$ . However, node  $InE1$  is related to nodes  $A$ ,  $B$  by AND logic and they have already been observed ( $A = s$ ,  $B = s$ ) in the previous step. Therefore, the node state of  $InE1$  is *success* state.

$$P_{(FT_2 = f)} = P(FT_2 = f|InE1 = s, C = f)P(InE1 = s|A = s, B = s)P(C = f) \quad (II.24)$$

- Since, there are no more conditional dependencies, the failure probability of

Sequence 2 can be calculated by substituting Eqs. ((II.22) - (II.24)) into Eq. (II.21). The probabilities for the conditional nodes is equal to 1 and  $P_A$ ,  $P_B$ , and  $P_C$  are the failure probabilities for events  $A$ ,  $B$ , and  $C$ , respectively.

$$\begin{aligned}
P_{Seq\ 2} &= P(Seq\ 2|Top = f)P(Top = f|FT_1 = s, FT_2 = f) \times \\
&\quad P(A = s)P(B = s)P(FT_2 = f|InE1 = s, C = f) \times \\
&\quad P(InE1 = s|A = s, B = s)P(C = f) \\
&= 1 \times P(A = s)P(B = s)P(C = f) \\
&= (1 - P_A)(1 - P_B)P_C \\
&= P_C - P_AP_C - P_BP_C + P_AP_BP_C
\end{aligned} \tag{II.25}$$

Eq. (II.25) is identical to Eq. (II.4) i.e., the Bayesian framework leads to the exact computation of probability for the failure of the accident sequence 2. Similar results are obtained for the failure sequence even when the common cause failures are present. Figure II.8 shows the mapped Bayesian network for the accident sequence 2 when common cause failures are present between event  $A$  and  $D$ . After using the basic parameter model to evaluate failure probability associated with CCFs and following the top down approach to solve the Bayesian network shown in Figure II.8, we obtain the following equation for the probability of failure of sequence 2:

$$P_{Seq\ 2, CCF} = (1 - P_{A_I})(1 - P_B)(1 - P_{C_{AD}})(P_C + P_EP_{D_I} - P_CP_{D_I}P_E) \tag{II.26}$$

Again, Eq. (II.26) is identical to Eq. (II.15) which leads to exact computation of the failure sequence. Therefore, Bayesian networks can be used as an alternative tool for risk assessment of BDB studies.

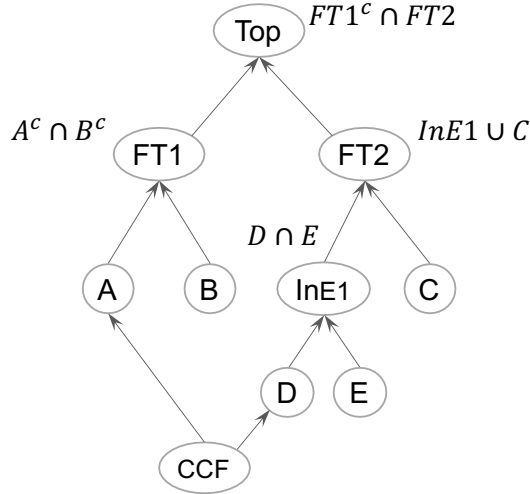


Figure II.8: Mapped BN for Failure sequence 2 with common cause failures

### II.3 Example Application to Nuclear Power Plant PRA

The discussion presented above uses simple systems to illustrate the limitations of the traditional PRA tools in evaluating the probabilities of failures and accident sequences accurately for the BDB scenarios. In this section, a relatively more complex example is considered to illustrate the same. The purpose is to explore the impact of assumptions in the traditional tools in a more complex system. For this purpose, an event tree connecting the failure and availability of four different heat removal systems is considered. Figure II.9 shows the event tree that is based on loss-of-coolant accident (LOCA) initiating event, which may have occurred due to any reason including external hazard. This event tree is based on the corresponding data presented in Mandelli et al. [69]. As shown in Figure II.9, the initiating LOCA triggers reactor trip scram (RTS). This is followed by the availability or failure of four different heat removal systems that consists of auxiliary feedwater system failure (AFWS), high pressure coolant injection (HPI), emergency core cooling system (ECCS), and residual heat removal (RHR). The sequence of failure of

these systems is taken from Kim et al. [70] and these systems are the most critical systems for evaluating the core damage probabilities in prior studies [71, 72, 73]. As mentioned earlier, the initiating event frequency is not considered for the quantification as its consideration would not change the identified critical path.

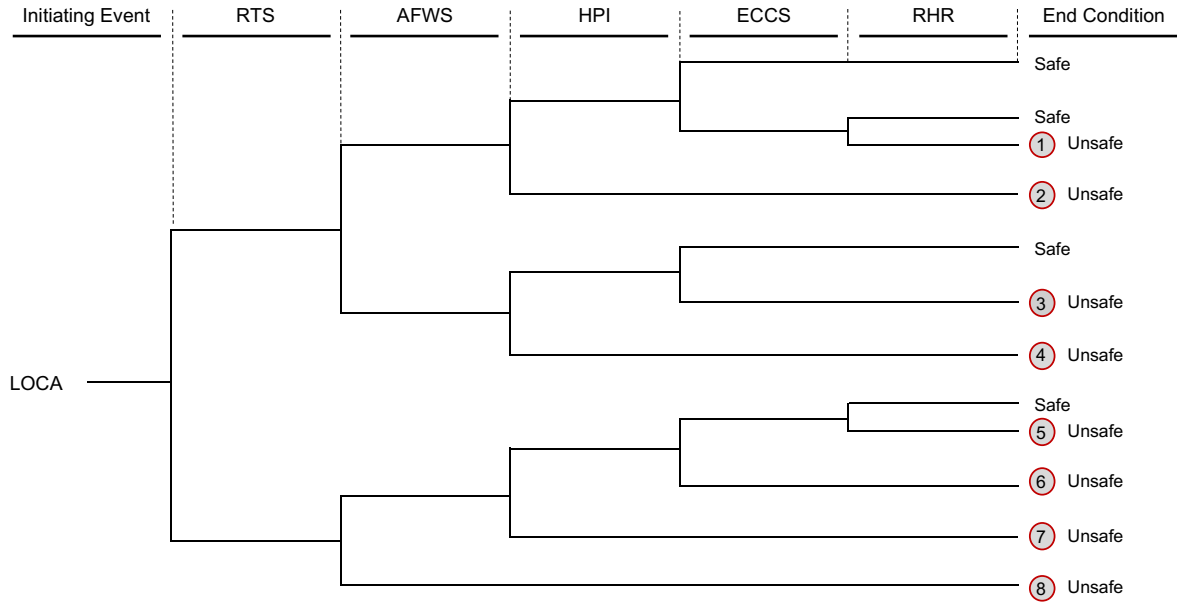


Figure II.9: Event tree diagram for risk assessment of nuclear power plant

Each protection system in the event tree can have its own fault tree. The fault trees for each system and the failure probabilities for the components in these fault trees are taken from existing literature listed in Table II.4. A fault tree is not considered for RTS. Instead, a constant value of failure probability (DB:  $4.8E - 6$ , BDB:  $4E - 1$ ) for RTS is considered for simplicity and brevity. This is appropriate in the context of this study because all the basic events and the intermediate events in the RTS fault tree are independent of the events in the fault trees of the other systems. The fault trees for all the four heat removal systems are simplified for illustration purpose in this study. A full-

scale fault tree of each systems consists of several redundant components. Simplification is achieved by eliminating redundancy and keeping all the critical events. Care is taken to ensure that the failure probability of the top event in the fault tree is either identical or very close to that in the full-scale fault tree available in literature. The fault tree diagram for top events AFWS, HPI, ECCS, and RHR failures are shown in Figure II.10 to Figure II.13.

In addition to these simplifications, common cause failures are introduced in the fault trees for heat removal systems. Since AFWS and HPI are also part of the emergency core cooling, the fault trees for ECCS shares common events or common cause failures with the fault trees for HPI and AFWS. The AFWS and ECCS share one common cause failure event (CCF1) related to electric pump failure. The size of common cause component group (CCCG) for CCF1 is 4. The fault tree of AFWS has one more common cause failure event (CCF2) related to lack of water with size of CCCG as 2. The fault tree of HPI has two common cause component group: one is related to valve failure (CCF3) and another is related to pump outlet failure (CCF4). The size of common cause component group for both CCF3 and CCF4 is 2.

Basic events that are common to two fault trees or are part of common cause failure are considered to have the same probabilities of failure. This leads to some change in the probability of failure for the top events in these fault trees but the change is minor in each case and the numbers are very close to the corresponding numbers from the full-scale fault trees given in literature.

The information on common cause failures such as identification of the events considered in common cause failure and their failure probability is taken from USNRC

database [59]. The final determination of the configuration and probabilities of the fault and event trees in this study are based on the values available in the existing literature on DB events. For BDB scenario, the available literature is quite limited. One comprehensive study for BDB scenario is given in Barto et al. [10]. The failure probabilities of basic events for DB events in Barto et al. [10] are on the same order as those given in Figure II.10 to Figure II.13. The failure probabilities of basic events for the BDB condition considered in Barto et al. [10] are about two or more order of magnitude higher than the corresponding DB values. Therefore, the failure probabilities of all the basic events for the BDB in this study are taken as about two orders higher than the corresponding DB values, as shown in Figure II.10 to Figure II.13.

Table II.4: Sources for the fault tree diagrams

System failure	Source
Reactor Trip Scram (RTS)	Kamal and Hill [74]
Auxiliary Feedwater System Failure (AFWS)	Lava et al. [75]
High Pressure Coolant Injection (HPI)	Heising et al. [76]
Emergency Core Cooling System (ECCS)	Purba et al. [77]
Residual Heat Removal (RHR)	Vasconcelos et al. [78]

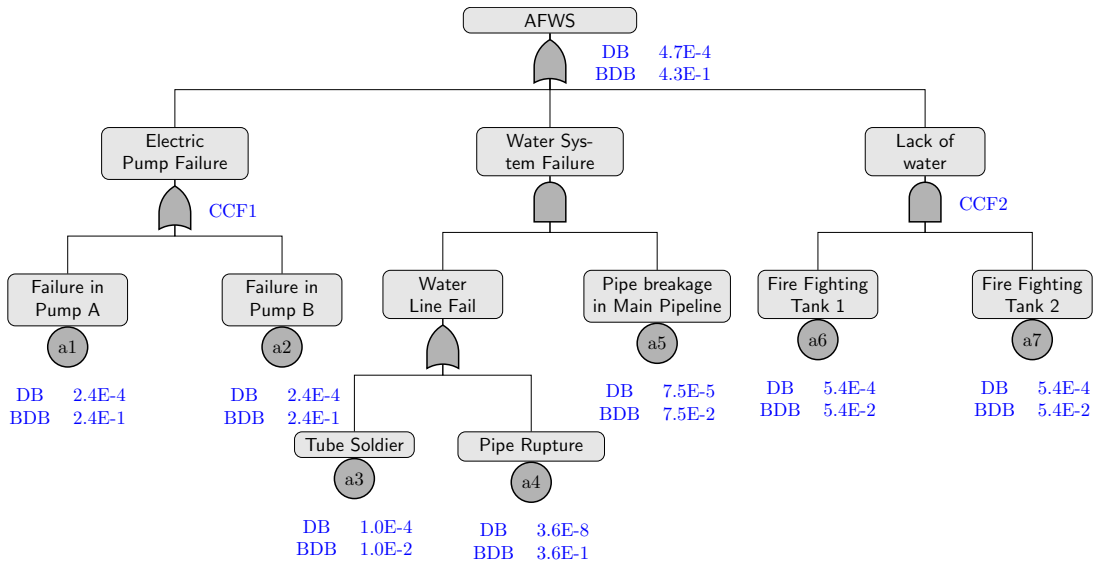


Figure II.10: Fault Tree diagram for Auxiliary Feedwater System (AFWS) Failure

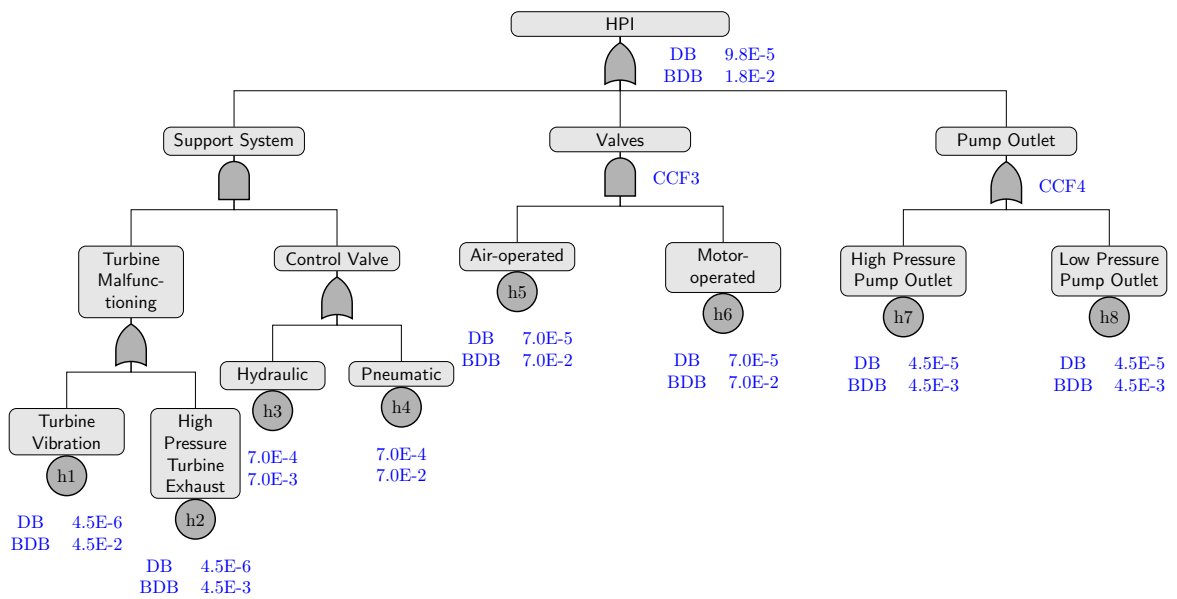


Figure II.11: Fault Tree diagram for High Pressure Coolant Injection (HPI) Failure

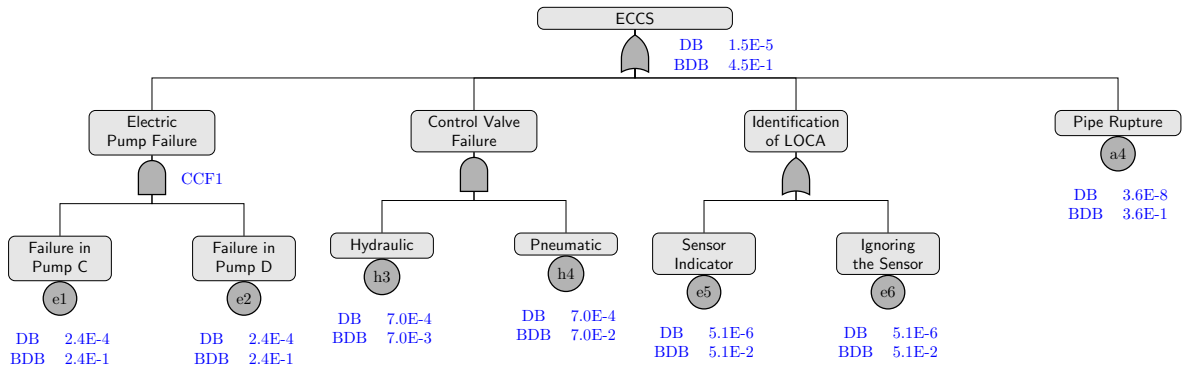


Figure II.12: Fault Tree diagram for Emergency Core Cooling System (ECCS) Failure

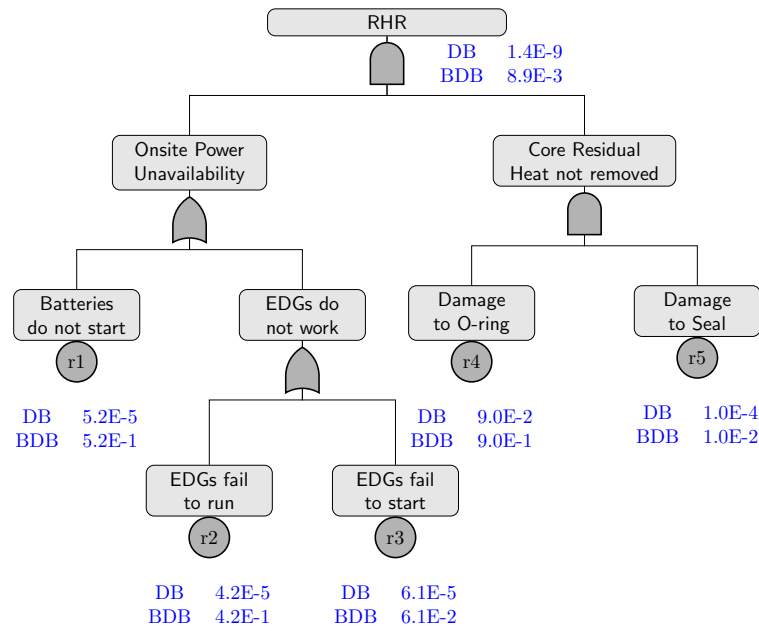


Figure II.13: Fault Tree diagram for Residual Heat Removal (RHR) Failure

Next, the fault and event trees considered in this example are mapped into the corresponding Bayesian network as shown in Figure II.14. All the intermediate events in the fault tree diagrams are converted to intermediate nodes (referred to as gates) in the Bayesian network.

For comparison purposes, two different traditional PRA quantification methods are used to compute the probabilities of failures and accident sequences in the event tree from the list of minimal cut sets. These are the upper bound approximation and the rare event approximation. The failure probabilities for the eight harmful accident sequences in the event tree from both these methods as well as those calculated from the corresponding Bayesian network are given in Table II.5 for the DB condition. As seen in Table II.5, the numbers from the three different approaches are almost identical. Next, the same computation is performed for the case of BDB condition. The corresponding results are shown in Table II.6. It can be seen that the failure probabilities for all the accident sequences in Table II.6 are close to each other from the traditional upper bound and rare event approaches but quite different from those calculated using the proposed Bayesian network approach. These differences are given in the table as percent error ( $\% \varepsilon$ ). In addition, the critical accident sequence from the traditional method is Seq. 3 whereas that from the Bayesian network approach is Seq. 8. Even the relative order of sequences based on the failure probability of core damage in each sequence is different from the traditional and the Bayesian approaches.

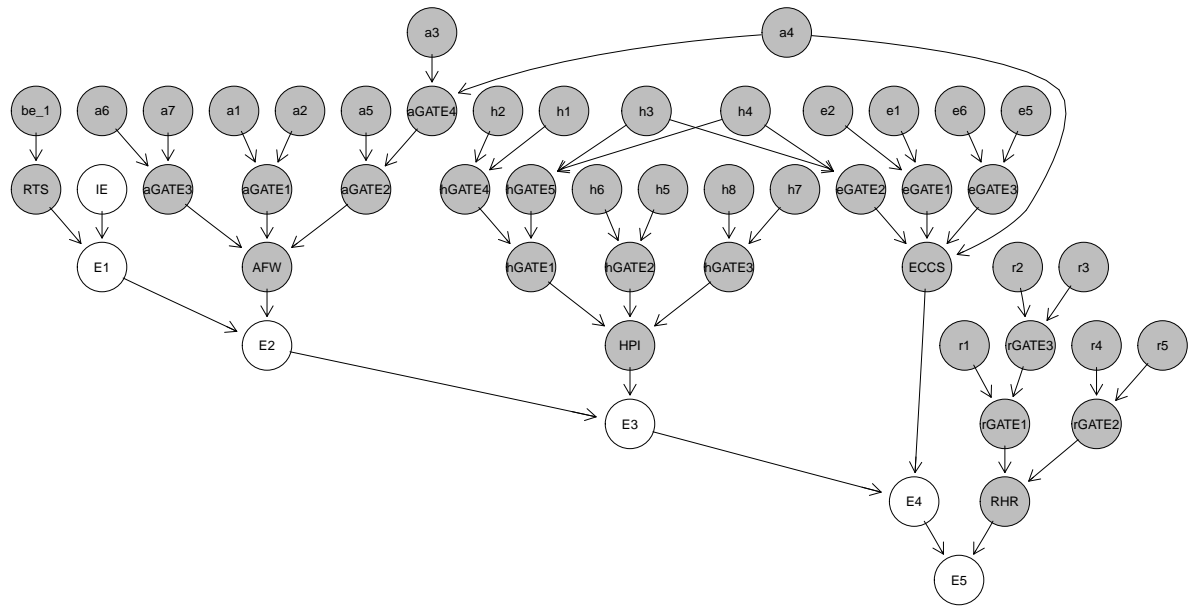


Figure II.14: Mapped Bayesian network corresponding to the event tree shown in Figure II.9

Table II.5: Comparison between traditional event tree analysis (ETA) and Bayesian network (BN) approach for DB scenario

Sequence	ETA Upper Bound (% $\varepsilon$ )	ETA Rare (% $\varepsilon$ )	BN
1	1.60E-14 (1.21)	1.62E-14 (0.06)	1.62E-14
2	9.81E-05 (0.05)	9.81E-05 (0.05)	9.81E-05
3	4.17E-06 (0.01)	4.17E-06 (0.01)	4.12E-06
4	4.69E-08 (0.02)	4.69E-08 (0.02)	4.69E-08
5	6.66E-15 (0.45)	6.70E-15 (0.06)	6.64E-15
6	5.57E-11 (0.06)	5.57E-11 (0.06)	5.57E-11
7	4.71E-10 (0.05)	4.71E-10 (0.05)	4.71E-10
8	2.29E-09 (0.01)	2.29E-09 (0.01)	2.29E-09

Table II.6: Comparison between traditional event tree analysis (ETA) and Bayesian network (BN) approach for BDB scenario

Sequence	ETA Upper Bound (% $\varepsilon$ )	ETA Rare (% $\varepsilon$ )	BN
1	0.004702 (382.0)	0.004711 (382.9)	0.000976
2	0.018234 (195.3)	0.018365 (197.5)	0.006174
3	0.248530 (101.7)	0.277495 (125.2)	0.123248
4	0.009251 (95.0)	0.009290 (95.8)	0.004744
5	0.003600 (335.9)	0.003604 (336.4)	0.000826
6	0.198524 (102.9)	0.209155 (113.8)	0.097838
7	0.007325 (78.0)	0.007346 (78.5)	0.004116
8	0.190721 (9.7)	0.202341 (16.4)	0.173813

## II.4 Conclusions

This study illustrates the limitations of the traditional PRA tools for event tree and fault tree analysis in the context of BDB scenario. The approximations due to key assumptions in traditional PRA approaches are studied and their impact on the evaluation of critical accident sequences is illustrated. A simple illustrative example is used to compare the outcome of traditional approach with that from an exact quantification. This comparison is used to demonstrate the impact of fundamental assumptions in the traditional PRA

approaches on risk quantification. It is shown that the traditional approaches may lead to significant differences in the estimation of failure probabilities of accident sequences and also an inaccurate identification of critical paths in the context of BDB scenario. More specifically, the assumptions that lead to such discrepancies are:

- Probabilities of failure for basic events in fault and event trees are very small.
- All the component failures are assumed to be independent of each other.

A quantification of exact failure probabilities for accident sequences in an event tree is computationally prohibitive for realistic systems. Moreover, event and fault trees pose additional limitations in certain applications of PRA that would consider: (a) statistical correlations between basic events, and (b) non-binary relations between intermediate and basic events. These limitations can be addressed by implementing Bayesian network representation of PRA. In this study, Bayesian network approach is used to quantify accident sequences with multiple common cause failure events across multiple fault trees. The event tree and its corresponding fault trees are mapped into a Bayesian network using a mapping algorithm. A simple top-down algorithm using fundamentals of logic gate properties is introduced to demonstrate the computation of failure sequence probability in Bayesian network representation of PRA. The results obtained from Bayesian network computation are identical to that of the exact quantification using set theory. In addition to simple illustrative examples, a relatively more complex scenario in nuclear power plants is used to compare the differences in risk estimates that are calculated based on traditional PRA tools and the corresponding quantification from Bayesian network analysis. The traditional approximation methods give accurate results for the case of DB scenario but can lead to incorrect risk estimates in the case of BDB scenario. It is also shown that incorrect risk estimates can also lead

to incorrect critical accident sequences in the systems analysis.

---

---

## PART III

---

### Computationally Efficient Approach for Risk-informed Decision Making

### III.1 Introduction

The nuclear industry has increased its reliance on probabilistic risk assessment tools for design, operation, life extension, and regulation. Probabilistic safety assessment (PSA) or probabilistic risk assessment (PRA) evaluates the risk associated with a specific hazard by a convolution of system fragility and hazard curve [1, 2, 3, 45, 46]. A fragility curve of structures, systems, or components (SSCs) is expressed as the conditional probability of failure for a given hazard and is a function of the uncertainties in empirical, experimental, and/or numerical data for the available physical models of the SSCs [4, 5, 6]. US Nuclear Regulatory Commission (USNRC) and International Atomic Energy Agency (IAEA) have issued guidelines for conducting a full scope PRA, where the plant level risk is calculated by combining the component and the subsystem fragility curves through a systems analysis [45, 46]. Typically, fault and event trees are used for conducting the systems analysis by logically combining the component level fragilities, and for convoluting the fragilities with the hazard curve [4, 5, 6].

Fault tree analysis is one of the most powerful tools in PRA to represent the failure probability of a nuclear plant based on the failure probabilities of its basic components. It was first invented in 1961 at the Bells Labs and thereafter it has been used throughout the nuclear power industry for safety and reliability analyses. The events in fault and event trees are modeled using binary states: failure or not a failure (success) [13]. The fault trees or event tree for an actual power plant unit can be fairly large in size with several different types of logic gates, interconnected events, and dependent events, etc. A large fault tree includes thousands of gates, basic events (BEs), multiple occurring events (MOEs), and dependent events. Complex connectivities can give rise to excessive computational

demand and storage requirement for the analysis. An assessment of logic trees with  $n$  events requires an analysis of  $2^n$  scenarios emerging from the various combinations of event failures. Hence, a complete analysis of a logic tree demands computational complexity of the order  $2^n$  in either time or space. The exponential order of complexity is required mainly to address the dependencies and dependent failure events.

Bottom-up methods [79] have been used when fault trees do not include MOEs and dependent events. However, for a complex fault tree, solvers such as minimal cut sets analysis [80] are implemented. The computation of the minimal cut sets of fault trees poses an intractable NP-hard problem [81, 82]. Therefore, the analysis of large fault trees requires significant computational resources, which makes the analysis of PRA models inefficient and time consuming. A brute force Monte Carlo simulation [83] is used as an alternative when the number of cut sets exceed the available computer memory.

The exact computation of fault and event tree outputs using the minimal cutsets approach is implemented in the widely used MOCUS algorithm [80]. However, this approach suffers from exponential complexity of computational time as a function of minimal cutsets in a network. If the network consists of only the OR gates then the time complexity would be  $O(2^{n_{BE}})$ , where  $n_{BE}$  is the number of basic events. Even with advancements in available computational power, it is impractical to solve real-world problems using algorithms for which computational time that increases exponentially with the increase in network size. The exponential complexity has been a major roadblock for performing the risk assessment accurately for a very long time.

There are several traditional methods that reduce the complexity but at the expense of accuracy [84]. In order to reduce the computational demands of the fault

and event tree networks, several assumptions or approximations are often relied upon. Modularization techniques have been used for pre-processing of large fault trees [16, 17]. It divides the entire fault tree into different modules and solves them independently. Several researchers have proposed methods to identify the modules efficiently [82, 85, 86]. But all the existing modularization methods are still not powerful enough to deal with a very large fault tree that contains several MOEs and dependent events. Binary decision diagram (BDD) is another established method where a fault tree is converted to BDD structure based on the Boolean logic expression of the FT [18, 19, 20]. However, the size of BDD structure increases exponentially with the number of variables. Additionally, the size of BDD structure is very sensitive to the order of variables [87]. Therefore, an excessive memory demand is an additional limitation for solving large FTs with BDD algorithm. Consequently, approximations are used. Even with approximations, it results in polynomial computational complexity [81, 85].

Recently, parallel processing techniques have been introduced in PRA that execute multiple critical processes simultaneously. This allows a reduction in the computational time and storage requirements efficiently for modularization [21, 22], BDD [15, 23], cut set dependency [24] and Monte Carlo simulations [25]. However, the complexity inherent in traditional methods still remains intractable for large size problems.

Bayesian networks have also been used for the analysis of the network diagrams. However, it appears that all formulations present some computational difficulties [26], which are due to the size of conditional probability tables [27, 28], the size of clique potentials in the junction-tree algorithm [29] or the recursive algorithm for the identification of minimum link sets [30].

Computational efficiency of PRAs has gained greater attention in recent years particularly after the Fukushima-Daiichi nuclear accident. There have been many studies focused on multihazard PRA since then. For example, seismically induced tsunami or hurricane induced wind and flooding scenarios have been studied quite widely. Prior to such studies, PRAs for individual hazards were conducted in isolation of each other. The fault and event tree network for each such single hazard is fairly large by itself. Consideration of multihazard scenarios have resulted in a single dependent initiating event which links the fault and event tree networks for individual hazards thereby resulting in a very large network for combined hazards. The need for improvement in computational efficiency of PRAs is also desired in the development of new technologies such as the development of Digital Twins for autonomous control of advanced small and micro reactors. Such advanced autonomous systems have to rely on risk-informed decision making in real-time which requires running hundreds of scenarios in a relatively short amount of time to evaluate the best strategy for controlling the plant. This study focuses on developing and proposing a new approach to solving such problems while improving the computational efficiency significantly. More specifically, an attempt is made to reduce the complexity for the analysis of MOEs and dependent events in fault trees. The proposed solution is just a beginning and several more scenarios will need to be considered and implemented before this approach can be used for solving true real-world applications. At the same time, it presents a significant improvements over existing implementations which makes it highly promising for additional development. The computational efficiency of the proposed approach over traditional approaches is illustrated by comparison for several different examples.

### III.2 Traditional Approaches

The response of a plant to an initiating external hazard is represented by an event tree. An event tree represents all the possible accident scenarios that will result in the failure of the plant. The frequency of the initiating external hazard is obtained from the hazard curve. The failure of each event in an Event tree corresponds to the top event of a fault tree. A fault tree is a graphical decomposition of a top event failure into intermediate events (IEs) and basic events through the use of Boolean logic gates such as an AND and OR gates. Failure of a top event in a fault tree is evaluated by identifying the minimal cut sets. The minimal cut sets are a list of sets such that the failure of even single set in the list would lead to the failure of the top event. A unique quality of minimal cut sets is that none of the cut sets in the list is a subset of another cut set. Several software [52, 88, 89], use MOCUS algorithm [80] to evaluate the minimal cut sets. The quantification of top event failure probability using the minimal cut sets is performed in two steps:

- **Step 1:** Calculation of individual minimal cut set probabilities – an individual minimal cut set is connected by an AND gate to all the basic events in it. Hence, an individual minimal cut set probability is determined by multiplying the probabilities of all the basic events in it:

$$P(CS_i) = \prod_{j=1}^{n_i} P(BE_j) \quad (\text{III.1})$$

where,  $P(CS_i)$  is the probability of minimal cut set  $i$ ,  $P(BE_j)$  is the failure probability of the  $j^{th}$  basic event in the  $i^{th}$  minimal cut set,  $n_i$  is the number of basic events in the  $i^{th}$  minimal cut set.

- **Step 2:** Combine the minimal cut sets probabilities to calculate the top event

failure probability. In current practice, three techniques are employed to find the probability for the union of the minimal cutsets, namely rare event approximation, minimal cut set upper bound, and min-max approach. The rare event approximation and minimal cut set upper bound are based on some approximations and provide conservative estimates. These approximations are often used to reduce the high computational cost that is otherwise needed to evaluate the exact probability. For most of the SSC failures in nuclear power plants, the approximate calculation of failure probabilities remain valid [84]. The min-max approach based on inclusion-exclusion rule leads to the most accurate quantification. However, it is employed only when the number of minimal cut sets are small, i.e., less than 50 in order to make the analysis computationally efficient. The formulations for the three techniques are summarized below:

***Rare event approximation:***

$$P(TE) = \sum_{i=1}^{n_{mc}} P(CS_i) \quad (\text{III.2})$$

***Upper bound approximation:***

$$P(TE) = 1 - \prod_{i=1}^{n_{mc}} (1 - P(CS_i)) \quad (\text{III.3})$$

***Min-max approach:***

$$P(TE) = \sum_{i=1}^m P(CS_i) - \sum_{i<j}^m P(CS_i \cap CS_j) + \sum_{i<j<k}^m P(CS_i \cap CS_j \cap CS_k) + \dots + (-1)^{m-1} P\left(\bigcap_{i=1}^m CS_i\right) \quad (\text{III.4})$$

where,  $n_{mc}$  is total number of minimal cutsets and  $P(CS_i)$  is the probability of the  $i$ -th minimal cutset.

### III.3 Proposed Algorithm

In this study, fault trees are represented using a general Tree data structure. Figure IV.1 (a) shows a generic fault tree and Figure IV.1 (b) shows the corresponding generic Tree data structure. The following terminologies of Tree data structure are defined to assist with the explanation of the proposed algorithms in the study.

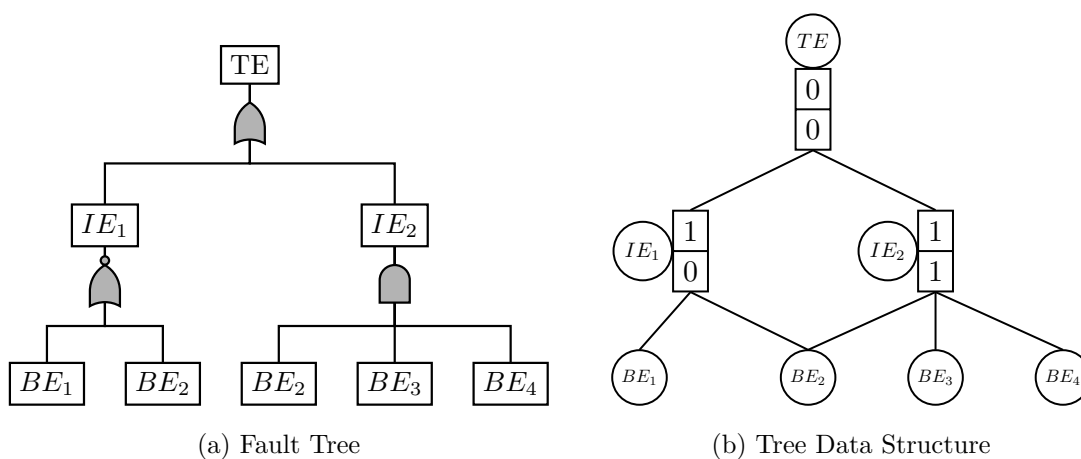


Figure III.1: Generic Fault tree and Tree data structure

#### III.3.1 Definitions and Terminology

- **Top/Root node:** the top node in a tree data structure.
- **Child node:** a node which is a descendant of any node. A node with no child nodes is called as a Leaf node.
- **Parent node:** a node which is a predecessor of any node.
- **Internal node:** a node which has at least one child node.

- **Edge:** a link which connects any two nodes.
- **Chain/Path:** a sequence of nodes and edges between two nodes.
- **Subtree:** a part of tree data structure which represents a node and all of its descendants.
- **Height of a node:** the total number of edges from a particular node to the leaf node in the longest chain. The height of all leaf nodes is zero.
- **Dependent/common node:** a node which has more than one parent.
- **Independent tree:** a tree with no dependent nodes.
- **Loop:** a loop is formed when the start node and the end node of two chains are same.
- **Logic Gate:** represents the parent/child relationship. The notation of logic gates are described in detailed in the following section.
- Nodes are represented by circles and the logic gates are denoted by rectangles. The basic event, intermediate event, and top event in the fault tree are referred to as leaf node, internal node, and top node in the Tree data structure.

### III.3.2 Compressed Truth Tables

In a fault tree, all the intermediate events and the top event are connected to the basic events through the use of logic gates as shown in Figure IV.1 (a). In a Tree data structure, the logic gates represent the parent/child relationship, and all the internal nodes have the logic gate attribute. In this study, we examine four logic gates in detail: AND, OR, NAND, and NOR. Table III.1 gives the truth table for input nodes  $A$  and  $B$  when they are connected to an internal node with any of the four logic gates. In the truth tables, the failure and success states of a node are represented using binary states 1 and 0, respectively. The two binary states for nodes  $A$  and  $B$  lead to 4 ( $2^2$ ) possible

combinations. If suppose  $n$  nodes are connected to a logic gate, then there will be  $2^n$  possible combinations of input node states. As seen in Table III.1, the highlighted output state is different from the output states of the rest of the three combinations in all the truth tables. This observation is even valid when  $n$  nodes are connected to a logic gate, i.e., output state of only one of the combinations will be different from rest of the  $2^n - 1$  combinations.

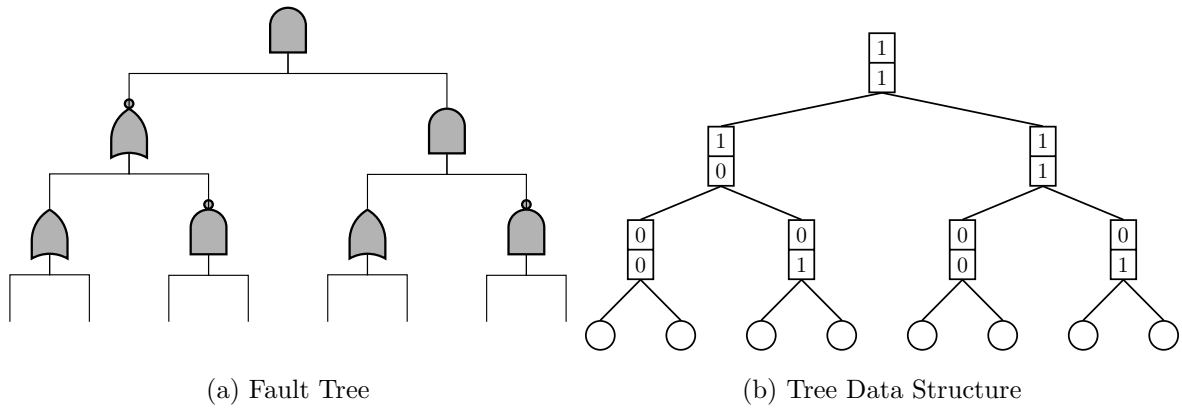


Figure III.2: Logic Gates

Table III.1: Truth Tables for logic gates with two input nodes  $A$  and  $B$  (*The term 0 and 1 in truth table translate to success and failure in fault and event trees*)

AND			NOR			OR			NAND		
$A$	$B$	$Out$	$A$	$B$	$Out$	$A$	$B$	$Out$	$A$	$B$	$Out$
0	0	0	0	0	1	0	0	0	0	0	1
1	0	0	1	0	0	1	0	1	1	0	1
0	1	0	0	1	0	0	1	1	0	1	1
1	1	1	1	1	0	1	1	1	1	1	0

**Compressed**

Output	$\boxed{1}$	Output	$\boxed{1}$	Output	$\boxed{0}$	Output	$\boxed{0}$
Input	$\boxed{1}$	Input	$\boxed{0}$	Input	$\boxed{0}$	Input	$\boxed{1}$

Furthermore, the binary states of each logic gate are mutually exclusive and collective exhaustive. Therefore, if the probability for the shaded output state ( $P_{Outputstate}$ ) of a logic gate is calculated, then the probability for its complement state can be calculated as  $1 - P_{Outputstate}$ . In this study, we compress the truth table for each logic gate in two terms: *input state* and *output state*. Table III.1 also shows the *input state* and *output state* for each of the logic gate in the Tree data structure. The logic gates in the fault tree are converted into the *input state* and *output state* in the Tree data structure and an illustration of the same can be seen in Figure IV.1 (b).

### III.4 Algorithm for Tree with Independent Nodes

The binary states probabilities of any internal node in a Tree data structure can be calculated based on its logic gate and the probability of its child nodes. The *output*

state and *input* state from the compressed truth table (Table III.1) are used to obtain these probabilities. The probabilities *output* state,  $P_{Out}(IN)$  and its complement state,  $P_{Out'}(IN)$  for an internal node are calculated using Eq. (III.5).

$$P_{Out}(IN) = \bigcap_{i=1}^{n_c} P_{Inp}(Child_i) \quad (III.5)$$

$$P_{Out'}(IN) = 1 - P_{Out}(IN)$$

where, *Out* and *Inp* are the output and input states of the logic gate corresponding to the internal node *IN*. *Out'* is the complement of the output state; i.e., if *Out* = 1 then *Out'* = 0.  $n_c$  is the total number of child nodes of the internal node *IN*.  $P_{Inp}(Child_i)$  is the probability of child *i*. Since all the nodes in the Tree data structure are independent, the intersection of their failure or success state probabilities can be simply obtained by multiplying their respective probabilities as shown in Eq. (III.6).

$$P_{Out}(IN) = \bigcap_{i=1}^{n_c} P_{Inp}(Child_i) = \prod_{i=1}^{n_c} P_{Inp}(Child_i) \quad (III.6)$$

In this study, the failure probability of the top node is obtained using a bottom-up approach. The approach is illustrated for a simple Tree data structure with independent nodes shown in Figure III.3. First, the height of all the internal nodes are calculated. The height of all the leaf nodes is zero. Height of an internal node is calculated by counting the number of edges from the internal node to a leaf node along its longest chain. For example, there are three chains from internal node  $IN_2$  to leaf nodes:  $IN_2 - IN_1 - C$ ,  $IN_2 - IN_1 - D$ ,  $IN_2 - B$ , where, the longest chain is  $IN_2 - IN_1 - C$ . Hence, the height of  $IN_2$  is equal to 2. Similarly, heights of all the other nodes are calculated

and all the internal nodes are sorted in ascending order based on the height as shown in Figure III.3. Next, the output probabilities of the internal nodes are calculated one by one in the sorted order  $[IN_1, IN_2, TN]$  as shown in Eq. (III.7) using Eq. (III.6).

$$\begin{aligned}
 P_0(IN_1) &= P_0(C) \times P_0(D), & P_1(IN_1) &= 1 - P_0(IN_1) \\
 P_1(IN_2) &= P_0(B) \times P_0(IN_1), & P_0(IN_2) &= 1 - P_1(IN_2) \\
 P_1(TN) &= P_1(A) \times P_0(IN_2), & P_0(TN) &= 1 - P_1(TN)
 \end{aligned} \tag{III.7}$$

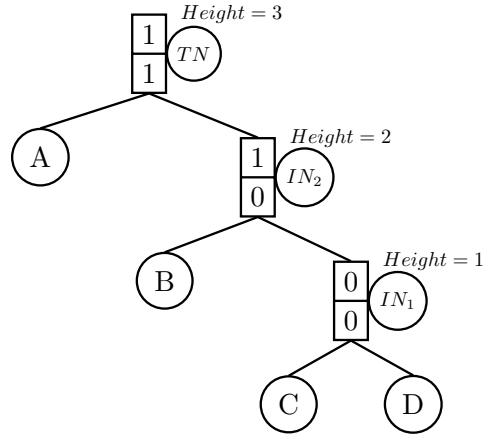


Figure III.3: A simple Tree data structure with independent nodes

We propose a generic algorithm to calculate the probabilities of all the independent nodes in a Tree data structure. The required steps for obtaining the probabilities are described below:

1. Calculate the height of all internal nodes in the Tree data structure.
2. Sort the internal nodes in ascending order based on their height.
3. Calculate the binary state probabilities of the first internal node in the sorted list based on its logic gate connection and using Eq. (III.6).

4. Repeat step 3 for all the internal nodes in the sorted list.
5. In the final step, calculate the binary state probabilities of the top node based on its logic gate connection and Eq. (III.6).

### III.5 Tree with Two Dependent Chains

The algorithm described in section III.4 is only valid for a Tree data structure with independent nodes. When dependent or common nodes are present in a Tree data structure, the intersection probabilities of node states can no longer be simply obtained by multiplying their respective probabilities as shown in Eq. (III.6). Therefore, in order to obtain the correct probability of the internal node, the intersection of its children nodes must be performed using Boolean algebra. In this study, we consider a simple Tree data structure (see Figure III.4) with only one dependent node and a single loop to illustrate the process for calculating the exact probability of top node  $TN$ . The top node probability is calculated using Eq. (III.8).

$$P_{Out}(TN) = \bigcap_{i=1}^{n_{chains}} P_{Inp}(Chain_i) \quad (III.8)$$

where,  $Out$  and  $Inp$  are the output and input states of the logic gate associated with the top node  $TN$ .

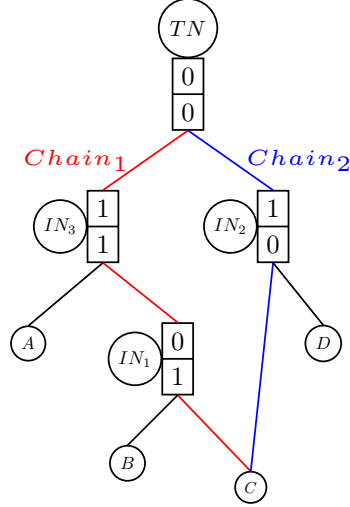


Figure III.4: A simple Tree data structure with one dependent node and single loop

The two chains in the loop are:  $C - IN_1 - IN_3 - TN$  and  $C - IN_2 - TN$ . The probability of each chain is evaluated separately using the algorithm presented in section III.4. However, the probability is represented in terms of leaf nodes so that Boolean algebra of sets can be applied to calculate the exact probability. Figure III.5 illustrates the calculation of chain 1 ( $C - IN_1 - IN_3 - TN$ ) and chain 2 ( $C - IN_2 - TN$ ) probabilities which are given in terms of leaf nodes and the final expressions are given by the following equations.

$$\begin{aligned}
 P_0(\text{Chain}_1) &= 1 - P(A) \times (1 - P(BC)) = 1 - P(A) + P(ABC) \\
 P_0(\text{Chain}_2) &= 1 - P(C' D')
 \end{aligned}
 \tag{III.9}$$

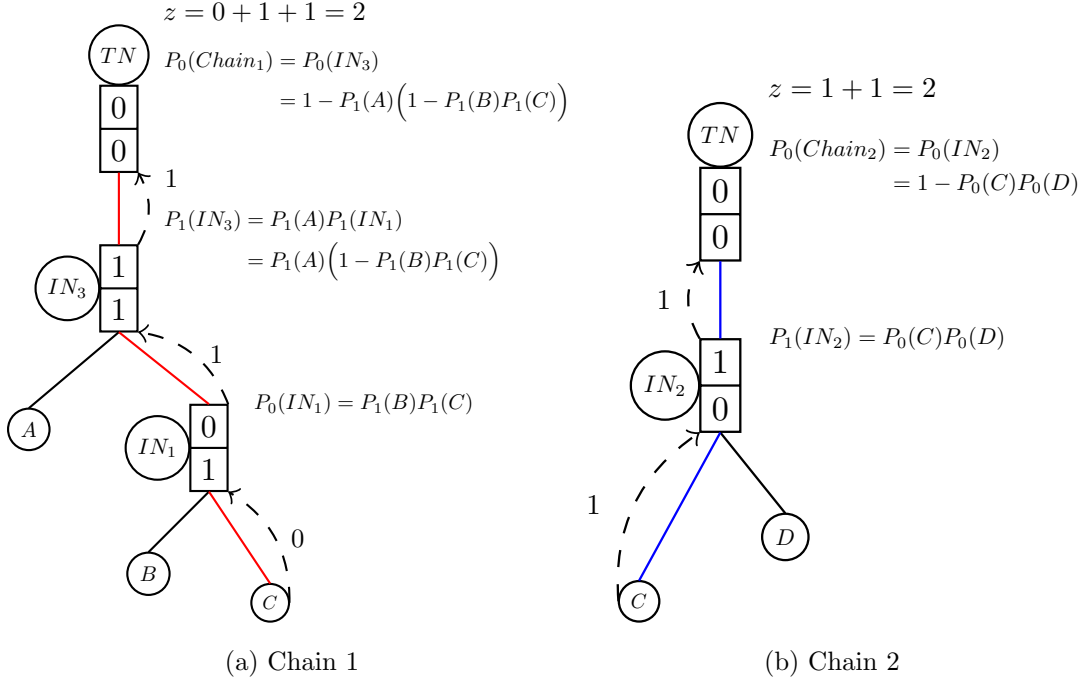


Figure III.5: Illustration of calculation of Chain probabilities

Then, the top node probability is calculated by substituting Eq. (III.9) in Eq. (III.8).

$$\begin{aligned}
 P_{Out=0}(TN) &= \prod_{i=1}^2 P_{Inp=0}(Chain_i) = P_0(Chain_1) \times P_0(Chain_2) \\
 &= (1 - P(A) + P(ABC)) \times (1 - P(C'D')) \\
 &= 1 - P(C'D') - P(A) + P(AC'D') + P(ABC) - P(ABCC'D') \\
 &= 1 - P(C'D') - P(A) + P(AC'D') + P(ABC) \quad (\because C \cap C' = 0)
 \end{aligned} \tag{III.10}$$

In this example, the  $Chain_1$  and  $Chain_2$  are dependent only on one common node  $C$ . Therefore, the intersection of chains can be performed simply in terms of the common

dependent node  $C$ . Upon further examination of Eq. (III.9) in detail, we can see that the probabilities of the chains can be expressed as a linear function of the common node's failure probability and is given by Eq. (III.11).

$$\begin{aligned}
P_0(Chain_1) &= 1 - P(A) + P(ABC) = 1 - P(A) + P(AB)P(C) \\
&= k_{1_1} + k_{2_1}P(C) \quad [k_{1_1} = 1 - P(A), k_{2_1} = P(AB)] \\
P_0(Chain_2) &= 1 - P(C'D') = 1 - P(D') \times (1 - P(C)) = 1 - P(D') + P(D')P(C) \\
&= k_{1_2} + k_{2_2}P(C) \quad [k_{1_2} = 1 - P(D'), k_{2_2} = P(D')]
\end{aligned} \tag{III.11}$$

where,  $k_{1_i}$  and  $k_{2_i}$  are the coefficients of the linear expression for chain  $i$ 's probability. Furthermore, the top node probability can also be expressed as a linear function of the common node's ( $C$ ) failure probability and is given by Eq. (III.12).

$$\begin{aligned}
P_0(TN) &= (k_{1_1} + k_{2_1}P(C)) \times (k_{1_2} + k_{2_2}P(C)) \\
&= k_{1_1}k_{1_2} + k_{1_1}k_{2_2}P(C) + k_{1_2}k_{2_1}P(C) + k_{2_1}k_{2_2}P(CC) \\
&= k_{1_1}k_{1_2} + (k_{1_1}k_{2_2} + k_{1_2}k_{2_1} + k_{2_1}k_{2_2})P(C) \quad (\because C \cap C = C) \\
&= K_1 + K_2P(C) \quad [K_1 = k_{1_1}k_{1_2}, K_2 = k_{1_1}k_{2_2} + k_{1_2}k_{2_1} + k_{2_1}k_{2_2}]
\end{aligned} \tag{III.12}$$

where,  $K_1$  and  $K_2$  are the coefficients of the linear expression for the top node ( $TN$ ) probability.

In this study, we propose a simple algorithm to calculate the coefficients  $k_{1_i}$  and  $k_{2_i}$  for a given chain  $i$  which are later used to calculate the coefficients  $K_1$  and  $K_2$  for the top node. As seen in Eq. (III.11) and Figure III.5,  $k_{2_i}$  can be calculated by multiplying the probabilities of all the leaf nodes that are connected to the internal nodes along the

chain  $i$  and is given by Eq. (III.13).

$$k_{2_i} = (-1)^z \prod_{j=1}^{n_i} P_{InpIN}(N_j) \quad (\text{III.13})$$

where,  $n_i$  is the total number of leaf nodes (except the common leaf node  $C$ ) and other internal nodes connected to the internal nodes along the chain  $i$ .  $P_{InpIN}(N_j)$  is the probability of the node  $N_j$  and the state of the node ( $InpIN$ ) is determined based on its connection to the logic gate associated with the internal node.  $z$  is the total number of state changes between a child node output state to its parent node input state along the chain. The input state of the common leaf node  $C$  is 1 (failure) and the estimation of  $z$  is illustrated in Figure III.5 using dashed arcs. Next, the term  $k_{1_i}$  is evaluated using Eq. (III.14).

$$k_{1_i} = P_{Ind}(Chain_i) - k_{2_i}P(C) \quad (\text{III.14})$$

where,  $P_{Ind}(Chain_i)$  is the probability of chain  $i$  estimated using the algorithm for the Tree data structure with independent nodes (see Eq. (III.9)).

### III.6 Tree with Multiple Dependent Chains

The algorithm proposed in the previous section for one dependent node with a single loop can even be extended to a loop with multiple chains in a Tree data structure. This is illustrated for a generic Tree data structure shown in Figure IV.2. If there are  $m$  chains in a loop with dependent node  $C$ , then the top node probability can be calculated exactly by considering the intersection between chains in terms of its dependent node  $C$  and is given by Eq. (III.15).

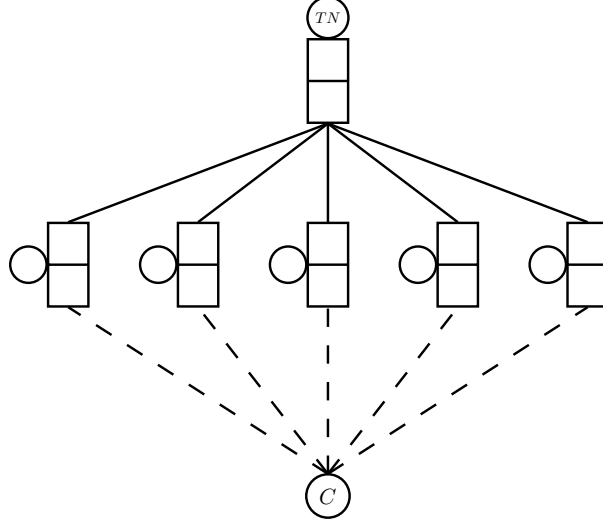


Figure III.6: A generic Tree data structure with one dependent node and multiple chains

$$\begin{aligned}
P_{Out}(TN_{dep}) &= \prod_{i=1}^m P_{Inp}(Chain_i) = P \left[ \bigcap_{i=1}^m (k_{1_i} + k_{2_i}C) \right] \\
&= \prod_{i=1}^m k_{1_i} + \left[ \sum_{j+l=m}^{j+l=m} \prod_{j,l=0}^m k_{1_j} k_{2_l} \right] P(C) \\
&= K_1 + K_2 P(C) \quad (\because C \cap \dots \cap C = C)
\end{aligned} \tag{III.15}$$

where,  $Out$  and  $Inp$  are the output and input states of the logic gate associated with the top node  $TN$ ,  $m$  is the number of chains,  $K_1 = \prod_{i=1}^m k_{1_i}$  and  $K_2 = \sum^{j+l=m} \prod_{j,l=0}^m k_{1_j} k_{2_l}$ .

As the estimation of  $K_2$  involves summation of  $2^m - 1$  terms with various combination of coefficients, it can be computationally expensive as  $m$  increases. Therefore, we propose an alternative approach to compute the coefficient  $K_2$ . The estimation of the probability of the top node without considering the intersection of the

common node is given by Eq. (III.16).

$$P_{Out}(TN_{ind}) = \bigcap_{i=1}^m (k_{1_i} + k_{2_i}P(C)) = \prod_{i=1}^m (k_{1_i} + k_{2_i}P(C)) \quad (\text{III.16})$$

The equation above is valid for various different values of  $P(C)$ . One such condition is  $P(C) = 1$  in which case the probability of the top node estimated using dependent approach  $P_{Out}(TN_{dep})$  is identical to that calculated using the independent approach  $P_{Out}(TN_{ind})$ . Therefore,  $K_2$  can be calculated using Eq. (III.17).

$$\begin{aligned} \because P_{Out}(TN_{dep}) &= P_{Out}(TN_{ind}) \quad \implies K_1 + K_2(1) = \prod_{i=1}^m (k_{1_i} + k_{2_i}(1)) \\ \implies K_2 &= \prod_{i=1}^m (k_{1_i} + k_{2_i}) - K_1 \end{aligned} \quad (\text{III.17})$$

The required steps for obtaining the probability of the top node when a Tree data structure contains a dependent node with multiple chains is summarized below:

1. For each chain, calculate the coefficients  $k_{2_i}$  and  $k_{1_i}$  using Eq. (III.13) and Eq. (III.14), respectively.
2. Calculate coefficients  $K_1$  and  $K_2$  for the combined chains:

$$K_1 = \prod_{i=1}^m k_{1_i}, \quad K_2 = \prod_{i=1}^m (k_{1_i} + k_{2_i}) - K_1 \quad (\text{III.18})$$

3. Estimate the top node probability

$$P_{Out}(TN) = K_1 + K_2P(C) \quad (\text{III.19})$$

### III.7 Simple Example Application

In this study, a simple example of an event tree is taken to illustrate the accuracy of the proposed approach. This is performed by comparing the event tree analysis with the proposed approach and exact calculations based on set theory. Figure III.7 shows an event tree with accident sequences and its corresponding fault trees of the two top events  $TE_1$  and  $TE_2$ . The failure probabilities of the basic events  $A, B,$  and  $C$  are denoted by  $P_1(A), P_1(B),$  and  $P_1(C)$ , respectively. In the event tree shown in Figure III.7 (a), sequences 2 and 3 lead to unsafe conditions at the plant. For risk assessment, it would be important to calculate the risk associated with both the accident sequences, i.e.  $P(Seq_2)$  and  $P(Seq_3)$ . In this study,  $Seq_2$  is considered to compare the accuracy of the proposed approach with exact calculations. For illustration purposes, the failure probability of basic events are chosen from the existing studies [10, 11] :  $P_1(A) = 0.125, P_1(B) = 0.08,$  and  $P_1(C) = 0.2$ .

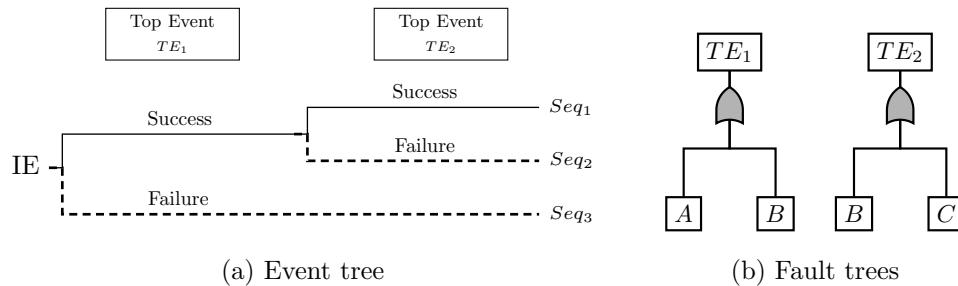


Figure III.7: Event tree and its corresponding fault trees for simple example

#### III.7.1 Exact Probability of Failure

The exact failure probability of a sequence can be calculated by using the fundamental concepts of set theory. In general, a failure sequence  $k$  would contain a total of  $n_k$  top

events which can be decomposed into  $m_k$  top events that exhibit “success” (or not a failure) and  $(n_k - m_k)$  top events that exhibit “failure”. The total probability of failure for a  $k^{th}$  sequence can be expressed using Eq. (III.20).

$$P(Seq_k) = \prod_{i=1}^{m_k} P(TE'_i) \prod_{j=m_k+1}^{n_k} P(TE_j) \quad (III.20)$$

where,  $P(TE_i)$  and  $P(TE'_i)$  represent the failure and the success of  $i^{th}$  top event, respectively. The failure probability of  $Seq_2$  shown in Figure III.7 (a) is given by Eq. (III.21).

$$P(Seq_2) = P(TE'_1) \cap P(TE_2) \quad (III.21)$$

The calculation of probabilities of top events  $TE_1$  and  $TE_2$  in terms of basic events  $A, B,$  and  $C$  can be further illustrated through the use of Venn diagrams. Figure III.8 shows the Venn diagrams for  $TE'_1, TE_2,$  and  $TE'_1 \cap TE_2$ .

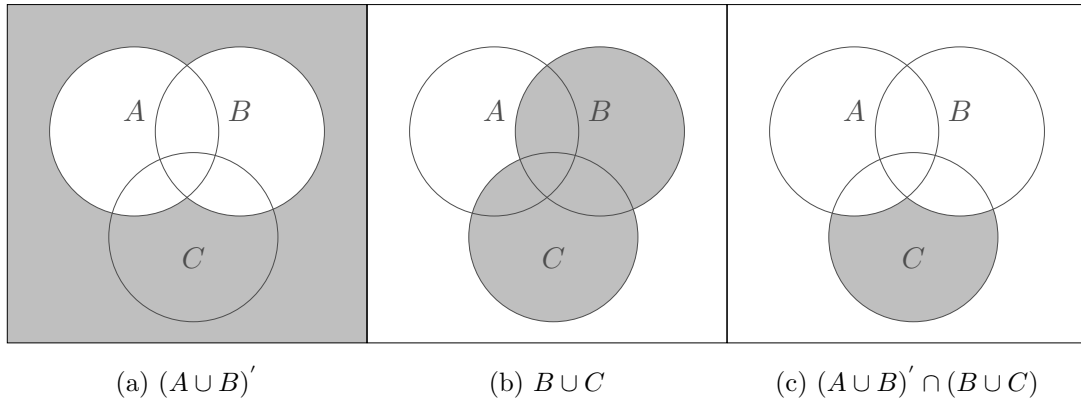


Figure III.8: Venn diagrams

The expression for Eq. (III.21) corresponding to accident sequence 2 can be written

in terms of basic events (assuming  $A \perp\!\!\!\perp B \perp\!\!\!\perp C$ ) and estimated using set theory as follows:

$$\begin{aligned}
P(Seq_2) &= P(A \cup B)' \cap P(B \cup C) \\
&= P(C) - P(C \cap A) - P(C \cap B) + P(A \cap B \cap C) \\
&= P_1(C) - P_1(A)P_1(C) - P_1(B)P_1(C) + P_1(A)P_1(B)P_1(C) \\
&= 0.20 - 0.20 \times 0.125 - 0.20 \times 0.08 + 0.20 \times 0.125 \times 0.08 = 0.161
\end{aligned} \tag{III.22}$$

### III.7.2 Proposed Methodology

In order to use the proposed methodology, the accident sequence 2 shown in Figure III.7 is converted into a Tree data structure as shown in Figure III.9 (a). The fault trees of all the top events in the accident sequence are connected to a new node with AND logic gate, which gives the failure probability of the accident sequence. However, the output states of all the success top events in the accident sequence are modified with the complement of its original state. In this example, the original output state of the  $TE_1$  is 0 (OR gate). As,  $TE_1$  appears as a success event in the accident sequence 2, its output state is changed to 1 in the calculation of probability of the accident sequence.

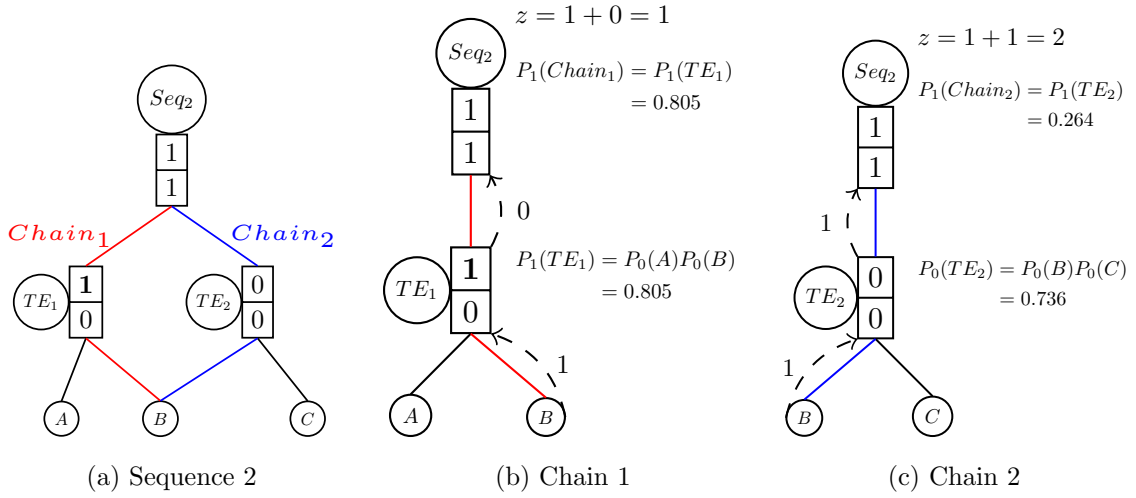


Figure III.9: Tree data structure of accident sequence 2

The Tree data structure shown in Figure III.9 (a) has a loop with two chains ( $B - TE_1 - Seq_2, B - TE_2 - Seq_2$ ) with a dependent node  $B$ . Next, the algorithm proposed in section III.6 is implemented to evaluate the accident sequence probability using the following steps:

- Calculate the probability of chains using a bottom-up approach presented in section III.4. The calculation for both the chains is illustrated in Figure III.9 (b) and Figure III.9 (c). The obtained probability of the chains are:  $P_1(Chain_1) = 0.805, P_1(Chain_2) = 0.264$ .
- Next, calculate coefficients  $k_{2_i}$  and  $k_{1_i}$  for chain  $i$  using Eq. (III.13) and Eq. (III.14), respectively. The estimation of  $z$  is illustrated in Figure III.9 (b) and Figure III.9 (c).

**Chain 1:**  $k_{2_1} = (-1)^1 P_0(A) = -0.875, \quad k_{1_1} = P_1(Chain_1) - k_{2_1} P_1(B) = 0.875$

**Chain 2:**  $k_{2_2} = (-1)^2 P_0(C) = 0.8, \quad k_{1_2} = P_1(Chain_2) - k_{2_2} P_1(B) = 0.2$

- Calculate coefficients  $K_1$  and  $K_2$  for the combined chains using Eq. (III.18).

$$K_1 = k_{1_1}k_{1_2} = 0.175 \quad K_2 = (k_{1_1} + k_{2_1})(k_{1_2} + k_{2_2}) - K_1 = -0.175$$

- Finally, estimate the accident sequence probability:

$$P_1(Seq_2) = K_1 + K_2P(B) = 0.161 \quad (III.23)$$

A comparison between Eq. (III.22) and Eq. (III.23) shows that the proposed algorithm leads to the exact quantification of the probability of failure for a given accident sequence.

### III.8 Complex Example Application

A relatively complex fault tree example is considered in this section to explain the algorithm for multiple chains with a dependent node. The fault tree shown in Figure III.10 consists of a basic event  $A$  that is common in three chains. The algorithm proposed in Section III.6 can be applied to analyze this type of fault tree.

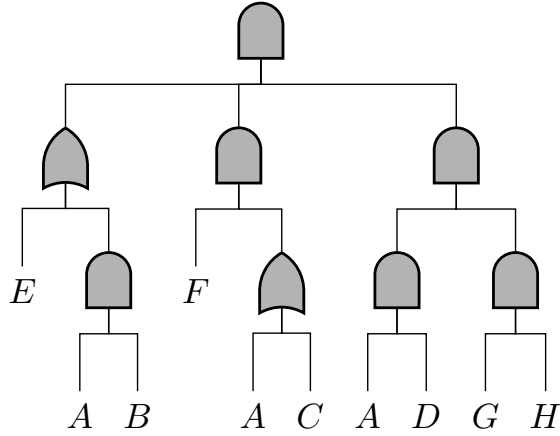


Figure III.10: Complex Fault Tree structure

First, the fault tree is analyzed using the traditional algorithms described in Section III.2 to calculate the failure probability of the top event. Next, the proposed approach is implemented and compared with the results obtained from traditional approach in order to illustrate the accuracy of the proposed approach.

The fault tree shown in Figure III.10 consists of 7 intermediate events with their logic connection and 8 basic events. The probability of failure for all basic events are shown in Table III.4. The failure probabilities of the basic events have been chosen randomly for the illustration purpose. In Table III.4,  $P_1()$  and  $P_0()$  denotes the failure and the success probability of events.

Table III.2: Probability of failure and success for each leaf node

	A	B	C	D	E	F	G	H
$P(1)$	0.005	0.023	0.076	0.0039	0.0164	0.134	0.098	0.00754
$P(0)$	0.995	0.977	0.924	0.9961	0.9836	0.866	0.902	0.99246

### III.8.1 Solution with Traditional Approach

In the traditional approach, first the minimal cut sets are obtained using the MOCUS algorithm. The MOCUS algorithm for the top event failure of the fault tree shown in Figure III.10 leads to two minimal cut sets:  $ABDFGH$  and  $ADEFGH$ . For simple systems such as that shown in Figure III.10, an exact failure probability can be calculated directly using min-max approach (Eq. (III.4)) without the computational approximation.

$$\begin{aligned} P_1(TE) &= \bigcup_{i=1}^2 P(CS_i) = P(ABDFGH \cup ADEFGH) \\ &= P(ABDFGH) + P(ADEFGH) - P(ABDEFGH) = 7.5345E - 11 \end{aligned} \quad (III.24)$$

### III.8.2 Solution with Proposed Algorithm

The fault tree shown in Figure III.10 is converted to the Tree data structure using truth tables as shown in Figure III.11 to implement the proposed algorithm. The leaf node  $A$  is common to the three chains that converge at the top node.

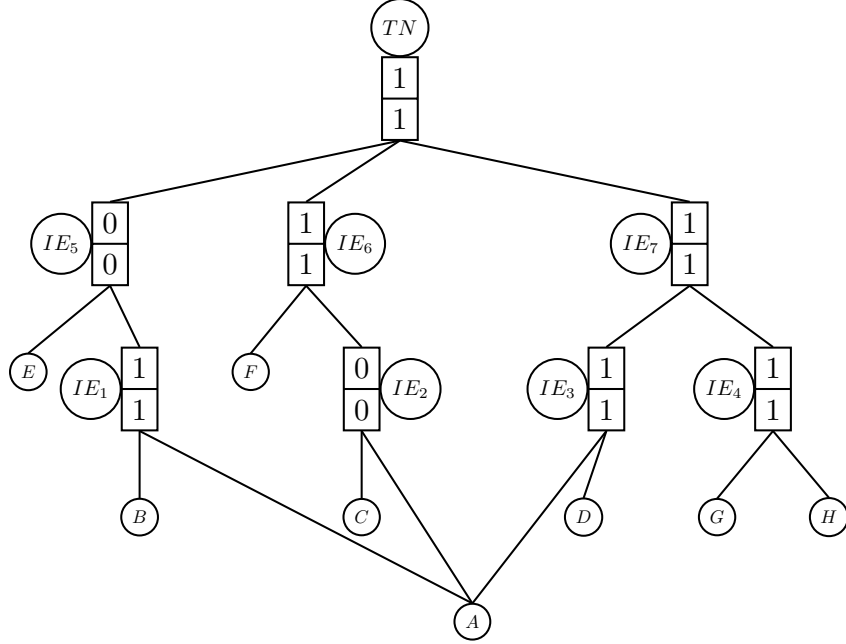


Figure III.11: Tree data structure for the fault tree shown in Figure III.10

First, all the dependent chains are analyzed separately using the algorithm for independent nodes. The top node probability for each independent chain can be calculated as shown in Eq. (III.25).

$$\begin{aligned}
 P_1(Chain_1) &= 1 - P_0(E)(1 - P_1(B)P_1(A)) = 0.016513114 \\
 P_1(Chain_2) &= P_1(F)(1 - P_0(C)P_1(A)) = 0.01080308 \\
 P_1(Chain_3) &= P(G)P(H)P(D)P(A) = 1.44089E - 08
 \end{aligned}
 \tag{III.25}$$

Second, the coefficients  $k_{1_i}$  and  $k_{2_i}$  are calculated for  $i^{th}$  chain using Eq. (III.13) and Eq. (III.14), respectively. These values are listed in Table III.3. The calculation for the term  $z$  in the expression of  $k_{2_i}$  in accordance with Eq. (III.13) is illustrated in Figure III.12.

Table III.3: Calculation of  $k_{1_i}$  and  $k_{2_i}$  for complex example with three chains

	Chain 1	Chain 2	Chain 3
$TN$	0.016513114	0.01080308	1.44089e-08
$k_2$	$(-1)^2 P_0(E)P(B)$	$(-1)^2 P(F)P_0(C)$	$(-1)^0 [P(GH)]P(D)$
	0.0226228	0.123816	2.88179E-06
$k_1 = TN - k_2 P(A)$	0.0164	0.010184	0
$k_1 + k_2$	0.0390228	0.134	2.88179E-06

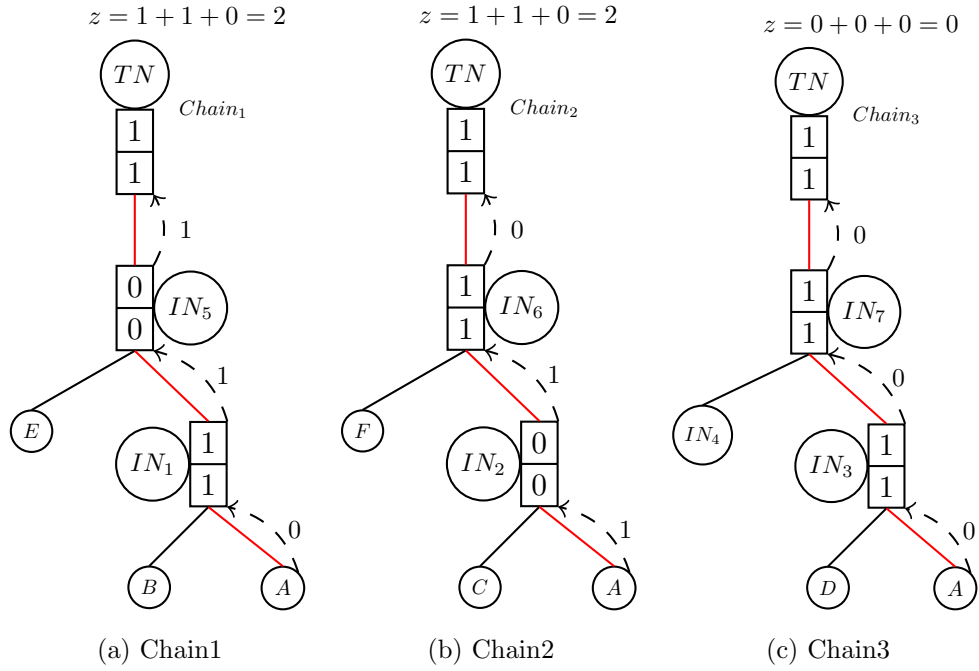


Figure III.12: Calculation of  $z$  for all the chains

Finally, the top node probability for the output state can be calculated using Eq. (III.19). The top node failure probability calculated using the traditional approach

and given in Eq. (III.24) is identical to that obtained using the proposed approach.

$$P_1(TE) = K_1 + K_2 P_1(A) = 7.53451E - 11 \quad (\text{III.26})$$

$$\text{where, } K_1 = \prod_{i=1}^3 k_{1_i} = 0, K_2 = \prod_{i=1}^3 (k_{1_i} + k_{2_i}) - K_1 = 1.5069E - 08$$

### III.9 Computational Efficiency

In this section, the computational efficiency of the proposed algorithm is compared to that of a traditional approach. To do so, relatively larger sized fault trees are considered and run times are calculated for both approaches. At first, three sets of fault trees are considered. The geometric configuration of each set such as width, height, total number of events, number of basic events is kept constant. These details are provided in Table III.2. For a given geometric configuration, multiple fault tree cases can be generated by varying the connectivity and the type of gates. Each case will result in a different computation time.

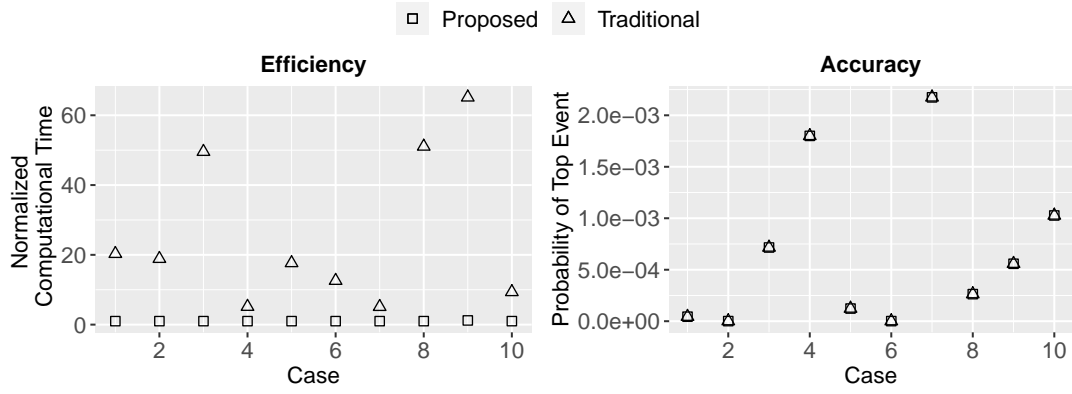
Table III.4: Details of the three set of fault tree structure

Set no.	Width (no. of BEs)	Height	Total events	Total no. of cases
Set 1	34	6	44	10
Set 2	117	6	132	10
Set 3	70	11	90	20

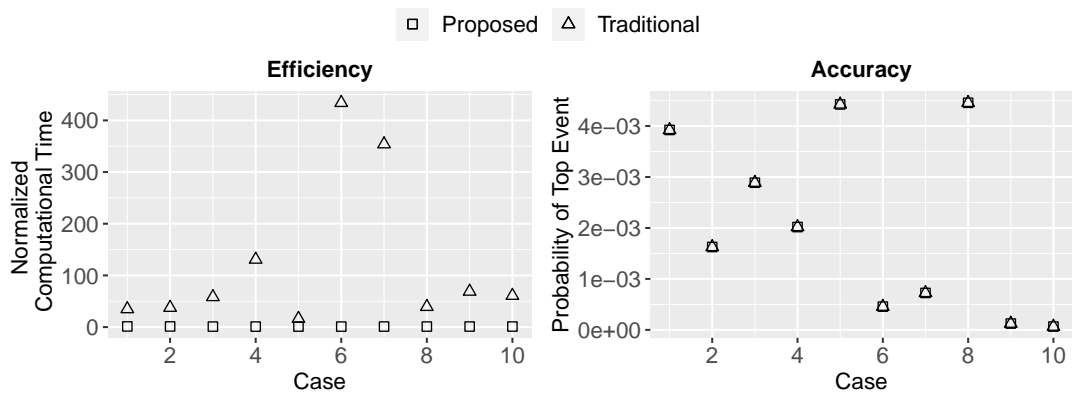
The fault trees considered in this section are analyzed with a traditional approach that utilizes MOCUS and upper bound approach, as discussed earlier in this chapter. The

true computation run time or CPU time is highly dependent upon the type of hardware used. Therefore, for a hardware independent comparison, run times are normalized with respect to the run time for the proposed method. This assists with visualization of relative difference in the computation times using the two approaches. The normalized run times for each case in the three sets are compared in Figure III.13. These figures also compare the failure probability of top event for each case as evaluated from the two methods for the illustrating the accuracy of the proposed approach. As seen in these figures, the run times of traditional approach can be significantly high in some cases depending upon the type and connectivity of different logic gates in the fault tree. An important aspect that is not directly evident in the figures is the run time for proposed approach remains constant for various different cases of a given set. It changes from one set to another. In other words, unlike the traditional approach, the run time for the proposed approach is not dependent upon the type of logic gates and their connectivity within a given fault tree. It is also important to note that for all cases and each set, the failure probabilities of top node are identical from the two approaches.

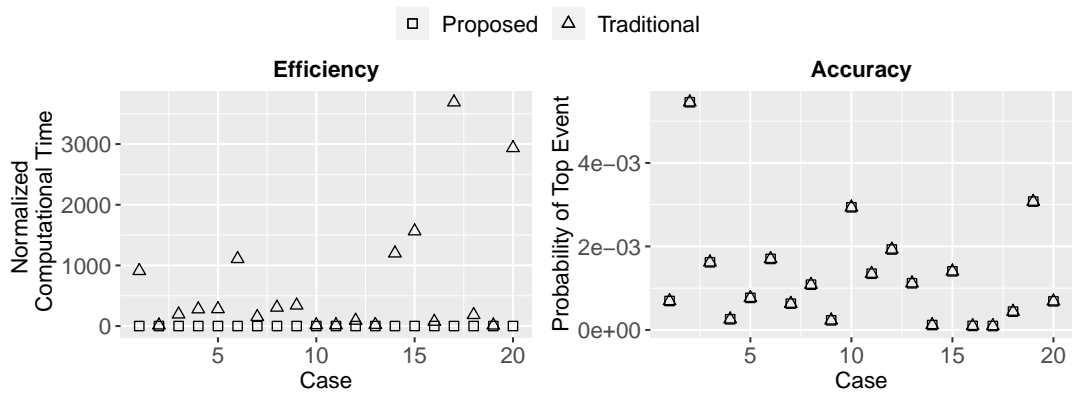
Next, we take a relative simpler case of connectivity and logic gate types but increase the size of fault trees by increasing the number of basic events. Figure III.14 shows such a fault tree that consists only three intermediate OR gates that are connected by an AND gate to give the top event. Each OR gate can be connected to multiple basic events. As the number of basic events increase, the computing run time increases. This example allows computation of exact values for failure probabilities of top event without any reliance on approximation of the min-max approach that has been used for traditional approaches in the previous examples described above. For such an example, the computational time is an exponential function of the number of minimal cut sets [52].



(a) Set 1



(b) Set 2



(c) Set 3

Figure III.13: Comparison of efficiency and accuracy of the proposed algorithm compared to traditional approach

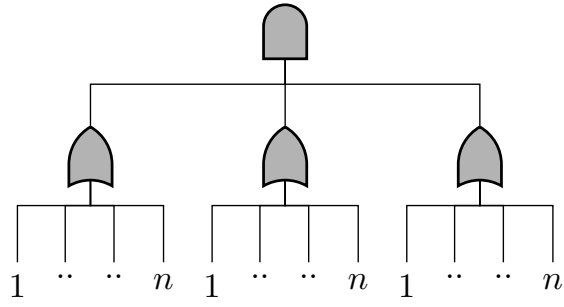


Figure III.14: A typical fault tree structure used for comparison of computational efficiency

Figure III.15 compares the computational run time in millisecond (ms) from the two approaches. It shows that the computational time for the proposed approach is increases linearly whereas it increases exponentially for the traditional approach. This comparison shows a significant improvement in the computational efficiency that can be achieved by the proposed algorithm.

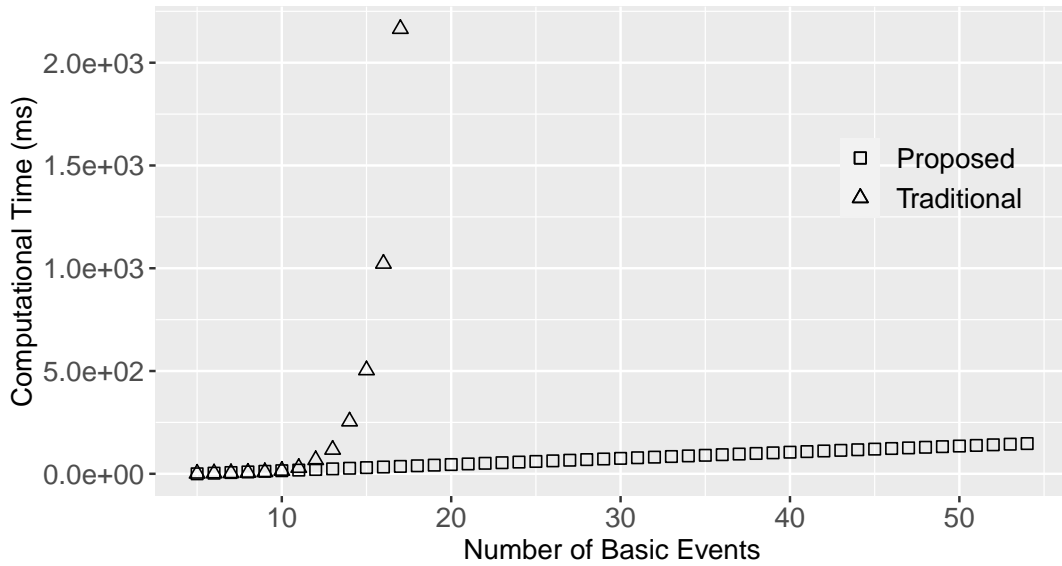


Figure III.15: Comparison of computational time between traditional and proposed approach for the fault tree structure in Figure III.14

### III.10 Conclusions

Fault and event trees are used for probabilistic risk assessment of nuclear power plant systems. A fault and event trees analysis for a power plant requires modelling of hundreds of component failures, logic gates, multiple occurring events, and dependent events. Such interconnection for large networks can lead to excessive computational demand. Most of the traditional methods address computational demand by using assumptions or rely on high performance computing facilities that allow implementations of parallel computing. This study presents a novel module based approach to address the computational demands of a PRA. The proposed algorithm is developed for networks with dependent nodes and multiple chains connecting the dependent nodes. The study converts logic gates into corresponding truth tables to achieve the desired efficiency. The study also presents application of the proposed algorithm to various simple and complex examples for illustrative purposes. Two types of network configurations are used to illustrate the computational efficiency of proposed algorithm. First, networks with same number of basic events and intermediate events are considered, i.e., networks with fixed height and width. For such networks, the computational demand of traditional approaches changes based on different connectivity of events and gates. For such networks, the computational demand of proposed algorithm remains constant. In addition, the computational demand of proposed algorithm can be less than that of traditional approach by an order of magnitude for some cases. Second, networks with fixed height and fixed logic gates (intermediate events) but varying widths are considered by increasing the number of basic events. For such networks, it is illustrated that the computational demand of proposed algorithm increases at a rate that is less than linear function of the number of

basic events. In contrast, the computational demand of traditional approach increases exponentially as  $O(2^n)$ . It is also shown that the proposed algorithm gives identical results to those obtained by the traditional approaches and, for a simple case, identical to that calculated using fundamental concept of set theory. While the proposed algorithm is a significant improvement of the currently available techniques, it has not been applied to and explored for many different scenarios that exist in a real PRA network. The proposed algorithm needs to be enhanced for a few such scenarios such as consideration of Common Cause Failures and  $n/m$  gates [52, 90]. Such improvements are recommended for future studies.

---

---

**PART IV**

---

**Computationally Efficient Approach for Multi hazard  
and Multi Unit Probabilistic Risk Assessment**

## IV.1 Introduction

The accident at Fukushima Daiichi has highlighted the importance of risk from external hazards associated with multiple nuclear reactor units and multiple hazards. As a result, nuclear research community has given considerable attention to multi-unit and multi-hazard probabilistic risk assessment (PRA) in the past few years. A multi-unit or multi-hazard PRA computes measures of risk and identifies the contributors to risk by conducting PRA for the entire site including multiple units. In such a case, any possible unit-to-unit interactions and dependencies should be modelled and accounted for. Typically, the the assessment of entire site for multi hazard leads to highly complex dependencies among the component failures. The fault and event tree employed for a regular probabilistic risk assessment of nuclear power plants consist of hundreds of components failures. Even for conventional single hazard PRA, the traditional fault and event tree analysis relies on assumptions, approximations or high performance computing resources that allow implementations of parallel computing. The computations for multi unit and multi hazard PRA with complex dependencies and multiply occurring events can lead to excessively large computational demand, making the traditional approaches impractical for such computations.

Chapter III proposed a novel framework based on modules to address the computational demand in the assessment of fault and event trees in PRA. In this framework, a bottom-up approach is proposed for analyzing FT/ET networks with dependent node and multiple chains connecting the dependent node. The accuracy and efficiency of the approach is illustrated through comparisons with traditional approaches for fault and event trees analysis. The approach presented in chapter III

shows a significant improvement of computational efficiency compared to a traditional fault trees analysis without compromising the accuracy. However, the algorithm has not been applied to and explored for many different scenarios that exist especially in the context of multi unit and multi hazard PRA.

In this manuscript, we propose major enhancements to the existing framework proposed in chapter III. The proposed enhancements are intended to address complex dependencies exhibited in the assessment of multi unit and multi hazard PRAs. More specifically, the proposed enhancements consider multiple dependent nodes, common cause failure events,  $n/m$  logic gates, and nested loops that can exist when one initiating event leads to more than one failure paths. The advantages of the proposed modifications are illustrated through application particularly to illustrate the efficiency of proposed approach as compared to a traditional approach.

## **IV.2 Summary of the Existing Framework**

The framework presented and described in chapter III consists of three parts. In the first part, a fault tree is converted to a Tree data structure. The second part presents a bottom-up approach for the quantification of a fault tree with independent events. The third part proposes an efficient algorithm for the quantification of a fault tree modules with one dependent event.

### **IV.2.1 Part 1: Conversion of Fault Tree to Tree Data Structure**

Figure IV.1 shows a generic fault tree and its corresponding Tree data structure. The basic event, intermediate event, and top event in the fault tree are referred to as leaf node, internal node, and top node in the Tree data structure. The logic gate represents the

relationship between parent node (predecessor of any node) and child nodes (descendant of any node). The logic gates in the fault tree are converted into a compressed truth table with an input state and an output state as shown in Table IV.1 in the Tree data structure. A sequence of nodes and edges between two nodes form a chain or a path. In a Tree data structure, a node with more than one parent node is called a dependent node and a loop is formed when the start node (dependent node) and the end node (typically the top node) of two chains are same.

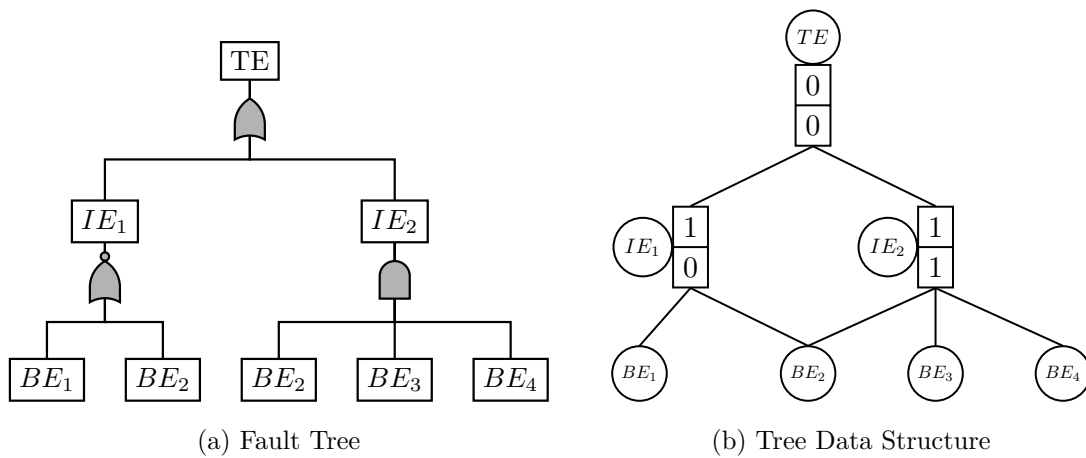


Figure IV.1: Generic Fault tree and Tree data structure

Table IV.1: Details of the three set of fault tree structure

Gate name	AND	OR	NAND	NOR
Output State	1	0	0	1
Input State	1	0	1	0

### IV.2.2 Part 2: A Bottom-up Approach for Tree Data Structure with Independent Nodes

The failure probability of the top node in an independent Tree data structure (no dependent nodes) is computed using a simple bottom-up approach. First, the internal nodes (including the top node) are sorted in ascending order based on their height which is the total number of edges from the given internal node to a leaf node along its longest chain. Then, the probabilities of the internal nodes are calculated in the sorted order using Eq. (IV.1).

$$P_{Out}(IN) = \prod_{i=1}^{n_c} P_{Inp}(Child_i) \quad (IV.1)$$

$$P_{Out'}(IN) = 1 - P_{Out}(IN)$$

where, subscript *Out* and *Inp* represents the output and input states of the logic node gate associated with an internal node *IN*. *Out'* is the complement of the output state of the internal node i.e., if  $P(Out) = 1$  then  $P(Out') = 0$ .  $n_c$  is the total number of child nodes of the internal node *IN*.  $P_{Inp}(Child_i)$  is the probability of *Child<sub>i</sub>*

### IV.2.3 Part 3: Complex Tree Structure with a Dependent Node

The probability of the top node in a Tree data structure with a single dependent node *X* and a loop with multiple chains as shown in Figure IV.2 is computed by estimating the probability of each chain as a linear function of the common dependent node as shown in Eq. (IV.2).

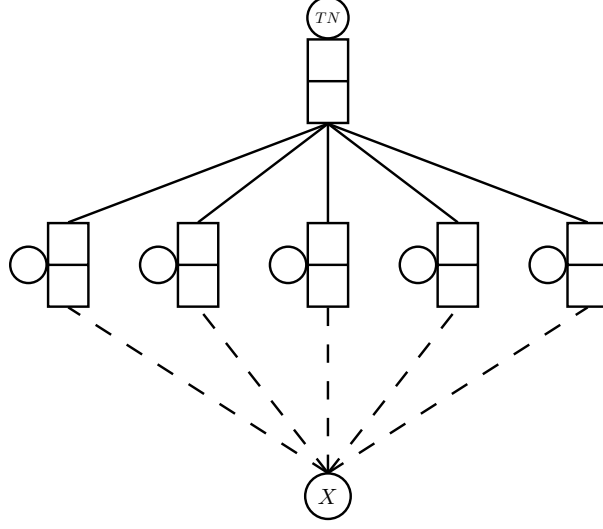


Figure IV.2: A generic Tree data structure with one dependent node and multiple chains

$$P_{Inp}(Chain_i) = P_{Inp}(IN_{last}) = \prod_{j=1}^{m_{IN_{last}}} P_{Inp_{IN_{last}}}(Child_{j_{IN_{last}}}) \quad (IV.2)$$

$$P_{Inp}(Chain_i) = k_{1_i} + k_{2_i} P_1(X)$$

where,  $IN_{last}$  is the internal node of the chain  $i$  at the maximum height in that chain,  $m_{IN_{last}}$  is the number of children of  $IN_{last}$ ,  $Inp_{IN_{last}}$  is the input state of the  $IN_{last}$ ,  $Child_{j_{IN_{last}}}$  is  $i^{th}$  child of the internal node  $IN_{last}$ ,  $k_{1_i}$  and  $k_{2_i}$  are the coefficients of chain  $i$ .  $P_1(X)$  is the failure probability of common node  $X$ . The coefficients  $k_{1_i}$  and  $k_{2_i}$  are calculated using following equations:

$$k_{2_i} = (-1)^z \prod_{j=1}^{n_i} P_{Inp_{IN}}(N_j) \quad (IV.3)$$

$$k_{1_i} = P_{Inp}(Chain_i) - k_{2_i}$$

where,  $n_i$  is the total number of children of leaf nodes (except the common leaf node  $X$ ) and other internal nodes connected to the internal nodes along the chain  $i$ .  $P_{InpIN}(N_j)$  is the probability of the node  $N_j$  and the state of the node is determined based on its connection to the logic gate associated with the internal node.  $z$  is the total number of state changes between a child node output state to its parent node input state along the chain.  $P_{Ind}(Chain_i)$  is the probability of the top node in the chain  $i$  and is estimated using the algorithm for independent nodes presented in the previous section. The probability of the top node is estimated by considering the intersection between chains and is given by Eq. (IV.4).

$$\begin{aligned} P_{Out}(TN_{dep}) &= \bigcap_{i=1}^m P_{Inp}(Chain_i) = \bigcap_{i=1}^m (k_{1_i} + k_{2_i}P(X)) \\ &= K_1 + K_2P(X) \end{aligned} \quad (IV.4)$$

where,  $Out$  and  $Inp$  are the output and input states of the logic gate associated with the top node,  $m$  is the number of dependent chains,  $K_1$  and  $K_2$  are calculated as shown below,

$$K_1 = \prod_{i=1}^m k_{1_i}, \quad K_2 = \prod_{i=1}^m (k_{1_i} + k_{2_i}) - K_1 \quad (IV.5)$$

### IV.3 Enhancements to the Existing Approach

The implementation of the existing approach is limited to trees with only one dependent event. However a multi unit and multi hazard accident scenario requires the modelling of multiple dependent events. Therefore, the existing methodology has been extended in this manuscript for the implementation to more complex scenarios that are often

observed in multi-unit and multi-hazard PRAs. More specifically, the three different aspects incorporated are:

1. Nested loops - This scenario is commonly observed in a multi-hazard risk PRAs when a failure of one component can lead to multiple failure paths. For example, an earthquake could initiate a fire or a flood or even a multi-unit failure. Although rare, single hazard PRAs or individual unit PRAs can also exhibit such complex dependencies. Such a scenario results in nested loops in the fault or event trees.
2. Consideration of common cause failures - The condition wherein the component failures are not statistically independent is typically referred to by “common cause failure (CCF)”. CCF events have gained significant attention in PRAs because their presence can lead to failure of multiple components simultaneously. Enhancements are proposed in this manuscript to consider CCF events.
3.  $N$  of the  $M$  input ( $n/m$ ) logic gates - This gate represents a specialized and somewhat rare event that may take place when a few different component failures result in progression of failure in the system analysis. This type of gate simplifies the definition of the logic for situations where a failure of a system or subsystems is defined as the combination of  $M$  failure out of  $N$  possible failures at any time. This type of gate is commonly considered in nuclear PRA software [52].

#### **IV.4 Nested Loop Structure**

The algorithm presented in the preceding section is valid for cases in which the chains are either independent or all the loops start from a single dependent node. Enhancement to the algorithm is needed for multi-hazard scenarios shown in Figure IV.3 that can have more than one dependent event leading to nested loops. As seen in the figure,

$C_{ol}$  and  $C_{il}$  are the two dependent nodes that result in nested loops. The inner loop is  $TN_{il} - A - C_{il} - B$  and the outer loop is  $TN_{ol} - D - E - F - C_{ol} - C_{il} - A/B - TN_{il}$ . The existing algorithm does not work because the probability of any chain in Eq. (IV.4) is calculated in terms of probability associated with only one dependent node that is common to more than one chain. In the case of a nested loop, the top event probability is dependent on two or more dependent nodes such as  $C_{il}$  and  $C_{ol}$  in the Figure IV.3.

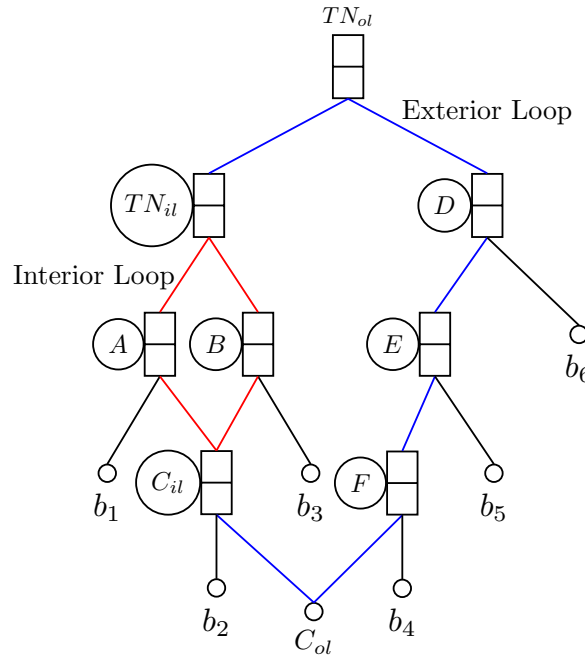


Figure IV.3: A generic fault case of loop inside loop in a Tree data structure

where,  $TN_{ol}$  and  $TN_{il}$  are the top nodes of outer and inner loop, respectively.

The proposed solution for such a scenario is based on the concept of replacing each inner loop with imaginary independent nodes in the tree data structure. The properties of an imaginary node are same as other internal or leaf nodes of the tree data structure. However, it does not represent any actual or real event in the PRA. Furthermore, it must

be noted that the probability of failure or success for an imaginary node can be either negative or more than 1. At first, this concept may appear strange. However, it can be explained as follows. Let us consider two sets  $A$  and  $B$ . The probability of their union is equal to the sum of their individual probabilities minus the probability of their intersection. One can replace the probability of their addition by an imaginary node and this sum may have a value greater than 1. Therefore, it is termed as “imaginary” node.

It can be observed that the inner loop consists of only one dependent node  $C_{il}$ . If the inner loop is analyzed separately by itself then its top event probability can be given using Eq. (IV.4) as  $K_{1_{il}} + K_{2_{il}}P(C_{il})$  where  $K_{1_{il}}$  and  $K_{2_{il}}$  are calculated specifically for the inner loop. Once the inner loop has been analyzed, only two nodes ( $TN_{il}$  and  $C_{il}$ ) can be used to establish a connection with the outer loop. Thus, the inner loop can be replaced by an independent chain that consists of  $TN_{il}$  and  $C_{il}$  with additional imaginary nodes to represent the properties of the entire inner loop. The independent chain is created such that the end node of this chain is  $C_{il}$  with a failure probability of  $P(C_{il})$  and the top node is  $TN_{il}$  with probability as  $K_{1_{il}} + K_{2_{il}}P(C_{il})$ . In order to create such a chain with imaginary nodes, two properties of existing approach for independent node as explained in Section 2 are recalled:

1. The probability of the output state of the parent is multiplication of probability of its children at its input state (Step 1, Eq. (IV.4)). This property allows top event probability to be expressed as the multiplication of probabilities for the two imaginary nodes.
2. When the output state of children and input state of parent are not same, then the complement of children probability is multiplied (Step 1, Eq. (IV.4)). The complement of a child's probability is given by  $(1 - C_{il})$ .

Based on the above properties, the inner loop top node probability,  $K_{1_{il}} + K_{2_{il}}P(C_{il})$ , can be replaced by either as the multiplication of two imaginary nodes' probabilities or the complement of imaginary nodes' probabilities. To facilitate this conversion, the top event probability  $P(TN_{il})$  of the inner loop is re-written below as shown in Eq. (IV.6)

$$P(TN_{il_{out}}) = K_{1_{il}} + K_{2_{il}}P(C_{il}) = K_{1_{il}} \times \left(1 - \left(\frac{K_{2_{il}}}{-K_{1_{il}}}\right) P(C_{il})\right) \quad (IV.6)$$

The re-structured equation for  $TN_{il_{out}}$  is a multiplication of two imaginary nodes whose probabilities are given by  $K_{1_{il}}$  and  $\left(1 - \left(\frac{K_{2_{il}}}{-K_{1_{il}}}\right) P(C_{il})\right)$ . The first imaginary child is simply a leaf node with its probability at  $TN_{il}$ 's input state as  $K_{1_{il}}$  whereas the second imaginary child is a internal node with its probability at  $TN_{il}$ 's input state as  $\left(1 - \left(\frac{K_{2_{il}}}{-K_{1_{il}}}\right) P(C_{il})\right)$ , which is the complement of a node's probability given by  $\left(\frac{K_{2_{il}}}{-K_{1_{il}}}\right) P(C_{il})$ . Therefore, the probabilities of the children of the imaginary internal nodes are given by  $\left(\frac{K_{2_{il}}}{-K_{1_{il}}}\right)$  and  $P(C_{il})$ . A conversion of inner loop into independent chain is demonstrated through Eq. (IV.7) and Figure V.4.

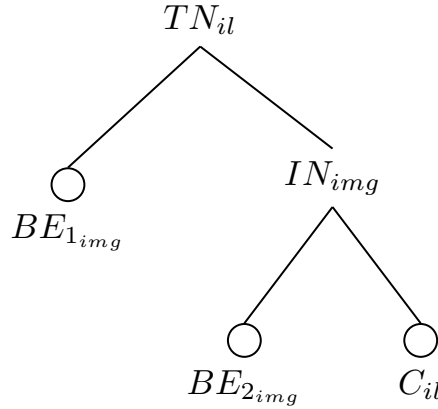


Figure IV.4: Conversion of a loop to independent chain

$$\begin{aligned}
P_{Inp}(BE_{1img}) &= K_{1il} \\
P_1(BE_{2img}) &= -\frac{K_{2il}}{K_{1il}} \\
P_{1-Inp}(IN_{img}) &= P_1(BE_{2img}) \times P_1(C_{il}) = -\frac{K_{2il}}{K_{1il}} \times P_1(C_{il}) \\
P_{Out}(TN_{il}) &= P_{inp}(BE_{1img}) \times P_{inp}(IN_{img}) \\
&= K_{1il} + K_{2il} P_1(C_{il})
\end{aligned} \tag{IV.7}$$

where,  $BE_{1img}$  and  $BE_{2img}$  are two new imaginary leaf nodes and  $IN_{img}$  is imaginary internal node inserted in the tree data structure to replace the inner loop.

The conversion of inner loop to an independent chain in a tree data structure is illustrated in Figure V.5. The tree data structure in the figure consists of only one loop with one common node. Hence, it can be analyzed using existing approach for dependent nodes as presented in Section 2.

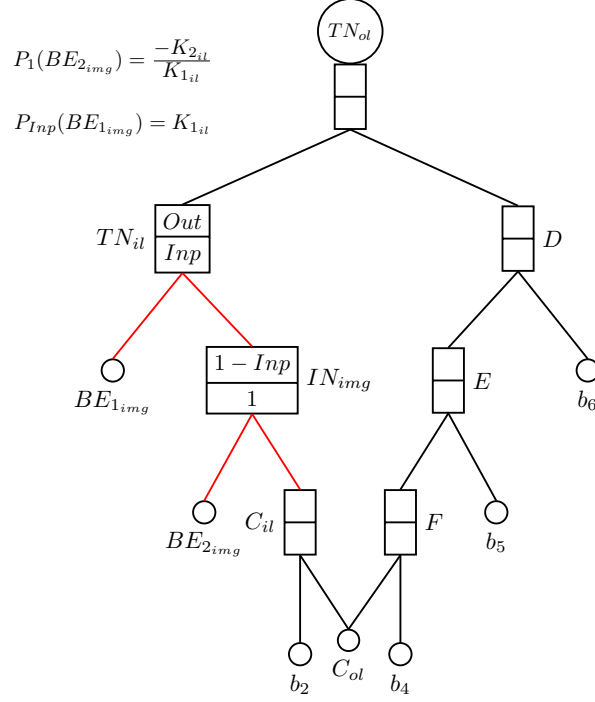


Figure IV.5: Conversion of a generic loop inside loop to a Tree data structure with one loop

However, the expression for  $P(BE_{2img})$  in Eq. (IV.7) can lead to an invalid probability when the numerical value of  $K_{1il}$  is zero. When this happens, Eq. (IV.7) and Figure V.5 would not be valid for the conversion of loop to an independent chain. In such a case, the top event probability can be expressed as follows:

$$P(TN_{il_{out}}) = K_{2il}P(C_{il}) = K_{2il} \times P(C_{il}) \quad (IV.8)$$

where,  $TN_{il_{out}}$  is simply a multiplication of  $K_{2il}$  and  $P(C_{il})$ . It implies that the top event ( $TN_{il}$ ) of inner loop can be replaced with two imaginary children whose probability at top event's input state is given by  $K_{2il}$  and  $P(C_{il})$ . This conversion has been illustrated

through Figure V.6 below.

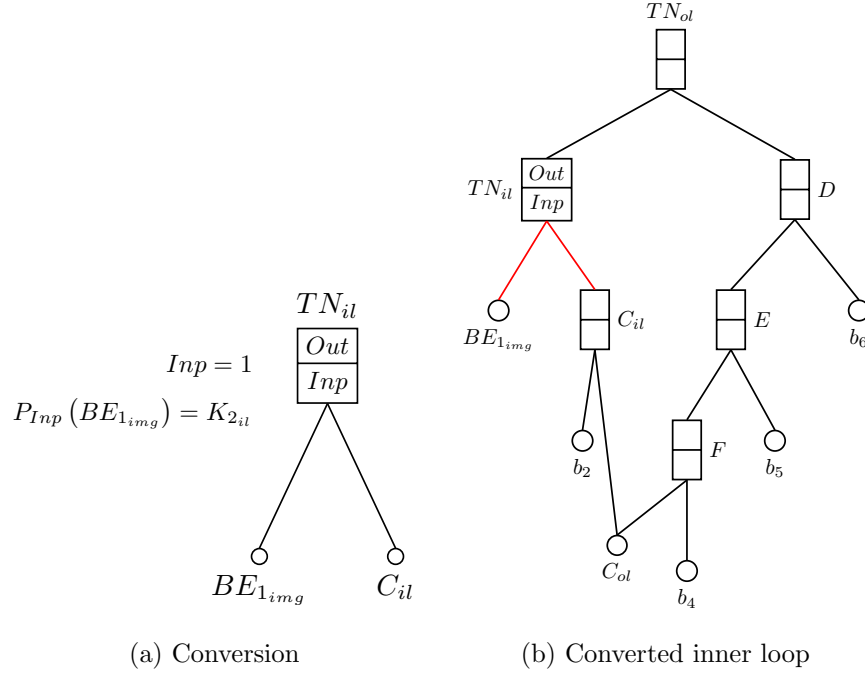


Figure IV.6: Conversion of inner loop to independent chain when  $K_1$  is equal to 0

The following steps give a summary of proposed approach that can be used for each  $j^{th}$  nested loop till only the outer loop is left in the tree data structure.

1. Analyze and calculate the top event probability  $P(TN_{il_j})$  of the  $j^{th}$  inner most loop as  $K_{1_j} + K_{2_j}P(C_{il_j})$ . The Eq. (IV.4) of the existing approach can be used to calculate the  $K_{1_j}$  and  $K_{2_j}$  for the  $j^{th}$  inner loop, while retaining the common node  $P(C_{il_j})$  and the top node  $P(TN_{il_j})$  of the  $j^{th}$  inner loop in the  $(j - 1)^{th}$  loop.
2. Calculate the properties of imaginary nodes as explained below to replace the  $j^{th}$  inner loop:
  - (a) If  $K_{1_j} = 0$ , the inner loop's internal nodes are replaced by one imaginary leaf

node. The probability of the imaginary leaf node  $P(BE_{1_{img}})_j$  is given by  $K_{2_j}$ . Another child of the top node is  $C_{il_j}$ .

- (b) When  $K_{1_j} \neq 0$ , the inner loop's internal nodes are replaced with one imaginary leaf node  $(BE_{1_{img}})_j$  and one imaginary internal node. The probabilities for these two nodes are given by  $K_{1_j}$  and  $\left(1 - \left(\frac{K_{2_j}}{-K_{1_j}}\right) P(C_{il_j})\right)$ , respectively. The children of the imaginary internal node are imaginary leaf node  $(BE_{2_{img}})_j$  and common node  $C_{il_j}$ . The imaginary node  $(BE_{2_{img}})_j$ 's probability is given by  $\frac{-K_{2_j}}{K_{1_j}}$ .

3. Finally calculate the top node probability of the  $(j - 1)^{th}$  inner loop by repeating step 1 and step 2 process until all the loops in the system are analyzed. After the replacement of all the loops into independent chains, the tree data structure can be analyzed using Part 1 of the Section 2.

#### IV.4.1 Application Example

A fault tree is considered to illustrate the application of proposed algorithm for nested loops. The fault tree shown in Figure V.7 consists of nested loops  $TN_{il} - IN_1 - C_{il} - IN_2$  inside the outer loop  $TN_{ol} - IN_3 - C_{ol} - C_{il} - IN_1/IN_2 - TN_{il}$ . First, the fault tree is analyzed with the traditional approach that is currently used in practice to calculate the failure probability of the top node. Next, the computed top node failure probability from the traditional approach is compared with the corresponding value as evaluated from the proposed algorithm in order to illustrate the accuracy of the proposed approach.

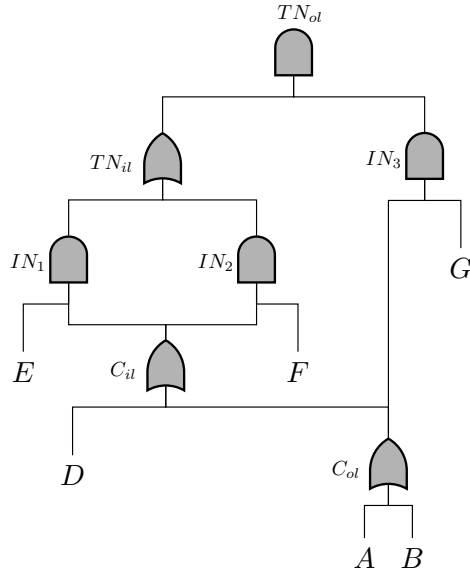


Figure IV.7: An illustrative example of nested loops in a fault tree structure

#### IV.4.1.1 Solution Using Traditional Approach

The fault tree in Figure V.7 consists of one top event, 6 intermediate events and 6 basic events. The probability of failure for all the basic events are shown in the Table IV.2. The failure probabilities of the basic events do not have any real significance and are chosen arbitrarily simply for illustrative purposes.

Table IV.2: Probability of failure and success for each basic events

	A	B	D	E	F	G
$P(1)$	0.005	0.023	0.076	0.0039	0.0164	0.134
$P(0)$	0.995	0.977	0.924	0.9961	0.9836	0.866

The traditional approaches MOCUS [80] and min-max are used to compute the top node probability. A traditional fault tree analysis based on MOCUS [80] involves a determination of minimal cut sets for the top event failure. The MOCUS algorithm for top event failure of the fault tree shown in Figure V.7 leads to five minimal cutsets:  $A - D - F$ ,  $A - B - F$ ,  $A - C - F$ ,  $A - E - F$  and  $B - E - F$ . The probability of the top event is then calculated by min-max approach (Eq. (II.17)) as 0.00019483056951600033.

$$P(TN_{ot}) = 0.00019483056951600033 \quad (\text{IV.9})$$

#### IV.4.1.2 Solution with Proposed Algorithm

The fault tree in Figure V.7 is converted to the Tree data structure as shown below using truth tables to implement the proposed algorithm.

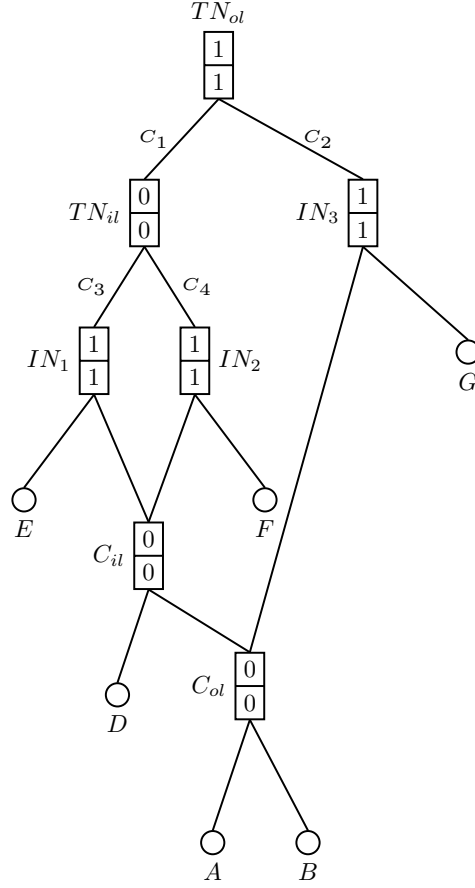


Figure IV.8: Tree data structure for the example of loop in loop

where,  $C_1$  refers to first chain ( $TN_{ol} - TN_{il} - IN_1 / IN_2 - C_{il} - C_{ol}$ ) of the outer loop,  $C_2$  refers to the second chain ( $TN_{ol} - IN_3 - C_{ol}$ ) of the outer loop,  $C_3$  refers to first chain ( $TN_{il} - IN_1 - C_{il}$ ) of the inner loop and  $C_4$  refers to the second chain ( $TN_{il} - IN_2 - C_{il}$ ) of the inner loop.

The inner loop ( $TN_{il} - IN_1 - C_{il} - IN_2$ ) is solved first. For this purpose, all the dependent chains of inner loop ( $C_3$  &  $C_4$ ) are analyzed using the algorithm for chains with independent nodes as explained in Section 2. The top event probability  $P(TN_{il})$  for

each independent chain ( $C_3$  &  $C_4$ ) is calculated in the Eq. (IV.10) below using Eq. (IV.2).

$$\begin{aligned} P(C_3) &= 1 - P(E) \times 1 = 0.9961 \\ P(C_4) &= 1 - P(F) \times 1 = 0.9836 \end{aligned} \tag{IV.10}$$

Second, the  $k_{1_i}$  and  $k_{2_i}$  are calculated for  $i^{th}$  chain in inner loop using Eq. (IV.3).

These values are listed in Table IV.3 below.

Table IV.3: Calculation of  $k_{1_i}$  and  $k_{2_i}$  for  $C_3$  and  $C_4$  of the inner loop

	$C_3$	$C_4$
$TN_i$	0.9961	0.9836
$k_2$	$(-1)^1 P(E) = -0.0039$	$(-1)^1 P(F) = -0.0164$
$k_1 = TN_i - k_2$	1	1
$k_1 + k_2$	0.9961	0.9836

Finally, the coefficients  $K_{1_{il}}$  and  $K_{2_{il}}$  for the inner loop can be calculated using Eq. (IV.5).

$$K_{1_{il}} = \prod_{i=1}^m k_{1_i} = 1, \quad K_{2_{il}} = \prod_{i=1}^m (k_{1_i} + k_{2_i}) - K_{1_{il}} = -0.02023604 \tag{IV.11}$$

With the top node probability  $P(TN_{il})$  as the function of the probability of the common node  $P(C_{il})$ , the imaginary nodes probabilities are calculated in the following

Eq. (IV.12) as per Eq. (IV.7).

$$\begin{aligned}
P_{Inp}(BE_{1_{img}}) &= K_{1_{il}} = 1 \\
P_1(BE_{2_{img}}) &= -\frac{K_2}{K_1} = 0.02023604 \\
Out_{IN_{img}} &= 1 - Inp_{TN_{il}} = 1 \\
Inp_{IN_{img}} &= 1
\end{aligned} \tag{IV.12}$$

where,  $Out_{IN_{img}}$  and  $Inp_{IN_{img}}$  are the output state and input state of the imaginary intermediate node  $IN_{img}$ , respectively.  $Inp_{TN_{il}}$  is the input state of the top node of the inner loop  $TN_{il}$ .

The inner loop is converted to independent chain using the imaginary nodes as shown in the following figure.

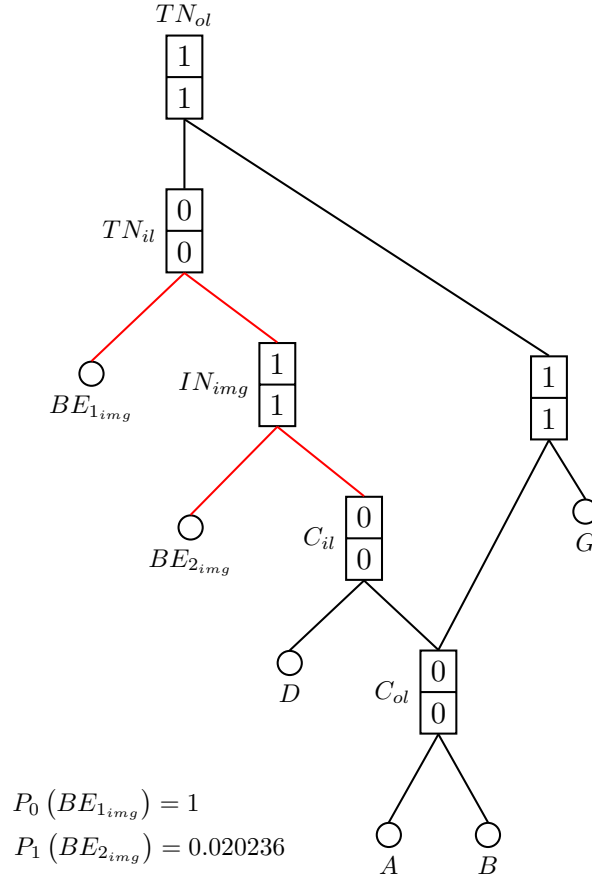


Figure IV.9: Conversion of Inner loop to independent chain in illustrative example

The top node probability for the outer loop is calculated using the approach presented in Section 2.3. Computation of coefficients  $k_{1i}$  and  $k_{2i}$  for each  $i^{th}$  chain of the outer loop is shown in Table IV.4 below.

Table IV.4: Calculation of  $k_{1_i}$  and  $k_{2_i}$  for illustrative example with two chains

	$C_1$	$C_2$
$TN_{ol}$	$1 - P_0(BE_{1_{img}})(1 - P(BE_{2_{img}})P_0(D))$ $= 0.020236$	$P(G)$ $= 0.134$
$k_2$	$(-1)^4 P_0(BE_{1_{img}})P(BE_{2_{img}})P_0(D)$ $= 0.0186981$	$(-1)^0 P(G)$ $= 0.134$
$k_1 = TN - k_2$	$0.0015379$	$0$
$k_1 + k_2$	$0.020236$	$0.134$

Finally, the top node probability  $P(TN_{ol})$  for the output state is calculated using Eq. (IV.13). The top node failure probability calculated using the traditional approach as given in Eq. (IV.6) is same as obtained using the proposed approach shown below.

$$P_{Out=1}(TN_0) = K_1 + K_2 P(C_{ol}) = 0.0019483056951600033 \quad (IV.13)$$

where,

$$K_1 = \prod_{i=1}^m k_{1_i} = 0, \quad K_2 = \prod_{i=1}^m (k_{1_i} + k_{2_i}) - K_1 = 0.002711624$$

$$P(C_{ol}) = 1 - P_0(A)P_0(B) = 0.027885$$

## IV.5 Consideration of Common Cause Failures (CCF)

The problem associated with CCF and their impact has been studied by many researchers [53, 54, 55]. CCFs have been modelled using traditional fault and event trees with an enhancement for consideration of CCF or by using Bayesian networks. The traditional approach is summarized in the following section before introducing the proposed algorithm based on the Tree data structure.

### IV.5.1 Traditional Approach to Model CCF

A basic parameter model (BPM) is used to model the dependent failures due to a common cause [56]. The events impacted by a common cause event form a common cause component group (CCCG). The BPM states that in a CCCG of size  $n_{CCF}$ ,  $m$  failures can occur simultaneously where  $m$  can range from 1 to  $n_{CCF}$ . A component failure can occur either independently or together with other components of CCCG. This leads to a total combinations of failures in a CCCG group as:

$$\binom{n_{CCF}}{1} + \binom{n_{CCF}}{2} + \dots + \binom{n_{CCF}}{n_{CCF}} = 2^{n_{CCF}} - 1 \quad (\text{IV.14})$$

where,  $\binom{n}{k}$  is a mathematical representation of  $\frac{n!}{k!(n-k)!}$ .

In a fault tree analysis, a component failure in CCCG is replaced by an OR gate which is connected to all the possible failures (basic events) due to common cause event linked to component failure of interest and an independent failure. This has been illustrated for a fault tree where two components  $A$  and  $B$  share a common cause failure event as shown in Figure IV.10.

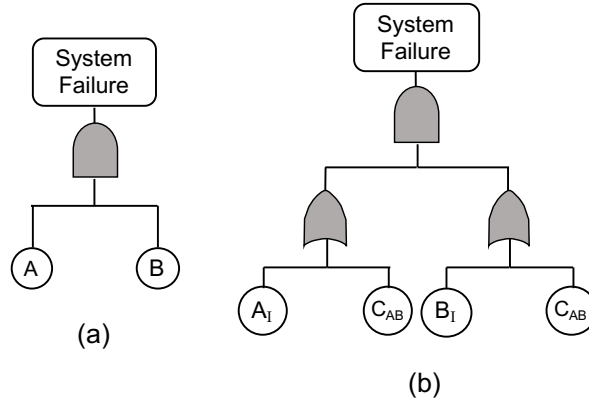


Figure IV.10: Illustration for the inclusion of common cause failures: (a) Fault tree for system failure without CCF, (b) Fault tree for system failure with CCF

where,  $C_{AB}$  is the common cause failure for events  $A$  and  $B$ . In general, the output of each of the newly added OR gate is dependent on all the failures associated with each CCF event. The total number of events connected to the OR gate for an event in CCCG of size  $n_{CCF}$  are given by:

$$\binom{1}{1} \binom{n_{CCF} - 1}{0} + \binom{1}{1} \binom{n_{CCF} - 1}{1} + \dots + \binom{1}{1} \binom{n_{CCF} - 1}{n_{CCF} - 1} = 2^{n_{CCF} - 1} \quad (\text{IV.15})$$

There are various models which can be employed to quantify the failure probability associated with a total of  $2^{n_{CCF}} - 1$  (Eq. (IV.14)) failure combinations. Some of these models are the alpha factor model, the beta factor model, and multi greek letter model [91, 92, 93]. These models require inputs related to component failures and their frequencies which are based on expert judgment and past failures at plants. The parameters in each model for various kinds of common cause failures require continuous updating based on failure data. In United States, US Nuclear

Regulatory Commission (USNRC) collects and disseminates data on such failures and provides an estimation for the parameters needed in the modeling [58, 59].

In this manuscript, alpha factor model is used to quantify the failure probability due to CCF. The alpha factor model for CCF relies on an underlying assumption of symmetry. It states that the probabilities of simultaneous failure of any  $m$  events out of  $n_{CCF}$  event is same, i.e.,  $P(C_{AB}) = P(C_{BC}) = P(C_{CA})$  or  $P(C_{A_i}) = P(C_{B_i}) = P(C_{C_i})$  for CCCG with events  $A$ ,  $B$  and  $C$ . The failure probability involving  $m$  simultaneous component failure in a CCCG of size  $n_{CCF}$  is given by  $Q_m^{n_{CCF}}$ :

$$Q_m^{n_{CCF}} = \frac{1}{\binom{n_{CCF}-1}{m-1}} \alpha_m Q_t \quad (\text{IV.16})$$

where,  $Q_t$  is the total probability of each component failing due to all independent and common cause events,  $\alpha_m$  is the probability of the failure of  $m$  components out of  $n_{CCF}$ .

Once all the events associated with CCF and their respective probabilities are identified using Eq. (IV.16) and Eq. (IV.15), the fault tree can be analyzed using either traditional fault and event tree approaches or network based analysis such as Bayesian networks [94].

#### IV.5.2 Proposed Approach for Analysis of CCF

The new approach summarized in Section 2 and enhanced in Section 3 for nested loops cannot address the common cause failure scenarios with size of CCCG greater than 2. For CCCG greater than 2, more than one common nodes exist in the tree data structure.

A tree data structure for CCCG of size 3 is shown in Figure IV.11. In this figure, the common cause failure events (A, B and C) are replaced with an OR gate and the property of its children are determined using traditional common cause failure analysis methods. As seen in Figure IV.11, more than one leaf node is common among the three dependent chains: Chain 1 ( $TN - IN_1 - CCF_A$ ), Chain 2 ( $TN - IN_2 - CCF_B$ ) and Chain 3 ( $TN - IN_3 - CCF_C$ ).

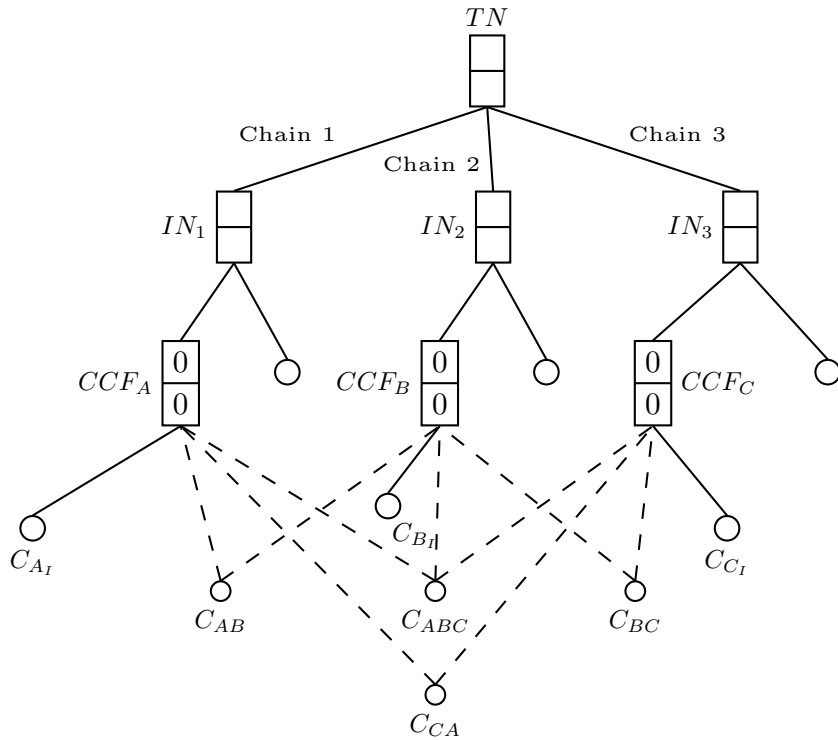


Figure IV.11: A generic case of common cause failures in a Tree data structure

The probability of the top node in each chain can be calculated using Eq. (IV.17).

$$P(TN_{c_i}) = k_{1_i} + k_{2_i} P_1(CCF_i) \quad (IV.17)$$

where,  $i$  represents the  $i^{th}$  common cause failures OR node and  $i$  varies from  $1^{st}$  to  $n_{CCF}^{th}$  CCF event.

The end node,  $CCF_i$  in each chain does not represent one dependent node, instead it depends on  $2^{n_{CCF}-1}$  dependent nodes. For the Tree data structure shown in Figure IV.11, the probability of the top node in Chain 1 can be written in the following form:

$$\begin{aligned}
P(TN_{c_1}) &= k_{1_1} + k_{2_1}P_1(CCF_A) \\
P(CCF_A) &= 1 - P_1(C_{AC}) - P_1(C_{AB}) - P_1(C_{ABC}) + P_1(C_{AC}C_{AB}) + P_1(C_{AC}C_{BC}) \\
&\quad + P_1(C_{BC}C_{AB}) - P_1(C_{BC}C_{AB}C_{AC})
\end{aligned} \tag{IV.18}$$

where, the leaf nodes ( $C_{AC}, C_{AB}, C_{ABC}$ ) represent failure of CCF event  $A$  and the leaf nodes are independent from one another according to the alpha factor method. Hence the intersection in Eq. (IV.18) can be replaced by multiplication of probability of each term. However, the numerical values of intersection cannot be replaced because these nodes are common with leaf nodes of other chains in the Tree data structure as shown in Figure IV.11.

As the estimation of  $P_1(CCF_i)$  in Eq. (IV.17) involves storage and computation of many terms,  $P(TN_{c_i})$  is expressed in terms of the success state of the end node of the chain. Since the end node of each chain in case of the CCF scenario is an OR gate,  $P(TN_{c_i})$  in Eq. (IV.17) is re-written below as given by Eq. (IV.26). The probability of the common node  $P_0(CCF_i)$  in Eq. (IV.26) can be simply replaced by multiplication of

success probability of  $2^{n_{CCF}-1}$  leaf nodes connected to  $CCF_i$ .

$$TN_{c_i} = k'_{1_i} + k'_{2_i}P_0(CCF_i) \quad (\text{IV.19})$$

where,  $k'_{1_i}$  and  $k'_{2_i}$  are the modified coefficients of  $k_{1_i}$  and  $k_{2_i}$ , respectively. The coefficients are modified to accommodate the success probability of end node instead of its failure probability.  $k'_{1_i}$  and  $k'_{2_i}$  are computed using Eq. (IV.3) with one modification in the calculation of  $z$ . The modified  $z$  is referred to as  $z'$  hereafter.  $z'$  is calculated as the total number of state changes between output state of a child node and input state of its parent node along the chain.

Here,  $z'$  is calculated as the function of total number of change is states along the corresponding chain starting from the internal node at height = 1 instead of common node at height = 0 (for  $z$ ). This change in  $z'$  is implemented since last node in the existing algorithm is always considered to be failure. However, in the current scenario the success of last node must be considered. Therefore, the last addition of change is state must be avoided for CCF scenario.

For the Tree data structure shown in Figure IV.11, we can observe that the leaf node  $C_{A_I}$  of Chain 1 is not a common node among other the dependent chains. Therefore, the contribution of the leaf node can be merged in the expression of  $k'_{1_1}$  and  $k'_{2_1}$  for Chain 1. Similarly, the contribution of leaf nodes  $C_{B_I}$  and  $C_{C_I}$  of Chain 2 and Chain 3 can be merged in the expression of  $k'_{1_i}$  and  $k'_{2_i}$  for each chain. The Eq. (IV.18) can be simplified for Chain 1 as follows:

$$TN_{c_1} = k'_{1_1} + k'_{2_1}P_0(CCF_A) = k'_{1_1} + k'_{2_1}P_0(C_{AC})P_0(C_{AB})P_0(C_{ABC}) \quad (\text{IV.20})$$

Since, probability of the top node  $P(TN)$  is not a linear function of one common node in CCF scenario,  $P(TN)$  is given by the intersection of the probability of each dependent chain.

$$P(TN) = \bigcap_{i=1}^n P(TN_{c_i}) \quad (\text{IV.21})$$

where,  $n$  is the total number of chains. The probability of the top node for the Tree data structure shown in Figure IV.11 is given by Eq. (IV.29).

$$\begin{aligned} P(TN) &= P(TN_{c_1}) \cap P(TN_{c_2}) \cap P(TN_{c_3}) \\ &= \left( k'_{1_1} + k'_{2_1} P_0(C_{AC}) P_0(C_{AB}) P_0(C_{ABC}) \right) \cap \left( k'_{1_2} + k'_{2_2} P_0(C_{BC}) P_0(C_{AB}) P_0(C_{ABC}) \right) \cap \\ &\quad \left( k'_{1_3} + k'_{2_3} P_0(C_{AC}) P_0(C_{BC}) P_0(C_{ABC}) \right) \end{aligned} \quad (\text{IV.22})$$

The calculation of intersection of all the terms in Eq. (IV.29) is computationally expensive compared to direct multiplication of probability of nodes. However, the proposed approach still provides computational advantages over other traditional approach because such intersection is only performed at the top node probability computation.

### IV.5.3 Application Example

The application of the proposed algorithm for CCF is illustrated through a simpler version of CCF example used in Rasmuson & Kelly [95] as shown in Figure IV.13. In this example, the failure criteria is considered as three components must fail to function. The example consists of two CCG, one relates to emergency diesel generator (EDG) fails to run and another to EDG fails to start. The referred paper [95] shows that the CCF example leads

to a total of 28 minimal cut sets and the top event failure probability is found out to be  $1.24E - 04$ .

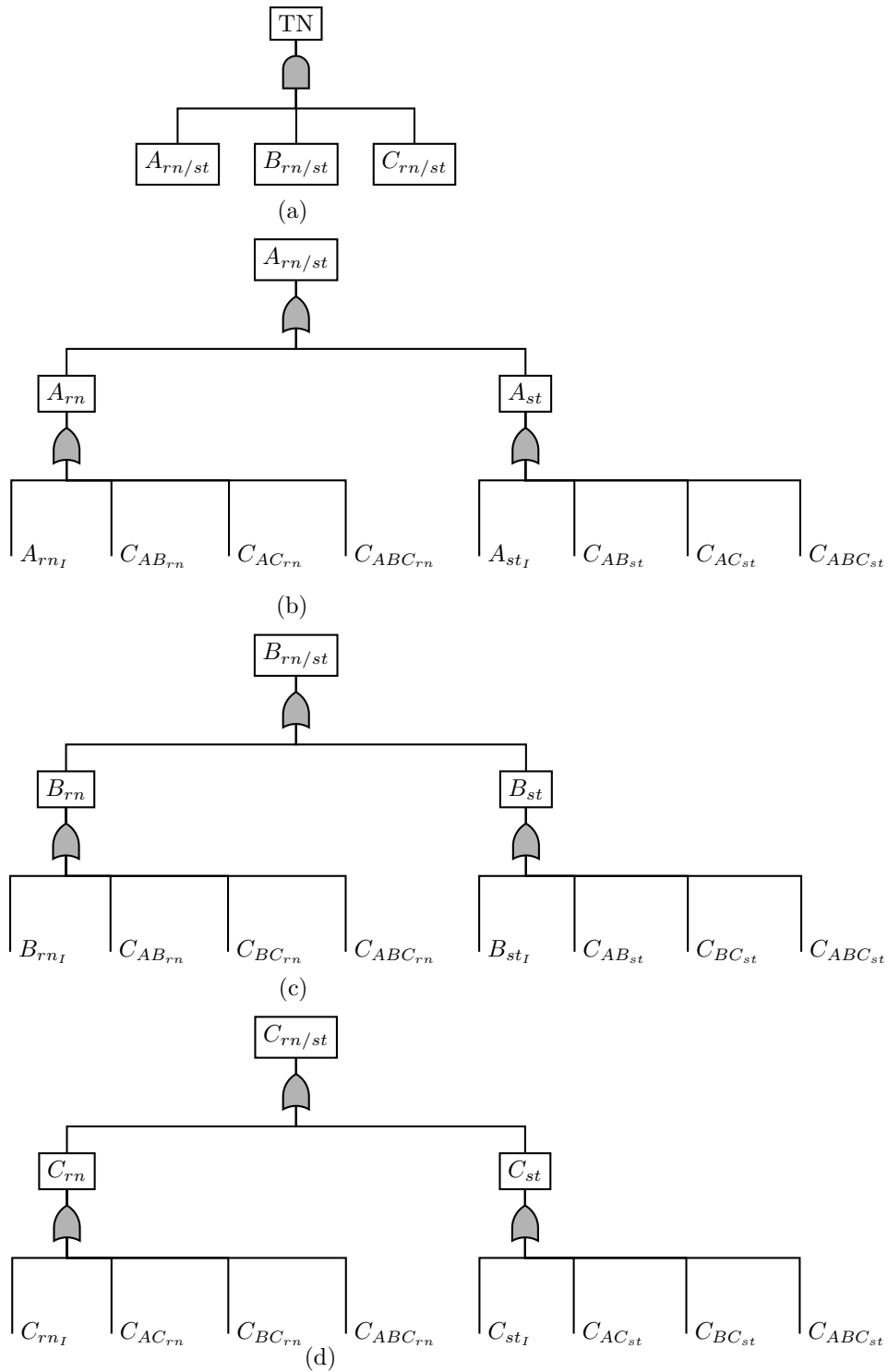


Figure IV.12: A simple fault tree with many common cause failures referred from [95]: (a) Fault tree with three transfer gates, (b) Fault tree connected at transfer gate A, (c) Fault tree connected at transfer gate B and (d) Fault tree connected at transfer gate C

where, nodes  $A$ ,  $B$  and  $C$  represent the failure of EDG-A, EDG-B and EDG-C.  $TN$  denotes failure of  $A$ ,  $B$  and  $C$ . Subscripts  $rn$ ,  $st$ , and  $rn/st$  denotes failure to run, failure to start, and failure to run or start, respectively. Nodes  $A_{rn_I}$ ,  $A_{st_I}$ ,  $B_{rn_I}$ ,  $B_{st_I}$ ,  $C_{rn_I}$  and  $C_{st_I}$  represent the independent failure of each event. Nodes  $C_{AB_{rn}}$ ,  $C_{AB_{st}}$ ,  $C_{AC_{rn}}$ ,  $C_{AC_{st}}$ ,  $C_{BC_{rn}}$  and  $C_{BC_{st}}$  represent the common cause failure associated with failure of two events. Nodes  $C_{ABC_{rn}}$  and  $C_{ABC_{st}}$  represent the common cause failure associated with failure of 3 events.

The probability of each individual leaf node associated with CCF are calculated using alpha parameter method in Bensi et al. [30] and are listed in the following table.

Table IV.5: Probability for CCF related basic event failures

BE	$P_f$	BE	$P_f$
$A_{rn_I}$	1.16E-02	$A_{st_I}$	4.91E-03
$B_{rn_I}$	1.16E-02	$B_{st_I}$	4.91E-03
$C_{rn_I}$	1.16E-02	$C_{st_I}$	4.91E-03
$C_{AB_{rn}}$	1.33E-04	$C_{AB_{st}}$	3.35E-05
$C_{AC_{rn}}$	1.33E-04	$C_{AC_{st}}$	3.35E-05
$C_{BC_{rn}}$	1.33E-04	$C_{BC_{st}}$	3.35E-05
$C_{ABC_{rn}}$	8.84E-05	$C_{ABC_{st}}$	2.33E-05

The Tree data structure consists of three main chains: Chain 1 ( $TN - A_{rn/st}$ ), Chain 2 ( $TN - B_{rn/st}$ ), and Chain 3 ( $TN - C_{rn/st}$ ). Each of these chains is splitted in two ends at the last node as: Chain 1 ( $A_{rn/st} - A_{rn}$  and  $A_{rn/st} - A_{st}$ ), Chain 2 ( $B_{rn/st} - B_{rn}$  and  $B_{rn/st} - B_{st}$ ), Chain 3 ( $C_{rn/st} - C_{rn}$  and  $C_{rn/st} - C_{st}$ ). The two splitted ends of each

each chain are connected through OR gates with the end of each chain and they represent two scenarios: (i) EDG fails to run, and (ii) EDG fails to start. The contribution from the two splitted ends of each chain is simply reflected in the expression of  $k'_1$  and  $k'_2$  for each of the chain.  $k'_1$  and  $k'_2$  are computed for each of the three chains according to Section IV.5.2 and are tabulated in the table below.

Table IV.6: Calculation of  $k'_{1_i}$  and  $k'_{2_i}$  for CCF example with three chains

	Chain 1	Chain 2	Chain 3
$P_{Ind}(TN_{c_i})$	$1 - P_0(A_{s_I})P_0(A_{r_I})$ =0.9999430	$1 - P_0(B_{s_I})P_0(B_{r_I})$ =0.9999430	$1 - P_0(C_{s_I})P_0(C_{r_I})$ =0.9999430
$k'_2$	$(-1)^1 P_0(A_{s_I})P_0(A_{r_I})$ =-5.695E-05	$(-1)^1 P_0(B_{s_I})P_0(B_{r_I})$ =-5.695E-05	$(-1)^1 P_0(C_{s_I})P_0(C_{r_I})$ =-5.695E-05
$k'_1 = P_{Ind}(TN_{c_i}) - k'_2$	1	1	1

After calculating  $k'_1$  and  $k'_2$ , the probability of the top node is calculated using Eq. (IV.29) and shown below:

$$\begin{aligned}
P(TN) &= P(TN_{c_1}) \cap P(TN_{c_2}) \cap P(TN_{c_3}) = 1.24E - 4 \\
P(TN_{c_1}) &= k'_{1_1} + k'_{2_1} P_0(C_{AC_s})P_0(C_{AB_s})P_0(C_{ABC_s})P_0(C_{AC_r})P_0(C_{AB_r})P_0(C_{ABC_r}) \\
P(TN_{c_2}) &= k'_{1_2} + k'_{2_2} P_0(C_{AB_s})P_0(C_{BC_s})P_0(C_{ABC_s})P_0(C_{AB_r})P_0(C_{BC_r})P_0(C_{ABC_r}) \\
P(TN_{c_3}) &= k'_{1_3} + k'_{2_3} P_0(C_{AC_s})P_0(C_{BC_s})P_0(C_{ABC_s})P_0(C_{AC_r})P_0(C_{BC_r})P_0(C_{ABC_r})
\end{aligned} \tag{IV.23}$$

The probability of the top node in Eq. (IV.23) is same as the value calculated in Rasmuson & Kelly [95]). The traditional algorithm implemented in Rasmuson & Kelly

[95] required a calculation of total 28 minimal cut sets for the same problem, while the proposed approach required computation of 8 expressions which makes the proposed algorithm more effective comparatively.

#### IV.6 $n$ of the $m$ Input ( $n/m$ ) Logic Gates

$n/m$  logic gate indicates that exactly  $n$  out of  $m$  events occur. Typically, a  $n/m$  gate in a fault tree is analyzed by converting the  $n/m$  gate to a set of AND and OR gates. The  $n/m$  gate is converted to a fault tree with top event as OR gate and the intermediate events as AND gates [52]. The total number of children for each AND gate are  $\binom{m}{n}$ . For example, a  $2/3$  gate with  $x$ ,  $y$  and  $z$  as input, 2 of the 3 input events must occur. The output probability for the logic gate in such a case is given by:  $(x \text{ AND } y) \text{ OR } (x \text{ AND } z) \text{ OR } (y \text{ AND } z)$ . An illustration of the same is shown in the following figure.

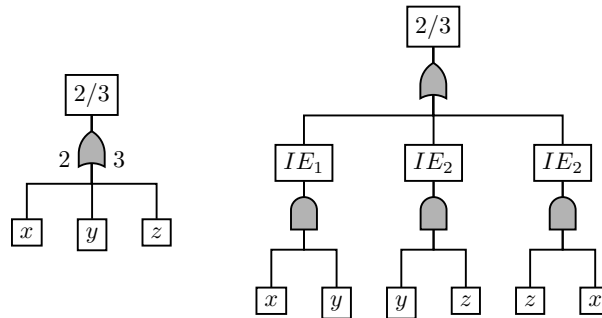


Figure IV.13: Conversion of  $n/m$  logic gate to a set of AND and OR gates.

In this manuscript, we propose to analyze the  $n/m$  gate using the algorithm presented for CCF events. This is because the end nodes of the chains in complex loop created for  $n/m$  gates, are also dependent on more than one dependent node. Therefore, the solution for  $n/m$  gate also follows the same steps involved for CCF analysis. However the method for the computation of  $k_{1_i}$  and  $k_{2_i}$  for each chain is updated based on the

structure of fault tree for  $n/m$  gate. The solution for  $n/m$  gates can be generalized as follows:

$$\begin{aligned}
 P(TN) &= \bigcap_{i=1}^m P(TN_{c_i}) \\
 P(TN_{c_i}) &= k_{1_i} + k_{2_i} \prod_{j=1}^n P_1(X_j) \\
 &= 1 - \prod_{j=1}^n P_1(X_j)
 \end{aligned} \tag{IV.24}$$

where,  $X_1$  to  $X_n$  are the leaf nodes connected to the end node of chain  $i$ . Here,  $k_{1_i}$  and  $k_{2_i}$  for each chain is calculated using Section IV.2.3. Their values for each chain  $i$  are found as,  $k_{1_i} = 1$  and  $k_{2_i} = -1$ . Hence, the Eq. (IV.24) can be written as,

$$P(TN_{c_i}) = 1 - \prod_{j=1}^n P_1(X_j) \tag{IV.25}$$

The algorithm presented here for  $n/m$  gates does not offer computational advantages over the traditional algorithms due to the complexity of the type of the gate. This algorithm is presented here for the utilization of the algorithm for this type of gate and scenarios.

## IV.7 Computational Efficiency Compared with Traditional Fault Tree Analysis Approaches

In this section, the computational efficiency of the proposed algorithm is compared to that of a traditional approach.

### IV.7.1 Efficiency for Nested Loop Algorithm

Five different fault trees with multiple nested loops are considered. The geometric configuration of each of the fault tree such as width, height, total number of events, number of basic events are provided in Table IV.7.

Table IV.7: Details of the five cases of fault tree structure

Case No	No of BE	No of IEs	Width	Height
Case 1	20	12	20	6
Case 2	65	9	65	3
Case 3	116	10	116	3
Case 4	165	8	165	3
Case 5	215	9	215	3

For a given geometric configuration, each case will result in a different computation time. The fault trees considered in this section are first analyzed with a traditional approach that utilizes MOCUS algorithm and upper bound approximation for the quantification of top event probability. The true computation run time or CPU times are highly dependent upon the type of hardware used. Therefore, for a hardware independent comparison, run times are normalized with respect to the run time for the proposed method. This assists with visualization of relative difference in the computation times using the two approaches. The normalized run times for each case of case is compared in Figure IV.14. As seen in the figure, the run time of traditional approach is significantly high in comparison to the computational time incurred by proposed algorithm.

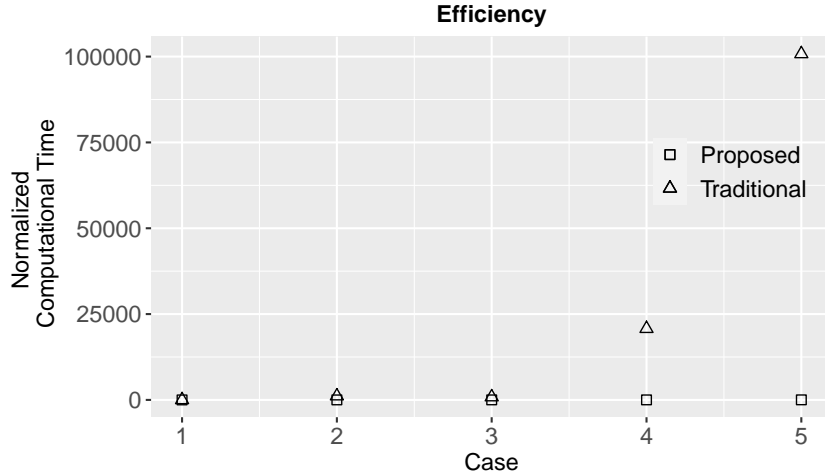


Figure IV.14: Computational efficiency of the proposed algorithm compared to traditional PRA approaches for nested loops

#### IV.7.2 Efficiency for CCF Algorithm

Next, we consider five cases for common cause failure analysis where the number of events associated with common cause failures are increased from 2 to 6. Each fault tree in the consideration consists of 12 intermediate events, 18 basic events and height as 4.

Figure IV.15 compares the normalized computational run time between traditional approaches employing upper bound approximation and the proposed algorithm. As seen in Figure IV.15, the computational time for the traditional approach increases exponentially with increase in number of CCF events whereas computational time for the proposed algorithm is comparatively very less. This comparison shows a significant improvement in the computational efficiency that can be achieved by the proposed algorithm in the solution of CCFs.

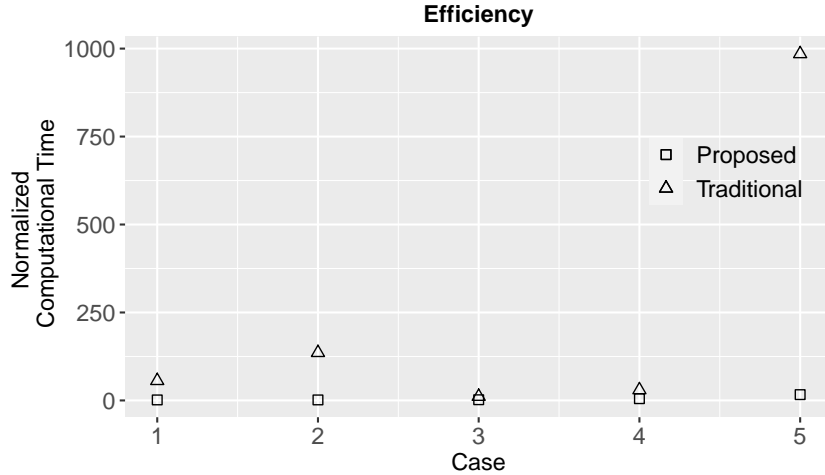


Figure IV.15: Computational efficiency of the proposed algorithm compared to traditional PRA approaches for CCF

#### IV.8 Comparison of Proposed Algorithm with other Bayesian Network Algorithms

In recent years, Bayesian networks have been widely used as an alternative tool instead of fault tree, event tree and binary diagrams. Bayesian networks have shown significant potential in PRA applications by accounting for statistical correlation between various events [94]. However, even with advances in scientific computing, Bayesian networks usually suffer from computational limitations [26] which are either due to the storage space required for creating conditional probability tables [27, 28] or due to the size of clique potentials in the junction-tree algorithm [29], etc. Recently, Bensi et al. [30] developed a compression algorithm which reduces the size of Bayesian network optimally such that the clique size for the compressed network is comparatively very less. The clique size is a measure of computational efforts, i.e. the computational efforts are linear function of clique size.

In this manuscript, the proposed algorithm is compared with the compression algorithm of Bensi et al. [30] to highlight the advantages of the proposed algorithm in terms of computational efficiency. The network shown in Figure IV.16 is taken from Bensi et al. [30] which models the flow of liquid from source to sink.

The presented compression algorithm starts with the original network shown in figure below which has the clique size of 224. They presented a solution which is based on number of minimal link sets and cut sets. Their presented solution reduces the clique size from 224 to 64. It also assumes that minimal cutsets are calculated before the application of the compression algorithm. Hence the computational efforts associated with the clique size of 64 does not take into account the NP-hard problem associated with the calculation of minimal link set or cut sets. The clique size estimation is provided for an equivalent binary network, where the original flow network variables were converted to interval nodes. For each set of interval variation, the binary network is solved using the compression algorithm.

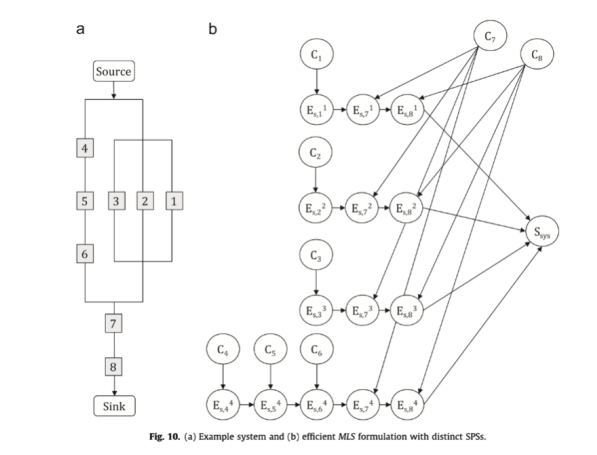


Figure IV.16: Example from Bensi et al. [30]

We have converted the flow network problem in Figure IV.16 to the binary network because the proposed algorithm is valid for tree data structure based on binary networks. The type of logic gates for this binary network are selected randomly because the computational time required by proposed approach does not depend on the type of logic gate. The corresponding tree data structure is shown in Figure IV.17. The inputs from  $BE_1 - BE_8$  are considered to be leaf nodes same as they were considered components in Bensi et al. [30]. This problem presents the nested loop scenario in the tree data structure. It is shown later in this paper that the solution for this tree data structure involves a total of 15 computational steps using proposed approach. For this example, the proposed approach is approximately 400% more efficient compared to the algorithm presented in Bensi et al. [30]. The steps involved in the solution of the tree data structure are described below,

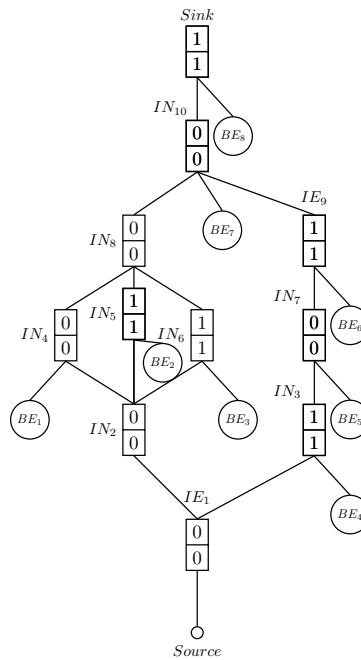


Figure IV.17: Tree data structure for example in Bensi et al. [30]

As previously explained, the inner loop is solved first in the nested loop structure. The  $K_{1_{ii}}$  and  $K_{2_{ii}}$  are evaluated in terms of common node  $IN_2$  for the inner loop ( $IN_8 - IN_4/IN_5/IN_6 - IN_2$ ). Then the inner loop is converted to an equivalent independent chain such that then the outer loop ( $IN_{10} - IN_9 - IN_7 - IN_3 - IN_1 - IN_2 - IN_4/IN_5/IN_6 - IN_8$ ) can be analyzed in terms of common node  $IE_1$  of the outer loop.

1. The Chain 1 of the inner loop ( $IN_8 - IN_4 - IN_2$ ) is evaluated as below.  $IN_{8_{c1}}$ ,  $k_{1_1}$  and  $k_{1_1}$  are calculated as numerical values in Eq. (IV.26). The complexity for this operation is given as a function of- Nos. of inner nodes in Chain 1 = 1),

$$\begin{aligned} IN_{8_{c1}} &= 0, k_{2_1} = (-1)^1 P_0(BE_1) \\ k_{1_1} &= IN_{8_{c1}} - k_{2_1} = P_0(BE_1) \end{aligned} \tag{IV.26}$$

2. The  $k_{1_2}$  and  $k_{2_2}$  variables are calculated for Chain 2 ( $IN_8 - IN_5 - IN_2$ ) of inner loop while assuming  $P_1(IN_3) = 1$ . Complexity =f(Nos. of inner nodes in Chain 2 = 1)

$$\begin{aligned} IN_{8_{c2}} &= 0, k_{2_2} = (-1)^1 P_0(BE_2) \\ k_{1_2} &= IN_{8_{c2}} - k_{2_2} = P_0(BE_2) \end{aligned} \tag{IV.27}$$

3. The  $k_{1_3}$  and  $k_{2_3}$  variables are calculated for Chain 3 while assuming  $P_1(IN_4) = 1$ . Complexity =(Nos of inner nodes in Chain 3 = 1)

$$\begin{aligned} IN_{8_{c3}} &= 0, k_{2_3} = (-1)^1 P_0(BE_3) \\ k_{1_3} &= IN_{8_{c3}} - k_{2_3} = P_0(BE_3) \end{aligned} \tag{IV.28}$$

4. The  $K_{1_{ii}}$  and  $K_{2_{ii}}$  are calculated for the inner loop using Eq. (IV.5). Complexity

= (1)

$$K_{1_{il}} = \prod_{i=1}^3 k_{1_i} = 0$$

$$K_{2_{il}} = \prod_{i=1}^3 (k_{1_i} + k_{2_i}) - K_{1_{il}}$$
(IV.29)

5. The equivalent independent chain properties are evaluated for the inner loop using the algorithm presented in Section IV.4 to replace the inner loop. This conversion has been shown in the Figure IV.18 below. Complexity =(1)
6. Replacing the Nodes Complexity =(1)

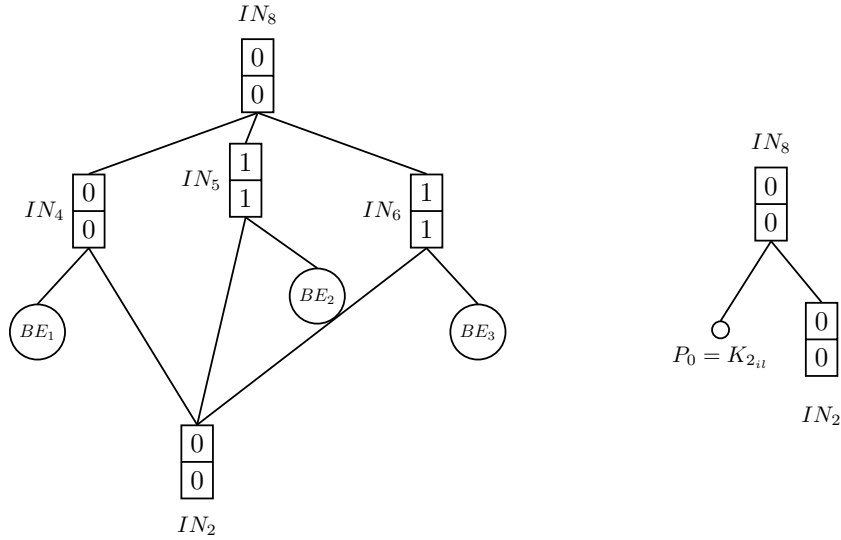


Figure IV.18: Equivalent tree with independent nodes for Inner loop

7. Lastly, the properties of Chain 1 ( $IN_{10} - IN_8 - IN_2 - IN_1$ ) of the outer loop are evaluated. Complexity =(Nos. of inner nodes in chain 1 = 2)
8. Similarly, properties of the Chain 2 ( $IN_{10} - IN_9 - IN_7 - IN_3 - IN_1$ ) of the outer loop are evaluated. Complexity =(Nos. of inner nodes in Chain 2 = 3)
9. The numeric values for the terms  $K_{1_{ol}}$  and  $K_{2_{ol}}$  are evaluated based on Eq. (IV.5). The probability for  $IN_{10}$  is determined in terms of  $K_{1_{ol}}$  and  $K_{2_{ol}}$ . Complexity =(1)

10. The final probability of  $IN_{10}$  at output state is determined by replacing the probability of failure for common node in the loop,  $IN_1$ . Complexity =(1)
11. Top node probability is determined using the existing approach for independent nodes. Complexity =(1)

The following tables shows the comparison of the number of steps involved in both the approaches. It can be seen that the total computational steps utilized by proposed enhancement is about 4 times lesser than that of the compression algorithm.

Table IV.8: Summary of computational steps

	Bensi et al. [30]	Proposed
No of Steps	$f(64)$	14

## IV.9 Conclusions

Fault and event trees are used for probabilistic risk assessment of nuclear power plant systems. A fault and even trees analysis for a power plant requires modelling of hundreds of component failures, logic gates, multiple occurring events, and dependent events. Such interconnection for large networks can lead to excessive computational demand. Most of the traditional methods address computational demand by using assumptions or rely on high performance computing facilities that allow implementations of parallel computing. In chapter III, a novel module based approach to address the computational demands of a PRA. This manuscript presents enhancement to this existing approach to provide solution for the following commonly observed scenario in PRA that were not addressed by the algorithm presented in chapter III.

- Nested loop structure particularly concerned with multi hazard, multi unit PRA
- Common cause failure scenario
- Application of  $n/m$  gates in a typical PRA

The basic of the algorithm remains on the same ground as chapter III where the logic gates of fault or event trees are converted to corresponding compressed truth tables to achieve the desired efficiency. The algorithm for nested loop provides an approach for converting the inner loops of the tree data structure into independent chains. The analysis of common cause failure scenario in fault or event tree network relies on the intersection of multiple chains' probabilities at the top node of the loops. It has been shown with illustrative example that the proposed approach provides exact results when compared to other traditional methods. In addition, two types of network configurations are used to illustrate the computational efficiency of proposed algorithm. First, networks with nested loops with an increase in number of basic events are considered, Second, networks with common cause failure events with increase in size of CCCG are considered. It is seen that the computational demand of proposed algorithm can be less than that of traditional approach by an order of magnitude for some cases in both the scenarios. The proposed algorithm is a significant improvement of the currently available techniques that deals with the complex interaction among various components or events of the fault and event tree networks.

---

---

## PART V

---

**Understanding Uncertainty Scaling for Flooding**

**PRA based on Experimental Data**

## V.1 Introduction

Nuclear industry has undertaken several studies to ensure safety against flooding in power plants after the tsunami induced flooding initiated the accident at Fukushima Daiichi power station in 2011 [35, 36]. A large number of such studies rely on advanced flooding simulation tools [37]. The simulation of catastrophic flooding events often suffers from a lack of confidence in the predictions due to inherent randomness, lack of knowledge about the physics of the complex interactions and the associated uncertainties. Some recent studies have focused on the development of a consistent methodology for verification and validation of these advanced simulation tools [38, 39, 40]. Yet, such validation studies are quite challenging due to a lack of real-world data. In almost all cases, simulation tools are validated by comparison with data from laboratory experiments. Furthermore, the data from laboratory experiments is also used for uncertainty quantification. However, it must be noted that a certain degree of validation based on laboratory experiments does not necessarily mean the same degree of validation at real-world scale. Hence, the premise used for the study presented in this paper is embedded in the belief that the degree of uncertainty in laboratory experiments does not translate into the same degree of uncertainty at the real-world scale. However, it is difficult to provide an explicit proof of this premise due to a lack of usable quantifiable data at real-world scale. Therefore, a comprehensive knowledge of the scaling phenomena is necessary.

An improved understanding of the scaling phenomena would help to reduce the gap between a real application at the plant level and a test facility at smaller scale. An implicit extrapolation of the data is performed when a simulation tool is validated through a smaller scale facility in laboratory for studying a real-world application. This could

lead to scale distortion and greater uncertainty in predictions [43, 44]. Lack of available quantifiable data at full scale, and a lack of understanding about the applicability of data obtained at reduced scale leads to a residual epistemic uncertainty in a simulation model.

Traditional way of addressing scaling phenomenon involves the analysis of complex flooding process through a dimensionless analysis of model at a smaller scale. It involves the analysis of various fluid dynamics phenomenon such as mass transfer among different phases, the interfacial tension, diffusion, dispersion, absorption, etc. The physical simulation of all these quantities leads to a large number of dimensionless parameters. These dimensionless parameters must be equal for a real-world prototype and a small-scale model. However, many a times this condition cannot be satisfied simultaneously between the prototype and model. Either geometric similarity or the kinematic similarity cannot be attained. Moreover, knowing the scale of a flooding at nuclear power plants beforehand is not practically possible. Presently, this inadequacy is addressed through expert opinion and professional judgment.

A possible solution to address the problem more technically is to categorize the experimental data at different scales ranging from small to medium sized experiments. Such a categorization can help in understanding and illustrating the scaling effect. The experimental data at different scale can be imported from the existing literature for similar kind of relevant experiments. In this paper, the scaling effect is studied and illustrated by such a categorization of available experimental data. Subsequently, uncertainties in the data at different scales are characterized and quantified to allow logical extrapolation for real life applications. Such an improved understanding can reduce the reliance on heuristic approaches. An increased awareness of the scaling phenomena can help guide future experimental studies to focus on collection of data at

appropriate scales and in effect enhance the performance of simulation codes. Such a reduction in scale distortion will enhance our ability to estimate flooding risks more realistically.

An external flooding scenario has been considered in this study wherein the flood water breaks into a facility and leads to inundation inside the facility. Various doors or windows in the facility can leave a passage for water entryway in case of plant flooding. Ventilation openings can also be considered as the key locations of concern that are susceptible to failure during a flood thereby creating an opening for flood waters to enter the building. This type of flooding scenario is studied by the existing experiment for flow over weirs. Usually, the opening is only partially covered with water, hence the flow over a weir experiments illustrate such conditions realistically. The results for flow over a weir experiments also apply to a scenario wherein a part of a floodwall collapses to create an opening for the flood waters to enter the plant landscape. The concept of flow over a weir has been studied extensively by the experts in the field of fluid mechanics. In real world application, the geometry of a rectangular weir depicts the flooding scenario most appropriately. Moreover, there is a significant volume of literature and experimental studies available from the research on rectangular weir and hence the selection of this particular case study in this paper.

## **V.2 Description of Application Case Study**

There are many experiments that have been conducted on rectangular weirs with different types of weirs such as broad crested weir, long crested weir, sharp crested weir and narrow crested weir out of which sharp crested weir is the most relevant for the case study. That is because it is made of thin plate which represents a partial opening, a protective dike,

a door or a window in the most appropriate manner. Figure V.1 shows a typical sharp crested rectangular weir with all the geometric parameters that can vary for different weirs. These parameters are: length of the channel ( $L$ ), weir total width ( $B$ ), weir opening width ( $b$ ), height of channel over the weir crest ( $H$ ), weir height ( $P$ ) and corner geometry. The weir opening width is referred to as weir width in the rest of the paper.

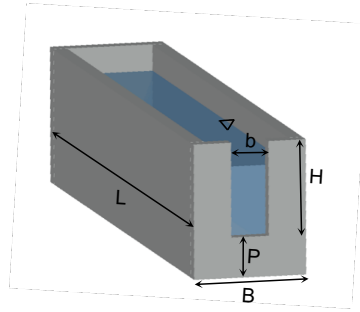


Figure V.1: Geometry of rectangular weir

### Figure 1: Geometry of rectangular weir

Flow rate or discharge (volumetric flow rate per second) over the weir is a crucial parameter for detecting the impact of the incoming flood. Various researchers have developed weir discharge formula based on water head, weir width, weir height, mass density, surface tension of the fluid, viscosity, etc. However, we have used a simplified weir discharge formula in this study that is most widely used for discharge calculation. Eq. (V.1) shows rectangular weir discharge formula based on the fluid mechanics principles which is most widely used for discharge calculation. The actual discharge ( $Q_{act}$ ) in Eq. (V.1) is depended on the water head ( $h$ ), weir width ( $b$ ) and coefficient of discharge ( $C_d$ ). Water head,  $h$  is measured above the weir height at a distance about  $3h-4h$  from the weir outlet at upstream [96] .

$$Q_{th} = \frac{2}{3}gbh^{1.5} \tag{V.1}$$

$$Q_{act} = C_d Q_{th}$$

where,  $Q_{th}$  is the theoretical discharge and  $Q_{act}$  is the actual discharge. The formula in Eq. (V.1) only rely on  $h$  and  $b$  whereas the effect of other weir parameters such as  $L$ , corners, weir height, etc. is reflected in the term  $C_d$ .

Theoretical discharge equation of a rectangular weir given in Eq. (V.1) considers the effect of only water head  $h$  and opening width  $b$  for the evaluation of discharge. However, it ignores the contribution of various other weir parameters such as  $P$ ,  $B$ ,  $H$ ,  $L$ , ratio  $b/B$ , ratio  $h/H$ , ratio  $h/P$ , shape of edges and corner geometry. The effect of these other weir parameters is incorporated in the discharge equation through coefficient of discharge term. Considerably, it is also impractical to consider large set of parameters since most of the data is not recorded in the existing literature and is unavailable at a real scale. For example, appropriate weir length for the modelling of flood approaching the protection dikes at the power plant cannot be defined. Uncertainty associated with these other parameters can be conveniently reflected into the coefficient of discharge. Henceforth, our focus in this paper remains on the three main parameters:  $h$ ,  $b$  and  $C_d$ . As the scale for the experiment changes, some parameters will embark significant effects on the overall discharge and others will have minor effects. Instead of quantifying the effect of each parameter separately,  $C_d$  allows the quantification of the overall effect of the scale change. Therefore, the effect of scaling is identified through the  $C_d$ . Note that such simplification introduces inadequacy in the theoretical model due to lack of knowledge of the underlying physics of the weir discharge. A total of seven experiments are considered from various studies in literature to understand and illustrate the impact

of uncertainties on discharge equation of rectangular sharp crested weir [97, 98]. Many of the weir parameters vary in these experiments such as weir width, weir height, and weir length, corner geometry, etc. which gives sufficient variability for the assessment of uncertainties. Weir discharge data is plotted as  $C_d$  vs  $h$  for experiments 1 to 7 in Figure V.2 to Figure V.8, respectively. These variabilities for each experiment are listed below.

- Experiment 1 to 3: These were conducted for only one geometry of rectangular weir (Figure V.2 to Figure V.4).
- Experiment 4: Three different weir heights ( $P$ ) as 0.10, 0.15, and 0.20 m were considered (Figure V.5).
- Experiment 5: Four kinds of round-crested weirs with different roundness ratio were considered where roundness ratio is defined as the ratio of radius of round upstream ( $u/s$ ) corner and round downstream ( $d/s$ ) corner (Figure V.6).
- Experiment 6: Three different weir widths: 0.2 m, 0.3 m and 0.4 m were considered (Figure V.7).
- Experiment 7: A series of experiments were performed where the opening weir width,  $b$  was changed from 0.02 m to 0.32 m at an interval of 0.02 m (Figure V.8).

While all the experiments were performed with a similar sharp crested weir set ups, their observations, discharge plots and experimental results show significant variation because they were performed at various laboratories located worldwide and by different researchers. Such variations in the experiment handling makes an appropriate case for the identification of the combination of epistemic and aleatoric uncertainties. Additionally, the variation of scale among all these experiments provides a unique data set incorporating scaling effect and uncertainties for the assessing the objective of the

current study.

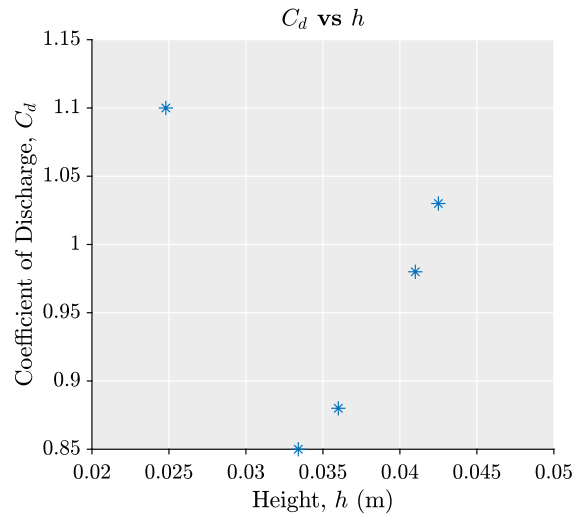


Figure V.2: Plot for  $C_d$  vs  $h$  for Experiment 1

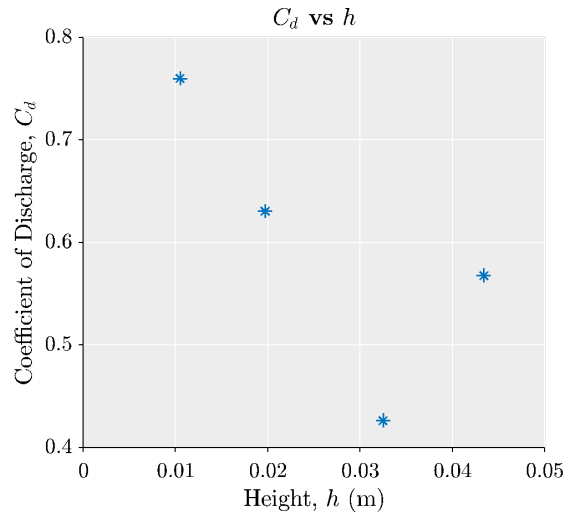


Figure V.3: Plot for  $C_d$  vs  $h$  for Experiment 2

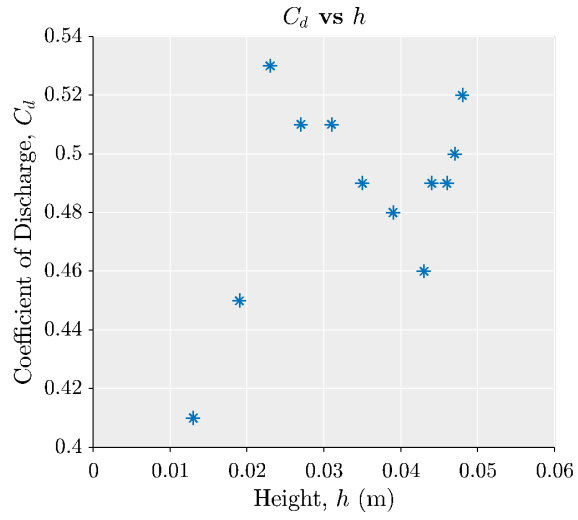


Figure V.4: Plot for  $C_d$  vs  $h$  for Experiment 3

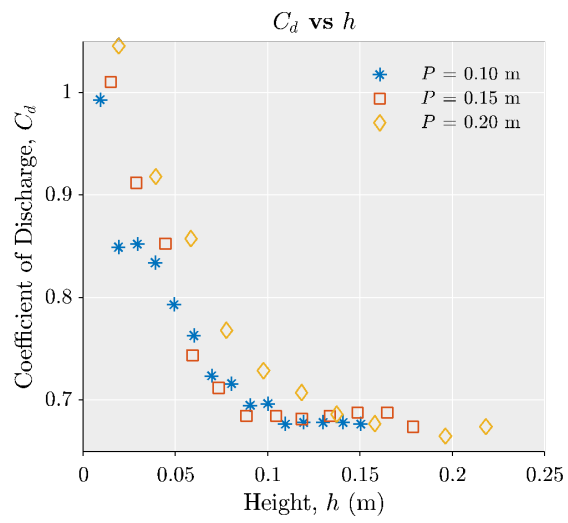


Figure V.5: Plot for  $C_d$  vs  $h$  for Experiment 4

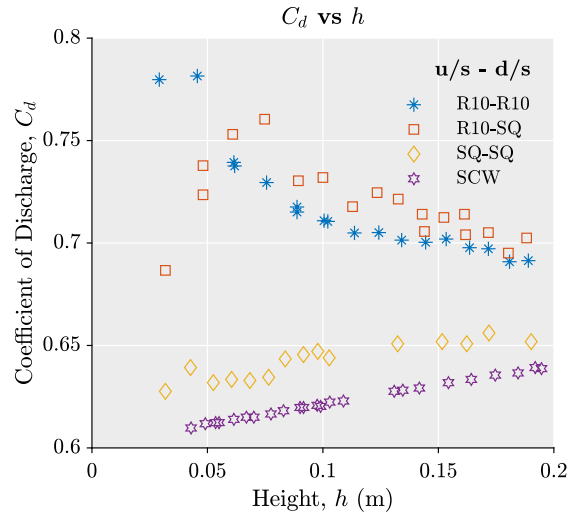


Figure V.6: Plot for  $C_d$  vs  $h$  for Experiment 5

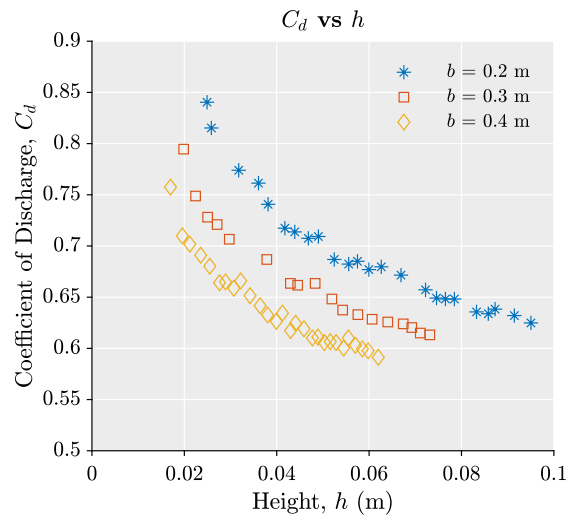


Figure V.7: Plot for  $C_d$  vs  $h$  for Experiment 6

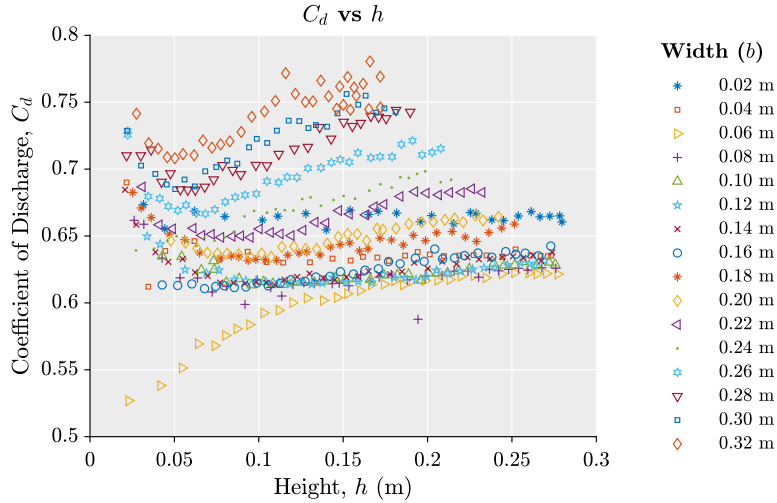


Figure V.8: Plot for  $C_d$  vs  $h$  for Experiment 7

Where,  $u/s$  is upstream radius and  $d/s$  is downstream radius of the weir opening corner, R10 denotes 10 mm radius, SQ denotes square weir and SCW is sharp created weir.

### V.3 Scaling Analysis

The critical weir parameters in Eq. (V.1). are the width of opening and the water height. Hence, these parameters are summarized for all the seven experiments in Table V.1. Only maximum and minimum opening widths are mentioned for Experiment 6 and 7 since there were conducted for a set of varying opening widths. The water height,  $h$  varies in all the experiments, therefore, only the maximum height of water is mentioned for all the experiments. Increase in the scale of height and width with respect to the smallest experiment scale is also listed in the table.

Table V.1: Dimension of weir outlet for all the experiments

Exp. No.	$b$ vs $h_{max}$ (m)	Width scaling compared to 0.01 m	Height scaling compared to 0.01m	Total Data Points
1	0.03 x 0.05	3	5	5
2	0.033 x 0.05	3.3	5	4
3	0.05 x 0.08	5	8	12
4	0.25 x 0.22	25	22	37
5	0.52 x 0.2	52	20	75
6	(0.2 – 0.4) x 0.1	40	10	70
7	(0.02 – 0.32) x 0.28	32	28	508

Scale variation for water height and weir width is about 30-50 times among several experiments. As the scale changes from the smallest scale experiment to the largest scale experiment among the seven experiments, the variation in coefficient of discharge can also be studied. Several data points from each of the experiment allow the identification of uncertainties and their variation at the different scales. The overall effect of change in scale in this type of experiment and the corresponding flooding scenario can be standardized using this case study. The biggest advantage of such a study is that a lack of data at a real scale would not hurt the prediction since the variability due to several weir parameter and scaling can be accounted using the coefficient of discharge for the particular case.

In order to assess the information in the experimental data and compare the data set with respect to standard scaling parameter, plots between  $C_d$  and a non-dimensional

ratio  $h/b$  for all the experiments are shown in Figure V.9. An ideal condition would be to plot  $C_d$  with respect to a non-dimensional parameter such as Reynolds number ( $Re$ ) or Froude's number ( $Fe$ ). However, the availability of only limited parameters for each of the experiments, a non-dimensional ratio ( $h/b$ ) is considered which is quantifiable for all the experiments.

The plot for  $C_d$  vs  $h/b$  shows that each experimental data is centered around a mean with certain standard deviation. The mean and standard deviation for each of the experiment is listed in Table V.2. The degree of spread depends on experimental uncertainty and observation errors. It is difficult to eliminate any epistemic errors from the data set because the experimental data is obtained from published literature. An attempt is made to understand the scaling pattern within the data despite these unquantifiable uncertainties. Before the rigorous analysis of data, a careful consideration of outliers is necessary. Outliers are those observations which deviate noticeably compared to observations from other datasets in the sample. Outliers are problematic for many statistical analyses because they can cause tests to either miss significant findings or distort real results. Outliers may indicate a bad data due to errors created from inaccurate calibration, offset in readings, machine error, etc. If such errors are found for the outlying data, such outlying values should be either deleted or corrected. However, in the current study correction of data and identification of such errors are not possible because the data is obtained from already published sources. Hence, if outliers exist, such data set must be avoided for further consideration and analysis.

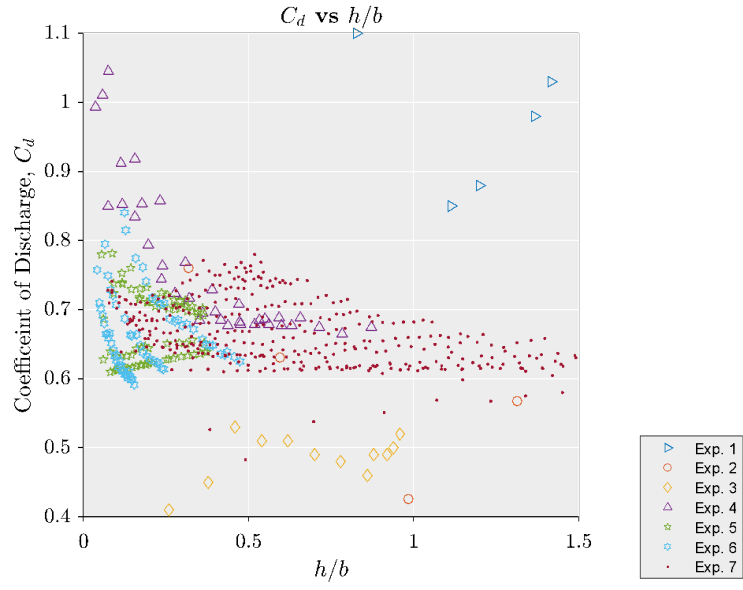


Figure V.9: Plot between  $C_d$  vs  $h/b$  for all the experiments

Table V.2: Mean and standard deviation of  $C_d$  for each experiment

Exp. No.	Mean $C_d$	Std. dev. $C_d$
1	0.968	0.104
2	0.596	0.138
3	0.486	0.033
4	0.758	0.106
5	0.672	0.048
6	0.740	0.148
7	0.656	0.018

### V.3.1 Outliers

First, an attempt is made to identify outliers in the experimental analysis. For this purpose, small scale experiments where weir opening width varies between 0.02 m to 0.06 m are considered at first. The relatively small variation in the scale does not require the data to be plotted with respect to non-dimensional quantity such as  $h/b$ ., therefore, the experimental data is plotted between  $C_d$  and  $h$  for a clear understanding of the data. Figure V.10 shows a plot between  $C_d$  and  $h$  from the experiments where the opening width varies from 0.02 to 0.06. The only experiments where the opening width ranges between 0.02 to 0.06 m are experiment 1, 2, 3 and 7, hence, the data from only these experiments is plotted in Figure V.10. Experiment 1 data lies completely outside the range of the data from other similar scale experiments in the plot. Sometimes data which deviates from the normal trend may explain about the experimental variability caused by certain entities, but in the present scenario additional data on the experiment 1 is not available which could have explained the deviation. Hence, the experiment 1 data is considered an outlier and not considered for further assessment.

No such outliers were found for the experimental data where the opening width is greater than 0.06m. Hence, no other experiment indicates a presence of outliers.

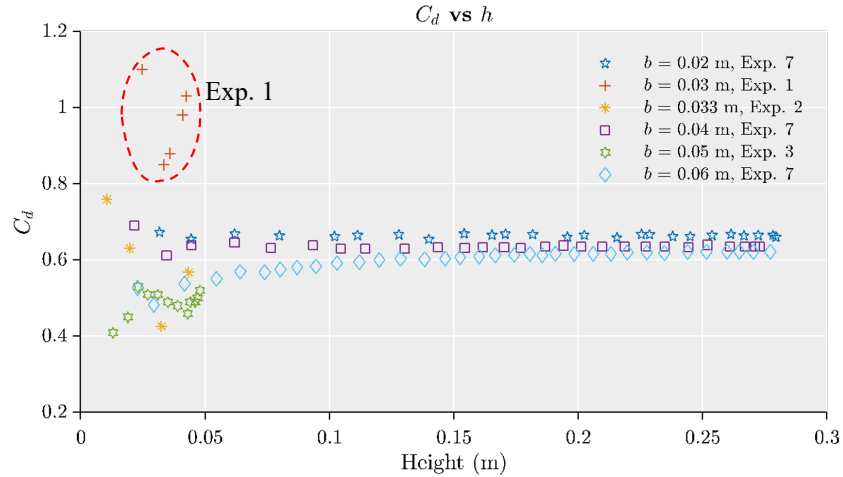


Figure V.10: Biasness observed in the Experiment 1

### V.3.2 The Effect of Scaling of Width

Figure V.11 compares  $C_d$  vs  $h/b$  plot for two different ranges of  $b$ . For lower ranges of  $b$  (Figure V.11a),  $C_d$  decreases as  $b$  increases for a given  $h/b$  ratio whereas for higher ranges of  $b$  (Figure V.11b),  $C_d$  increases as  $b$  increases for a given  $h/b$ . It is intriguing that such a change in the pattern occurs at a certain value of  $b$  and their behavior must be analyzed separately. For example, this change occurs at  $b \approx 0.08m$ . It should also be noted that  $h/b < 0.5$  has a very different pattern than the rest of the plot for each  $b$ . The possible reason for such behavior is the significant boundary effects on low water heights or extremely wide weir conditions.

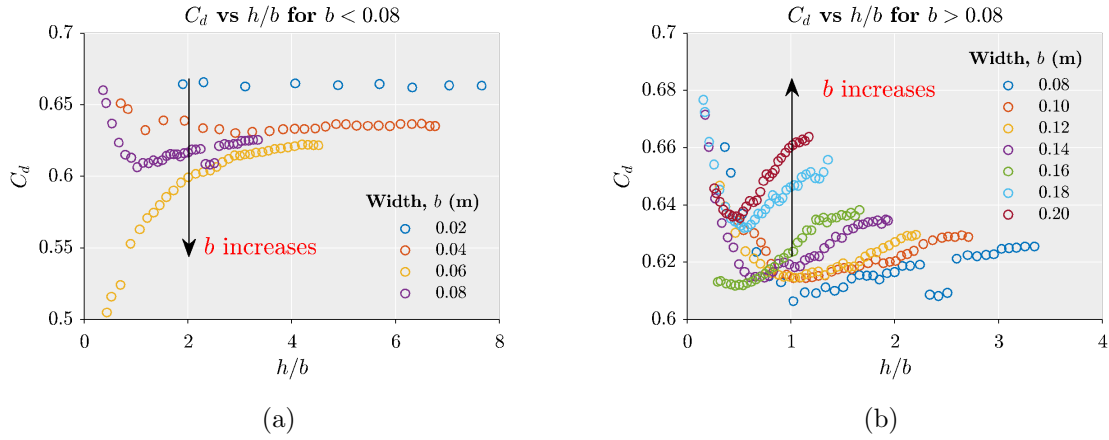


Figure V.11: Plot for  $C_d$  vs  $h/b$ : (a)  $b < 0.06$  and (b)  $b > 0.06$

$C_d$  vs  $h/b$  plot for all the opening widths more than 0.08 m ( $b/B > 0.25$ ) shows a peculiar pattern between  $C_d$  and  $h/b$  in Figure V.12a. An idealized shape for  $C_d$  vs  $h/b$  curve for  $h/b > 0.1$  can be plotted as Figure V.12b. The limit on  $h/b$  is applied because for a low  $h$  or large  $b$ , the boundary condition will change the flow characteristics. The idealized shape for  $C_d$  vs  $h/b$  plot has two segments intersecting at the lowest point of the graph (point 3). Slope of both the segments and the coordinates of point 3 can define the complete shape of  $C_d$  vs  $h/b$  curve. Hence, relation of these four quantities with the weir width is found in Figure V.13a to Figure V.13d. Use of weir width  $b$  is not appropriate when the scaling pattern is established. Hence, the parameters (slope and lowest coordinate) for idealized shape of  $C_d$  vs  $h/b$  curve are plotted with respect to dimensionless ratio  $b/B$ . Abscissa in these plots varies from 0 to 1 as weir opening width cannot exceed the overall width of the weir.

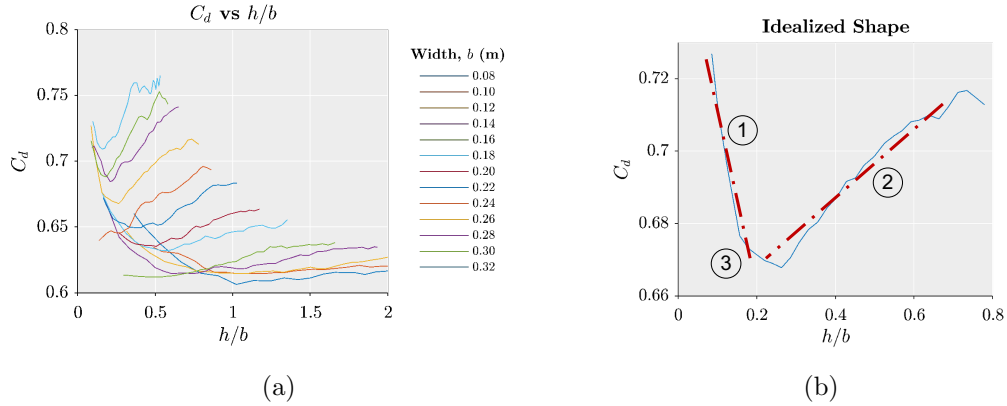


Figure V.12: (a) Plot for  $C_d$  vs  $h/b$  from all the experiments with  $b/B > 0.25$  and (b) Idealized shape for  $C_d$  vs  $h/b$

### V.3.3 The Idealization for Scaling Effect for Sharp Crested Rectangular Weir

Figure V.13 shows that slopes of segment 1 & 2 and coordinates of point 3 are linearly proportional to  $b/B$ . Slope of segment 2 and ordinate of the lowest point in curve (3) decrease as  $b/B$  increases whereas slope of segment 1 and abscissa of the lowest point increases. Linear correlation equations obtained from these 4 plots would define the  $C_d$  vs  $h/b$  curve completely. It should be noted that these relations are not valid when the ratio  $b/B < 0.25$  and  $h/b < 0.1$  as discussed earlier. Such rectangular weir with low  $b/B$  or  $h/b$  ratios would not show the same pattern (Figure V.12b) of flow as a normal rectangular weir due to boundary effects. A consideration of the effect of change in height and width scale significantly reduces the uncertainty associated with extrapolation.

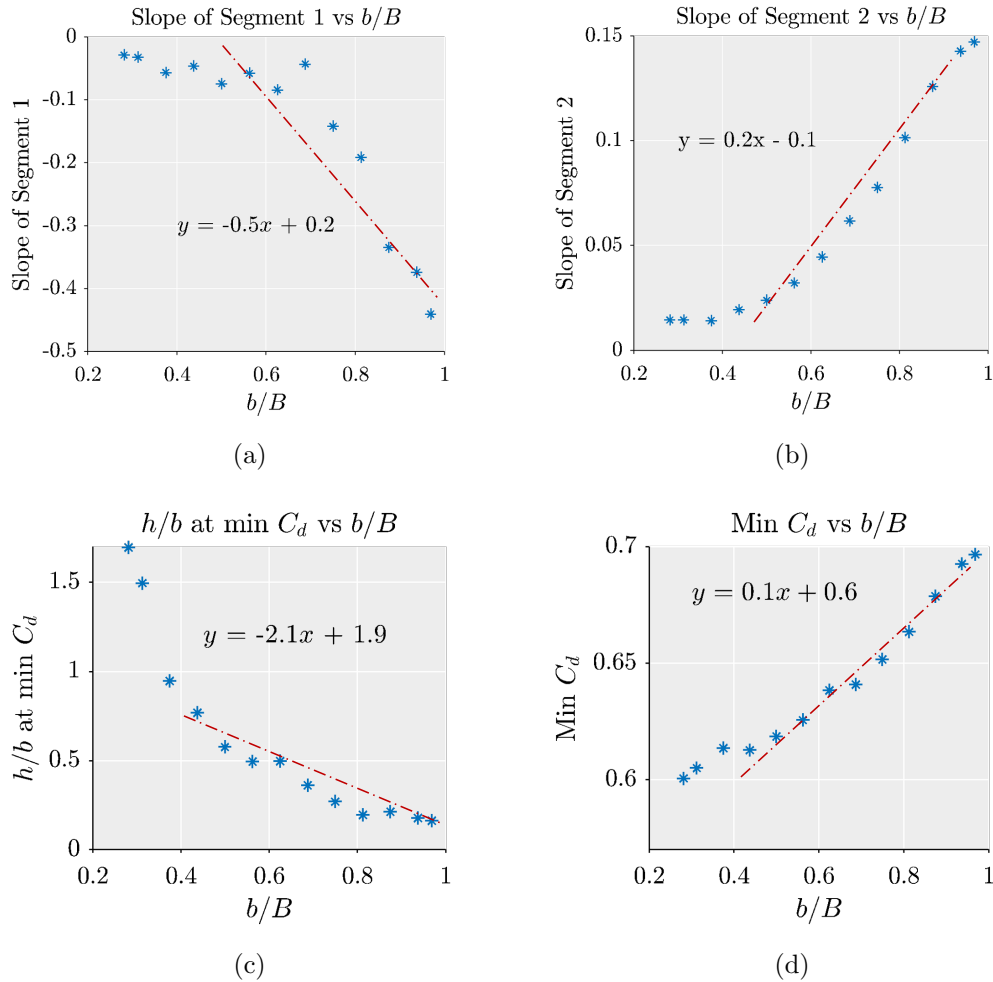


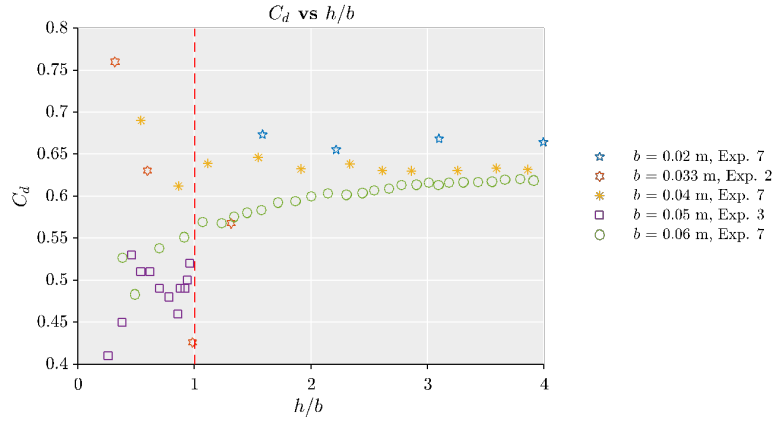
Figure V.13: Plot for parameters for  $C_d$  vs  $h$  plot: (a) Slope of segment 1 vs  $b/B$ , (b) Slope of segment 2 vs  $b/B$ , (c)  $h/b$  at the lowest point vs  $b/B$  and (d) Minimum values of  $C_d$  vs  $b/B$

The actual flow rate of flood can be interpolated with the above correlations for a given opening width and the probable water height. The relations obtained above are result of data analysis from Exp. 7. Such results could be obtained because Exp. 7 has large data set, e.g. 508 data points. The relations provide  $C_d$  vs  $h/b$  plots for any given rectangular weir in the form of a bilinear curve. One weakness of having

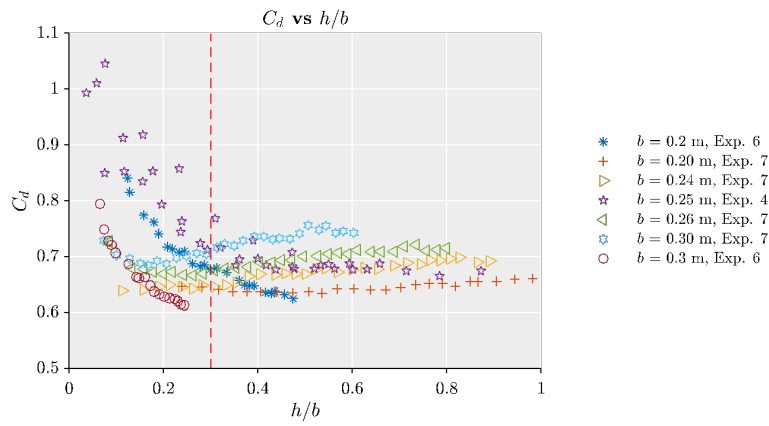
bilinear curve is that it ignores the variation that is centered about the straight lines. Hence it is necessary to determine possible variations in the form of standard deviation or probabilistic distributions. Data from other experiments is utilized to establish the understanding of these variations.

#### **V.3.4 Extrapolation: Linear or Non-Linear**

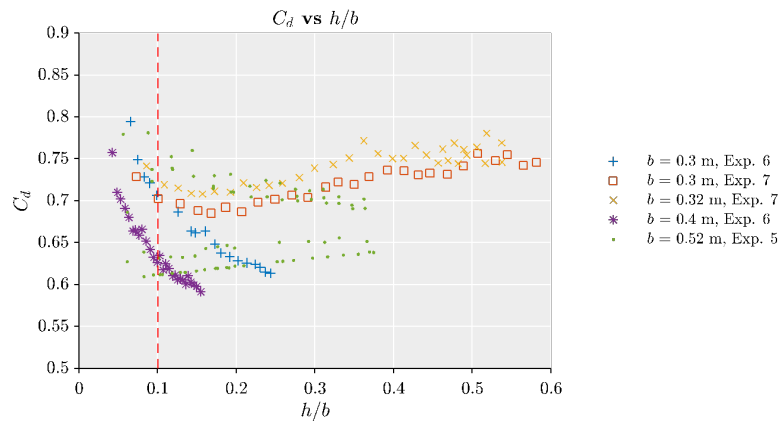
High uncertainty is associated in  $C_d$  vs  $h/b$  curve till a larger  $h/b$  limit when  $b$  is very small. As the  $b$  increases, the range of high uncertainty region also reduces. It can be observed in Figure V.14a to Figure V.14c which are plotted for different range of  $b$ . The decrease in range of region with high uncertainty in  $C_d$  vs  $h/b$  curve could be due to the diminishing effects of boundary as  $b$  increases.



(a)



(b)



(c)

Figure V.14: Plot of  $C_d$  vs  $h/b$  for various range of  $b$ : (a)  $b = 0.02\text{m} - 0.06$  m, (b)  $b = 0.2\text{m} - 0.3$  m & (c)  $b = 0.3\text{m} - 0.5$  m

The table below shows the apparent relation between the lengths of non-linear region in  $C_d$  vs  $h/b$  curve with  $b$ . It can be observed that the non-linear range in the curve expands over higher range for lower values of  $b$  compared to higher values of  $b$ . This indicates that real-world data should not be validated with small-scale experiments where  $b$  is lesser than 0.05 m for sharp crested weirs.

Table V.3: Change in Non-linear range of  $C_d$  vs  $h/b$  with  $b$

$h/b$ limit for Non-linear range	Range of $b$
0-1	0.02-0.05
0-0.4	0.2-0.3
0-0.1	0.3-0.5

### V.3.5 Effect of Width for Boundary Effects

The experiments 6 and 7 are performed for a varied set of rectangular weir parameters for conducting the discharge analysis. Hence a plot of standard deviation of  $C_d$  with respect to width can be plotted. Figure V.15 shows the plot between standard deviations of  $C_d$  with a change in width. Exp. 6 shows decrease in standard deviation with increase in  $b$  while Exp. 7 shows increase in standard deviation with increase in width. The plot for Exp. 7 in Figure V.15b shows that the increase in width allows larger possibilities of uncertainties, while the measure of these uncertainties increases linearly with respect to width. However, this observation is not valid for Exp. 6 because in addition to the change in height and discharge at each width value, the weir height  $P$  is also varied at each  $b$ . Therefore, the change in  $P$  plays an inherent role in resulting larger standard deviation. It shows that effect of change in  $P$  is less for a larger  $b$  while effect of change

in  $P$  is more for a smaller  $b$ , which indicates for a larger  $b$ , the boundary effects such as change in  $P$  have less significance in determining the flow characteristics.

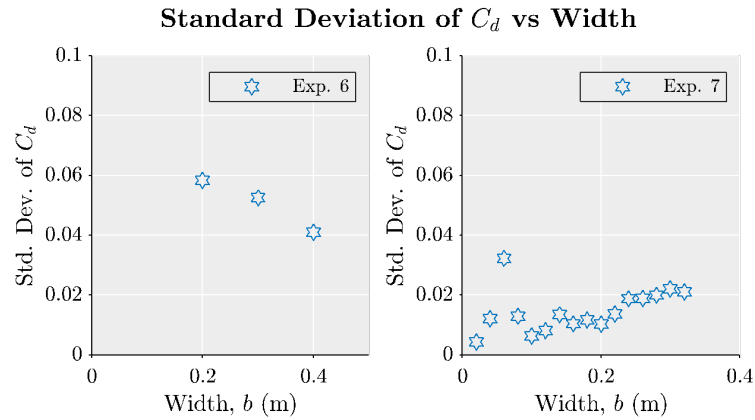


Figure V.15: Plot for standard deviation of  $C_d$  with change in width

The experiments 1, 2 & 3 are extremely small-scale experiments which lead to larger standard deviations for  $C_d$  at each width  $b$ , and for experiments 4, 5 & 6 either  $P$  changes within the data or the boundary condition changes. Boundary conditions for Exp. 7 do not change compared to other experiments except the weir width. These experiments were performed at laboratory conditions, however, at the real-world scale, it is necessary to consider variabilities in weir parameters due to inherent uncertainty in any of the input data. Therefore, the evaluation of mean for  $C_d$  through regression equations (Figure V.13) is not sufficient. The standard deviation for  $C_d$  must also be accounted. An attempt is made to evaluate the variation of standard deviation with respect to weir parameters using the experimental results.

Figure V.16a shows the standard deviation of each experiment with respect to  $b$ . The experimental results lying in non-linear range of the curve as shown in Table V.3 are removed from Figure V.16a to analyze the data that can be used for extrapolation. The

set of experimental data generated for a varied boundary conditions is also separated from the overall data in Figure V.16a to eliminate the addition of standard deviation values due to boundaries. After these removals, we obtain a revised curve for standard deviation with  $b$  as shown in Figure V.16b. The standard deviation relation with width in Figure 16b is similar to the relation shown in Figure V.15b.

Hence, we propose that the data lies in the non-linear range of the  $C_d$  vs  $b$  curve must not be considered for extrapolation. In addition, the  $h/b$  ranges causing non-linear relation only causes insignificant flooding at a real scale scenario. It should be noted that a consistency of boundary conditions cannot be ensured at the real world scale. The variation of boundary condition lead to comparatively larger values of standard deviation. Therefore, standard deviation for  $C_d$  calculated using same boundary condition for experiments, should be increased to reflect the variation in boundary condition at real-scale.

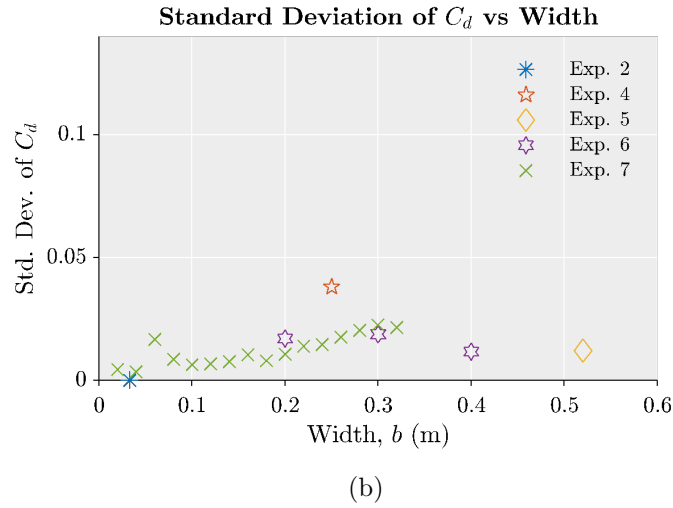
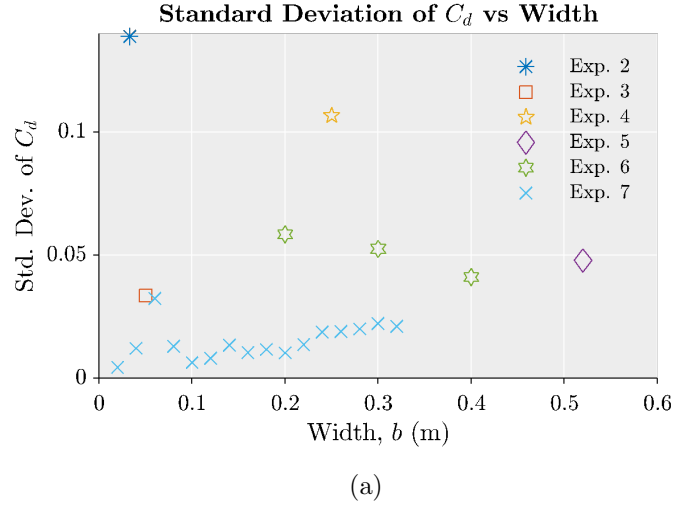


Figure V.16: Plot of standard deviation of  $C_d$  with width  $b$ : (a) All available data for the experiments and (b) Plot after removal of data impacted by boundary effect

Figure V.17 shows a plot between mean of  $C_d$  and weir width,  $b$  for each experiment. Figure V.17 shows an increase in mean of  $C_d$  with an increase in  $b$ . Eq. (V.1) shows that an increase in  $C_d$  leads to an increase in  $Q_{act}$ , therefore increasing flow efficiency of the weir. Therefore, increase in width is directly proportional increase flow efficiency. This is because the boundary effects such as turbulence at corners

causing the reduction in flow efficiency are only limited to corners of the rectangular weirs. As the width increases, their effect nullifies. Increase in mean with weir width is not linear, especially when the data from different experiments is used. Flooding event at plant site can be assumed to be a different experiment each time and hence, such an uncertainty in the mean of  $C_d$  should be included during a flooding simulation.

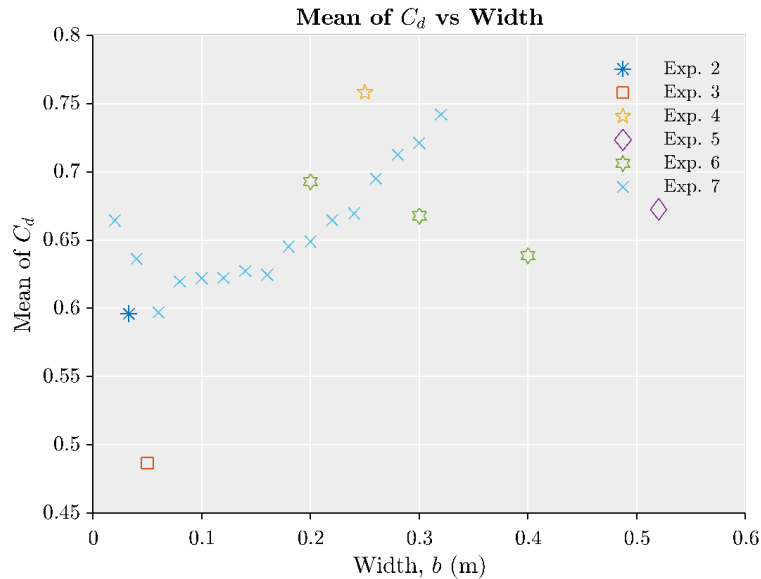


Figure V.17: Mean of  $C_d$  with increase in width

### V.3.6 Characterization of Statistical Distributions to Quantify Uncertainty

Next, uncertainty in the coefficient of discharge is characterized by fitting the statistical distributions for Lognormal and Beta distributions to the histograms of  $C_d$  in Figure V.18. The uncertainty quantification and propagation are performed based on the data from Exp. 7 because only Exp. 7 consist of sufficient data to perform the uncertainty analysis. Beta distribution is found to be a better representation for the uncertainty in  $C_d$  based on a comparison between Beta and Lognormal distributions. The fitted distributions also

show an increase in mean and standard deviation of  $C_d$  with the increase in the width.

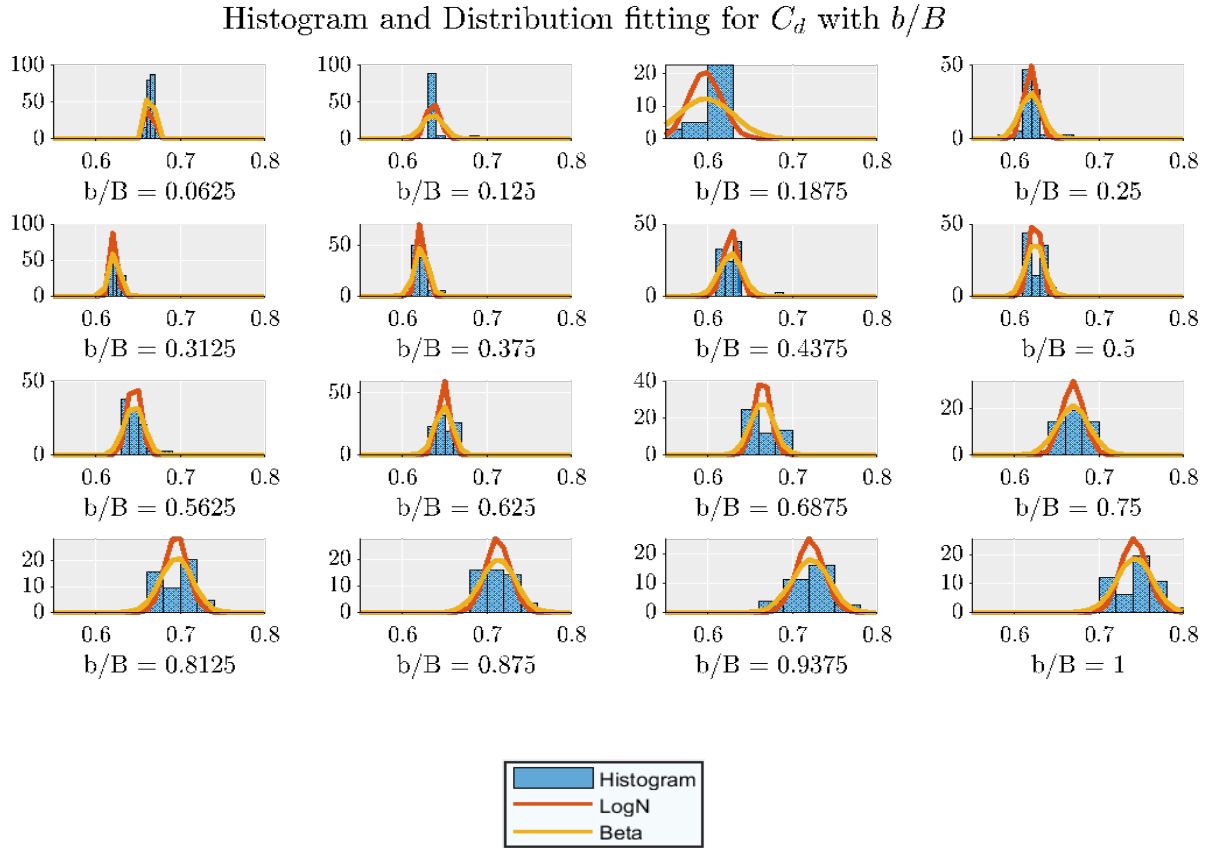


Figure V.18: Fitting the Lognormal and Beta distribution for  $C_d$

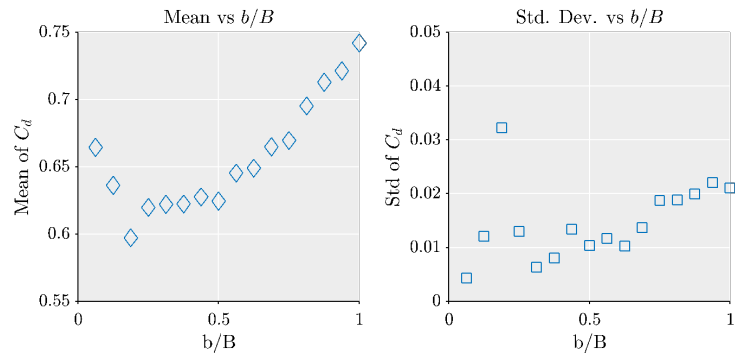
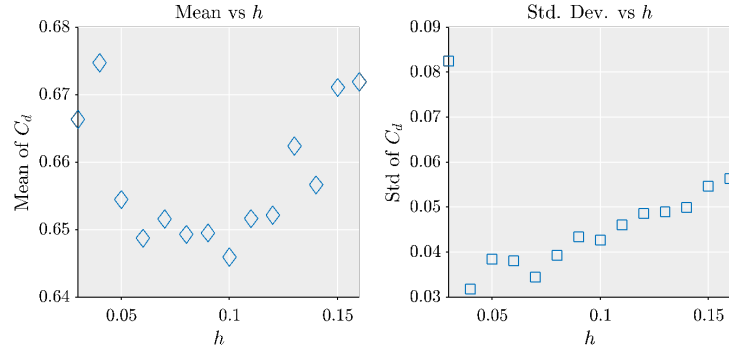


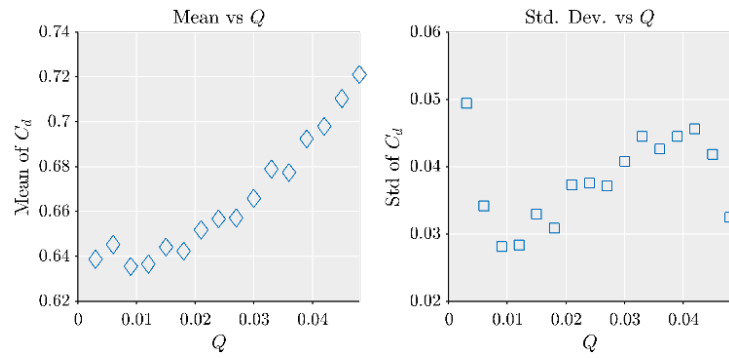
Figure V.19: Parameters for fitted Beta distribution for  $C_d$

A similar uncertainty quantification is performed with respect to  $h$ ,  $Q$  and  $h/b$  for Exp. 7 in Figure V.20a to Figure V.20c. The histograms and distributions obtained from the analysis are not shown in this paper for brevity. However, the relationship between mean and standard deviation is shown in Figure V.19 to explain the behavior. It can be seen that as  $Q$  and  $h$  increase, the standard deviation and mean of  $C_d$  also increase except for very low values of  $h$  and  $Q$  where the boundary effects would be significant. High  $h$  and  $Q$  allow for better flow efficiency, therefore mean of  $C_d$  increases with the increase in  $h$  and  $Q$ . Generally,  $h$  and  $Q$  are proportional to each other, still standard deviation of  $C_d$  has more variability with respect to  $Q$  as compared to  $h$ . There could be many reasons for this increased variability such as turbulence at corners, boundary effects and difficulty in measurement of exact  $Q$ .

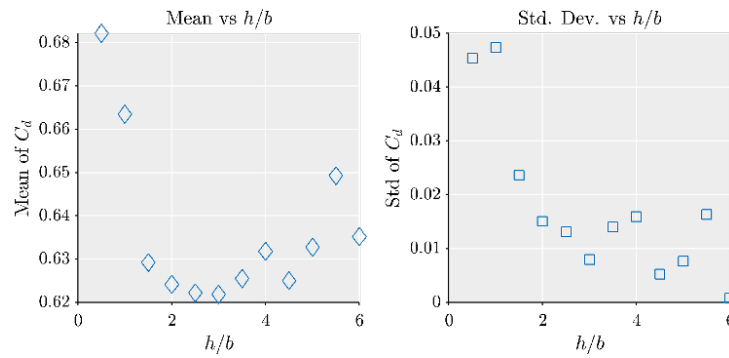
The relation between mean of  $C_d$  and  $h/b$  in Figure V.19a is similar to the bilinear curve that was obtained with individual  $b$  in Figure V.17b. Even when Figure V.19a shows a cumulative effect of all the weir widths, the curve can be represented as bilinear with certain standard deviation, whereas the standard deviation of  $C_d$  does not have certain form with respect to  $h/b$  as shown in Figure V.19b. The standard deviation is very high for low values of  $h/b$  due to boundary effects, but it just varies along a mean for higher values of  $h/b$ . It should be noted that this dimensionless quantity takes ratio of width and height which are two adjacent parameters in the experiment. Increase in  $h/b$  ratio either represent increase in  $h$  or decrease in  $b$ , which will cause  $C_d$  to increase (because of  $h$ ) and decrease (because of  $b$ ). Hence, the cumulative effect does not show any pattern for  $C_d$  with the change in  $h/b$ .



(a)



(b)



(c)

Figure V.20: Plot of mean and standard deviation of  $C_d$  with (a) water height  $h$ , (b) water discharge  $Q$  and (c) height and width ratio  $h/b$

It should be noted that the uncertainty analysis is only performed using the Exp.

7 data because the other experiments do not have enough data to show an apparent relationship of uncertainty with weir parameters. The uncertainties for other experiments can be included for completeness with an increase in standard deviation as found in the Section V.3.5. In overall summary of this study, the presented analysis quantifies the uncertainty in the evaluation of  $C_d$  and also illustrates the variation in this uncertainty at different scales limitation of reduced scale experiments in validating the large-scale experiments or simulation thereby making a case for appropriate consideration of scaling effect in uncertainty quantification at real-world scale.

#### **V.4 Summary and Conclusions**

Several advanced simulation tools used for studying real world flooding scenarios are validated through smaller scale laboratory experiments. An implicit extrapolation is performed on the smaller scale data generally beyond the domain of validation when these simulation tools are used for real-world applications. This leads to scale distortion and adds uncertainty in the prediction. Such issues are often resolved only by professional expertise. Hence, the current study is an attempt to understand scaling phenomenon to enhance and support the scaling extrapolation and reduce the reliance on heuristics. This paper introduces and implements a methodology which examines smaller scale experiments at multiple scales to infer the effect of scaling.

Available literature on experiments for flow over a rectangular weir is used to illustrate the application of the proposed methodology. Experiments are selected such that the largest width and height of rectangular weirs in the experiments have a scale that is at least 30 times larger than the smallest scale experiment. Various different sources of uncertainty are embodied in the coefficient of discharge as part of the mathematical

model.

There are two primary factors which influence the flow dynamics at the rectangular weir. First, weir parameters such as corner geometry, weir crest geometry, weir height, etc. alter the flow dynamics. However, accommodating such details for the plant site flooding event analysis is impractical. Such variations in parameters must be considered through appropriate uncertainty characterization and quantification at the real scale simulation compared to the uncertainties observed in small-scale experiments. Second is the consideration of a variation in flow characteristics as the scale of the experiment changes. Uncertainties in the estimation of coefficient of discharge increase linearly as the scale of the experiment increases for a certain range with respect to discharge ( $Q$ ), water height ( $h$ ) or weir width ( $b$ ). Such uncertainty propagation is reversed when extremely small-scale experiments are performed i.e., uncertainty decreases as the scale increases. It is also found that the effect of geometry discussed earlier becomes less significant as the scale of the experiment is increased whereas for very small scale experiment these are significant influence of due to boundary effects. Therefore, a validation of real-world scale flooding simulation should not be performed based on extremely small-scale experiments where the flow characteristics are completely dissimilar.

Uncertainty in coefficient of discharge is characterized using beta distribution since it fits the data most appropriately for experiments at different scales. The effect of scaling presented in the study at a small scale can be extended for use at plant level flooding to minimize the uncertainties associated with scale distortion if an appropriate scale of experiment is used. The paper establishes a methodology to improve the validation of advanced simulations by illustrating the analysis on multiple smaller scale experiments.

---

---

## PART VI

---

### Summary, Conclusions, and Recommendations for Future Research

## VI.1 Summary and Conclusions

This dissertation presents computationally efficient approaches for uncertainty characterization and its propagation in external hazard probabilistic risk assessment. This dissertation includes four main manuscripts that focus on describing the following key topics:

- Limitations of traditional tools for beyond design basis external hazard PRA
- Computationally efficient approach for risk-informed decision making
- Computationally efficient approach for multi-hazard and multi-unit probabilistic risk assessment
- Understanding uncertainty scaling for flooding PRA based on experimental data

The summary and conclusions for each of these manuscripts are given below:

### VI.1.1 Limitations of Traditional Tools for Beyond Design Basis External Hazard PRA

This manuscript illustrates the limitations of the traditional PRA tools for event tree and fault tree analysis in the context of BDB scenario. The approximations due to key assumptions in traditional PRA approaches are studied and their impact on the evaluation of critical accident sequences is illustrated. A simple illustrative example is used to compare the outcome of traditional approach with that from an exact quantification. This comparison is used to demonstrate the impact of fundamental assumptions in the traditional PRA approaches on risk quantification. An alternative method based on fundamentals of logic gate properties and Bayesian network is presented and implemented for delivering an accurate assessment of failure probabilities

in PRA. The key conclusions of this study are summarized as follows:

- The following two key assumptions in the traditional PRA approaches lead to discrepancies in the estimation of failure probabilities for BDB scenarios,
  - Probabilities of failure for basic events in fault and event trees are very small.
  - All the component failures are assumed to be independent of each other.
- It is found that the traditional approach for an event tree analysis can lead to inaccurate estimation of failure probabilities of accident sequences and incorrect identification of the critical path in the context of BDB scenario.
- The limitations incurred due to fundamental assumptions in traditional fault and event tree analysis approaches can be mitigated by representing and analyzing the trees with Bayesian networks.
- It is shown that the results obtained from Bayesian network computation are identical to that of the exact quantification using set theory.
- A simple top-down algorithm using fundamentals of logic gate properties is introduced to compute failure sequence probability in the Bayesian network representation of PRA.
- A comparison of top event probabilities calculated using traditional PRA tools and Bayesian network analysis showed that traditional approximation methods give accurate results for the case of DB scenario but can lead to incorrect risk estimates in the case of BDB scenario.

## VI.1.2 Computationally Efficient Approach for Risk-informed Decision Making

Fault and event trees are used for probabilistic risk assessment of nuclear power plant systems. A fault and event trees analysis for a power plant requires modelling of hundreds of component failures, logic gates, multiple occurring events, and dependent events. Such interconnection for large networks can lead to excessive computational demand. Most of the traditional methods address computational demand by using assumptions or rely on high performance computing facilities that allow implementations of parallel computing. A novel framework for analyzing fault and event tree is presented to address the computational demands of PRA without incorporating assumptions and approximations. The key conclusions of this study are summarized as follows:

- The logic gates of the fault and event trees are converted to truth tables notations for achieving the desired efficiency in the proposed algorithm.
- A demonstration for the implementation of proposed algorithm for various simple and complex fault tree examples is provided.
- It is shown that the computational demand of proposed algorithm increases only linearly with the size of the tree in terms of the total number of basic events and intermediate events.
- The following two kinds of trees are used to compare the computational demand of the proposed approach with traditional fault and event tree analysis approaches:
  - Trees with same number of basic events and intermediate events- the computational demand of traditional approaches changes based on different

connectivity of events and gates, while the computational demand of proposed algorithm remains constant.

- Trees with fixed height and fixed number of logic gates (intermediate events) but varying widths by increasing the number of basic events- the computational demand of proposed algorithm increases at a rate that is a linear function of the number of basic events, while the computational demand of traditional approach increases exponentially as  $O(2^n)$ .
- It is also shown that the proposed algorithm gives identical results to those obtained by the traditional approaches and, for a simple case, identical to that calculated using fundamental concept of set theory.
- The proposed algorithm is developed for networks with a dependent node and multiple chains connecting one dependent node and provide efficient results for only these cases.
- It is recommended that the proposed algorithm needs to be enhanced for multi-hazard scenarios such as consideration of common cause failures, nested loop structure and  $n/m$  logic gates.

### VI.1.3 Computationally Efficient Approach for Multi hazard and Multi Unit Probabilistic Risk Assessment

This study proposes enhancement to the work presented in Chapter IV to provide solution for the commonly observed external hazard scenario in a typical PRA particularly concerning multi-hazard and multi-unit accident that were not addressed. The primary conclusions of this study can be summarized as follows:

- An enhancement is proposed for the analysis of nested loop structure in fault and event trees that can be commonly observed in multi-hazard and multi-unit PRA.
- A comparison of computational demand of the proposed approach with traditional approaches for trees with nested loop is shown with an illustrative example where the width of the tree is increased by increasing the number of basic events. The computational demand of proposed algorithm increases only linearly with the size of the tree while it is comparatively high with the traditional approach.
- Another enhancement is proposed for the analysis of common cause failure scenario in PRA.
- The computational time for the proposed enhancement for CCF increases with more than a linear order because an intersection is performed at the top node of the loops connecting CCF events. However the computational demand still remains less than that of traditional approach by an order of magnitude.
- Lastly, an enhancement is proposed for the analysis of  $n/m$  gates typically present in fault and event trees networks in PRA. The proposed enhancement for  $n/m$  gates does not offer computational advantages over the traditional algorithms due to the complexity of the type of the gate.

#### **VI.1.4 Understanding Uncertainty Scaling for Flooding PRA based on Experimental Data**

This study is performed to understand scaling phenomenon to enhance and support the scaling extrapolation and reduce the reliance on heuristics. This paper introduces and implements a methodology which examines smaller scale experiments at multiple scales to infer the effect of scaling. The key conclusions of this study are summarized as follows:

- Available literature on experiments for flow over a rectangular weir is used to illustrate the application of the proposed methodology.
- Various different sources of uncertainties are embodied in the coefficient of discharge as part of the mathematical model.
- It is found that the uncertainty propagation is reversed when extremely small-scale experiments are performed i.e., uncertainty decreases as the scale increases.
- It is also found that the effect of corner or weir crest geometry becomes less significant as the scale of the experiment is increased whereas it influences the coefficient of discharge significantly the scale of the experiment is very small due to the enhanced boundary effects.
- It is recommended that a validation of real-world scale flooding simulation should not be performed based on extremely small-scale experiments where the flow characteristics becomes completely dissimilar.
- The effect of scaling presented in the study at a small scale experiment can be extended for use at plant level flooding to minimize the uncertainties associated with scale distortion if an appropriate scale of experiment is used.
- The paper establishes a methodology to improve the validation of advanced

simulations by illustrating the analysis on multiple smaller scale experiments.

## VI.2 Recommendations for Future Research

Based on the experience gained in conducting this research, it is suggested that further research should consider the following aspects:

- The simple top down algorithm proposed in Chapter II is verified for only accuracy. A further study can be performed for evaluating its efficiency compared to other fault and event tree analysis approaches.
- The computationally efficient bottom-up approach proposed in Chapter III and IV should be compared with commercially available software for PRA such as, EPRI CAFTA, ISOGRAPH FAULT+, OPENFTA, etc. [14].
- A similar approach should be developed for an efficient identification of the dependent loop, complex loop, nested loop, etc. scenarios in the fault or event tree networks.
- Even though the proposed bottom-up approach showed efficient results for illustrative examples with hundreds of events, the real-world problems should be analyzed to address any unforeseen limitations in the approach. If any limitations are found for realistic examples then these must be addressed.
- The proposed approach should be compared with PRA approaches other than traditional MOCUS algorithms. These efficient approaches can include but are not limited to, Bayesian networks, BDDs, monte carlo simulation, etc.
- The scaling of uncertainties are studied for only a single flooding scenario. A similar study should be performed for other flooding scenarios.

## REFERENCES

- [1] H. SHIBATA, D. LAPPA, P. PINTO, R. BUDNITAZ, A. IBRAHIM ET AL., “Probabilistic safety assessment for seismic events (IAEA-TECDOC-724),” *Vienna, Austria: International Atomic Energy Agency* (1993).
- [2] J. W. REED and R. P. KENNEDY, “Methodology for developing seismic fragilities,” *Final Report TR-103959, EPRI* (1994).
- [3] N. SIU, M. STUTZKE, S. DENNIS, and D. HARRISON, *Probabilistic risk assessment and regulatory decision making: Some frequently asked questions*, United States Nuclear Regulatory Commission, Office of Nuclear Regulatory, NUREG-2201 (2016).
- [4] A. H.-S. ANG and W. H. TANG, “Probability concepts in engineering planning and design, vol. 2: Decision, risk, and reliability.” *John Wiley & Sons, Inc., 605 Third Ave., New York, NY 10158, USA, 1984, 608* (1984).
- [5] R. FERDOUS, F. KHAN, R. SADIQ, P. AMYOTTE, and B. VEITCH, “Fault and Event Tree Analyses for Process Systems Risk Analysis: Uncertainty Handling Formulations,” *Risk Analysis*, **31**, 1, 86 (2011); <https://doi.org/10.1111/j.1539-6924.2010.01475.x>.
- [6] R. KENNEDY and M. RAVINDRA, “Seismic fragilities for nuclear power plant risk studies,” *Nuclear Engineering and Design*, **79**, 1, 47 (1984); [https://doi.org/10.1016/0029-5493\(84\)90188-2](https://doi.org/10.1016/0029-5493(84)90188-2).
- [7] M. MODARRES and H. DEZFULI, “A truncation methodology for evaluating large fault trees,” *IEEE transactions on reliability*, **33**, 4, 325 (1984).

- [8] P. BUCCI, J. KIRSCHENBAUM, L. A. MANGAN, T. ALDEMIR, C. SMITH, and T. WOOD, “Construction of event-tree/fault-tree models from a Markov approach to dynamic system reliability,” *Reliability Engineering & System Safety*, **93**, 11, 1616 (2008).
- [9] S. LEE, “Analysis of Severe Accidents in Pressurized Heavy Water Reactors,” IAEA-TECDOC-1594, Vienna (2008).
- [10] A. BARTO, Y. J. CHANG, K. COMPTON, H. ESMAILI, D. HELTON, A. MURPHY, A. NOSEK, J. PIRES, F. SCHOFER, and B. WAGNER, *Consequence study of a beyond-design-basis earthquake affecting the spent fuel pool for a US Mark I boiling water reactor*, United States Nuclear Regulatory Commission, Office of Nuclear Regulatory, NUREG-2161 (2014).
- [11] Y. TIAN, C.-C. CHAO, C.-C. HSU, P.-J. CHIU, Y.-T. CHIOU, and T.-S. LIN, “The Study of Spent Fuel Pool Risk at Decommissioning Nuclear Power Plant in Taiwan,” *Probabilistic Safety Assessment and Management PSAM 14* (2018).
- [12] M. VAN DER MEULEN, “On the use of smart sensors, common cause failure and the need for diversity,” *6th International Symposium Programmable Electronic Systems in Safety Related Applications*, Citeseer (2004).
- [13] H. H. HWANG, “Seismic probabilistic risk assessment and seismic margins studies for nuclear power plants,” *Probabilistic engineering mechanics*, **3**, 4, 170 (1988).
- [14] E. RUIJTERS and M. STOELINGA, “Fault tree analysis: A survey of the state-of-the-art in modeling, analysis and tools,” *Computer science review*, **15**, 29 (2015).

- [15] F. SIHOMBING and M. TORBOL, "Parallel fault tree analysis for accurate reliability of complex systems," *Structural Safety*, **72**, 41 (2018).
- [16] S. CONTINI and V. MATUZAS, "Analysis of large fault trees based on functional decomposition," *Reliability engineering & system safety*, **96**, 3, 383 (2011).
- [17] P. CHATTERJEE, "MODULARIZATION OF FAULT TREES: A METHOD TO REDUCE THE COST OF ANALYSIS." (1975).
- [18] S. B. AKERS, "Binary decision diagrams," *IEEE Computer Architecture Letters*, **27**, 06, 509 (1978).
- [19] F. I. KHAN and S. ABBASI, "Analytical simulation and PROFAT II: a new methodology and a computer automated tool for fault tree analysis in chemical process industries," *Journal of Hazardous Materials*, **75**, 1, 1 (2000).
- [20] R. M. SINNAMON and J. ANDREWS, "Improved efficiency in qualitative fault tree analysis," *Quality and Reliability Engineering International*, **13**, 5, 293 (1997).
- [21] Z. LI, Y. REN, L. LIU, and Z. WANG, "Parallel algorithm for finding modules of large-scale coherent fault trees," *Microelectronics Reliability*, **55**, 9-10, 1400 (2015).
- [22] S. CHEN, J. WANG, J. WANG, F. WANG, and L. HU, "Efficient reduction and modularization for large fault trees stored by pages," *Annals of Nuclear Energy*, **90**, 22 (2016).
- [23] Y. DENG, H. WANG, and B. GUO, "BDD algorithms based on modularization for fault tree analysis," *Progress in Nuclear Energy*, **85**, 192 (2015).
- [24] S. KABIR, "An overview of fault tree analysis and its application in model based dependability analysis," *Expert Systems with Applications*, **77**, 114 (2017).

- [25] H. AGHASSI and F. AGHASSI, “Fault tree analysis speed-up with GPU parallel computing,” *Computer Information Systems and Industrial Management Applications, Tehran*, **5**, 106 (2012).
- [26] F. CAVALIERI, P. FRANCHIN, P. GEHL, and D. D’AYALA, “Bayesian networks and infrastructure systems: Computational and methodological challenges,” *Risk and reliability analysis: theory and applications*, 385–415, Springer.
- [27] W. BUNTINE, “Theory refinement on Bayesian networks,” *Uncertainty Proceedings 1991*, 52–60, Elsevier.
- [28] A. DARWICHE, “A differential approach to inference in Bayesian networks,” *Journal of the ACM (JACM)*, **50**, 3, 280 (2003).
- [29] D. KAHLE, T. SAVITSKY, S. SCHNELLE, and V. CEVHER, “Junction tree algorithm,” *Stat*, **631** (2008).
- [30] M. BENSI, A. DER KIUREGHIAN, and D. STRAUB, “Efficient Bayesian network modeling of systems,” *Reliability Engineering & System Safety*, **112**, 200 (2013).
- [31] J.-E. YANG, “Fukushima Dai-Ichi accident: lessons learned and future actions from the risk perspectives,” *Nuclear Engineering and Technology*, **46**, 1, 27 (2014).
- [32] Y. CAI and M. W. GOLAY, “Multiunit nuclear power plant accident scenarios and improvements including those based upon interviews with TEPCO engineers concerning the 2011 Fukushima accidents,” *Nuclear Engineering and Design*, **365**, 110707 (2020).
- [33] J.-E. YANG, “Multi-unit risk assessment of nuclear power plants: Current status and issues,” *Nuclear Engineering and Technology*, **50**, 8, 1199 (2018).

- [34] M. MODARRES, T. ZHOU, and M. MASSOUD, “Advances in multi-unit nuclear power plant probabilistic risk assessment,” *Reliability Engineering & System Safety*, **157**, 87 (2017).
- [35] T. SRINIVASAN and T. G. RETHINARAJ, “Fukushima and thereafter: Reassessment of risks of nuclear power,” *Energy policy*, **52**, 726 (2013).
- [36] H. WANG and M. J. DRUZDZEL, “User interface tools for navigation in conditional probability tables and elicitation of probabilities in Bayesian networks,” *arXiv preprint arXiv:1301.4430* (2013).
- [37] Z. MA, C. SMITH, and S. PRESCOTT, “A simulation-based dynamic approach for external flooding analysis in nuclear power plants,” *International Conference Pacific Basin Nuclear Conference*, 55–69, Springer (2016).
- [38] G. PALMIOTTI and M. SALVATORES, “The role of experiments and of sensitivity analysis in simulation validation strategies with emphasis on reactor physics,” *Annals of Nuclear Energy*, **52**, 10 (2013).
- [39] W. L. OBERKAMPF, T. G. TRUCANO, and C. HIRSCH, “Verification, validation, and predictive capability in computational engineering and physics,” *Appl. Mech. Rev.*, **57**, 5, 345 (2004).
- [40] N. DINH, H. ABDEL-KHALIK, A. GUPTA, X. SUN, I. BOLOTNOV, J. BAUGH, M. AVRAMOVA, P. BARDET, R. YOUNGBLOOD, C. RABITI ET AL., “Development and Application of a Data-Driven Methodology for Validation of Risk-Informed Safety Margin Characterization Models,” *North Carolina State University* (2015).
- [41] S. IVANOV, A. HERMS, and G. LUKAS, “Experimental validation of the ns-2

- wireless model using simulation, emulation, and real network,” *Communication in Distributed Systems-15. ITG/GI Symposium*, 1–12, VDE (2007).
- [42] G. BARONE, A. BUONOMANO, C. FORZANO, and A. PALOMBO, “Building energy performance analysis: an experimental validation of an in-house dynamic simulation tool through a real test room,” *Energies*, **12**, 21, 4107 (2019).
- [43] L. LIN, N. MONTANARI, S. PRESCOTT, R. SAMPATH, H. BAO, and N. DINH, “Adequacy evaluation of smoothed particle hydrodynamics methods for simulating the external-flooding scenario,” *Nuclear Engineering and Design*, **365**, 110720 (2020).
- [44] H. BAO, R. YOUNGBLOOD, H. ZHANG, N. DINH, L. LIN, and J. LANE, “A Data-driven Approach to Scale Bridging in System Thermal-hydraulic Simulation,” *NURETH-18. Portland, Oregon* (2019).
- [45] U. N. R. COMMISSION, “A Guide to the Performance of Probabilistic Risk Assessments for Nuclear Power Plants,” *Washington DC NUREG/CR-2300* (1983).
- [46] IAEA-CORPORATE-AUTHOR, *Development and Application of Level 1 Probabilistic Safety Assessment for Nuclear Power Plants Specific Safety Guide*, International Atomic Energy Agency (2010).
- [47] K. TAKARAGI, R. SASAKI, and S. SHINGAI, “An algorithm for obtaining simplified prime implicant sets in fault-tree and event-tree analysis,” *IEEE Transactions on Reliability*, **32**, 4, 386 (1983).
- [48] E. LINDER, G. PATIL, and D. S. VAUGHAN, “Application of event tree risk analysis to fisheries management,” *Ecological Modelling*, **36**, 1-2, 15 (1987).

- [49] O. P. NUSBAUMER, “Analytical solutions of linked fault tree probabilistic risk assessments using binary decision diagrams with emphasis on nuclear safety applications,” PhD Thesis, ETH Zurich (2007).
- [50] J. L. AKINODE and S. OLORUNTOBA, “Algorithms for Reducing Cut Sets in Fault Tree Analysis,” *International Journal of Advanced Research in Computer and Communication Engineering*, **6**, 12 (2017).
- [51] M. L. MCKELVIN JR, “A methodology and tool support for the design and evaluation of fault tolerant, distributed embedded systems,” PhD Thesis, UC Berkeley (2011).
- [52] C. SMITH, S. WOOD, W. GALYEAN, J. SCHROEDER, and M. SATTISON, “Saphire 8 volume 2-technical reference (No. INL/EXT-09-17010),” Idaho National Laboratory (INL) (2011).
- [53] S. QIU, Y. HOU, and H. X. MING, “An implicit method for probabilistic common-cause failure analysis using Bayesian Network,” *IFAC-PapersOnLine*, **51**, 24, 1037 (2018).
- [54] Y. XIAOWEI, “Common Cause Failure Model of System Reliability Based on Bayesian Networks.” *International Journal of Performability Engineering*, **6**, 3 (2010).
- [55] I. VRBANIC, I. KOSUTIC, I. VUKOVIC, Z. SIMIC, and K. NUCLEAR POWER PLANT KRSKO, “Presentation of common cause failures in fault tree structure of Krsko PSA: An historical overview,” *International Conference—Nuclear Energy for New Europe*, 445–452, Nuclear Society of Slovenia (2003).

- [56] C. GENTILLON, D. RASMUSON, S. EIDE, and T. WIERMAN, “Common-Cause Failure Analysis for Reactor Protection System Reliability Studies,” Idaho National Engineering and Environmental Lab., Idaho Falls, ID (US) (1999).
- [57] H. KUMAMOTO, *Satisfying safety goals by probabilistic risk assessment*, Springer Science & Business Media (2007).
- [58] U. S. NRC, “Common-cause failure database and analysis system: event data collection, Classification, and coding,” NUREG/CR-6268. Washington, DC: US Nuclear Regulatory Commission (2007).
- [59] U. S. NRC, *Common-cause failure parameter estimations, 2015 Update*, Safety Programs Division, Office for Analysis and Evaluation of Operational (2016)URL <http://nrcoe.inel.gov/resultsdb/ParamEstSpar/>.
- [60] K. DOHMEN, “Inclusion-exclusion and network reliability,” *the electronic journal of combinatorics*, R36–R36 (1998).
- [61] J. COLLET, “Some remarks on rare-event approximation,” *IEEE Transactions on Reliability*, **45**, 1, 106 (1996).
- [62] S. KWAG and A. GUPTA, “Probabilistic risk assessment framework for structural systems under multiple hazards using Bayesian statistics,” *Nuclear Engineering and Design*, **315**, 20 (2017); 10.1016/j.nucengdes.2017.02.009., URL <http://dx.doi.org/10.1016/j.nucengdes.2017.02.009>.
- [63] A. BOBBIO, L. PORTINALE, M. MINICHINO, and E. CIANCAMERLA, “Improving the analysis of dependable systems by mapping fault trees into Bayesian networks,” *Reliability Engineering & System Safety*, **71**, 3, 249 (2001).

- [64] I. TIEN and A. DER KIUREGHIAN, “Algorithms for Bayesian network modeling and reliability assessment of infrastructure systems,” *Reliability Engineering & System Safety*, **156**, 134 (2016).
- [65] Y. TONG and I. TIEN, “Algorithms for Bayesian network modeling, inference, and reliability assessment for multistate flow networks,” *Journal of Computing in Civil Engineering*, **31**, 5, 04017051 (2017).
- [66] M. CHAVIRA and A. DARWICHE, “Compiling Bayesian Networks Using Variable Elimination.” *IJCAI*, 2443–2449 (2007).
- [67] A. MENDIBURU, R. SANTANA, and J. LOZANO, “Introducing belief propagation in estimation of distribution algorithms: A parallel framework,” *Department of Computer Science and Artificial Intelligence, University of the Basque Country, Tech. Rep. EHU-KAT-IK-11/07* (2007).
- [68] P. F. FELZENSZWALB and D. P. HUTTENLOCHER, “Efficient belief propagation for early vision,” *International journal of computer vision*, **70**, 1, 41 (2006).
- [69] D. MANDELLI, A. YILMAZ, T. ALDEMIR, K. METZROTH, and R. DENNING, “Scenario clustering and dynamic probabilistic risk assessment,” *Reliability Engineering & System Safety*, **115**, 146 (2013).
- [70] S. G. KIM, Y. G. NO, and P. H. SEONG, “Prediction of severe accident occurrence time using support vector machines,” *Nuclear Engineering and Technology*, **47**, 1, 74 (2015).
- [71] H. C. KUNREUTHER, V. M. BIER, J. R. PHIMISTER ET AL., *Accident precursor*

*analysis and management: reducing technological risk through diligence*, National Academies Press (2004).

- [72] S. JANG, S. PARK, and M. JAE, “Development of an Accident Sequence Precursor Methodology and its Application to Significant Accident Precursors,” *Nuclear Engineering and Technology*, **49**, 2, 313 (2017).
- [73] S. AUTHÉN, O. BÄCKSTRÖM, J.-E. HOLMBERG, M. PORTHIN, and T. TYRVÄINEN, “Modelling of Digital I&C, MODIG—Interim report 2015,” *NKS-361, Nordic nuclear safety research (NKS)*, Roskilde (2016).
- [74] S. A. KAMAL and D. J. HILL, “Fault tree analysis of the EBR-II reactor shutdown system,” Argonne National Lab., IL (United States) (1992).
- [75] D. D. LAVA, D. D. S. BORGES, A. C. F. GUIMARÃES, and M. D. L. MOREIRA, “Reliability study of the auxiliary feed-water system of a pressurized water reactor by faults tree and Bayesian Network,” *International Nuclear Atlantic Conference - INAC* (2017).
- [76] C. D. HEISING and S. C. DINSMORE, “A method for quantifying the reliability of instrumentation for engineered safety features to improve accident sequence recognition in reactor control rooms,” *Nuclear Engineering and Design*, **74**, 2, 287 (1983).
- [77] J. H. PURBA, J. LU, D. RUAN, and G. ZHANG, “A failure possibility-based reliability algorithm for nuclear safety assessment by fault tree analysis,” *The 1st International Workshop on Safety & Security Risk Assessment and Organizational Cultures* (2012).

- [78] V. DE VASCONCELOS, W. A. SOARES, A. C. L. DA COSTA, and A. L. RASO, “Treatment of Uncertainties in Probabilistic Risk Assessment,” *Reliability and Maintenance-an Overview of Cases*, IntechOpen (2019).
- [79] R. E. BARLOW and P. CHATTERJEE, “Introduction to fault tree analysis,” California Univ Berkeley Operations Research Center (1973).
- [80] J. FUSSELL, E. HENRY, and N. MARSHALL, “MOCUS: A computer program to obtain minimal sets from fault trees,” Aerojet Nuclear Co., Idaho Falls, Idaho (USA) (1974).
- [81] A. RAUZY, “New algorithms for fault trees analysis,” *Reliability Engineering & System Safety*, **40**, 3, 203 (1993).
- [82] A. ROSENTHAL, “Decomposition methods for fault tree analysis,” *IEEE Transactions on Reliability*, **29**, 2, 136 (1980).
- [83] E. E. LEWIS and F. BÖHM, “Monte Carlo simulation of Markov unreliability models,” *Nuclear engineering and design*, **77**, 1, 49 (1984).
- [84] P. VAISHANAV, A. GUPTA, and S. S. BODDA, “Limitations of traditional tools for beyond design basis external hazard PRA,” *Nuclear Engineering and Design*, **370**, 110899 (2020).
- [85] Y. DUTUIT and A. RAUZY, “A linear-time algorithm to find modules of fault trees,” *IEEE Transactions on Reliability*, **45**, 3, 422 (1996).
- [86] T. KOHDA, E. J. HENLEY, and K. INOUE, “Finding modules in fault trees,” *IEEE Transactions on Reliability*, **38**, 2, 165 (1989).

- [87] A. KUMAR, A. KUMAR, S. CHOUDHARY, and P. VARDE, “Optimization of binary decision diagram using genetic algorithm,” *2010 2nd International Conference on Reliability, Safety and Hazard-Risk-Based Technologies and Physics-of-Failure Methods (ICRESH)*, 168–175, IEEE (2010).
- [88] I. RCPP, L. RCPP, R. A. D. SILKWORTH, J. ORMEROD, M. J. ORMEROD, and C. SYSTEMREQUIREMENTS, “Package ‘FaultTree’,” .
- [89] W.-S. LEE, D. L. GROSH, F. A. TILLMAN, and C. H. LIE, “Fault tree analysis, methods, and applications a review,” *IEEE transactions on reliability*, **34**, 3, 194 (1985).
- [90] R. HUGHES, “A new approach to common cause failure,” *Reliability Engineering*, **17**, 3, 211 (1987).
- [91] U. N. R. COMMISSION ET AL., “Procedures for treating common-cause failure in safety and reliability studies: analytical background and techniques, vol. 2,” NUREG/CR-4780, EPRI NP-5613 (1989).
- [92] R. CORPORATION, “Treating common cause failure in fault trees,” (2005)URL <https://www.weibull.com/hotwire/issue54/relbasics54.htm>, [Online; posted August-2005].
- [93] R. CORPORATION, “The Parametric Models for Common Cause Failure Analysis,” (2011)URL <https://www.weibull.com/hotwire/issue125/hottopics125.htm>, [Online; posted July-2011].
- [94] S. MAHADEVAN, R. ZHANG, and N. SMITH, “Bayesian networks for system reliability reassessment,” *Structural Safety*, **23**, 3, 231 (2001).

- [95] D. M. RASMUSON and D. L. KELLY, “Common-cause failure analysis in event assessment,” *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, **222**, 4, 521 (2008).
- [96] A. C. TWORT, D. D. RATNAYAKA, and M. J. BRANDT, *Water supply*, Elsevier (2000).
- [97] S. GHARAJJEH, “Experimental investigation on sharp-crested rectangular weirs,” PhD Thesis, MSc thesis. Department of Civil Engineering, Middle East Technical . . . (2012).
- [98] S. TEKADE and A. VASUDEO, “Head discharge relationship of thin plated rectangular lab fabricated sharp crested weirs,” *Applied Fluid Mechanics*, **9**, 3, 1231 (2016).

## APPENDIX

---

---

## **APPENDIX A**

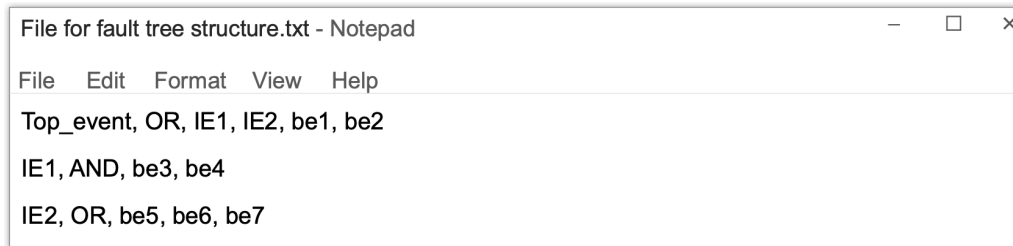
---

### **Fault Tree Files**

## A.1 Explanation for the input DATA format

### A.1.1 File for fault tree structure

Fault tree or event tree structure is described here with the format of CSV text files. In the CSV text file, component failures are logically combined through several gates to result in top event failure. These gates could be AND gate, NAND gate, NOR gate or OR gate. The intermediate events can have a dummy name (only one word used). The name for component/ basic event failure cannot be used as gates. Again, for the failure of component, one-word description is used. In the fault tree CSV input, each row refers to a sub-tree in which two levels (top-intermediate or intermediate-bottom) of failures are connected to each other through a logical gate (AND/OR/NAND/NOR). The first row of the input always refers to the top event failure and its dependence. The first element of each row denotes the top event of the sub-tree, the second element of a row denotes the logical gate (AND/OR/NAND/NOR) and the subsequent elements of a row denotes the second level events of a sub-tree. For example, figure



```
File for fault tree structure.txt - Notepad
File Edit Format View Help
Top_event, OR, IE1, IE2, be1, be2
IE1, AND, be3, be4
IE2, OR, be5, be6, be7
```

Figure A.1: Data format for Fault tree structure

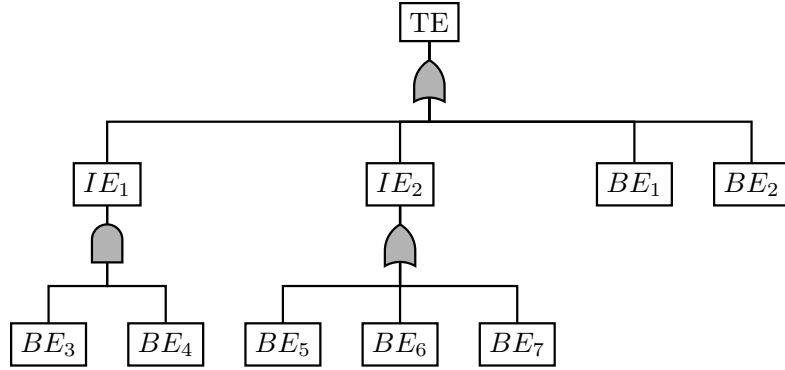


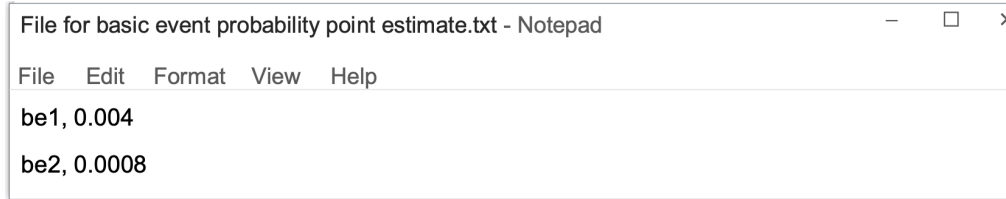
Figure A.2: Fault tree structure for the given Data format

In the first row, the *top\_event* is connected to two intermediate events *IE1* and *IE2* and two basic events *be1* and *be2*. The intermediate events for a fault tree are identified when they are again referred in the rows as a top event of another sub-tree, e.g., *IE1* and *IE2* are referred again in second and third row. The rows other than the first row can appear in any order as long as the intermediate event is referred in the rows mentioned above it.

### A.1.2 File for probability values

All the component failure probabilities are also described here with text files format. The failure probability related to each component is listed in a text box. The probability value is displayed for each of the basic event used in the fault tree structure. Component name is same as the one used in File for fault tree structures. The only probability type used in this dissertation is Point Estimate which is denoted by PE and written right after the basic event name. However, PE is not displayed in the files listed the appendix for the brevity. The following numeric value indicates the point estimate of the probability of failure. The subsequent value in the row denotes the point estimate of the probability

of the failure for the component.



```
File for basic event probability point estimate.txt - Notepad
File Edit Format View Help
be1, 0.004
be2, 0.0008
```

Figure A.3: Data format for Probabilities of the component failures

## A.2 Data used in section III.9

### A.2.1 Set 1 in Figure III.13

- Case 1

Fault Tree Structure

```
GATE0, OR, GATE1, A, B, C
GATE1, AND, GATE2, D, E, F, GATE3, G, H, I, J
GATE2, AND, GATE4, K
GATE3, AND, GATE5, L, M, N, O, P, Q, GATE6, R
GATE4, OR, GATE7, S
GATE5, OR, GATE8, T, U, V, W, X
GATE6, OR, GATE9, Y
GATE7, OR, Z, BA, BB, BC
GATE8, OR, BD, BE
GATE9, OR, BF, BG, BH
```

Probability of Component Failures

A,3e-05 B,9e-06 C,5e-07 D,8e-07 E,2e-07 F,3e-09 G,6e-09 H,5e-07  
I,9e-04 J,9e-04 K,6e-06 L,2e-08 M,7e-08 N,9e-07 O,2e-07 P,7e-05  
Q,9e-06 R,4e-09 S,6e-05 T,8e-07 U,5e-07 V,3e-09 W,3e-04 X,3e-07  
Y,3e-08 Z,7e-09 BA,4e-09 BB,7e-09 BC,5e-04 BD,4e-06 BE,8e-07  
BF,2e-07 BG,3e-09 BH,4e-10

- Case 2

Fault Tree Structure

GATE0, OR, GATE1, A, B, C  
GATE1, AND, GATE2, D, E, F, GATE3, G, H, I, J  
GATE2, AND, GATE4, K  
GATE3, AND, GATE5, L, M, N, O, P, Q, GATE6, R  
GATE4, AND, GATE7, S  
GATE5, OR, GATE8, T, U, V, W, X  
GATE6, AND, GATE9, Y  
GATE7, AND, Z, BA, BB, BC  
GATE8, AND, BD, BE  
GATE9, OR, BF, BG, BH

Probability of Component Failures

A,9e-07 B,1e-09 C,4e-08 D,3e-07 E,6e-09 F,7e-05 G,6e-05 H,3e-08  
I,8e-08 J,2e-04 K,8e-05 L,8e-09 M,6e-04 N,1e-06 O,3e-05 P,7e-05  
Q,2e-09 R,5e-09 S,4e-06 T,2e-07 U,8e-08 V,9e-05 W,8e-06 X,9e-06  
Y,1e-06 Z,8e-05 BA,4e-05 BB,5e-07 BC,4e-09 BD,1e-06 BE,8e-04  
BF,9e-05 BG,2e-05 BH,1e-04

- Case 3

Fault Tree Structure

GATE0, OR, GATE1, A, B, C  
GATE1, AND, GATE2, D, E, F, GATE3, G, H, I, J  
GATE2, OR, GATE4, K  
GATE3, OR, GATE5, L, M, N, O, P, Q, GATE6, R  
GATE4, OR, GATE7, S  
GATE5, AND, GATE8, T, U, V, W, X  
GATE6, AND, GATE9, Y  
GATE7, OR, Z, BA, BB, BC  
GATE8, AND, BD, BE  
GATE9, OR, BF, BG, BH

Probability of Component Failures

A,3e-05 B,7e-04 C,7e-08 D,7e-08 E,3e-07 F,5e-10 G,6e-07 H,5e-09  
I,7e-05 J,8e-09 K,4e-09 L,9e-07 M,3e-05 N,1e-04 O,4e-08 P,6e-06  
Q,2e-04 R,8e-06 S,4e-04 T,1e-07 U,6e-05 V,3e-04 W,9e-04 X,4e-04  
Y,7e-07 Z,7e-05 BA,3e-07 BB,6e-05 BC,2e-08 BD,3e-05 BE,1e-04  
BF,7e-08 BG,2e-06 BH,6e-08

- Case 4

Fault Tree Structure

GATE0, OR, GATE1, A, B, C  
GATE1, OR, GATE2, D, E, F, GATE3, G, H, I, J  
GATE2, OR, GATE4, K  
GATE3, OR, GATE5, L, M, N, O, P, Q, GATE6, R  
GATE4, AND, GATE7, S  
GATE5, AND, GATE8, T, U, V, W, X  
GATE6, AND, GATE9, Y  
GATE7, OR, Z, BA, BB, BC  
GATE8, AND, BD, BE  
GATE9, OR, BF, BG, BH

Probability of Component Failures

A,4e-07 B,8e-04 C,8e-09 D,4e-08 E,1e-05 F,3e-07 G,9e-09 H,9e-08  
I,1e-06 J,8e-08 K,5e-04 L,2e-08 M,3e-04 N,4e-06 O,7e-05 P,7e-06  
Q,8e-05 R,2e-05 S,6e-09 T,6e-07 U,2e-06 V,4e-06 W,7e-08 X,5e-04  
Y,1e-08 Z,3e-06 BA,8e-09 BB,8e-05 BC,5e-09 BD,8e-09 BE,1e-05  
BF,7e-08 BG,7e-05 BH,5e-09

- Case 5

Fault Tree Structure

GATE0, OR, GATE1, A, B, C  
GATE1, AND, GATE2, D, E, F, GATE3, G, H, I, J  
GATE2, AND, GATE4, K  
GATE3, OR, GATE5, L, M, N, O, P, Q, GATE6, R  
GATE4, AND, GATE7, S  
GATE5, OR, GATE8, T, U, V, W, X  
GATE6, AND, GATE9, Y  
GATE7, AND, Z, BA, BB, BC  
GATE8, OR, BD, BE  
GATE9, AND, BF, BG, BH

Probability of Component Failures

A,8e-09 B,1e-04 C,1e-05 D,7e-09 E,7e-09 F,9e-05 G,1e-06 H,7e-04  
I,4e-05 J,3e-08 K,6e-05 L,8e-06 M,9e-05 N,3e-09 O,2e-05 P,5e-09  
Q,9e-09 R,4e-06 S,4e-05 T,3e-09 U,1e-07 V,6e-06 W,1e-05 X,7e-04  
Y,2e-05 Z,9e-06 BA,1e-08 BB,6e-08 BC,2e-07 BD,3e-04 BE,6e-06  
BF,4e-06 BG,1e-08 BH,7e-07

- Case 6

Fault Tree Structure

GATE0, OR, GATE1, A, B, C  
GATE1, AND, GATE2, D, E, F, GATE3, G, H, I, J  
GATE2, OR, GATE4, K  
GATE3, AND, GATE5, L, M, N, O, P, Q, GATE6, R  
GATE4, AND, GATE7, S  
GATE5, AND, GATE8, T, U, V, W, X  
GATE6, OR, GATE9, Y  
GATE7, AND, Z, BA, BB, BC  
GATE8, AND, BD, BE  
GATE9, OR, BF, BG, BH

Probability of Component Failures

A,7e-07 B,3e-09 C,8e-08 D,7e-09 E,8e-06 F,1e-06 G,8e-08 H,8e-08  
I,5e-06 J,2e-08 K,1e-08 L,8e-05 M,1e-07 N,3e-06 O,2e-04 P,9e-08  
Q,8e-09 R,6e-07 S,6e-08 T,8e-06 U,6e-10 V,9e-07 W,2e-05 X,6e-07  
Y,7e-05 Z,2e-07 BA,6e-05 BB,4e-09 BC,1e-09 BD,2e-06 BE,2e-06  
BF,8e-07 BG,2e-05 BH,8e-05

- Case 7

Fault Tree Structure

GATE0, OR, GATE1, A, B, C  
GATE1, OR, GATE2, D, E, F, GATE3, G, H, I, J  
GATE2, AND, GATE4, K  
GATE3, OR, GATE5, L, M, N, O, P, Q, GATE6, R  
GATE4, OR, GATE7, S  
GATE5, OR, GATE8, T, U, V, W, X  
GATE6, OR, GATE9, Y  
GATE7, AND, Z, BA, BB, BC  
GATE8, AND, BD, BE  
GATE9, AND, BF, BG, BH

Probability of Component Failures

A,4e-06 B,9e-04 C,1e-08 D,1e-06 E,4e-05 F,9e-06 G,6e-08 H,8e-04  
I,7e-09 J,5e-08 K,3e-07 L,2e-06 M,3e-09 N,5e-07 O,4e-06 P,5e-09  
Q,6e-08 R,7e-05 S,5e-04 T,1e-07 U,2e-07 V,3e-04 W,4e-09 X,6e-06  
Y,9e-08 Z,8e-07 BA,8e-08 BB,8e-06 BC,6e-07 BD,5e-08 BE,2e-08  
BF,3e-07 BG,8e-08 BH,8e-06

- Case 8

Fault Tree Structure

GATE0, OR, GATE1, A, B, C  
GATE1, AND, GATE2, D, E, F, GATE3, G, H, I, J  
GATE2, OR, GATE4, K  
GATE3, OR, GATE5, L, M, N, O, P, Q, GATE6, R  
GATE4, OR, GATE7, S  
GATE5, AND, GATE8, T, U, V, W, X  
GATE6, OR, GATE9, Y  
GATE7, OR, Z, BA, BB, BC  
GATE8, AND, BD, BE  
GATE9, AND, BF, BG, BH

Probability of Component Failures

A,3e-06 B,3e-04 C,9e-06 D,5e-04 E,7e-06 F,5e-05 G,3e-08 H,6e-08  
I,9e-04 J,1e-05 K,4e-09 L,3e-04 M,7e-05 N,2e-09 O,4e-07 P,5e-09  
Q,9e-08 R,7e-09 S,4e-05 T,9e-07 U,3e-09 V,2e-08 W,9e-06 X,7e-08  
Y,7e-09 Z,6e-07 BA,2e-05 BB,7e-05 BC,3e-09 BD,7e-08 BE,3e-05  
BF,1e-03 BG,1e-06 BH,9e-08

- Case 9

Fault Tree Structure

GATE0, OR, GATE1, A, B, C  
GATE1, AND, GATE2, D, E, F, GATE3, G, H, I, J  
GATE2, AND, GATE4, K  
GATE3, AND, GATE5, L, M, N, O, P, Q, GATE6, R  
GATE4, AND, GATE7, S  
GATE5, OR, GATE8, T, U, V, W, X  
GATE6, OR, GATE9, Y  
GATE7, OR, Z, BA, BB, BC  
GATE8, AND, BD, BE  
GATE9, AND, BF, BG, BH

Probability of Component Failures

A,5e-04 B,1e-07 C,5e-05 D,7e-05 E,7e-08 F,7e-09 G,8e-05 H,5e-06  
I,9e-09 J,6e-05 K,8e-08 L,6e-04 M,2e-07 N,9e-05 O,7e-09 P,7e-09  
Q,9e-08 R,6e-09 S,4e-04 T,4e-04 U,1e-07 V,9e-04 W,6e-08 X,1e-09  
Y,6e-04 Z,3e-06 BA,6e-10 BB,9e-05 BC,6e-05 BD,4e-06 BE,6e-04  
BF,4e-04 BG,5e-05 BH,1e-06

- Case 10

Fault Tree Structure

GATE0, OR, GATE1, A, B, C  
GATE1, OR, GATE2, D, E, F, GATE3, G, H, I, J  
GATE2, AND, GATE4, K  
GATE3, AND, GATE5, L, M, N, O, P, Q, GATE6, R  
GATE4, AND, GATE7, S  
GATE5, AND, GATE8, T, U, V, W, X  
GATE6, OR, GATE9, Y  
GATE7, OR, Z, BA, BB, BC  
GATE8, OR, BD, BE  
GATE9, OR, BF, BG, BH

Probability of Component Failures

A,3e-04 B,1e-08 C,7e-08 D,2e-09 E,8e-08 F,8e-04 G,7e-07 H,2e-07  
 I,6e-07 J,4e-09 K,7e-09 L,7e-08 M,5e-08 N,3e-06 O,1e-06 P,5e-05  
 Q,7e-05 R,5e-05 S,6e-05 T,3e-06 U,4e-09 V,3e-08 W,3e-09 X,1e-04  
 Y,2e-08 Z,2e-05 BA,3e-06 BB,6e-07 BC,9e-04 BD,4e-05 BE,6e-09  
 BF,6e-06 BG,3e-04 BH,9e-08

### A.2.2 Set 2 in Figure III.13

- Case 1

Fault Tree Structure

GATE0, OR, GATE1, A, B, GATE2, C  
 GATE1, OR, GATE3, D, E, F, G, H  
 GATE2, OR, GATE4, I, J  
 GATE3, OR, GATE5, K, L, M, N, O, P, GATE6, Q, R, S, T  
 GATE4, AND, GATE7, U, V, W, X, Y, Z  
 GATE5, AND, GATE8, BA, BB, BC, BD, BE, BF, BG, BH  
 GATE6, OR, GATE9, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS  
 GATE7, AND, GATE10, GATE11, BT, BU, BV, BW, BX, BY, BZ, CA,  
 GATE12  
 GATE8, AND, GATE13, GATE14, CB, CC, CD, CE, CF, CG, CH, CI, CJ,  
 CK, CL, CM, CN, CO  
 GATE9, AND, CP, CQ, CR, CS  
 GATE10, AND, CT, CU, CV, CW, CX, CY, CZ, DA

GATE11, OR, DB, DC, DD

GATE12, OR, DE, DF, DG, DH, DI, DJ, DK, DL, DM, DN, DO

GATE13, AND, DP, DQ, DR, DS, DT, DU, DV, DW, DX, DY, DZ, EA, EB,  
EC

GATE14, AND, ED, EE, EF, EG

### Probability of Component Failures

A,1e-06 B,3e-06 C,7e-06 D,6e-08 E,6e-08 F,6e-11 G,3e-05 H,2e-09  
I,5e-09 J,5e-06 K,1e-04 L,2e-05 M,8e-09 N,9e-05 O,9e-09 P,9e-08  
Q,7e-04 R,2e-09 S,3e-09 T,7e-09 U,4e-07 V,2e-04 W,8e-09 X,1e-07  
Y,7e-04 Z,4e-08 BA,9e-04 BB,5e-08 BC,6e-04 BD,3e-07 BE,6e-07  
BF,9e-07 BG,9e-05 BH,6e-09 BI,2e-04 BJ,9e-04 BK,6e-05 BL,8e-05  
BM,6e-07 BN,3e-07 BO,7e-04 BP,9e-09 BQ,5e-07 BR,9e-04 BS,6e-10  
BT,7e-07 BU,2e-04 BV,8e-04 BW,6e-09 BX,3e-07 BY,3e-04 BZ,9e-06  
CA,1e-06 CB,8e-04 CC,2e-04 CD,9e-06 CE,6e-08 CF,4e-08 CG,1e-06  
CH,6e-05 CI,7e-09 CJ,9e-08 CK,2e-04 CL,6e-07 CM,4e-07 CN,9e-07  
CO,5e-08 CP,6e-06 CQ,2e-07 CR,4e-05 CS,5e-09 CT,7e-05 CU,7e-09  
CV,2e-06 CW,9e-08 CX,2e-05 CY,3e-05 CZ,8e-09 DA,9e-05 DB,5e-05  
DC,6e-04 DD,2e-08 DE,6e-04 DF,6e-08 DG,7e-04 DH,1e-06 DI,2e-04  
DJ,7e-09 DK,7e-08 DL,7e-07 DM,4e-06 DN,4e-07 DO,2e-09 DP,8e-04  
DQ,4e-06 DR,3e-08 DS,6e-08 DT,7e-08 DU,3e-07 DV,9e-07 DW,3e-06  
DX,6e-05 DY,5e-08 DZ,9e-05 EA,4e-06 EB,8e-06 EC,9e-08 ED,8e-05  
EE,2e-04 EF,3e-05 EG,7e-09

- Case 2

Fault Tree Structure

GATE0, OR, GATE1, A, B, GATE2, C

GATE1, AND, GATE3, D, E, F, G, H

GATE2, OR, GATE4, I, J

GATE3, OR, GATE5, K, L, M, N, O, P, GATE6, Q, R, S, T

GATE4, OR, GATE7, U, V, W, X, Y, Z

GATE5, AND, GATE8, BA, BB, BC, BD, BE, BF, BG, BH

GATE6, OR, GATE9, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS

GATE7, OR, GATE10, GATE11, BT, BU, BV, BW, BX, BY, BZ, CA,  
GATE12

GATE8, OR, GATE13, GATE14, CB, CC, CD, CE, CF, CG, CH, CI, CJ,  
CK, CL, CM, CN, CO

GATE9, AND, CP, CQ, CR, CS

GATE10, AND, CT, CU, CV, CW, CX, CY, CZ, DA

GATE11, AND, DB, DC, DD

GATE12, OR, DE, DF, DG, DH, DI, DJ, DK, DL, DM, DN, DO

GATE13, OR, DP, DQ, DR, DS, DT, DU, DV, DW, DX, DY, DZ, EA, EB,  
EC

GATE14, OR, ED, EE, EF, EG

Probability of Component Failures

A,9e-07 B,9e-04 C,9e-09 D,2e-06 E,6e-04 F,9e-04 G,8e-08 H,2e-09  
 I,8e-05 J,1e-06 K,7e-04 L,5e-06 M,5e-06 N,6e-10 O,6e-05 P,6e-09  
 Q,7e-08 R,3e-07 S,9e-04 T,5e-08 U,1e-07 V,6e-06 W,1e-04 X,4e-07  
 Y,3e-08 Z,2e-07 BA,1e-04 BB,1e-08 BC,8e-09 BD,2e-04 BE,6e-05  
 BF,1e-06 BG,4e-07 BH,7e-04 BI,7e-04 BJ,3e-06 BK,4e-05 BL,1e-07  
 BM,1e-07 BN,1e-08 BO,3e-06 BP,7e-09 BQ,5e-06 BR,2e-08 BS,9e-07  
 BT,2e-04 BU,8e-08 BV,8e-05 BW,4e-09 BX,6e-05 BY,9e-07 BZ,5e-05  
 CA,6e-08 CB,8e-08 CC,2e-10 CD,5e-08 CE,7e-08 CF,7e-08 CG,7e-09  
 CH,6e-04 CI,7e-04 CJ,8e-04 CK,3e-07 CL,4e-07 CM,9e-09 CN,4e-11  
 CO,1e-07 CP,1e-06 CQ,8e-09 CR,8e-06 CS,2e-07 CT,5e-09 CU,7e-04  
 CV,8e-10 CW,5e-09 CX,4e-05 CY,5e-07 CZ,3e-04 DA,1e-04 DB,9e-09  
 DC,9e-05 DD,5e-09 DE,4e-08 DF,5e-05 DG,8e-08 DH,2e-07 DI,2e-06  
 DJ,5e-07 DK,4e-06 DL,8e-07 DM,8e-05 DN,5e-07 DO,8e-08 DP,7e-04  
 DQ,8e-07 DR,2e-07 DS,7e-09 DT,5e-06 DU,3e-05 DV,5e-08 DW,3e-08  
 DX,2e-05 DY,8e-09 DZ,6e-05 EA,9e-09 EB,3e-04 EC,6e-04 ED,6e-07  
 EE,8e-05 EF,1e-06 EG,4e-04

- Case 3

Fault Tree Structure

GATE0, OR, GATE1, A, B, GATE2, C  
 GATE1, AND, GATE3, D, E, F, G, H  
 GATE2, OR, GATE4, I, J  
 GATE3, OR, GATE5, K, L, M, N, O, P, GATE6, Q, R, S, T

GATE4, OR, GATE7, U, V, W, X, Y, Z  
 GATE5, AND, GATE8, BA, BB, BC, BD, BE, BF, BG, BH  
 GATE6, AND, GATE9, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS  
 GATE7, OR, GATE10, GATE11, BT, BU, BV, BW, BX, BY, BZ, CA,  
 GATE12  
 GATE8, AND, GATE13, GATE14, CB, CC, CD, CE, CF, CG, CH, CI, CJ,  
 CK, CL, CM, CN, CO  
 GATE9, OR, CP, CQ, CR, CS  
 GATE10, OR, CT, CU, CV, CW, CX, CY, CZ, DA  
 GATE11, OR, DB, DC, DD  
 GATE12, AND, DE, DF, DG, DH, DI, DJ, DK, DL, DM, DN, DO  
 GATE13, OR, DP, DQ, DR, DS, DT, DU, DV, DW, DX, DY, DZ, EA, EB,  
 EC  
 GATE14, OR, ED, EE, EF, EG

#### Probability of Component Failures

A,5e-05 B,9e-07 C,2e-09 D,2e-06 E,2e-05 F,4e-04 G,9e-08 H,1e-06  
 I,6e-06 J,6e-05 K,4e-09 L,2e-07 M,1e-05 N,8e-07 O,5e-08 P,6e-04  
 Q,2e-08 R,8e-08 S,1e-09 T,5e-09 U,2e-05 V,1e-05 W,2e-06 X,2e-04  
 Y,8e-06 Z,3e-05 BA,3e-06 BB,4e-05 BC,9e-09 BD,3e-04 BE,8e-07  
 BF,4e-09 BG,2e-04 BH,8e-09 BI,5e-08 BJ,7e-06 BK,9e-07 BL,1e-05  
 BM,7e-05 BN,1e-03 BO,6e-05 BP,8e-07 BQ,5e-05 BR,6e-05 BS,2e-05  
 BT,2e-06 BU,2e-08 BV,9e-07 BW,1e-03 BX,7e-09 BY,1e-03 BZ,7e-05

CA,9e-09 CB,4e-07 CC,2e-07 CD,3e-07 CE,7e-08 CF,8e-07 CG,6e-07  
 CH,2e-07 CI,1e-03 CJ,3e-05 CK,1e-08 CL,9e-07 CM,1e-09 CN,4e-08  
 CO,4e-09 CP,2e-06 CQ,9e-04 CR,5e-04 CS,5e-08 CT,4e-05 CU,6e-09  
 CV,8e-06 CW,1e-08 CX,8e-09 CY,3e-05 CZ,1e-04 DA,2e-04 DB,8e-07  
 DC,6e-05 DD,2e-06 DE,7e-07 DF,5e-07 DG,1e-03 DH,2e-09 DI,7e-04  
 DJ,8e-09 DK,9e-08 DL,5e-08 DM,8e-06 DN,7e-06 DO,9e-08 DP,4e-08  
 DQ,9e-09 DR,6e-07 DS,6e-06 DT,3e-05 DU,3e-04 DV,8e-08 DW,1e-09  
 DX,5e-06 DY,9e-09 DZ,2e-05 EA,1e-06 EB,9e-09 EC,5e-07 ED,9e-04  
 EE,2e-06 EF,4e-04 EG,1e-05

- Case 4

Fault Tree Structure

GATE0, OR, GATE1, A, B, GATE2, C  
 GATE1, OR, GATE3, D, E, F, G, H  
 GATE2, AND, GATE4, I, J  
 GATE3, AND, GATE5, K, L, M, N, O, P, GATE6, Q, R, S, T  
 GATE4, AND, GATE7, U, V, W, X, Y, Z  
 GATE5, AND, GATE8, BA, BB, BC, BD, BE, BF, BG, BH  
 GATE6, OR, GATE9, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS  
 GATE7, OR, GATE10, GATE11, BT, BU, BV, BW, BX, BY, BZ, CA,  
 GATE12  
 GATE8, AND, GATE13, GATE14, CB, CC, CD, CE, CF, CG, CH, CI, CJ,  
 CK, CL, CM, CN, CO

GATE9, AND, CP, CQ, CR, CS

GATE10, AND, CT, CU, CV, CW, CX, CY, CZ, DA

GATE11, OR, DB, DC, DD

GATE12, OR, DE, DF, DG, DH, DI, DJ, DK, DL, DM, DN, DO

GATE13, AND, DP, DQ, DR, DS, DT, DU, DV, DW, DX, DY, DZ, EA, EB,  
EC

GATE14, OR, ED, EE, EF, EG

### Probability of Component Failures

A,8e-04 B,9e-05 C,2e-09 D,1e-05 E,6e-06 F,4e-04 G,7e-04 H,2e-08  
I,3e-05 J,2e-07 K,3e-05 L,1e-08 M,6e-06 N,7e-08 O,7e-04 P,6e-04  
Q,3e-09 R,8e-10 S,4e-09 T,5e-05 U,8e-08 V,4e-05 W,1e-06 X,5e-06  
Y,2e-05 Z,9e-08 BA,4e-04 BB,9e-09 BC,7e-06 BD,2e-06 BE,7e-04  
BF,6e-09 BG,9e-05 BH,5e-06 BI,8e-05 BJ,3e-09 BK,7e-05 BL,5e-04  
BM,3e-06 BN,5e-06 BO,8e-06 BP,1e-09 BQ,4e-05 BR,5e-08 BS,1e-07  
BT,1e-05 BU,3e-07 BV,4e-04 BW,4e-08 BX,1e-03 BY,7e-08 BZ,8e-06  
CA,5e-09 CB,4e-08 CC,4e-04 CD,6e-08 CE,7e-04 CF,4e-08 CG,1e-05  
CH,3e-09 CI,4e-06 CJ,6e-04 CK,1e-07 CL,2e-07 CM,3e-08 CN,4e-05  
CO,5e-06 CP,7e-07 CQ,8e-05 CR,7e-04 CS,8e-07 CT,7e-06 CU,2e-05  
CV,5e-07 CW,8e-07 CX,1e-09 CY,7e-07 CZ,1e-08 DA,1e-08 DB,3e-05  
DC,9e-08 DD,1e-06 DE,9e-05 DF,2e-08 DG,4e-06 DH,5e-07 DI,9e-05  
DJ,1e-04 DK,3e-09 DL,8e-06 DM,7e-05 DN,7e-08 DO,7e-07 DP,6e-08  
DQ,5e-08 DR,7e-05 DS,5e-07 DT,2e-05 DU,6e-08 DV,1e-08 DW,6e-05

DX,2e-04 DY,2e-04 DZ,2e-08 EA,7e-04 EB,3e-09 EC,4e-06 ED,1e-06  
EE,8e-09 EF,7e-05 EG,1e-07

- Case 5

Fault Tree Structure

GATE0, OR, GATE1, A, B, GATE2, C  
GATE1, OR, GATE3, D, E, F, G, H  
GATE2, OR, GATE4, I, J  
GATE3, OR, GATE5, K, L, M, N, O, P, GATE6, Q, R, S, T  
GATE4, OR, GATE7, U, V, W, X, Y, Z  
GATE5, AND, GATE8, BA, BB, BC, BD, BE, BF, BG, BH  
GATE6, OR, GATE9, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS  
GATE7, OR, GATE10, GATE11, BT, BU, BV, BW, BX, BY, BZ, CA,  
GATE12  
GATE8, OR, GATE13, GATE14, CB, CC, CD, CE, CF, CG, CH, CI, CJ,  
CK, CL, CM, CN, CO  
GATE9, AND, CP, CQ, CR, CS  
GATE10, AND, CT, CU, CV, CW, CX, CY, CZ, DA  
GATE11, AND, DB, DC, DD  
GATE12, AND, DE, DF, DG, DH, DI, DJ, DK, DL, DM, DN, DO  
GATE13, AND, DP, DQ, DR, DS, DT, DU, DV, DW, DX, DY, DZ, EA, EB,  
EC  
GATE14, AND, ED, EE, EF, EG

## Probability of Component Failures

A,7e-10 B,1e-08 C,3e-07 D,6e-07 E,9e-09 F,2e-06 G,3e-06 H,6e-07  
I,9e-04 J,4e-07 K,9e-06 L,7e-04 M,9e-05 N,7e-09 O,2e-05 P,7e-07  
Q,3e-08 R,5e-04 S,2e-05 T,9e-09 U,6e-05 V,8e-06 W,5e-06 X,5e-10  
Y,1e-09 Z,7e-07 BA,5e-09 BB,5e-04 BC,2e-07 BD,6e-06 BE,1e-08  
BF,1e-08 BG,4e-04 BH,8e-05 BI,2e-08 BJ,2e-04 BK,2e-06 BL,4e-06  
BM,7e-05 BN,3e-09 BO,8e-04 BP,5e-05 BQ,8e-09 BR,1e-04 BS,1e-06  
BT,2e-04 BU,3e-07 BV,9e-09 BW,8e-06 BX,6e-07 BY,4e-09 BZ,6e-04  
CA,1e-08 CB,9e-05 CC,6e-07 CD,3e-06 CE,8e-06 CF,9e-04 CG,3e-08  
CH,8e-08 CI,4e-08 CJ,2e-06 CK,8e-05 CL,1e-07 CM,9e-05 CN,8e-05  
CO,6e-06 CP,1e-08 CQ,8e-06 CR,6e-06 CS,9e-07 CT,1e-03 CU,6e-05  
CV,4e-07 CW,4e-06 CX,6e-05 CY,4e-09 CZ,2e-06 DA,5e-04 DB,2e-06  
DC,4e-08 DD,9e-09 DE,9e-04 DF,4e-09 DG,1e-06 DH,9e-07 DI,1e-07  
DJ,9e-04 DK,2e-06 DL,9e-04 DM,1e-08 DN,6e-08 DO,7e-07 DP,3e-08  
DQ,1e-05 DR,7e-09 DS,3e-06 DT,4e-07 DU,7e-04 DV,6e-09 DW,1e-05  
DX,7e-07 DY,6e-04 DZ,8e-06 EA,9e-05 EB,2e-04 EC,1e-03 ED,3e-05  
EE,2e-06 EF,8e-09 EG,5e-05

- Case 6

### Fault Tree Structure

GATE0, OR, GATE1, A, B, GATE2, C  
GATE1, AND, GATE3, D, E, F, G, H

GATE2, AND, GATE4, I, J  
 GATE3, AND, GATE5, K, L, M, N, O, P, GATE6, Q, R, S, T  
 GATE4, OR, GATE7, U, V, W, X, Y, Z  
 GATE5, AND, GATE8, BA, BB, BC, BD, BE, BF, BG, BH  
 GATE6, OR, GATE9, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS  
 GATE7, OR, GATE10, GATE11, BT, BU, BV, BW, BX, BY, BZ, CA,  
 GATE12  
 GATE8, OR, GATE13, GATE14, CB, CC, CD, CE, CF, CG, CH, CI, CJ,  
 CK, CL, CM, CN, CO  
 GATE9, OR, CP, CQ, CR, CS  
 GATE10, AND, CT, CU, CV, CW, CX, CY, CZ, DA  
 GATE11, OR, DB, DC, DD  
 GATE12, OR, DE, DF, DG, DH, DI, DJ, DK, DL, DM, DN, DO  
 GATE13, OR, DP, DQ, DR, DS, DT, DU, DV, DW, DX, DY, DZ, EA, EB,  
 EC  
 GATE14, AND, ED, EE, EF, EG

#### Probability of Component Failures

A,3e-04 B,5e-07 C,2e-04 D,9e-06 E,9e-04 F,9e-06 G,9e-09 H,5e-06  
 I,6e-05 J,2e-05 K,2e-09 L,4e-06 M,8e-07 N,7e-08 O,4e-04 P,8e-05  
 Q,3e-09 R,1e-06 S,8e-04 T,1e-06 U,8e-08 V,8e-08 W,3e-06 X,9e-09  
 Y,5e-05 Z,1e-05 BA,8e-09 BB,5e-07 BC,6e-05 BD,8e-08 BE,8e-07  
 BF,5e-06 BG,3e-04 BH,5e-04 BI,6e-07 BJ,6e-05 BK,2e-09 BL,1e-04

BM,6e-08 BN,5e-07 BO,5e-04 BP,5e-08 BQ,3e-06 BR,9e-10 BS,3e-06  
 BT,1e-06 BU,7e-09 BV,9e-08 BW,6e-06 BX,2e-07 BY,5e-07 BZ,8e-07  
 CA,6e-09 CB,1e-07 CC,2e-06 CD,5e-06 CE,3e-04 CF,2e-09 CG,6e-06  
 CH,1e-07 CI,3e-06 CJ,7e-08 CK,1e-04 CL,8e-04 CM,8e-07 CN,5e-09  
 CO,2e-09 CP,4e-05 CQ,2e-07 CR,1e-03 CS,3e-05 CT,8e-06 CU,8e-08  
 CV,3e-06 CW,2e-05 CX,2e-06 CY,2e-08 CZ,4e-07 DA,8e-06 DB,3e-04  
 DC,1e-05 DD,1e-03 DE,6e-07 DF,8e-06 DG,9e-06 DH,9e-04 DI,5e-08  
 DJ,2e-05 DK,9e-04 DL,7e-04 DM,6e-04 DN,1e-08 DO,7e-04 DP,4e-05  
 DQ,2e-07 DR,1e-05 DS,6e-06 DT,2e-08 DU,2e-05 DV,8e-07 DW,5e-08  
 DX,3e-09 DY,8e-09 DZ,2e-04 EA,6e-04 EB,9e-09 EC,3e-07 ED,4e-09  
 EE,4e-06 EF,8e-10 EG,4e-06

- Case 7

Fault Tree Structure

GATE0, OR, GATE1, A, B, GATE2, C  
 GATE1, AND, GATE3, D, E, F, G, H  
 GATE2, OR, GATE4, I, J  
 GATE3, AND, GATE5, K, L, M, N, O, P, GATE6, Q, R, S, T  
 GATE4, AND, GATE7, U, V, W, X, Y, Z  
 GATE5, OR, GATE8, BA, BB, BC, BD, BE, BF, BG, BH  
 GATE6, OR, GATE9, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS  
 GATE7, OR, GATE10, GATE11, BT, BU, BV, BW, BX, BY, BZ, CA,  
 GATE12

GATE8, OR, GATE13, GATE14, CB, CC, CD, CE, CF, CG, CH, CI, CJ,  
CK, CL, CM, CN, CO

GATE9, OR, CP, CQ, CR, CS

GATE10, AND, CT, CU, CV, CW, CX, CY, CZ, DA

GATE11, AND, DB, DC, DD

GATE12, AND, DE, DF, DG, DH, DI, DJ, DK, DL, DM, DN, DO

GATE13, AND, DP, DQ, DR, DS, DT, DU, DV, DW, DX, DY, DZ, EA, EB,  
EC

GATE14, AND, ED, EE, EF, EG

#### Probability of Component Failures

A,6e-09 B,5e-05 C,7e-04 D,7e-05 E,8e-06 F,6e-06 G,8e-05 H,7e-04  
I,2e-09 J,7e-08 K,4e-08 L,2e-04 M,6e-09 N,9e-08 O,4e-08 P,8e-07  
Q,4e-05 R,2e-04 S,6e-06 T,6e-08 U,2e-08 V,5e-06 W,3e-07 X,3e-07  
Y,6e-06 Z,9e-05 BA,7e-05 BB,9e-06 BC,1e-05 BD,2e-08 BE,7e-05  
BF,4e-08 BG,9e-06 BH,1e-06 BI,1e-06 BJ,1e-08 BK,7e-05 BL,9e-09  
BM,8e-09 BN,4e-04 BO,4e-07 BP,7e-07 BQ,2e-05 BR,3e-07 BS,1e-04  
BT,8e-07 BU,5e-04 BV,4e-08 BW,1e-05 BX,9e-05 BY,4e-04 BZ,4e-06  
CA,7e-07 CB,2e-07 CC,1e-06 CD,4e-09 CE,8e-04 CF,2e-06 CG,5e-07  
CH,1e-06 CI,2e-06 CJ,7e-06 CK,3e-08 CL,1e-05 CM,9e-06 CN,9e-06  
CO,7e-07 CP,3e-09 CQ,2e-09 CR,1e-04 CS,8e-11 CT,9e-05 CU,3e-05  
CV,7e-06 CW,4e-04 CX,9e-05 CY,1e-09 CZ,7e-09 DA,2e-08 DB,7e-07  
DC,9e-05 DD,8e-04 DE,5e-07 DF,8e-07 DG,2e-04 DH,9e-05 DI,1e-06

DJ,5e-08 DK,1e-03 DL,1e-07 DM,6e-08 DN,5e-05 DO,5e-08 DP,9e-08  
DQ,4e-06 DR,7e-04 DS,9e-09 DT,5e-07 DU,3e-10 DV,2e-09 DW,5e-08  
DX,8e-07 DY,6e-08 DZ,9e-08 EA,6e-05 EB,8e-08 EC,6e-08 ED,4e-05  
EE,8e-09 EF,3e-06 EG,7e-10

- Case 8

Fault Tree Structure

GATE0, OR, GATE1, A, B, GATE2, C  
GATE1, OR, GATE3, D, E, F, G, H  
GATE2, AND, GATE4, I, J  
GATE3, OR, GATE5, K, L, M, N, O, P, GATE6, Q, R, S, T  
GATE4, OR, GATE7, U, V, W, X, Y, Z  
GATE5, OR, GATE8, BA, BB, BC, BD, BE, BF, BG, BH  
GATE6, OR, GATE9, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS  
GATE7, AND, GATE10, GATE11, BT, BU, BV, BW, BX, BY, BZ, CA,  
GATE12  
GATE8, AND, GATE13, GATE14, CB, CC, CD, CE, CF, CG, CH, CI, CJ,  
CK, CL, CM, CN, CO  
GATE9, AND, CP, CQ, CR, CS  
GATE10, OR, CT, CU, CV, CW, CX, CY, CZ, DA  
GATE11, AND, DB, DC, DD  
GATE12, AND, DE, DF, DG, DH, DI, DJ, DK, DL, DM, DN, DO  
GATE13, OR, DP, DQ, DR, DS, DT, DU, DV, DW, DX, DY, DZ, EA, EB,

EC

GATE14, OR, ED, EE, EF, EG

### Probability of Component Failures

A,4e-07 B,1e-04 C,9e-09 D,5e-09 E,6e-09 F,4e-09 G,6e-04 H,2e-09  
I,1e-08 J,7e-05 K,4e-07 L,4e-05 M,9e-08 N,8e-04 O,1e-04 P,4e-05  
Q,1e-08 R,6e-04 S,8e-05 T,3e-06 U,4e-06 V,4e-09 W,3e-09 X,5e-09  
Y,6e-06 Z,9e-10 BA,8e-05 BB,5e-07 BC,5e-04 BD,4e-08 BE,6e-08  
BF,3e-09 BG,6e-09 BH,6e-07 BI,7e-04 BJ,4e-08 BK,7e-05 BL,5e-05  
BM,5e-07 BN,2e-07 BO,7e-04 BP,3e-08 BQ,1e-05 BR,1e-08 BS,7e-08  
BT,6e-09 BU,9e-06 BV,3e-06 BW,8e-08 BX,3e-09 BY,6e-08 BZ,4e-06  
CA,9e-06 CB,3e-06 CC,3e-09 CD,1e-06 CE,7e-04 CF,5e-08 CG,9e-04  
CH,7e-06 CI,2e-04 CJ,9e-04 CK,1e-08 CL,4e-04 CM,1e-06 CN,7e-08  
CO,3e-07 CP,2e-07 CQ,3e-05 CR,5e-06 CS,9e-04 CT,9e-09 CU,6e-05  
CV,2e-06 CW,6e-05 CX,2e-08 CY,8e-09 CZ,9e-05 DA,9e-05 DB,2e-07  
DC,1e-04 DD,5e-05 DE,9e-09 DF,8e-07 DG,1e-05 DH,2e-05 DI,3e-07  
DJ,5e-09 DK,7e-06 DL,3e-05 DM,7e-09 DN,4e-04 DO,6e-04 DP,4e-07  
DQ,1e-03 DR,7e-05 DS,4e-08 DT,5e-08 DU,2e-09 DV,9e-04 DW,1e-04  
DX,1e-05 DY,9e-08 DZ,7e-08 EA,7e-05 EB,7e-07 EC,2e-06 ED,1e-07  
EE,4e-06 EF,6e-06 EG,9e-07

- Case 9

## Fault Tree Structure

GATE0, OR, GATE1, A, B, GATE2, C

GATE1, AND, GATE3, D, E, F, G, H

GATE2, AND, GATE4, I, J

GATE3, OR, GATE5, K, L, M, N, O, P, GATE6, Q, R, S, T

GATE4, OR, GATE7, U, V, W, X, Y, Z

GATE5, OR, GATE8, BA, BB, BC, BD, BE, BF, BG, BH

GATE6, AND, GATE9, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS

GATE7, AND, GATE10, GATE11, BT, BU, BV, BW, BX, BY, BZ, CA,  
GATE12

GATE8, OR, GATE13, GATE14, CB, CC, CD, CE, CF, CG, CH, CI, CJ,  
CK, CL, CM, CN, CO

GATE9, AND, CP, CQ, CR, CS

GATE10, OR, CT, CU, CV, CW, CX, CY, CZ, DA

GATE11, AND, DB, DC, DD

GATE12, OR, DE, DF, DG, DH, DI, DJ, DK, DL, DM, DN, DO

GATE13, OR, DP, DQ, DR, DS, DT, DU, DV, DW, DX, DY, DZ, EA, EB,  
EC

GATE14, OR, ED, EE, EF, EG

## Probability of Component Failures

A,9e-05 B,4e-05 C,7e-08 D,6e-05 E,4e-04 F,8e-09 G,8e-05 H,1e-07  
 I,1e-06 J,8e-07 K,2e-08 L,8e-07 M,5e-04 N,9e-07 O,9e-07 P,1e-07  
 Q,5e-07 R,5e-07 S,2e-09 T,7e-06 U,7e-05 V,2e-09 W,7e-06 X,3e-05  
 Y,8e-06 Z,2e-09 BA,5e-08 BB,2e-08 BC,3e-07 BD,5e-06 BE,4e-06  
 BF,8e-09 BG,5e-07 BH,9e-05 BI,4e-08 BJ,4e-08 BK,9e-08 BL,6e-08  
 BM,4e-08 BN,5e-04 BO,4e-09 BP,9e-06 BQ,5e-06 BR,2e-07 BS,3e-04  
 BT,3e-09 BU,5e-06 BV,6e-04 BW,7e-08 BX,7e-08 BY,4e-07 BZ,6e-06  
 CA,8e-07 CB,5e-09 CC,2e-04 CD,5e-05 CE,2e-09 CF,4e-05 CG,8e-04  
 CH,8e-05 CI,2e-05 CJ,7e-04 CK,2e-09 CL,5e-05 CM,8e-04 CN,5e-08  
 CO,9e-08 CP,7e-06 CQ,7e-07 CR,6e-08 CS,1e-07 CT,6e-04 CU,1e-04  
 CV,4e-04 CW,1e-03 CX,4e-05 CY,4e-04 CZ,5e-04 DA,4e-05 DB,6e-05  
 DC,1e-08 DD,5e-06 DE,7e-09 DF,1e-05 DG,1e-06 DH,4e-06 DI,3e-04  
 DJ,2e-08 DK,4e-05 DL,9e-04 DM,2e-06 DN,2e-07 DO,7e-06 DP,3e-08  
 DQ,4e-07 DR,3e-04 DS,6e-07 DT,9e-05 DU,2e-09 DV,6e-07 DW,8e-04  
 DX,6e-06 DY,9e-09 DZ,4e-06 EA,6e-05 EB,3e-04 EC,4e-08 ED,6e-05  
 EE,5e-06 EF,2e-08 EG,1e-09

- Case 10

Fault Tree Structure

GATE0, OR, GATE1, A, B, GATE2, C  
 GATE1, AND, GATE3, D, E, F, G, H  
 GATE2, OR, GATE4, I, J  
 GATE3, OR, GATE5, K, L, M, N, O, P, GATE6, Q, R, S, T

GATE4, AND, GATE7, U, V, W, X, Y, Z  
 GATE5, AND, GATE8, BA, BB, BC, BD, BE, BF, BG, BH  
 GATE6, OR, GATE9, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS  
 GATE7, AND, GATE10, GATE11, BT, BU, BV, BW, BX, BY, BZ, CA,  
 GATE12  
 GATE8, OR, GATE13, GATE14, CB, CC, CD, CE, CF, CG, CH, CI, CJ,  
 CK, CL, CM, CN, CO  
 GATE9, OR, CP, CQ, CR, CS  
 GATE10, AND, CT, CU, CV, CW, CX, CY, CZ, DA  
 GATE11, OR, DB, DC, DD  
 GATE12, OR, DE, DF, DG, DH, DI, DJ, DK, DL, DM, DN, DO  
 GATE13, OR, DP, DQ, DR, DS, DT, DU, DV, DW, DX, DY, DZ, EA, EB,  
 EC  
 GATE14, AND, ED, EE, EF, EG

#### Probability of Component Failures

A,5e-09 B,2e-09 C,5e-05 D,5e-06 E,1e-07 F,2e-05 G,4e-06 H,5e-04  
 I,2e-06 J,9e-06 K,8e-08 L,9e-08 M,2e-08 N,2e-08 O,1e-07 P,9e-07  
 Q,5e-06 R,2e-04 S,8e-09 T,7e-07 U,9e-06 V,7e-07 W,7e-05 X,3e-08  
 Y,2e-07 Z,2e-09 BA,7e-07 BB,5e-09 BC,6e-06 BD,7e-05 BE,9e-10  
 BF,1e-07 BG,6e-07 BH,3e-06 BI,3e-05 BJ,3e-08 BK,5e-04 BL,8e-04  
 BM,8e-06 BN,2e-08 BO,3e-05 BP,9e-09 BQ,6e-08 BR,3e-09 BS,9e-06  
 BT,3e-08 BU,5e-05 BV,7e-08 BW,2e-05 BX,2e-08 BY,2e-05 BZ,8e-06

CA,5e-09 CB,8e-09 CC,5e-04 CD,2e-05 CE,3e-04 CF,9e-05 CG,5e-07  
 CH,6e-06 CI,4e-07 CJ,8e-08 CK,4e-07 CL,9e-07 CM,5e-04 CN,5e-05  
 CO,7e-04 CP,7e-05 CQ,6e-06 CR,5e-05 CS,4e-07 CT,1e-06 CU,1e-04  
 CV,3e-05 CW,9e-06 CX,1e-06 CY,2e-08 CZ,1e-04 DA,3e-06 DB,6e-09  
 DC,9e-09 DD,6e-05 DE,5e-09 DF,1e-05 DG,4e-05 DH,6e-04 DI,4e-07  
 DJ,6e-08 DK,1e-06 DL,9e-06 DM,3e-04 DN,1e-07 DO,6e-05 DP,4e-07  
 DQ,5e-07 DR,1e-07 DS,6e-07 DT,3e-07 DU,9e-08 DV,7e-06 DW,2e-09  
 DX,7e-08 DY,6e-04 DZ,3e-08 EA,2e-08 EB,7e-09 EC,8e-05 ED,8e-04  
 EE,4e-09 EF,4e-04 EG,4e-09

### A.2.3 Set 3 in Figure III.13

- Case 1

Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
 GATE1, AND, GATE2, G, H, GATE3, I, J, K, L  
 GATE2, OR, GATE4, M, N, O, P, Q  
 GATE3, AND, GATE5, R, S, T, U, V, GATE6, GATE7  
 GATE4, OR, GATE8, W  
 GATE5, AND, GATE9, X, Y, Z  
 GATE6, OR, GATE10, BA  
 GATE7, AND, GATE11, BB, BC, BD, BE  
 GATE8, AND, GATE12, BF, BG, BH, BI

GATE9, AND, GATE13, BJ, BK, BL, BM, BN, BO, BP  
 GATE10, OR, GATE14, GATE15, BQ, BR, BS, BT, BU  
 GATE11, OR, GATE16, BV  
 GATE12, AND, GATE17, BW, BX, BY, BZ, CA  
 GATE13, OR, GATE18, CB  
 GATE14, OR, GATE19, CC, CD  
 GATE15, OR, CE, CF, CG  
 GATE16, AND, CH, CI, CJ, CK  
 GATE17, OR, CL, CM, CN, CO  
 GATE18, AND, CP, CQ, CR, CS  
 GATE19, AND, CT, CU, CV, CW, CX, CY, CZ

#### Probability of Component Failures

A,3e-07 B,3e-07 C,3e-05 D,6e-04 E,5e-05 F,4e-08 G,6e-06 H,5e-06  
 I,7e-05 J,5e-09 K,3e-08 L,6e-08 M,5e-08 N,3e-06 O,7e-09 P,6e-08  
 Q,1e-04 R,1e-05 S,6e-06 T,9e-08 U,6e-06 V,9e-05 W,2e-08 X,3e-06  
 Y,8e-09 Z,5e-08 BA,4e-06 BB,7e-06 BC,8e-05 BD,4e-07 BE,3e-07  
 BF,9e-06 BG,7e-07 BH,8e-09 BI,9e-07 BJ,6e-09 BK,2e-09 BL,3e-06  
 BM,9e-07 BN,4e-09 BO,3e-04 BP,5e-05 BQ,6e-04 BR,2e-06 BS,1e-07  
 BT,3e-06 BU,3e-08 BV,7e-05 BW,4e-07 BX,8e-08 BY,6e-04 BZ,4e-06  
 CA,7e-07 CB,5e-09 CC,2e-09 CD,3e-05 CE,7e-04 CF,8e-04 CG,4e-07  
 CH,2e-04 CI,4e-09 CJ,8e-05 CK,6e-09 CL,5e-05 CM,4e-09 CN,3e-09  
 CO,3e-04 CP,9e-08 CQ,9e-05 CR,9e-05 CS,7e-07 CT,3e-09 CU,5e-05

CV,7e-05 CW,9e-08 CX,3e-06 CY,5e-06 CZ,6e-09

- Case 2

Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
GATE1, OR, GATE2, G, H, GATE3, I, J, K, L  
GATE2, AND, GATE4, M, N, O, P, Q  
GATE3, OR, GATE5, R, S, T, U, V, GATE6, GATE7  
GATE4, OR, GATE8, W  
GATE5, OR, GATE9, X, Y, Z  
GATE6, AND, GATE10, BA  
GATE7, OR, GATE11, BB, BC, BD, BE  
GATE8, AND, GATE12, BF, BG, BH, BI  
GATE9, OR, GATE13, BJ, BK, BL, BM, BN, BO, BP  
GATE10, OR, GATE14, GATE15, BQ, BR, BS, BT, BU  
GATE11, AND, GATE16, BV  
GATE12, AND, GATE17, BW, BX, BY, BZ, CA  
GATE13, AND, GATE18, CB  
GATE14, AND, GATE19, CC, CD  
GATE15, OR, CE, CF, CG  
GATE16, AND, CH, CI, CJ, CK  
GATE17, AND, CL, CM, CN, CO  
GATE18, OR, CP, CQ, CR, CS

GATE19, AND, CT, CU, CV, CW, CX, CY, CZ

### Probability of Component Failures

A,7e-04 B,4e-09 C,2e-08 D,7e-04 E,1e-04 F,4e-09 G,3e-06 H,1e-09  
I,6e-06 J,5e-05 K,3e-04 L,4e-06 M,7e-07 N,3e-07 O,6e-05 P,3e-05  
Q,1e-06 R,3e-07 S,3e-07 T,5e-09 U,8e-04 V,7e-04 W,3e-07 X,5e-08  
Y,7e-05 Z,1e-06 BA,8e-04 BB,7e-06 BC,8e-04 BD,1e-05 BE,4e-05  
BF,8e-05 BG,1e-09 BH,1e-08 BI,5e-04 BJ,3e-04 BK,8e-08 BL,7e-05  
BM,7e-04 BN,1e-06 BO,8e-06 BP,4e-07 BQ,6e-05 BR,4e-10 BS,4e-06  
BT,2e-04 BU,9e-04 BV,9e-08 BW,5e-07 BX,7e-09 BY,8e-06 BZ,3e-07  
CA,1e-08 CB,8e-09 CC,3e-09 CD,6e-06 CE,4e-08 CF,7e-07 CG,1e-09  
CH,5e-08 CI,3e-05 CJ,1e-04 CK,6e-09 CL,4e-05 CM,1e-06 CN,1e-07  
CO,8e-07 CP,8e-09 CQ,6e-06 CR,2e-04 CS,9e-05 CT,8e-07 CU,1e-05  
CV,4e-06 CW,1e-06 CX,5e-06 CY,1e-08 CZ,5e-08

- Case 3

### Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
GATE1, OR, GATE2, G, H, GATE3, I, J, K, L  
GATE2, OR, GATE4, M, N, O, P, Q  
GATE3, AND, GATE5, R, S, T, U, V, GATE6, GATE7

GATE4, AND, GATE8, W  
 GATE5, OR, GATE9, X, Y, Z  
 GATE6, OR, GATE10, BA  
 GATE7, AND, GATE11, BB, BC, BD, BE  
 GATE8, OR, GATE12, BF, BG, BH, BI  
 GATE9, AND, GATE13, BJ, BK, BL, BM, BN, BO, BP  
 GATE10, OR, GATE14, GATE15, BQ, BR, BS, BT, BU  
 GATE11, OR, GATE16, BV  
 GATE12, AND, GATE17, BW, BX, BY, BZ, CA  
 GATE13, AND, GATE18, CB  
 GATE14, OR, GATE19, CC, CD  
 GATE15, OR, CE, CF, CG  
 GATE16, AND, CH, CI, CJ, CK  
 GATE17, AND, CL, CM, CN, CO  
 GATE18, AND, CP, CQ, CR, CS  
 GATE19, OR, CT, CU, CV, CW, CX, CY, CZ

#### Probability of Component Failures

A,8e-06 B,3e-07 C,9e-08 D,2e-04 E,8e-09 F,8e-06 G,6e-06 H,1e-07  
 I,6e-07 J,5e-08 K,8e-08 L,5e-06 M,5e-07 N,4e-08 O,7e-04 P,7e-08  
 Q,7e-04 R,8e-06 S,8e-08 T,2e-09 U,1e-08 V,9e-04 W,3e-05 X,5e-07  
 Y,6e-08 Z,6e-07 BA,7e-05 BB,8e-05 BC,1e-05 BD,8e-06 BE,9e-04  
 BF,3e-08 BG,2e-09 BH,4e-06 BI,1e-05 BJ,4e-05 BK,2e-08 BL,8e-08

BM,5e-06 BN,5e-08 BO,4e-09 BP,6e-07 BQ,3e-04 BR,3e-09 BS,6e-05  
BT,3e-05 BU,3e-08 BV,3e-05 BW,8e-05 BX,8e-09 BY,3e-06 BZ,1e-05  
CA,4e-05 CB,5e-04 CC,6e-08 CD,2e-09 CE,7e-09 CF,1e-03 CG,1e-05  
CH,2e-05 CI,4e-05 CJ,4e-09 CK,2e-08 CL,1e-06 CM,1e-07 CN,7e-04  
CO,6e-07 CP,2e-04 CQ,5e-07 CR,9e-06 CS,7e-05 CT,1e-05 CU,4e-07  
CV,7e-04 CW,7e-07 CX,7e-07 CY,2e-08 CZ,2e-04

- Case 4

Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
GATE1, AND, GATE2, G, H, GATE3, I, J, K, L  
GATE2, OR, GATE4, M, N, O, P, Q  
GATE3, OR, GATE5, R, S, T, U, V, GATE6, GATE7  
GATE4, AND, GATE8, W  
GATE5, AND, GATE9, X, Y, Z  
GATE6, AND, GATE10, BA  
GATE7, AND, GATE11, BB, BC, BD, BE  
GATE8, AND, GATE12, BF, BG, BH, BI  
GATE9, AND, GATE13, BJ, BK, BL, BM, BN, BO, BP  
GATE10, OR, GATE14, GATE15, BQ, BR, BS, BT, BU  
GATE11, AND, GATE16, BV  
GATE12, OR, GATE17, BW, BX, BY, BZ, CA  
GATE13, OR, GATE18, CB

GATE14, AND, GATE19, CC, CD  
 GATE15, AND, CE, CF, CG  
 GATE16, AND, CH, CI, CJ, CK  
 GATE17, AND, CL, CM, CN, CO  
 GATE18, OR, CP, CQ, CR, CS  
 GATE19, OR, CT, CU, CV, CW, CX, CY, CZ

### Probability of Component Failures

A,4e-09 B,8e-06 C,2e-04 D,4e-09 E,2e-05 F,6e-05 G,6e-04 H,5e-08  
 I,5e-05 J,9e-09 K,1e-05 L,4e-08 M,4e-08 N,7e-06 O,7e-05 P,8e-09  
 Q,7e-06 R,4e-04 S,8e-08 T,9e-08 U,8e-05 V,1e-08 W,6e-08 X,4e-08  
 Y,2e-08 Z,3e-09 BA,3e-05 BB,4e-09 BC,7e-05 BD,6e-06 BE,8e-09  
 BF,5e-07 BG,4e-06 BH,4e-06 BI,2e-06 BJ,8e-06 BK,2e-09 BL,4e-05  
 BM,4e-10 BN,4e-08 BO,1e-08 BP,8e-06 BQ,4e-09 BR,2e-09 BS,4e-08  
 BT,6e-06 BU,4e-05 BV,5e-04 BW,3e-04 BX,4e-06 BY,1e-06 BZ,8e-09  
 CA,2e-09 CB,5e-09 CC,2e-07 CD,7e-06 CE,6e-07 CF,7e-08 CG,6e-06  
 CH,9e-07 CI,3e-07 CJ,5e-06 CK,2e-07 CL,9e-04 CM,8e-09 CN,6e-07  
 CO,1e-05 CP,4e-09 CQ,8e-06 CR,4e-08 CS,3e-08 CT,1e-04 CU,7e-05  
 CV,1e-04 CW,8e-09 CX,3e-09 CY,2e-08 CZ,5e-09

- Case 5

Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
GATE1, AND, GATE2, G, H, GATE3, I, J, K, L  
GATE2, OR, GATE4, M, N, O, P, Q  
GATE3, OR, GATE5, R, S, T, U, V, GATE6, GATE7  
GATE4, AND, GATE8, W  
GATE5, OR, GATE9, X, Y, Z  
GATE6, AND, GATE10, BA  
GATE7, OR, GATE11, BB, BC, BD, BE  
GATE8, AND, GATE12, BF, BG, BH, BI  
GATE9, OR, GATE13, BJ, BK, BL, BM, BN, BO, BP  
GATE10, AND, GATE14, GATE15, BQ, BR, BS, BT, BU  
GATE11, OR, GATE16, BV  
GATE12, OR, GATE17, BW, BX, BY, BZ, CA  
GATE13, OR, GATE18, CB  
GATE14, AND, GATE19, CC, CD  
GATE15, AND, CE, CF, CG  
GATE16, AND, CH, CI, CJ, CK  
GATE17, AND, CL, CM, CN, CO  
GATE18, OR, CP, CQ, CR, CS  
GATE19, AND, CT, CU, CV, CW, CX, CY, CZ

Probability of Component Failures

A,7e-09 B,7e-04 C,4e-09 D,9e-07 E,7e-06 F,5e-05 G,4e-06 H,5e-05  
 I,3e-08 J,8e-07 K,2e-07 L,1e-07 M,4e-08 N,1e-04 O,1e-09 P,5e-04  
 Q,3e-07 R,6e-06 S,7e-08 T,2e-06 U,6e-08 V,2e-09 W,3e-06 X,6e-06  
 Y,1e-06 Z,2e-06 BA,9e-08 BB,3e-09 BC,5e-04 BD,4e-07 BE,3e-09  
 BF,9e-07 BG,6e-07 BH,1e-08 BI,7e-04 BJ,4e-05 BK,7e-06 BL,2e-08  
 BM,2e-04 BN,6e-09 BO,4e-06 BP,3e-04 BQ,7e-05 BR,9e-07 BS,4e-07  
 BT,5e-06 BU,4e-06 BV,4e-09 BW,5e-05 BX,2e-05 BY,8e-05 BZ,8e-06  
 CA,7e-05 CB,6e-05 CC,4e-06 CD,5e-04 CE,2e-07 CF,7e-06 CG,5e-08  
 CH,3e-06 CI,1e-08 CJ,1e-07 CK,8e-07 CL,2e-07 CM,9e-08 CN,5e-07  
 CO,2e-09 CP,7e-07 CQ,1e-03 CR,2e-05 CS,8e-08 CT,9e-05 CU,2e-09  
 CV,8e-07 CW,2e-07 CX,9e-09 CY,8e-05 CZ,4e-04

- Case 6

Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
 GATE1, AND, GATE2, G, H, GATE3, I, J, K, L  
 GATE2, OR, GATE4, M, N, O, P, Q  
 GATE3, AND, GATE5, R, S, T, U, V, GATE6, GATE7  
 GATE4, OR, GATE8, W  
 GATE5, AND, GATE9, X, Y, Z  
 GATE6, OR, GATE10, BA  
 GATE7, AND, GATE11, BB, BC, BD, BE  
 GATE8, OR, GATE12, BF, BG, BH, BI

GATE9, OR, GATE13, BJ, BK, BL, BM, BN, BO, BP  
 GATE10, AND, GATE14, GATE15, BQ, BR, BS, BT, BU  
 GATE11, AND, GATE16, BV  
 GATE12, OR, GATE17, BW, BX, BY, BZ, CA  
 GATE13, OR, GATE18, CB  
 GATE14, OR, GATE19, CC, CD  
 GATE15, AND, CE, CF, CG  
 GATE16, AND, CH, CI, CJ, CK  
 GATE17, OR, CL, CM, CN, CO  
 GATE18, OR, CP, CQ, CR, CS  
 GATE19, AND, CT, CU, CV, CW, CX, CY, CZ

#### Probability of Component Failures

A,7e-04 B,2e-04 C,7e-08 D,8e-04 E,6e-08 F,6e-06 G,2e-07 H,5e-09  
 I,9e-05 J,5e-07 K,6e-04 L,5e-06 M,3e-05 N,6e-09 O,1e-11 P,5e-05  
 Q,9e-05 R,6e-07 S,4e-09 T,1e-09 U,2e-06 V,4e-07 W,9e-07 X,7e-09  
 Y,3e-06 Z,7e-09 BA,4e-07 BB,4e-09 BC,4e-08 BD,6e-06 BE,4e-09  
 BF,8e-06 BG,8e-06 BH,1e-04 BI,2e-04 BJ,8e-05 BK,4e-07 BL,2e-04  
 BM,7e-05 BN,4e-04 BO,1e-05 BP,7e-09 BQ,3e-04 BR,7e-08 BS,8e-05  
 BT,8e-05 BU,2e-09 BV,9e-07 BW,7e-07 BX,3e-09 BY,1e-07 BZ,8e-09  
 CA,9e-04 CB,2e-08 CC,4e-09 CD,7e-09 CE,6e-07 CF,6e-09 CG,8e-09  
 CH,1e-07 CI,5e-08 CJ,5e-06 CK,2e-09 CL,9e-08 CM,9e-06 CN,4e-06  
 CO,9e-09 CP,2e-06 CQ,7e-04 CR,9e-04 CS,1e-08 CT,4e-06 CU,5e-08

CV,2e-04 CW,4e-07 CX,3e-08 CY,4e-06 CZ,3e-09

- Case 7

Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
GATE1, OR, GATE2, G, H, GATE3, I, J, K, L  
GATE2, AND, GATE4, M, N, O, P, Q  
GATE3, AND, GATE5, R, S, T, U, V, GATE6, GATE7  
GATE4, AND, GATE8, W  
GATE5, AND, GATE9, X, Y, Z  
GATE6, AND, GATE10, BA  
GATE7, OR, GATE11, BB, BC, BD, BE  
GATE8, AND, GATE12, BF, BG, BH, BI  
GATE9, AND, GATE13, BJ, BK, BL, BM, BN, BO, BP  
GATE10, AND, GATE14, GATE15, BQ, BR, BS, BT, BU  
GATE11, AND, GATE16, BV  
GATE12, OR, GATE17, BW, BX, BY, BZ, CA  
GATE13, AND, GATE18, CB  
GATE14, OR, GATE19, CC, CD  
GATE15, AND, CE, CF, CG  
GATE16, AND, CH, CI, CJ, CK  
GATE17, OR, CL, CM, CN, CO  
GATE18, OR, CP, CQ, CR, CS

GATE19, OR, CT, CU, CV, CW, CX, CY, CZ

### Probability of Component Failures

A,7e-05 B,1e-05 C,8e-09 D,1e-07 E,3e-08 F,2e-04 G,6e-05 H,1e-04  
I,5e-06 J,2e-11 K,6e-07 L,1e-04 M,1e-06 N,5e-06 O,1e-06 P,2e-04  
Q,8e-08 R,7e-06 S,2e-07 T,6e-05 U,7e-10 V,9e-08 W,7e-05 X,5e-04  
Y,7e-04 Z,8e-05 BA,3e-06 BB,6e-07 BC,1e-08 BD,6e-09 BE,5e-09  
BF,6e-07 BG,9e-08 BH,5e-05 BI,2e-07 BJ,7e-04 BK,7e-08 BL,5e-08  
BM,3e-05 BN,2e-04 BO,8e-08 BP,5e-06 BQ,3e-04 BR,2e-06 BS,1e-05  
BT,3e-05 BU,7e-08 BV,8e-09 BW,6e-06 BX,6e-06 BY,7e-04 BZ,7e-07  
CA,9e-04 CB,8e-05 CC,1e-09 CD,2e-06 CE,8e-08 CF,4e-09 CG,4e-05  
CH,8e-09 CI,4e-07 CJ,4e-04 CK,3e-04 CL,6e-08 CM,8e-05 CN,9e-06  
CO,6e-07 CP,6e-05 CQ,7e-04 CR,1e-08 CS,9e-10 CT,7e-08 CU,4e-05  
CV,6e-09 CW,6e-08 CX,8e-07 CY,5e-04 CZ,2e-04

- Case 8

### Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
GATE1, OR, GATE2, G, H, GATE3, I, J, K, L  
GATE2, OR, GATE4, M, N, O, P, Q  
GATE3, AND, GATE5, R, S, T, U, V, GATE6, GATE7

GATE4, OR, GATE8, W  
 GATE5, OR, GATE9, X, Y, Z  
 GATE6, OR, GATE10, BA  
 GATE7, OR, GATE11, BB, BC, BD, BE  
 GATE8, OR, GATE12, BF, BG, BH, BI  
 GATE9, AND, GATE13, BJ, BK, BL, BM, BN, BO, BP  
 GATE10, AND, GATE14, GATE15, BQ, BR, BS, BT, BU  
 GATE11, OR, GATE16, BV  
 GATE12, AND, GATE17, BW, BX, BY, BZ, CA  
 GATE13, AND, GATE18, CB  
 GATE14, AND, GATE19, CC, CD  
 GATE15, OR, CE, CF, CG  
 GATE16, OR, CH, CI, CJ, CK  
 GATE17, AND, CL, CM, CN, CO  
 GATE18, OR, CP, CQ, CR, CS  
 GATE19, AND, CT, CU, CV, CW, CX, CY, CZ

#### Probability of Component Failures

A,2e-08 B,9e-09 C,4e-07 D,1e-08 E,1e-05 F,5e-09 G,6e-04 H,5e-06  
 I,4e-05 J,7e-06 K,3e-09 L,9e-07 M,4e-08 N,8e-06 O,4e-06 P,4e-08  
 Q,3e-06 R,4e-08 S,2e-04 T,1e-04 U,4e-07 V,4e-09 W,5e-08 X,2e-04  
 Y,6e-06 Z,1e-03 BA,2e-07 BB,3e-06 BC,8e-05 BD,9e-08 BE,3e-09  
 BF,9e-05 BG,8e-09 BH,3e-04 BI,5e-08 BJ,8e-06 BK,5e-06 BL,7e-09

BM,2e-09 BN,2e-09 BO,5e-06 BP,3e-05 BQ,8e-05 BR,1e-06 BS,2e-09  
BT,3e-07 BU,7e-05 BV,3e-05 BW,8e-07 BX,8e-05 BY,2e-09 BZ,7e-04  
CA,7e-09 CB,7e-08 CC,6e-09 CD,2e-06 CE,8e-08 CF,4e-04 CG,9e-05  
CH,7e-08 CI,4e-04 CJ,9e-04 CK,2e-09 CL,1e-03 CM,3e-06 CN,4e-09  
CO,2e-08 CP,8e-06 CQ,3e-09 CR,2e-07 CS,4e-07 CT,8e-06 CU,3e-04  
CV,2e-08 CW,6e-06 CX,9e-07 CY,5e-07 CZ,7e-05

- Case 9

Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
GATE1, AND, GATE2, G, H, GATE3, I, J, K, L  
GATE2, AND, GATE4, M, N, O, P, Q  
GATE3, OR, GATE5, R, S, T, U, V, GATE6, GATE7  
GATE4, OR, GATE8, W  
GATE5, OR, GATE9, X, Y, Z  
GATE6, OR, GATE10, BA  
GATE7, OR, GATE11, BB, BC, BD, BE  
GATE8, OR, GATE12, BF, BG, BH, BI  
GATE9, OR, GATE13, BJ, BK, BL, BM, BN, BO, BP  
GATE10, OR, GATE14, GATE15, BQ, BR, BS, BT, BU  
GATE11, AND, GATE16, BV  
GATE12, AND, GATE17, BW, BX, BY, BZ, CA  
GATE13, AND, GATE18, CB

GATE14, OR, GATE19, CC, CD  
 GATE15, OR, CE, CF, CG  
 GATE16, AND, CH, CI, CJ, CK  
 GATE17, AND, CL, CM, CN, CO  
 GATE18, AND, CP, CQ, CR, CS  
 GATE19, OR, CT, CU, CV, CW, CX, CY, CZ

### Probability of Component Failures

A,5e-05 B,4e-05 C,5e-05 D,8e-06 E,7e-08 F,7e-05 G,1e-04 H,1e-05  
 I,6e-07 J,4e-07 K,5e-08 L,9e-09 M,3e-05 N,2e-04 O,7e-09 P,9e-05  
 Q,1e-08 R,2e-06 S,6e-04 T,2e-09 U,3e-08 V,5e-05 W,5e-06 X,1e-03  
 Y,9e-04 Z,9e-05 BA,6e-06 BB,6e-07 BC,4e-09 BD,5e-07 BE,6e-09  
 BF,6e-04 BG,7e-08 BH,9e-07 BI,3e-05 BJ,3e-07 BK,3e-05 BL,9e-05  
 BM,6e-05 BN,8e-06 BO,2e-05 BP,2e-08 BQ,9e-06 BR,8e-07 BS,1e-05  
 BT,7e-08 BU,4e-08 BV,4e-05 BW,2e-07 BX,9e-04 BY,5e-08 BZ,9e-07  
 CA,1e-04 CB,1e-06 CC,1e-06 CD,7e-06 CE,9e-09 CF,7e-09 CG,2e-07  
 CH,3e-07 CI,5e-07 CJ,3e-07 CK,4e-08 CL,3e-06 CM,8e-08 CN,6e-10  
 CO,8e-07 CP,6e-04 CQ,9e-07 CR,5e-07 CS,9e-05 CT,3e-05 CU,3e-09  
 CV,5e-07 CW,9e-06 CX,1e-06 CY,6e-05 CZ,1e-07

- Case 10

Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
GATE1, OR, GATE2, G, H, GATE3, I, J, K, L  
GATE2, OR, GATE4, M, N, O, P, Q  
GATE3, OR, GATE5, R, S, T, U, V, GATE6, GATE7  
GATE4, AND, GATE8, W  
GATE5, AND, GATE9, X, Y, Z  
GATE6, AND, GATE10, BA  
GATE7, AND, GATE11, BB, BC, BD, BE  
GATE8, AND, GATE12, BF, BG, BH, BI  
GATE9, OR, GATE13, BJ, BK, BL, BM, BN, BO, BP  
GATE10, OR, GATE14, GATE15, BQ, BR, BS, BT, BU  
GATE11, OR, GATE16, BV  
GATE12, OR, GATE17, BW, BX, BY, BZ, CA  
GATE13, AND, GATE18, CB  
GATE14, AND, GATE19, CC, CD  
GATE15, OR, CE, CF, CG  
GATE16, OR, CH, CI, CJ, CK  
GATE17, AND, CL, CM, CN, CO  
GATE18, OR, CP, CQ, CR, CS  
GATE19, OR, CT, CU, CV, CW, CX, CY, CZ

Probability of Component Failures

A,6e-08 B,1e-03 C,4e-04 D,2e-09 E,9e-06 F,2e-06 G,7e-07 H,1e-04  
 I,7e-09 J,6e-07 K,4e-07 L,8e-08 M,9e-06 N,2e-08 O,5e-04 P,1e-05  
 Q,6e-07 R,6e-04 S,6e-09 T,4e-04 U,3e-09 V,9e-09 W,3e-04 X,2e-09  
 Y,9e-04 Z,7e-04 BA,2e-08 BB,2e-06 BC,4e-07 BD,9e-08 BE,9e-08  
 BF,4e-09 BG,2e-05 BH,6e-08 BI,8e-05 BJ,9e-05 BK,9e-07 BL,6e-05  
 BM,7e-09 BN,4e-07 BO,9e-05 BP,1e-06 BQ,7e-05 BR,7e-07 BS,5e-07  
 BT,9e-06 BU,4e-09 BV,7e-04 BW,6e-07 BX,5e-09 BY,2e-09 BZ,9e-05  
 CA,5e-08 CB,4e-04 CC,3e-04 CD,7e-08 CE,1e-07 CF,4e-09 CG,7e-04  
 CH,4e-06 CI,2e-05 CJ,3e-07 CK,2e-08 CL,7e-05 CM,4e-07 CN,3e-08  
 CO,4e-06 CP,2e-05 CQ,3e-04 CR,5e-08 CS,6e-08 CT,6e-09 CU,6e-09  
 CV,6e-08 CW,2e-07 CX,7e-05 CY,4e-06 CZ,3e-10

- Case 11

Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
 GATE1, OR, GATE2, G, H, GATE3, I, J, K, L  
 GATE2, AND, GATE4, M, N, O, P, Q  
 GATE3, OR, GATE5, R, S, T, U, V, GATE6, GATE7  
 GATE4, OR, GATE8, W  
 GATE5, AND, GATE9, X, Y, Z  
 GATE6, AND, GATE10, BA  
 GATE7, OR, GATE11, BB, BC, BD, BE  
 GATE8, AND, GATE12, BF, BG, BH, BI

GATE9, OR, GATE13, BJ, BK, BL, BM, BN, BO, BP  
 GATE10, OR, GATE14, GATE15, BQ, BR, BS, BT, BU  
 GATE11, OR, GATE16, BV  
 GATE12, AND, GATE17, BW, BX, BY, BZ, CA  
 GATE13, AND, GATE18, CB  
 GATE14, OR, GATE19, CC, CD  
 GATE15, AND, CE, CF, CG  
 GATE16, OR, CH, CI, CJ, CK  
 GATE17, OR, CL, CM, CN, CO  
 GATE18, AND, CP, CQ, CR, CS  
 GATE19, AND, CT, CU, CV, CW, CX, CY, CZ

#### Probability of Component Failures

A,3e-07 B,8e-05 C,2e-09 D,2e-04 E,5e-06 F,5e-05 G,3e-08 H,7e-05  
 I,3e-06 J,2e-07 K,6e-06 L,6e-08 M,3e-04 N,7e-05 O,9e-04 P,6e-08  
 Q,2e-04 R,9e-09 S,2e-04 T,8e-05 U,8e-08 V,5e-08 W,4e-07 X,3e-08  
 Y,4e-05 Z,3e-07 BA,9e-07 BB,9e-06 BC,3e-08 BD,6e-04 BE,3e-08  
 BF,2e-06 BG,2e-04 BH,6e-05 BI,6e-05 BJ,8e-07 BK,6e-05 BL,4e-05  
 BM,5e-08 BN,9e-09 BO,5e-08 BP,2e-04 BQ,9e-07 BR,1e-03 BS,6e-05  
 BT,6e-07 BU,5e-08 BV,4e-09 BW,8e-07 BX,3e-07 BY,3e-09 BZ,9e-08  
 CA,1e-04 CB,7e-06 CC,5e-06 CD,5e-07 CE,9e-05 CF,6e-07 CG,4e-06  
 CH,1e-05 CI,1e-07 CJ,2e-08 CK,4e-09 CL,4e-05 CM,8e-06 CN,6e-07  
 CO,8e-05 CP,9e-07 CQ,8e-04 CR,7e-06 CS,5e-04 CT,1e-04 CU,5e-08

CV,6e-04 CW,3e-05 CX,9e-06 CY,7e-04 CZ,2e-05

- Case 12

Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
GATE1, OR, GATE2, G, H, GATE3, I, J, K, L  
GATE2, AND, GATE4, M, N, O, P, Q  
GATE3, AND, GATE5, R, S, T, U, V, GATE6, GATE7  
GATE4, AND, GATE8, W  
GATE5, AND, GATE9, X, Y, Z  
GATE6, AND, GATE10, BA  
GATE7, AND, GATE11, BB, BC, BD, BE  
GATE8, AND, GATE12, BF, BG, BH, BI  
GATE9, OR, GATE13, BJ, BK, BL, BM, BN, BO, BP  
GATE10, OR, GATE14, GATE15, BQ, BR, BS, BT, BU  
GATE11, AND, GATE16, BV  
GATE12, OR, GATE17, BW, BX, BY, BZ, CA  
GATE13, AND, GATE18, CB  
GATE14, AND, GATE19, CC, CD  
GATE15, OR, CE, CF, CG  
GATE16, AND, CH, CI, CJ, CK  
GATE17, OR, CL, CM, CN, CO  
GATE18, OR, CP, CQ, CR, CS

GATE19, AND, CT, CU, CV, CW, CX, CY, CZ

### Probability of Component Failures

A,4e-05 B,7e-06 C,4e-06 D,3e-08 E,9e-04 F,3e-08 G,9e-09 H,8e-04  
I,3e-08 J,7e-05 K,6e-08 L,4e-05 M,9e-05 N,7e-09 O,8e-09 P,1e-06  
Q,8e-08 R,4e-05 S,3e-04 T,1e-04 U,6e-09 V,1e-04 W,3e-04 X,5e-09  
Y,1e-04 Z,5e-07 BA,7e-07 BB,3e-05 BC,7e-04 BD,8e-05 BE,4e-07  
BF,4e-04 BG,7e-06 BH,5e-09 BI,7e-05 BJ,3e-05 BK,9e-07 BL,3e-08  
BM,8e-08 BN,5e-05 BO,1e-09 BP,7e-10 BQ,9e-08 BR,2e-08 BS,4e-08  
BT,2e-09 BU,3e-06 BV,6e-06 BW,4e-08 BX,6e-09 BY,3e-04 BZ,6e-06  
CA,8e-09 CB,3e-08 CC,4e-07 CD,2e-08 CE,9e-05 CF,6e-04 CG,7e-05  
CH,8e-04 CI,7e-04 CJ,3e-05 CK,2e-07 CL,8e-08 CM,4e-06 CN,4e-05  
CO,2e-07 CP,8e-08 CQ,3e-04 CR,5e-08 CS,3e-08 CT,8e-06 CU,5e-09  
CV,4e-05 CW,8e-05 CX,4e-05 CY,2e-06 CZ,9e-05

- Case 13

### Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
GATE1, OR, GATE2, G, H, GATE3, I, J, K, L  
GATE2, AND, GATE4, M, N, O, P, Q  
GATE3, OR, GATE5, R, S, T, U, V, GATE6, GATE7

GATE4, OR, GATE8, W  
 GATE5, OR, GATE9, X, Y, Z  
 GATE6, AND, GATE10, BA  
 GATE7, AND, GATE11, BB, BC, BD, BE  
 GATE8, OR, GATE12, BF, BG, BH, BI  
 GATE9, AND, GATE13, BJ, BK, BL, BM, BN, BO, BP  
 GATE10, AND, GATE14, GATE15, BQ, BR, BS, BT, BU  
 GATE11, OR, GATE16, BV  
 GATE12, OR, GATE17, BW, BX, BY, BZ, CA  
 GATE13, OR, GATE18, CB  
 GATE14, OR, GATE19, CC, CD  
 GATE15, AND, CE, CF, CG  
 GATE16, OR, CH, CI, CJ, CK  
 GATE17, OR, CL, CM, CN, CO  
 GATE18, AND, CP, CQ, CR, CS  
 GATE19, AND, CT, CU, CV, CW, CX, CY, CZ

#### Probability of Component Failures

A,9e-07 B,3e-09 C,6e-09 D,8e-08 E,4e-10 F,9e-04 G,4e-08 H,5e-09  
 I,7e-08 J,9e-05 K,9e-07 L,1e-06 M,8e-07 N,8e-09 O,8e-04 P,7e-09  
 Q,7e-06 R,4e-08 S,9e-07 T,1e-05 U,7e-06 V,6e-09 W,7e-09 X,8e-05  
 Y,7e-06 Z,4e-05 BA,5e-04 BB,4e-08 BC,6e-08 BD,1e-06 BE,5e-06  
 BF,2e-06 BG,7e-08 BH,9e-07 BI,4e-09 BJ,2e-09 BK,3e-09 BL,6e-05

BM,4e-07 BN,8e-04 BO,7e-04 BP,3e-09 BQ,5e-04 BR,3e-04 BS,1e-07  
BT,7e-09 BU,8e-09 BV,2e-06 BW,2e-05 BX,7e-08 BY,3e-04 BZ,4e-07  
CA,9e-09 CB,8e-07 CC,6e-05 CD,5e-09 CE,9e-05 CF,6e-07 CG,9e-07  
CH,8e-04 CI,8e-09 CJ,3e-07 CK,9e-07 CL,3e-05 CM,6e-06 CN,8e-06  
CO,6e-05 CP,2e-04 CQ,7e-09 CR,4e-05 CS,4e-09 CT,6e-08 CU,2e-09  
CV,6e-06 CW,1e-03 CX,5e-04 CY,7e-08 CZ,3e-07

- Case 14

Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
GATE1, AND, GATE2, G, H, GATE3, I, J, K, L  
GATE2, AND, GATE4, M, N, O, P, Q  
GATE3, AND, GATE5, R, S, T, U, V, GATE6, GATE7  
GATE4, AND, GATE8, W  
GATE5, AND, GATE9, X, Y, Z  
GATE6, OR, GATE10, BA  
GATE7, AND, GATE11, BB, BC, BD, BE  
GATE8, AND, GATE12, BF, BG, BH, BI  
GATE9, OR, GATE13, BJ, BK, BL, BM, BN, BO, BP  
GATE10, AND, GATE14, GATE15, BQ, BR, BS, BT, BU  
GATE11, AND, GATE16, BV  
GATE12, OR, GATE17, BW, BX, BY, BZ, CA  
GATE13, OR, GATE18, CB

GATE14, AND, GATE19, CC, CD  
 GATE15, AND, CE, CF, CG  
 GATE16, OR, CH, CI, CJ, CK  
 GATE17, OR, CL, CM, CN, CO  
 GATE18, OR, CP, CQ, CR, CS  
 GATE19, AND, CT, CU, CV, CW, CX, CY, CZ

### Probability of Component Failures

A,7e-07 B,3e-09 C,4e-05 D,4e-07 E,8e-05 F,2e-06 G,2e-08 H,6e-08  
 I,1e-03 J,4e-05 K,7e-10 L,7e-06 M,2e-06 N,5e-07 O,1e-09 P,9e-05  
 Q,3e-07 R,8e-09 S,8e-08 T,9e-04 U,5e-05 V,8e-07 W,7e-09 X,7e-06  
 Y,1e-03 Z,4e-08 BA,8e-05 BB,2e-09 BC,5e-04 BD,7e-08 BE,4e-08  
 BF,5e-08 BG,1e-06 BH,7e-05 BI,6e-06 BJ,2e-06 BK,3e-08 BL,7e-04  
 BM,4e-09 BN,6e-06 BO,8e-04 BP,9e-09 BQ,1e-08 BR,1e-09 BS,4e-07  
 BT,1e-08 BU,9e-08 BV,4e-08 BW,5e-07 BX,6e-10 BY,9e-07 BZ,9e-04  
 CA,2e-09 CB,1e-05 CC,2e-04 CD,9e-04 CE,7e-06 CF,8e-07 CG,4e-05  
 CH,8e-08 CI,3e-09 CJ,2e-07 CK,2e-06 CL,7e-06 CM,4e-06 CN,8e-09  
 CO,2e-08 CP,8e-07 CQ,4e-06 CR,8e-05 CS,5e-08 CT,5e-07 CU,6e-05  
 CV,7e-07 CW,7e-06 CX,4e-09 CY,7e-06 CZ,6e-05

- Case 15

Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
GATE1, AND, GATE2, G, H, GATE3, I, J, K, L  
GATE2, OR, GATE4, M, N, O, P, Q  
GATE3, AND, GATE5, R, S, T, U, V, GATE6, GATE7  
GATE4, AND, GATE8, W  
GATE5, OR, GATE9, X, Y, Z  
GATE6, OR, GATE10, BA  
GATE7, AND, GATE11, BB, BC, BD, BE  
GATE8, OR, GATE12, BF, BG, BH, BI  
GATE9, OR, GATE13, BJ, BK, BL, BM, BN, BO, BP  
GATE10, AND, GATE14, GATE15, BQ, BR, BS, BT, BU  
GATE11, AND, GATE16, BV  
GATE12, OR, GATE17, BW, BX, BY, BZ, CA  
GATE13, AND, GATE18, CB  
GATE14, OR, GATE19, CC, CD  
GATE15, AND, CE, CF, CG  
GATE16, AND, CH, CI, CJ, CK  
GATE17, OR, CL, CM, CN, CO  
GATE18, OR, CP, CQ, CR, CS  
GATE19, AND, CT, CU, CV, CW, CX, CY, CZ

Probability of Component Failures

A,5e-09 B,7e-08 C,7e-04 D,5e-04 E,5e-05 F,7e-05 G,2e-09 H,6e-08  
 I,5e-08 J,2e-08 K,9e-09 L,8e-06 M,4e-06 N,8e-08 O,6e-04 P,5e-06  
 Q,7e-08 R,4e-08 S,7e-09 T,1e-04 U,3e-06 V,1e-05 W,5e-08 X,6e-04  
 Y,8e-07 Z,8e-08 BA,7e-09 BB,6e-08 BC,5e-09 BD,3e-05 BE,9e-04  
 BF,3e-08 BG,9e-09 BH,4e-07 BI,1e-07 BJ,9e-05 BK,7e-08 BL,2e-06  
 BM,1e-08 BN,2e-05 BO,5e-04 BP,3e-08 BQ,8e-07 BR,6e-07 BS,1e-07  
 BT,8e-09 BU,2e-05 BV,7e-10 BW,3e-06 BX,9e-09 BY,2e-08 BZ,1e-05  
 CA,1e-09 CB,8e-09 CC,2e-09 CD,7e-09 CE,9e-09 CF,2e-09 CG,4e-09  
 CH,9e-04 CI,6e-08 CJ,7e-05 CK,8e-04 CL,2e-08 CM,2e-08 CN,3e-07  
 CO,6e-04 CP,3e-07 CQ,9e-05 CR,2e-06 CS,9e-05 CT,4e-08 CU,8e-07  
 CV,9e-05 CW,9e-05 CX,2e-08 CY,2e-05 CZ,9e-09

- Case 16

Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
 GATE1, OR, GATE2, G, H, GATE3, I, J, K, L  
 GATE2, AND, GATE4, M, N, O, P, Q  
 GATE3, AND, GATE5, R, S, T, U, V, GATE6, GATE7  
 GATE4, OR, GATE8, W  
 GATE5, AND, GATE9, X, Y, Z  
 GATE6, AND, GATE10, BA  
 GATE7, OR, GATE11, BB, BC, BD, BE  
 GATE8, OR, GATE12, BF, BG, BH, BI

GATE9, AND, GATE13, BJ, BK, BL, BM, BN, BO, BP  
 GATE10, AND, GATE14, GATE15, BQ, BR, BS, BT, BU  
 GATE11, AND, GATE16, BV  
 GATE12, AND, GATE17, BW, BX, BY, BZ, CA  
 GATE13, AND, GATE18, CB  
 GATE14, OR, GATE19, CC, CD  
 GATE15, AND, CE, CF, CG  
 GATE16, AND, CH, CI, CJ, CK  
 GATE17, AND, CL, CM, CN, CO  
 GATE18, OR, CP, CQ, CR, CS  
 GATE19, AND, CT, CU, CV, CW, CX, CY, CZ

#### Probability of Component Failures

A,7e-09 B,1e-07 C,6e-09 D,7e-05 E,4e-06 F,9e-08 G,3e-09 H,5e-09  
 I,1e-05 J,7e-06 K,3e-09 L,9e-07 M,1e-07 N,1e-04 O,6e-07 P,6e-05  
 Q,6e-06 R,3e-09 S,5e-06 T,7e-08 U,3e-07 V,7e-06 W,1e-06 X,6e-08  
 Y,6e-08 Z,3e-05 BA,1e-06 BB,6e-05 BC,1e-08 BD,2e-08 BE,4e-07  
 BF,1e-03 BG,7e-06 BH,1e-06 BI,1e-05 BJ,2e-07 BK,5e-06 BL,9e-08  
 BM,9e-05 BN,1e-05 BO,6e-07 BP,3e-04 BQ,1e-09 BR,3e-05 BS,9e-08  
 BT,3e-06 BU,8e-04 BV,5e-04 BW,1e-05 BX,9e-10 BY,6e-05 BZ,3e-08  
 CA,6e-09 CB,5e-06 CC,8e-04 CD,8e-09 CE,8e-06 CF,1e-04 CG,9e-05  
 CH,5e-04 CI,6e-08 CJ,1e-07 CK,1e-08 CL,9e-05 CM,6e-10 CN,9e-09  
 CO,4e-09 CP,5e-08 CQ,7e-07 CR,8e-04 CS,9e-07 CT,9e-04 CU,5e-09

CV,9e-06 CW,6e-04 CX,3e-06 CY,3e-09 CZ,3e-07

- Case 17

Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
GATE1, AND, GATE2, G, H, GATE3, I, J, K, L  
GATE2, AND, GATE4, M, N, O, P, Q  
GATE3, AND, GATE5, R, S, T, U, V, GATE6, GATE7  
GATE4, OR, GATE8, W  
GATE5, OR, GATE9, X, Y, Z  
GATE6, OR, GATE10, BA  
GATE7, OR, GATE11, BB, BC, BD, BE  
GATE8, OR, GATE12, BF, BG, BH, BI  
GATE9, AND, GATE13, BJ, BK, BL, BM, BN, BO, BP  
GATE10, OR, GATE14, GATE15, BQ, BR, BS, BT, BU  
GATE11, AND, GATE16, BV  
GATE12, OR, GATE17, BW, BX, BY, BZ, CA  
GATE13, AND, GATE18, CB  
GATE14, OR, GATE19, CC, CD  
GATE15, AND, CE, CF, CG  
GATE16, OR, CH, CI, CJ, CK  
GATE17, OR, CL, CM, CN, CO  
GATE18, AND, CP, CQ, CR, CS

GATE19, AND, CT, CU, CV, CW, CX, CY, CZ

### Probability of Component Failures

A,7e-06 B,3e-09 C,9e-08 D,8e-07 E,9e-05 F,8e-07 G,8e-04 H,6e-04  
I,6e-04 J,6e-09 K,2e-08 L,9e-09 M,2e-04 N,1e-04 O,4e-04 P,8e-06  
Q,2e-09 R,8e-04 S,6e-05 T,7e-06 U,9e-08 V,2e-07 W,1e-07 X,4e-09  
Y,3e-07 Z,1e-09 BA,2e-04 BB,4e-06 BC,5e-04 BD,6e-07 BE,8e-05  
BF,5e-08 BG,4e-08 BH,6e-06 BI,3e-05 BJ,5e-08 BK,1e-04 BL,4e-05  
BM,9e-07 BN,4e-08 BO,3e-07 BP,7e-09 BQ,6e-08 BR,6e-04 BS,4e-05  
BT,6e-07 BU,2e-07 BV,2e-09 BW,4e-08 BX,1e-05 BY,5e-05 BZ,5e-05  
CA,4e-09 CB,5e-04 CC,9e-08 CD,6e-04 CE,4e-08 CF,1e-06 CG,3e-08  
CH,9e-09 CI,8e-04 CJ,3e-04 CK,1e-06 CL,4e-08 CM,9e-09 CN,1e-07  
CO,6e-07 CP,5e-09 CQ,3e-05 CR,8e-04 CS,4e-09 CT,1e-04 CU,6e-07  
CV,6e-04 CW,8e-06 CX,5e-06 CY,5e-05 CZ,2e-04

- Case 18

### Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
GATE1, AND, GATE2, G, H, GATE3, I, J, K, L  
GATE2, AND, GATE4, M, N, O, P, Q  
GATE3, OR, GATE5, R, S, T, U, V, GATE6, GATE7

GATE4, AND, GATE8, W  
 GATE5, OR, GATE9, X, Y, Z  
 GATE6, AND, GATE10, BA  
 GATE7, OR, GATE11, BB, BC, BD, BE  
 GATE8, AND, GATE12, BF, BG, BH, BI  
 GATE9, AND, GATE13, BJ, BK, BL, BM, BN, BO, BP  
 GATE10, OR, GATE14, GATE15, BQ, BR, BS, BT, BU  
 GATE11, AND, GATE16, BV  
 GATE12, AND, GATE17, BW, BX, BY, BZ, CA  
 GATE13, OR, GATE18, CB  
 GATE14, AND, GATE19, CC, CD  
 GATE15, OR, CE, CF, CG  
 GATE16, AND, CH, CI, CJ, CK  
 GATE17, OR, CL, CM, CN, CO  
 GATE18, OR, CP, CQ, CR, CS  
 GATE19, AND, CT, CU, CV, CW, CX, CY, CZ

#### Probability of Component Failures

A,7e-05 B,2e-08 C,6e-07 D,4e-04 E,7e-08 F,4e-08 G,5e-09 H,8e-04  
 I,7e-04 J,3e-07 K,9e-08 L,4e-09 M,9e-07 N,4e-08 O,2e-06 P,9e-06  
 Q,3e-07 R,8e-07 S,9e-08 T,4e-06 U,1e-08 V,6e-04 W,7e-06 X,3e-10  
 Y,9e-05 Z,8e-04 BA,7e-09 BB,3e-09 BC,2e-07 BD,1e-07 BE,7e-05  
 BF,5e-06 BG,9e-09 BH,6e-06 BI,8e-09 BJ,9e-05 BK,2e-09 BL,5e-06

BM,8e-06 BN,3e-05 BO,7e-09 BP,6e-05 BQ,1e-08 BR,1e-05 BS,4e-04  
BT,7e-08 BU,1e-08 BV,8e-09 BW,6e-11 BX,7e-05 BY,8e-05 BZ,5e-06  
CA,7e-05 CB,2e-05 CC,3e-06 CD,1e-08 CE,9e-06 CF,6e-06 CG,8e-08  
CH,1e-05 CI,7e-04 CJ,5e-06 CK,3e-06 CL,3e-07 CM,1e-05 CN,7e-07  
CO,3e-04 CP,8e-08 CQ,4e-04 CR,7e-08 CS,2e-08 CT,1e-08 CU,7e-06  
CV,5e-08 CW,6e-06 CX,1e-08 CY,2e-06 CZ,5e-07

- Case 19

Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
GATE1, OR, GATE2, G, H, GATE3, I, J, K, L  
GATE2, OR, GATE4, M, N, O, P, Q  
GATE3, OR, GATE5, R, S, T, U, V, GATE6, GATE7  
GATE4, AND, GATE8, W  
GATE5, OR, GATE9, X, Y, Z  
GATE6, OR, GATE10, BA  
GATE7, OR, GATE11, BB, BC, BD, BE  
GATE8, AND, GATE12, BF, BG, BH, BI  
GATE9, AND, GATE13, BJ, BK, BL, BM, BN, BO, BP  
GATE10, OR, GATE14, GATE15, BQ, BR, BS, BT, BU  
GATE11, AND, GATE16, BV  
GATE12, AND, GATE17, BW, BX, BY, BZ, CA  
GATE13, AND, GATE18, CB

GATE14, AND, GATE19, CC, CD  
 GATE15, AND, CE, CF, CG  
 GATE16, AND, CH, CI, CJ, CK  
 GATE17, AND, CL, CM, CN, CO  
 GATE18, OR, CP, CQ, CR, CS  
 GATE19, AND, CT, CU, CV, CW, CX, CY, CZ

### Probability of Component Failures

A,4e-08 B,8e-08 C,3e-09 D,7e-09 E,2e-05 F,4e-08 G,1e-09 H,4e-09  
 I,6e-04 J,5e-06 K,2e-05 L,5e-06 M,8e-08 N,8e-07 O,9e-09 P,1e-07  
 Q,2e-06 R,3e-09 S,4e-04 T,8e-09 U,8e-04 V,2e-04 W,8e-09 X,9e-08  
 Y,1e-03 Z,2e-05 BA,7e-08 BB,4e-07 BC,3e-07 BD,8e-09 BE,7e-07  
 BF,4e-06 BG,9e-05 BH,5e-05 BI,6e-06 BJ,9e-05 BK,4e-06 BL,1e-05  
 BM,4e-08 BN,3e-09 BO,6e-09 BP,8e-06 BQ,1e-07 BR,4e-08 BS,4e-06  
 BT,6e-07 BU,6e-08 BV,4e-07 BW,9e-08 BX,1e-04 BY,3e-08 BZ,4e-09  
 CA,5e-09 CB,8e-04 CC,8e-04 CD,3e-05 CE,8e-09 CF,2e-07 CG,7e-07  
 CH,4e-09 CI,1e-09 CJ,9e-09 CK,4e-05 CL,1e-08 CM,1e-08 CN,6e-06  
 CO,6e-05 CP,7e-04 CQ,1e-09 CR,5e-07 CS,5e-06 CT,6e-05 CU,2e-06  
 CV,3e-05 CW,2e-09 CX,4e-08 CY,2e-08 CZ,4e-05

- Case 20

Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F  
GATE1, AND, GATE2, G, H, GATE3, I, J, K, L  
GATE2, OR, GATE4, M, N, O, P, Q  
GATE3, AND, GATE5, R, S, T, U, V, GATE6, GATE7  
GATE4, AND, GATE8, W  
GATE5, OR, GATE9, X, Y, Z  
GATE6, AND, GATE10, BA  
GATE7, OR, GATE11, BB, BC, BD, BE  
GATE8, AND, GATE12, BF, BG, BH, BI  
GATE9, AND, GATE13, BJ, BK, BL, BM, BN, BO, BP  
GATE10, OR, GATE14, GATE15, BQ, BR, BS, BT, BU  
GATE11, OR, GATE16, BV  
GATE12, AND, GATE17, BW, BX, BY, BZ, CA  
GATE13, OR, GATE18, CB  
GATE14, AND, GATE19, CC, CD  
GATE15, AND, CE, CF, CG  
GATE16, AND, CH, CI, CJ, CK  
GATE17, AND, CL, CM, CN, CO  
GATE18, OR, CP, CQ, CR, CS  
GATE19, AND, CT, CU, CV, CW, CX, CY, CZ

Probability of Component Failures

A,1e-08 B,8e-05 C,7e-08 D,8e-06 E,5e-05 F,6e-04 G,5e-07 H,9e-07  
 I,4e-05 J,2e-04 K,4e-05 L,7e-06 M,9e-09 N,6e-09 O,2e-06 P,6e-09  
 Q,7e-07 R,3e-09 S,8e-05 T,7e-09 U,6e-08 V,7e-04 W,1e-09 X,4e-04  
 Y,4e-05 Z,4e-09 BA,5e-08 BB,1e-05 BC,8e-09 BD,5e-04 BE,6e-09  
 BF,8e-07 BG,6e-09 BH,8e-04 BI,2e-08 BJ,9e-07 BK,4e-07 BL,7e-04  
 BM,6e-05 BN,9e-04 BO,1e-05 BP,2e-07 BQ,6e-09 BR,4e-04 BS,5e-07  
 BT,5e-08 BU,6e-09 BV,3e-07 BW,7e-07 BX,3e-09 BY,3e-08 BZ,4e-06  
 CA,6e-06 CB,2e-10 CC,7e-05 CD,4e-09 CE,6e-05 CF,5e-08 CG,1e-06  
 CH,7e-05 CI,4e-09 CJ,7e-06 CK,7e-07 CL,1e-05 CM,1e-05 CN,4e-04  
 CO,7e-07 CP,4e-08 CQ,2e-06 CR,6e-07 CS,8e-05 CT,8e-07 CU,4e-08  
 CV,6e-09 CW,7e-08 CX,9e-04 CY,9e-06 CZ,2e-04

#### A.2.4 Cases in Figure III.15

- Case 1

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
 GATE1, OR, A, B  
 GATE2, OR, C, D  
 GATE3, OR, E

Probability of Component Failures

A,9e-05 B,3e-05 C,8e-04 D,8e-04 E,1e-06

- Case 2

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B

GATE2, OR, C, D

GATE3, OR, E, F

Probability of Component Failures

A,5e-01 B,6e-05 C,8e-02 D,3e-05 E,5e-03 F,4e-06

- Case 3

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C

GATE2, OR, D, E

GATE3, OR, F, G

Probability of Component Failures

A,7e-01 B,1e-03 C,5e-05 D,5e-06 E,2e-03 F,2e-05 G,7e-06

- Case 4

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C

GATE2, OR, D, E, F

GATE3, OR, G, H

Probability of Component Failures

A,5e-03 B,9e-03 C,3e-07 D,2e-04 E,2e-03 F,2e-06 G,3e-03 H,2e-06

- Case 5

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C

GATE2, OR, D, E, F

GATE3, OR, G, H, I

Probability of Component Failures

A,8e-04 B,3e-04 C,7e-02 D,8e-04 E,6e-01 F,2e-03 G,7e-02 H,4e-01  
I,3e-01

- Case 6

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D

GATE2, OR, E, F, G

GATE3, OR, H, I, J

Probability of Component Failures

A,9e-04 B,4e-03 C,6e-01 D,5e-02 E,7e-02 F,2e-01 G,6e-06 H,1e-04  
I,5e-02 J,6e-06

- Case 7

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D

GATE2, OR, E, F, G, H

GATE3, OR, I, J, K

## Probability of Component Failures

A,1e-05 B,4e-05 C,9e-06 D,3e-06 E,9e-06 F,8e-03 G,9e-01 H,6e-02  
I,2e-04 J,6e-01 K,3e-03

- Case 8

### Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
GATE1, OR, A, B, C, D  
GATE2, OR, E, F, G, H  
GATE3, OR, I, J, K, L

## Probability of Component Failures

A,4e-05 B,2e-01 C,5e-02 D,5e-03 E,9e-02 F,3e-02 G,7e-06 H,9e-03  
I,9e-06 J,3e-03 K,3e-01 L,3e-01

- Case 9

### Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
GATE1, OR, A, B, C, D, E  
GATE2, OR, F, G, H, I

GATE3, OR, J, K, L, M

Probability of Component Failures

A,2e-06 B,7e-02 C,8e-03 D,3e-04 E,2e-05 F,8e-03 G,2e-03 H,7e-03  
I,5e-02 J,1e-04 K,9e-02 L,8e-05 M,6e-04

- Case 10

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
GATE1, OR, A, B, C, D, E  
GATE2, OR, F, G, H, I, J  
GATE3, OR, K, L, M, N

Probability of Component Failures

A,1e-01 B,2e-03 C,7e-01 D,8e-01 E,4e-02 F,4e-06 G,4e-04 H,7e-05  
I,9e-06 J,5e-02 K,8e-06 L,7e-02 M,8e-02 N,8e-05

- Case 11

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E

GATE2, OR, F, G, H, I, J

GATE3, OR, K, L, M, N, O

Probability of Component Failures

A,9e-05 B,9e-04 C,9e-03 D,1e-02 E,7e-04 F,3e-03 G,1e-03 H,2e-01

I,6e-04 J,1e-04 K,5e-04 L,2e-01 M,3e-04 N,9e-06 O,8e-03

- Case 12

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E, F

GATE2, OR, G, H, I, J, K

GATE3, OR, L, M, N, O, P

Probability of Component Failures

A,8e-02 B,2e-02 C,9e-03 D,9e-03 E,4e-05 F,8e-01 G,1e-03 H,9e-01

I,9e-06 J,2e-05 K,5e-05 L,6e-05 M,1e-03 N,5e-07 O,8e-06 P,3e-03

- Case 13

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E, F

GATE2, OR, G, H, I, J, K, L

GATE3, OR, M, N, O, P, Q

Probability of Component Failures

A,7e-01 B,4e-05 C,3e-02 D,8e-02 E,6e-04 F,9e-03 G,3e-05 H,1e-04

I,4e-03 J,8e-01 K,8e-02 L,9e-02 M,3e-03 N,9e-07 O,3e-04 P,7e-01

Q,5e-06

- Case 14

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E, F

GATE2, OR, G, H, I, J, K, L

GATE3, OR, M, N, O, P, Q, R

Probability of Component Failures

A,8e-03 B,8e-02 C,9e-08 D,5e-03 E,9e-04 F,2e-03 G,4e-01 H,3e-01  
I,6e-05 J,5e-03 K,8e-06 L,7e-04 M,5e-06 N,2e-05 O,3e-06 P,9e-06  
Q,6e-06 R,1e-01

- Case 15

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
GATE1, OR, A, B, C, D, E, F, G  
GATE2, OR, H, I, J, K, L, M  
GATE3, OR, N, O, P, Q, R, S

Probability of Component Failures

A,1e-05 B,2e-01 C,6e-02 D,3e-01 E,7e-01 F,9e-02 G,5e-03 H,7e-03  
I,1e-05 J,3e-06 K,8e-04 L,5e-04 M,4e-02 N,2e-06 O,4e-05 P,3e-02  
Q,5e-01 R,7e-02 S,4e-03

- Case 16

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E, F, G

GATE2, OR, H, I, J, K, L, M, N

GATE3, OR, O, P, Q, R, S, T

Probability of Component Failures

A,2e-04 B,2e-02 C,6e-05 D,6e-04 E,2e-05 F,2e-02 G,6e-01 H,3e-04

I,7e-06 J,4e-04 K,8e-07 L,4e-05 M,5e-03 N,5e-05 O,5e-01 P,4e-04

Q,6e-05 R,9e-03 S,3e-03 T,7e-05

- Case 17

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E, F, G

GATE2, OR, H, I, J, K, L, M, N

GATE3, OR, O, P, Q, R, S, T, U

Probability of Component Failures

A,1e-01 B,4e-05 C,8e-06 D,2e-05 E,1e-06 F,2e-01 G,2e-05 H,4e-01  
I,3e-01 J,5e-02 K,9e-02 L,5e-06 M,9e-05 N,3e-03 O,7e-04 P,1e-01  
Q,7e-04 R,3e-05 S,6e-05 T,8e-03 U,7e-06

- Case 18

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
GATE1, OR, A, B, C, D, E, F, G, H  
GATE2, OR, I, J, K, L, M, N, O  
GATE3, OR, P, Q, R, S, T, U, V

Probability of Component Failures

A,2e-05 B,7e-01 C,2e-06 D,8e-05 E,4e-03 F,8e-02 G,4e-03 H,4e-04  
I,1e-02 J,2e-05 K,6e-04 L,3e-06 M,1e-04 N,1e-05 O,7e-05 P,8e-06  
Q,7e-06 R,5e-02 S,9e-03 T,9e-05 U,9e-03 V,3e-02

- Case 19

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E, F, G, H

GATE2, OR, I, J, K, L, M, N, O, P

GATE3, OR, Q, R, S, T, U, V, W

Probability of Component Failures

A,8e-04 B,7e-04 C,2e-03 D,2e-06 E,3e-03 F,9e-01 G,4e-04 H,2e-06

I,4e-03 J,1e-05 K,3e-03 L,2e-03 M,2e-05 N,9e-07 O,1e-05 P,3e-04

Q,4e-05 R,6e-03 S,5e-03 T,9e-02 U,3e-04 V,5e-01 W,5e-05

- Case 20

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E, F, G, H

GATE2, OR, I, J, K, L, M, N, O, P

GATE3, OR, Q, R, S, T, U, V, W, X

Probability of Component Failures

A,1e-05 B,7e-01 C,8e-04 D,4e-06 E,8e-03 F,7e-04 G,5e-01 H,1e-01  
I,7e-05 J,6e-06 K,6e-06 L,3e-04 M,6e-03 N,1e-03 O,9e-04 P,3e-01  
Q,3e-04 R,4e-06 S,1e-02 T,3e-02 U,2e-02 V,3e-05 W,8e-01 X,4e-02

- Case 21

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
GATE1, OR, A, B, C, D, E, F, G, H, I  
GATE2, OR, J, K, L, M, N, O, P, Q  
GATE3, OR, R, S, T, U, V, W, X, Y

Probability of Component Failures

A,1e-03 B,1e-02 C,5e-03 D,2e-05 E,5e-01 F,3e-06 G,4e-01 H,2e-04  
I,9e-04 J,4e-02 K,4e-07 L,9e-04 M,2e-04 N,9e-03 O,2e-06 P,6e-02  
Q,6e-06 R,2e-01 S,4e-04 T,2e-01 U,7e-02 V,2e-03 W,6e-03 X,5e-01  
Y,9e-01

- Case 22

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E, F, G, H, I

GATE2, OR, J, K, L, M, N, O, P, Q, R

GATE3, OR, S, T, U, V, W, X, Y, Z

Probability of Component Failures

A,2e-04 B,4e-03 C,8e-01 D,3e-04 E,4e-06 F,8e-02 G,2e-01 H,4e-03

I,9e-05 J,2e-01 K,9e-02 L,5e-05 M,9e-05 N,7e-03 O,8e-04 P,4e-01

Q,9e-06 R,9e-01 S,7e-01 T,3e-04 U,2e-02 V,9e-06 W,3e-01 X,8e-07

Y,9e-05 Z,3e-02

- Case 23

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E, F, G, H, I

GATE2, OR, J, K, L, M, N, O, P, Q, R

GATE3, OR, S, T, U, V, W, X, Y, Z, BA

Probability of Component Failures

A,3e-05	B,2e-04	C,5e-05	D,7e-05	E,1e+00	F,2e-03	G,1e-03	H,6e-02
I,1e-01	J,6e-06	K,4e-03	L,3e-02	M,4e-05	N,2e-03	O,8e-03	P,8e-05
Q,5e-04	R,5e-04	S,6e-05	T,8e-02	U,8e-01	V,9e-04	W,2e-04	X,9e-06
Y,3e-03	Z,8e-02	BA,5e-04					

- Case 24

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3
GATE1, OR, A, B, C, D, E, F, G, H, I, J
GATE2, OR, K, L, M, N, O, P, Q, R, S
GATE3, OR, T, U, V, W, X, Y, Z, BA, BB

Probability of Component Failures

A,9e-05	B,3e-06	C,6e-05	D,5e-05	E,7e-05	F,4e-05	G,3e-06	H,2e-01
I,9e-01	J,8e-01	K,7e-01	L,8e-02	M,7e-01	N,8e-05	O,6e-04	P,9e-05
Q,8e-02	R,9e-05	S,7e-04	T,3e-06	U,2e-01	V,7e-02	W,5e-05	X,9e-06
Y,1e-05	Z,8e-04	BA,6e-02	BB,2e-03				

- Case 25

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
GATE1, OR, A, B, C, D, E, F, G, H, I, J  
GATE2, OR, K, L, M, N, O, P, Q, R, S, T  
GATE3, OR, U, V, W, X, Y, Z, BA, BB, BC

#### Probability of Component Failures

A,1e-04 B,9e-04 C,2e-05 D,2e-02 E,8e-04 F,5e-03 G,9e-05 H,7e-02  
I,6e-05 J,9e-02 K,8e-02 L,4e-05 M,1e-05 N,6e-07 O,7e-03 P,8e-05  
Q,3e-06 R,6e-03 S,2e-03 T,1e-05 U,3e-04 V,6e-06 W,5e-02 X,8e-05  
Y,3e-06 Z,2e-01 BA,3e-01 BB,6e-02 BC,8e-04

- Case 26

#### Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
GATE1, OR, A, B, C, D, E, F, G, H, I, J  
GATE2, OR, K, L, M, N, O, P, Q, R, S, T  
GATE3, OR, U, V, W, X, Y, Z, BA, BB, BC, BD

#### Probability of Component Failures

A,6e-03 B,5e-02 C,6e-04 D,3e-03 E,4e-06 F,3e-06 G,5e-06 H,5e-02  
I,5e-02 J,4e-01 K,7e-01 L,2e-06 M,2e-03 N,2e-01 O,3e-03 P,6e-05  
Q,9e-01 R,8e-03 S,5e-05 T,8e-02 U,8e-05 V,7e-06 W,7e-02 X,1e-04  
Y,3e-03 Z,3e-05 BA,6e-06 BB,9e-04 BC,3e-06 BD,5e-05

- Case 27

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
GATE1, OR, A, B, C, D, E, F, G, H, I, J, K  
GATE2, OR, L, M, N, O, P, Q, R, S, T, U  
GATE3, OR, V, W, X, Y, Z, BA, BB, BC, BD, BE

Probability of Component Failures

A,8e-06 B,9e-06 C,2e-04 D,9e-05 E,9e-06 F,4e-03 G,5e-03 H,8e-01  
I,9e-02 J,7e-06 K,2e-01 L,4e-05 M,7e-06 N,7e-03 O,5e-05 P,4e-02  
Q,3e-03 R,1e-02 S,5e-04 T,9e-04 U,5e-02 V,1e-03 W,9e-06 X,9e-05  
Y,1e-06 Z,8e-01 BA,1e-02 BB,6e-01 BC,1e-04 BD,1e-02 BE,3e-05

- Case 28

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
GATE1, OR, A, B, C, D, E, F, G, H, I, J, K  
GATE2, OR, L, M, N, O, P, Q, R, S, T, U, V  
GATE3, OR, W, X, Y, Z, BA, BB, BC, BD, BE, BF

#### Probability of Component Failures

A,8e-02 B,8e-01 C,5e-04 D,1e-02 E,6e-03 F,9e-05 G,7e-01 H,1e-01  
I,4e-05 J,3e-03 K,6e-05 L,3e-05 M,4e-07 N,6e-06 O,9e-03 P,1e-05  
Q,6e-02 R,8e-02 S,1e-02 T,5e-05 U,5e-05 V,8e-02 W,7e-03 X,4e-06  
Y,6e-06 Z,7e-06 BA,3e-04 BB,1e-06 BC,5e-04 BD,6e-01 BE,2e-02  
BF,1e-04

- Case 29

#### Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
GATE1, OR, A, B, C, D, E, F, G, H, I, J, K  
GATE2, OR, L, M, N, O, P, Q, R, S, T, U, V  
GATE3, OR, W, X, Y, Z, BA, BB, BC, BD, BE, BF, BG

#### Probability of Component Failures

A,2e-05 B,5e-01 C,5e-02 D,4e-04 E,3e-05 F,1e-04 G,4e-05 H,7e-04  
 I,3e-01 J,1e-02 K,9e-04 L,7e-02 M,2e-05 N,7e-02 O,3e-01 P,9e-02  
 Q,1e-01 R,3e-03 S,1e-01 T,9e-05 U,8e-03 V,3e-01 W,2e-03 X,8e-01  
 Y,6e-05 Z,9e-05 BA,4e-02 BB,1e-03 BC,7e-01 BD,4e-04 BE,6e-06  
 BF,1e-06 BG,2e-03

- Case 30

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
 GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L  
 GATE2, OR, M, N, O, P, Q, R, S, T, U, V, W  
 GATE3, OR, X, Y, Z, BA, BB, BC, BD, BE, BF, BG, BH

Probability of Component Failures

A,4e-06 B,4e-01 C,8e-04 D,2e-04 E,8e-01 F,2e-06 G,4e-04 H,2e-05  
 I,1e+00 J,9e-03 K,8e-06 L,9e-02 M,3e-04 N,1e-05 O,8e-07 P,4e-03  
 Q,6e-04 R,3e-02 S,1e-02 T,4e-06 U,2e-04 V,3e-03 W,5e-05 X,2e-04  
 Y,6e-06 Z,1e-02 BA,4e-02 BB,6e-03 BC,7e-01 BD,4e-01 BE,7e-04  
 BF,4e-05 BG,1e-04 BH,3e-05

- Case 31

### Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L  
GATE2, OR, M, N, O, P, Q, R, S, T, U, V, W, X  
GATE3, OR, Y, Z, BA, BB, BC, BD, BE, BF, BG, BH, BI

### Probability of Component Failures

A,5e-06 B,5e-05 C,4e-01 D,7e-02 E,4e-01 F,6e-04 G,9e-01 H,3e-02  
I,9e-04 J,1e-04 K,1e-04 L,4e-04 M,5e-02 N,9e-03 O,5e-03 P,8e-04  
Q,4e-05 R,7e-01 S,8e-03 T,6e-06 U,1e-05 V,1e-06 W,2e-03 X,4e-04  
Y,6e-06 Z,6e-03 BA,7e-04 BB,2e-03 BC,6e-04 BD,1e-01 BE,5e-05  
BF,6e-06 BG,6e-03 BH,5e-04 BI,5e-03

- Case 32

### Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L  
GATE2, OR, M, N, O, P, Q, R, S, T, U, V, W, X  
GATE3, OR, Y, Z, BA, BB, BC, BD, BE, BF, BG, BH, BI, BJ

### Probability of Component Failures

A,1e-05 B,7e-02 C,4e-01 D,3e-06 E,4e-06 F,3e-06 G,8e-06 H,3e-03  
 I,8e-06 J,5e-01 K,2e-01 L,4e-04 M,8e-06 N,4e-03 O,7e-02 P,2e-05  
 Q,8e-01 R,3e-03 S,1e-05 T,4e-04 U,7e-06 V,7e-04 W,5e-03 X,6e-04  
 Y,6e-04 Z,5e-04 BA,5e-01 BB,6e-04 BC,4e-01 BD,4e-01 BE,4e-04  
 BF,8e-06 BG,6e-05 BH,6e-04 BI,7e-06 BJ,2e-06

- Case 33

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
 GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M  
 GATE2, OR, N, O, P, Q, R, S, T, U, V, W, X, Y  
 GATE3, OR, Z, BA, BB, BC, BD, BE, BF, BG, BH, BI, BJ, BK

Probability of Component Failures

A,7e-05 B,9e-03 C,9e-05 D,1e-06 E,6e-03 F,9e-03 G,4e-01 H,3e-01  
 I,1e-03 J,3e-03 K,1e-03 L,2e-07 M,4e-04 N,6e-02 O,4e-01 P,9e-01  
 Q,8e-01 R,4e-04 S,3e-04 T,5e-04 U,5e-04 V,2e-01 W,6e-04 X,6e-01  
 Y,7e-03 Z,3e-05 BA,7e-06 BB,2e-04 BC,7e-05 BD,6e-05 BE,2e-05  
 BF,2e-06 BG,1e-04 BH,7e-01 BI,8e-05 BJ,4e-01 BK,1e-03

- Case 34

### Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M

GATE2, OR, N, O, P, Q, R, S, T, U, V, W, X, Y, Z

GATE3, OR, BA, BB, BC, BD, BE, BF, BG, BH, BI, BJ, BK, BL

### Probability of Component Failures

A,6e-06 B,2e-05 C,5e-04 D,6e-02 E,4e-02 F,4e-04 G,5e-06 H,9e-06

I,8e-06 J,3e-06 K,7e-03 L,2e-03 M,3e-06 N,9e-04 O,5e-02 P,1e-01

Q,3e-01 R,3e-01 S,6e-06 T,6e-06 U,4e-04 V,3e-02 W,7e-06 X,2e-01

Y,8e-05 Z,5e-03 BA,2e-06 BB,9e-01 BC,8e-04 BD,5e-01 BE,3e-03

BF,3e-02 BG,5e-01 BH,1e-01 BI,2e-04 BJ,8e-03 BK,7e-05 BL,5e-03

- Case 35

### Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M

GATE2, OR, N, O, P, Q, R, S, T, U, V, W, X, Y, Z

GATE3, OR, BA, BB, BC, BD, BE, BF, BG, BH, BI, BJ, BK, BL, BM

### Probability of Component Failures

A,3e-05 B,8e-04 C,9e-05 D,1e-03 E,4e-02 F,6e-03 G,5e-04 H,4e-05  
 I,2e-06 J,8e-06 K,8e-06 L,7e-06 M,3e-06 N,5e-03 O,1e+00 P,4e-03  
 Q,7e-06 R,2e-04 S,9e-01 T,7e-06 U,8e-02 V,6e-03 W,2e-01 X,2e-05  
 Y,7e-02 Z,3e-02 BA,3e-06 BB,3e-07 BC,2e-04 BD,6e-06 BE,1e+00  
 BF,5e-03 BG,6e-03 BH,8e-02 BI,9e-05 BJ,3e-03 BK,8e-02 BL,1e-01  
 BM,9e-06

- Case 36

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
 GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M, N  
 GATE2, OR, O, P, Q, R, S, T, U, V, W, X, Y, Z, BA  
 GATE3, OR, BB, BC, BD, BE, BF, BG, BH, BI, BJ, BK, BL, BM, BN

Probability of Component Failures

A,4e-06 B,6e-06 C,4e-04 D,2e-03 E,5e-05 F,1e-01 G,1e-05 H,6e-06  
 I,4e-05 J,9e-04 K,7e-06 L,1e-06 M,5e-06 N,8e-04 O,8e-03 P,2e-03  
 Q,8e-02 R,1e+00 S,6e-05 T,1e-01 U,8e-03 V,8e-05 W,7e-04 X,3e-03  
 Y,5e-03 Z,1e-06 BA,8e-04 BB,6e-04 BC,1e-05 BD,3e-06 BE,5e-03  
 BF,3e-03 BG,7e-04 BH,4e-05 BI,7e-05 BJ,4e-03 BK,5e-06 BL,4e-01  
 BM,2e-06 BN,4e-02

- Case 37

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M, N

GATE2, OR, O, P, Q, R, S, T, U, V, W, X, Y, Z, BA, BB

GATE3, OR, BC, BD, BE, BF, BG, BH, BI, BJ, BK, BL, BM, BN, BO

Probability of Component Failures

A,7e-04 B,3e-02 C,8e-04 D,3e-05 E,9e-06 F,9e-05 G,9e-03 H,3e-03

I,8e-05 J,3e-02 K,9e-04 L,8e-02 M,1e+00 N,9e-02 O,4e-02 P,7e-02

Q,7e-01 R,8e-04 S,9e-01 T,1e-03 U,4e-03 V,1e-04 W,3e-05 X,8e-06

Y,2e-07 Z,7e-01 BA,8e-04 BB,6e-06 BC,6e-05 BD,1e-03 BE,1e-02

BF,6e-06 BG,6e-05 BH,5e-05 BI,5e-03 BJ,5e-03 BK,9e-02 BL,8e-04

BM,3e-02 BN,6e-03 BO,5e-03

- Case 38

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M, N

GATE2, OR, O, P, Q, R, S, T, U, V, W, X, Y, Z, BA, BB

GATE3, OR, BC, BD, BE, BF, BG, BH, BI, BJ, BK, BL, BM, BN, BO, BP

## Probability of Component Failures

A,4e-04 B,7e-03 C,5e-06 D,1e-05 E,1e+00 F,1e-01 G,7e-06 H,2e-02  
I,8e-01 J,5e-06 K,3e-05 L,2e-05 M,9e-01 N,1e-02 O,5e-01 P,4e-03  
Q,6e-01 R,5e-07 S,7e-03 T,5e-04 U,7e-04 V,6e-05 W,5e-05 X,3e-03  
Y,9e-07 Z,5e-03 BA,8e-04 BB,7e-05 BC,3e-01 BD,5e-02 BE,5e-02  
BF,1e-05 BG,4e-05 BH,7e-05 BI,5e-03 BJ,3e-01 BK,8e-04 BL,7e-03  
BM,8e-04 BN,5e-02 BO,6e-04 BP,9e-05

- Case 39

### Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O  
GATE2, OR, P, Q, R, S, T, U, V, W, X, Y, Z, BA, BB, BC  
GATE3, OR, BD, BE, BF, BG, BH, BI, BJ, BK, BL, BM, BN, BO, BP, BQ

## Probability of Component Failures

A,3e-04 B,1e-02 C,5e-03 D,7e-05 E,2e-05 F,8e-06 G,7e-03 H,1e-06  
I,1e-06 J,6e-05 K,4e-07 L,7e-03 M,6e-06 N,9e-02 O,6e-06 P,5e-01  
Q,8e-06 R,4e-05 S,9e-06 T,9e-02 U,5e-04 V,8e-02 W,4e-03 X,4e-06  
Y,7e-01 Z,9e-03 BA,9e-04 BB,4e-05 BC,7e-04 BD,9e-02 BE,3e-04  
BF,6e-03 BG,4e-06 BH,3e-06 BI,8e-04 BJ,2e-04 BK,8e-03 BL,1e-03

BM,8e-01 BN,2e-03 BO,5e-06 BP,3e-01 BQ,6e-02

- Case 40

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O

GATE2, OR, P, Q, R, S, T, U, V, W, X, Y, Z, BA, BB, BC, BD

GATE3, OR, BE, BF, BG, BH, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR

Probability of Component Failures

A,3e-01 B,5e-01 C,8e-03 D,5e-05 E,4e-01 F,3e-02 G,3e-05 H,6e-06

I,7e-01 J,7e-03 K,5e-02 L,9e-01 M,4e-05 N,8e-06 O,4e-03 P,9e-02

Q,9e-02 R,6e-05 S,2e-04 T,5e-06 U,9e-06 V,6e-01 W,4e-06 X,1e-02

Y,2e-03 Z,5e-04 BA,2e-03 BB,3e-06 BC,6e-03 BD,9e-06 BE,5e-05

BF,8e-05 BG,2e-02 BH,2e-03 BI,5e-04 BJ,9e-02 BK,4e-06 BL,1e+00

BM,9e-06 BN,9e-05 BO,9e-03 BP,1e-04 BQ,6e-02 BR,3e-02

- Case 41

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
 GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O  
 GATE2, OR, P, Q, R, S, T, U, V, W, X, Y, Z, BA, BB, BC, BD  
 GATE3, OR, BE, BF, BG, BH, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR,  
 BS

### Probability of Component Failures

A,5e-05 B,2e-03 C,1e-04 D,3e-01 E,7e-03 F,7e-06 G,9e-01 H,7e-05  
 I,1e-04 J,6e-03 K,9e-06 L,6e-02 M,7e-06 N,5e-06 O,1e-05 P,6e-04  
 Q,7e-06 R,6e-02 S,7e-02 T,9e-06 U,1e-03 V,1e-02 W,6e-05 X,6e-03  
 Y,3e-03 Z,6e-03 BA,3e-04 BB,8e-03 BC,6e-02 BD,3e-06 BE,2e-01  
 BF,3e-02 BG,5e-01 BH,5e-04 BI,6e-04 BJ,1e-05 BK,3e-01 BL,4e-01  
 BM,3e-02 BN,3e-06 BO,2e-06 BP,1e-05 BQ,6e-05 BR,6e-02 BS,9e-02

- Case 42

### Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
 GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P  
 GATE2, OR, Q, R, S, T, U, V, W, X, Y, Z, BA, BB, BC, BD, BE  
 GATE3, OR, BF, BG, BH, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS,  
 BT

## Probability of Component Failures

A,5e-04 B,6e-02 C,8e-04 D,1e-02 E,3e-06 F,9e-01 G,9e-07 H,9e-01  
I,4e-06 J,9e-02 K,3e-04 L,4e-02 M,7e-02 N,1e-03 O,1e-05 P,8e-05  
Q,2e-07 R,9e-06 S,6e-04 T,8e-01 U,9e-06 V,7e-06 W,1e-02 X,8e-02  
Y,4e-06 Z,5e-06 BA,3e-01 BB,8e-02 BC,1e-01 BD,5e-03 BE,7e-04  
BF,3e-05 BG,4e-03 BH,1e-06 BI,6e-01 BJ,1e-04 BK,4e-05 BL,2e-04  
BM,7e-06 BN,2e-03 BO,1e-03 BP,8e-03 BQ,7e-05 BR,7e-02 BS,9e-01  
BT,5e-02

- Case 43

### Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P  
GATE2, OR, Q, R, S, T, U, V, W, X, Y, Z, BA, BB, BC, BD, BE, BF  
GATE3, OR, BG, BH, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS, BT,  
BU

## Probability of Component Failures

A,9e-05 B,6e-05 C,6e-03 D,6e-06 E,2e-02 F,3e-01 G,8e-01 H,3e-02  
I,3e-04 J,2e-04 K,3e-01 L,5e-03 M,1e-03 N,1e-03 O,6e-03 P,1e-02  
Q,2e-05 R,3e-02 S,8e-01 T,1e-03 U,3e-02 V,7e-06 W,8e-06 X,2e-03

Y,3e-03 Z,6e-05 BA,1e-01 BB,2e-06 BC,4e-02 BD,9e-01 BE,8e-06  
 BF,9e-02 BG,4e-06 BH,1e-06 BI,1e-04 BJ,5e-04 BK,7e-04 BL,8e-01  
 BM,5e-04 BN,1e-04 BO,5e-06 BP,7e-03 BQ,4e-03 BR,5e-02 BS,9e-03  
 BT,1e-01 BU,2e-05

- Case 44

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
 GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P  
 GATE2, OR, Q, R, S, T, U, V, W, X, Y, Z, BA, BB, BC, BD, BE, BF  
 GATE3, OR, BG, BH, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS, BT,  
 BU, BV

Probability of Component Failures

A,8e-02 B,2e-05 C,9e-06 D,1e-04 E,2e-04 F,6e-02 G,6e-04 H,8e-04  
 I,7e-01 J,3e-04 K,2e-04 L,8e-06 M,8e-05 N,7e-02 O,9e-03 P,4e-06  
 Q,7e-06 R,7e-06 S,8e-03 T,2e-04 U,9e-03 V,5e-01 W,9e-01 X,3e-05  
 Y,2e-06 Z,7e-03 BA,1e+00 BB,4e-07 BC,6e-03 BD,2e-04 BE,9e-03  
 BF,7e-04 BG,4e-05 BH,5e-05 BI,6e-04 BJ,8e-05 BK,4e-05 BL,8e-06  
 BM,2e-05 BN,4e-06 BO,2e-03 BP,1e-04 BQ,7e-04 BR,7e-03 BS,9e-02  
 BT,1e-03 BU,2e-04 BV,2e-05

- Case 45

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q

GATE2, OR, R, S, T, U, V, W, X, Y, Z, BA, BB, BC, BD, BE, BF, BG

GATE3, OR, BH, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS, BT, BU,  
BV, BW

Probability of Component Failures

A,6e-06 B,4e-05 C,3e-06 D,2e-01 E,6e-03 F,9e-05 G,8e-04 H,3e-03  
I,7e-03 J,7e-01 K,9e-01 L,1e-05 M,6e-03 N,6e-02 O,1e-02 P,2e-03  
Q,8e-05 R,3e-06 S,5e-06 T,2e-04 U,2e-04 V,7e-06 W,6e-05 X,9e-02  
Y,3e-02 Z,8e-02 BA,8e-04 BB,9e-03 BC,8e-05 BD,5e-04 BE,7e-03  
BF,3e-01 BG,1e-04 BH,5e-01 BI,1e-05 BJ,8e-03 BK,8e-06 BL,6e-03  
BM,8e-01 BN,5e-04 BO,2e-02 BP,4e-02 BQ,1e-05 BR,2e-05 BS,2e-03  
BT,8e-06 BU,9e-02 BV,5e-06 BW,6e-01

- Case 46

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q

GATE2, OR, R, S, T, U, V, W, X, Y, Z, BA, BB, BC, BD, BE, BF, BG, BH

GATE3, OR, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS, BT, BU, BV,  
BW, BX

### Probability of Component Failures

A,9e-04 B,3e-01 C,5e-06 D,1e-01 E,9e-04 F,7e-06 G,4e-05 H,8e-04

I,6e-05 J,8e-06 K,9e-05 L,2e-05 M,9e-06 N,3e-05 O,8e-04 P,9e-05

Q,3e-06 R,7e-01 S,4e-06 T,6e-07 U,4e-05 V,6e-03 W,6e-03 X,4e-01

Y,1e-02 Z,9e-06 BA,8e-01 BB,1e-07 BC,5e-01 BD,7e-02 BE,4e-05

BF,7e-04 BG,1e-01 BH,5e-03 BI,6e-04 BJ,4e-03 BK,8e-01 BL,8e-05

BM,7e-03 BN,3e-01 BO,1e+00 BP,8e-01 BQ,9e-05 BR,3e-01 BS,8e-04

BT,5e-01 BU,3e-04 BV,1e-03 BW,9e-03 BX,6e-06

- Case 47

### Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3

GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q

GATE2, OR, R, S, T, U, V, W, X, Y, Z, BA, BB, BC, BD, BE, BF, BG, BH

GATE3, OR, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS, BT, BU, BV,

BW, BX, BY

### Probability of Component Failures

A,3e-01 B,3e-03 C,2e-04 D,2e-03 E,8e-02 F,1e-01 G,9e-03 H,7e-01  
I,3e-05 J,5e-02 K,6e-01 L,8e-05 M,7e-04 N,8e-04 O,2e-01 P,1e-03  
Q,4e-05 R,2e-05 S,1e-04 T,4e-04 U,2e-02 V,5e-01 W,6e-02 X,5e-02  
Y,9e-02 Z,3e-04 BA,5e-04 BB,6e-01 BC,3e-03 BD,6e-02 BE,1e-01  
BF,2e-05 BG,5e-06 BH,9e-02 BI,6e-06 BJ,9e-01 BK,8e-02 BL,1e-02  
BM,6e-04 BN,5e-02 BO,4e-06 BP,5e-06 BQ,9e-01 BR,6e-04 BS,1e-01  
BT,2e-01 BU,6e-04 BV,1e-05 BW,2e-04 BX,7e-02 BY,8e-06

- Case 48

### Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R  
GATE2, OR, S, T, U, V, W, X, Y, Z, BA, BB, BC, BD, BE, BF, BG, BH,  
BI  
GATE3, OR, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS, BT, BU, BV,  
BW, BX, BY, BZ

### Probability of Component Failures

A,7e-02 B,9e-01 C,4e-04 D,6e-04 E,5e-06 F,7e-01 G,5e-05 H,7e-01  
 I,1e-05 J,9e-06 K,5e-05 L,8e-06 M,4e-04 N,7e-01 O,2e-03 P,4e-01  
 Q,1e-03 R,4e-07 S,6e-03 T,8e-04 U,8e-05 V,6e-02 W,7e-06 X,8e-06  
 Y,8e-04 Z,1e-02 BA,8e-05 BB,5e-05 BC,9e-06 BD,2e-01 BE,8e-06  
 BF,4e-01 BG,4e-03 BH,5e-04 BI,6e-01 BJ,7e-06 BK,9e-03 BL,7e-06  
 BM,6e-05 BN,2e-01 BO,6e-01 BP,7e-05 BQ,7e-03 BR,6e-04 BS,6e-05  
 BT,4e-04 BU,8e-01 BV,5e-05 BW,6e-04 BX,6e-04 BY,7e-02 BZ,5e-04

- Case 49

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
 GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R  
 GATE2, OR, S, T, U, V, W, X, Y, Z, BA, BB, BC, BD, BE, BF, BG, BH,  
 BI, BJ  
 GATE3, OR, BK, BL, BM, BN, BO, BP, BQ, BR, BS, BT, BU, BV, BW,  
 BX, BY, BZ, CA

Probability of Component Failures

A,9e-06 B,9e-02 C,7e-05 D,1e-04 E,4e-02 F,4e-01 G,3e-02 H,8e-01  
 I,5e-04 J,6e-04 K,2e-04 L,7e-06 M,7e-03 N,4e-02 O,1e-02 P,7e-04  
 Q,8e-06 R,4e-05 S,7e-03 T,9e-05 U,9e-04 V,2e-03 W,6e-04 X,9e-06

Y,1e-05 Z,7e-06 BA,8e-05 BB,8e-02 BC,7e-01 BD,4e-02 BE,9e-04  
 BF,8e-02 BG,1e-01 BH,8e-05 BI,2e-03 BJ,7e-03 BK,9e-04 BL,9e-05  
 BM,7e-07 BN,4e-05 BO,1e-04 BP,2e-05 BQ,6e-06 BR,6e-01 BS,2e-06  
 BT,9e-06 BU,1e-01 BV,2e-04 BW,1e+00 BX,5e-01 BY,3e-06 BZ,2e-06  
 CA,9e-03

- Case 50

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3  
 GATE1, OR, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R  
 GATE2, OR, S, T, U, V, W, X, Y, Z, BA, BB, BC, BD, BE, BF, BG, BH,  
 BI, BJ  
 GATE3, OR, BK, BL, BM, BN, BO, BP, BQ, BR, BS, BT, BU, BV, BW,  
 BX, BY, BZ, CA, CB

Probability of Component Failures

A,7e-04 B,4e-02 C,4e-02 D,2e-05 E,6e-04 F,3e-02 G,8e-02 H,3e-02  
 I,6e-05 J,1e+00 K,7e-05 L,4e-06 M,4e-01 N,9e-05 O,3e-05 P,7e-04  
 Q,5e-06 R,2e-05 S,9e-03 T,8e-05 U,3e-04 V,1e-05 W,9e-05 X,6e-03  
 Y,4e-05 Z,1e-05 BA,1e-06 BB,2e-01 BC,9e-02 BD,6e-04 BE,2e-06  
 BF,4e-06 BG,1e-04 BH,6e-05 BI,8e-04 BJ,6e-02 BK,8e-03 BL,5e-01

BM,7e-05 BN,2e-06 BO,3e-02 BP,2e-01 BQ,3e-04 BR,7e-03 BS,7e-03  
BT,2e-05 BU,6e-06 BV,4e-05 BW,1e+00 BX,8e-04 BY,3e-05 BZ,3e-05  
CA,8e-05 CB,2e-02

### A.3 Data used in section IV.7

#### A.3.1 Cases in Figure IV.14

- Case 1

Fault Tree Structure

GATE0, AND, GATE1, A, B  
GATE1, AND, GATE2, C, D, GATE8  
GATE2, AND, GATE3, E, F, GATE5  
GATE3, AND, GATE4, G, H  
GATE4, AND, GATE5, I, J  
GATE5, AND, GATE6, K, L  
GATE6, AND, GATE7, M, N  
GATE7, OR, GATE8, O, P  
GATE8, OR, GATE9, Q, R  
GATE9, OR, S, T

Probability of Component Failures

A,6e-04	B,8e-03	C,6e-05	D,9e-01	E,1e-01	F,4e-02	G,4e-06	H,1e-04
I,6e-02	J,4e-04	K,6e-06	L,3e-01	M,2e-06	N,2e-02	O,4e-01	P,8e-04
Q,6e-05	R,8e-06	S,8e-04	T,2e-03				

- Case 2

### Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F, G, X
GATE1, AND, GATE2, H, I, J, GATE3, K, L, M, GATE4
GATE2, OR, GATE4, GATE5, N, O, P, GATE6, Q, R
GATE3, OR, GATE7, S, T, U, GATE8, V, W, GATE9
GATE4, OR, X, Y, Z, BA, BB, BC, BD
GATE5, OR, BE, BF, BG, BH, BI, BJ, BK
GATE6, OR, BL, BM, BN, BO, BP, BQ, BR
GATE7, OR, BS, BT, BU, BV, BW, BX, BY
GATE8, AND, BZ, CA, CB, CC, CD, CE, CF
GATE9, OR, CG, CH, CI, CJ, CK, CL, CM

### Probability of Component Failures

A,2e-05	B,3e-03	C,8e-03	D,2e-01	E,7e-04	F,2e-02	G,7e-02	H,3e-05
I,7e-02	J,8e-03	K,5e-03	L,4e-05	M,3e-03	N,7e-02	O,1e-02	P,1e-06
Q,3e-02	R,1e-03	S,5e-05	T,8e-04	U,3e-01	V,8e-01	W,6e-01	X,2e-05

Y,9e-06 Z,7e-06 BA,1e-03 BB,9e-06 BC,5e-06 BD,6e-06 BE,6e-06  
 BF,3e-06 BG,2e-02 BH,1e+00 BI,4e-03 BJ,2e-05 BK,1e-04 BL,9e-05  
 BM,2e-04 BN,4e-01 BO,6e-06 BP,3e-06 BQ,6e-03 BR,2e-06 BS,5e-05  
 BT,2e-03 BU,5e-03 BV,8e-03 BW,8e-02 BX,9e-06 BY,1e-06 BZ,9e-04  
 CA,4e-03 CB,1e-06 CC,9e-02 CD,9e-03 CE,1e-03 CF,9e-06 CG,5e-03  
 CH,1e-01 CI,8e-02 CJ,9e-02 CK,4e-05 CL,9e-01 CM,8e-05

- Case 3

Fault Tree Structure

GATE0, OR, GATE1, A, B, C, D, E, F, G, H, I, J, K, L, EA  
 GATE1, AND, GATE2, M, GATE3, N, O, P, Q, R, S, T, U, V, W, GATE4  
 GATE2, OR, GATE4, X, Y, Z, BA, BB, BC, BD, BE, BF, BG, GATE5, BH  
 GATE3, AND, GATE6, BI, BJ, BK, BL, BM, BN, BO, BP, BQ, BR, BS,  
 GATE7  
 GATE4, OR, GATE8, BT, GATE9, BU, BV, BW, BX, BY, BZ, CA, CB,  
 CC, CD  
 GATE5, AND, CE, CF, CG, CH, CI, CJ, CK, CL, CM, CN, CO, CP  
 GATE6, AND, CQ, CR, CS, CT, CU, CV, CW, CX, CY, CZ, DA, DB  
 GATE7, AND, DC, DD, DE, DF, DG, DH, DI, DJ, DK, DL, DM, DN  
 GATE8, AND, DO, DP, DQ, DR, DS, DT, DU, DV, DW, DX, DY, DZ  
 GATE9, AND, EA, EB, EC, ED, EE, EF, EG, EH, EI, EJ, EK, EL

## Probability of Component Failures

A,6e-03 B,4e-06 C,3e-03 D,8e-03 E,2e-01 F,2e-06 G,9e-03 H,1e-02  
I,2e-01 J,5e-01 K,1e-06 L,8e-04 M,4e-04 N,2e-05 O,8e-05 P,3e-06  
Q,3e-05 R,6e-04 S,1e-02 T,9e-04 U,8e-05 V,6e-05 W,6e-03 X,7e-06  
Y,4e-06 Z,7e-02 BA,2e-01 BB,2e-06 BC,1e-02 BD,8e-06 BE,6e-02  
BF,8e-02 BG,9e-05 BH,8e-01 BI,7e-03 BJ,5e-02 BK,1e-04 BL,4e-02  
BM,4e-03 BN,5e-06 BO,2e-01 BP,7e-02 BQ,4e-05 BR,3e-02 BS,2e-03  
BT,2e-03 BU,8e-02 BV,9e-03 BW,6e-06 BX,4e-02 BY,7e-04 BZ,9e-03  
CA,5e-04 CB,9e-02 CC,8e-02 CD,4e-01 CE,9e-05 CF,6e-06 CG,9e-06  
CH,8e-01 CI,1e-04 CJ,2e-04 CK,5e-05 CL,4e-03 CM,8e-03 CN,6e-06  
CO,1e-01 CP,4e-05 CQ,1e-01 CR,7e-02 CS,9e-04 CT,7e-01 CU,2e-05  
CV,4e-02 CW,2e-01 CX,4e-04 CY,9e-05 CZ,2e-01 DA,4e-03 DB,9e-05  
DC,4e-06 DD,5e-01 DE,6e-03 DF,2e-05 DG,8e-06 DH,4e-04 DI,9e-03  
DJ,7e-06 DK,3e-03 DL,1e-05 DM,2e-02 DN,7e-02 DO,6e-02 DP,1e-03  
DQ,7e-03 DR,7e-03 DS,5e-05 DT,7e-07 DU,8e-03 DV,5e-03 DW,8e-03  
DX,6e-03 DY,4e-03 DZ,5e-06 EA,1e-01 EB,9e-06 EC,3e-03 ED,3e-02  
EE,9e-05 EF,7e-03 EG,8e-06 EH,8e-03 EI,2e-04 EJ,3e-03 EK,1e-02  
EL,2e-01

- Case 4

### Fault Tree Structure

GATE0, AND, GATE1, A, B, C, D, E, F, G, GATE2, H, I, J, K, L, M, N, O, P

GATE1, OR, GATE3, Q, R, S, T, U, GATE4, V, W, X, Y, Z, BA, BB, BC, BD, BE, BF

GATE2, OR, GATE5, BG, BH, BI, BJ, BK, BL, GATE6, BM, BN, BO, BP, BQ, BR, BS, GATE7, BT, BU

GATE3, AND, GATE8, BV, BW, BX, BY, BZ, CA, CB, CC, GATE9, CD, CE, CF, CG, CH, CI, CJ, CK

GATE4, AND, CL, CM, CN, CO, CP, CQ, CR, CS, CT, CU, CV, CW, CX, CY, CZ, DA, DB, DC

GATE5, OR, DC, DD, DE, DF, DG, DH, DI, DJ, DK, DL, DM, DN, DO, DP, DQ, DR, DS

GATE6, OR, DT, DU, DV, DW, DX, DY, DZ, EA, EB, EC, ED, EE, EF, EG, EH, EI, EJ

GATE7, AND, EK, EL, EM, EN, EO, EP, EQ, ER, ES, ET, EU, EV, EW, EX, EY, EZ, FA

GATE8, AND, FB, FC, FD, FE, FF, FG, FH, FI, FJ, FK, FL, FM, FN, FO, FP, FQ, FR

GATE9, AND, FS, FT, FU, FV, FW, FX, FY, FZ, GA, GB, GC, GD, GE, GF, GG, GH, GI

Probability of Component Failures

A,6e-02 B,6e-05 C,5e-03 D,4e-02 E,3e-02 F,2e-06 G,9e-04 H,6e-02  
I,1e-05 J,2e-02 K,1e-01 L,8e-06 M,4e-06 N,2e-02 O,9e-04 P,8e-02  
Q,1e-02 R,1e-04 S,6e-02 T,3e-06 U,4e-07 V,7e-05 W,8e-01 X,5e-06  
Y,5e-06 Z,2e-06 BA,7e-06 BB,5e-01 BC,8e-05 BD,4e-04 BE,2e-04  
BF,4e-04 BG,7e-01 BH,7e-01 BI,1e-06 BJ,3e-03 BK,4e-04 BL,4e-05  
BM,9e-04 BN,5e-05 BO,1e-03 BP,9e-01 BQ,2e-05 BR,8e-06 BS,3e-02  
BT,9e-02 BU,7e-05 BV,7e-05 BW,9e-06 BX,1e-05 BY,5e-06 BZ,5e-04  
CA,3e-01 CB,8e-04 CC,5e-05 CD,3e-05 CE,4e-03 CF,6e-04 CG,6e-01  
CH,2e-02 CI,3e-03 CJ,2e-03 CK,8e-03 CL,7e-01 CM,7e-05 CN,1e-02  
CO,9e-02 CP,7e-06 CQ,6e-02 CR,2e-04 CS,5e-03 CT,3e-05 CU,5e-05  
CV,5e-06 CW,8e-04 CX,8e-05 CY,9e-01 CZ,2e-02 DA,1e-04 DB,6e-06  
DC,4e-03 DD,9e-06 DE,3e-01 DF,8e-04 DG,3e-01 DH,2e-02 DI,4e-04  
DJ,9e-04 DK,4e-03 DL,7e-02 DM,5e-02 DN,9e-05 DO,5e-04 DP,9e-04  
DQ,3e-03 DR,9e-05 DS,7e-05 DT,8e-03 DU,4e-01 DV,4e-01 DW,9e-01  
DX,4e-02 DY,3e-06 DZ,4e-04 EA,3e-01 EB,4e-03 EC,8e-01 ED,7e-06  
EE,6e-03 EF,6e-03 EG,9e-06 EH,7e-06 EI,8e-04 EJ,9e-06 EK,3e-04  
EL,4e-01 EM,3e-01 EN,4e-04 EO,5e-02 EP,2e-04 EQ,4e-04 ER,5e-04  
ES,8e-03 ET,4e-02 EU,8e-01 EV,3e-05 EW,6e-03 EX,3e-01 EY,3e-01  
EZ,4e-01 FA,7e-03 FB,4e-06 FC,1e-05 FD,4e-07 FE,1e-05 FF,2e-03  
FG,2e-01 FH,2e-06 FI,6e-05 FJ,3e-03 FK,4e-02 FL,1e-03 FM,2e-02  
FN,5e-06 FO,1e-04 FP,4e-04 FQ,4e-05 FR,2e-01 FS,9e-06 FT,4e-05  
FU,5e-06 FV,9e-02 FW,9e-01 FX,5e-01 FY,9e-02 FZ,4e-04 GA,9e-01  
GB,8e-06 GC,2e-05 GD,7e-02 GE,8e-04 GF,3e-03 GG,9e-01 GH,8e-06  
GI,3e-05

- Case 5

Fault Tree Structure

GATE0, AND, GATE1, GATE2, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, GATE3, S, T, DF

GATE1, AND, GATE4, U, V, W, X, Y, Z, BA, BB, BC, BD, BE, BF, BG, BH, BI, BJ, BK, BL, BM, BN, BO, BP

GATE2, OR, GATE5, BQ, BR, BS, BT, BU, GATE6, BV, BW, BX, BY, BZ, CA, CB, CC, CD, CE, CF, CG, CH, CI, CJ, CK

GATE3, AND, GATE7, CL, CM, CN, GATE8, CO, CP, CQ, CR, CS, CT, CU, CV, CW, CX, CY, GATE9, CZ, DA, DB, DC, DD, DE, GP

GATE4, AND, DF, DG, DH, DI, DJ, DK, DL, DM, DN, DO, DP, DQ, DR, DS, DT, DU, DV, DW, DX, DY, DZ, EA

GATE5, OR, EB, EC, ED, EE, EF, EG, EH, EI, EJ, EK, EL, EM, EN, EO, EP, EQ, ER, ES, ET, EU, EV, EW

GATE6, AND, EX, EY, EZ, FA, FB, FC, FD, FE, FF, FG, FH, FI, FJ, FK, FL, FM, FN, FO, FP, FQ, FR, FS

GATE7, OR, FT, FU, FV, FW, FX, FY, FZ, GA, GB, GC, GD, GE, GF, GG, GH, GI, GJ, GK, GL, GM, GN, GO

GATE8, AND, GP, GQ, GR, GS, GT, GU, GV, GW, GX, GY, GZ, HA, HB, HC, HD, HE, HF, HG, HH, HI, HJ, HK

GATE9, AND, HL, HM, HN, HO, HP, HQ, HR, HS, HT, HU, HV, HW, HX, HY, HZ, IA, IB, IC, ID, IE, IF, IG

## Probability of Component Failures

A,8e-04 B,6e-04 C,5e-02 D,1e-03 E,2e-06 F,2e-06 G,3e-04 H,4e-06  
I,3e-04 J,8e-06 K,1e-02 L,3e-01 M,7e-01 N,3e-04 O,8e-03 P,7e-06  
Q,2e-06 R,2e-06 S,2e-04 T,7e-07 U,9e-03 V,4e-04 W,7e-06 X,4e-02  
Y,4e-04 Z,4e-04 BA,2e-05 BB,8e-06 BC,8e-04 BD,7e-03 BE,6e-03  
BF,2e-05 BG,6e-02 BH,8e-04 BI,6e-05 BJ,3e-01 BK,5e-03 BL,7e-05  
BM,9e-06 BN,9e-05 BO,3e-02 BP,6e-06 BQ,3e-03 BR,4e-04 BS,4e-05  
BT,9e-05 BU,9e-03 BV,1e-03 BW,4e-03 BX,2e-02 BY,8e-05 BZ,4e-06  
CA,3e-02 CB,2e-02 CC,8e-02 CD,3e-02 CE,6e-06 CF,6e-02 CG,9e-04  
CH,7e-03 CI,9e-04 CJ,5e-03 CK,9e-05 CL,1e-03 CM,8e-02 CN,3e-02  
CO,7e-05 CP,8e-02 CQ,1e-04 CR,2e-05 CS,4e-03 CT,2e-04 CU,5e-03  
CV,3e-03 CW,8e-06 CX,8e-06 CY,5e-06 CZ,3e-02 DA,1e-03 DB,8e-01  
DC,5e-03 DD,7e-05 DE,2e-01 DF,1e+00 DG,1e-04 DH,3e-01 DI,4e-05  
DJ,4e-06 DK,4e-05 DL,5e-02 DM,2e-02 DN,8e-05 DO,3e-03 DP,5e-01  
DQ,8e-03 DR,2e-03 DS,8e-03 DT,1e-02 DU,2e-04 DV,1e-02 DW,8e-06  
DX,6e-04 DY,6e-04 DZ,1e-03 EA,1e-04 EB,2e-03 EC,7e-02 ED,4e-01  
EE,8e-01 EF,3e-06 EG,4e-02 EH,4e-06 EI,3e-06 EJ,5e-02 EK,9e-03  
EL,8e-07 EM,4e-02 EN,5e-04 EO,4e-02 EP,4e-03 EQ,1e-05 ER,5e-06  
ES,5e-01 ET,3e-06 EU,9e-04 EV,8e-02 EW,6e-02 EX,2e-05 EY,9e-06  
EZ,3e-04 FA,6e-03 FB,4e-04 FC,8e-03 FD,9e-06 FE,2e-03 FF,1e-04  
FG,5e-04 FH,3e-02 FI,1e-01 FJ,4e-04 FK,4e-02 FL,8e-03 FM,7e-06  
FN,5e-02 FO,9e-03 FP,4e-03 FQ,4e-05 FR,3e-04 FS,4e-04 FT,4e-03  
FU,7e-03 FV,5e-06 FW,3e-07 FX,7e-07 FY,4e-04 FZ,9e-03 GA,1e-01  
GB,7e-03 GC,3e-04 GD,2e-06 GE,4e-06 GF,7e-02 GG,2e-04 GH,7e-05

GI,2e-04 GJ,4e-07 GK,4e-02 GL,9e-06 GM,9e-01 GN,7e-05 GO,4e-04  
 GP,3e-03 GQ,7e-03 GR,3e-02 GS,8e-01 GT,7e-02 GU,8e-03 GV,9e-05  
 GW,9e-06 GX,9e-05 GY,1e-04 GZ,2e-04 HA,1e-04 HB,4e-05 HC,6e-03  
 HD,7e-02 HE,2e-02 HF,5e-05 HG,7e-02 HH,7e-05 HI,3e-06 HJ,8e-02  
 HK,7e-06 HL,8e-06 HM,3e-04 HN,4e-05 HO,5e-02 HP,9e-04 HQ,7e-05  
 HR,7e-04 HS,6e-01 HT,7e-01 HU,7e-06 HV,1e-02 HW,7e-04 HX,3e-  
 04 HY,1e-02 HZ,7e-02 IA,1e-04 IB,7e-02 IC,5e-03 ID,6e-05 IE,5e-06  
 IF,6e-02 IG,8e-07

### A.3.2 Cases in Figure IV.15

The fault tree structure for all the following cases in this section is same. However, the basic events associated with the common cause failure group changes in each of the case. Hence an updated fault tree structure after applying beta parameter method is shown for each case and only one probability file is provided at the end to display the point estimates of probabilities for each basic event.

- Case 1

Fault Tree Structure

GATE0, OR, GATE1, GATE2, GATE3, GATE4, GATE5  
 GATE1, AND, A, GATE6  
 GATE2, OR, GATE7, G  
 GATE3, OR, GATE8, B

GATE4, AND, GATE9, L

GATE5, OR, GATE10, GATE12,  $R_G^1$

GATE6, AND, C,  $D_G^1$

GATE7, OR,  $E_G^1$ , F

GATE8, AND,  $H_G^1$ , I

GATE9, OR,  $J_G^1$ , K

GATE10, OR, GATE11, O

GATE11, AND, M, N

GATE12, AND, P, Q

$E_G^1$ , OR,  $E^1$ ,  $DE^1$ ,  $EH^1$ ,  $EJ^1$ ,  $ER^1$ ,  $DEH^1$ ,  $DEJ^1$ ,  $DER^1$ ,  $EHJ^1$ ,  $EHR^1$ ,  
 $EJR^1$ ,  $DEHJ^1$ ,  $DEHR^1$ ,  $DEJR^1$ ,  $EHJR^1$ ,  $DEHJR^1$

$R_G^1$ , OR,  $R^1$ ,  $DR^1$ ,  $ER^1$ ,  $HR^1$ ,  $JR^1$ ,  $DER^1$ ,  $DHR^1$ ,  $DJR^1$ ,  $EHR^1$ ,  $EJR^1$ ,  
 $HJR^1$ ,  $DEHR^1$ ,  $DEJR^1$ ,  $DHJR^1$ ,  $EHJR^1$ ,  $DEHJR^1$

$H_G^1$ , OR,  $H^1$ ,  $DH^1$ ,  $EH^1$ ,  $HJ^1$ ,  $HR^1$ ,  $DEH^1$ ,  $DHJ^1$ ,  $DHR^1$ ,  $EHJ^1$ ,  $EHR^1$ ,  
 $HJR^1$ ,  $DEHJ^1$ ,  $DEHR^1$ ,  $DHJR^1$ ,  $EHJR^1$ ,  $DEHJR^1$

$J_G^1$ , OR,  $J^1$ ,  $DJ^1$ ,  $EJ^1$ ,  $HJ^1$ ,  $JR^1$ ,  $DEJ^1$ ,  $DHJ^1$ ,  $DJR^1$ ,  $EHJ^1$ ,  $EJR^1$ ,  
 $HJR^1$ ,  $DEHJ^1$ ,  $DEJR^1$ ,  $DHJR^1$ ,  $EHJR^1$ ,  $DEHJR^1$

$D_G^1$ , OR,  $D^1$ ,  $DE^1$ ,  $DH^1$ ,  $DJ^1$ ,  $DR^1$ ,  $DEH^1$ ,  $DEJ^1$ ,  $DER^1$ ,  $DHJ^1$ ,  $DHR^1$ ,  
 $DJR^1$ ,  $DEHJ^1$ ,  $DEHR^1$ ,  $DEJR^1$ ,  $DHJR^1$ ,  $DEHJR^1$

where, a superscript denotes a failure associated with common cause failure group and subscript  $G$  denotes a gate name.

- Case 2

Fault Tree Structure

GATE0, OR, GATE1, GATE2, GATE3, GATE4, GATE5

GATE1, OR, A, GATE6

GATE2, AND, GATE7, G

GATE3, AND, GATE8, B

GATE4, OR, GATE9,  $L_G^2$

GATE5, OR, GATE10, GATE12, R

GATE6, AND,  $C_G^2$ , D

GATE7, OR, E,  $F_G^2$

GATE8, OR, H,  $I_G^2$

GATE9, OR, J, K

GATE10, AND, GATE11, O

GATE11, AND, M, N

GATE12, AND, P, Q

$L_G^2$ , OR,  $L^2$ ,  $CL^2$ ,  $FL^2$ ,  $IL^2$ ,  $CFL^2$ ,  $CIL^2$ ,  $FIL^2$ ,  $CFIL^2$

$C_G^2$ , OR,  $C^2$ ,  $CF^2$ ,  $CI^2$ ,  $CL^2$ ,  $CFI^2$ ,  $CFL^2$ ,  $CIL^2$ ,  $CFIL^2$

$F_G^2$ , OR,  $F^2$ ,  $CF^2$ ,  $FI^2$ ,  $FL^2$ ,  $CFI^2$ ,  $CFL^2$ ,  $FIL^2$ ,  $CFIL^2$

$I_G^2$ , OR,  $I^2$ ,  $CI^2$ ,  $FI^2$ ,  $IL^2$ ,  $CFI^2$ ,  $CIL^2$ ,  $FIL^2$ ,  $CFIL^2$

- Case 3

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3, GATE4, GATE5

GATE1, OR,  $A_G^3$ , GATE6

GATE2, AND, GATE7, G

GATE3, OR, GATE8, B  
 GATE4, OR, GATE9, L  
 GATE5, OR, GATE10, GATE12, R  
 GATE6, AND, C, D  
 GATE7, OR, E, F  
 GATE8, AND, H, I  
 GATE9, OR, J,  $K_G^3$   
 GATE10, AND, GATE11, O  
 GATE11, OR, M,  $N_G^3$   
 GATE12, AND, P, Q  
 $A_G^3$ , OR,  $A^3$ ,  $NA^3$ ,  $KA^3$ ,  $NKA^3$   
 $K_G^3$ , OR,  $K^3$ ,  $NK^3$ ,  $KA^3$ ,  $NKA^3$   
 $N_G^3$ , OR,  $N^3$ ,  $NK^3$ ,  $NA^3$ ,  $NKA^3$

- Case 4

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3, GATE4, GATE5  
 GATE1, OR, A, GATE6  
 GATE2, OR, GATE7, G  
 GATE3, OR, GATE8, B  
 GATE4, AND, GATE9, L  
 GATE5, AND, GATE10, GATE12, R  
 GATE6, AND, C,  $D_G^4$

GATE7, OR, E, F  
 GATE8, OR, H, I  
 GATE9, AND, J, K  
 GATE10, AND, GATE11, O  
 GATE11, OR,  $M_G^4$ , N  
 GATE12, AND, P, Q  
 $M_G^4$ , OR,  $M^4$ ,  $MD^4$   
 $D_G^4$ , OR,  $D^4$ ,  $MD^4$

- Case 5

Fault Tree Structure

GATE0, AND, GATE1, GATE2, GATE3, GATE4, GATE5,  $S_G^5$   
 GATE1, OR,  $A_G^5$ , GATE6  
 GATE2, AND, GATE7, G  
 GATE3, OR, GATE8, B  
 GATE4, AND, GATE9,  $L_G^5$   
 GATE5, OR, GATE10, GATE12, R  
 GATE6, AND, C, D  
 GATE7, OR,  $E_G^5$ , F  
 GATE8, AND,  $H_G^5$ , I  
 GATE9, AND, J, K  
 GATE10, AND, GATE11,  $O_G^5$   
 GATE11, OR, M, N

GATE12, OR, P, Q

$L_G^5$ , OR,  $L^5$ ,  $AL^5$ ,  $EL^5$ ,  $HL^5$ ,  $OL^5$ ,  $LS^5$ ,  $AEL^5$ ,  $AHL^5$ ,  $AOL^5$ ,  $ALS^5$ ,  $EHL^5$ ,  
 $EOL^5$ ,  $ELS^5$ ,  $HOL^5$ ,  $HLS^5$ ,  $OLS^5$ ,  $AEHL^5$ ,  $AEOL^5$ ,  $AELS^5$ ,  $AHOL^5$ ,  
 $AHLS^5$ ,  $AOLS^5$ ,  $EHOL^5$ ,  $EHLS^5$ ,  $EOLS^5$ ,  $HOLS^5$ ,  $AEHOL^5$ ,  $AEHLS^5$ ,  
 $AEOLS^5$ ,  $AHOLS^5$ ,  $EHOLS^5$ ,  $AEHOLS^5$

$S_G^5$ , OR,  $S^5$ ,  $AS^5$ ,  $ES^5$ ,  $HS^5$ ,  $OS^5$ ,  $LS^5$ ,  $AES^5$ ,  $AHS^5$ ,  $AOS^5$ ,  $ALS^5$ ,  $EHS^5$ ,  
 $EOS^5$ ,  $ELS^5$ ,  $HOS^5$ ,  $HLS^5$ ,  $OLS^5$ ,  $AEHS^5$ ,  $AEOS^5$ ,  $AELS^5$ ,  $AHOS^5$ ,  
 $AHLS^5$ ,  $AOLS^5$ ,  $EHOS^5$ ,  $EHLS^5$ ,  $EOLS^5$ ,  $HOLS^5$ ,  $AEHOS^5$ ,  $AEHLS^5$ ,  
 $AEOLS^5$ ,  $AHOLS^5$ ,  $EHOLS^5$ ,  $AEHOLS^5$

$A_G^5$ , OR,  $A^5$ ,  $AE^5$ ,  $AH^5$ ,  $AO^5$ ,  $AL^5$ ,  $AS^5$ ,  $AEH^5$ ,  $AEO^5$ ,  $AEL^5$ ,  $AES^5$ ,  
 $AHO^5$ ,  $AHL^5$ ,  $AHS^5$ ,  $AOL^5$ ,  $AOS^5$ ,  $ALS^5$ ,  $AEHO^5$ ,  $AEHL^5$ ,  $AEHS^5$ ,  
 $AEOL^5$ ,  $AEOS^5$ ,  $AELS^5$ ,  $AHOL^5$ ,  $AHOS^5$ ,  $AHLS^5$ ,  $AOLS^5$ ,  $AEHOL^5$ ,  
 $AEHOS^5$ ,  $AEHLS^5$ ,  $AEOLS^5$ ,  $AHOLS^5$ ,  $AEHOLS^5$

$E_G^5$ , OR,  $E^5$ ,  $AE^5$ ,  $EH^5$ ,  $EO^5$ ,  $EL^5$ ,  $ES^5$ ,  $AEH^5$ ,  $AEO^5$ ,  $AEL^5$ ,  $AES^5$ ,  
 $EHO^5$ ,  $EHL^5$ ,  $EHS^5$ ,  $EOL^5$ ,  $EOS^5$ ,  $ELS^5$ ,  $AEHO^5$ ,  $AEHL^5$ ,  $AEHS^5$ ,  
 $AEOL^5$ ,  $AEOS^5$ ,  $AELS^5$ ,  $EHOL^5$ ,  $EHOS^5$ ,  $EHLS^5$ ,  $EOLS^5$ ,  $AEHOL^5$ ,  
 $AEHOS^5$ ,  $AEHLS^5$ ,  $AEOLS^5$ ,  $EHOLS^5$ ,  $AEHOLS^5$

$H_G^5$ , OR,  $H^5$ ,  $AH^5$ ,  $EH^5$ ,  $HO^5$ ,  $HL^5$ ,  $HS^5$ ,  $AEH^5$ ,  $AHO^5$ ,  $AHL^5$ ,  $AHS^5$ ,  
 $EHO^5$ ,  $EHL^5$ ,  $EHS^5$ ,  $HOL^5$ ,  $HOS^5$ ,  $HLS^5$ ,  $AEHO^5$ ,  $AEHL^5$ ,  $AEHS^5$ ,  
 $AHOL^5$ ,  $AHOS^5$ ,  $AHLS^5$ ,  $EHOL^5$ ,  $EHOS^5$ ,  $EHLS^5$ ,  $HOLS^5$ ,  $AEHOL^5$ ,  
 $AEHOS^5$ ,  $AEHLS^5$ ,  $AHOLS^5$ ,  $EHOLS^5$ ,  $AEHOLS^5$

$O_G^5$ , OR,  $O^5$ ,  $AO^5$ ,  $EO^5$ ,  $HO^5$ ,  $OL^5$ ,  $OS^5$ ,  $AEO^5$ ,  $AHO^5$ ,  $AOL^5$ ,  $AOS^5$ ,  
 $EHO^5$ ,  $EOL^5$ ,  $EOS^5$ ,  $HOL^5$ ,  $HOS^5$ ,  $OLS^5$ ,  $AEHO^5$ ,  $AEOL^5$ ,  $AEOS^5$ ,  
 $AHOL^5$ ,  $AHOS^5$ ,  $AOLS^5$ ,  $EHOL^5$ ,  $EHOS^5$ ,  $EOLS^5$ ,  $HOLS^5$ ,  $AEHOL^5$ ,

*AEHOS<sup>5</sup>, AEOLS<sup>5</sup>, AHOLS<sup>5</sup>, EHOLS<sup>5</sup>, AEHOLS<sup>5</sup>*

Probability of Component Failures

*M<sup>4</sup>,4.4e-03 D<sup>4</sup>,4.4e-03 MD<sup>4</sup>,1.3e-05 A<sup>5</sup>,7.5e-02 E<sup>5</sup>,7.5e-02 H<sup>5</sup>,7.5e-02*  
*O<sup>5</sup>,7.5e-02 L<sup>5</sup>,7.5e-02 S<sup>5</sup>,7.5e-02 AE<sup>5</sup>,8.0e-05 AH<sup>5</sup>,8.0e-05 AO<sup>5</sup>,8.0e-*  
*05 AL<sup>5</sup>,8.0e-05 AS<sup>5</sup>,8.0e-05 EH<sup>5</sup>,8.0e-05 EO<sup>5</sup>,8.0e-05 EL<sup>5</sup>,8.0e-05*  
*ES<sup>5</sup>,8.0e-05 HO<sup>5</sup>,8.0e-05 HL<sup>5</sup>,8.0e-05 HS<sup>5</sup>,8.0e-05 OL<sup>5</sup>,8.0e-05 OS<sup>5</sup>,8.0e-*  
*05 LS<sup>5</sup>,8.0e-05 AEH<sup>5</sup>,4.9e-05 AEO<sup>5</sup>,4.9e-05 AEL<sup>5</sup>,4.9e-05 AES<sup>5</sup>,4.9e-*  
*05 AHO<sup>5</sup>,4.9e-05 AHL<sup>5</sup>,4.9e-05 AHS<sup>5</sup>,4.9e-05 AOL<sup>5</sup>,4.9e-05 AOS<sup>5</sup>,4.9e-*  
*05 ALS<sup>5</sup>,4.9e-05 EHO<sup>5</sup>,4.9e-05 EHL<sup>5</sup>,4.9e-05 EHS<sup>5</sup>,4.9e-05 EOL<sup>5</sup>,4.9e-*  
*05 EOS<sup>5</sup>,4.9e-05 ELS<sup>5</sup>,4.9e-05 HOL<sup>5</sup>,4.9e-05 HOS<sup>5</sup>,4.9e-05 HLS<sup>5</sup>,4.9e-05*  
*OLS<sup>5</sup>,4.9e-05 AEHO<sup>5</sup>,4.0e-05 AEHL<sup>5</sup>,4.0e-05 AEHS<sup>5</sup>,4.0e-05 AEOL<sup>5</sup>,4.0e-*  
*05 AEOS<sup>5</sup>,4.0e-05 AELS<sup>5</sup>,4.0e-05 AHOL<sup>5</sup>,4.0e-05 AHOS<sup>5</sup>,4.0e-05*  
*AHLS<sup>5</sup>,4.0e-05 AOLS<sup>5</sup>,4.0e-05 EHOL<sup>5</sup>,4.0e-05 EHOS<sup>5</sup>,4.0e-05 EHLS<sup>5</sup>,4.0e-*  
*05 EOLS<sup>5</sup>,4.0e-05 HOLS<sup>5</sup>,4.0e-05 AEHOL<sup>5</sup>,3.9e-05 AEHOS<sup>5</sup>,3.9e-*  
*05 AEHLS<sup>5</sup>,3.9e-05 AEOLS<sup>5</sup>,3.9e-05 AHOLS<sup>5</sup>,3.9e-05 EHOLS<sup>5</sup>,3.9e-05*  
*AEHOLS<sup>5</sup>,4.2e-05 D<sup>1</sup>,2.3e-01 E<sup>1</sup>,2.3e-01 H<sup>1</sup>,2.3e-01 J<sup>1</sup>,2.3e-01 R<sup>1</sup>,2.3e-01*  
*DE<sup>1</sup>,4.7e-04 DH<sup>1</sup>,4.7e-04 DJ<sup>1</sup>,4.7e-04 DR<sup>1</sup>,4.7e-04 EH<sup>1</sup>,4.7e-04 EJ<sup>1</sup>,4.7e-*  
*04 ER<sup>1</sup>,4.7e-04 HJ<sup>1</sup>,4.7e-04 HR<sup>1</sup>,4.7e-04 JR<sup>1</sup>,4.7e-04 DEH<sup>1</sup>,3.3e-04*  
*DEJ<sup>1</sup>,3.3e-04 DER<sup>1</sup>,3.3e-04 DHJ<sup>1</sup>,3.3e-04 DHR<sup>1</sup>,3.3e-04 DJR<sup>1</sup>,3.3e-04*  
*EHJ<sup>1</sup>,3.3e-04 EHR<sup>1</sup>,3.3e-04 EJR<sup>1</sup>,3.3e-04 HJR<sup>1</sup>,3.3e-04 DEHJ<sup>1</sup>,3.0e-*

04  $DEHR^1, 3.0e-04$   $DEJR^1, 3.0e-04$   $DHJR^1, 3.0e-04$   $EHJR^1, 3.0e-04$   
 $DEHJR^1, 3.2e-04$   $C^2, 5.3e-02$   $F^2, 5.3e-02$   $I^2, 5.3e-02$   $L^2, 5.3e-02$   $CF^2, 2.1e-04$   
 $CI^2, 2.1e-04$   $CL^2, 2.1e-04$   $FI^2, 2.1e-04$   $FL^2, 2.1e-04$   $IL^2, 2.1e-04$   $CFI^2, 1.7e-$   
 04  $CFL^2, 1.7e-04$   $CIL^2, 1.7e-04$   $FIL^2, 1.7e-04$   $CFIL^2, 1.8e-04$   $N^3, 6.8e-02$   
 $K^3, 6.8e-02$   $A^3, 6.8e-02$   $NK^3, 6.7e-04$   $NA^3, 6.7e-04$   $KA^3, 6.7e-04$   $NKA^3, 6.1e-$   
 04  $A, 9.2e-02$   $B, 7.1e-03$   $C, 9.1e-03$   $D, 6.7e-03$   $E, 1.3e-05$   $F, 9.5e-04$   $G, 2.4e-$   
 04  $H, 2.6e-02$   $I, 1.5e-03$   $J, 6.5e-02$   $K, 1.2e-06$   $L, 3.6e-03$   $M, 4.3e-05$   $N, 6.5e-$   
 03  $O, 9.0e-07$   $P, 6.9e-04$   $Q, 4.0e-03$   $R, 8.1e-04$   $S, 5.7e-06$   $T, 6.3e-06$   $U, 7.3e-05$   
 $V, 2.4e-06$   $W, 8.1e-01$   $X, 4.6e-06$   $Y, 5.1e-06$   $Z, 2.4e-06$   $BA, 6.9e-06$   $BB, 4.7e-$   
 01  $BC, 8.4e-05$   $BD, 3.9e-04$   $BE, 1.8e-04$   $BF, 4.4e-04$   $BG, 6.9e-01$   $BH, 7.0e-01$   
 $BI, 1.0e-06$   $BJ, 3.4e-03$