Dynamic Accident Sequence Analysis using Dynamic Flowgraph Method and Markov/Cell-to-Cell Mapping Technique

by

Chireuding Zeliang

A Thesis submitted in Partial Fulfilment
of the Requirements for the Degree of

Master of Applied Science in Nuclear Engineering

The Faculty of Energy Systems and Nuclear Science
University of Ontario Institute of Technology
2000 Simcoe Street North, Oshawa, Ontario, Canada, L1H 7K4

July 2018

THESIS EXAMINATION INFORMATION

Submitted by: **Chireuding Zeliang**

Master of Applied Science in **Nuclear Engineering**

Thesis Title:
Dynamic Accident Sequence Analysis using Dynamic Flowgraph Method and Markov/Cell-to-cell Mapping Technique

An oral defense of this thesis took place on July 24, 2018 in front of the following examining committee:

Examining Committee:

Chair of Examining Committee	Prof. Eleodor Nichita
Research Supervisor	Prof. Akira Tokuhiro
Examining Committee Member	Prof. Glenn Harvel
External Examiner	Dr. Salam K. Ali, Ontario Power Generation

The above committee determined that the thesis is acceptable in form and content and that a satisfactory knowledge of the field covered by the thesis was demonstrated by the candidate during an oral examination. A signed copy of the Certificate of Approval is available from the School of Graduate and Postdoctoral Studies.

ABSTRACT

In the recent years, numerous concerns have been raised regarding the capabilities and adequacy of classical probabilistic safety assessment (PSA) techniques (fault/event tree) to account for dynamic system interactions and time-dependent accident sequence evolution. Subsequently there is an interest within the PSA community to complement the classical techniques with dynamic methodologies; driven in addition by a goal to eventually develop a framework for integrated safety assessment. The first phase of this research investigates and addresses the limitations of classical techniques and performs a methodological comparison between classical and dynamic PSA techniques. The dynamic flowgraph method (DFM) and Markov model coupled with cell-to-cell mapping technique (Markov-CCMT) were the two dynamic methodologies selected for this research. These methods are ranked as the top methodologies with favorable and minimal uncertainties as determined by the United States Nuclear Regulatory Commission. The capabilities and limitations of the techniques are demonstrated by applying it to a benchmark liquid level control system exhibiting dynamic characteristics and interactions. Reliability analysis of the system using ET/FT are performed using the CAFTA code. DFM model of the benchmark system is developed using the DYMONDA code and coupling of Markov-CCMT model is performed using Fortran95 and MATLAB code. Classical techniques were found to overestimate the predicted top event frequencies by more than one order of magnitude depending on whether or not dynamic interactions among the units through the state variable is accounted for. The study shows that DFM focuses more on sequential probabilistic system evolution, whereas Markov-CCMT emphasizes the exact timing of a failure event. The second phase involves the development of a novel approach for integrated reliability assessment of passive safety systems in small modular reactors. A stochastic model of a passive Isolation Condenser System (ICS) was developed, and its state transition probabilities are computed using finite element method. The analysis predicts high system reliability, with the ICS most likely to fail by pressure boundary breach followed by condensate return and venting unit failure.

Key words: Dynamic and classical PSA; DFM; Markov-CCMT model; SMRs; passive safety systems

AUTHOR'S DECLARATION

I hereby declare that this thesis consists of original work of which I have authored. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize the University of Ontario Institute of Technology to lend this thesis to other institutions or individuals for the purpose of scholarly research. I further authorize University of Ontario Institute of Technology to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research. I understand that my thesis will be made electronically available to the public.

Chireuding Zeliang

STATEMENT OF CONTRIBUTIONS

I hereby certify that I am the sole author of this thesis. The current research contributes in uncovering the limitations and capabilities of classical probabilistic risk assessment techniques used in nuclear power plants; and demonstrates the advantages of dynamic probabilistic risk assessment techniques (Chapter 4), and how some risk-significant scenarios may not be captured by classical techniques. The work described in Chapter 5 presents a novel approach for integrated risk assessment of passive safety systems in small modular reactor technology.

ACKNOWLEDGEMENT

I express my sincere gratitude and appreciation to the Dean and Professor Akira Tokuhiro for his invaluable guidance and continuous support throughout my research activities. His advice on my research as well as my professional career have been priceless. I would not have imagined a better advisor and mentor for my research works.

I would like to extend my gratitude to Professor Lixuan Lu for her advice and insightful contributions to my research pathways.

My sincere acknowledgement goes to the International Atomic Energy Agency and University of Ontario Institute of Technology for accepting my research proposal, partially funding my research, and providing me the opportunity to work as the Co-Scientific Investigator of the IAEA-UOIT Coordinated Research Project.

I am highly indebted to my mentor, Dr. Atam Rao, without whose support and encouragement, I would not have made it this far. I shall ever remain grateful to you.

I thank all the other members of my advisory committee for their constructive comments and suggestions.

Last but not the least, words cannot express how grateful I am to my family for the constant support, encouragement and sacrifices that they have made on my behalf. To my parents, you have been and continue to be an inspiration in my life.

This thesis is dedicated to my parents and siblings.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

BE	: Basic Event
CCF	: Common cause failure
CCMT	: Cell-to-cell Mapping Technique
CRP	: Coordinated Research Project
DFM	: Dynamic Flowgraph Method
ET	: Event Tree
FMEA	: Failure modes and effect analysis
FT	: Fault Tree
HX	: Heat Exchanger
IAEA	: International Atomic Energy Agency
ICS	: Isolation Condenser System
IE	: Initiating Event
iPWR	: Integral Pressurized Water Reactor
MCS	: Minimal cut-set
MVL	: Multi-valued Logic
NPPs	: Nuclear Power Plant
PI	: Prime Implicants
PRA	: Probabilistic Risk Assessment
PSA	: Probabilistic Safety Assessment
PSSs	: Passive Safety Systems
RCS	: Reactor Coolant System
RPV	: Reactor Pressure Vessel
SMRs	: Small Modular Reactors
US NRC	: United States Nuclear Regulatory Commission

NOMENCLATURE

l	: Liquid level in the tank
$\tilde{a}_l, \tilde{b}_l$	: Lowest and highest allowable magnitudes of state variable l
$c_m(n_m/n'_m, k\Delta t)$	: Transition probability for Unit-m from state $n'_m \rightarrow n_m$ during the time step $[k\Delta t, (k+1)\Delta t]$
f_n and $f_{n'}$	: Flowrate for the component state combination n and n'
$f_n(x)$	: L-vectors with components $f_{n,l}(x)$
$g(j/j', n', k\Delta t)$	: Pr{Controlled state variables are in cell j at time $(k+1)\Delta t$ given that the controlled variables are in cell j' at time t
$h(n/n', k\Delta t)$	: Conditional probability that the unit state combination at time $t = (k+1)\Delta t$ is at n, given that the unit state combination at time $t = k\Delta t$ is at n'
$h(n/n', j' \rightarrow j, k\Delta t)$	: Pr{component state combination is in state n at time $t = (k+1)\Delta t$ given that the component state combination is in state n' at time $t = k\Delta t$, and the controlled variables move from cell j' to cell j during $k\Delta t \leq t \leq (k+1)\Delta t$}
I_k	: Number of states for unit k
J	: Total number of V_j
$\tilde{J_r}, J_r$	: Number of V_j in $\tilde{V_r}$ and ordering of V_j in V respectively, $J_r = J_{r-1} + \tilde{J}_r$ for $r = 1, 2, \ldots\ldots R$; $J_o = 0$
L	: Number of state variables
M	: Number of units in the benchmark system
n	: Component/unit state combination index
n_m	: Component/unit state index ($n_m = 1, 2, \ldots\ldots N_m$
N_m	: Total number of n_m
N	: Number of component state combinations
$P(n, x, t)$	: $Pr\{i(t) = S_n, x(t) = x\}$
$P_{n,j}\big((k+1).\Delta t\big)$	: Probability that the system is in cell j and unit state combination n at time $(k+1)\Delta t$
$q_{n,j}^{n',j'}(k\Delta t)$	: Elements of the transition matrix for the Markov chain
R	: Number of control regions in V
S_n	: Ordered set $\{n_i, \ldots, n_k\}$
t	: Time

V	: Control space, $V \equiv \{x;\ \tilde{a}_l < x_l < \tilde{b}_l\}$
$\bar{V}$	: Complement of *V*, *(V U $\bar{V}$) = E*
V_j	: Cells that partition the state variable state space $j = 1, 2, \ldots . J$
$\tilde{V}_r$	: Control region *r*, *r = 1*, ..., *R*
V_γ	: Pairwise disjoint intervals in $\bar{V}$, $\gamma = J_R + 1, \ldots \ldots, J_R + \Gamma$
$v_{j'}$	: Volume of the cell $V_{j'}$
$x_l(t)$	: Magnitude of the state variable *l* at time *t* (l= 1, 2, ….., L)
$x(t + \Delta t)$	: State vector magnitude at time $(t + \Delta t)$ or arrival point
$\tilde{x}_{(k+1)\Delta t}(x', \alpha_{n'}, k\Delta t)$	: Location of the system in the state space at time $t = (k + 1)\Delta t$, given that the system location is x' at time $t = k\Delta t$ and the component state combination is at n'
Γ	: Number of system failure types
$\lambda_m^{fo}/\lambda_m^{fc}$	: Failure rates of the units with the subscript failed-open and closed
$\delta_{n',n}$	: Kronecker delta
Δt	: Time step
$\psi_i(n)$	: Probability of the system being in cell i at $t = n$
	: Logic AND gate
	: Logic OR gate

CHAPTER 1: INTRODUCTION

1.1. Background

Nuclear energy provides a viable option in contributing towards a reduced carbon world, and to meet the ever-increasing global energy demand in a safe, reliable and sustainable way. In nuclear power plants (NPPs), safety assurance is achieved through the application of defense-in-depth strategy and a robust regulatory framework in the process of design, commissioning, operation and decommissioning. Since the inception of nuclear power technology, safety analysis has been performed to demonstrate and understand the safety of an NPP as well as to meet the stringent regulatory requirements. An overall safety assessment of a NPP includes deterministic safety analysis (DSA), hazard analysis and probabilistic safety assessment (PSA) *[CNSC-REGDOC-2.5.2 (2014)]*. This thesis focusses on the application of PSA techniques to estimate the risk associated with an NPP. To assist in the process of an integrated risk-informed decision making, PSA strive to answer some of the fundamental questions possessed by the application of nuclear technology, including, "what can go wrong?" "how likely is it?" and "what are its consequences?" [*WASH-1400 (1975)*]. Thus, a comprehensive PSA considers the likelihood of a failure event, accident progression, and consequences resulting from the failure event. Here, failure event may include equipment failure, human error or any event that challenges the plant normal operation. This approach provides insights into the strengths and weaknesses of the design and operation of an NPP as well as give confidence in the design alignment with fundamental safety objectives *[CNSC-REGDOC-2.4.2 (2014)]*. Due to the specific features of nuclear installation, PSA is categorized into three (3) levels: Level 1, Level 2 and Level 3. A Level 1 PSA estimates the frequency of accidents leading to a core damage, i.e., an estimation of core damage frequency (CDF). Level 2 PSA estimates the frequency of accidents that release radioactivity from an NPP. The results obtained from Level 1 forms the basis for Level 2 analysis. Finally, Level 3 PSA starts with the results obtained from Level 2 i.e., the radioactivity release accidents, and estimates the accident consequences in terms of injury to the public and environmental damage *[IAEA-INSAG-6 (1992)]*.

This research focus on Level 1 PSA which models the various plant response to an initiating event (IE) that challenges the normal plant operation. The plant response implies the activation/operation/failure of several safety and mitigation systems (also called frontline

systems) following an initiating event. These sequence paths (i.e., success or failure of frontline systems) is called the accident sequence and may include several sequences for an IE which can result in a safe reactor state or can lead to a reactor core damage. Historically, the modelling of such accident sequences in nuclear industry is visualized graphically with the event tree (ET) technique, in which, the frontline systems are represented as top events that are required to respond to an IE. Accident sequence modelling with ET can be called as a system level analysis. In order to determine the root cause of a failure event, the frontline systems are modelled individually with the graphical fault tree (FT) technique. The FT method constitute a component level analysis. Finally, an integrated risk assessment model of a plant and an estimate of the frequency of core damage is accomplished by linking the ET and FT model [*WASH-1400 (1975), NUREG/CR-2300 (1983)*]. Besides the FT technique, there exist several other methodologies for individual system analysis such as the discrete space-time Markov model which is a major part of this research. In this thesis, risk estimation with the fault tree, Markov model and event tree methodologies will be categorized under the term classical PSA techniques. The classical approach is widely accepted and recognized as a standard tool for licensing, regulation and safety analysis within the nuclear industry. Most of the current existing NPPs PSA model are based on the classical approach (ET/FT) substantiated with sensitivity, importance and uncertainty analysis to add credit and increase confidence in the model.

While a high level of safety is maintained with a conservative and an integrated safety assessment process, the nuclear industry nevertheless has experienced major accidents including, Three Mile Island accident (1979), Chernobyl accident (1986) and the recent Fukushima Daiichi accident (2011); all three accidents resulting from human error. The nuclear community has been incorporating the lessons learned from these major accidents and the practical countermeasures to cope with such an accident, resulting into an evolving rigorous and stringent regulatory/safety requirements. This has further led the PSA community to re-examine and question the capabilities of classical PSA techniques to model human error and adequately account for risk-significant accident sequence occurring from dynamic system behavior and interactions. In the past decades, numerous concerns have been raised in several reviews regarding the capabilities of classical PSA techniques, including: [*Sui et al. (1989, 1994)*; *Marzio et al. (1998); Aldemir et al. (2007, 2013); Devooght et al. (1992(a), 1996*; *Acosta et al. (1991); Marseguerra et al. (1998); NUREG/CR 6901 (2006); Kirschenbaum et al. (2009)]*. In brief summary, the classical PSA techniques lack:

1. Treatment of time element *[Devooght et al. (1992(a)); Marseguerra et al. (1996); Marzio et al. (1998); NUREG/CR-6901 (2006)]*;
2. Dependencies arising from dynamic interactions *[Hassan et al. (1990); Belhadj et al. (1992)]*;
3. Limited capability in modelling system/components with multi-states [*Acosta et al. (1991)];*
4. Treatment of event sequence ordering *[Aldemir (1987)];*
5. Inadequate treatment of human interactions/error [*Acosta et al. (1993)*].

The lack of treatment of the above-mentioned factors arising from dynamic interactions means that the classical PSA approach may not identify or properly quantify potentially risk-significant sequences and dependencies between the failure events. Furthermore, such interactions may lead to coupling between stochastic logical events (e.g., valve openings, pump startups) during an accident scenario, which can significantly impact the predicted system failure probabilities *[Belhadj et al. (1992)]*. These limitations of classical PSA techniques entailed the development of an advanced class of methodologies commonly called dynamic PSA (DPSA) methodologies or integrated deterministic and probabilistic safety assessment (IDPSA) techniques. Dynamic PSA methodologies are defined as those which use a time-dependent phenomenological model of system evolution along with its stochastic behavior to account for possible dependencies between failure events [*Aldemir (2013); NUREG-6901 (2006)*]. The use of the term *"dynamic"* in this thesis implies an approach that combines explicit modeling of system deterministic evolution with stochastic modeling. Dynamic methods explicitly model the time element, state ordering and the mutual interactions among state variables, system components, and operators. Furthermore, an adequate assessment of the overall risk requires accounting for aleatory uncertainties arising from the stochastic nature of component failures as well as epistemic uncertainties arising from limited knowledge of the physical phenomenon relevant to the system. Dynamic PSA provide an integrated framework to account for both the uncertainties simultaneously *[Zio (2014)]*. Thus, treating the deterministic and probabilistic approach in an integrated fashion, unlike the classical approach where the two are treated separately. Several dynamic methodologies have been proposed with most of the methodologies in the developing phase including and have been reviewed in the literatures *[Sui (1994); Labeau et al. (2000);*

Aldemir (2013); NUREG-6901]. In this thesis, DFM and Markov-CCMT model is selected for further research, and demonstration of the features, advantages and limitations of the methodologies. *Figure 1* presents a flowchart of the research approach and methodology selection.

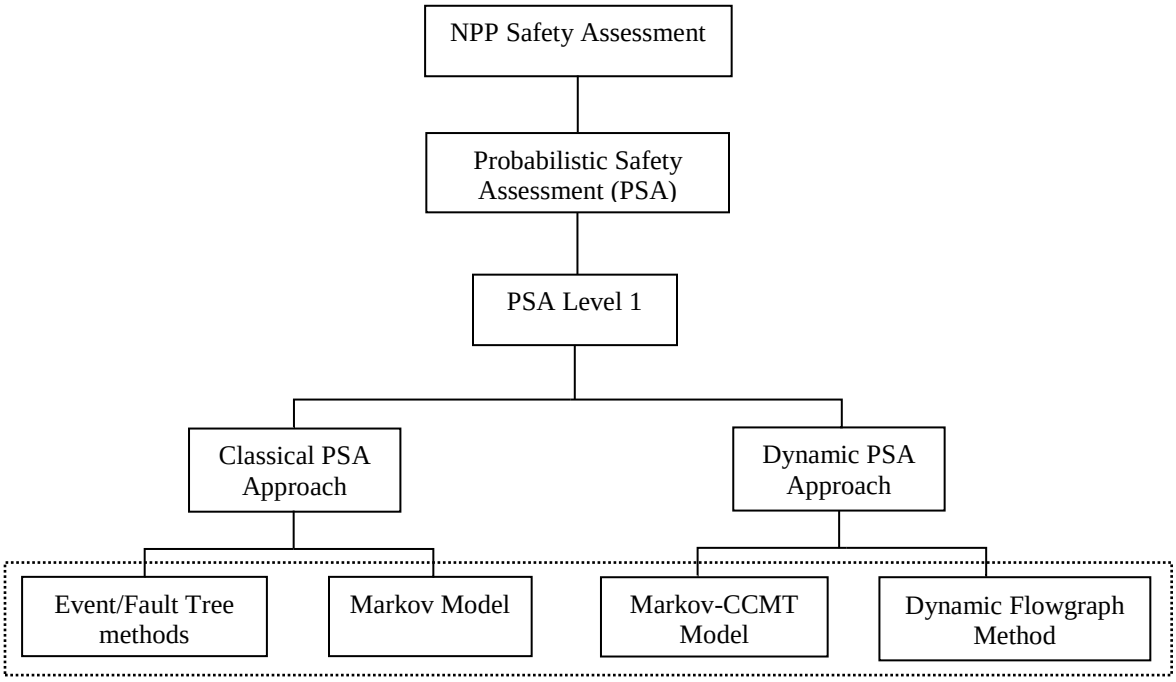


Figure 1: Flowchart of the research approach

Before further detail discussion into the specific methodologies, the five (5) factors mentioned earlier will be described briefly in context to the classical approach. It should be noted that this research is bounded within the first four (4) elements. First, the ET/FT methodologies are based on static logic approach. Neither does it capture the time of occurrence of a failure event nor the time distribution before an undesired event is reached, i.e., the time available for the subsequent failure event following an IE. Due to its static nature, it assumes a system failure as soon as a minimal cut-set (MCS) occurs; this may lead to overestimating the failure probability. Of course, time is a significant factor in shaping the system dynamics, i.e., the state variables (e.g., reactor coolant system pressure) are time-dependent. Second, the classical techniques are neither developed nor intended to model integrated system dynamics and probabilistic accident evolution. An accident scenario is simply described by a set of success or failure events, where a risk-significant accident sequence lead to a single core degraded state. The identification and

analysis of these accident sequence can be performed with ease using the well-known logic based approach and the laws of probability. Thus accounting for dependencies emerging from dynamic interactions using the classical approach does not arise. This can lead to an inaccurate estimate of risk when quantifying risk associated with scenarios for which the system dynamic behavior is a significant factor *[Marseguerra et al. (1998)]*. Since this research attempts to model and adequately account for dependencies, a clear distinction between static and dynamic dependencies will be discussed along with the dependencies covered by classical PSA techniques. The principle of defense-in-depth design characteristics in nuclear power plants implies that a risk-significant accident scenario must involve a failure of multiple barriers, which includes protection, safety and containment systems. Thus, in order to accurately estimate a plant risk, it is essential to determine the probability of multiple barrier failures. In probability theory, the joint occurrence of any two failure events A and B is given by: *[Vesely et al. (1981)]*

$$P(A,B) = P(A).P(B|A) = P(B).P(A|B)$$

Where, $P(B|A)$ and $P(A|B)$ quantifies the dependencies between event A and B. An overly conservative or optimistic estimate of the dependencies can result in an error of the estimated risk. Much effort has been put in to adequately identify and quantify these dependencies between failure events. *Acosta et al. (1991)* discuss three type of dependencies treated by classical PSA techniques: (a) Dependencies arising due to common cause failures; (b) Dependencies due to functional coupling; and (c) Multiple top events sharing the same basic events or shared equipment (well treated in the fault tree analysis). However, dependencies outside of these categories arising due to the dynamic interactions are not well treated by the classical techniques [*Sui et al. (1989)*]. These three dependencies are briefly described:

1. *Direct dynamic dependencies*

These types of dependencies arise from direct interaction between two safety systems through state variables. A failure of one safety system results in a deviation of the state variables, which in turn, affects the performance of the other safety system. For example, the sequence of events that occurred at Hatch 1 plant in 1985 involving a safety relief valve (SRV) and vessel integrity with the reactor coolant system (RCS) temperature as the state variable.

2. *Indirect dynamic dependencies*

These dependencies are similar to the direct dynamic dependencies except that there is an intermediary between the demanded safety systems. The event sequence is as follow: failure of the first safety system causes a deviation in the state variables, which lead to the intermediate system to affect the operation of the second safety system. For example, this sequence of events occurred at the Three Mile Island-II accident, 1979; the failure of a pilot-operated relief valve (first demanded system) to reclose resulted in a high pressurizer level (the state variable) which led the operator (intermittent) to throttle the high pressure injection (HPI) system (second demanded system). The state of HPI is affected by the operator action, which was influenced by the pressurizer level. The classical approach does not have the feature to account for pressurizer level-operator interaction.

3. Cyclical dynamic dependencies

These types of dependencies involve the multiple occurrence of certain events. For example, multiple opening and closing of safety relief valves at Davis-Besse plant, 1985 with the RCS pressure as the state variable. The affected functions in these sequence of events was the demand for safety relief valves to open and subsequent demand to reclose due to the variation in RCS pressure (the state variable). Standard event tree typically models these event sequence (opening and reclosing) just once in the tree.

Third, the classical approach is based on binary logic (normal or failure), and there exist no well-defined approach to modelling system/components having multiple states or top events and may not yield satisfactory result *[Hassan et al. (1990); NUREG/CR-6942 (2007)]*. For example, modelling a control valve having the states; normal, fail-closed, fail-open, fail in 50% position, fail in 80% position, etc. Fault tree tend to model the failure modes independently, rather than taking into account the competing failure modes. This is especially true while computing the reliability of a system having several components with multiple failure modes. This phenomenon is well explained and demonstrated using the FT and Markov model in Chapter 3, Section 3 with a simple redundant system with two (2) components having three failure modes. We illustrate how FT approach overestimates the failure probability.

Fourth, in the classical approach, the order in which the frontline system operates following an IE is normally predefined by the analyst. In the event tree method, the sequencing of failure events is fixed. For instance, given a demanded system that operates successfully, the sequence is

terminated, i.e., leading to a successful reactor state. However, event sequence can be generated wherein the system is demanded multiple times due to the system dynamics. Additionally, the order of occurrence of failure events can lead to a different undesired system state. This is demonstrated in Chapter 4 where the order of unit failure leads to a system overflow or drained. As for the fault tree technique, it is not intended to model ordering of failure event, rather it represents the top event in terms of combinations of basic events (minimal cut-sets) without any particular ordering index. In contrast, the sequencing of events is not predetermined in dynamic methodologies but rather is derived from the time-dependent system model solution (system code) as the system evolves.

Dynamic PSA methodologies thus have features and capabilities to overcome the limitations encountered in classical PSA techniques. This is especially pertinent when modelling passive systems where epistemic uncertainties are significant due to lack of knowledge of physical phenomenon and operating experience, as compared to active systems which have been employed in most current operating NPPs and have a vast operating experience. This may be especially true for SMRs with either different forced or free convective systems. The IAEA defines passive safety systems as "a system that is composed entirely of passive components and structures or a system, which uses active components in a very limited way to initiate subsequent passive operation". [*IAEA-TECDOC-626*]

Passive safety systems (PSSs) have been employed in several (Gen-III and Gen-III+) reactor designs [AP1000, ESBWR, NuScale, mPower, SMART] to meet the increasing safety requirements, to take advantage of natural forces (e.g., gravity), less dependence on active systems/components (e.g., emergency diesel generators), design simplicity, increased reliability and economic competitiveness. However, due to their reliance on a small driving force, lack of data and significant uncertainties; passive safety systems reliability, model development and performance under normal and accident conditions still remains an open issue [*Burgazzi (2009)*]. Also, the implementation of passive systems in NPPs is challenging the state-of-the-art safety analyses (classical PSA/DSA) due to the additional uncertainties rendering a difficult a priori judgement of the conservative scenarios selected for DSA and PSA [*Zio (2014)*]. These elements necessitate for the development and demonstration of consistent methodologies and approaches for evaluating the reliability of passive safety systems. *[Burgazzi (2007, 2009, 2011, 2017); Marques et al. (2015) Nayak et al. (2014); Zio et al. (2009); IAEA-TECDOC-1752 (2014)]*

1.2. Motivations

There still exist no consensus on a standardized dynamic PSA technique, unlike the classical techniques (FT/ET), which are universally accepted all around the world. Rather, most of the dynamic PSA methodologies are still in the developing and validating stage *[Labeau et al. (2000)]*. Furthermore, there is no agreement on the specific application and under what scenario dynamic PSA methodologies might be appropriate. The motivations for this research thus include:

1. Limitations in classical PSA techniques to account for time element, system interactions, dynamic dependencies, multistate modelling and state ordering;
2. Dynamic PSA techniques provide an integrated framework by unifying the system stochastic model as well as its dynamic evolution; thus are anticipated to overcome the limitations encountered in classical PSA techniques. This is expected and envisioned within the PSA community to contribute significantly in providing a realistic accident progression scenario by capturing risk important interactions;
3. Currently there exist no standardized or universally accepted dynamic methodology as well as a lack of consensus within the PSA community on the applicability and the context to which dynamic methodologies are required;
4. *NUREG/CR-6901* identified and ranked DFM and Markov-CCMT as the top two dynamic methodologies with the most positive features and least negative or uncertain features;
5. *NUREG/CR-6942* recommended thorough research to resolve the challenges for practical implementation of dynamics PSA methods. The report further went on to state that "resolving these challenges would involve a stand-alone reliability modeling of full benchmark system using DFM, Markov/CCMT and classical ET/FT approach";
6. Dynamic PSA approach has not been applied for integrated reliability assessment of passive safety systems (to the author's knowledge). The author was thus motivated to evaluate the possibility of modelling PSSs using dynamic methods;
7. The supervisor's experience at NuScale Power, and a full dynamic PRA using system code (RELAP5) coupling experience for light water reactors (LWRs) under a benchmark station blackout.

1.3. Research Objectives

The objectives of the research are:

1. Formulate the coupling of the Markov model with Cell-to-cell Mapping Technique for dynamic probabilistic safety assessment;
2. Demonstrate how DFM and Markov-CCMT model can be implemented for modelling stochastic dynamic system evolution and interactions. This can contribute in performing system reliability analysis using the two dynamic methodologies as well as uncover the advantages and disadvantages of the techniques;
3. Illustration of the applicability of backtracking algorithm for timed-fault trees generation from DFM model;
4. Methodological comparative assessment (qualitative and quantitative) of classical and dynamic PSA techniques;
5. Development of a novel approach for integrated functional reliability assessment of passive safety systems in integrated small modular reactors (SMRs).

1.4. Scope of the Research

This research consists of two independent areas that will cover the applicability of DFM and Markov-CCMT model for dynamic probabilistic safety assessment, and development of a novel approach for integrated functional reliability assessment of PSSs in integrated small modular reactors. First, the need for dynamic PSA techniques was explored, and two dynamic PSA methodologies, namely, DFM and Markov-CCMT model was selected for further research and demonstration purpose. Second, the Markov-CCMT model was selected for reliability modelling and performance analysis of a passive isolation condenser system (ICS) implemented in a generic integral pressurized water reactor (iPWR) type SMRs.

1.5. Thesis Organization

This thesis consists of six (6) chapters, and is organized as follow:

Chapter 1: This chapter presents the background of the research, motivations for selecting the research topic, objectives of the research and the structure of the thesis.

Chapter 2: This chapter presents the literature review performed during the course of this research. The review includes classical PSA techniques, their limitations; thus the need of dynamic PSA methodologies. A more extensive review of DFM and Markov-CCMT model is presented.

Chapter 3: This chapter presents the methodologies implemented in this research in a sequential fashion, which includes, FT/ET analysis, classical Markov model, DFM and coupled Markov-CCMT model.

Chapter 4: This chapter demonstrate application of methods to a simple liquid level control system. First, reliability analysis of the benchmark system was performed using ET/FT and time-dependent Markov model. These form the classical techniques. Then, DFM and Markov-CCMT model are applied for dynamic reliability analysis of the benchmark system using DYMONDA and Fortran95 software and programming tools.

Chapter 5: This chapter presents a novel approach for dynamic reliability assessment of PSSs. A brief description of the ICS to be used as a benchmark system for the project is provided. A stochastic model of the ICS and computation of transition state probabilities is presented.

Chapter 6: Finally, this chapter summarizes the research contributions and also suggests future research tasks.

Appendix-I: Multi-state modelling with FT and Markov model

Appendix-II: Fortran95 code for the benchmark liquid level control system

Appendix-III: Benchmark system control space discretization in MATLAB using CCMT

Appendix-IV: Benchmark system state vector trajectories results from MATLAB

Appendix-V: Fortran95 code for the passive isolation condenser system

Appendix-VI: Powerpoint presentation used in thesis examination

CHAPTER 2: LITERATURE REVIEW

This chapter presents a literature review that was performed during the course of this research. Literature review starts with the well-known classical PSA methodologies, their application in the nuclear industry and available algorithms for minimal cut-set (MCS) computation. A review of literatures to address limitations of classical methods and currently available dynamic PSA methods was performed, of which, DFM and the Markov model coupled with CCMT are selected in this thesis for further research and finally application/demonstration. The literature review is provided in a sequential order from classical PSA techniques, limitations in classical techniques and thus, the need for dynamic PSA techniques, and finally a comprehensive review of the selected methodologies is provided. Literature review during the course of this research was focused on dynamic methods. Note that in this thesis, usage of the term dynamic PSA and Integrated Deterministic and Probabilistic Safety Assessment (IDPSA) is synonymous.

A. Classical Probabilistic Safety Assessment Techniques

The classical PSA techniques discussed in this thesis mainly consist of the well-known and commonly used risk assessment techniques in current NPPs, i.e., the fault tree and event tree (FT/ET) analysis. The application of classical PSA techniques in NPPs was introduced by the *WASH-1400 (1975) "The Rasmussen Report"*, with the objective to identify possible accident scenarios following an initiating event, and to quantify the likelihood of a reactor core damage and its consequences by observing the response (success/failure) of the frontline mitigating and safety systems. *WASH-1400* demonstrated the usefulness and practicality of classical techniques. The risk assessment review group report to the USNRC through *NUREG/CR-0400* summarized that:

> "The fault-tree/event-tree methodology is sound, and both can and should be more widely used by NRC. Proper application of the methodology can therefore provide a tool for the NRC to make the licensing and regulatory process more rational, in more properly matching its resources to the risks provided by the proper application of the methodology".

A comprehensive and systematic approach to the construction, qualitative and quantitative analysis of fault trees is provided in *"Fault tree handbook"* by *Vesely et al. (1981)* and

Stamatelatos et al. (2002). The authors described FTA as an analytical technique whereby an undesired event or top event is pre-defined by an analyst, and system analysis is performed backward to uncover the root cause of the top event. The methodology basically consists of three steps: (1) construction of the tree; (2) qualitative analysis of the tree; and (3) quantification of minimal cut-sets. A detail guideline to performing a PSA for NPPs via classical techniques is provided in *NUREG/CR-2300 (1983)*. Fault tree construction *(Fussell, 1973)* can be done automatically, for instance, *Salem et al. (1976, 1977)* developed a new methodology based on decision tables and implemented in the code Computer Automated Tree (CAT) for automatic construction of FTs. The analysis of a FT is performed using Boolean algebra, and the top event is entirely described by an equivalent set of Boolean equation. There exist several algorithms for computing and quantifying the MCS for a top event, such as the method of obtaining cut sets (MOCUS) and minimal cut-sets upward (MICSUP) *[Rasmussen (1978); Kumamoto et al. (2000); Ruijters et al. (2015)]*. Furthermore, there are several advanced computer codes for automated construction of FT/ETs, and subsequent generation of MCS/accident sequence including, CAFTA and FaulTree+ code.

B. Dynamic Probabilistic Safety Assessment Techniques

While classical PSA techniques are accepted as standard safety analysis technique in the PSA community, numerous concerns have been raised in the recent years regarding its capability to adequately account for the time element, multistate modelling, state ordering and stochastic system interaction that shapes the dynamic probabilistic evolution of an accident sequence [*Sui et al. (1989, 1994*); *Marzio et al. (1998); Aldemir et al. (2007, 2013); Zio (2014); Devooght and Smidts (1992(a)); Acosta et al. (1991, 1993); Marseguerra et al. (1996)*]. The behavior of the plant state variables resulting in a coupling of failure events, with the failure events being probabilistically dependent, in contrast to the logical dependencies caused by direct systems interaction have been debated within the PSA community for several decades [*Aldemir et al. (1987); Sui et al. (1989); Devooght and Smidts (1992(a))*; *Belhadj et al. (1992); Labeau et al. (2000)*].

Sui (1989) characterized these dependencies into three classes by reviewing the licensee events reports (1969-1979 and 1985) for incidents that occurred at the US operating commercial light water reactors, namely: (1) Direct physical dependencies; (2) Indirect physical dependencies; and

(3) Cyclical physical dependencies. *Sui* underlined the importance of dynamic dependencies and how it played an important role in the Three Mile Island (TMI-2) accident. Classical PSA techniques lack the ability to capture these dependencies, and hence may fail to identify risk-significant scenarios. Four (4) potential dynamic methodologies and their attributes were discussed for incorporating dynamic behavior of the plant into the accident sequence analysis, which includes, Expanded Event Tree, Logic Flowgraph Method, Markov model and Dynamic Logical Analytical Methodology (DYLAM) *(Sui et al. (1989)). Sui (1994)* further attempted to address the limitations of classical techniques in accident sequence modelling and suggested possible dynamic methodologies for explicit treatment of time dependency.

The need for dynamic PSA for accident sequence modelling to account for evolution of the state variable, operator state of mind and scenario history was further reinforced by *Acosta et al. (1991)*. The author showed how an overly conservative or optimistic assumptions of failure events could result in an inaccurate risk estimate. The coupling among the mentioned factors was demonstrated by applying the Dynamic Event Tree Analysis Method to a SG tube rupture accident and concluded that the dynamic method can uncover risk-significant scenarios and better define dependencies between failure events as compared to the classical methods, *Acosta et al. (1993). Marzio et al. (1998)* discussed the specific field of application of the dynamic PSA methods, and the viability of the Monte Carlo technique as a tool to model the stochastic part of the analysis to reduce computation time. The importance of dynamic methods in PSA modelling of the passive safety system was underscored and demonstrated with two simple examples *(Aldemir (2013)).*

On the other hand, *Devooght and Smidts (1992(a))* took an analytical approach and formalized the concept of probabilistic reactor dynamics by providing a rigorous mathematical framework of continuous event tree in which the deterministic aspect of the reactor state variable trajectories in state-space is supplemented by the stochastic nature of the reactor configuration (e.g., random component failure, human error, etc.). The system dynamics is described by a set of partial differential equations from the Chapman-Kolmogorov equation assuming a Markovian system. *Smidts and Devooght (1992)* demonstrated the methodology by applying it to a realistic accident transient in a fast reactor primary coolant system. The capability of the method to model complex interaction between operators and the reactor during a transient is present in [*Devooght and Smidts (1992(b)); Smidts (1992)]*. However, the applicability of the method to realistic systems is limited

due to its computational demand and continuous time-dependent plant data requirements. A review of the achievements of the theory of probabilistic dynamics and its adjoint formulation to determine the outcome of a transient is presented in *Devooght and Smidts (1996).*

A comprehensive review of the state of dynamic PSA methodologies for reliability analysis of digital systems in NPPs is reported in *NUREG/CR- 6901 (2006)*. DFM and dynamic event tree or Markov model approach was ranked as the top two methodologies with the most positive features and least negative or uncertain features, meeting most of the requirements, and with each methodology having some advantages as well as limitations. For a proof of concept, a benchmark digital feedwater control system is analyzed using DFM and Markov-CCMT in the reports *NUREG/CR-6942 (2006)* and *NUREG/CR-6985 (2009)* as a follow-up of *NUREG/CR-6901 (2006)*. A detail review of currently available dynamic PSA techniques, and a modular approach to the development of high-level and user-friendly tools to implement dynamic methods in industries is proposed by *Labeau et al. (2000).* The literature underlined some important issues, which includes, a significant amount of research is required for the development of the computational algorithm and the determination of an optimal computational engine. *Zio (2014)* laid out the concept, challenges and research directions for industrial application of IDPSA. The author also noted that, IDPSA should be consider as a complement to the existing DSA and PSA and not as an alternative. *Zio (2014)* and *Aldemir (2013)* showed how IDPSA can explore a more complete scenario space and coverage of undesired events by integrating a system dynamic and stochastic aspect, and consistent treatment of uncertainties (aleatory and epistemic) in the analysis. *Table 1* groups and summarizes dynamic PSA techniques into three (3) categories, namely, (1) Extension of classical methodologies; (2) Explicit state-transition methods; and (3) Implicit state-transition methods.

Table 1: A summary of dynamic PSA techniques

Parameters	*Dynamic PSA Approaches*		
	Extension of classical PSA methodologies	*Explicit state-transition methods*	*Implicit state-transition methods*
Input requirements	All dynamic techniques require: (1) Time-dependent system model; (2) System configurations under normal/abnormal operating scenario; and (3) Transition probabilities among these system states		
Methodologies	Dynamic Flowgraph Method; Digraph-based fault tree; Expanded Event Tree; GO-FLOW	Explicit Markov chain models; Markov-CCMT model; Event Sequence Diagrams	Continuous Event Trees; Dynamic Event Trees; Discrete event (Monte Carlo) simulation
Technique	An extension of classical approach incorporating system dynamics	Discretized version of Continuous Event Trees	Analytical based
Accounting for system dynamics	Limited capability	Explicitly account for system dynamics and control laws	Explicitly account for system dynamics and control laws
State ordering	Sequential top event ordering in tree structure	A priori explicitly ordered system states	Evolving sequence from probabilistic system dynamics
Scenario history	Explicit representation of scenario history in the tree structure	Markovian approach-memoryless	Capability for treatment of scenario history
Recent development to address some drawbacks	Reduction technique for component combinatory explosion	State merging technique employed to counter state space explosion	Truncation rule by probability or event type; intelligent sampling scheme to reduce run time
Treatment of time element	Limited treatment of time dependencies	Explicit treatment of time element	Explicit treatment of time element

Complex system modeling	Limited	Limited	Enables treatment of complex/realistic models
Integrated user friendly codes	Several integrated platform exist, e.g., Dymonda, GO-GLOW.	None. Requires code coupling	None. Requires code coupling
Operator behavior model	Limited capability to model state variables and operator behavior	Limited capability to model operator states	Explicitly deals with operator states and dynamic man-machine interface systems
Post-processing	Output similar to that of classical techniques	Requires considerable post processing	Requires considerable post processing
Advantages	Phase mission scenario implementation	Ease of treatment of rare events	Eliminates expert judgement
Drawbacks	No direct treatment and computation of common cause failure and importance measure	Requires explicit evaluation of transition probability matrix; Difficult to envision the set of all possible system states prior to scenario development	Requires significant computational time and resources to run realistic models; Requires algorithm development specific to system under analysis (e.g., situation specific rules for operator state transition)

Note: All dynamic PSA methods encounter the phenomenon of state space explosion, however manifested in different ways for different approaches

B.1. The Dynamic Flowgraph Method

The DFM is a multi-valued logic (MVL) diagraph-based method that express the logical and dynamic behavior of a system in terms of causal relationships among system parameters, and a series of discrete state transitions *[Garrett et al. (1995); Yau et al. (1995, 1998); Guarro et al. (1996)]*. DFM is a three-step approach: model development, model analysis (qualitative) and

finally the quantitative analysis which includes uncertainty, sensitivity analysis, etc. All the above three steps can be performed in a single platform using the code DYMONDA *(ASCA Inc., 2013)*. Once the model is developed, analysis or generation of prime implicants (PIs) can be performed via an inductive or a deductive algorithm. Modelling of a system involves the construction of decision tables, which is an extension of the well-established truth table. Application of decision tables to FT construction was introduced by *Salem et al. (1979)*, and the application of MVL decision table for risk analysis was comprehensively discussed in *Ogunbiyi (1981b)*. MVL rules implemented for analysis of critical transition table to obtain the complete base is discussed in the literatures *Ogunbiyi et al. (1981(a), 1981(b))* and *Yau (1997)*. The generation of PIs complete base can be performed using several methodologies, which includes *[Garribba et al. (1985)]*: (1) Tabular method; (2) Nelson method; and (3) Method of Generalized Consensus. Since the DYMONDA code is based on the Generalized Consensus method, the same methodology is used for the current thesis. Identification of failure events using PIs and the need for generation of complete base of PIs are presented in *Yau (1997)*.

A comprehensive and integrated framework for modelling and analysis of system reliability and safety assurance is captured in the literature *[Garrett et al. (1995, 2002); Yau et al. (1995, 1997); Guarro et al. (1996)]*. The integrated DFM approach was first applied for reliability assessment and verification purpose of a software-driven embedded system by *Garrett et al. (1995)*. Early on, the methodology was mainly aimed at identifying an optimal testing strategy based on system behavior analysis, *Garrett et al. (1995)*. A further demonstration of the methodology was performed using the Titan II space launch vehicle digital flight control system with the objective to complement classical PSA techniques, *Yau et al. (1993)*. The authors showed how time-dependent behavior and switching logic can be captured, and how certain postulated events (desired/undesired) that can occur in a system may be identified by the DFM model. *Yau et al. (1998)* then applied the methodology to a realistic PWR steam generator level control system, in which the author used a fault injection technique to check as to whether the failure event can be detected. The USNRC developed a full-scale DFM software tool to model, analyze, and test software design to provide a high level of safety assurance for software-based control systems, the details of which can be found in *NUREG/CR-6465*. This NRC report presented distinct features of the methodology and demonstrated by applying it to a generic process system as well as to a full scale SG level control system implemented in current NPPs. *NUREG/CR-6942* report

(as a follow-up of *NUREG/CR-6901*) further applied the methodology to model a digital feedwater control system and to check the extent to which the methodology meets the requirements set-out in *NUREG/CR-6901*. The report also provided a framework on how PIs from DFM can be integrated into an existing PSA model. Finally, generation of timed-FTs from a DFM model is discussed in works by, [*Guarro et al. (1996); Yau (1997); Zeliang et al. (2017)*].

Table 2: Summary of review of DFM for dynamic PSA

Authors	*Contributions*	*Limitations*
Salem et al., 1979	Computerized automatic construction of fault tree using decision table	Lacks the treatment of multistate variables; requires analyst judgement and significant amount of time for construction of component decision tables
Ogunbiyi et al. [1981(a); 1981(b)]	Extended the binary Boolean algebra to encompass multistate variables, and simplify consensus algorithm for non-coherent systems	Only a theoretical framework; Applicability and efficiency of the approach to treat realistic systems remains an open issue
Garribba et al., 1985	Provided a theoretical framework for treatment of critical transition table using Tabular method, Nelson algorithm and generalized consensus theory	Lacks the discussion as to which systems these theories are suitable and applicable.
Yau et al. 1993, 1995	Demonstrated the capabilities of DFM by applying it to a realistic Titan II space launch vehicle digital flight control system	Focused mostly on the construction of decision tables for software subroutines using Newton-Raphson method
Garrett et al. (1995, 2002)	Provided an integrated modelling framework for reliability assessment and verification of embedded control systems	Focused only on risk assessment of software driven systems and its testing strategy

Guarro et al. (1996); Yau (1997); NUREG/CR-6942	Provided a theoretical and systematic framework for generation of timed-fault tree from DFM model	Limited discussion on specific application of dynamic and physical consistency rule, and treatment of logic loops
Yau et al. (1997, 1998)	Applied the methodology to a realistic PWR software driven SG level control system	Requires further inductive analysis to locate fault condition, i.e., backtracking must be employed even after obtaining the complete base
USNRC-NUREG/CR-6465	Developed a full-scale DFM tool for safety analysis of control software in advanced reactors	Focused mainly on assurance and design verification of software driven closed loop systems, rather than identification of failure events
USNRC-NUREG/CR-6942	Dynamic reliability modelling of benchmark digital feedwater control system	Only a proof of concept; acceptability of failure data; requires further research to validate the practicality of DFM
Zeliang et al. (2017)	Generated timed-fault tree from DFM model of a PWR SG level control system	The approach requires significant amount of time for systematic post processing of critical decision tables; particularly inefficient in the treatment of logic loops, where an event can be backtracked to itself, and consistency rules must be checked in every steps.

B.2. Coupled Markov/Cell-to-cell Mapping Technique

Markov Chain: In 1907, A. A. Markov developed the Markov chain or the discrete Markov process that consisted of a simple chain with infinite sequence of random variables connected in

such a way that x_{i+1} for any i is independent of the past evolution if x_i is known *(Basharin et al. (2004))*. The Markov chain is a well-established method and can be found in many standard references *[Pukite (1998); Privault et al. (2013); Taylor et al. (2013), chapter-8, 9 and 10; Sheskin (2016)]*. The modelling of dependencies and common cause failures between failures of component in redundant systems using a continuous time, four (4) state Markov chain was discussed and illustrated by *Platz (1984)*. Practical application of the methodology to real systems was unrealized due to a very large transition matrix, or the phenomenon called state space explosion. *Papazoglou (1977)* laid down two novel approaches to deduce large transition probability matrices by systematic state ordering and merging of processes for system exhibiting symmetries respectively. Several algorithms were developed for systematic generation and treatment of structural properties of system transition matrix in a computer aided environment *[Amoia et al. (1981); Cafaro et al. (1986); Lesanovsky (1988)]*.

Cell-to-cell Mapping Technique: The theory of cell-to-cell mapping was first formulated by *Hsu (1980a, 1980b)* to describe global behavior of non-linear dynamic systems, in which, a state variable evolution is treated as the probability of transition among a collection of computational cells in the state-space. The theory was based on the idea of describing a dynamic system governed by ordinary differential equations (ODEs) with a finite number of computational cells. *Hsu (1981)* generalized the theory of cell-to-cell mapping to treat complex non-linear global behavior pattern by enabling a cell to have multiple transition probabilities or image cells. The generalize cell mapping was deduced to the well-known Markov chain in which, the dynamical properties of the system under analysis is entirely constituted in the transition probability matrix. The application and improvements of cell mapping technique for determination of global long-term behavior of non-linear dynamic systems using several approaches such as, cell refinement and lumping can be found in *[Mo-Hong (1993); Spek (1994); Chen (2004)]*.

Coupling Markov model with CCMT: Markov-CCMT model describes a system dynamics as probabilistic evolution of state variables in discrete space-time conditioned on system configuration, *Aldemir (1987)*. The multi-state system components and configuration is modelled by a Markov chain, and CCMT describes the system dynamics in terms of probability of transition among a finite set of computational cells in discretized state space and time *[Hassan et al. (1990);*

Belhadj et al. (1992); NUREG/CR-6942 (2007)]. A comprehensive theoretical basis of the methodology can be found in [*Aldemir (1987); Dinca (1997)]*.

The concept of coupling Markov model with CCMT was first formulated by *Aldemir (1987)* to describe the probabilistic evolution of system behavior in discrete space-time. The process of coupling with sets of ODEs and its simulation by a Markov chain was demonstrated by applying it to a hypothetical process control system failure. *Aldemir (1989)* further illustrated the limitations of static methods in failure modelling of process control systems and how the overall failure probability may be overestimated. *Hassan et al. (1990)* applied the methodology to a realistic closed loop control system i.e., the high-pressure injection system and SRVs in BWRs and observed the probabilistic behavior of the systems following a small loss of coolant accident (SB-LOCA). The authors demonstrated that cell-to-cell transition probabilities can be computed, provided a data base of system behavior under normal and abnormal operation. *Belhadj et al. (1992)* further demonstrated capabilities of the methodology in modelling time-dependent behavior by applying it to a feed-bleed cooling scenario in BWRs through intermittent operation of high pressure core spray system and SRVs following a SBLOCA. In an effort to reduce the computational time and computer storage requirements, *Belhadj et al. (1995)* developed two algorithms based on the Chapman-Kolmogorov equation. Accounting for uncertainties in system parameters and initial conditions was demonstrated by *Aldemir et al. (1996). Tombuyses et al. (1997)* performed a parametric study on computational efficiency of continuous CCMT and concluded that a fourth order Runge-Kutta and mid-point implicit scheme seems to be suitable for short and steady state system behavior simulation.

A benchmark system analysis i.e., a typical SG feedwater control system in PWR to be used as a basis of the methodology was proposed in *NUREG/CR-6942 (2007)* and *Kirschenbaum et al. (2009)*. *Gomes et al. (2013)* demonstrated the superiority of the methodology in capturing interactions and identifying possible failure events in digital systems by implementing it to a typical PWR digital SG level control system. *Yang et al. (2016)* developed an algorithm for deductive implementation of the model, and concluded that, the approach in principle is similar to FT, in that, it starts with a predefined top event and backtrack for root cause analysis.

Table 3: Summary of review of Markov-CCMT model for dynamic PSA

Authors	*Contributions*	*Limitations*
Aldemir (1987, 1989)	Coupled classical Markov model with CCMT for dynamic reliability assessment and failure modelling of process control system	Computationally intensive and impractical with increasing failure modes and state variables; only for binary failure modes
Aldemir et al (1996)	Theoretical framework for treatment of uncertainties in system parameters and initial conditions	Considerable resource requirements, and a very fine partitioning of state space required for uncertainty analysis
Hassan et al. (1990)	Observed that failure characteristic maybe dependent on the order of failure and exact timing of failure event	The approach is useful only if there is an existing data base of state variable evolution
Belhadj et al. (1992)	Demonstrated that the predicted failure probability may significantly change depending on whether dynamic interactions is accounted for or not.	Classical approach (FT) is used for state merging purpose, therefore not completely dynamic in nature
Tombuyses et al. (1997)	Computational efficiency of CCCMT maybe increased using 4^{th} order Runge-Kutta and mid-point implicit scheme for short and long term system behavior respectively	Predicted results are not validated or compared with CCMT space-time discretization schemes; approach not suitable for increased step size as noted by the authors.
Kirschenbaum et al. (2009)	Applied the methodology to a realistic feedwater control system for operating PWR	Quantification of top event probability was not performed

Gomes et al. (2013)	Applied the methodology to a realistic SG level control system in a typical PWR	The study only focused on the main and backup computers, and not on mechanical components (e.g., control valves and pumps)
Yang et al. (2016)	Proposed a theoretical framework for deductive implementation of the methodology	The approach is applicable only when the transition probability matrix is symmetric

CHAPTER 3: PROBABILISTIC SAFETY ASSESSMENT TECHNIQUES

In this thesis, several PSA methodologies was studied, and are categorized into two parts- Classical PSA and Dynamic PSA methodologies. The classical methods consist of the well-established static ET/FT analysis, whereas dynamic methods consist of DFM and Markov-CCMT model. More discussion is provided on dynamic PSA methodologies, and a detail theoretical basis of Markov-CCMT model was emphasized due to its nature of tight coupling between stochastic and dynamic model. Several examples are performed to get an intuitive understanding of the methodologies as well as to make a direct comparison among the methodologies. *Sub-section 3.1* and *3.2* discusses the FT/ET technique respectively; *Sub-section 3.3* presents the DFM technique; *Sub-section 3.4* discusses the classical Markov chain; *Sub-section 3.5* provide a theoretical basis for CCMT, and *Sub-section 3.6* provide a detail discussion of Markov-CCMT model.

3.1. Fault Tree Analysis

Fault tree analysis (FTA) is a well-established logic method, and one of the most prominent probabilistic technique used in probabilistic safety assessment and reliability assessment of engineered systems in NPPs. FT is a logical, systematic and comprehensive approach for system safety analysis that is capable of uncovering design and operational weakness *[Stamatelatos et al. (2002)]*. The methodology starts by defining an undesired system state (top event) and is traced back to the basic events to determine the possible causes of the predefined top event. A basic event (BE) which normally is a failure event can be a failure of system components, human error, software error, or any other events which can lead to the top event. A FT thus depicts the logical interrelationships of BEs that lead to the top event of the FT, i.e., it models the propagation of failures through a system, and how these failure interactions can lead to a total system-failure mode. Thus, for a given FT, the top event is characterized by a set of BEs, in that, if all BE occurs, the top event is expected to occur. These set of basic events is called the "*cut-set*". A cut-set can be further deduced to a "*minimal cut-set (MCS)*", such that if any BE is removed from the set, the remaining BEs collectively are no longer a cut-set. There exist several approach for determination of MCS based on Boolean algebra, and the process of deduction is called the qualitative analysis. Thus, the objective of the methodology, or an extensive effort is put in to determine the MCSs and its corresponding probabilities that ultimately dictates the occurrence probability of a top

event. Propagation of MCSs given the BE failure probabilities through the tree for computation of top event probability is called quantitative analysis. It should be noted that the fundamental concept of the methodology is based on binary logic, i.e., a component can either be in normal or failed state only. Hence, the top event can only take two states since it is a function of BEs.

For illustration and comprehension purpose, a generic FT taken from *Vesely et al. (1981)* and constructed in CAFTA *[EPRI, 2013]* is shown in *Figure 2*. The system consists of a pressure tank, pump and associated control system that regulates the tank pressure by controlling the pump. When the pressure setpoint is reached in the tank, the control system opens the pressure switch and thus removing power to the pump, causing the pump to stop operation. A failure of the pressure switch or relay contacts activates an alarm that enables the operator to manually switch-off the pump. Of course, a combination of these failures will guarantee a system failure (tank over pressurized) as shown in the FT.

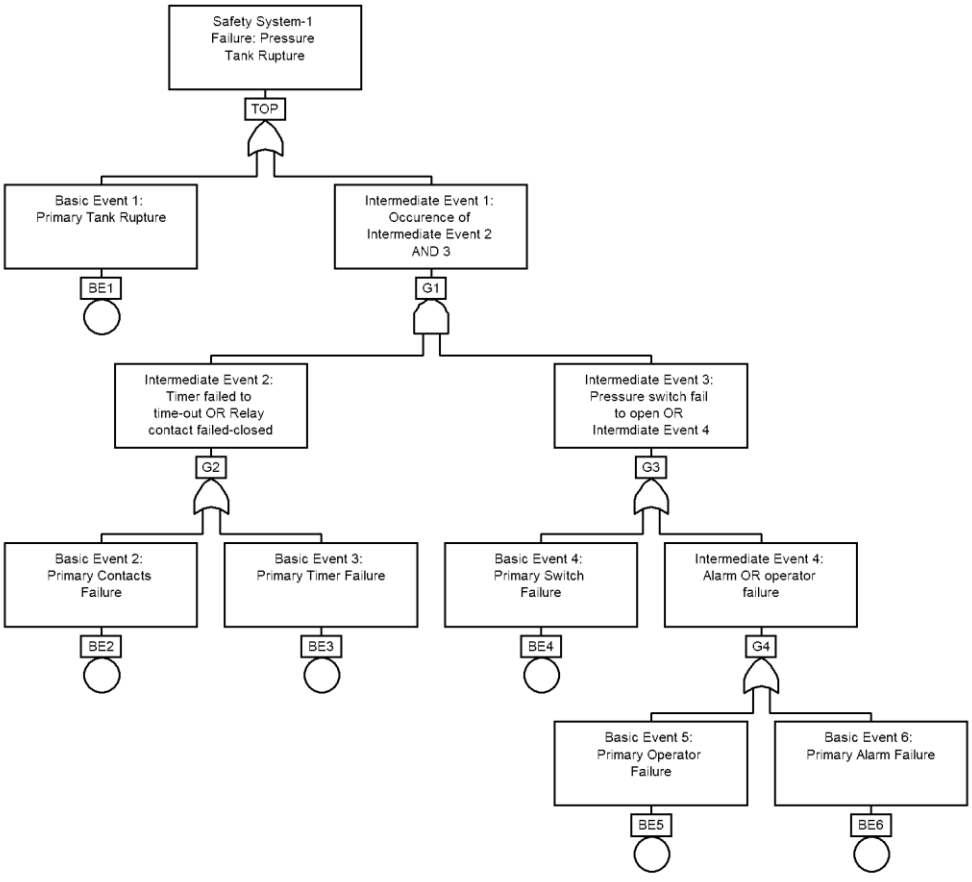


Figure 2: Example fault tree (pressure tank)

The FT shown in *Figure 2* is compose of the top event, intermediate events (1 to 4), logic gates (G1 to G4 including the TOP gate) and basic/primary events (BE1 to BE6). Firstly, the BEs are those that requires no further development and are represented by a circle in the fault tree which signifies the limit of resolution. Secondly, the logic gates establish a relationship among the basic events that is required for the occurrence of a higher event. The higher event being the output of the gate, and the lower events being the inputs of the gate. The gate symbol denotes the type of relationship of the input events required for the output event.

3.1.1. Qualitative Evaluation of Fault Trees

The qualitative evaluations transform the FT logic into logically equivalent forms that provide more focused information *[Stamatelatos et al. (2002)]*. There exist several methods for qualitative analysis of standard fault trees with the same objective of determining the MCS that establish a direct relationship between BEs and a predefined top event *[Vesely et al. (1981)]*. Qualitative analysis can be performed using inductive (Bottom-Up) as well as deductive approach (Top-Down) once the tree is translated to its equivalent Boolean equations. Before going ahead with detail discussion of available algorithm for qualitative and quantitative analysis, it is necessary to depict some logic rules and Boolean algebra utilized in the algorithms (*See Table 4*).

Table 4: Boolean algebra binary reduction rules

Mathematical symbolism	*Laws*
$\vee$	Logic OR gate
$\wedge$	Logic AND gate
$A \vee B = B \vee A$	Commutative Law
$A \vee A = A$	Idempotent Law
$A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$	Distributive Law
$A \wedge (B \wedge C) = (A \wedge B) \wedge C$	Associative Law
$A \wedge (A \vee B) = A$	Law of Absorption
$(A \vee B)' = A' \vee B'$	De Morgan's theorem

The MCSs for a top event can be obtained using the well-known and the most common 'Top-Down' approach. The approach can be implemented with the method of obtaining cut-sets

(MOCUS) algorithm *[Vesely et al. (1981); Kumamoto et al. (2000)]*. It is based on the observation that logic OR gates increase the number of cut-sets, whereas logic AND gates enlarge the size of the cut-sets. MOCUS algorithm is demonstrated using the example FT.

Step-1: Naming logic gates

The logic gates are named as shown in the FT which includes, TOP, G1, G2, G3 and G4 gate.

Step 2: Numbering basic events

All basic events in the FT is named which includes; BE1, BE2, BE3, BE4, BE5 and BE6.

Step 3: Expansion of TOP

The uppermost logic gate "TOP" is identified, which forms the first element of the matrix. This is an 'OR' gate, and thus "TOP" is replaced by a vertical array of the input (BE1 and G1) to the gate:

TOP	BE1
	G1

Here, only G1 can be further expanded since BE1 is already a basic event.

Step 4: Expansion of G1

It may be noticed that G1 is a logic 'AND' gate, and thus G1 is replaced by a horizontal array of input (G2 and G3) to the gate:

BE1
G2, G3

Step 5: Expansion of G2

Notice that G2 is and OR gate and thus it is replaced by a vertical array of the input (BE2 and BE3) to the gate:

BE1
BE2, G3
BE3, G3

Here, BE2 and BE3 are basic events and cannot be further expanded. Only G3 can be expanded.

Step 5: Expansion of G3

The gate G3 is a logic OR gate, and thus it is replaced by a vertical array of the input (BE4 and G4) to the gate:

BE1
BE2, BE4
BE2, G4
BE3, BE4
BE3, G4

Step 5: Expansion of G4

The gate G4 is a logic OR gate, and thus it is replaced by a vertical array of the input (BE5 and BE6) to the gate:

BE1
BE2, BE4
BE2, BE5
BE2, BE6
BE3, BE4
BE3, BE5
BE3, BE6

The above seven (7) rows in Step-5 represents seven (7) cut-sets for the FT. Notice that there are no supersets (i.e., cut-sets that contains other complete cut-sets) in the cut-sets, and hence the 7 cut-sets are the minimal cut-sets of the tree.

{BE1}, {BE2, BE4}, {BE2, BE5}, {BE2, BE6}, {BE3, BE4}, {BE3, BE5}, and {BE3, BE6}

3.1.2. Quantitative Evaluation of Fault Trees

The quantification of FT top event (system unavailability/unreliability) can be performed given the probability of occurrence of the individual BEs. MCS generated from qualitative analysis is used for the quantification of top events. There are several methods for computation of the top event such as the structure function method, rare event approximation method and the MCS upper bound method, *Beeson (2002)*. For illustration purpose assume the following failure probability of BEs: BE1= 5×10^{-6}, BE2= 3×10^{-5}, BE3= 1×10^{-4}, BE4= 1×10^{-4}, BE5= 1×10^{-6}, and BE6= 5×10^{-6}. Generally, approximation methods are used for the computation of top event probability because the probabilities are relatively small, and computation of the exact

probability becomes complex due to the dependencies between cut-sets. In this thesis, the MCS upper bound method is used due to its simplicity and availability of software (CAFTA) that uses the methodology for top event probability computation. A system failure occurs if at least one of the MCS exist, and hence:

$$Pr\ (\text{System failure}) = Pr\ (\text{Atleast once MCS exists})$$

Thus;

$$Pr\ (TOP) = \left[1 - \prod_{i=1}^{n}(1 - P(MCS_i))\right] \tag{3-1}$$

Where, $P(MCS_i) =$ The probability of the i^{th} cut-set; and $n =$ Total number of cut-sets

Computation of the top event for the example FT using the above approximation method gives:

$$Pr\ (TOP) = 5.013 \times 10^{-6}$$

Furthermore, the relative importance of any MCSs can be obtained by taking the ratio of the MCS probability to the total system probability. For example, taking the first MCS (a first order) of the example system, the relative importance is determined to be $Im_{mcs1} = 99.73\ \%$, which implies the BE1 is the most critical component in the system.

3.2. Event Tree Analysis

Event tree analysis (ETA) is a well-established technique *[WASH-1400 (1975); NUREG-1150]* that model an integrated overall plant response to an abnormal event. An event tree (ET) depicts possible sequence of events logically and graphically following an initiating event (abnormal condition that challenges the normal plant operation). The progression of an accident scenario from an initiating event to some final plant state is systematically analyzed taking into account the safety systems available and operator actions (success or failure) to counter or mitigate the progression of an accident. ETA is an inductive technique that starts with an abnormal event and progresses through the tree not only to determine the resulting consequences but also to enumerate all possible accident scenarios. It should be noted here that ET technique is based on binary logic. The accident sequence is characterized by the top events that generally represents the frontline systems that is needed to respond to an initiating event. The process of obtaining these sequences is called the qualitative analysis. Top events in an ET is typically modeled using FT technique.

Within the current PRA framework, FTA and ETA are generally linked to determine the overall plant core damage frequency, i.e., a FT model the individual systems and an ET model the overall plant states [*NUREG-1150; Kumamoto et al. (2000)]*. Upon obtaining the top event probabilities from FTs, accident sequences in the ET can be quantified.

For illustration purpose, consider a simple ET shown in *Figure 3*. The overall system consists of three (3) safety systems (SS) that can either be functional or can be in a failed state. For instance, these safety systems can be emergency core cooling system, electrical power, containment system, etc. Given an initiating event (IE) with a probability P_{IE}, the progression of the accident scenario in the ET results in a total of 8 accident sequences. The sequences can be divided into two general factors- the occurrence of IE and failure of frontline systems. Frontline systems in this example are SS-1, 2 and 3, which can individually be modelled via FTs and linked to the ET. *Figure 2* FT which is considered as SS-1 is linked to the ET *(Figure 3)* for illustration purpose.

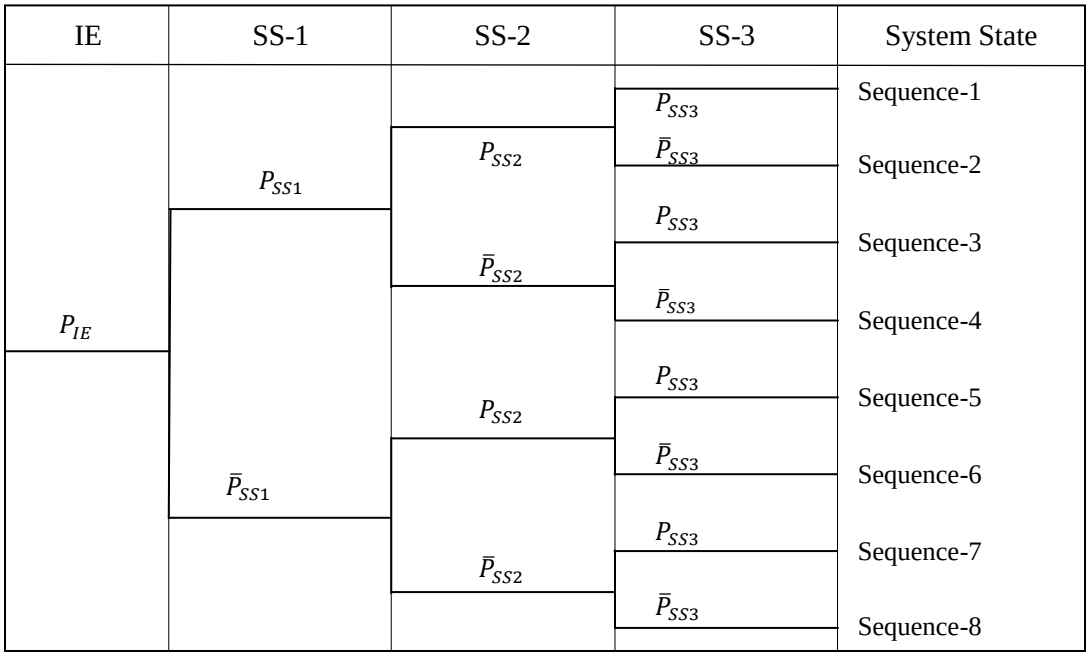


Figure 3: Example event tree

Assuming that the safety systems failure is independent from each other, accident sequences from ET can simply be obtained by multiplying the probabilities of passing along each branch point on any path through the tree by IE frequency. The accident sequences are shown in *Table 5*. For instance, $P_{IE}, P_{SS1}, P_{SS2}\ and\ P_{SS3}$ implies the probability of IE frequency, success probability of safety system 1, 2 and 3 respectively, whereas $\bar{P}_{SS1}$ implies SS-1 failure probability.

Table 5: Accident sequence from the event tree

Accident Sequence		*Probability*
Sequence-1	$P_{IE}P_{SS1}P_{SS2}P_{SS3}$	0.998
Sequence-2	$P_{IE}P_{SS1}P_{SS2}(1-P_{SS3})$	1.25E-07
Sequence-3	$P_{IE}P_{SS1}(1-P_{SS2})P_{SS3}$	5.34E-07
Sequence-4	$P_{IE}P_{SS1}(1-P_{SS2})(1-P_{SS3})$	4.48E-11
Sequence-5	$P_{IE}(1-P_{SS1})P_{SS2}P_{SS3}$	7.52E-09
Sequence-6	$P_{IE}(1-P_{SS1})P_{SS2}(1-P_{SS3})$	6.24E-13
Sequence-7	$P_{IE}(1-P_{SS1})(1-P_{SS2})P_{SS3}$	2.71E-12
Sequence-8	$P_{IE}(1-P_{SS1})(1-P_{SS2})(1-P_{SS3})$	2.25E-16

If SS-1 failure probability ($\bar{P}_{SS1}$) is given by the top event of the FT in *Figure 2*, and assuming that $P_{IE} = 1.5 \times 10^{-3}$, $\bar{P}_{SS2} = 3.6 \times 10^{-4}$, and $\bar{P}_{SS3} = 8.3 \times 10^{-5}$ respectively. $\bar{P}_{SS2}$ and $\bar{P}_{SS3}$ failure probability can be from another coupled FT similar to the P_{SS1} that was obtained from the FT of SS-1. Frequency of each of the accident sequence can be computed with ease once the failure probability of each of the frontline system (top event) is obtained. For example, the accident sequence-1 frequency can be computed as:

$$\text{Sequence-1} = P_{IE}P_{SS1}P_{SS2}P_{SS3} = 0.998$$

Similarly, the frequency of the rest of the sequences can be computed as shown in *Table 5*. Once the individual sequences probability is calculated, the total system failure frequency can be computed by summing all the accident sequences (AS) that lead to a system failure.

$$\text{Total failure frequency} = \sum_{i=1}^{n} AS_i \tag{3-2}$$

Where; $n =$ total number of accident sequences leading to a system failure.

The overall failure frequency of the system can be computed from the ET in *Figure 3*, and is determined to be $6.72E-07$. Furthermore, assuming that a failure of SS-1 results in an overall system failure, ET can be reduced from 8 to 5 accident sequences. The 5th accident sequence frequency is: $\text{Sequence-5} = P_{IE}\bar{P}_{SS1} = 7.52E-09$.

3.3. Dynamic Flowgraph Method

The DFM approach is based on representing the system of interest in a digraph (directed graph) model, which is enriched with explicit identification of the cause-effect and time dependencies among significant states of the system parameters that describe the system behavior *[Garrett et al. (1995); Guarro et al. (1996)]*. DFM provides an integrated analytical framework for systematically capturing the logical and dynamic behavior of a system respectively. The system logical nature is expressed in terms of causal relationships between physical variables, whereas dynamic behaviors are represented as a series of discrete state transitions. A system model can be developed in DYMONDA platform with key system parameters that represents discrete-state discrete-time system evolution. Once the DFM model is developed, the analysis can be performed in an inductive or deductive manner. An inductive algorithm can be performed with a set of initial and boundary conditions such as component states, time steps, direction rules, rate rules, etc., and trace the model in the forward direction to obtain a sequence of events that follows from the initial conditions. Whereas in the deductive algorithm, the user defines the desired specific system states and time sequence, and the model is traced backward in time to identify the logical combination of events leading to the defined top event.

The results obtained from an inductive or deductive algorithm are in the form of prime implicants (PIs) that are analogous to the conventional FT MCS, with the exception that the PIs are time-stamped. The probability of a PI occurrence can be calculated given the probability of each state of a component/system parameter (user defined). Then, the quantification of the PIs can be performed to obtain the exact top event probability, or the probability of system being in a specific state sought by the analyst. Once a DFM model is built, it can be used repeatedly to produce MVL and time-dependent PIs for a large number of possible top events that may be postulated for the system. In this thesis, the methodology is implemented using the software DYMONDA developed by ASCA Inc. A detail theoretical description, modelling process and analysis is discussed in the subsequent sections.

3.3.1. Theoretical Basis

The DFM can be thought of as a series of snapshots of the classical FT technique, and hence the methodologies share several similarities despite their differences. Similar to the FTA, the

objective of DFM is to generate the PIs (MVL analogous of MCS encountered in binary FT) of a desired top event. Some definitions are vital for the purpose of theoretical discussion.

Conjunction:	The logical 'OR' operator $(+, \cup, \text{or } \vee)$
Disjunction:	The logical 'AND' operator $(., \cap, \text{or } \wedge)$
Literals:	A primary event taking one of its state, e.g., event A_1 (event A taking state 1 from n number of possible states)
Monomial:	A conjunction of literals, e.g., $X = A_1 B_2 C_3$ (a monomial X is a conjunction of event A, B and C in state 1, 2 and 3 respectively)
Top event:	A disjunction of monomials
Implicant:	A monomial X of disjunctive form of a top event, such that $TOP \cap X = X$ (MVL analogous to cut-set)
Prime Implicant:	X is an implicant of a top event, and any other monomial Y subsumed by X is not an implicant of the top event (MVL analogous to MCS)
Base of top event:	Any disjunction of prime implicants which is equivalent to the top function
Irredundant Base:	A base which ceases to be a base if one of its prime implicants is removed
Complete Base:	The disjunction of all prime implicants

Prime Implicants:

The prime implicants in DFM can be computed from a deductive as well as inductive analysis. In this thesis, the deductive algorithm is emphasized due to its similarity with deductive FTA, which makes it easier for comparison purpose. A deductive analysis starts with a defined top event and track causality in reverse to uncover the root conditions that can cause the top event. As discussed earlier, in the classical FTA, once a tree is developed, Boolean algebra can be applied to reduce the tree to a unique disjunctive normal form in terms of its MCS. The MCSs are computed as a conjunction of primary events *[NUREG-6942]*:

$$MCS_j = \prod_{i=1}^{n} X_i^{(j)} \tag{3-3}$$

Where, $MCS_j =$ Indicator variable for the j^{th} minimal cut-set

$X_i^{(j)}$ = Indicator variable for the i^{th} primary event in the j^{th} minimal cut-set

n = Number of primary events in the j^{th} minimal cut-set

The indicator variable for the top event (condition of interest- fail/success) can then be expressed in disjunctive form as:

$$X_{TOP} = 1 - \prod_{j=1}^{m} \left(1 - MCS_j\right) \tag{3-4}$$

where, m = Total number of PIs.

The set of n PIs obtained from a deductive analysis for a top event of interest, shown in *Equation (3-5)*, is first converted into a set of m mutually exclusive implicants (MEIs) shown in *Equation (3-6)*. These MEIs can be thought of as the MVL equivalent of cut-sets that do not yield any cross product term. The sum of the probabilities of these MEIs yields the exact probability of the top event, as shown in *Equation (3-7)*: *Yau et al. (2007)*

$$\text{Top Event} = PI_1 \vee PI_2 \vee \ldots PI_n \tag{3-5}$$

$PI_i \notin PI_j;\ for\ any\ i \neq j$

$$\text{Top Event} = MEI_1 \vee MEI_2 \vee \ldots MEI_m \tag{3-6}$$

Where; $MEI_i \wedge MEI_j = 0; for\ i \neq j$

$$P(\text{Top Event}) = P(MEI_1) + P(MEI_2) + \cdots + P(MEI_m) \tag{3-7}$$

This way the probability of the top event occurrence can be computed with the knowledge of the probability of the primary events that constitutes the prime implicants. All PIs are time-stamped and are referred to as "timed prime implicants". For instance, $A_i@t = 0$ implies a component A is in state i at time $t = 0$. PIs identified in a DFM analysis are conjunctions of primary events that is sufficient to cause the top event but does not contain any shorter conjunction of events that is sufficient to cause the top event. The base of a top event is first determined, which are of two distinct types i.e., the irredundant base (IB) and complete base (CB) are more difficult to obtain as compared to the FTA. Only the CB are unique and finite, *Philip (2016)*.

3.3.2. Method of Generalized Consensus

The determination of CB of a top event is of prime importance in the methodology. Several methods exist for the determination of CB which include; Tabular method, Nelson method and the Method of Generalized Consensus *[Garribba et al. (1985); Philip (2016)]*. In this thesis, the CB is computed using the Method of Generalized Consensus due to the availability of well-developed software code, which enables one to implement the methodology to a more complex and realistic system. Quine developed the method of consensus for binary logic, which was then extended by *Yau (1997)* to treat MVL. The methodology starts at the component level/basic event having multiple states, and decision tables that establish a relationship among these components. The decision table maps the input to the output parameter and can be constructed in an inductive as well as deductive manner. A detail iterative process for the development of decision tables is provided in the literature *Salem et al. (1976; 1977; 1979).* These decision tables are then merged into a single table called the 'critical transition table'. Several logic rules are applied on the critical transition table to obtain the PIs *[Ogunbiyi et al. (1981); Yau (1997)]*. Thus, the determination of PIs using the Generalized Consensus method consist of two iterative steps: *Garribba et al. (1985)*

a) *Reduction:* Application of multi-valued logic reduction rules including absorption, reduction, merging and reduction-merging operation;
b) *Development:* Addition of monomials to the reduced set of implicants to obtain the consensus terms (can be an implicant or PI).

The above two steps are repeated in an iterative manner until no more new consensus terms are generated, and the total set of PIs include the reduced implicants from the original critical transition table as well as the new PIs generated, *[Garribba et al. (1985); Philip (2016)]*.

Decision Table Development

Decision tables are used to describe each possible output state of a component/sub-system as a set of combinations of states of inputs, i.e., it maps the input parameters to the output. The use of decision tables for FT construction have been studied comprehensively in *[Salem et al. (1976; 1977; 1979)]*. Decision tables can be constructed via inductive or deductive algorithm. The same concept is implemented in DFM with the exception that a decision tables maps the state variables as well as multistate system parameters. Detailed relationship between the multi-state nodes are represented in decision tables. For example, if a sensor can be in three states (*See Table 6*) and is

measuring a liquid level in a tank that can be at 0%, 50% and 100% (*See Table 7)*, the measured level which is the output of the sensor is clearly dependent on the sensor state and the liquid level. The mapping of the input (liquid level- L and sensor state- SS) to output (measured level- ML) can be performed as shown in *Table 8*. Of course, the measure level (ML) will have the same number of states as the liquid level (L).

Table 6: Possible sensor states

No.	*Sensor State (SS)*	*Representation*
1.	Normal	0
2.	Failed Low	-1
3.	Failed High	+1

Table 7: Liquid level in the tank and measured level

No.	*Liquid level State (L or ML)*	*Representation*
1.	Level at 0 %	0
2.	Level at 50 %	1
3.	Level at 100 %	2

The decision table that maps the input to output is given in *Table 8*.

Table 8: Output decision table- the measured liquid level

No.	*Input*		*Output*
	SS @t=-1	*L @t=-1*	*ML @t=-1*
1.	0	0	0
2.	0	1	1
3.	0	2	2
4.	-1	-	0
5.	+1	-	2

The above table can be interpreted as, if the level sensor is in normal condition, it will indicate liquid level values as the original level in the tank. However, if the sensor is failed low or high, it will indicate a low level (0 %) or high level (100 %) respectively no matter what the liquid level is currently in, i.e., a “Don’t care condition”.

A. Multi-Valued Logic Operations

The PIs of a top event can be obtained by merging the decision tables of individual constituents in the system *[Ogunbiyi et al. (1981)]*. The merging operation is dependent on the structure of the DFM model and the top event sought by the analyst, *Philip (2016)*. The MVL operation consist of absorption, merging, reduction and reduction-merging that are implemented to simply the merged critical transition table, and eventually to determine the PIs of a system top event sought by the analyst. The consensus operation is then applied to identify other PIs from the irredundant base and is a vital step in obtaining the complete base of the top event *[Yau (1997)]*.

1. Absorption Rule

A monomial X is said to subsume (absorb) a second monomial Y in a decision table if both the monomials have identical output, and every input event in X also occurs in Y *[Ogunbiyi et al. (1981)]*. This is equivalent to the absorption law in Boolean algebra. For example, if A, B and C are three primary variables with multiple states, and:

$$X = A_1.B_2.C_2 \text{ and } Y = A_1.C_2 \tag{3-8}$$

Then;

$$X + Y = A_1.B_2.C_2 + A_1.C_2 = A_1.C_2 = Y \tag{3-9}$$

Here, event A is in state 1, event B is in state 2 and event C is in state 2. The absorption rule implemented in the transition table can be shown as:

Monomials	*Primary events*				*Primary events*	
	A	*B*	*C*		*A*	*C*
X	1	2	2	≡	1	2
Y	1	−	2			

i.e., monomial X is absorbed into Y.

2. Merging

In a set of monomials $X_1, X_2, \dots, X_n$, if each X_i contains identical literals except for one, and the different literal contains a primary variable that is enumerated in all its states in $X_1, X_2, \dots, X_n$, then the set of monomials can be merged into a single monomial *[Yau (1997)]*. For example, if

A, B and C are primary variables, of which B has three distinct states 1, 2 and 3, and X_1, X_2 and X_3 are monomials with:

X_1	X_2	X_3
$A_1.B_1.C_2$	$A_1.B_2.C_2$	$A_1.B_3.C_2$

It can be observed that event A and C remains the same, whereas event B is enumerated in all its possible three states (1, 2, and 3). Thus monomials X_1, X_2 and X_3 can be merged into a single monomials. For example:

$$X_1 + X_2 + X_3 = A_1.B_1.C_2 + A_1.B_2.C_2 + A_1.B_3.C_2 = A_1.C_2 \quad (3\text{-}10)$$

The merging rule implemented in the transition table can be shown as:

Monomials	*Primary events*				*Primary events*	
	A	*B*	*C*		*A*	*C*
X_1	1	1	2	≡	1	2
X_2	1	2	2			
X_3	1	3	2			

i.e., three rows in the transition table have been merged into a single row.

3. Reduction

In a set of monomials $X_1, X_2, \dots, X_n$, if every literals of X_i except one does not contradict with the literals of the other X_j (either X_j contains the same literal, or X_j does not have a literal with the same primary event), and the contradicting literal contains a primary event that is enumerated in all its states in $X_1, X_2, \dots, X_n$, then X_i can be reduced by removing the contradicting literal containing that primary event, *Yau (1997)*. For example, if A, B and C are primary events, with B having three distinct states 1, 2 and 3; and X_1, X_2 and X_3 are monomials with:

X_1	X_2	X_3
$A_1.B_1.C_2$	B_2	$A_1.B_3$

The primary event A and C in X_1 does not contradict with those in X_2 and X_3, and event B is enumerated in all its three states in the X_i, then X_1 can be reduced as:

$$X_1 + X_2 + X_3 = A_1.B_1.C_2 + B_2 + A_1.B_3 = A_1.C_2 + B_2 + A_1.B_3 \quad (3\text{-}11)$$

The reduction rule implemented in the transition table can be shown as:

Monomials	*Primary events*				*Primary events*		
	A	B	C		A	B	C
X_1	1	1	2	≡	1	−	2
X_2	−	2	−		−	2	−
X_3	1	3	−		1	3	−

It can be observed that X_1 has been reduced from $A_1.B_1.C_2$ to $A_1.C_2$.

4. Reduction-Merging

The reduction-merging rule is a combination of reduction and merging operation and is only applicable to MVL systems. It is used to reduce an MVL decision table into its irredundant (most reduced) form, *Philip (2016).* If a monomial obtained after reduction operation on a set of monomials $X_1, X_2, \dots, X_n$, can absorb other monomials in the set, those subsuming monomials are removed from the set. For example, if A, B and C are primary events, with event B having three distinct states 1, 2 and 3, and X_1, X_2 and X_3 are monomials with: *[Yau (1997)]*

X_1	X_2	X_3
$A_1.B_1.C_2$	$A_1.B_2.C_2$	$B_3.C_1$

Then;

$$X_1 + X_2 + X_3 = A_1 \cdot B_1 \cdot C_2 + A_1 \cdot B_2 \cdot C_2 + B_3 \cdot C_1 \quad (3\text{-}12)$$

$$= A_1.C_2 + A_1.B_2.C_2 + B_3.C_1 \quad (3\text{-}13)$$

$$X_1 + X_2 + X_3 = A_1.C_2 + B_3.C_1 \quad (3\text{-}14)$$

i.e., X_1 is first reduced from $A_1.B_1.C_2$ to $A_1.C_2$, and then X_2 $(A_1.B_2.C_2)$ is absorbed into the new monomial $A_1.C_2$. The reduction-merging rule implemented in the transition table is shown as:

Monomials	*Primary events*				*Primary events*		
	A	*B*	*C*		*A*	*B*	*C*
X_1	1	1	2	≡	1	–	2
X_2	1	2	2		–	–	–
X_3	–	3	1		–	3	1

5. The Consensus Operation

A critical transition table is not guaranteed to contain a complete set of PIs, particularly if multistate components and success states exist. Hence to obtain a CB of a top event, the consensus operation is performed that creates new terms out of the existing terms in the critical decision table by mixing and matching their input events *[Ogunbiyi et al. (1981)]*. The consensus operation can be illustrated as:

Given a primary event A having n distinct state $1, 2, \dots, n$, and monomials $X_1, X_2, \dots, X_n$ do not contain the primary event A, then a consensus operation on the monomials $A_1 \cdot X_1, A_2 \cdot X_2, \dots, A_n \cdot X_n$ yields a new monomial $X_1 \cdot X_2 \cdot \dots \cdot X_n$ provided this new monomial is not inherently false. This can be expressed as: *[Yau (1997)]*

$$A_1 \cdot X_1 + A_2 \cdot X_2 + \cdots + A_n \cdot X_n = A_1 \cdot X_1 + A_2 \cdot X_2 + \cdots + A_n \cdot X_n + X_1 \cdot X_2 \cdot \dots \cdot X_n \quad (3\text{-}15)$$

It can be observed that the consensus operation yields a new implicant $(X_1 \cdot X_2 \cdot \dots \cdot X_n)$ from the base $A_1 \cdot X_1 + A_2 \cdot X_2 + \cdots + A_n \cdot X_n$.

For example, if A, B and C are primary events, with event A having two distinct states 1 and 2, and X_1 and X_2 are monomials such that:

$$X_1 = B_1 \text{ and } X_2 = C_2 \quad (3\text{-}16)$$

The consensus operation on the monomials $A_1.B_1$ and $A_2.C_2$ is given as:

$$A_1 \cdot X_1 + A_2 \cdot X_2 = A_1 \cdot B_1 + A_2 \cdot C_2 \quad (3\text{-}17)$$

$$= A_1 \cdot B_1 + A_2 \cdot C_2 + B_1 C_2 \quad (3\text{-}18)$$

A new monomial $X_3 = B_1 C_2$ is generated from the existing monomial X_1 and X_2. The consensus operation implemented in the transition table is shown as:

Monomials	*Primary events*		
	A	B	C
X_1	1	1	–
X_2	2	–	2
X_3	–	–	–

≡

Primary events		
A	B	C
1	1	–
2	–	2
–	1	2

The consensus operation on the two monomials (two rows in the table) yields a third row in the critical transition table or a third monomial X_3.

3.3.3. DFM Modelling Process

A DFM model expresses the logical and dynamic behavior of a system with a network of nodes created by discretizing the key system parameters and state variables. These system nodes are then linked together to represent the cause-and-effect and time-dependent relationships which exist among key system parameters. Decision tables within the nodes constructed from the equations governing the system behavior represents the functional mappings among the system parameters *[Yau et al. (1998)]*. The application of DFM within a PRA framework typically involves three essential steps: *[NUREG-6465]*

1. Development of the DFM model for which safety analysis is desired
2. Analysis of DFM model (deductive or inductive)
3. Quantification of the prime implicants.

The DFM model encompasses all the key system parameters and are represented by a network of nodes. The fundamental DFM modelling elements are shown in *Figure 4*. A DFM model utilize these elements to represent the temporal and logical relations that exist among key system parameters, *Yau (1997)*.

Process variable nodes:

They represent physical variables that captures the essential functional behaviour, continuous or discrete of a system. A variable represented by a process variable node is discretized into finite number of intervals. The number of discretized intervals of a state variable is chosen on the basis of balance between the accuracy of the model, and the complexity introduced by higher numbers of variable states. For example, steam generator liquid level discretization.

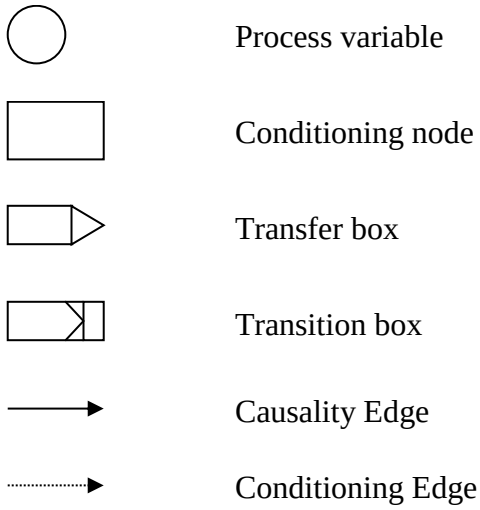


Figure 4: Basic building blocks of DFM Model

Causality edge:

They are used to connect process variable nodes to indicate the existence of a cause-and-effect relationship between the variables described by the nodes. Each causality edge is connected to a transfer box to describe the precise functional relationships among the connected nodes.

Transfer box:

A transfer box represents a transfer function among system parameter nodes. A decision table is associated with each transfer box and is used to quantify the relationships between its input and output system parameters. A decision table is a multi-dimensional matrix whose dimension is equal to one plus the number of its inputs.

Transition box:

A transition box is similar to transfer box with the exception that, a time lag or time transition is assumed to occur between the time when the input variable become true and the time when the output variable associated with the inputs is reached. This time delay is a characteristic of the transition which is being modelled and is treated as an attribute of the transition box.

Conditioning Nodes:

It explicitly identifies component failure states, changes of process operation regimes and modes, and switching actions. These nodes can affect the logic superstructure of a system by modifying the causal relations between process variable nodes.

Conditioning Edge:

They are used to represent true discrete behaviour in the system. Conditioning edge link parameter nodes to transfer boxes, indicating the possibility of using a different transfer function to map input variable into output variable states.

3.3.4. Timed-Fault Tree Construction

Timed-fault tree maybe viewed as sequences of static fault trees at different time steps representing the evolution of logical combinations of events leading to a top event, *Garrett et al. (1995)*. The generation of a timed-fault tree is of prime importance due to the fact that most existing NPP PRAs model are based on classical techniques. In order that the PIs generated from the DFM model are incorporated into an existing plant PRA model, its conversion into FT become necessary. Timed-fault trees can be constructed from a DFM model using by backtracking through the decision tables. Throughout the backtracking process, the following consistency rules are applied: *[Yau (1997); NUREG/CR-6465]*

a. Physical consistency rule

Physical consistency rules are applied to eliminate physically impossible conditions from the timed fault trees. For instance, a state variable cannot be in two different states at the same time step in the time FTs; example, a component cannot be in ‘ON’ and ‘OFF’ states in the same time step. The physical consistency rule is similar to the consistency rules applied in the classical FT. If a system parameter appears twice, in different states, in the same time step and under the same logic AND gate, then all the parameters beneath the first logic AND gate above the second occurrence of the event is pruned from the tree due to physical inconsistency. *[Garrett et al. (1995); Yau (1997)]*.

b. Dynamic consistency rule

Dynamic consistency rules are applied to timed FTs to eliminate branches which cannot occur due to constraints from the dynamic behavior of the system under consideration. Dynamic rules are developed from the analyst's knowledge and assumptions with regard to system dynamic behavior. For instance, if the time step considered in a DFM model is 1 second, and water level in a steam generator was initially at 0%, the water level in the next time step cannot increase to 80% (impossible condition due to system dynamics). Similar to physically consistency rule, the dynamically inconsistent branches, including all of the sub-branches connected to it via the first

parent AND gate are pruned. Further pruning is performed if eliminated branches cause other events to become impossible *[Garrett et al. (1995); Yau (1997)]*.

3.4. Classical Markov Chain

Markov chain enables one to model a sequence of random variables which correspond to the states of a system, and the system state at any point in time is dependent only on the system state in the previous time. Markov chain captures system dynamic nature with its two-fundamental property of characterizing a process evolution by a set of distinct system states and transition between these states. The mathematical model of a Markovian system is thus a collection of system states and the conditional probabilities between these states. The system dynamics is represented by a set of coupled linear differential equations (Chapman-Kolmogorov equation), which can be solved using standard analytical methods.

3.4.1. Formulation of the Markov Chain

Consider a discrete-time (n) stochastic process with a sequence of random variables $X_0, X_1, X_2, X_3, \dots, X_n$ $(X_n, n \geq 0)$ such that each random variable X_n takes values in a discrete set S $(S = \mathbb{N})$ i.e., the state space, then:

$$\mathbb{P}(X_{n+1} = j | X_n = i, X_{n-1} = i_{n-1}, \dots, X_0 = i_0) = \mathbb{P}(X_{n+1} = j | X_n = i)$$
$$\forall n \geq 0;\ j, i, i_{n-1}, \dots \dots, i_0 \in S \tag{3-19}$$

It may be observed from the equation that the process is memoryless, and the next system state is dependent only on the current state. Furthermore, X_n is time homogeneous if:

$$\mathbb{P}(X_{n+1} = j | X_n = i) = (X_1 = j | X_0 = i) = P_{ij} \tag{3-20}$$

i.e., the transition probabilities are independent of n. Thus, for an $(n + m)$ time steps with $n \geq 0, m \geq 0$, the above equation become:

$$\mathbb{P}(X_{n+m} = j, X_n = k | X_0 = i) = {P_{ik}}^n {P_{kj}}^m \tag{3-21}$$

Summing up all the k yields the Chapman-Kolmogorov equation:

$$P_{ij}{}^{n+m} = \sum_{k \in S} P_{ik}{}^{n} P_{kj}{}^{m} \tag{3-22}$$

Also by necessary probability condition:

$$\sum_{j \in S} \mathbb{P}(X_1 = j | X_0 = i) = 1$$

Here, P_{ij} denotes the probability that the chain, whenever in state i transits into the next state j; and takes a value $0 \leq P_{ij} \leq 1$ for all i and j. For each $i \in S$, the transition probability matrix satisfies the following condition (rows of the transition matrix):

$$\sum_{j \in S} P_{ij} = 1 \tag{3-23}$$

3.4.2. Computation of State Probabilities

For a Markovian system with a finite number of discrete-state, the objective is to determine the system state transition probabilities at a given point in time. The stochastic rate of transition among system states is described by a set of ordinary linear differential equations. The state probabilities can be determined by solving the set of coupled Chapman-Kolmogorov differential equations. The general form of the first order linear differential equations is as follow:

$$\frac{dP_i(t)}{dt} = -\sum_{\substack{j=1 \\ j \neq i}}^{n} a_{ij} P_i(t) + \sum_{\substack{j=1 \\ j \neq i}}^{n} a_{ji} P_j(t) \tag{3-24}$$

Where; $n =$ total number of system states

$P_i(t) =$ state probability for the i^{th} state

$a_{ij} =$ the rate of transition from state i to state j (typically component repair and failure rates)

The computation of state probabilities or solution of a Markov model involves three steps:

a) *Model development:* This step involves the development of state transition diagram such as determination of system states, the possible transition between these states, and the transition rates. It may also include labeling of system states in a qualitative manner such as system operational, failure or partial failure.

b) *Generation of ODEs:* The Markovian state transition diagram developed in the preceding step is utilized to generate a set of linear ordinary differential equations that describes the system behavior and characteristics.

c) *Determination of State Probabilities:* The obtained linear ODEs are solved to determine the system states probabilities. Many standard techniques are available for solving the set of ODEs, such as analytical methods, Laplace transform and numerical integration.

For illustration purpose on computation of state probabilities, two cases are considered: (1) Non-repairable system; and (2) Repairable system.

1. A non-repairable system

Consider a non-repairable component with two states, i.e., normal and failed states. The system can only make a transition from operational state to the failed state with a constant failure rate of λ due to its nature of non-reparability. The system can possibly take two states, and can be represented in the transition diagram (*Figure 5*):

State 1: System operational (S_o)

State 2: System failed, or absorbing state (S_1)

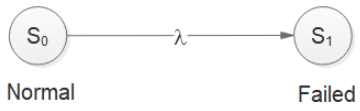


Figure 5: Markov state-transition diagram of a two state component

The system of linear ordinary differential equations (ODEs) from the state-transition diagram are:

$$\frac{dS_0(t)}{dt} = -\lambda S_0(t) \tag{3-25}$$

$$\frac{dS_1(t)}{dt} = \lambda S_0(t) \tag{3-26}$$

With the assumption that the system is fully operational at $t = 0$, $S_0(0) = 1$ and $S_1(0) = 0$, solution of the differential equation, and probability density function of system failure is:

$$S_0(t) = e^{-\lambda t} \text{ and } S_1(t) = (1 - e^{-\lambda t})$$

$$f(t) = \frac{dS_1(t)}{dt} = \frac{d}{dt}\left(1 - e^{-\lambda t}\right) = \lambda e^{-\lambda t} \tag{3-27}$$

From definition, the system reliability and unreliability are simply represented by $S_0(t)$ and $S_1(t)$. For a component with a constant failure rate of $\lambda = 0.005$ failures per hr and a mission time of 1200 hours, the reliability and unreliability of the system is plotted in *Figure 6*.

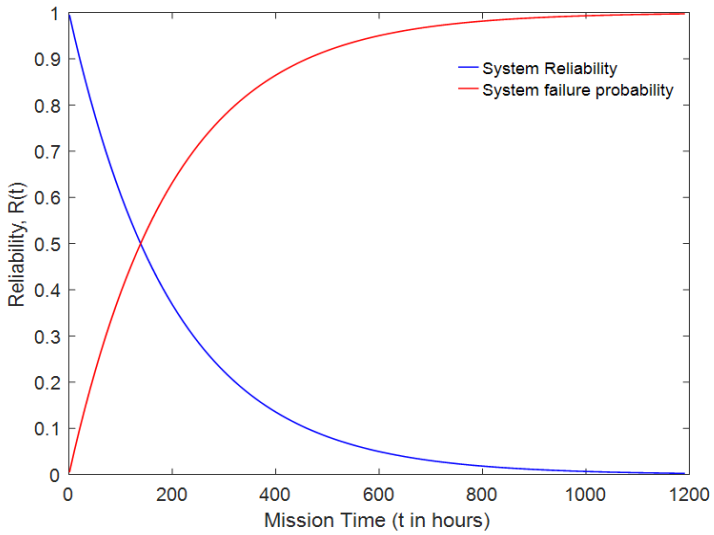


Figure 6: System reliability and unreliability for a two state non-repairable system

2. A repairable system

Similar to the previous case, consider a system with two states i.e., operational and failed, with the difference that the system is repairable. The system can make a transition from operational to failed state with a rate λ, and from failed state to operational state with a repair rate of μ. Hence, the probability that the system will remain in a given state is $(1 - \lambda)$ and $(1 - \mu)$ for state $S_0(t)$ and $S_1(t)$ respectively. The Markov state transition diagram is shown in *Figure 7*.

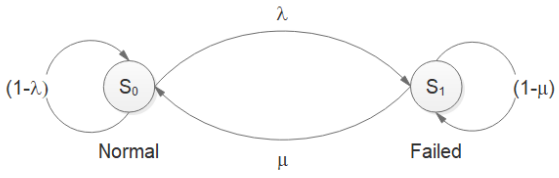


Figure 7: State-transition diagram of a two state component with repair

The linear differential equations describing the time-dependent system characteristics are:

$$\frac{dS_0(t)}{dt} = (1-\lambda)S_0(t) + \mu S_1(t) \tag{3-28}$$

$$\frac{dS_1(t)}{dt} = \lambda S_0(t) + (1-\mu)S_1(t) \tag{3-29}$$

The solution of the above ODEs is:

$$S_0(t) = \frac{\mu}{(\lambda+\mu)} + \frac{\lambda}{(\lambda+\mu)}e^{-(\lambda+\mu)t} \text{ and } S_1(t) = \frac{\lambda}{(\lambda+\mu)}\left(1 - e^{-(\lambda+\mu)t}\right) \tag{3-30}$$

For a component with a constant failure rate of $\lambda = 0.005$ failures per hr, repair rate of $\mu = 0.00125$ repairs per hr and a mission time of 1500 hours, the reliability $(S_0(t))$ and unreliability $(S_1(t))$ of the system is plotted in *Figure 8*.

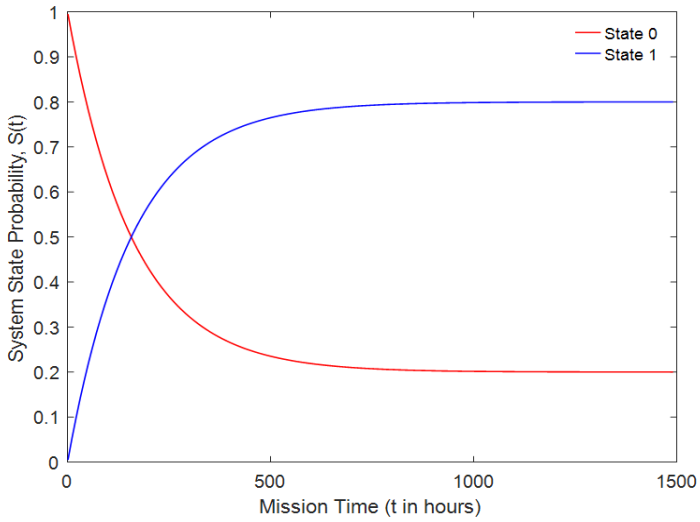


Figure 8: State probabilities for component with repair

3.5. Cell-to-Cell Mapping Technique

Cell-to-Cell Mapping Technique (CCMT) is a systematic procedure to describe the dynamics of both linear and nonlinear systems in discrete time and discretized system state-space, or the subspace of the state variables only, *Hsu (1980)*. The fundamental concept of the methodology is to discretize the state space into a finite number of cells and determine the probability that the system will occupy a discretized cell as time evolve by means of standard numerical integration methods such as Runge-Kutta method. Thus, a state variable represents the dynamics of a system

in discrete quantity that takes every possible value $x \in \mathbb{R}$. Cell mapping technique can be categorized as: simple cell mapping, generalized cell mapping and interpolated cell mapping, *Spek (1994)*. In this thesis, generalized cell mapping is used for benchmark system analysis, due to its efficiency and accuracy of the methodology to predict the long term dynamic behavior of linear and non-linear systems *[Hsu (1980); Spek (1994)]*. CCMT defines the system states in terms of both system configurations i.e., vectors of the discrete states occupied by the system components, and cells occupied by state variables. This allows modelling system configuration (instantaneous) changes upon crossings of threshold values (e.g., a valve closing when the liquid level exceeds a pre-set value). Consider a simple dynamic system governed by the ODE:

$$\frac{dx}{dt} = f(x, \alpha, t) \tag{3-31}$$

Where, x is an N dimensional state variable vector in $\mathbb{R}^N$, α is the K dimensional system parameter vector, t is the time variable, and f is a non-linear function vector. *Equation (3-22)* can be integrated over one period to relate the state of the system at the end of one period to the state at the end of the next period.

$$x(t + \Delta t) = x(t) + \int_{t}^{t+\Delta t} f(x, \alpha) dt \tag{3-32}$$

The above solution for x in discrete point mapping form can be written as:

$$x(n + 1) = G(x(n), \alpha) \tag{3-33}$$

Here, a point $x(n)$ in the state space is mapped by the mapping transition probability matrix (G) after one discrete time step into a point $x(n + 1)$, and hence the name point mapping method. This enables one to represent the system dynamics in finite sequence of discrete system states. This very feature of point mapping of the system dynamics as a continuum of dimension N state space results in poor computational efficiency. This leads to the idea of considering a state variable not as a continuum of points but as a collection of finite number of cells, *Hsu (1980)*. The region of interest Ω in the state space is chosen and partitioned into finite number of subsets $J_i \in \Omega$. Each subset J_i is considered as the smallest entity in the control state space and called a regular or computational cell. All the cells can be labeled with an integer z, from 1 to N_c. The

region outside the control space Γ can also be divided into finite number of cells called the sink cells, from 1 to $\bar{N}_s$.

The computational cells can be discretized or partitioned in several ways and is dependent on the analyst's choice. For example, the coordinate axis of a state variable x_i $(i = 1, 2, \dots, N)$ can be divided into a finite number of intervals with an interval size of h_i. The interval Z_i of the x_i axis is defined to be one which contains all x_i satisfying: *[Hsu (1980)]*

$$(Z_i - 1/2)h_i \leq x_i < (Z_i + 1/2)h_i$$

$$h_i = \frac{x_i^{(max)} - x_i^{(min)}}{M_i}, i = 1, 2, \dots, N \tag{3-34}$$

Where, Z_i is an integer, and M_i is the number of intervals. A N-tuple $Z_i, i = 1, 2, \dots, N$ is called a cell vector of the state space and is denoted by $\mathbb{Z}$. The collection of the regular cells and the sink cell forms a cell state space $S = \{0, 1, \dots, N_c\}$. This way the continuous state space is transformed into a 1-dimensional integer array. *Figure 9* shows an example state space discretization scheme with N_c computational cells in the control space, and $\bar{N}_s$ sink cell in Γ.

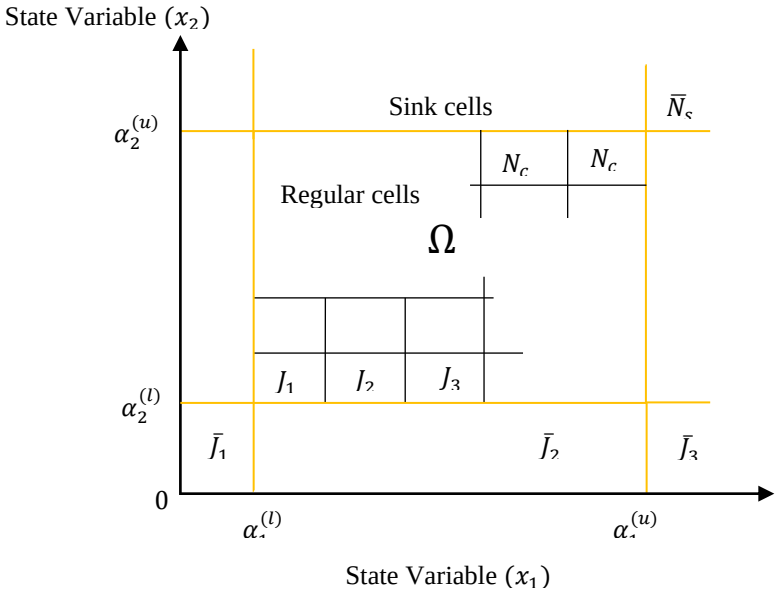


Figure 9: A two dimensional state space discretization

A point $x(x_i, i = 1, 2, \dots, N)$ belongs to a cell $\mathbb{Z}(Z_i = 1, 2, \dots, N)$ if x_i belongs to Z_i for all i. By an appropriate application of rules to identify $x(n)$ and $x(n+1)$ with corresponding computational cells $Z(n)$ and $Z(n+1)$, one can associate to the point-to-point mapping (*Equation 3-24*) a cell-to-cell mapping $\mathbb{C}$. Thus, cell mapping is derived from point mapping by appropriate discretization process.

$$\mathbb{Z}(n+1) = \mathbb{C}(\mathbb{Z}(n), \alpha)$$

$$\mathbb{Z}_i(n+1) = \mathbb{C}_i(\mathbb{Z}(n), \alpha) \tag{3-35}$$

Where $\mathbb{C}$ maps a set of integers to a set of integers, and thus $\mathbb{C}$ is an integer-valued cell mapping.

Note that, it is not necessary for all the computational cells to be identical, rather a computational cell can be of any shape. However, rectangle is the most commonly used shape because of its convenience, *Chen (2004)*. Further *Hsu (1981)* developed the generalized cell mapping theory by removing the restriction in simple cells mapping that a computational cell $\mathbb{Z}(n)$ is mapped by $\mathbb{C}$ into only a single cell $\mathbb{Z}(n+1)$. The generalized theory enables a computational cell $\mathbb{Z}(n)$ to be mapped into several image cells, with each of the image cells having a definite fraction of the total probability. For instance, if a system is in a computational cell $\mathbb{Z}(n)$ at time $t = n$, the evolution of the system state in the next time step $(n+1)$ is given by:

$$\mathbb{Z}^{(1)}(n+1) = p^{(1)}, \mathbb{Z}^{(2)}(n+1) = p^{(2)}, \dots, \mathbb{Z}^{(i)}(n+1) = p^{(i)}\ ; (i = 1, 2, \dots, N)$$

Here, $p^{(1)}$ implies the probability of mapping $\mathbb{Z}(n)$ into cell (1), and $p^{(2)}$ implies the probability of mapping $\mathbb{Z}(n)$ into cell (2), and so forth. i.e., a single computational cell can be mapped into several other cells, or even into itself with some cell-to-cell transition probabilities. Of course, by definition and probability theory:

$$\sum_i p^{(i)} = 1 \tag{3-36}$$

i.e., the sum parameter covers all the possible image cell that a computational cell can take in the next time step. This allows one to describe the system state at any point in time with adequate probabilities of finding a system in several cells. Let:

$S =$ Closed set of cells of interest (cells within the control space)

$\psi_i(n) =$ the probability of the state of the system being in cell i at $t = n$

The vector $\boldsymbol{\psi}(n)$ with component $\psi_i(n), i = 1, 2, \dots, N$ is called the cell probability vector.

$p_{ij} =$ Probability of cell j being mapped into cell i in one mapping time step

$P =$ Transition probability matrix with components p_{ij}

The following properties of $\psi_i(n)$ and p_{ij} is inferred from the above, as is given by:

$$\sum_{i \in S} \psi_i(n) = 1 \tag{3-37}$$

$$\sum_{i \in S} p_{ij} = 1 \tag{3-38}$$

$\psi_i(n) \geq 0$ and $p_{ij} \geq 0$

Thus, the system evolution can be represented by:

$$\boldsymbol{\psi}(n+1) = P\boldsymbol{\psi}(n) \tag{3-39}$$

For a given initial cell probability vector $\boldsymbol{\psi}(0)$ at $n = 0$, the subsequent system evolution is simply given by:

$$\boldsymbol{\psi}(n) = P^n\boldsymbol{\psi}(0) \tag{3-40}$$

The mapping probability matrix, *P* completely describe the system evolution and dynamics in the cell space. Thus, useful information of the system can be obtained by solving the transition mapping probability matrix *P*. It can be observed that the above equations transformed the system dynamics into a finite state Markov model with stochastic matrix *P* governing the system dynamics.

3.6. Coupled Markov-CCMT Model

The coupled Markov-CCMT model integrates the classical Markov model with CCMT to represent the possible coupling between failure events that can originate from the dynamic interactions between system components and controlled state variables, and among the different

constituents of the system *[NUREG-6985]*. *Figure 10* provide the framework and methodology implementation flowchart.

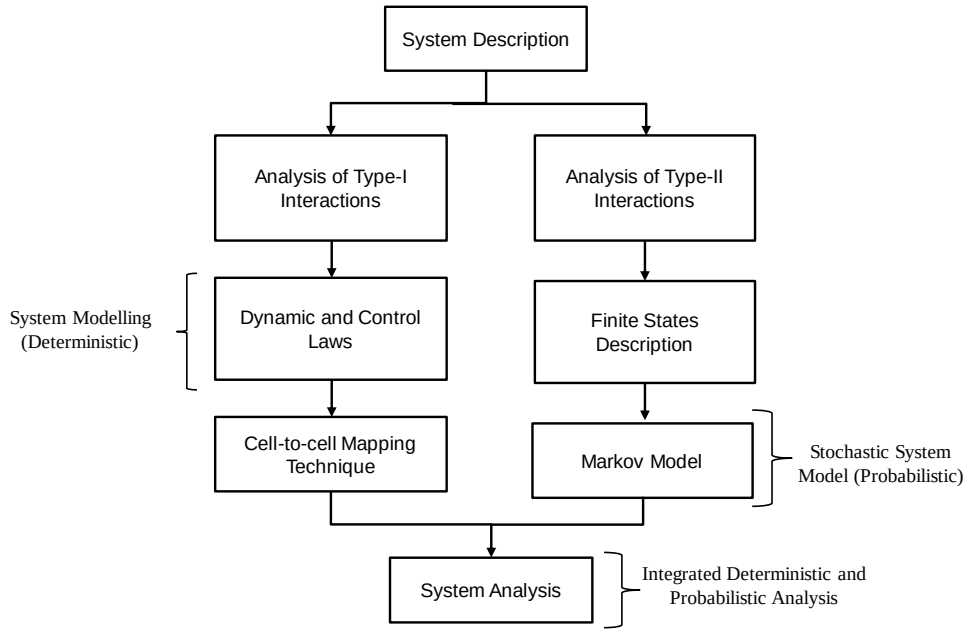


Figure 10: Coupled Markov-CCMT flowchart for integrated system analysis

The coupled Markov-CCMT models a system evolution in discrete time through the probability $P_{n,j}(t)$ that the state variables are in a predefined regions or cells V_j in the state space at time $(k+1)\Delta t$ with the components state combination n (e.g., valve failed-closed), given that the state variables was in cell $V_{j'}$ at time $k\Delta t$ with a component state combination n', *Hassan and Aldemir (1990)*. The dynamic behavior of the system is usually described by a set of differential or algebraic equations, as well as the set of control laws.

1. Model Assumptions

The methodology is based on the following assumptions *[Aldemir (1987); Hassan and Aldemir (1990); Belhadj and Aldemir (1992)]:*

1. The system configuration or components state do not change during the time interval $[k\Delta t, (k+1)\Delta t]$ but possibly at $(k+1)\Delta t$;

2. For a given component state combination n and cell V_j, $P_{n,j}(k\Delta t)$ is uniformly distributed over V_j;
3. If the modeling is conducted in the state variable state space, no two controlled variable trajectories arrive at the same point in state space at the same time and move in different directions for the same component state combinations.

Assumptions 1 and 2 lead to an approximation of the probabilistic system dynamics. Assumption 1 also leads to an approximation of the failure characteristics of the components. Under these assumptions, the probabilistic evolution of the system in time $P_{n,j}(k+1)\Delta t$ can be determined recursively following an inductive Markov-CCMT implementation: *[Aldemir (1987); Hassan and Aldemir (1990)]*

$$P_{n,j}((k+1)\Delta t) = \sum_{n'=1}^{N}\sum_{j'=1}^{J} g(j/j',n',k\Delta t).h(n/n',j' \to j,k\Delta t).P_{n',j'}(k\Delta t) \tag{3-41}$$

Where;

$$q_{n,j}^{n',j'}(k\Delta t) = g(j/j',n',k\Delta t).h(n/n',j' \to j,k\Delta t) \tag{3-42}$$

Therefore;

$$P_{n,j}((k+1)\Delta t) = \sum_{n'=1}^{N}\sum_{j'=1}^{J} q_{n,j}^{n',j'}(k\Delta t).P_{n',j'}(k\Delta t) \tag{3-43}$$

The above equation is equivalent to the Chapman–Kolmogorov equation in discretized state space and discretized time. Here;

$\Delta t =$ Cell-to-cell mapping time step

$P_{n',j'}(k\Delta t) =$ Pr{Controlled variables are in cell j', and component state combination is in state n' at time $t = k\Delta t$

$P_{n,j}((k+1)\Delta t) =$ Pr{Controlled variables are in cell j, and component state combination is in state n at time $t = (k+1)\Delta t$

$g(j/j',n',k\Delta t) =$ Pr{Controlled variables are in cell j at time $(k+1)\Delta t$ given that the controlled variables are in cell j' at time t

$h(n/n', j' \rightarrow j, k\Delta t) =$ Pr{Component state combination is in state n at time $(k+1)\Delta t$ given that the component state combination is in state n' at time t, and the state variables move from cell j' to cell j during $[k\Delta t, (k+1)\Delta t]$.

$q_{n,j}^{n',j'} =$ Elements of the transition matrix for the Markov model

Since, V_j cover the whole control variable state space and N includes all possible component state combinations, by essence of probability theory:

$$\sum_{n=1}^{N}\sum_{j=1}^{J} P_{n,j}(t) = 1 \tag{3-44}$$

$$\sum_{n'=1}^{N}\sum_{j'=1}^{J} q_{n,j}^{n',j'}(k\Delta t) = 1 \tag{3-45}$$

The parameter $q_{n,j}^{n',j'}$ is a conditional probability that accounts for the simultaneous occurrence, and interactions between the state variables and the system components which are statistically dependent. Hence, to compute $q_{n,j}^{n',j'}$, the conditional probability between g and h have to be known. However, neither g or h can be determined individually due to the statistical dependency *Aldemir (1987)*. To overcome this drawback, the methodology approximates g using Assumption 1 that the system state does not change during the short interval $[t, (t+k\Delta t)]$ but can simultaneously change during $(t+k\Delta t)$.

2. *Determination of cell-to-cell transition probabilities, $g(j/j', n', k\Delta t)$*

The cell-to-cell transition probabilities depends on the system dynamics, the deterministic control laws, and the possible system configurations. $g(j/j', n', k\Delta t)$ represents the system dynamics under normal and failed state of the components/sub-systems, i.e., conditional probability that the state variables are in cell V_j at time $t = (k+1)\Delta t$ given that:

- The state variables are in cell $V_{j'}$ at time $t = k\Delta t$
- The component state combination is in n' at time t.

The $g(j/j', n', k\Delta t)$ can be found from *[Hassan and Aldemir (1990); Belhadj et al. (1992)]:*

$$g(j/j^{'},n^{'},k\Delta t)=\begin{cases}\int_{V_{j^{'}}}\frac{dx^{'}}{v_{j^{'}}}e_j\left(\tilde{x}_{(k+1)\Delta t}(x^{'},\alpha_{n^{'}},k\Delta t)\right);\ if\ V_{j^{'}}\in R\\ \delta_{j,j^{'}} \qquad\qquad\qquad\qquad\qquad\qquad Otherwise\end{cases} \tag{3-46}$$

Where;

$$\delta_{j,j^{'}}=Kronecker\ delta=\begin{cases}1 & ;if\ j^{'}=j\\ 0 & otherwise\end{cases} \tag{3-47}$$

$$e_j(x)=Step\ function=\begin{cases}1 & ;if\ \tilde{x}_{(k+1)\Delta t}\in V_j\\ 0 & otherwise\end{cases} \tag{3-48}$$

Where; $x'=$ Initial value of the state variable at time $t=k\Delta t$

$\tilde{x}_{(k+1)\Delta t}(x',\alpha_{n'},k\Delta t)=$ The location of the system in the state variable space at time $t=(k+1)\Delta t$, given that the system location is x' at time $t=k\Delta t$ and the component state combination is at n'. This system trajectories is typically determined using system code or simulator.

$v_{j'}=$ Volume of the cell $V_{j'}$

$\frac{dx'}{v_{j'}}=$ The probability that the state variables are within the infinitesimal volume element dx' around x' at time t. This probability is constant throughout each $V_{j'}$ (uniform distribution).

Since the functional form of $\{\tilde{x}(x',\alpha_{n'},k\Delta t)\}$ is generally unknown, the integral in the above equation is evaluated numerically as follow: *[NUREG-6942]*

- Partition the cell j' into N_p number of sub-cells
- Choose the midpoint of each sub-cell as initial conditions over the time interval $k\Delta t\le t\le(k+1)\Delta t$ under the assumption that the component state combination remains at n' at all times during $k\Delta t\le t\le(k+1)\Delta t$
- Observe the number of arrivals in N_{p+1} at time $t=(k+1)\Delta t$, $\tilde{x}_{(k+1)\Delta t}(x',\alpha_{n'},k\Delta t)$
- Compute the cell-to-cell transition probabilities, $g(j/j',n',k\Delta t)=\frac{N_p}{N_{p+1}}$

3. *Determination of the component state transition probabilities,* $h(n/n',j'\to j,k\Delta t)$

The stochastic behavior of system components is represented by $h(n/n',j'\to j,k\Delta t)$, i.e., the conditional probability that component state combination is in n at time $t=(k+1)\Delta t$, given:

- The component state combination was at n' at time $t = k\Delta t$; and
- The state variables make a transition from cell $V_{j'}$ to cell V_j during $k\Delta t \leq t < (k+1)\Delta t$

For system components with statistically dependent failures, the probabilities $h(n/n', j' \rightarrow j, k\Delta t)$ is given by the products of the individual component failure or non-failure probabilities during the mapping time step from $k\Delta t$ to $(k+1)\Delta t$, i.e.,

$$h(n/n^{'}, j^{'} \rightarrow j, k\Delta t) = \prod_{m=1}^{M} c_m(n_m/n'_m, j^{'} \rightarrow j, k\Delta t) \tag{3-49}$$

Whereas for statistically independent failure, the probabilities $h(n/n', j' \rightarrow j, k\Delta t)$ is given by:

$$h(n/n^{'}, j^{'} \rightarrow j, k\Delta t) = \prod_{m=1}^{M} c_m(n_m/n'_m, k\Delta t) \tag{3-50}$$

Where, $m =$ Number of components or units in the system under consideration

$c_m(n_m/n'_m, j' \rightarrow j, k\Delta t)$: The component state combination at the individual component or unit level and is the transition probability component m from the state combination $n'_m \rightarrow n_m$ during the time step $[k\Delta t, (k+1)\Delta t]$.

$c_m(n_m/n'_m, j' \rightarrow j, k\Delta t)$ can be determined systematically from given component failure rates, and the set of possible failure modes S_m for a system transition from cell $V_{j'}$ to cell V_j $\{i.e., S_m(j', j)\}$ for each $n_m, n_{m'}, j'$ and j.

$$c_m(n_m/n^{'}_m, j^{'} \rightarrow j, \Delta t) = 1 - \sum_{n^{'}_m \in S_m(j^{'}, j)} \lambda_{m, n^{'}_m} . \Delta t \rightarrow if\ n_m \in normal\ state \tag{3-51}$$

$$c_m(n_m/n^{'}_m, j^{'} \rightarrow j, \Delta t) = \lambda_{m, n^{'}_m} \Delta t \rightarrow if\ n_m \in failed\ state \tag{3-52}$$

4. *Computation of the system transition probability matrix,* $q_{n,j}^{n',j'}(k\Delta t)$ and $P_{n,j}(k\Delta t)$

The stochastic transition probability matrix $q_{n,j}^{n',j'}(k\Delta t)$ describes the probabilistic dynamic system behavior and is a function of both $h(n/n', j' \rightarrow j, k\Delta t)$ and $g(j/j', n', k\Delta t)$ with the

matrix having a $N \times J$ dimension. The elements of $q_{n,j}^{n',j'}(\Delta t)$ for the state variable within the control state space is given by:

$$q_{n,j}^{n',j'}(\Delta t) = \frac{1}{v_{j'}} \prod_{m=1}^{M} c_m(n_m/n'_m, j' \to j, \Delta t) \int_{V_{j'}} dv' e_j\{\tilde{x}_{(k+1)\Delta t}(x', \alpha_{n'}, k\Delta t)\};\ j', j = 1, \dots J$$

Thus, the probability of the system being in state n and at cell j is simply given by:

$$P_{n,j}((k+1)\Delta t) = \sum_{n'=1}^{N} \sum_{j'=1}^{J} q_{n,j}^{n',j'}(\Delta t)\, P_{n',j'}(k\Delta t) \tag{3-53}$$

The Markov-CCMT model will be used for its comparison with the classical techniques as well as with the DFM to meet the objectives of the research. Furthermore, the treatment of transition probability matrix will be evaluated using a different approach as well as combination of techniques, such as state merging and transforming the transition matrix to a canonical form.

CHAPTER 4: APPLICATION OF CLASSICAL AND DYNAMIC METHODOLOGIES

This chapter implements the classical (ET/FT and Markov model) and dynamic PSA (DFM and Markov-CCMT) methodologies to a benchmark liquid level control system (*Aldemir, 1987)*. *Section 4.1* provide a detail description of the benchmark system (BS), system dynamics and associated control laws. *Section 4.2* and *4.3* presents analysis of the BS using classical ET/FT analysis. *Section 4.4* provide a detail time-dependent analysis (qualitative and quantitative) of the BS using a classical Markov model. *Section 4.5* presents application of DFM to the BS, and subsequent generation of timed-fault trees from the model. *Section 4.6* presents a detail modelling and analysis of the BS using tightly coupled Markov-CCMT model to generate dynamic accident sequence. *Section 4.7* concludes the chapter with observations made and perform a detail comparison of classical and dynamic PSA techniques.

4.1. The Benchmark System Description

The benchmark example system consists of a fluid hold up tank with three (3) independent fluid level control units. Each control unit consists of a separate level sensor that measures the fluid level in the tank, and whose output is an input to the control units. *Figure 11* depicts the schematic diagram of the BS. Operational states of the control units are dependent on the feedback signal from level sensors. Mission of the BS is to maintain liquid level in the tank to a predefined control region. The system failure occurs when the liquid level in the tank drained or overflows. Unit-1 and Unit-2 are the two liquid supply units (inflow), whereas Unit-3 is the liquid drain unit (outflow). Each unit can be thought of as a controller which switches the units "ON" or "OFF" based on the feedback signal from the level sensors. There are four (4) possible states for each control units: ON, OFF, fail-open and fail-closed.

The tank liquid level is discretized into an arbitrary number of mutually exclusive regions or intervals, with the nominal level at 0 meters. All the other possible levels are measured against the reference level. The maximum level of the tank is 3 meters (at point b) and the minimum level of the tank is -3 meters (at point a). The system fails if the state variable is outside the defined control space. Within the maximum and minimum range, two set-points are defined at a_1 (-1 meter) and b_1 $(+1 \text{ meter})$. These set-points partition the control space into three (3) regions

for system operation. The discretization scheme is show in *Table 9*. Each control region defines a specific operating states of the units, and hence the system configuration.

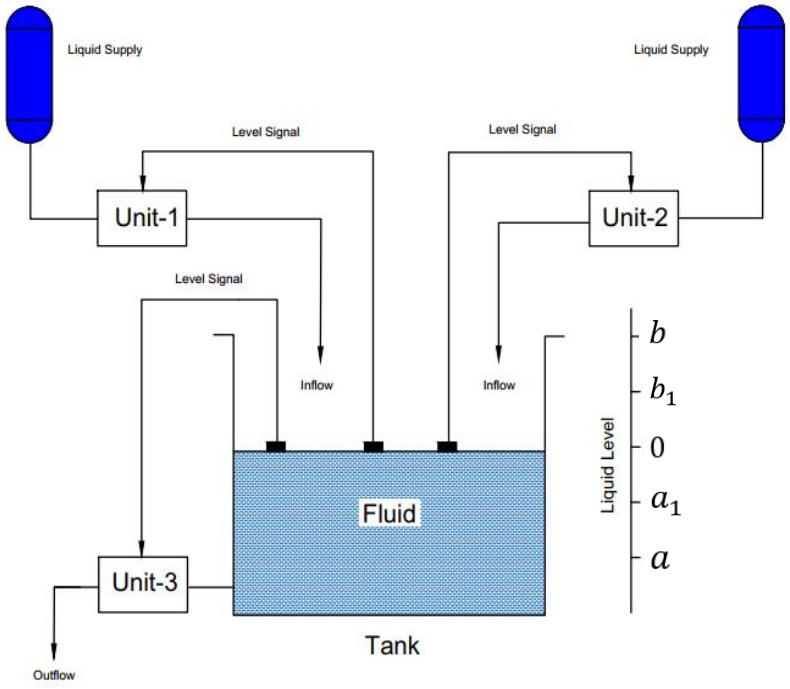


Figure 11: The benchmark liquid level control system

Table 9: State variable discretization scheme

Control Region	*Liquid Level* (x)	*Liquid Level (in meters)*
Region 5	$x > b$	Overflow
Region 4	$b_1 \leq x \leq b$	$+1 \leq x \leq +3$
Region 3	$a_1 \leq x \leq b_1$	$-1 \leq x \leq +1$
Region 2	$a \leq x \leq a_1$	$-3 \leq x \leq -1$
Region 1	$x < a$	Drained

Region 1 and Region 5 are considered as the absorbing states in which the top event or system failure occurs with drained and overflow respectively. No control action is taken once the system enters the absorbing states. During normal operating condition, the liquid level is in Region 3 $(a_1 \leq x \leq b_1)$. In this region, Unit-1 and Unit-3 are 'ON' state balancing the liquid flow-in and flow-out of the tank, whereas Unit-2 is in standby mode or "OFF" state. When the level transit

from Region 3 to Region 2, Unit-3 receives a signal to turn off whereas Unit-1 and Unit-2 receives a signal to turn on. Each time the level makes a transition from one region to the other region, the control system demands certain deterministic actions to all the normal units in order to bring back the level to the nominal region or at least maintain the system from entering into Region 1 and 5. The deterministic control laws are given in *Table 10*.

Table 10: Control laws for the benchmark system

Liquid Level (x)	*Control Unit State*		
	Unit-1	*Unit-2*	*Unit-3*
$x > b$	-	-	-
$b_1 < x$	OFF	OFF	ON
$a_1 \leq x \leq b_1$	ON	OFF	ON
$x < a_1$	ON	ON	OFF
$x < a$	-	-	-

A. Assumptions

The following assumptions are made for the purpose of analysis, which includes:

1. The liquid supply is inexhaustible, and failures of unit are non-repairable;
2. The control units have discrete states and are nominally independent;
3. The response is instantaneous, and the time delay is negligible;
4. The probability of a unit failing-closed and failing-open are assumed to be equally likely with the failure rates given below *(Aldemir, 1987).*

   $\lambda_1^{fc}(failed\ closed) = \lambda_1^{fo}(failed\ open) = 2.283 \times 10^{-3}/hr.$

   $\lambda_2^{fc}(failed\ closed) = \lambda_2^{fo}(failed\ open) = 2.857 \times 10^{-3}/hr$

   $\lambda_3^{fc}(failed\ closed) = \lambda_3^{fo}(failed\ open) = 1.563 \times 10^{-3}/hr$

B. Benchmark System Dynamics

The tank is considered to be at the nominal level (Region 3) at the start of the system operation. If any transient occurs due to failure of a unit, the state variable can transit out of the nominal control region which can lead to system failure or can be kept within control space depending on the operational states of the remaining units. The state variable transition to failure space is

dependent on the initial level as well as the rate of change of liquid level in the tank. Since the initial liquid level is a known input, the element of interest is the rate of liquid level change. The system dynamics is given by:

$$\frac{dx(t)}{dt} = f_n(x) \tag{4-1}$$

Where; $x =$ The liquid level in the tank

$f_n =$ Rate of change of liquid level as a function of control unit states (n).

Table 11 list f_n with respect to the unit states, where $\hat{x}_1, \hat{x}_2$ and $\hat{x}_3$ are the flowrates from Unit-1, Unit-2 and Unit-3 respectively. The flowrates at states fail-open and fail-closed for the units are not depicted in the table since the rate of change of level are the same with units' states ON and OFF respectively. Considering an infinitesimally small time step (Δt) relative to the system dynamics and failure rate of the units, and with the assumption that the system configuration does not change during this small interval, the liquid level in the tank at $(t + \Delta t)$ can be represented by the following equation.

$$x(t + \Delta t) \approx x'(t) + f_{n'}(x').\Delta t \tag{4-2}$$

Where; $x'(t) =$ Liquid level at time t

$f_{n'}(x') =$ Rate of change of liquid level as a function of system configuration n'

$x(t + \Delta t) =$ Liquid level at time $(t + \Delta t)$

Table 11: Total net fluid flow into/out of the tank

Control Unit State			*Rate of Level change* (f_n)	*Net fluid flow*
Unit-1	*Unit-2*	*Unit-3*		
ON	ON	ON	$\hat{x}_1 + \hat{x}_2 - \hat{x}_3$	$\hat{x}$
ON	ON	OFF	$\hat{x}_1 + \hat{x}_2$	$2\hat{x}$
ON	OFF	ON	$\hat{x}_1 - \hat{x}_3$	0
ON	OFF	OFF	$\hat{x}_1$	$\hat{x}$
OFF	ON	ON	$\hat{x}_2 - \hat{x}_3$	0
OFF	ON	OFF	$\hat{x}_2$	$\hat{x}$

OFF	OFF	ON	$-\hat{x}_3$	$-\hat{x}$
OFF	OFF	OFF	0	0

As mentioned earlier, the system is assumed to be in the nominal control region, where Unit-1 is ON, Unit-2 is OFF and Unit-3 is ON. This operating condition will continue until one of the units make a transition from normal to failure states. This transition to failure state will cause the fluid level to deviate from its nominal value, which can be level high or level low depending on which unit has failed and to which failure state. The system will change its configuration based on the new control region with the objective to bring back the level to its nominal value. The new system state may or may not result to a stable condition.

4.2. Fault Tree Analysis of the Benchmark System

Fault tree analysis for the benchmark system can be performed in two different ways depending on the definition of the top events. Fault trees for:

1. Binary unit/system state (normal or failure state);
2. Multiple unit/system states (normal, failed-closed and failed-open states).

Fault tree following the first approach (binary states) can be constructed as shown in *Figure 12*. The top event probability is evaluated as:

Top Event $= G_1G_2 + G_3 = 3.15 \times 10^{-3}$

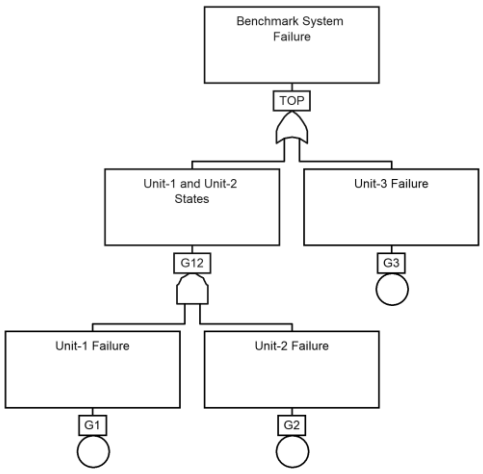


Figure 12: Fault tree for the benchmark system failure (binary state)

The second approach is necessary when top events are defined in terms of system state or state variable magnitudes. Since the benchmark system failure is defined in terms of state variable magnitude, it is a more natural way to construct FTs using the second approach. FTs for the benchmark system overflow and drained as shown in *Figure 13* and *Figure 14* respectively.

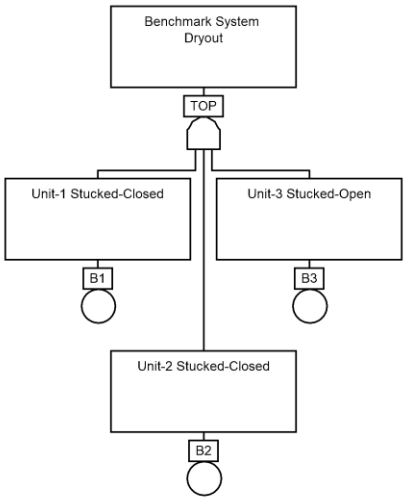


Figure 13: Fault tree for the benchmark system drained

Minimal cut-set for BS drained $= B_1B_2B_3 = 1.02E-08$

Minimal cut-set for BS overflow $= B_1B_2 + B_1B_3 + B_2B_3 = 1.455E-05$

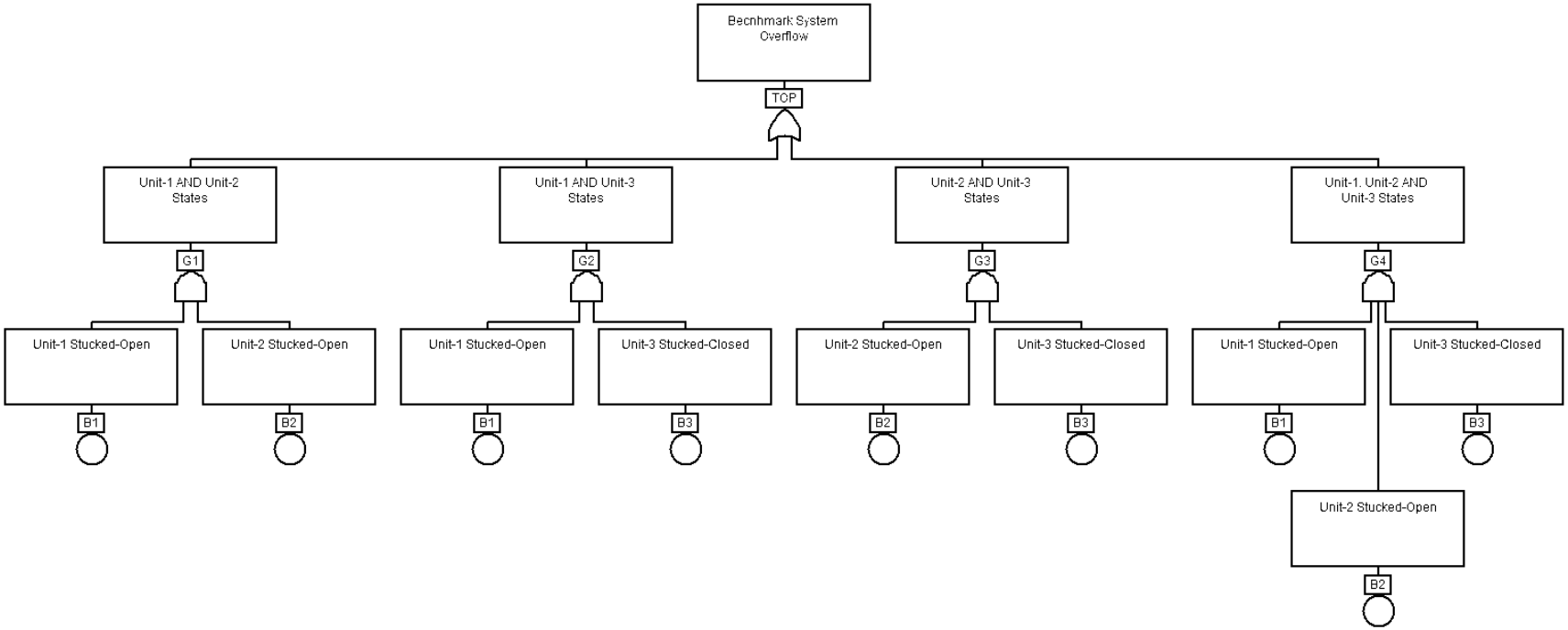


Figure 14: Fault Tree for the Benchmark System Overflow

4.3. Event Tree Analysis of the Benchmark System

Provided an initiating event (IE), the state ordering or sequential operation of the frontline systems/units is fixed for a classical ET. The unit response, accident sequence and hence the final system state is significantly affected by the modes of unit failure. Furthermore, the sequence of unit state transition can affect the type of system failure or may lead the system to a quasi-stable state. Whereas in case of a FT, all sequence or cut-sets result to a system failure which is not necessarily true in this case. Since the benchmark system responds dynamically to any IEs, an ET is a more essential approach than FT for modeling the benchmark system. The unit failure modes have been explicitly modelled in the ET since different modes lead to different system states. For analysis purpose, hardware-oriented ETs are constructed for the system's response to the following IEs. These IEs was selected since they possess an interesting case and reveals some limitations of the technique.

1. Unit-1 failed-closed and failed-open;
2. Unit-2 failed-open;
3. Unit-3 failed-closed.

For the ETs constructed, quantification of the accident sequence is straight forward as can be observed in *Figure 15*. The analysis was performed using CAFTA code. Only the first case is illustrated for discussion purpose.

a. Unit-1 Failed-Closed (λ_1^{fc})

Event tree with an IE probability of $\lambda_1^{fc} = 2.28 \times 10^{-3}/hr$ is depicted in *Figure 15*. It can be observed that with an IE Unit-1 failed-closed, probability of the benchmark system failing by overflow and drained condition is equally likely with a probability of $1.02\mathrm{E}-08/hr$. Hence, there is a 50% probability that the system will overflow or drained if unit-1 failed-closed. The argument for this can be made as; given that unit-1 failed closed, liquid level in the tank will decrease with time since unit-3 is in normal state. However, when the liquid level transit below the nominal region, unit-2 will be turned on and unit-3 will be turned off as defined in the control laws. This action of unit-2 and unit-3 will cause the liquid level to rise till it reaches the nominal region. Once the level is in the nominal region, unit-2 will be tuned off and hence the level will start to decrease again. This oscillation will take place until one of the two units failure occur. Given that the unit-

3 and unit-2 failing closed and failing open respectively have the same probabilities, it can be concluded that there is a 50% probability for system overflow and drained scenario. Also, it is evident that the system will be operating mostly around liquid level a_1. Hence, after a qualitative analysis of the evolution of state variable, one may conclude that it is more likely for the system to fail by drained than overflow, or at least it will take more amount of time of the system to overflow for this particular scenario. However, this conclusion can only be definitive and quantified when the analyst has information about the timing of failure event.

Note that the ordering of the unit response to the IE have been fixed a prior by the analyst, i.e., unit-3 response first to the IE and then the unit-2. From the figure, it can be observed that Unit-3 being normal will lead the system to a stable or quasi stable state. Unit-3 failing open and unit-2 failing close will lead the system to a drained scenario. Similarly, unit-3 failing-closed and unit-2 failing open will result to a system overflow condition. It is important to note here that the failure of two (2) or three (3) units does not necessarily result to a system failure condition. For instance, the sequence Unit-1 failed-closed AND Unit-3 failed-open AND Unit-1 failed-open lead to a system quasi-stable scenario, i.e., the liquid level in the nominal region even after a failure of all the units. In fact, only two (2) event sequence lead to a system failure condition out of nine (9) sequences generated.

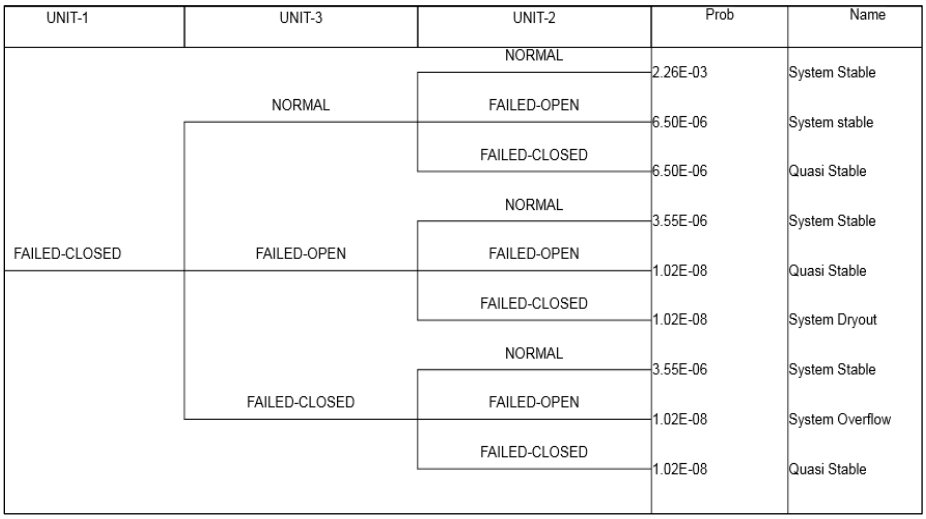


Figure 15: Event tree for initiating event "Unit-1 Failed-Closed"

The probability of the benchmark system overflow or drained scenario can be calculated by summing up all the accident sequences leading to a system overflow or drained respectively:

System overflow $= 2.91E-05$

System drained $= 3.06E-08$

It is evident that ET approach provide the analyst with more information about the dynamic system response and the possible end states compared to the FT technique. However, as mentioned earlier, the event ordering is fixed which may not be true in some scenarios. Again, the system response is dependent on evolution of the state variables when the next failure occurs in the sequence, timing of the failure event and pre-defined deterministic control laws. For example, Unit-2 may respond first in contrast to Unit-3 for IE unit-1 failed-closed. The ETs presented above explicitly models the different failure modes of the units, since the change in failure modes can result to a different system states or type. Thus, ETs must be constructed for each IEs, which becomes tedious for a system with several IEs. However, this approach enables the analyst to observe all possible scenarios for an IE, which can further be treated using well-established system codes. For instance, an accident sequence may lead the BS to quasi stable or failure state; however, the time taken from the system to reach these states, and final liquid level in the tank can be determined only if the system dynamics is considered.

4.4. Markov Model of the Benchmark System

This section presents the time-dependent stochastic modelling of the BS using discrete-state discrete-time Markov chain. The time dependent behavior of the BS can be modelled by constructing state transition diagrams consisting of system states and possible transitions among these states. State ordering is explicitly modeled in defining the possible transitions in/out of a system states. Thus, Markov model provide a possible solution to overcome the limitations of capturing the time element and state ordering which was encountered in FT/ET techniques. Furthermore, the methodology provides a superior approach and solution for modeling components/units with multiple failure modes and system states. In constructing the Markov transition diagram, some assumptions were made, including:

1. The units are non-repairable, and the units fail only once in a given failure sequence, i.e., once a unit failure mode have occurred, other modes cannot occur for the same unit. This assumption is true since units failing closed/open have different effect on the system and must be modelled separately;

2. The system is initially at normal operating condition, i.e., all units are normal and liquid level is at the nominal region at $t = 0$;
3. The system states are defined uniquely by distinct combination of the individual unit states and are statistically independent.

At this point it should be made clear that for a unit in normal state, the unit can either be in ON or OFF depending on the user defined control laws, which in turn is characterized by the liquid level in the tank. Hence, the ON and OFF states can be combined and represented by a normal state, as was done while constructing the transition diagram *(See Figure 16).* Since each of the units have three (3) distinct states, the total number of distinct states that the system can have is:

$$N = (Possible\ failure\ modes)^{(number\ of\ units)} = 3^3$$

$$N = 27 \text{ system states}$$

The 27 distinct system states can be written as an ordered set of distinct unit state or sub-sets for ease of system modelling with Markov chain. Consider, $n_1 = 0$ implies Unit-1 in normal state; $n_1 = 1$ implies Unit-1 in failed-closed state; and $n_1 = 2$ implies Unit-1 in failed-open state. Similarly, the same is true for Unit-2 and Unit-3. The possible unit state combinations which was used to construct the transition diagram is presented in *Table 12.*

The transition diagram starts with all the units being in normal state at $t = 0$, i.e., system state $n = 0$. All the possible transition out of the normal system state are the sum of all the individual failure modes of each unit, i.e., system states with a single failure. Similarly, as the time step increase the transition from a single unit failure to two (2) unit failure can occur, and subsequently from a two (2) unit failure to three (3) unit failure, i.e., each additional failure generates a new system state. Hence, the transition diagram can be represented in a layer wise, with four (4) possible transition layers in the Markov model (*See Table 13).*

Table 12: Possible unit state combination

Individual Unit States			*System States* (n)		*Possible end states*
Unit-1 (n_1)	*Unit-2* (n_2)	*Unit-3* (n_3)			
$n_1 = 0$	$n_2 = 0$	$n_3 = 0$	0	$P_0(t)$	System stable
$n_1 = 1$	$n_2 = 0$	$n_3 = 0$	1	$P_1(t)$	System stable
$n_1 = 2$	$n_2 = 0$	$n_3 = 0$	2	$P_2(t)$	System stable
$n_1 = 0$	$n_2 = 1$	$n_3 = 0$	3	$P_3(t)$	System stable
$n_1 = 0$	$n_2 = 2$	$n_3 = 0$	4	$P_4(t)$	System stable
$n_1 = 0$	$n_2 = 0$	$n_3 = 1$	5	$P_5(t)$	System stable
$n_1 = 0$	$n_2 = 0$	$n_3 = 2$	6	$P_6(t)$	System stable
$n_1 = 1$	$n_2 = 1$	$n_3 = 0$	7	$P_7(t)$	System quasi-stable
$n_1 = 1$	$n_2 = 2$	$n_3 = 0$	8	$P_8(t)$	System quasi-stable
$n_1 = 1$	$n_2 = 0$	$n_3 = 1$	9	$P_9(t)$	System quasi-stable
$n_1 = 1$	$n_2 = 0$	$n_3 = 2$	10	$P_{10}(t)$	System quasi-stable
$n_1 = 2$	$n_2 = 1$	$n_3 = 0$	11	$P_{11}(t)$	System quasi-stable
$n_1 = 2$	$n_2 = 2$	$n_3 = 0$	12	$P_{12}(t)$	*System Overflow*
$n_1 = 2$	$n_2 = 0$	$n_3 = 1$	13	$P_{13}(t)$	*System Overflow*
$n_1 = 2$	$n_2 = 0$	$n_3 = 2$	14	$P_{14}(t)$	System quasi-stable
$n_1 = 0$	$n_2 = 1$	$n_3 = 1$	15	$P_{15}(t)$	System quasi-stable
$n_1 = 0$	$n_2 = 1$	$n_3 = 2$	16	$P_{16}(t)$	System quasi-stable
$n_1 = 0$	$n_2 = 2$	$n_3 = 1$	17	$P_{17}(t)$	*System Overflow*
$n_1 = 0$	$n_2 = 2$	$n_3 = 2$	18	$P_{18}(t)$	System quasi-stable
$n_1 = 1$	$n_2 = 1$	$n_3 = 1$	19	$P_{19}(t)$	System quasi-stable
$n_1 = 1$	$n_2 = 1$	$n_3 = 2$	20	$P_{20}(t)$	*System Dryout*
$n_1 = 1$	$n_2 = 2$	$n_3 = 1$	21	$P_{21}(t)$	*System Overflow*
$n_1 = 1$	$n_2 = 2$	$n_3 = 2$	22	$P_{22}(t)$	System quasi-stable
$n_1 = 2$	$n_2 = 1$	$n_3 = 1$	23	$P_{23}(t)$	*System Overflow*
$n_1 = 2$	$n_2 = 1$	$n_3 = 2$	24	$P_{24}(t)$	System quasi-stable
$n_1 = 2$	$n_2 = 2$	$n_3 = 1$	25	$P_{25}(t)$	*System Overflow*
$n_1 = 2$	$n_2 = 2$	$n_3 = 2$	26	$P_{26}(t)$	*System Overflow*

The system states are depicted systematically in a sequential layer representation. The first possible transition from normal system state is presented in *Figure 16.* Note that the order of failures should be strictly followed when generating failure sequences from the model because the variation in state ordering can produce different end states. Since the number of system states and the number of possible transition from one state to the other state is large, it is easier to construct the transition diagram considering a specific failure mode. *Figure 17* depict the possible transition path from Layer-1 to Layer-3 (only one path depicted for illustration purpose).

Table 13: Transitional layers representation of system states

No.	*Layers*	*System State*
1.	Layer-0	All units in normal state
2.	Layer-1	One (1) unit failure
3.	Layer-2	Two (2) units failure
4.	Layer-3	Three (3) units failure

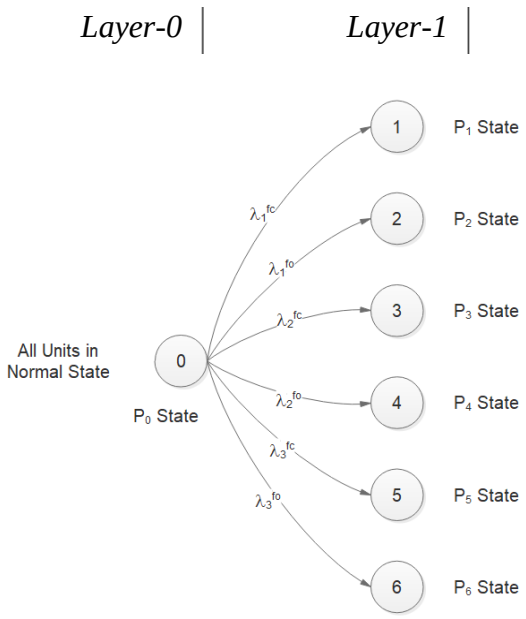


Figure 16: Transition diagram from Layer-1 to Layer-2

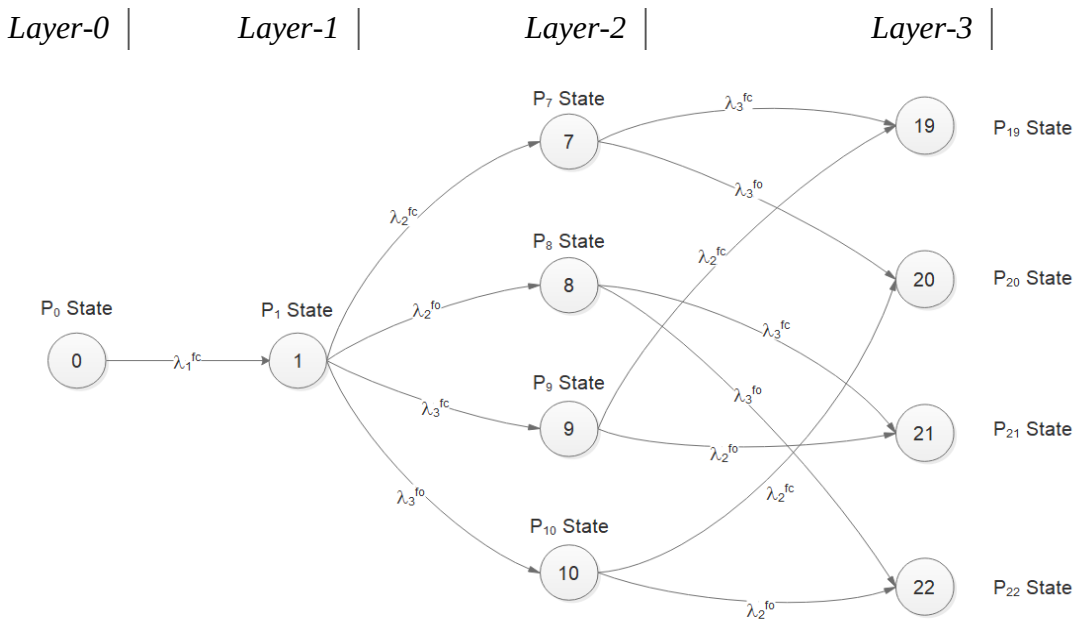


Figure 17: Transition path from Layer-0 to Unit-1 failed-closed to Layer-3

From the constructed Markov transition diagram, a set of ODEs can be obtained. [1]

$$\frac{dP_0(t)}{dt} = -(\lambda_1^{fc} + \lambda_1^{fo} + \lambda_2^{fc} + \lambda_2^{fo} + \lambda_3^{fc} + \lambda_3^{fo})P_0(t)$$

$$\frac{dP_1(t)}{dt} = \lambda_1^{fc}P_0(t) - (\lambda_2^{fc} + \lambda_2^{fo} + \lambda_3^{fc} + \lambda_3^{fo})P_1(t)$$

$$\frac{dP_2(t)}{dt} = \lambda_1^{fo}P_0(t) - (\lambda_2^{fc} + \lambda_2^{fo} + \lambda_3^{fc} + \lambda_3^{fo})P_2(t)$$

$$\frac{dP_3(t)}{dt} = \lambda_2^{fc}P_0(t) - (\lambda_1^{fc} + \lambda_1^{fo} + \lambda_3^{fc} + \lambda_3^{fo})P_3(t)$$

$$\frac{dP_4(t)}{dt} = \lambda_2^{fo}P_0(t) - (\lambda_1^{fc} + \lambda_1^{fo} + \lambda_3^{fc} + \lambda_3^{fo})P_4(t)$$

$$\frac{dP_5(t)}{dt} = \lambda_3^{fc}P_0(t) - (\lambda_1^{fc} + \lambda_1^{fo} + \lambda_2^{fc} + \lambda_2^{fo})P_5(t)$$

$$\frac{dP_6(t)}{dt} = \lambda_3^{fo}P_0(t) - (\lambda_1^{fc} + \lambda_1^{fo} + \lambda_2^{fc} + \lambda_2^{fo})P_6(t)$$

$$\frac{dP_7(t)}{dt} = \lambda_2^{fc}P_1(t) + \lambda_1^{fc}P_3(t) - (\lambda_3^{fc} + \lambda_3^{fo})P_7(t)$$

$$\frac{dP_8(t)}{dt} = \lambda_2^{fo}P_1(t) + \lambda_1^{fc}P_4(t) - (\lambda_3^{fc} + \lambda_3^{fo})P_8(t)$$

[1] Note that all the 27 ODEs are grouped and depicted by a single equation numbering scheme for easier representation

$$\frac{dP_9(t)}{dt} = \lambda_3^{fc} P_1(t) + \lambda_1^{fc} P_5(t) - (\lambda_2^{fc} + \lambda_2^{fo}) P_9(t)$$
$$\frac{dP_{10}(t)}{dt} = \lambda_3^{fo} P_1(t) + \lambda_1^{fc} P_5(t) - (\lambda_2^{fc} + \lambda_2^{fo}) P_{10}(t)$$
$$\frac{dP_{11}(t)}{dt} = \lambda_2^{fc} P_2(t) + \lambda_1^{fo} P_3(t) - (\lambda_3^{fc} + \lambda_3^{fo}) P_{11}(t)$$
$$\frac{dP_{12}(t)}{dt} = \lambda_2^{fo} P_2(t) + \lambda_1^{fo} P_4(t) - (\lambda_3^{fc} + \lambda_3^{fo}) P_{12}(t)$$
$$\frac{dP_{13}(t)}{dt} = \lambda_3^{fc} P_2(t) + \lambda_1^{fo} P_5(t) - (\lambda_2^{fc} + \lambda_2^{fo}) P_{13}(t)$$
$$\frac{dP_{14}(t)}{dt} = \lambda_3^{fo} P_2(t) + \lambda_1^{fo} P_6(t) - (\lambda_2^{fc} + \lambda_2^{fo}) P_{14}(t)$$
$$\frac{dP_{15}(t)}{dt} = \lambda_3^{fc} P_3(t) + \lambda_2^{fc} P_5(t) - (\lambda_1^{fc} + \lambda_1^{fo}) P_{15}(t)$$
$$\frac{dP_{16}(t)}{dt} = \lambda_3^{fo} P_3(t) + \lambda_2^{fc} P_6(t) - (\lambda_1^{fc} + \lambda_1^{fo}) P_{16}(t)$$
$$\frac{dP_{17}(t)}{dt} = \lambda_3^{fc} P_4(t) + \lambda_2^{fo} P_5(t) - (\lambda_1^{fc} + \lambda_1^{fo}) P_{17}(t)$$
$$\frac{dP_{18}(t)}{dt} = \lambda_3^{fo} P_4(t) + \lambda_2^{fo} P_6(t) - (\lambda_1^{fc} + \lambda_1^{fo}) P_{18}(t)$$
$$\frac{dP_{19}(t)}{dt} = \lambda_3^{fc} P_7(t) + \lambda_2^{fc} P_9(t) + \lambda_1^{fc} P_{15}(t)$$
$$\frac{dP_{20}(t)}{dt} = \lambda_3^{fo} P_7(t) + \lambda_2^{fc} P_{10}(t) + \lambda_1^{fc} P_{16}(t)$$
$$\frac{dP_{21}(t)}{dt} = \lambda_3^{fc} P_8(t) + \lambda_2^{fo} P_9(t) + \lambda_1^{fc} P_{17}(t)$$
$$\frac{dP_{22}(t)}{dt} = \lambda_3^{fo} P_8(t) + \lambda_2^{fo} P_{10}(t) + \lambda_1^{fc} P_{18}(t)$$
$$\frac{dP_{23}(t)}{dt} = \lambda_3^{fc} P_{11}(t) + \lambda_2^{fc} P_{13}(t) + \lambda_1^{fo} P_{15}(t)$$
$$\frac{dP_{24}(t)}{dt} = \lambda_3^{fo} P_{11}(t) + \lambda_2^{fc} P_{14}(t) + \lambda_1^{fo} P_{16}(t)$$
$$\frac{dP_{25}(t)}{dt} = \lambda_3^{fc} P_{12}(t) + \lambda_2^{fo} P_{13}(t) + \lambda_1^{fo} P_{17}(t)$$
$$\frac{dP_{26}(t)}{dt} = \lambda_3^{fo} P_{12}(t) + \lambda_2^{fo} P_{14}(t) + \lambda_1^{fo} P_{18}(t)$$

(4-3)

The above ODEs can be solved with given initial conditions. However, before going ahead to solving the above transition rate equations, a qualitative overview is provided for better comprehension of the problem at hand.

4.4.1. Qualitative Assessment

A preliminary qualitative analysis can be performed with the knowledge of the system states defined in the previous *Table 12*. It is important to note here that this qualitative assessment is only an approximation. It can be observed that failure probability of the BS to maintain nominal liquid level with all possible IEs is 29.63 %. Whereas, probability of the BS failing by overflow and drained is 87.5 % and 12.5 % respectively.

Taking a step further, if the order and sequence of unit failure is strictly taken into account, the probability of a unit failing first must be considered and treated separately. The probability that a unit will fail first is given by the ratio of failure rate for that unit divided by the sum of the failure rates for all the units. Thus, probability of Unit-1 failing first is:

$$P(\text{Unit} - 1 \rightarrow \text{ failing first}) = \frac{\lambda_1}{\lambda_1+\lambda_2+\lambda_3} \tag{4-4}$$

$$P(\text{Unit} - 1 \rightarrow \text{ failing first}) = 34.06\ \%$$

Similarly, the probability of Unit-2 and Unit-3 failing first is determined to be:

$$P(\text{Unit} - 2 \rightarrow \text{ failing first}) = 42.63\ \%$$

$$P(\text{Unit} - 3 \rightarrow \text{ failing first}) = 23.31\ \%$$

The strict systematic ordering and sequencing of transition, and its subsequent end state with all possible unit failure modes is shown in *Figure 18*. This approach has some differences with the conventional Markov chain, in the sense that the number of end states and absorbing states are different.

It can be observed that all system end state with failure modes ‘Unit-1 failed-open’ will eventually lead to a system overflow scenario or an absorbing state. Hence, with the failure mode ‘Unit-1 failed-open’, there is a 100 % probability that the system will make a transition to an absorbing state or more specifically, the system will fail by overflow condition. For Unit-1 failed-closed failure mode, the liquid level will decrease until it reaches Region -1. Then, Unit-2 will be turned on and Unit-3 will be turned off as per control laws defined with respect to the liquid level. This causes the fluid level to rise until it reaches Region +1, at which time Unit-2 will be turned off and Unit-3 will be tuned on. This causes the liquid level to uniformly oscillate between $Region - 1$ and $Region + 1$. Since the liquid flowrates from all the units are equal, the BS spends an equal

amount of time in the increasing and decreasing liquid level conditions. Thus, it can be concluded that, the BS is equally likely to fail in an overflow or drained state. Therefore, with the failure mode Unit-1 failed-closed, there is a 50 % chance that the system will fail by overflow and drained respectively. Combining all the possible system failures for both the failure modes from Unit-1 (failed-closed + failed-open), it can be calculated that the probability of BS failing by overflow and drained is 80 % and 20 % respectively. Similarly, the state sequences transition with failure modes Unit-2 failed-closed and failed-open are the same as Unit-1.

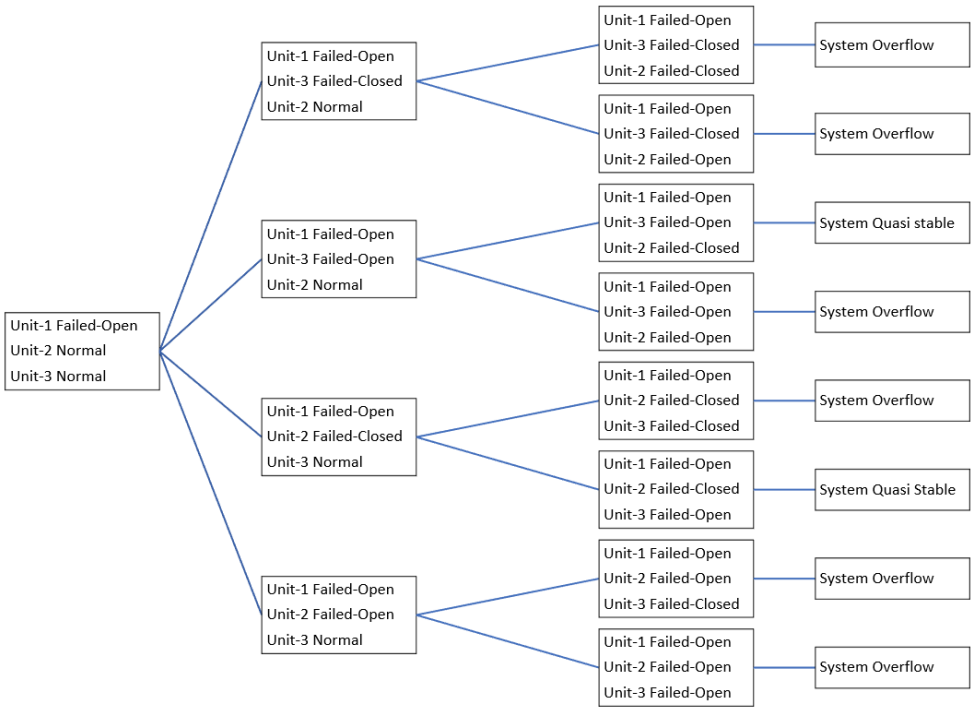


Figure 18: System state transition for Unit-1 failed-open

For Unit-3 (the only drain unit), all system failures caused by the failure mode Unit-3 failed-open, there is a 50 % probability of system failing by overflow and drained respectively. This is true since the flowrates from all the units are assumed to be equal. For the failure mode Unit-3 failed-closed, there is 100 % probability that the system will fail by overflow condition regardless of the relative flow rates provided by the three units. Unit-3 failing-closed will cause the liquid level to rise until it crosses the nominal region, at which time Unit-1 will be turned off. The tank will remain in this condition until either Unit-1 or Unit-2 fail-open, either of which will lead directly to the BS overflow. Thus, the probability of system failing by overflow and drained is 80 % and

20 % respectively. Utilizing the calculated probability values for individual units failing first, the probability that the BS failing by overflow is determined as:

$$(0.3406213 \times 0.8) + (0.4262653 \times 0.8) + (0.233113 \times 0.8) = 0.80$$

Thus, the BS will fail by overflow condition approximately 80 % of the time and fail by system drained approximately 20 % of the time. Note that, the system being stable or quasi-stable are not considered for the current analysis since we are interested only on the state ordering and qualitative aspect of the sequence transition and absorbing states. These aspects will be treated later when Markov model is coupled with CCMT to capture the system dynamics.

The strict states ordering results in 48 total numbers of failure sequences/end states in contrast to the 27 system states generated by applying the classical Markov chain. The failure probability of the system states generated by the two approaches are different, wherein the first approach overestimates the probability of system failing by overflow and underestimates the probability of system failing by drained. The differences in the two approaches are presented in *Table 14.*

Considering that all unit flowrates are equal for the BS, and that some unit failure type does not necessarily lead to a change in system configuration due to the control laws, these unit failure type can be omitted from IEs. For instance, if the system is in normal operating condition, Unit-3 failing open does not lead to a change in system configuration since Unit-1 fully open can compensate for these failures. The transition probabilities without a system configuration change is zero, in which case, the IE can be omitted. Similarly, the same argument holds for Unit-1 failing-open and Unit-2 failing-closed since these failures does not lead to a change in system configuration. Note that this consideration is true only if the system is under normal operating condition. Thus, considering only the three (3) IEs; namely Unit-1 failed-closed, Unit-2 failed-open and Unit-3 failed-closed, the qualitative assessment can be performed for the system evolution leading to 18 failure sequences. Taking the IE Unit-3 failed-closed, all system failure event sequence eventually leads to system overflow condition i.e., a 100 % failure probability. For Unit-1 failed-closed, it is similar to as described in the previous section, i.e., there is a 50 % chance that the system will fail by overflow and drained respectively. Similarly, for the IE Unit-2 failed-open, there is a 100 % probability that the system will fail by overflow condition. Utilizing the calculated probability for individual units failing first, the probability that the BS failing by overflow is:

$$(0.341 \times 50\%) + (0.426 \times 100\%) + (0.233 \times 100\%) = 0.83$$

Thus, the BS will fail by overflow condition approximately 83 % of the time and fail by system drained approximately 17 % of the time. Differences in the qualitative assessment are presented in *Table 14*

Table 14: A qualitative comparison of three different approach

Approaches	*Number of Failure Sequence*	*System failure states*	
		Overflow (%)	*Drained (%)*
Conventional Markov chain	27	87.50 %	12.50 %
Markov chain with explicit state ordering	48	80.00 %	20.00 %
Markov chain with explicit state ordering for 3 IEs	18	83.00 %	17.00 %

The third approach provide a good approximation in the sense that the system failures by overflow and drained is not overestimated or under estimated. This is due to the fact that a specific unit failure is not necessarily an IE leading to a system failure. Hence, these failures are not included as IEs for analysis purpose. Of course, like all Markov chain state representation, this approximation is only true with the assumption that two failures do not occur simultaneously in a single time step.

4.4.2. Quantitative Assessment

The ordinary differential equations (ODEs) obtained from the Markov state transition diagram can be solved using two approaches: (1) Analytical methods; (2) Numerical methods.

1. Analytical methods

From the assumption that the BS is in normal operating condition at time $t = 0$, or that the system is in state $P_0(t = 0)$, the initial conditions for all the ODEs can be defined as:

$$P_0(0) = 1;\ P_1(0) = 0;\ P_2(0) = 0;\ P_3(0) = 0;\ P_4(0) = 0; \ldots \ldots P_{26}(0) = 0$$

The solution of the transition probabilities with given initial conditions are:

$$P_0(t) = \exp\{-(\lambda_1^{fc} + \lambda_1^{fo} + \lambda_2^{fc} + \lambda_2^{fo} + \lambda_3^{fc} + \lambda_3^{fo})t\} \quad (4\text{-}5)$$

$$P_7(t) = \frac{\lambda_1^{fc}\lambda_2^{fc}}{\lambda_1^{fc}\lambda_2^{fc} + \lambda_1^{fc}\lambda_2^{fo} + \lambda_1^{fo}\lambda_2^{fc} + \lambda_1^{fo}\lambda_2^{fo}} e^{-\left(\lambda_1^{fc}+\lambda_1^{fo}+\lambda_2^{fc}+\lambda_2^{fo}+\lambda_3^{fc}+\lambda_3^{fo}\right)t}$$
$$- \frac{\lambda_1^{fc}\lambda_2^{fc}}{\lambda_1^{fc}\lambda_2^{fc} + \lambda_1^{fc}\lambda_2^{fo} + \lambda_1^{fo}\lambda_2^{fc} + \lambda_1^{fo}\lambda_2^{fo}} e^{-\left(\lambda_1^{fc}+\lambda_1^{fo}+\lambda_3^{fc}+\lambda_3^{fo}\right)t}$$
$$- \frac{\lambda_1^{fc}\lambda_2^{fc}}{\lambda_1^{fc}\lambda_2^{fc} + \lambda_1^{fc}\lambda_2^{fo} + \lambda_1^{fo}\lambda_2^{fc} + \lambda_1^{fo}\lambda_2^{fo}} e^{-\left(\lambda_2^{fc}+\lambda_2^{fo}+\lambda_3^{fc}+\lambda_3^{fo}\right)t}$$
$$+ \frac{\lambda_1^{fc}\lambda_2^{fc}}{\lambda_1^{fc}\lambda_2^{fc} + \lambda_1^{fc}\lambda_2^{fo} + \lambda_1^{fo}\lambda_2^{fc} + \lambda_1^{fo}\lambda_2^{fo}} e^{-\left(\lambda_3^{fc}+\lambda_3^{fo}\right)t} \quad (4\text{-}6)$$

And so on.

Solving the coupled ODEs via analytical methods become extremely complex and time consuming. Furthermore, with an increase in the number of component states, it becomes impractical to solve ODEs by analytical methods. Thus, numerical methods provide a better approach in solving the coupled ODEs.

2. Numerical methods

Numerical method was used in this thesis for obtaining the solution of the coupled ODEs since it provides a more robust and practical approach. Finite difference method (FDM) has been used for solving the coupled ODEs. The transition probability ODEs can be written in state matrix form as:

$$\frac{dP(t)}{dt} = A.P(t) \quad (4\text{-}7)$$

Where, the matrix "A" is the transition probability matrix, and can be validated by using:

$$\sum_{j=1}^{n}\sum_{i=1}^{n} A_{ij} = 1 \quad (4\text{-}8)$$

Left hand side of *Equation (35)* can be re-written in the following way;

$$\frac{dP(t)}{dt} = \frac{P(t+\Delta t) - P(t)}{(t+\Delta t) - t} \quad (4\text{-}9)$$

Substituting *Equation (4-5)* in *Equation (4-3)* yields:

$$\frac{P(t+\Delta t)-P(t)}{\Delta t} = A.P(t)$$

$$P(t+\Delta t) = (I + A.\Delta t).P(t) \quad (4\text{-}10)$$

The above equation is equivalent to the Chapman–Kolmogorov equation in discrete time state-space. With the assumption that the BS is in normal operating condition at time $t = 0$, the state transition probabilities are computed by implementing FDM in FORTRAN95 code. Considering a time step of $\Delta t = 1\ hour$, we obtain the transition probabilities for 3 time steps *(See Table 15).*

At any given time step, system failure probability can be determined by summing up the states leading to either system overflow or drained scenario. For instance, from *Table 15*, it can be observed that states $12, 13, 17, 21, 23, 25$ and 26 result in a system overflow scenario. For illustration purpose, the probability of system overflow at time step $k = 4$ is determined as:

System overflow $(k = 4) = P_{12}(t) + P_{13}(t) + P_{17}(t) + P_{21}(t) + P_{23}(t) + P_{25}(t) + P_{26}(t)$

$$\text{System overflow } (k = 4) = 8.68 \times 10^{-05}$$

Similarly, the probability of system drained at $k = 4$ can be obtained as:

$$\text{System drained } (k = 4) = P_{20}(t)$$

$$\text{System drained } (k = 4) = 6.12 \times 10^{-08}$$

Similarly, the probability of system failure (overflow or drained) for any given time step can be obtained using the same procedure as depicted above.

Table 15: Transition probabilities in terms of $P_n(t)$

#	$P_0(t)$	$P_1(t)$	$P_2(t)$	$P_3(t)$	$P_4(t)$	$P_5(t)$	$P_6(t)$	$P_7(t)$	$P_8(t)$
$k=1$	1	0	0	0	0	0	0	0	0
$k=2$	0.987	2.28E-3	2.28E-3	2.86E-3	2.857E-3	1.563E-3	1.563E-3	0	0
$k=3$	0.974	4.52E-3	4.52E-3	5.65E-03	5.654E-03	3.088E-03	3.088E-03	1.305E-5	1.305E-5
$k=4$	0.96	6.698E-3	6.698E-3	8.39E-3	8.392E-3	4.577E-03	4.577E-03	3.88E-5	3.882E-5

#	$P_9(t)$	$P_{10}(t)$	$P_{11}(t)$	$P_{12}(t)$	$P_{13}(t)$	$P_{14}(t)$	$P_{15}(t)$	$P_{16}(t)$	$P_{17}(t)$
$k=1$	0	0	0	0	0	0	0	0	0
$k=2$	0	0	0	0	0	0	0	0	0
$k=3$	7.12E-6	7.135E-6	1.305E-5	1.30E-5	7.135E-6	7.13E-6	8.929E-6	8.93E-6	8.929E-6
$k=4$	2.11E-05	2.11E-05	3.882E-5	3.88E-5	2.12E-05	2.12E-05	2.66E-05	2.66E-05	2.66E-05

#	$P_{18}(t)$	$P_{19}(t)$	$P_{20}(t)$	$P_{21}(t)$	$P_{22}(t)$	$P_{23}(t)$	$P_{24}(t)$	$P_{25}(t)$	$P_{26}(t)$
$k=1$	0	0	0	0	0	0	0	0	0
$k=2$	0	0	0	0	0	0	0	0	0
$k=3$	8.93E-6	0	0	0	0	0	0	0	0
$k=4$	2.65E-05	6.12E-08	6.12E-08	6.12E-08	6.12E-08	6.12E-08	6.12E-08	6.12E-08	6.12E-08

4.5. DFM Model of the Benchmark System

A DFM model of the BS represents the temporal behavior and dynamic evolution of the system. To make an accurate representation of the system, vital components that dictates the system behavior is identified and included in the system model. For instance, considering the case of the BS, all the three (3) control units must be included in the model. Furthermore, each of the control units are considered as a single component rather than a system. Once the vital parameters are identified, causal, temporal and conditional relationships are established among the parameters. These relationships are manifested by algebraic or differential equations, which can be solved to construct the decision tables. For analysis purpose and model development, several assumptions are made, including:

1. All assumptions made in the previous section applies;
2. Level sensor failure characteristics are omitted;
3. Unit fail-closed and fail-open are considered as sink states.

A stepwise representation of the methodology application and model development of the benchmark system in the computer code DYMONDA is provided.

1. Identification of system hardware components

All hardware components are included in the DFM model and are represented by process variables nodes in the model. The BS hardware components are presented in *Table 16*.

Table 16: System hardware identification

No.	*Description*	*Representation*
1.	Unit-1	U1
2.	Unit-2 (Standby unit)	U2-SB
3.	Unit-3	U3
4.	Level Sensor	ML

2. Identification of system parameters

Parameters that captures the attributes of the hardware components are identified and modelled as process variables nodes in the DFM model. The BS parameters are presented in *Table 17.*

Table 17: System parameter identification

No.	*Description*	*Representation*
1.	Tank Level	TL
2.	Total Net Fluid Flow	TNF
3.	Total Liquid Inflow	TIF
4.	Total Liquid Outflow	TOF
5.	Liquid Flow from Unit-1	U1-F
6.	Liquid Flow from Unit-2	U2-F

3. Identification of Condition Nodes

A condition node, like the process variable nodes represents physical parameters. However, conditioning nodes more explicitly identify the switching action or failure modes/states of a component. Condition nodes of the BS are presented in *Table 18.*

Table 18: System conditioning node identification

No.	*Description*	*Representation*
1.	Unit-1 State or failure modes	U1-S
2.	Unit-2 State or failure modes	U2-S
3.	Unit-3 State or failure modes	U3-S

4. Relationship Establishment (causal, temporal and conditional)

The process variable nodes are linked together by causality edges through transfer boxes and transition boxes to model the cause-effect and temporal relationships among the parameters. The conditioning nodes are linked to the transfer or transition box by the condition edges. For example, consider *Figure 19*: the total liquid flow from Unit-1 (U1-F) is dependent on the command signal to Unit-1 (U1) and the failure modes of Unit-1 (U1-S) which are stochastic in nature. U1 and U1-F have a causal relationship which are represented and linked by causality edges, whereas U1-S and U1-F have a conditional relationship which is represented and linked by a condition edge (as can be seen from figure below).

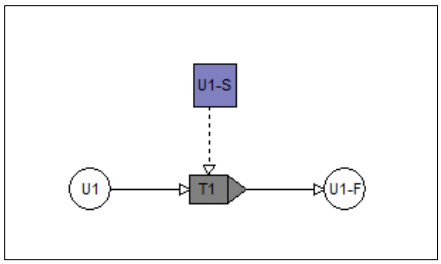


Figure 19: DFM model of Total Liquid Flow from Unit-1

The process of node linking via causality and condition edges is carried out for among all the parameters having a causal, temporal and conditional relationships. Eventually, this result in an integrated causality and time transition network as shown in *Figure 20*.

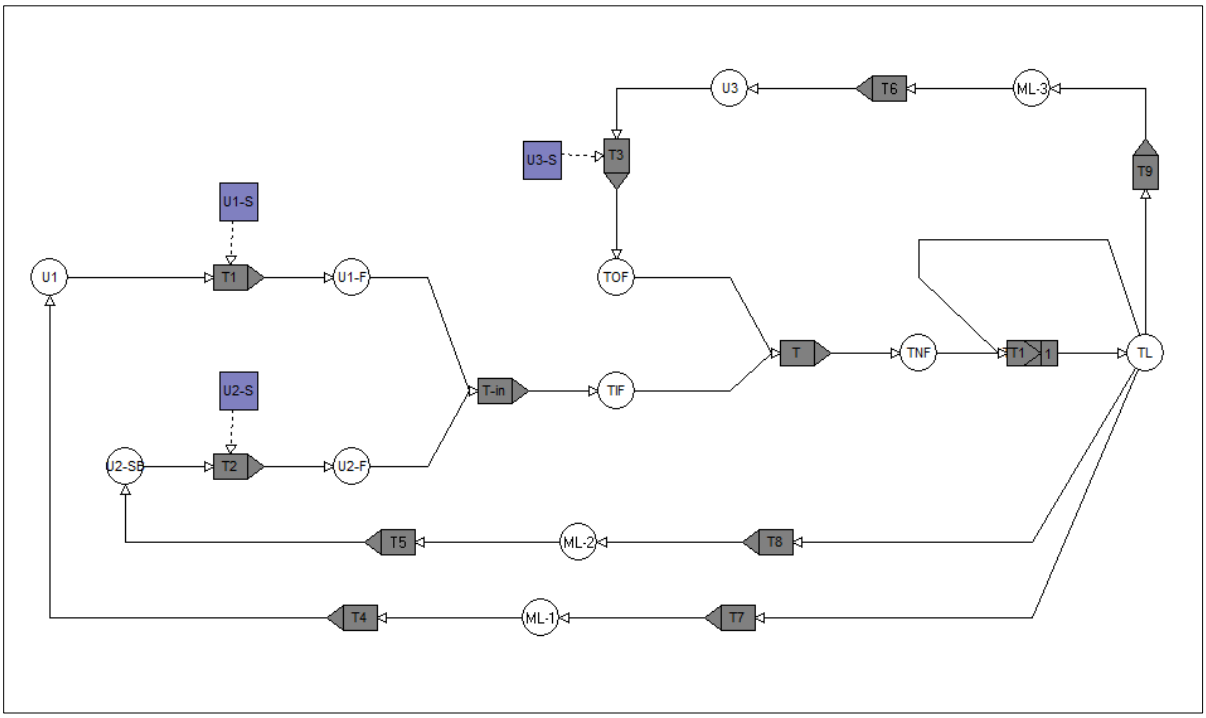


Figure 20: DFM Model of the Benchmark System

5. Discretization of system parameters

The identified process variable nodes and the condition nodes are then discretized into finite number of states for the purpose of analysis. The number of discretized states or cells depends on the analyst choice, level of in-depth modelling and the desired outcome. A sample discretization of liquid level is into five (5) mutually exclusive intervals is depicted in *Table 19*.

Table 19: Discretization of TL into 5 intervals

State	*Intervals/Cells*	*Description*
-2	Low-low (Sink cell)	$(x < -3)\ meters$
-1	Low $(Region: -1)$	$(-3\ to\ -1)\ meters$
0	Nominal level $(Region: 0)$	$(-1\ to\ +1)\ meters$
+1	High $(Region: +1)$	$(+1\ to\ +3)\ meters$
+2	High-high (Sink cell)	$(x > +3)\ meters$

6. Development of decision table

The decision tables within the transfer and transition boxes are constructed systematically via inductive approach, i.e., enumerating all possible combinations of input states and then finding the output state for each combinations. This ensures a complete and comprehensive table. Since enumerating all possible input combinations can be time taking, decision table reduction method is utilized to produce a compact table, suitable for use in the DYMONDA code. Sample decision table is depicted in *Table 20*. *Note:* "*" represents a "Don't care" condition.

Table 20: Decision table for TT1 with 5 discretized liquid level intervals

TNF @t=-1	*TL @t=-1*	*TL @t=0*
*	-2	-2 (sink state)
*	+2	+2 (sink state)
-1	-1	-2
-1	0	-1
-1	+1	0
0	-1	-1
0	0	0
0	+1	+1
+1	-1	0
+1	0	+1
+1	+1	+2
+2	-1	0
+2	0	+1
+2	+1	+2

The decision table TT1 captures the time element in the model. Notice that there are two TL but with different time element (@t= -1 and @t= 0). This implies that the current liquid level @t= 0 is dependent on the net liquid flow and the liquid level in the previous time step @t= -1. This temporal transition relationship (system dynamics) is capture by the decision *Table 20*.

7. Model Analysis

Two scenarios were analyzed i.e., overflow and drained with two different discretization scheme: (1) With 5 discretized state variable intervals; (2) With 8 discretized state variable intervals. The top events in the model are defined in terms of state variable magnitudes, e.g., TL=+2 @t=0 and TL=-2 @t=0. For instance, TL=+2 @t=0 is interpreted as the liquid level being above 3 meters at time, t= 0. The simulation time step is chosen in such a way that transition occurs from an interval to the adjacent interval in a single time step. This avoid multiple intervals transition and provide smooth and finer results. Taking into account the number of discretized tank level, a single simulation time step was approximated to be 1 hour. Once the top event and simulation time step is defined, backtracking is performed through the model to generate timed PIs. For illustration purpose, few generated results are presented below.

4.5.1. DFM Model results

A. For TL=+2 @t=0 or 'overflow' with single time step (5 intervals)

The first step is to define a top event in terms of the process variable node. The defined top event is then expressed as a transition table. The depth of backtracking process is selected (analyst choice). For the current case, a single time step is selected for demonstration purpose. A reduced transition table is shown in *Table 21*.

Table 21: Transition table after eliminating U2-SB @t=-1

U3-S @t= -1	*U1-S @t= -1*	*U2-S @t= -1*	*TL @t= -1*	*TOP TL=+2 @t=0*
*	*	*	+2	True
*	+1	+1	+1	True
-1	*	+1	+1	True
-1	+1	*	+1	True

The above transition table cannot be further reduced since the analysis is carried out only for a single time step. Thus, the backtracking process and transition table expansion is terminated and completed for the current analysis time step, t=-1. The transition table in *Table 21* is the critical transition table. A further absorption and reduction merging as well as consensus operation is performed on the critical transition table to obtain the complete base, but they do not produce any change in the critical transition table. Thus, *Table 21* is the complete base for the top event. The rows of *Table 21* are the 4 PIs obtained for the Top Event "TL = +2 @ t = 0". Quantification of the above PIs can be carried out with ease by inputting failure rates of each units in the respective process variable or conditioning nodes. The sum of the probabilities of mutually exclusive PIs yields the exact probability of the top event. For illustration purpose, quantification of the top event for system overflow with single time step is shown *Table 22*.

Table 22: Quantification of prime implicants for system overflow

#	*Prime Implicants*	*Time*	*Logic*	*Probability*
1.	Tank Liquid Level was at +3 meters	@t= -1		-
2.	Tank Liquid Level was between +1 and +3 meters	@t= -1	AND	6.52×10^{-6}
	Unit-1 fail-open	@t= -1	AND	
	Unit-3 fail-closed	@t= -1		
3.	Tank Liquid Level was between +1 and +3 meters	@t= -1	AND	4.46×10^{-6}
	Unit-2 fail-open	@t= -1	AND	
	Unit-3 fail-closed	@t= -1		
4.	Tank Liquid Level was between +1 and +3 meters	@t= -1	AND	3.57×10^{-6}
	Unit-1 fail-open	@t= -1	AND	
	Unit-2 fail-open	@t= -1		
Top Event Probability				1.455×10^{-5}

B. For TL=+2 @t=0 with two time step

System analysis can be performed for the top event "TL=+2 @t=0" or "system overflow" as done in the previous section, however with 2 simulation time steps or in other words for 2 hours in this case. This implies that backtracking process will be carried out for 2 cycles. Complete base of the top event "TL=+2 @t=0" with 2 simulation time steps is shown in *Table 23.*

Table 23: Complete base for the top event "TL=+2 @t=0" with 2 time steps

U1-S @t= -1	U1-S @t=-2	U2-S @t= -1	U2-S @t= -2	U3-S @t= -1	U3-S @t= -2	TL @t= -1	TOP
*	*	*	*	*	*	+2	True
*	*	+1	0	-1	-1	+1	True
*	*	+1	+1	-1	-1	0	True
*	*	+1	+1	-1	-1	+1	True
*	*	+1	+1	-1	0	+1	True
*	0	+1	+1	-1	0	0	True
*	0	+1	0	-1	-1	0	True
+1	0	*	*	-1	-1	0	True
+1	0	*	*	-1	-1	+1	True
+1	+1	*	*	-1	-1	0	True
+1	+1	*	*	-1	-1	+1	True
+1	+1	*	*	-1	0	+1	True
+1	+1	+1	+1	*	*	0	True
+1	0	+1	+1	*	*	+1	True
+1	+1	+1	0	*	*	+1	True
+1	+1	+1	+1	*	*	+1	True
+1	0	+1	+1	*	*	0	True

The rows of *Table 23* are the 17 prime implicants obtained for the top event "TL = +2 @ t = 0" with 2 simulation time steps. The quantification of the PIs results in:

$$TL = +2\ (@t = 0) = 7.698E - 05$$

C. For 'TL=+3 @t=0' with 1 backtracking depth and 8 state variable intervals

The model analysis is performed with increased number of state variable intervals to check the sensitivity of the PIs with respect to the number of states variable intervals. *Table 24* presents the PIs for system overflow with 8 state variable intervals for 1 backtracking depth.

Table 24: PIs for system overflow "TL=+3 @t= 0"

#	*Prime Implicants*	*Time*	*Logic*
1.	Tank Liquid Level was between +2 and +3 meters	@t= -1	AND
	Unit-1 Fail-High	@t= -1	AND
	Unit-2 Fail-High	@t= -1	
2.	Tank Liquid Level was between +2 and +3 meters	@t= -1	AND
	Unit-1 Fail-High	@t= -1	AND
	Unit-3 Fail-Low	@t= -1	
3.	Tank Liquid Level was between +2 and +3 meters	@t= -1	AND
	Unit-2 Fail-High	@t= -1	AND
	Unit-3 Fail-Low	@t= -1	
4.	Tank Liquid Level was between +1 and +2 meters	@t= -1	AND
	Unit-1 Fail-High	@t= -1	AND
	Unit-2 Fail-High	@t= -1	AND
	Unit-3 Fail-Low	@t= -1	
System overflow [$(TL = +3\ (@t = 0)$]		$1.456E-05$	

From *Table 24*, it may be observed that the PIs generated for the top event 'system overflow' with state variable intervals of 5 (Case-I) differs from the one with 8 state variable intervals (Case-II). Specifically, the prime implicant 4 (PI-4) is an additional PI. PI-4 does not appear in Case-I due to the fact that it is eliminated during the simplification process by Boolean algebra or absorption rule. Recall that PI-4 appears as a cut-set in the fault tree analysis, but not as a minimal cut-set. Similarly, PI-4 appears as an implicant in Case-I, but not as a PI. This can be attributed to less number of mutually exclusive state variable intervals for Case-I. Notice that in PI-4, the tank liquid level was between +1 and +2 meters @t= -1". This additional discretized interval of state variable causes the PI-4 to be a timed-prime implicants in the complete base, even though a combination

of “Unit-1 fail-high AND Unit-2 fail-high AND Unit-3 fail-low @t=-1” is not a PI. It is important to note that, due to the dynamic evolution of the state variable with time and the dependencies of the state variable on the unit state combination, a system overflow scenario is observed in a single time step with PI-4. This is true since the volume of the discretized intervals covering the entire state space for Case-I is larger than for Case-II. Hence, many transition from an interval will tend to remain in the same interval given that the simulation time is small, and the volume of the discretized intervals are large. However, for Case-II, the state variable can make a transition to another discretized intervals due to the smaller interval volume.

Similarly, simulation for 2 backtracking depth generates 28 PIs for Case-II, whereas for Case-I only 17 PIs are generated. The increased number of state intervals provide the analyst a more detail information and also increases the accuracy of the result obtained. However, computation time and complexity increase significantly with increased number of mutually exclusive intervals.

D. Variation of PIs for 5 and 8 state variable intervals

It may be observed that the number of PIs increases as the backtracking depth is increased. For instance, the number of PIs increased from 4 to 28 as the depth increased from 1 to 2. This implies that, with increasing depth there will be an increased number of possible ways for the system to fail due to the availability of time for the units to make transitions from normal to failure state. This analysis is performed with 5 and 8 state variable intervals to check the sensitivity of the number of PIs with respect to variation in number of state variable intervals. The sensitivity plot is depicted in *Figure 21* for 10 backtracking depth. Of course, the state variable can be discretized into a larger number of intervals to obtain a finer result. However, system modelling and development of decision table becomes complex with increased number of discretized intervals.

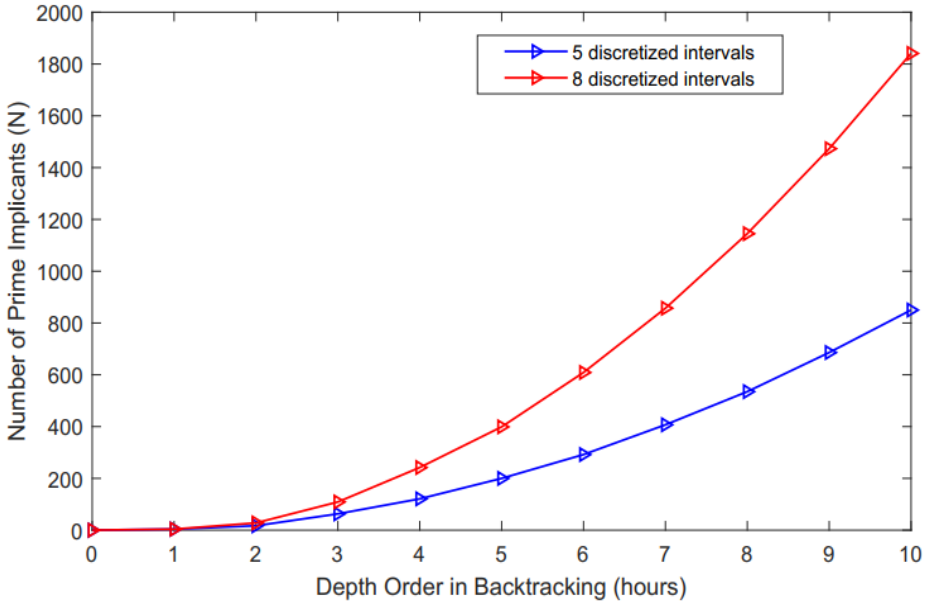


Figure 21: Sensitivity of number of PIs to discretized state variable intervals

4.5.2. Timed-Fault Tree Generation from DFM

For illustration and discussion purpose, the generation of timed-fault trees for system overflow scenario with 5 discretized state variable intervals is presented below. Similar to the classical FT technique, the first step is to define a desired top event in terms of state variable magnitude i.e., "TL=+2 @t=0" which is associated with the transition table "TT1". The decision table of TT1 is utilized to determine the inputs that causes the tank liquid level to be in TL=+2 @t=0. Since the TT1 associated with time delays, it changes the time at which a particular variable state occurs. The first transition from time $t = 0$ to $t = -1$ can be observed in the timed-fault tree given below *(See Figure 22)*. Throughout the backtracking process, dynamic and physical consistency rules are applied to process variables and conditioning nodes similar to that applied while developing the DFM model.

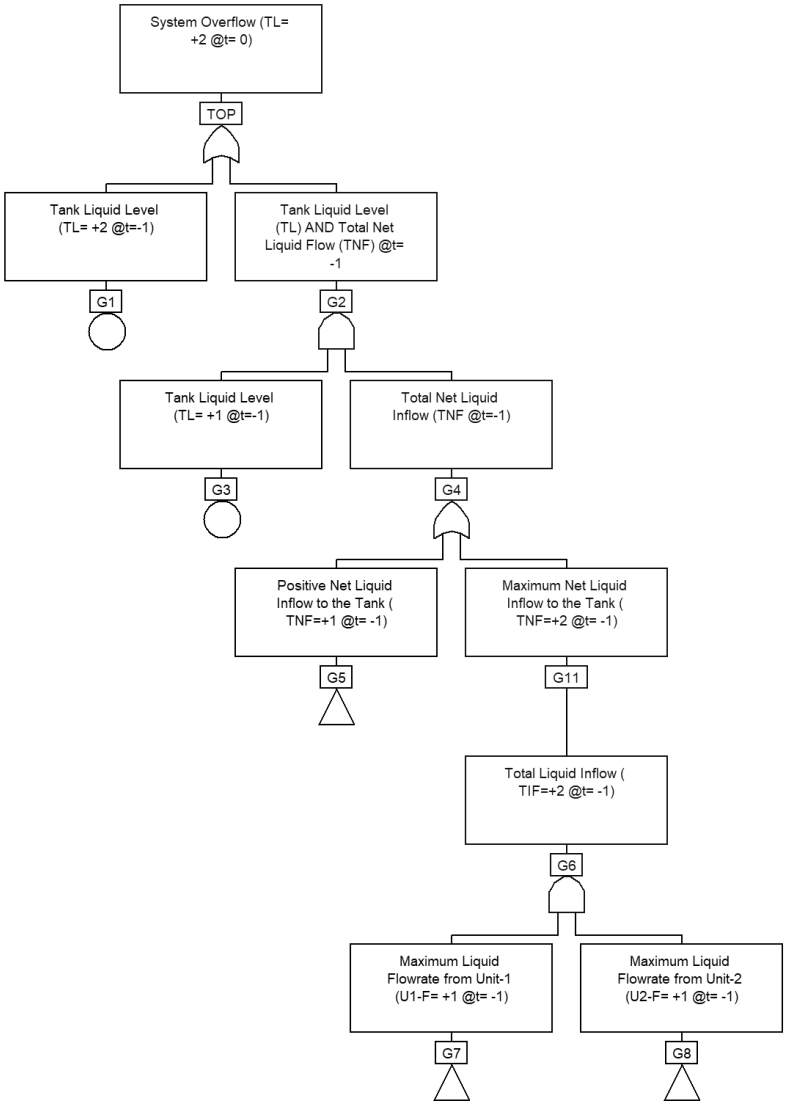


Figure 22: Timed-fault tree top event and its transition from @t=0 to @t= -1

It can be observed in *Figure A-III-1* (*Appendix III*) that the backtracking process from the top event "TL= +2 @t=0" results in the tank level TL= -1 @t= -1 or TL= -2 @t= -1 (G28 and G33). However, the tank level TL= +1 @t= -1 have already occurred under the same parent AND gate (G3 in *Figure 22).* This condition is impossible due to physical consistency i.e., tank liquid level cannot assume two different values/state in the same time step. Hence, G28 and G33 is eliminated from the tree. This elimination further leads to pruning of the parent AND gate G25.

In *Figure A-III-2 (*transfer-in gate G7), the backtracking process for single time step $@t = -1$ leads to tank level being at $TL = 0 \cup -1 \cup -2\ @t = -1$ (G37). However, under the same parent

AND gate (G2), $TL = +2\ @t = -1$ has already occurred. By physical consistency check, this condition is impossible to occur since liquid level cannot be in two or more different state in the same time step. This results in an elimination of gate G37. The removal of G37 further lead to a pruning of the AND gate G35, since initial pruning of G37 made the succeeding event occurrence to be impossible. The same argument holds for G46 and G48 in the transfer gate G8 (*See Figure A-III-3*).

Application of physical and dynamic consistency rule reduces the overall timed-fault tree to *Figure A-III-6:* . The FT in *Figure A-III-4* is further pruned by application of physical consistency rule to generate the final timed-FT shown *in Figure 23.* Since the Unit-1 fail-high have already occurred in G7, Unit-1 fail-closed (G23) cannot occur due to physical consistency, i.e., the Unit-1 cannot be in two different state in the same time step. Similarly, this argument holds for Unit-2 state. Unit-2 fail-high has already occurred in G8, hence Unit-2 fail-closed (G53) cannot occur due to physical consistency, i.e., Unit-2 cannot be in two different state in the same time step. These pruning are due to the fact that the gates G7 and G8 first occurs under the AND gate G6 rather than the AND gate G14. The pruning would have been reversed if gate G14 appears first in the timed-FT then gate G6, i.e., G7 and G8 would have been pruned, whereas G23 and G53 would have appeared in the MCS. The MCSs for the final reduced timed-fault tree are given in *Table 25***.**

Table 25: Minimal cut-set for the final timed-fault tree

1.	TL= +2 @t= –1	OR
2.	(TL= +1 AND U1-S= +1 AND U2-S= +1) @t= –1	OR
3.	(TL= +1 AND U1-S= +1 AND U3-S= –1) @t= –1	OR
4.	(TL= +1 AND U2-S= +1 AND U3-S= –1) @t= –1	

It can be observed that the MCS for the timed-fault tree is exactly the same as the PIs generated using the DFM model. In other words, this process of backtracking and generation of time-fault trees from the DFM model points out the fact that results obtained from dynamic methods can be incorporated into an existing static PRA model.

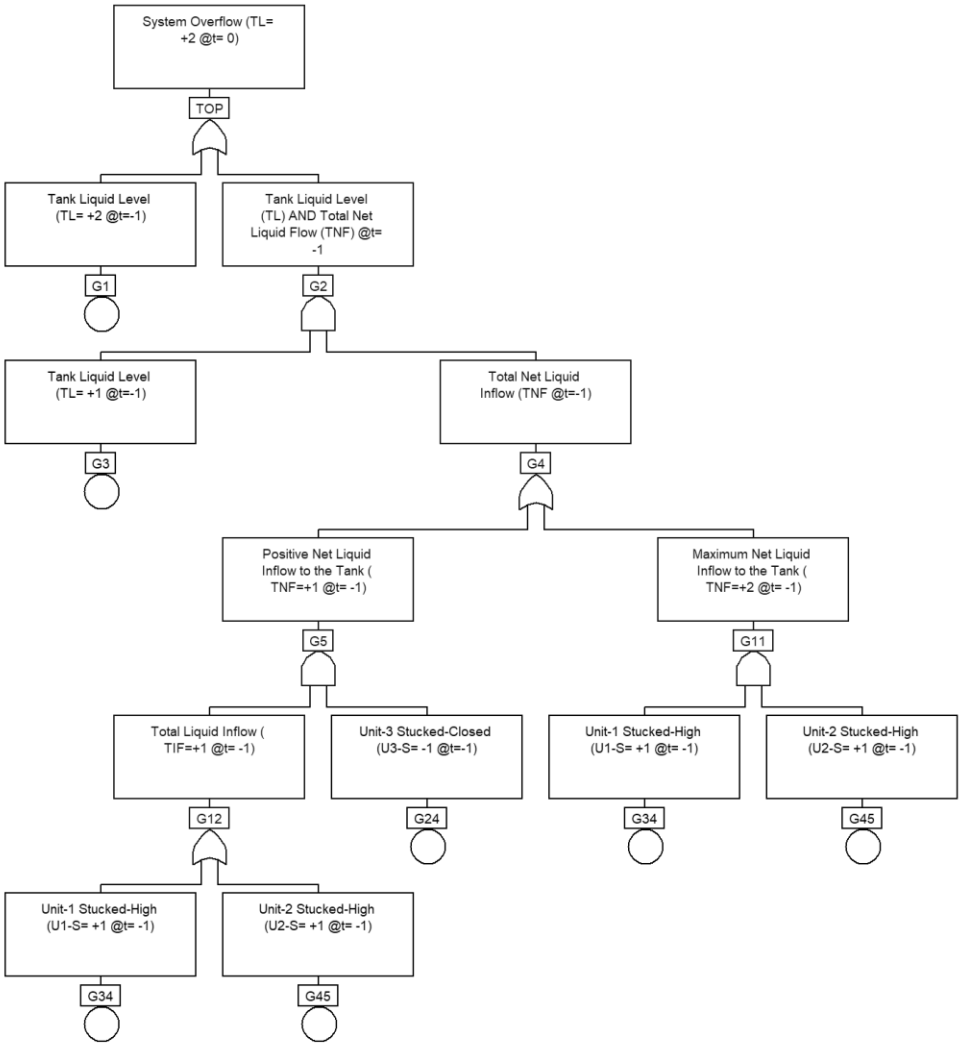


Figure 23: Final timed-fault tree after consistency check

4.6. Markov-CCMT Model of the Benchmark System

The classical Markov model of the benchmark system presented provide a time-dependent behavior of the system via a transition matrix, however lacks capability to capture system dynamics. This section couples the classical Markov model with CCMT.

a. Computation of $h(n/n', k\Delta t)$

The state transition matrix can be developed considering assumption number 4 in *Sub-section 3.2.3*, in that the units have a statistically independent failure. The unit stochastic behavior is represented through $h(n/n', k\Delta t)$, which is the probability that the unit state combination at time $t = (k+1)\Delta t$ is at n, given that the unit state at time $t = k\Delta t$ is at n'. It can be expressed as:

$$h(n/n', (k+1)\Delta t) = h\{n((k+1)\Delta t) = n/n(k\Delta t) = n'\} \tag{4-11}$$

It can be observed that $h(n/n', k\Delta t)$ is a conditional probability of the component state combination being at n at time $t = (k+1)\Delta t$, given that the component state combination was at n' at time $t = k\Delta t$. $h(n/n', k\Delta t)$ can be expressed as in *Equation (3-50).*

For instance, $n_1 = 0, n_2 = 0\ and\ \ n_3 = 0$ at time $k\Delta t$ implies that all the three units are normal, and hence $n' = 0$. If $n_1 = 0, n_2 = 0\ and\ \ n_3 = 0$ at time $(k+1)\Delta t$ i.e., Unit-1, Unit-2 and Unit-3 are normal, which implies $n = 0$ (as per *Table 12*). Thus $h(n/n', (k+1)\Delta t)$ can be written as;

$$h(0/0, (k+1)\Delta t) = (n_1 = 0/{n'}_1 = 0) * (n_2 = 0/{n'}_2 = 0) * (n_3 = 0/{n'}_3 = 0) \tag{4-12}$$

i.e., a transition of the system from normal state to normal state at time step $k\Delta t$, $[(k+1)\Delta t]$.

$h(0/0, (k+1)\Delta t)$ can be quantified in terms of the unit failure probabilities:

$$h(0/0, (k+1)\Delta t) = (1 - \lambda_1\Delta t) * (1 - \lambda_2\Delta t) * (1 - \lambda_3\Delta t) \tag{4-13}$$

$$h(0/0, (k+1)\Delta t) = 0.987$$

The transition probabilities among the other state combinations can be computed in the same fashion, and the general form is given in *Table 26.*

Table 26: System state transition probabilities

$h(n/n', (k+1)\Delta t) = \prod_{m=1}^{M} c_m(n_m/n'_m, k\Delta t)$		
Unit-1 $(n_1/n'_1, k\Delta t)$	**Unit-2** $(n_2/n'_2, k\Delta t)$	**Unit-3** $(n_3/n'_3, k\Delta t)$
$P(n_1 = 0/n'_1 = 0)$ $= \left(1 - (\lambda_1^{fc} + \lambda_1^{fo})\Delta t\right)$	$P(n_2 = 0/n'_2 = 0)$ $= \left(1 - \left(\lambda_2^{fc} + \lambda_2^{fo}\right)\Delta t\right)$	$P(n_3 = 0/n'_3 = 0)$ $= \left(1 - \left(\lambda_3^{fc} + \lambda_3^{fo}\right)\Delta t\right)$
$P(n_1 = 1/n'_1 = 0) = \lambda_1^{fc}\Delta t$	$P(n_2 = 1/n'_2 = 0) = \lambda_2^{fc}\Delta t$	$P(n_3 = 1/n'_3 = 0) = \lambda_3^{fc}\Delta t$
$P(n_1 = 2/n'_1 = 0) = \lambda_1^{fo}\Delta t$	$P(n_2 = 2/n'_2 = 0) = \lambda_2^{fo}\Delta t$	$P(n_3 = 2/n'_3 = 0) = \lambda_3^{fo}\Delta t$
$P(n_1/n'_1) = 0$, otherwise	$P(n_2/n'_2) = 0$, otherwise	$P(n_3/n'_3) = 0$, otherwise

Since there are 27 unit state combinations, $h(n/n', (k+1)\Delta t)$ will have a 27×27 dimension matrix with the structure like *Table 27*. The algorithm for generating the overall matrix is developed and implemented using Fortran95. Note that, $h(n/n', (k+1)\Delta t)$ is expressed as a product of individual unit failure probabilities due to the initial assumption made for the BS that unit failures are statistically independent.

Table 27: Possible state transition among distinct unit state combinations

$\boldsymbol{n'}$	$\boldsymbol{n}$						
	1	**2**	**3**	**……..**	$\boldsymbol{i}$	**……..**	$\boldsymbol{n}$
1	$\mu_{11}\Delta t$	$\lambda_{12}\Delta t$	$\lambda_{13}\Delta t$	……..	$\lambda_{1i}\Delta t$	……..	$\lambda_{1n}\Delta t$
2	0	$\mu_{22}\Delta t$	$\lambda_{23}\Delta t$	……..	$\lambda_{2i}\Delta t$	……..	$\lambda_{2n}\Delta t$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
$\boldsymbol{j}$	0	0	0	……..	$\mu_{ji}\Delta t$	……..	$\lambda_{jn}\Delta t$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
$\boldsymbol{n'}$	0	0	0	0	0	……..	$\mu_{n'n}\Delta t$

For the BS under consideration, the conditional transition probability matrix $h(n/n', k\Delta t)$ represent the BS stochastic time-dependent properties. A further analysis of the transition

probability matrix can be performed by decomposing the matrix into four (4) sub-matrices and re-writing the transition probability matrix 'P' in the canonical form as shown in *Table 28*.

Table 28: Canonical form of the transition probability matric

P	$\boldsymbol{S'}$	$\boldsymbol{T'}$
$\boldsymbol{S}$	$Persistent\ states\ (I)$	$Transient\ states\ (T)$
$\boldsymbol{T}$	$Zero\ states\ (0)$	$Transient\ states\ (Q)$

The decomposition is performed to analyze the evolution of the system from a transient state by making use of the transient sub-matrices T and Q. This enable one to understand the steady state of the system, or the probability that the system will eventually end up in a specific sink system state. Hence, for analysis purpose, layer-3 of the Markov model i.e., all unit failure is considered as the sink state of the system. This consideration is valid since as time $t \to \infty$, the system will end up in any of the states within layer-3. The sub-matrices I and Q are square matrix, and T are rectangular matrix. Hence, if the total number of states in the system is n states and the number of absorbing states is m, then the total number of non-absorbing states is $(n - m)$ states. It is known that the sub-matrix I is composed of absorbing states since it is an identity matrix, and thus have the dimension of $(m \times m)$ matrix. Since Q is a non-absorbing transient square sub-matrix, its dimension is $[(n - m) \times (n - m)]$ matrix, which lead us to the matrices T and 0 having a dimension of $[m \times (n - m)]$ and $[(n - m) \times m]$ respectively. It is obvious from the above matrix decomposition that the transient group (T) describe the transition of states from transient group to persistent groups. Whereas, the transient group (Q) describe the transition of system states from transient group (Q) to transient group (T). The transient sub-matrices are extracted from the main transition probability matrix for further analysis of matrices T and Q. The fundamental matrix (N) of P can be determined using:

$$N = (I - Q)^{-1} \tag{4-14}$$

Key information can be extracted from the matrix N. The sum of the elements of the j^{th} column of N gives the expected absorption time of the j^{th} column transient state to be absorbed into a persistent states (*See Table 29*). For example, taking the last column of the fundamental matrix N,

the sum of the elements is $\cong 448$, which is the expected (mean) absorption time of the transient state 0 (all units normal) into any of the persistent states conditioned upon that the system started at state 0 (initial condition). A further analysis can be performed by considering the individual elements of the last column vector in matrix N. For instance, considering the last element of the column vector, which is $\cong 75$, it can be interpreted as the mean number of time the system is in state-0 given that the system started in state-0 initially. Or, considering the first element which is $\cong 17$; it can be interpreted as the number of time the system is in state-18 given that the system was initially in State-0. Similarly, other elements of the transition matrix can be explained.

The matrix TN gives the steady-state probabilities for ending in any absorbing state given that the system started in a transient state. The $(i,j)^{th}$elements of the transient matrix TN is the probability of being absorbed into persistent or absorbing state i from the transient state j. This absorption probability of the transient states is represented by α_{ji}. For illustration purpose considering the matrix element $\alpha_{0,26} = 0.125$, it can be interpreted as the probability of transition from the transient State-0 to the absorbing State-26. Or, considering the column vector of TN $\alpha_{0,i} = 0.125$, which means if the system started at State-0 (all units normal), it is equally likely for the system to end up in any of the absorbing states. Again, consider the first column of the matrix TN. It can be observed that $\alpha_{18,26} = \alpha_{18,22} = 0.5$, i.e., the probability of the system being absorbed in state-26 given that the system was in state-18 is equally likely, whereas the other elements $\alpha_{18,j}$ is zero. This can be validated by going back to the Markov model state representation of the system, i.e., if the system is in state-18 ($n_1 = 0, n_2 = 2, n_3 = 2$), it can only make a transition to either State-22 or 26 ($n_1 = 1\ or\ 2, n_2 = 2, n_3 = 2$). This is due to the fact that the BS is a non-repairable system. The matrix TN can be validated as the columns of the matrix must always be equal to 1.00.

Table 29: Fundamental matrix $N = (I - Q)^{-1}$

N= Inv (I - Q)																		
System States																		
18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
219	0	0	0	0	0	0	0	0	0	0	0	60.7	0	44.4	0	0	0	16.5
0	219	0	0	0	0	0	0	0	0	0	0	0	60.7	44.4	0	0	0	16.5
0	0	219	0	0	0	0	0	0	0	0	0	60.7	0	0	44.4	0	0	16.5
0	0	0	219	0	0	0	0	0	0	0	0	0	60.7	0	44.4	0	0	16.5
0	0	0	0	175	0	0	0	0	0	0	0	38.7	0	0	0	30.8	0	9.8
0	0	0	0	0	175	0	0	0	0	0	0	0	38.7	0	0	30.8	0	9.8
0	0	0	0	0	0	320	0	0	0	0	0	0	0	94.9	0	103.3	0	37.8
0	0	0	0	0	0	0	320	0	0	0	0	0	0	0	94.9	103.3	0	37.8
0	0	0	0	0	0	0	0	175	0	0	0	38.7	0	0	0	0	30.8	9.8
0	0	0	0	0	0	0	0	0	175	0	0	0	38.7	0	0	0	30.8	9.8
0	0	0	0	0	0	0	0	0	0	320	0	0	0	94.9	0	0	103.3	37.8
0	0	0	0	0	0	0	0	0	0	0	320	0	0	0	94.9	0	103.3	37.8
0	0	0	0	0	0	0	0	0	0	0	0	97.5	0	0	0	0	0	11.3
0	0	0	0	0	0	0	0	0	0	0	0	0	97.5	0	0	0	0	11.3
0	0	0	0	0	0	0	0	0	0	0	0	0	0	130.3	0	0	0	27.7
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	130.3	0	0	27.7
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	113.4	0	19.2
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	113.4	19.2
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	74.9
219	219	219	219	175	175	320	320	175	175	320	320	296.5	296.5	408.7	408.7	381.6	381.6	447.8

Table 30: Matrix TN

TN matrix																			
Absorbing System States (i)	*Transient System States* (j)																		
	18	*17*	*16*	*15*	*14*	*13*	*12*	*11*	*10*	*9*	*8*	*7*	*6*	*5*	*4*	*3*	*2*	*1*	*0*
26	0.5	0	0	0	0.5	0	0.5	0	0	0	0	0	0.25	0	0.25	0	0.25	0	0.125
25	0	0.5	0	0	0	0.5	0.5	0	0	0	0	0	0	0.25	0.25	0	0.25	0	0.125
24	0	0	0.5	0	0.5	0	0	0.5	0	0	0	0	0.25	0	0	0.25	0.25	0	0.125
23	0	0	0	0.5	0	0.5	0	0.5	0	0	0	0	0	0.25	0	0.25	0.25	0	0.125
22	0.5	0	0	0	0	0	0	0	0.5	0	0.5	0	0.25	0	0.25	0	0	0.25	0.125
21	0	0.5	0	0	0	0	0	0	0	0.5	0.5	0	0	0.25	0.25	0	0	0.25	0.125
20	0	0	0.5	0	0	0	0	0	0.5	0	0	0.5	0.25	0	0	0.25	0	0.25	0.125
19	0	0	0	0.5	0	0	0	0	0	0.5	0	0.5	0	0.25	0	0.25	0	0.25	0.125
Sum	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Limitations of the time-dependent Markov chain can be observed after a detail analysis of state conditional probability matrix, and also recalling the system qualitative analysis performed in Section 4.4. Note that, these limitations were underlined with the knowledge of dynamic PSA.

- For the classical Markov chain, it was assumed that the system states 19 to 26 are absorbing states, however this is not true from the perspective of dynamic PRA, i.e., states 19, 22 and 24 that belongs to the set of absorbing states are not necessarily absorbing since these states do not lead to system overflow or drained scenario.
- The system states 12, 13 and 17 are assumed as transient states in the classical Markov model, however this in not true from a dynamic PRA standpoint since these states lead to system overflow and drained condition. Thus, the system states 12, 13 and 17 must be considered as an absorbing states for dynamic PRA.
- The most important aspect to be taken from this analysis is that, the classical Markov model is concerned with only the system hardware states thus neglecting the dynamic evolution of the system. System failure is explicitly defined only on system configuration, i.e., system failure occurs when all the 3 units fail. This is not necessarily true since the system can end up in a quasi-stable state even if all the 3 units' fail (the phenomenon was observed in event tree analysis).

The conditional transition probabilities matrix $h(n/n', k\Delta t)$ generated can be coupled with system dynamics to generate a joint system probability matrix, and hence providing a possible approach to perform an integrated deterministic and probabilistic system assessment.

b. Computation of $g(j/j', n', k\Delta t)$

For determination of $g(j/j', n', k\Delta t)$, the sate space is first discretized into finite number of computational cells. The equal weight quadrature scheme was used to partition the state variable magnitude. It is important to note that the transition among discretized cells depends on both the initial cell location and the system configuration at that instant. Thus, care must be taken in defining the initial location of the state variable within a cell. For the BS, it is obvious that the controlled variable state space is one dimensional i.e., liquid level (x). CCMT requires the knowledge of top events in terms of state variable magnitude or computational absorbing cells. Discretization scheme of the state variable along with the control regions is shown in *Table 31*.

Table 31: Controlled variable discretization scheme

V_r	*Variable state* (J)	*Variable magnitude range*	V_j	*System State*
-	$j = 1$	$x < -3\ m$	V_1	Drained
$V_r = 1$	$j = 2$	$-3\ m \leq x \leq -1\ m$	V_2	Low
$V_r = 2$	$j = 3$	$-1\ m\ \leq x \leq +1\ m$	V_3	Normal
$V_r = 3$	$j = 4$	$+1\ m\ \leq x \leq +3\ m$	V_4	High
-	$j = 5$	$x > +3\ m$	V_5	Overflow

Note: The top events are shaded (green) in the table, i.e., overflow and drained

The three (3) control regions V_r can be further divided into arbitrary number of sub-cells depending on the analyst choice and the outcome desired. For the benchmark system, since $\widetilde{\mathrm{V}}_{\mathrm{r}}$ are disjoint intervals themselves, V_j are identical to $\widetilde{\mathrm{V}}_{\mathrm{r}}$ for minimum $\tilde{J}_r$ (i.e., $\tilde{J}_r = 1$) and partitioning is unique with a single departure point $P = 1$ from each region. Of course, more number of sub-cells increases the result accuracy. However, computational complexity and intensity increases with increased number of sub-cells/cells. A possible equal weight quadrature scheme with increased number of sub-cells is shown in *Table 32*. For the current analysis, $P = 3$ is chosen for demonstration of the methodology at the same time balancing the accuracy and computation complexity. It can be observed from *Table 32* that there are three departure points $P = 3$ from each sub-cells, or the initial liquid level in the tank which is an input for system simulation.

Table 32: Cell discretization via equal weight quadrature scheme

<table>
<tr><th rowspan="2">J</th><th rowspan="2">*Liquid Level* (x)</th><th colspan="3">*Equal weight Quadrature scheme*</th></tr>
<tr><th>$P = 1$</th><th>$P = 3$</th><th>$P = 5$</th></tr>
<tr><td>$j = 1$</td><td>$x < -3\ m$</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td rowspan="6">$j = 2$</td><td rowspan="6">$-3\ m \leq x \leq -1\ m$</td><td rowspan="6">$-2\ m$</td><td rowspan="2">$-2.5\ m$</td><td>$-2.75\ m$</td></tr>
<tr><td>$-2.25\ m$</td></tr>
<tr><td>$-2.0\ m$</td><td>$-2.0\ m$</td></tr>
<tr><td rowspan="2">$-1.5\ m$</td><td>$-1.75\ m$</td></tr>
<tr><td>$-1.25\ m$</td></tr>
<tr><td>$j = 3$</td><td>$-1\ m\ \leq x \leq +1\ m$</td><td>$0\ m$</td><td>$-0.5\ m$</td><td>$-0.75\ m$</td></tr>
</table>

				$-0.25\ m$
			$0\ m$	$0\ m$
			$+0.5\ m$	$+0.25\ m$
				$+0.75\ m$
$j = 4$	$+1\ m \leq x \leq +3\ m$	$+2\ m$	$+1.5\ m$	$+1.25\ m$
				$+1.75\ m$
			$+2.0\ m$	$+2.0\ m$
			$+2.5\ m$	$+2.25\ m$
				$+2.75\ m$
$j = 5$	$x > +3\ m$	-	-	-

The state space discretization can be fairly mechanized by implementing it in a computer code given that the control space and number of sub-cells are defined. The source code for state space discretization is shown in the *Appendix IV*. Once the control space is discretized into sub-spaces, the trajectories of the state vector in the discretized space must be determine. For the benchmark system, the dynamic evolution of the liquid level in the tank is defined by:

$$\frac{dx(t)}{dt} = f_n \tag{4-15}$$

Or, *Equation (4-15)* can be re-write as:

$$dx(t) = f_n * dt$$

$$x(t + \Delta t) - x(t) = f_n * \Delta t \tag{4-16}$$

Thus, liquid level in the next time step is given by:

$$x(t + \Delta t) = x(t) + f_n * \Delta t \tag{4-17}$$

Here, we can re-define the above equation for our analysis. It is considered here that:

$$x(t + \Delta t) = x(t) + f_{n'} * \Delta t \tag{4-18}$$

The above system dynamics is written in n' terms due to the assumption in the methodology that the component state combination does not change for a small-time step Δt, i.e., the computation

of $g(j/j', n', k\Delta t)$ is conditioned upon that n' does not change in the small time step Δt. In fact, $g(j/j', n', k\Delta t)$ itself constitute a Markov chain. Since all the parameters on the right hand side of the *Equation 42* is known or can be defined by the user, $x(t + \Delta t)$ can be determined with ease for any given time step. The dimension of the state vector trajectories $x(t + \Delta t)$ is dependent on the number of departure points $x(t)$, with both having the same dimension, and hence $f_{n'} * \Delta t$. Once $x(t + \Delta t)$ is computed from the system code, $g(j/j', n', k\Delta t)$ can be determined by integrating over the control space.

For illustration purpose, first consider the control region-2 which is partitioned into 3 equally distributed sub-cells. Hence, there are three trajectory segments departing from V_2 at time t with the same n', and arriving in IC number of V_j at $(t + \Delta t)$, depending on location of the controlled variables within V_2 which is the $V_{j'}$ for this case. For the benchmark system at state $n'(t) = 1$, if the trajectory segments depart from $V_{21}, V_{22}\ and\ V_{23}$ at time t, then the arrival points V_j at $(t + \Delta t)$ and hence $g(j/j', n', k\Delta t)$ can be determined using the below algorithm. Since initially the state variable is in $V_{21}, V_{22}\ and\ V_{23}$ sub-cell which are within the cell V_2 at time t, the total probability of $P_{rj,n'}(t)$ must be:

$$P_{21,1}(t) + P_{22,1}(t) + P_{23,1}(t) = 1 \tag{4-19}$$

Here;

$$P_{rj,n'}(t) = P_{21,1}(t) = Pr\{n(t) = 1, x(t) \in V_{21}/x(t) \in V_2\} \tag{4-20}$$

$$P_{22,1}(t) = Pr\{n(t) = 1, x(t) \in V_{22}/x(t) \in V_2\} \tag{4-21}$$

$$P_{23,1}(t) = Pr\{n(t) = 1, x(t) \in V_{23}/x(t) \in V_2\} \tag{4-22}$$

By definition or by equal weight quadrature discretization scheme, the sub-cells $V_{21}, V_{22}\ and\ V_{23}$ have equal volumes, i.e.,

$$P_{21,1}(t) = P_{22,1}(t) = P_{23,1}(t) = \frac{1}{3} \tag{4-23}$$

Hence, the transition from V_1 to V_1 can be determined as follow:

$$g(V_2/V_2, 1) = g(V_2/V_{21}, 1).P_{21,1}(t) + g(V_2/V_{22}, 1).P_{22,1}(t) + g(V_2/V_{23}, 1).P_{23,1}(t) \tag{4-24}$$

$$g(V_2/V_2, 1) = (1 * 0.333) + (1 * 0.333) + (1 * 0.333)$$

$$g(V_2/V_2\,,1) = 1.00$$

And hence $g\left(V_j/V_2\,,1\right) = 0$ for $j \neq 1$, since the state vector trajectories will remain in region-2 will 100% probability rather than making a transition to the other control regions. For the second case, consider for $n' = 12$ with the state vector depart from $V_{21}, V_{22}\ and\ V_{23}$ at time t which is represented by $V_{j'}$. The state vector trajectories and the arrival point V_j can be computed a follow. By equal weight quadrature discretization scheme:

$$P_{21,12}(t) = P_{22,12}(t) = P_{23,12}(t) = \frac{1}{3} \tag{4-25}$$

The transition from V_2 to V_2 for $n' = 12$ can be determined as:

$$\begin{aligned} g(V_2/V_2\,,12) = g(V_2/V_{21}\,,12).P_{21,12}(t) + g(V_2/V_{22}\,,12).P_{22,12}(t) \\ +g(V_2/V_{23}\,,12).P_{23,12}(t) \end{aligned} \tag{4-26}$$

$$g(V_1/V_1\,,12) = (1 * 0.333) + 0 + 0$$

$$g(V_2/V_2\,,12) = 0.333$$

For transition from V_2 to V_3, we have:

$$\begin{aligned} g(V_3/V_2\,,12) = g(V_3/V_{21}\,,12).P_{21,12}(t) + g(V_3/V_{22}\,,12).P_{22,12}(t) \\ +g(V_3/V_{13}\,,12).P_{23,12}(t) \end{aligned} \tag{4-27}$$

$$g(V_3/V_2\,,12) = 0 + (1 * 0.333) + (1 * 0.333)$$

$$g(V_3/V_2\,,12) = 0.667$$

Hence, $g\left(V_j/V_2\,,1\right) = 0$ for $j \neq 2,3$; since the state vector trajectories will remain in region-2 with probability of 33 % and in region-3 with a probability of 66.7 %. Similarly, the rest of the trajectories can be computed mechanically via computer codes as shown in the *Table 33.* $Pr\{g(j/j', n', k\Delta t)\}$ is a 5×5 matrix for each unit state combinations, which results in 135×5 dimensional matrix for 27 unit state combination.

Table 33: $Pr\{g(j/j',n',k\Delta t)\}$ for $k = 1$

SI.No.	n'	*From/to*	j					*Sum*
		j'	**1**	**2**	**3**	**4**	**5**	
1	0	**1**	1	0	0	0	0	1.00
2	0	**2**	0	1	0	0	0	1.00
3	0	**3**	0	0	1	0	0	1.00
4	0	**4**	0	0	0	1	0	1.00
5	0	**5**	0	0	0	0	1	1.00
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
131	26	**1**	1	0	0	0	0	1.00
132	26	**2**	0	0.33	0.67	0	0	1.00
133	26	**3**	0	0	0.33	0.67	0	1.00
134	26	**4**	0	0	00	0.33	0.67	1.00
135	26	**5**	0	0	0	0	1	1.00

c. Computation of transition probability matrix $q_{n,j}^{n',j'}(k\Delta t)$

The elements of the overall system transition matrix $q_{n,j}^{n',j'}(k\Delta t)$ is a function of both the cell to cell transition probabilities $g(j/j',n',k\Delta t)$ and the conditional unit state transition probabilities $h(n/n',j' \to j,k\Delta t)$. $q_{n,j}^{n',j'}(k\Delta t)$ can be computed as follow:

$$q_{n,j}^{n',j'}(k\Delta t) = g(j/j',n',k\Delta t).h(n/n',j' \to j,k\Delta t) \quad (4\text{-}28)$$

The sum of the column elements of $q_{n,j}^{n',j'}(\Delta t)$ must be equal to 1.00. Hence, the transition matrix can be verified and validated once obtained before performing further computation. Since there are 27 distinct system states and 5 cells of the controlled variable state space, the $q_{n,j}^{n',j'}(k\Delta t)$ matrix will have an overall matrix dimension of $27 \times 5 = 135$ rows and 135 columns, where $N = 27$ and $J = 5$. Thus, the transition matrix $q_{n,j}^{n',j'}(k\Delta t) = 135 \times 135$ square matrix (a small portion of which shown in *Table 34*).

Table 34: A small portion of the overall system transition matrix $q(n, j/j', n', k\Delta t)$ for $k = 1$

q(n,j/n',j',kdt)			n' =1					n' =2					n' =3					n' =4					n' =5					n' =6		
			j'					j'					j'					j'					j'					j'		
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28
n=1	j	1	0.98665	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		2	0	0.98665	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		3	0	0	0.98665	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		4	0	0	0	0.98665	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		5	0	0	0	0	0.98665	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
n= 2	j	6	0.00226	0	0	0	0	0.99118	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		7	0	0.00226	0	0	0	0	0.99118	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		8	0	0	0.00226	0	0	0	0	0.99118	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		9	0	0	0	0.00226	0	0	0	0	0.99118	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		10	0	0	0	0	0.00226	0	0	0	0	0.99118	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
n= 3	j	11	0.00226	0	0	0	0	0	0	0	0	0	0.99118	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		12	0	0.00226	0	0	0	0	0	0	0	0	0	0.99118	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		13	0	0	0.00226	0	0	0	0	0	0	0	0	0	0.99118	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		14	0	0	0	0.00226	0	0	0	0	0	0	0	0	0	0.99118	0	0	0	0	0	0	0	0	0	0	0	0	0	
		15	0	0	0	0	0.00226	0	0	0	0	0	0	0	0	0	0.99118	0	0	0	0	0	0	0	0	0	0	0	0	
n= 4	j	16	0.00284	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.99232	0	0	0	0	0	0	0	0	0	0	0	
		17	0	0.00284	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.99232	0	0	0	0	0	0	0	0	0	0	
		18	0	0	0.00284	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.99232	0	0	0	0	0	0	0	0	0	
		19	0	0	0	0.00284	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.99232	0	0	0	0	0	0	0	0	
		20	0	0	0	0	0.00284	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.99232	0	0	0	0	0	0	0	
n= 5	j	21	0.00284	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.99232	0	0	0	0	0	0	
		22	0	0.00284	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.99232	0	0	0	0	0	
		23	0	0	0.00284	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.99232	0	0	0	0	
		24	0	0	0	0.00284	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.99232	0	0	0	
		25	0	0	0	0	0.00284	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.99232	0	0	
n= 6	j	26	0.00155	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.98975	0	
		27	0	0.00155	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.98975	
		28	0	0	0.00155	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9897
		29	0	0	0	0.00155	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		30	0	0	0	0	0.00155	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

d. Computation of $P_{n,j}((k+1)\Delta t)$

The computation of $P_{n,j}(k\Delta t)$ for any specific time step can be performed using the inductive as well as the deductive algorithm. For the current case, an inductive algorithm has been implemented with the initial condition $P_{n',j'}(0)$ to determine $P_{n,j}((k+1)\Delta t)$. For analysis purpose, the system is assumed to be in normal condition with all the units being in normal state and the liquid level in the nominal region. This can be quantitively expressed as;

$$P_{n',j'}(0) = \begin{cases} 1 & ; for\ n' = 1, j' = 3 \\ 0 & ; otherwise \end{cases} \tag{4-29}$$

Or, the initial system condition is expressed as:

$$P_{1,1}(0) = 0;\ P_{1,2}(0) = 0;\ P_{1,3}(0) = 1;\ P_{1,4}(0) = 0;\ P_{1,5}(0) = 0;\ P_{2,1}(0) = 0; \ldots\ldots P_{N,J}(0) = 0$$

Once the initial condition and the time step are fixed, $P_{n,j}((k+1)\Delta t)$ can be computed utilizing $q(n, j/j', n', k\Delta t)$ and by implementing the Forward algorithm. In order to determine the top event probabilities at any given time step, we must recall the space discretization scheme in which $j = 1$ and $j = 5$ are considered as absorbing cells in the system. Hence, the top events are described in terms of the value of computational cells j. More specifically, $j = 1$ implies liquid level $x < -3\ m$ which is the Drained condition, and $j = 5$ implies $x > +3\ m$ which is the overflow condition. Thus, for a specific time step, summing up all the elements of $P_{n,j}\big((k+1).\Delta t\big)$ for $j = 1$ and $j = 5$ will give the probabilities of system drained and overflow scenario respectively. If the overall probability of system failure (overflow + drained) at any given time step is represented by $P_{OD}(k\Delta t)$, it can be written as the summation of:

$$P_{OD}(k\Delta t) = \sum_{n=1}^{N} P_{n,1}(k\Delta t) + \sum_{n=1}^{N} P_{n,5}(k\Delta t) \tag{4-30}$$

Hence the system reliability $R(k\Delta t)$ is given by *(See Table 35)*:

$$R(k\Delta t) = 1 - P_{OD}(k\Delta t) \tag{4-31}$$

Table 35: Top event probabilities for the benchmark system

Time step	$P_{n,1}(k\Delta t)$	$P_{n,5}(k\Delta t)$	$P_{OD}(k\Delta t)$
$k = 0$	0	0	0
$k = 1$	0	0	0
$k = 2$	0	$1.02\text{E} - 08$	$1.02\text{E} - 08$
$k = 3$	$1.86\text{E} - 07$	$9.17\text{E} - 05$	$9.19\text{E} - 05$
$k = 4$	$2.15\text{E} - 06$	$5.11E - 04$	$5.14\text{E} - 04$

Table 36: Sample state transition probabilities

$\boldsymbol{P_{n,j}}$	$\boldsymbol{P(k=0)}$	$\boldsymbol{P(k=1)}$	$\boldsymbol{P(k=2)}$
$P_{0,3}$	1	0.9867	0.96
$P_{1,3}$	0	0.0023	0.0067
$P_{2,3}$	0	0.0023	0.0067
$P_{3,3}$	0	0.00283	0.0083
$P_{4,3}$	0	0.00283	0.0083
$P_{5,3}$	0	0.00155	0.00457
$P_{6,3}$	0	0.00155	0.00457
$P_{7,3}$	0	$6.50\text{E} - 06$	$5.77\text{E} - 05$
$P_{8,3}$	0	$6.50\text{E} - 06$	$5.77\text{E} - 05$
$P_{9,3}$	0	$3.55\text{E} - 06$	$3.14\text{E} - 05$
$P_{10,3}$	0	$3.55\text{E} - 06$	$3.14\text{E} - 05$
$P_{11,3}$	0	$6.50\text{E} - 06$	$5.77\text{E} - 05$
$P_{12,3}$	0	$6.50\text{E} - 06$	$5.12\text{E} - 05$
$P_{12,4}$	0	0	$6.46\text{E} - 06$
$P_{13,3}$	0	$3.55\text{E} - 06$	$2.79\text{E} - 05$
$P_{13,4}$	0	0	$3.51\text{E} - 06$
$P_{14,3}$	0	$3.55\text{E} - 06$	$3.14\text{E} - 05$
$P_{15,3}$	0	$4.44\text{E} - 06$	$3.95\text{E} - 05$
$P_{16,3}$	0	$4.44\text{E} - 06$	$3.95\text{E} - 05$

$P_{17,3}$	0	$4.44\text{E}-06$	$3.51\text{E}-05$
$P_{17,4}$	0	0	$4.40\text{E}-06$
$P_{18,3}$	0	$4.44\text{E}-06$	$3.95\text{E}-05$
$P_{19,3}$	0	$1.02\text{E}-08$	$2.73\text{E}-07$
$P_{20,2}$	0	0	$1.02\text{E}-08$
$P_{20,3}$	0	$1.02\text{E}-08$	$2.62\text{E}-07$
$P_{21,3}$	0	$1.02\text{E}-08$	$2.42\text{E}-07$
$P_{21,4}$	0	0	$3.06\text{E}-08$
$P_{22,3}$	0	$1.02\text{E}-08$	$2.73\text{E}-07$
$P_{23,3}$	0	$1.02\text{E}-08$	$2.42\text{E}-07$
$P_{23,4}$	0	0	$3.04\text{E}-08$
$P_{24,3}$	0	$1.02\text{E}-08$	$2.73\text{E}-07$
$P_{25,3}$	0	$1.02\text{E}-08$	$2.02\text{E}-07$
$P_{25,4}$	0	0	$6.08\text{E}-08$
$P_{25,5}$	0	0	$1.02\text{E}-08$
$P_{26,3}$	0	$1.02\text{E}-08$	$2.42\text{E}-07$
$P_{26,4}$	0	0	$3.04\text{E}-08$

Note: Only the non-zero state probabilities are depicted in the table. Of course, as the time step increases the whole state space will be covered.

For discussion and illustration purpose, consider *Table 35* and *Table 36*.

1. For $k = 1$, the top event probability $P_{n,5}(k\Delta t) = 0$. This implies that the system overflow scenario will not occur in a single time step given that the system was in the nominal region at $k = 0$. It can be observed that even though the system does not result to a condition, there exist some probabilities for the system to make transitions from the system nominal operation state to several other system states. This can be evidently observed in the table, where the $P_{n,j}(1)$ elements for $n \neq 0$ and $j = 3$ have non-zero probabilities. These elements can be interpreted as the probabilities of the system states transition due to the stochastic nature of the unit failures, however without leading to any system failure condition. This provide the analyst an important information that a defined top event in terms of state vector magnitude does not

necessarily occur even if a hardware/component failure may occur at a given time. This is due to the fact that the system dynamics have been accounted for, and the initial/previous position of the state vector at $((k-1)\Delta t)$ in the discretized space and its subsequent evolution for a given system state dictates the top event probabilities. In contrast to the FT analysis where instant system failure occurs, or the top event probability is dictated only by the state of the basic events, thus neglecting dynamic evolution of state vector. The methodology also provides significant quantitative information on the possible states a system can take at any point in time. Furthermore, the probability of the system being in normal condition i.e., all units normal and liquid level at nominal region is 0.986. This value implies the dynamic reliability of the system or an explicit representation of hardware state and state variable. Besides the normal operating condition, the most probable state that the system can take in a single time step are $P_{3,3}(1) = P_{4,3}(1) = 2.84\text{E}-03$. $P_{3,3}(1)$ and $P_{4,3}(1)$ both implies a change in system state $[(n_1 = 0 \cap n_2 = 1 \cap n_3 = 0)\ and\ (n_1 = 0 \cap n_2 = 2 \cap n_3 = 0)]$, but constant state variable magnitude (Region-3).

2. For $k = 2$, summing up all the elements for $j = 5$, we obtain the top event probability of system overflow is $P_{n,5}(2) = 1.02\text{E}-08$ and system drained is $P_{n,1}(2) = 0$. And the probability of the system being in normal condition is 0.96. It is self-explanatory that the model output explicitly identifies system configuration, state variable magnitude and unit state order.

The obtained results of $P_{n,j}\big((k+1)\Delta t\big)$ can be validated by checking the necessary condition which is; $\sum_{n=1}^{N}\sum_{j=1}^{J} P_{n,j}(k\Delta t) = 1$. This validation assures the analyst that the computation was performed correctly with the computational cells covering the entire space without overlapping, and that the probability of finding the state vector x in the discretized space j for a given system configuration n at any given time step $k\Delta t$ is 1.00.

4.7. Conclusion and Comparison

In this chapter, a detail analysis of the BS was performed using FT/ET analysis, classical Markov model, DFM and coupled Markov-CCMT method. It was observed that the dominant classical techniques (ET/FT) have limitations in the modeling and treatment of time-dependent interactions that shape dynamic event sequences, and state variable evolution. For instance, for the BS, the end states depend on the order of unit failure, timing of failure events and magnitude of the state

variable. Classical PRA approach which is static in nature do not have the capability to capture these interactions and may omit risk-significant event sequence. Classical Markov model have the capability to capture time-dependent system behavior through state transition matrix. However, the model lacks the capability to capture system dynamics. Dynamic PRA methods attempt to address the above-mentioned issues and drawbacks by accounting simultaneously for the time element, stochastic state transition and dynamic system evolution. A comparison of the methodologies is presented below.

1. Predicted benchmark system failure probability

The failure probability of the benchmark system predicted by each of the methodologies for four (4) time steps are presented in the tables below.

Table 37: Predicted failure probability of the BS using FT (binary)

FT (binary)	*Predicted system failure probability*			
	$t = 1$	$t = 2$	$t = 3$	$t = 4$
Total failure probability	$3.15E-03$	$6.36E-03$	$9.61E-03$	$1.29E-02$

Table 38: Predicted failure probability of the BS using FT (multi-state)

FT (multi-states)	*Predicted system failure probability*			
	$t = 1$	$t = 2$	$t = 3$	$t = 4$
Overflow	$1.46E-05$	$5.82E-05$	$1.31E-04$	$2.33E-04$
Drained	$1.02E-08$	$8.16E-08$	$2.75E-07$	$6.53E-07$
Total failure probability	$1.46E-05$	$5.83E-05$	$1.31E-04$	$2.34E-04$

Table 39: Predicted failure probability of the BS using ET

Event Tree	*Predicted system failure probability*			
	$t = 1$	$t = 2$	$t = 3$	$t = 4$
Overflow	$2.91E-05$	$1.83E-04$	$5.69E-05$	$1.29E-03$
Drained	$3.06E-08$	$2.45E-07$	$8.26E-07$	$1.96E-06$
Total failure probability	$2.91E-05$	$1.83E-04$	$5.70E-04$	$1.29E-03$

Table 40: Predicted failure probability of the BS using Markov model with the qualitative consideration taken in *Table 12*.

Markov model	*Predicted system failure probability*			
	$t = 1$	$t = 2$	$t = 3$	$t = 4$
Overflow	0	$2.91E-05$	$8.68E-05$	$1.73E-04$
Drained	0	0	0	$6.12E-08$
Total failure probability	0	$2.91E-05$	$8.68E-05$	$1.73E-04$

For the dynamic methods presented in the below tables, the initial condition of the tank level was assumed to be in the nominal region.

Table 41: Predicted failure probability of the BS using Markov-CCMT model

Markov-CCMT model	*Predicted system failure probability*			
	$t = 1$	$t = 2$	$t = 3$	$t = 4$
Overflow	0	$1.02E-08$	$9.17E-05$	$5.11E-04$
Drained	0	0	$1.86E-07$	$2.15E-06$
Total failure probability	0	$1.02E-08$	$9.19E-05$	$5.14E-04$

Table 42: Predicted failure probability of the BS using DFM

DFM	*Predicted system failure probability*			
	$t = 1$	$t = 2$	$t = 3$	$t = 4$
Overflow	0	$3.345E-05$	$9.54E-05$	$1.86E-04$
Drained	0	$1.36E-07$	$4.985E-06$	$2.345E-05$
Total failure probability	0	$3.36E-05$	$1.00E-04$	$2.09E-04$

2. Multistate Modelling

Fault Tree Analysis:

Classical FTA is based on binary representation of component/system states, and hence only two states (normal and failed) are analyzed for system reliability assessment. However, realistically, there is always a possibility for a component/system to have several failure modes. This is especially true while modelling a redundant system. For instance, a valve can fail-high, fail-closed,

fail in 50% position, etc.; and a system can overflow, drained or quasi-stable. In FTA, component failure modes are treated independently, rather than in an integrate fashion (*refer Appendix I*). This result in an overestimation of system failure probability. Furthermore, for a system with multiple failure modes, each top event requires a separate construction of FT.

Event Tree Analysis:

Similar to the FTA, classical ETA is also based on binary representation of component/system states. ETA being an inductive approach provide a possibility to model multiple component or system states, in the sense that every possible failure mode of a component results in multiple system states. It may be observed from the BS analysis that, each unit failure mode results in 9 possible system states. However similar to the FTA, all failure modes of a component are treated independently, and each failure modes requires a separate analysis and construction of ET. Hence for a component with multiple failure modes the analysis become tedious and time consuming.

Classical Markov Model:

The classical Markov model clearly provide a superior way to analyze systems or components with multiple states as compared to the classical FT/ET (*Appendix I*). Markov model of a component/system takes into account the competition among states of a component, i.e., the likelihood of occurrence of a particular component state among all possible states. This provide the analyst a realistic result and does not overestimate the system failure probability. At a system level, it may be observed from the BS analysis that the model results in 27 distinct system states.

Dynamic Flowgraph Method:

DFM is based on a multi-valued logic, where multiple component/system states are expressed in terms of causal relationships among system parameters. Any number of component/system states can be enumerated in the decision table. For instance, a three-state variable (fail low, fail high and fail at 50% position) is represented as three inputs to a decision table resulting in different output of the decision table. Furthermore, a single DFM model once developed can analyze any system condition of possible interest, i.e., any number of top event of interest can be evaluated with a single model. This is a significant advantage over the classical PSA techniques.

Coupled Markov-CCMT Model:

Markov-CCMT model is based on a multi-state system characterization, with the feature to represent an arbitrary number of system states depending on the number of component states and

partitioned state variable states. It can be observed from the analysis of benchmark system that the model result in a 135 distinct system states. This provide significant information to an analyst with regard to possible states a system can take. However, a drawback in explicit states definition is that it is difficult to visualize or foresee the total set of possible states a system can take prior to scenario development.

3. Time Dependencies

Fault Tree and Event Tree Analysis:

The classical FT/ ET is based on static logic modelling, and hence do not account for the time element (exact timing of failure event) and lacks the capability to accurately quantify the event sequence probability deviation with time besides the state variable evolution in time.

Classical Markov Model:

The Markov model is a rigorous time-dependent model with explicit modelling of time element during states transition. Time-dependent system state probabilities can be obtained once the model is constructed. However, an accident sequence with state variable evolution in time cannot be captured in the model due to the fact that system dynamics is ignored in the model.

Dynamic Flowgraph Method:

The DFM can explicitly represent time element as well as the evolution of the state variable in time besides depicting the type of unit failure. All the PIs generated via DFM model are time-stamped and dictates the type of system failure scenario. The backtracking process can also be performed to unveil system state in the previous time steps (dependent on the depth of backtracking), as was demonstrated in this chapter. This allow one to observe the development of event sequence in time.

Coupled Markov-CCMT Model:

The coupled Markov-CCMT model can explicitly account for the time element as well as the state variable evolution. Besides the unit states, the exact timing of failure event is taken into account that eventually influence the end state (overflow or drained). The methodology utilizes real time, and thus emphasize on the exact timing of unit failure events as compared to the DFM model.

4. System Dynamics

Fault and Event Tree Analysis:

The classical FTA and ETA represent a system state with a set of success/failure states of the system components. The methodologies are neither developed nor is deliberated to model system dynamic response and its evolution. Thus, the classical methodologies lack the capability to capture unit/component dynamic interactions. It was observed that ET technique provide a better approach compared to FT to model systems that response dynamically to IEs, in that, it can provide a correct failure logic and a qualitative assessment of system end states.

Markov Model:

The classical Markov model is intended for discrete time discrete system state representation, but not to model system dynamics or state variable evolution, i.e., the deterministic aspect. The model is oriented to system hardware states modelling rather than system dynamics. Hence it lacks the capability to model dynamic system evolution.

Dynamic Flowgraph Method:

DFM provide a favorable approach to model system dynamics by explicit representation of time element and state variable evolution. Dynamic system behaviors are represented as a series of discrete state discrete time transitions and is modelled in the transition box decision tables. Furthermore, dynamic consistency rules are applied to eliminate impossible events and generate a consistent accident sequence. However, decision tables have to be constructed separately for varying time steps size and state variable intervals. The methodology is oriented towards and sensitive to the partitioned state variable interval size. A drawback of the methodology is that, the influence of state variable evolution on the failure/demand rate of the units cannot be accounted. The methodology rather focuses on probabilistic system dynamic and provide an overview of the evolution of failure as well as normal events for each time steps. Overall DFM can accurately represent, analyze and uncover all risk-significant dynamic event sequence.

Coupled Markov-CCMT model:

Due to the limitation of the classical Markov model to account for system dynamics, CCMT is coupled with Markov model to represent system dynamics in discrete-space discrete-time. The joint transition probability matrix of the model explicitly depicts evolution of the state variable, the conditional dependencies between the unit states. Thus, the model captures the unit states

transition caused by deterministic laws as well as by stochastic nature of the units. This feature of the methodology enables one to simultaneously account for system dynamics and stochastic state transition (tightly coupled). In addition to accounting for time-dependent system states probability, the model has the capability to capture the possibility of failure frequency deviation with state variable evolution i.e., likelihood of a subsequent failure given an IE due to the state variable evolution. For instance, if all the units are in normal state and the state variable make a transition from region $j' \rightarrow j = 2 \rightarrow 3$ at $t = 0$, the deterministic law will demand unit-1 to remain open, unit-2 to turn off and unit-3 to open. Hence, the possible unit failure states are: unit-1 can fail-closed, unit-2 can fail-open and unit-3 can fail-closed. Unit-1 fail-open failure rate is not taken into account since this state of the unit does not contribute to the system failure conditioned upon that the control laws demand the unit to open. Hence, for this particular case unit-1 fail-open rather contribute to the success of the system at this particular time step and state variable magnitude transition. The same argument holds for the case of unit-2 and unit-3. This dynamic scenario and system response to an event thus changes the probability of unit failure, besides the deviation of unit failure rate with time.

5. Event Ordering

Fault Tree Analysis:

The FTA is not intended to model state or failure event ordering, rather it represents the top event in terms of combinations of basic events (minimal cut-sets) without any particular ordering index.

Event Tree Analysis:

In ETA, the sequence ordering in pre-set by the analyst as was observed and pointed out in the ET of the benchmark system. Hence, the methodology lacks the capability to capture risk significant dynamic event sequences that can probabilistically evolve from dynamic system interactions, which would then remain uncovered.

Classical Markov Model:

In the Markov model, the order in which a unit/component state occur in the accident sequence is explicitly modelled due to the dependence of top events on state ordering. This is especially true for systems with multiple top events. The state sequence and possible number of system state is a priori fixed by the analyst. It can be observed from the BS analysis that each of the 27 states have

unique state ordering, and the probability of occurrence of a state is dictated not only by time and transition rate, but also by the order in which unit failure occurs. The state ordering is simplified by categorizing the system states in a layer wise approach (as done in this case).

Dynamic Flowgraph Method:

In DFM, the event sequence is not pre-determined by the analyst but rather the sequence evolves probabilistically from the time-dependent system model, and dynamic interactions among the units. The methodology strictly takes into account the order of unit failures in each time step. This allow one to observe the time-dependent scenario development in an orderly fashion. The dynamic accident sequence is presented as a set of timed-prime implicants for any top event of interest with explicit representation of the state variable magnitude and time element. Size of the PIs as well as the number of PIs increases with increasing time steps, i.e., similar to the logic 'AND' and 'OR' gate in FT where AND gate increases the MCS size and OR gate increases the number of MCSs.

Coupled Markov-CCMT:

Similar to the classical Markov model, state sequence is explicitly modelled, and the possible number of system states are a priori set by the analyst. This is also a limitation since it is difficult to envision all the possible system states prior to scenario development. Only the state probabilities changes, and the model is memoryless, in that, time-dependent sequence development cannot be observed. Unlike DFM, the order, size and number of system states in Markov-CCMT model remains fixed throughout the analysis. Here, DFM provide an advantage as it allow one to observe the dynamic event sequence.

6. Computational Demand, Complexity and Time

Fault Tree and Event Tree Analysis:

The classical FTA and ETA are well-established methodologies that can be implemented systematically requiring less computational resource and time. The time requirement however may increase significantly depending on the number of top events and IEs for FT and ET respectively.

Classical Markov Model:

For system with large number of possible system states, the computational demand and complexity increases significantly. Furthermore, construction and modelling of the transition diagram can be considerably time consuming depending on the number of states.

Dynamic Flowgraph Method:

In comparison with other methodologies, DFM provide the flexibility for system analysis via inductive as well as deductive approach. However, construction of decision tables can become complex and time consuming with increased number of system parameters and discretized states, requiring other pseudo codes for decision table construction. For instance, decision table TT1 for the BS was constructed in MATLAB. The methodology also requires a highly time-dependent system model which may require significant amount for time for model development. The results obtained from DFM model requires significant post-processing so that they can be integrated into an existing classical PRA model. The post-processing via backtracking process, and subsequent generation of timed-FTs requires significant amount of time and diligent application of consistency rules in every branching point, thus requiring a high analytical skill level.

Coupled Markov-CCMT:

It was observed from the BS analysis that the computational demand for coupling physical and stochastic model can be very intensive, complex and timing consuming. The computational increases significantly with the number of state variables, possible system states and discretized number of computational cells in the state-space. This phenomenon is well known as state space explosion. The size of the transition matrix can become very large requiring significant amount of time for its evaluation. In contrast to the above, the methodology provides features to validate the model at any point in time during its implementation. Another drawback of the methodology is that, it requires a highly time-dependent system model. Furthermore, post-processing of the results obtained from Markov-CCMT model requires significant time. Last but not the least, the analytical skill level required to model and implement the methodology is considerably demanding.

7. Interface

Fault Tree and Event Tree Analysis:

FTA and ETA are user friendly, and there exist several well-established computer codes such as CAFTA, FaulTree+ and RiskSpectrum.

Classical Markov Model:

Solving the set of coupled ODEs can be challenging with increased number of system states and transition among these states. However, there exist many well established numerical methods for

obtaining the ODE solutions and can be systematically implemented in computer codes such as Fortran95, C and Python.

Dynamic Flowgraph Method:

System modelling and analysis in DFM can be performed in DYMONDA code, a graphical user interface. This is a significant advantage over other dynamic techniques which does not have an integrated and user friendly interface.

Coupled Markov-CCMT Model:

There exist no well-established integrated computer codes for implementation of the methodology. Stochastic nature of the components and the deterministic aspects of the system have to be modelled separately using different codes, and then eventually coupling the two model. For the BS, the Markov model is implemented in Fortran95 and CCMT or deterministic system model is implemented in MATLAB. Thus, coupling of Markov-CCMT model is relatively complex and demands high computational time especially for realistic systems.

CHAPTER 5: AN INTEGRATED APPROACH FOR RELIABILITY ASSESSMENT OF PASSIVE SAFETY SYSTEM

5.1. Introduction

This chapter presents a preliminary roadmap and activities to be carried out for the project on "Design and Performance Assessment of Passive Engineered Safety Features in Advanced Small Modular Reactors". the method of selection of the technology and system to be analyzed is presented in *Section 5.2*. *Section 5.3* discusses a novel approach to modelling generic passive safety systems (PSSs) in an integrated framework for dynamic reliability assessment. *Section 5.4* provide a detail description of a generic isolation condenser system (ICS) to be used as a benchmark system for the CRP. Note that the current research is not design/plant specific but concerns a typical design of an ICS implemented in iPWR SMRs. *Section 5.5* characterize the ICS into macro-components or units, with each unit modelled separately using FT (failure rate evaluation method) and Markov model. *Section 5.6* integrates the individual units to predict the overall ICS system behavior. Note that only sample results are presented for illustration purpose.

5.2. Research Project Roadmap

This sub-section provide a roadmap for the coordinated research project (CRP) on, "Design and Performance Assessment of Passive Engineered Safety Features in integral PWR-type Small Modular Reactors". The task and responsibilities include:

- *For 2017:* literature survey of defence-in-depth being adopted in SMR designs and draft a technical report on technical description of passive engineered safety features in SMRs;
- *For 2018:* develop manual and support CRP participants in using certain PSA techniques;
- *For 2019:* draft the scope and main content of a technical document (project outcome).

Recognizing and taking into account the above responsibilities to be fulfilled, a review of current SMR technology and passive safety systems in advanced reactor designs was performed as an initial step of the project. An overview of the project roadmap is presented in *Figure 24*.

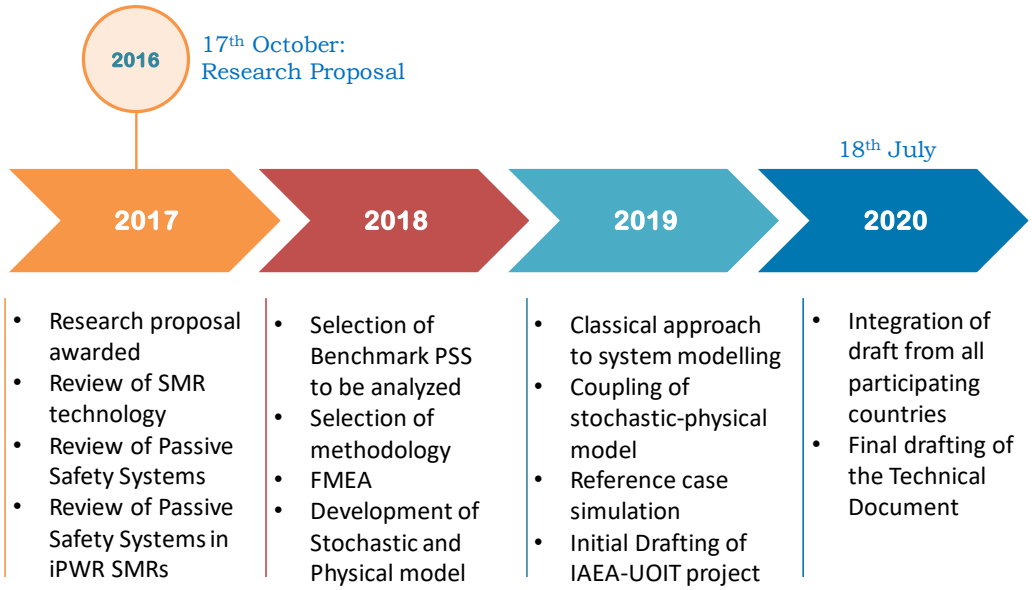


Figure 24: Proposed coordinated research project roadmap

Aforementioned in Chapter 4, Markov-CCMT model was chosen over DFM to model passive systems due to its capability to capture time-dependent dynamic interactions. The author intends to provide the basis of how dynamic PRA can be more applicable and suitable for passive systems analysis in iPWR-type SMRs.

1. SMRs are simple in design; thus, fewer number of components that address both steady and transient conditions. Yet even for smaller core, off normal reactivity control must be as quick as large reactors. Thus, post shutdown system must remove decay heat with little human intervention for approximately 72 hours mission time. Thus, the extent to which dynamic analysis like that presented in Chapter 4 should be investigated in order to assure that classical PRA is sufficient or insufficient;
2. Provided that iPWR-type SMRs as designed in simple terms with less number of components, the phenomenon of state-space explosion can be investigated to develop and demonstrate dynamic PRA methods.
3. Passive systems performance is not only dependent on the hardware state, but also on critical system parameters such as presence of non-condensable, heat loss, fouling, etc. These parameters which influence the system dynamics and hence the functional reliability must be taken considered in the analysis. The dynamic methods presented in Chapter 4

have the capabilities to account for these parameters in an integrated fashion. However, the applicability of the techniques to model and analyze passive systems reliability should be demonstrated via a benchmark system.

The IAEA defines PSSs as "a system that is composed entirely of passive components and structures or a system, which uses active components in a very limited way to initiate subsequent passive operation". PSSs can be categorized into four (4) class: [*IAEA-TECDOC-626*]

I. *Category A*: Characterized by systems that has no signal inputs of intelligence, no external power sources or forces, no moving mechanical parts and no moving working fluid, e.g., accumulators.
II. *Category B*: Characterized by systems that has no signal inputs of intelligence, no external power sources or forces, no moving mechanical parts but have a moving working fluid, e.g., passive containment cooling systems.
III. *Category C*: Characterized by systems that has no signal inputs of intelligence, no external power sources or forces, but composed of moving mechanical parts with or without moving working fluids, e.g., relief valves.
IV. *Category D*: Intermediary zone between active and passive where the execution of the safety function is made through passive methods, i.e., passive execution/active initiation, e.g., emergency shutdown systems based on gravity.

A comprehensive review of active, passive and hybrid safety systems implemented in SMRs technology with focus on iPWR design was performed. Additionally, the author categorized the PSSs based on their functions, including:

1. Passive residual heat removal systems (e.g., mPower)
2. Passive safety injection system (e.g., SMART)
3. Passive containment cooling system (e.g., NuScale)
4. Passive/automatic depressurization system (e.g., CAREM).

The iPWR SMR designs that implements passive systems to achieve the above mentioned functions are provided as an example. A stepwise assessment and flowchart of the reviews performed to select the reactor design and passive safety system to be used as a benchmark for the project is depicted in *Figure 25*.

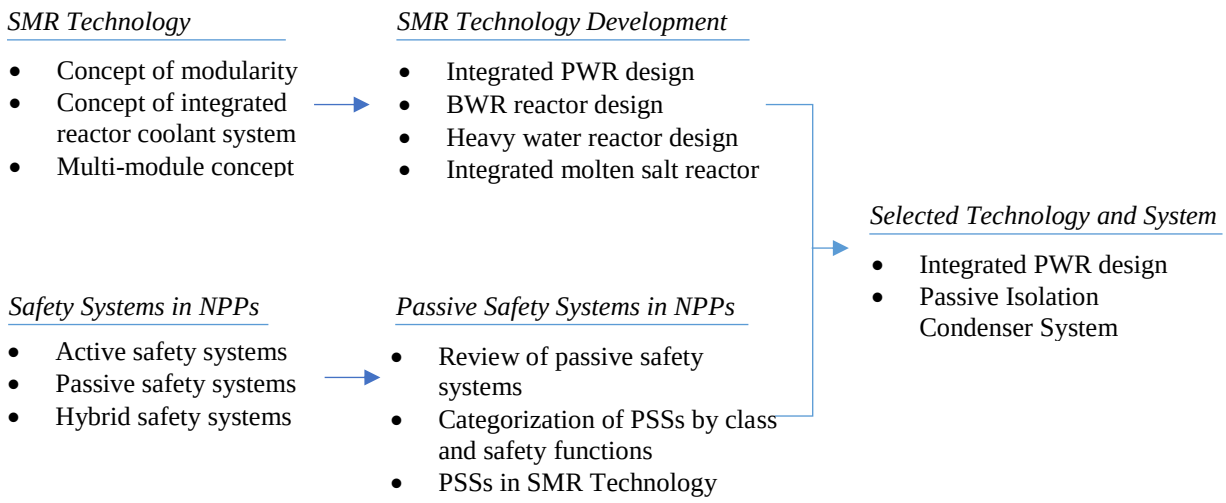


Figure 25: Stepwise review and flowchart for design and system selection

The passive ICS belonging to the category of passive reactor depressurization systems was selected to be used as a benchmark system for the project. According to the IAEA classification on passive systems, the ICS is within the Category D. The basis for the selection of ICS as a benchmark system includes:

- The ability to maintain reactor coolant system pressure without the loss of primary coolant;
- The ICS has been implemented in iPWR SMRs (e.g., CAREM, IRIS, AHWR) with varying objectives such as reactor depressurization, to maintain reactor hot standby and to reduce the frequency of SRVs operation;
- The ICS has been used as a benchmark system for validation of developing methodologies for reliability assessment of passive safety systems [*IAEA-TECDOC-1752*];
- Large reactor designs such as ESBWR has implemented ICS. This increases the availability of research materials on ICS, physical modelling approach and experimental data;
- An ongoing effort within the nuclear community to model and predict system behavior based on natural circulation.

After the selection of reactor design (i.e., iPWR) and passive safety system to be analyzed (i.e., the ICS), the next step is the selection of a methodology to assess and predict the deterministic as well as probabilistic behavior of the isolation condenser system. A comprehensive review of current methodologies for assessment of PSSs was performed, of which reliability methods for

passive safety functions (RMPS) *[Ricotti et al. (2002); Jafari et al. (2003); Marques et al. (2005)]* and Assessment of Passive System Reliability (APSRA) *[Nayak et al. (2007, 2008(a))]* methodologies provide a promising approach [*IAEA-TECDOC-1752 (2014); Zio et al. (2009)*]. The two methodologies differ from the fact that APSRA attributes the passive system failure to hardware failures (e.g., condensate return valves), whereas RMPS methodology emphasize more on the uncertainties arising from physical phenomenon such as heat loss, non-condensable fraction, oxidation, etc. In PSSs system dynamics have a huge impact on its behaviour. Both the methodologies attribute the passive system failure to the deviation of system parameters and component failures, however the two are treated separately, i.e., independent modelling of system deterministic and probabilistic aspect. However, the performance of passive systems can be significantly affected by the system dynamics especially due to the low driving force. For instance, during the course of passive system operation, a component can fail due to its stochastic nature which will in turn affect the system parameters. This change in system parameters can further influence the state of the hardware components. These complex dynamic interactions can rapidly result to a system failure and may evolve to a system state not anticipated or predicted by the deterministic assessment and classical PSA approach. To the author's knowledge, the current passive system reliability assessment methodologies do not account for these complex dynamic interactions and subsequent probabilistic dynamic system evolution. This motivates the author to develop a novel approach and provide an integrated framework to account for risk-significant scenarios arising from dynamic interactions. The integrated dynamic approach is particularly important for passive safety systems assessment that operates extensively on natural and physical laws that have a small driving force, as compared to the active systems that operates on high driving forces from external input energy. This implies that a small deviation in critical system parameters can significantly influence the system performance and hence the reliability. Given an IE, it is very likely that the state variables can make a transition out of the control space very quickly as compared to active systems. Furthermore, there exist uncertainties with regard to complex physical phenomenon such as natural circulation, thermal stratification, etc., which are not well understood or modelled till date. The proposed methodology has the potential to account for these complex interactions, capture the physical phenomenon by accounting for state variable evolution, and provide a realistic estimate of failure probability associated with passive systems.

5.3. An Integrated Framework for Dynamic Reliability Assessment of Passive Safety Systems

A novel approach for systematic modelling and integrated analysis of passive safety systems (PSSs) is outlined in this sub-section. The approach provides a framework to couple the system deterministic evolution with the system stochastic nature. The proposed approach consists of several steps and provide a roadmap for the IAEA-UOIT CRP as well as to perform a meaningful comparison between the classical PSA techniques and dynamic methodologies. The objectives of the methodology include:

- Provide a coherent approach to modelling PSSs in NPPs;
- An integrated framework to couple deterministic and probabilistic aspect of a system;
- A cohesive treatment of both aleatory and epistemic uncertainties;
- The inclusion of PSSs as a frontline system in accident sequence analysis.

Note that the methodology in many ways is explained using the passive ICS as an example so that the readers could comprehend the approach with ease. The project roadmap and the proposed methodology flowchart is shown in *Figure 26*.

1. System Identification and description

The first step of the proposed methodology involves the identification and selection of the passive safety system to be analyzed. A brief description of the system and principle of operation may be provided along with the scenarios in which the system is expected to operate.

2. Accident/transient scenario

The descriptions of scenarios (transient/accident) for which the passive safety system is designed, along with the system state during normal reactor operations. A brief operational characteristic and the system initiation following a reactor transition from normal to transient condition.

3. Definition of System Mission

The mission of a system are the objectives to be accomplished by the system under a priori defined normal or accident scenarios. Generally, the sets of objectives are predefined by the designers and can be correlated. Overall, the scenario and time at which the system is required to operate as well

as the duration of operation (mission time) must be manifested in this step. For example, the ICS maintains the RCS pressure by removing decay heat from the reactor for a period of 72 hours.

4. Success and Failure Criteria of the System

Success and failure criteria of the system must be manifested for all possible modes of operation and scenarios. The criteria may include hardware failures, state variables exceeding a pre-defined threshold limit or activation/operation of other safety systems. For example, the ICS can be considered failed, if the safety relief valves operate or if the system pressure is above 7.5 MPa. Often, these criteria are related to the system mission, and hence while defining the criteria, system mission must be taken into account.

5. Operational characteristics and parameters identification

The operational characteristics and the parameters influencing the operation of PSSs must be identified. These two factors are correlated since system parameters effect the PSSs operational behavior. The goal of this step is to comprehend the operating principle and characteristics from a qualitative view point, and not to accurately model and predict the system behavior. Of course, this will naturally involve identification of system operating parameters. For example, a natural circulation operates on coolant density difference between the heat source and sink, and the amount of heat removed can be dependent on the coolant flowrate, pressure and temperature in the sink and source side, etc. All the system parameters involved must be identified and listed in this step.

6. Identification of failure modes affecting the system performance

This step involves systematic identification of failure modes affecting the system performance. This can be achieved by the well-established and commonly used standard techniques such as failure mode and effect analysis (FMEA), and hazard and operability analysis (HAZOP). The traditional approach is mostly oriented towards hardware failure modes, however physical (virtual) phenomenon failure modes that degrade the physical mechanism must be taken into account in case of PSSs analysis due to its reliance on small driving force. For instance, performance of the ICS is significantly affected by pipe fouling, oxidations, presence of non-condensable gases, corrosion, degraded heat transfer, etc. Furthermore, the modes of failure should be prioritized with regard to the system performance.

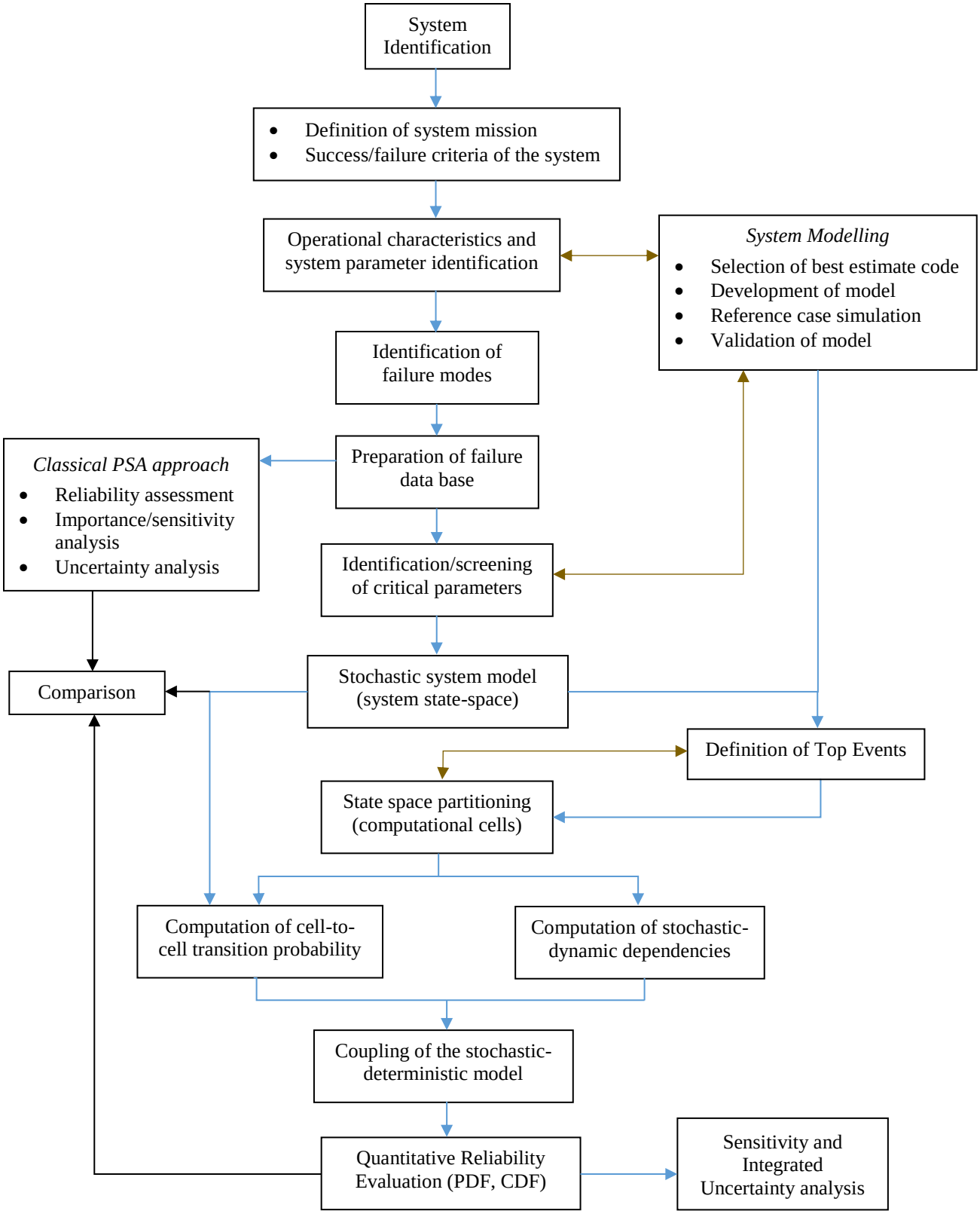


Figure 26: The project overview and methodology flowchart

7. Preparation of failure data base

A data base must be developed for the purpose of quantitative assessment. The failure data can be given in terms of probability of failure or failure rate with an exposure time, representing an unavailability of a component. Each of the failure modes identified in step 5 must be assigned with a failure probability or failure rate. Failure data may be obtained from international reliability data base such as the IAEA, USNRC and IREP.

8. Identification/screening of critical parameters affecting the system performance

This step is a continuation of step 5, in that, all the identified parameters are ranked or screened according to their importance or the level of influence on the system performance. The critical parameters are direct indicator of the system performance and include both hardware state as well as state variables. For example, the ICS operation is significantly affected by drain valve state, the presence of non-condensable gases, water temperature in the pool, differential pressure, etc. Importance analysis can be performed in the fault tree analysis to rank the critical hardware components. However, a physical model is required to rank the physical system parameters. This step enables an analyst to deduce the system modelling and analysis to a feasible and manageable state by eliminating the insignificant parameters.

9. Identification of parameters relationship and dependencies

The dependencies between critical system parameters must be identified and taken into account adequately in the process of quantification. Conditional probability can be used for the classical approach whereas dependencies arising due to dynamic interactions is treated in the process of coupling the system stochastic model with system dynamic model. The negligence of relevant dependencies can result in an inaccurate assessment of system reliability. For example, in the ICS, operation of the drain valve is dependent of the loop pressure. Loop pressure could be affected by the heat transfer rate, leading to a reduced heat transfer to the pool and hence further increasing the pressure in the loop. This could further lead to a reduced opening of the drain valve, resulting to a rapid increase of failure.

10. Reliability assessment via classical PSA techniques

Once the failure modes are identified, prioritized and failure rate data base are created, system reliability assessment (qualitative and quantitative) can be performed using the well-known

standard classical fault tree technique. The combination of failure modes resulting to a system pre-defined failure i.e., the MCSs and its subsequent quantification can be performed using well-established computer codes such as CAFTA. There can be multiple top events with respect to PSSs initiation and operation as shown in *Figure 27*. P_{SSS} implies passive safety system start-up, $\bar{P}_{SSS}$ implies PSS failure to start, P_{SSO} implies PSS operation, $\bar{P}_{SSO}$ implies failed operation, P_{SSP} implies PSS successful performance or it meet the design criteria and $\bar{P}_{SSP}$ implies passive safety in operation but failed to meet the criteria. For example, PSSs based on natural circulation in principle will not fail as long as there is a heat sink and source. However, it may not meet the success criteria such as heat transferred, coolant flowrate, etc.

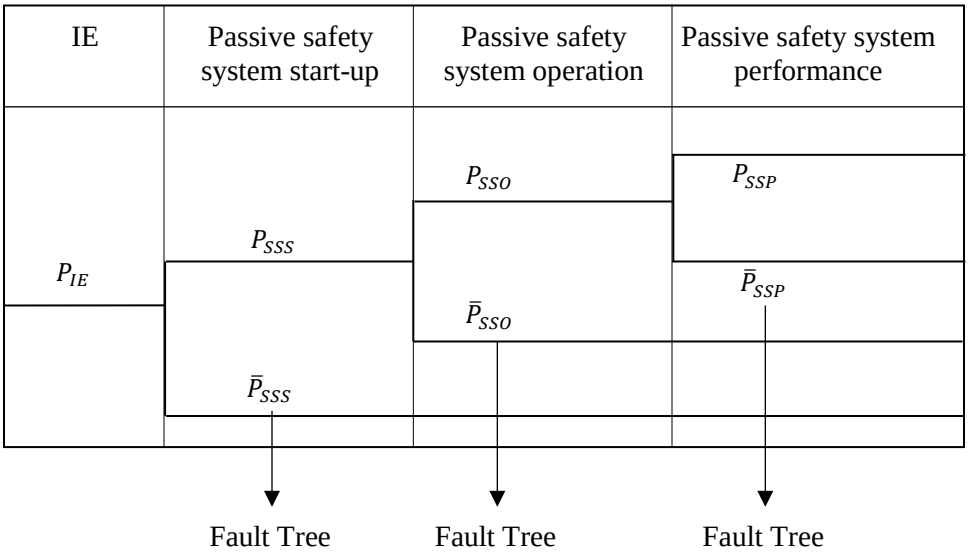


Figure 27: Representation of passive safety system in classical ET/FT

11. Stochastic modelling of the system

This step involves modelling of the system using the well-known time-dependent Markov chain. The system is described by a finite number of system states, and the probability of the system being in any of the pre-defined states is computed. Each of the system states are further characterized by the individual component states, i.e., the discretized state-space is characterized by the smallest element in the system (basic event). The ordering of states can also be performed if it effects the end state of the system. To reduce the number of system states and avoid state space explosion, the system can be characterized into macro-components or units, where the individual units can be modelled separately, e.g., using fault tree.

12. Development of system physical model

This step involves a detail development of the system physical model using qualified best estimate thermo-hydraulics system codes (e.g., RELAP5) or a simpler standalone model. Uncertainties arise in the modelling process especially due to lack of knowledge of physical phenomenon in PSSs, lack of operating experience, the input variables, approximation in system geometry, etc. This in turn effects the predicted system behavior. Uncertainties in the predicted system behavior can be reduced using or by comparing against experimental data in the modelling of physical phenomenon.

13. Model Validation: reference or design case system simulation

Once the physical model is developed, a standard reference case can be run to validate the system model. Select an initiating event that requires operation of the PSS and observe the system behavior and performance. For instance, for the ICS, a closure of MSIV or reactor scram can be selected as an initiating event that increases the system pressure and eventually demands the ICS to operate. The system model validation can also be performed if experimental data are available.

14. Define the top event of interest

In contrast to the classical approach (FT) and Markov model, the top events in dynamic methodologies are defined in terms of the state variables. For instance, for the ICS, the top event can be defined as 'RCS pressure above a defined threshold limit'. Of course, the computational complexity and time increases significantly with the increase in state variables. Again, taking the ICS as an example, the top event can be described as 'system pressure and peak cladding temperature above the threshold limit'. As an initial step and with the intention to demonstrate the capabilities of the methodology in modelling PSSs, only one state variable is considered, i.e., the RCS pressure.

15. State-space discretization: controlled state variables

The state space is discretized into finite number of disjoint computational cells covering the entire space, with the control region defined by the analyst. The control space can be identified with the knowledge of system set-points and boundaries. Information of the top event defined in step 13 can be utilized for the purpose of state-space discretization and vice versa. Step 13 and 14 are in a sense complementary to each other. The number of computational cells in the state-space and

control space is user defined. Depending on the analyst choice, the state space can be single or multi-dimensional. For example, a 3-dimensional sate space can consist of system pressure, peak cladding temperature and reactor water level. Of course, the computational complexity will increase significantly with the increase in the number of computational cells and state variables. On the other hand, too less number of cells will result in an inaccurate prediction of the system behavior. Hence, in the discretization process, the analysts should keep in mind a balance between the prediction accuracy and computational complexity have to be made.

16. Computation of cell-to-cell transition probability

The system dynamics is modelled as transition between the discretized computational cells in the state space, i.e., the probability of making a transition from one cell to the other. Here, one is interested in the evolution of the state variables in the state-space for a given system state. The discretized system state defined in step 10 is an input, whereas the state variable trajectories is the output. The number of simulation run is dependent on the number of distinct system states, which can be very large and can quickly become unmanageable. Depending on the knowledge of the system, state merging can be done to reduce the number of runs. The computation of cell-to-cell transition probabilities are explained in detail in chapter 3 and 4.

17. Stochastic-dynamic dependencies model

This step is arguably the most important part of the methodology. The stochastic time-dependent system model in step 10 is modified taking into account dynamic interactions between the system hardware and state variables. This step captures the dependencies between the two elements as well as dependencies arising from human error. For example, the system pressure will deviate based on the state of the drain valve, or the demand frequency of the drain valve can increase or decrease in a certain time interval depending on the system pressure. These dynamic interactions can affect the failure rate magnitude of a component. The term “stochastic-dynamic dependencies model” is used due to the fact that the model simultaneously takes into account the random component failures as well as dependencies arising from system dynamic evolution.

18. Coupling of the stochastic and deterministic model

The cell-to-cell and system state transition probabilities obtained from step 13 and 14 respectively are merged into a single state transition probability matrix that describes the system stochastic and

dynamic behavior. The system dynamic evolution is transformed into a Markovian model, where the probabilistic mapping is performed via the state transition matrix. A detail theoretical and numerical implementation of the coupling process is given in chapter 3.

19. Quantitative Reliability Evaluation

This step involves a systematic computation of state probabilities at any given point in time (similar to the classical discrete Markov chain). Once computation of state probabilities is obtained for each user defined time step, the probability density function (PDF) and cumulative distribution function (CDF) of a predefined top event can be determined. The statistical importance of any system states/configuration can also be computed for a given top event.

20. Assignment of probability distributions to critical parameters

In order to add credit to the point estimate predicted system behavior, critical parameters identified in Step-7 can be assigned with some probability distributions with the objective to perform an uncertainty analysis. The choice of the distributions is dependent on the state of knowledge of the parameter, availability of data and expert judgement. If there is a very limited knowledge of the system parameter, a uniform (Gaussian) distribution can be assigned i.e., all the data points within the bounded limits are equally likely. The assignment of distributions to the parameters must be done with great care since it significantly affects the predicted reliability of the PSS.

21. Integrated uncertainty evaluation

Upon assignment of probability distributions to the critical parameters which include both hardware and physical parameters, propagation of the distribution in the model can be performed using standard techniques such as direct Monte Carlo simulation. The superiority of the methodology is that, both the aleatory and epistemic uncertainties can be captured in an integrated fashion. Uncertainties arising due to lack of knowledge of physical phenomenon can be accounted in the probabilistic state variables mapping, whereas uncertainties due to the random component failure is accounted in the stochastic model.

5.4. The Benchmark Passive Isolation Condenser System

1. System identification and description

The isolation condenser system (ICS) is employed in many advanced generation and innovative reactor designs (Gen III and Gen III+), including integrated pressurized water reactor (iPWR) type small modular reactors (e.g., CAREM25, IRIS). The ICS consist of a heat exchanger, IC pool, isolation valves, drain valves, bypass drain valves and vent valves. The schematic diagram is depicted in *Figure 28*. The principle of operation is based on natural circulation.

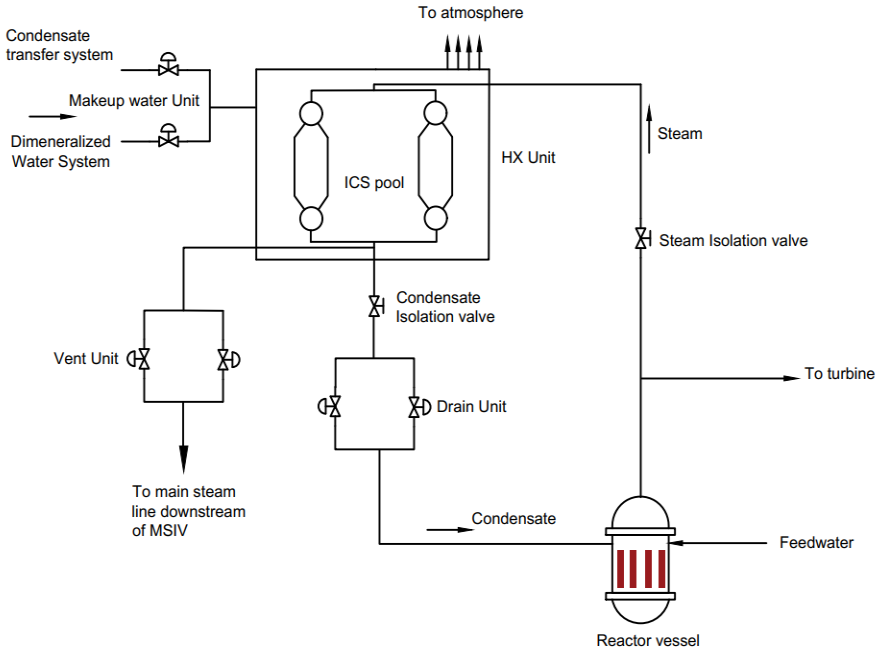


Figure 28: Passive Isolation Condenser system

The ICS operates in closed loop natural circulation mode (gravity driven) by condensing the incoming saturated steam inside the IC HX and returning the condensate back via a dedicated condensate return line to the RPV. This operational mechanism of buoyancy driven pump is created by the heat source and sink with an elevation difference between the RPV and the IC water pool. Decay heat removal from the reactor (heat source) is achieved by transferring heat to the isolation condenser (IC) water pool (heat sink) via IC heat exchangers (HX) that are immersed in the IC pool, located outside the containment and above the RPV *(See Figure 28).* Steam in tube side of the IC HX is condensed by boiling pool water in the shell side of the HX and venting the evaporated pool water to atmosphere. Due to the opening of the valve on the condensate line in order to trigger its operation, the ICS comply with the IAEA Category D passive system, which addresses the intermediary zone between active and passive, where passive execution of the safety

function is accomplished through passive means (i.e., natural circulation), but the process is initiated by active components (i.e., condensate valves actuation).

2. Scenario identification

2.A. Normal reactor operation:

During normal reactor operation the ICS is in standby mode with the steam line isolation valves in fully open position and the drain valves in closed position. The steam line isolation valves are open so that the HX tube bundles are at reactor system pressure. This allows condensate to build-up in the HX up to piping high point and filling the condensate drain line, which are maintained at a sub-cooled temperature by IC pool water. Live reactor steam is present in the steam supply line up through the horizontal distribution branch piping that feeds steam to the IC HX upper headers, *[SBWR standard safety analysis report (1992)]*. The vent and water makeup valves are also closed during normal operation, unless during maintenance and water replenishment/pool cleanup.

2.B. Transient scenario:

The ICS is a standby high-pressure system that removes residual and decay heat from the reactor pressure vessel (RPV) in the event of a reactor SCRAM in which, the reactor becomes isolated from the main condenser, or if any other high-pressure abnormal condition exists. The ICS aids in reactor vessel depressurization in the event that either the feedwater coolant injection or high-pressure coolant injection system fails. In other words, the ICS passively removes core decay heat following a reactor SCRAM from 100% full power operation and when the normal heat removal system is unavailable *[29]*. Typically, the ICS is required to remove up to 4% (approx.) of rated power which results from decay heat. Overall, ICS could be thought of as a pressure regulating system during abnormal operation, and core decay heat removal system.

The ICS is initiated by any of the following signals/action:

1. RPV pressure above a specified threshold value;
2. Main steam line isolation valve (MSIV) closure fraction;
3. RPV water level below a specified threshold value;
4. Remote manual initiation by operators.

For instance, the ICS is automatically initiated if a high reactor pressure condition is sustained for 15 seconds (generic). The time delay prevents unnecessary system initiation during turbine trips

[*29*]. Additionally, in many reactors design the ICS is automatically initiated on a low RPV water level to aid in reactor depressurization for small line breaks. These initiating signals places the ICS into operation by opening the condensate return line isolation valve, which results in draining of the accumulated condensate. Saturated steam flows from the RPV to the HX tubes via steam supply line and are condensed in the HX tubes. The condensate returns to the downcomer region of the RPV by gravity via condensate drain line. The start-up process is sufficiently fast to limit the reactor pressure rise resulting from reactor isolation to well below the pressure set-point of the SRVs for all non-accident transient isolation event *[SBWR design description]*. The natural circulation i.e., the buoyant force generated from coolant density difference dictates the coolant flowrate in the IC loop. In other words, pressure in the condensate return line region determines the coolant flowrate into the RPV. The opening of the condensate drain valve causes the liquid level in the loop to drop and to increase the available tube surface area for vapor condensation *[Khan et al. (1992)]*. This results in transfer of heat from the IC loop to the IC water pool, and the heat transfer rate is dependent on the IC pool conditions, heat transfer characteristics of the IC tubes, presence of non-condensable gases, state of the HX, etc. The extended operation of the ICS i.e., a mission time of 72 hours is accomplished by replenishing water into the IC pool through dedicated water makeup systems, thereby keeping the HX submerged in the IC water pool.

3. System mission

The primary objective/mission of the ICS includes:

1. Reactor pressure vessel depressurization;
2. Remove sensible and decay heat from the reactor for 72 hours (mission time);
3. Maintain fuel peak cladding temperature within design limits.

The above-mentioned objectives are typically achieved by employing a number of totally independent IC loops for the purpose of redundancy. First, in trying to achieve the above mission, the ICS also prevent unnecessary activation of safety relief valves (SRVs) by maintaining system pressure below SRVs set-point. This in turn eliminate or mitigate loss-of-coolant accident (LOCA) that may result from SRVs being fail-open. In other words, the ICS reduces the cycling frequency (opening and closing) of the SRVs, and hence can decrease the probability of SRVs failure on demand. Second, system depressurization under defined transient events can be achieved without the loss of primary coolant inventory. Thus, the ICS remove excess sensible and core decay heat

from the reactor in a passive way with minimal loss of coolant inventory when normal heat removal systems are unavailable. Third, operation of high pressure coolant injection system could potentially be eliminated or delay while maintaining the system pressure. Furthermore, in the event of a small LOCA, the ICS is demanded to depressurize the RCS so that high pressure injection system could be activated for coolant injection. The above three functions can be considered as secondary objectives.

4. System success/failure criteria

The ICS failure criteria includes:

1. Failure to maintain the reactor coolant system pressure below threshold value;
2. Failure to remove specified decay and residual heat produced from the reactor, i.e., decay heat removed $<$ decay heat generated;
3. Failure to maintain peak fuel cladding temperature below threshold.

It may be observed that the above criteria are interrelated. For instance, criteria 2 can cause both criteria 1 and 3 to occur, i.e., a failure to remove decay heat from the reactor will result to an increase in system pressure as well as a heat up of the fuel which can eventually lead to fuel failure. Also, a deviation in differential pressure can influence the condensate flowrate in the IC loop, which can in turn affect the amount of heat transfer rate to the IC pool. These inter-dependencies must be accounted for when developing the physical model of the system. For instance, the success/failure criteria may be given in terms of performance indicator (PI) as:

$$PI = \frac{\text{Heat removed}}{\text{Decay heat generated}} = \frac{\text{HR}}{\text{DHG}} \tag{5-1}$$

$$PI = \begin{cases} 1; & \text{Ideal success state (HR = DHG)} \\ \geq 1; & \text{Not of interest for analysis} \\ < 1; & \text{System failure or partial failure} \end{cases} \tag{5-2}$$

A typical ICS is designed to remove four (4) percent of reactor rated power, which means that five minutes after a scram and initiation of the ICS, the heat removal capacity of the IC system must equal the decay heat production rate of the shutdown reactor. This will ensure that the system pressure is below threshold limit. And note that, the reactor coolant system pressure is considered as the state variable of interest for analysis purpose.

5. The ICS components

The ICS consist of the following components: [Note that the ICS description is generic in nature and does not imply any specific design]

a. Steam supply line piping (vapor phase)

The steam supply line connects the RPV to the HX tubes, i.e., live reactor steam enters the HX tubes via steam line. This region remains under reactor coolant system pressure, and in vapor state. The steam supply line is normally open with the steam line isolation valve in fully open position during normal operation. The isolation valve is closed in case of leakage/rupture in the ICS piping or rupture in the HX tubes. This region is typically guardpiped so that a break in the line is fully contained.

b. The heat exchanger

The heat exchanger is a critical component of the ICS in which the live reactor steam is condensed inside the HX tubes. The HX consist of a number of vertical placed tubes and is emerged in the ICS water pool (shell side) creating an interface with the IC loop (tube side). The heat transfer from one loop to the other takes place in the HX.

c. Condensate drain line (liquid phase)

The condensate return line begins from the lower header of the IC HX and are normally filled with sub-cooled condensate. The condensate drain line consist of a series pair of isolation valves and a parallel pair of condensate drain valves. The pressure in this region controls the performance of the ICS during a transient event. The drain line piping ends at a dedicated condensate return nozzle that are typically located at the mid-height of the RPV.

d. Condensate drain valves

The condensate drain valves are the most critical components of the ICS. The initiation, operation and performance of the ICS is highly dependent on drain valves state. The coolant flowrate in the IC loop and hence the heat transferred is controlled by the drain valve position. The drain valves consist of the main and bypass valve that are connected in parallel. Typically, the main valve is a motor operated with fail-as-is, and the bypass valve- a nitrogen piston operated with fail-open.

e. Vent lines

The vent lines are typically installed at the lower and upper headers of the IC HXs with the main purpose to purge air and non-condensable gases from the HXs which could significantly affect and degrade the natural circulation of the coolant in the IC loop. They consist of a number of vent valves (spring and motor operated) that are normally closed. The vent lines for each of the ICS HXs are routed into the containment, and then to the suppression pool (typical advanced BWR reactor designs). These lines are provided with two main and two bypass valves located in a series of valves that are required to open at high reactor pressure during ICS operation.

f. Vent valves

During normal plant operation, air and non-condensable gases may accumulate in the ICS condenser due to hydrogen buildup from water chemistry control additions and air entrained in the feedwater. This could degrade the long-term heat removal capacity of the ICS. The purpose of the vent valves is to remove accumulated air and non-condensable gases from the ICS loop during its operation. The vent valves are installed in the vent line with a parallel configuration for redundancy diversity. The vent valves operation is controlled by automatic logics as well as by operator manual actuation. Venting is initiated whenever a combination of two signals is present:

- High RPV pressure;
- Operation/opening of the condensate drain valves.

The operating pressure of the venting unit is established below the set-point at which the lowest-set SRVs will actuate. This help ensures that the SRVs will not actuate during a reactor isolation transient even without operator intervention *[29]*.

g. Water makeup systems

The purpose of the water makeup system is to supply water to the IC water pool during extended ICS operation, and to ensure that the HX tubes remain covered at all time. A reduced water level will lead to uncover of HX tubes and thus degrading the heat transfer rate from the IC loop to the IC pool. Typically, a dedicated condensate water makeup system is in place, with the firewater system as an alternate water supply system (typical advanced BWR designs).

5.5. The ICS system characterization

The ICS hardware components is characterized into several units for stochastic modelling of the system. The units are as follow:

(b) Condensate Drain Unit (DU)
(c) Heat Exchanger Unit (HXU)
(d) Vent Unit (VU)
(e) Water Make-up Unit (MWU)
(f) Primary boundary envelope (EF)

Of course, the ICS operation is significantly affected by the hardware unit states. However, there are several physical parameters that significantly influence the ICS performance such as presence of non-condensable gases, oxidation, pipe fouling and thermal stratification. Due to the unavailability of failure data and their nature of influence on the system failure, these parameters cannot be included in the stochastic system model at this point. Hence, only the hardware system components are modelled using the Markov model. The author has proposed two approaches to modelling the overall ICS system reliability:

1. *Small Markov model and large fault tree:* The individual system units are modelled using Markov model whereas the overall system reliability is modelled using FT technique. It must be underlined that the individual unit reliability is given in terms of probability of unit failure, and hence quantification can be performed using any standard FT codes. The system reliability is given in terms of probability of failure.
2. *Large Markov model and small fault tree:* In this approach, the individual units are modelled in FT using the failure rate evaluation method. Note that the unit reliability is in terms of failure rates, rather than the failure probability as in the first approach. Once the failure rates of the individual units are obtained, the overall system reliability is modelled using time-dependent Markov model. The system reliability is given in terms of probability of failure. This approach is implemented in this thesis, since it allows for coupling the model with CCMT.

The key difference between the two approaches is that the first approach uses probability evaluation method to determine the individual unit's reliability, whereas the second approach uses failure rate evaluation for individual units. However, both methods provide the result in terms of

probability of failure. The second approach have an advantage over the first since it can yield a time-dependent model of the overall system, and the interactions among the units. In a way, these two approaches can be used as a complementary to each other and add credit to the predicted system reliability. The two (2) approaches are used to model the individual units of the ICS. The results obtained from the individual unit analysis is utilized for IC system reliability assessment. The computation of the time-dependent failure rate under a logic AND gate is determined as:

$$\lambda_S(t) = \frac{\sum_{i=1}^{n} \lambda_i(\alpha_i - 1)}{\prod_{i=1}^{n} \alpha_i - 1} \tag{5-3}$$

Where; $\alpha_i = \frac{1}{(1-e^{-(\lambda_i t)})}$

$\lambda_i =$ individual component failure rate

$n =$ number of components connected in parallel and are within the unit

$\lambda_S(t) =$ time-dependent unit failure rate

For two identical components in parallel with the same failure rate given by λ, the time dependent system failure rate is:

$$\lambda_S(t) = \frac{2\lambda}{\alpha + 1} \tag{5-4}$$

The above relations are implemented throughout this sub-section to determine the time-dependent unit failure rate, besides the Markov model that directly yields the unit failure probability.

A. Condensate Drain Unit

The condensate drain unit consist of two drain valves i.e., the main drain valve and bypass valve connected in parallel for redundancy and diversity of the unit. Each of the valve is considered to have three (3) possible states as shown in *Table 43*. It is assumed that the main and bypass valve have the same modes of failure including common cause failure (CCF) and failure rate (for simplicity).

Table 43: Failure modes and rate of the main/bypass valve

Valves	*Failure modes*	*Failure rate (per yr)*
Main/bypass drain valve	Normal	–
	Failed to remain open (λ_2^D)	8.6×10^{-09}
	Failed-to-open (λ_1^D)	1.4×10^{-06}
	Failed-to-open due to CCF ($\lambda_{1,CCF}^D$)	1.2×10^{-04}
	Failed to remain open due to CCF ($\lambda_{2,CCF}^D$)	7.2×10^{-07}

Enumerating all possible component state combination (*See Table 44*), the drain unit can have nine (9) possible states (*See Table 45*). Furthermore, state merging can be performed to reduce the overall unit states. For the drain unit, the nine (9) states are merged into two (2) unit states, i.e., normal or failure. The unit is considered failed if both the units are in any failed or partially failed state, and a unit normal/operational state if any of the valve are in normal state.

Table 44: Possible drain unit state combination

Unit state combination		*Unit States* (n)
Main valve states	*Bypass valve states*	
Normal	Normal	Normal
Normal	Fail-to-open	Normal
Normal	Failed to remain open	Normal
Fail-to-open	Normal	Normal
Fail-to-open	Fail-to-open	Failure
Fail-to-open	Failed to remain open	Failure
Failed to remain open	Normal	Normal
Failed to remain open	Fail-to-open	Failure
Failed to remain open	Failed to remain open	Failure

Table 45: Unit state ordering and the inclusion of CCF

Unit State Combination		*CCF of the valves*	*Unit States* (n)
Main valve states	*Bypass valve states*		
$n_1 = 0$	$n_2 = 0$	-	1
$n_1 = 0$	$n_2 = 1$	-	2
$n_1 = 0$	$n_2 = 2$	-	3
$n_1 = 1$	$n_2 = 0$	-	4
$n_1 = 1$	$n_2 = 1$	CCF	5
$n_1 = 1$	$n_2 = 2$	-	6
$n_1 = 2$	$n_2 = 0$	-	7
$n_1 = 2$	$n_2 = 1$	-	8
$n_1 = 2$	$n_2 = 2$	CCF	9

Note: $0 =$ component normal; $1 =$ component fail-to-open; $2 =$ component fail to remain open.

The Markov state transition diagram of the condensate drain unit is given in *Figure 29*.

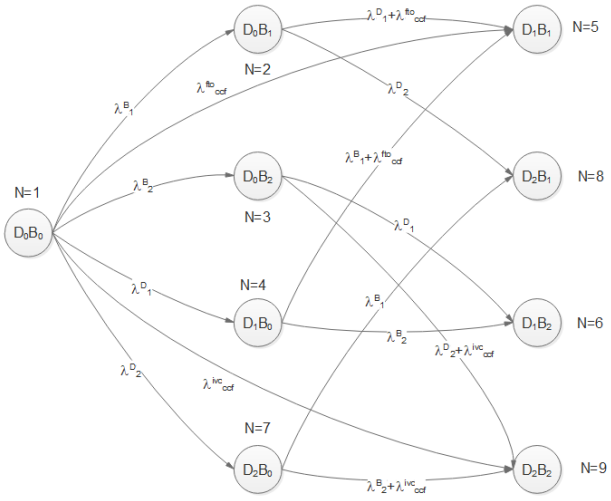


Figure 29: Markov transition diagram of the condensate drain unit

The ODEs from the above transition diagram are:

$$\frac{dP_1(t)}{dt} = -(\lambda_1^B + \lambda_2^B + \lambda_1^D + \lambda_2^D + \lambda_{CCF}^{fto} + \lambda_{CCF}^{IVC})P_1(t)$$

$$\frac{dP_2(t)}{dt} = \lambda_1^B P_1(t) - (\lambda_1^D + \lambda_2^D + \lambda_{CCF}^{fto})P_2(t)$$

$$\frac{dP_3(t)}{dt} = \lambda_2^B P_1(t) - (\lambda_1^D + \lambda_2^D + \lambda_{CCF}^{IVC})P_3(t)$$

$$\frac{dP_4(t)}{dt} = \lambda_1^D P_1(t) - (\lambda_1^B + \lambda_2^B + \lambda_{CCF}^{fto})P_4(t)$$

$$\frac{dP_7(t)}{dt} = \lambda_2^D P_1(t) - (\lambda_1^B + \lambda_2^B + \lambda_{CCF}^{IVC})P_7(t)$$

$$\frac{dP_5(t)}{dt} = \lambda_{CCF}^{fto} P_1(t) + (\lambda_1^D + \lambda_{CCF}^{fto})P_2(t) + (\lambda_1^B + \lambda_{CCF}^{fto})P_4(t)$$

$$\frac{dP_8(t)}{dt} = \lambda_2^D P_2(t) + \lambda_1^B P_7(t)$$

$$\frac{dP_6(t)}{dt} = \lambda_1^D P_3(t) + \lambda_2^B P_4(t)$$

$$\frac{dP_9(t)}{dt} = \lambda_{CCF}^{IVC} P_1(t) + (\lambda_2^B + \lambda_{CCF}^{IVC})P_7(t) + (\lambda_2^D + \lambda_{CCF}^{IVC})P_3(t)$$

(5-5)

The solution of the abo[2]ve ODEs is obtained using finite difference method implemented in Fortran95 code. Sample unit state transition probabilities for 4 time steps are shown in *Table 46.*

Table 46: Condensate drain unit state transition probabilities

Unit states	$k = 0$	$k = 1$	$k = 2$	$k = 3$
$P_1(k\Delta t)$	1.00	0.998	0.997	0.996
$P_2(k\Delta t)$	0	1.4E-05	2.797E-05	4.19E-05
$P_3(k\Delta t)$	0	8.6E-08	1.72E-07	2.58E-07
$P_4(k\Delta t)$	0	1.4E-05	2.79E-05	4.19E-05
$P_5(k\Delta t)$	0	1.2E-03	2.39E-03	3.59E-03
$P_6(k\Delta t)$	0	0	1.04E-10	3.13E-10
$P_7(k\Delta t)$	0	8.6E-08	1.72E-07	2.58E-07
$P_8(k\Delta t)$	0	0	2.41E-12	7.22E-12
$P_9(k\Delta t)$	0	7.2E-06	1.44E-05	2.16E-05
Sum	1.00	1.00	1.00	1.00

A state merging on *Table 46* gives *Table 47*.

[2] Note that the ODEs for individual units are grouped and depicted by a single equation for ease of representation

Table 47: Merged condensate drain unit state

Unit state	$k = 0$	$k = 1$	$k = 2$	$k = 3$
Unit normal	1.00	0.998	0.997	0.996
Unit failure	0	1.21E-03	2.41E-03	3.62E-03

The FT of the condensate drain unit is depicted in *Figure 30*.

Since the failure rates of main and bypass valve are identical, the unit failure rate is given as:

$$\lambda_S(t) = \frac{2\lambda}{\alpha + 1} \tag{5-6}$$

Where; $\alpha = \frac{1}{(1-e^{-(\lambda t)})}$

The top event or failure rate of the condensate drain unit is, $\lambda_{D\_Unit}(t) = 1.21 \times 10^{-04}\ per\ yr$.

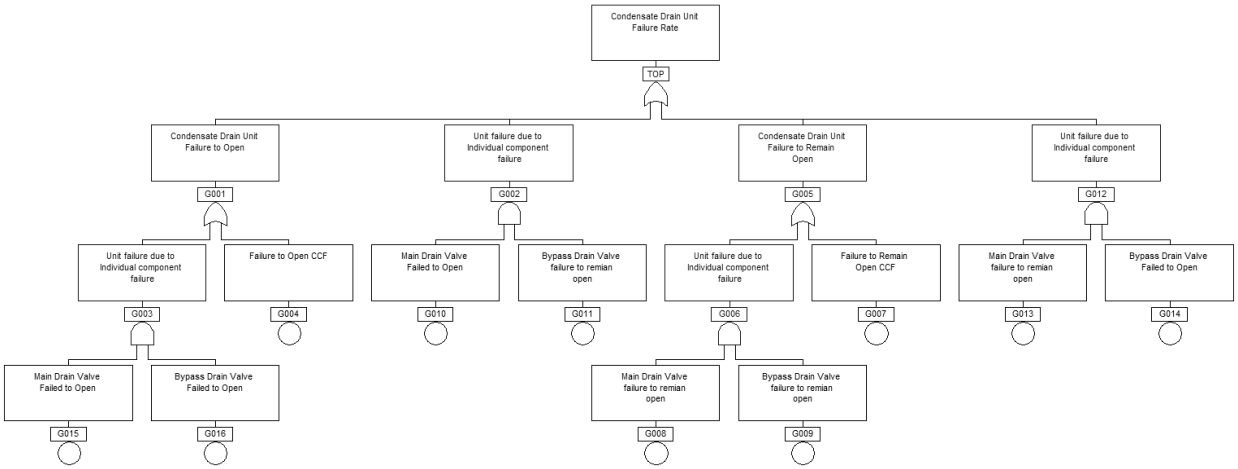


Figure 30: Fault tree of the condensate drain unit

B. Vent Unit

The vent unit consist of two vent valves connected in parallel for redundancy of the unit. Each of the valve (component level) is considered to have three (3) possible states (*See Table 48*). A common cause failure (CCF) of the valves are considered, and the vent valves are considered to have the same failure rate (for simplicity).

Table 48: Failure modes and rate of the vent valve

Component	*Failure modes*	*Failure rate (per yr)*
Vent valve	Normal	–
	Failed closed (λ_V^{fc})	1.7×10^{-08}
	Failed open (λ_V^{fo})	2.9×10^{-06}
	Failed closed due to CCF (λ_{CCF}^{fc})	7.2×10^{-07}
	Failed open due to CCF (λ_{CCF}^{fo})	1.2×10^{-04}

The overall vent unit is characterized and merged into three (3) possible states: (1) Normal; (2) Partial unit failure; and (3) Total unit failure. The set of possible states of the venting unit or component state combination is found by enumerating all the possible modes of failure of the two vent valves *(See Table 49).*

Table 49: Unit state numbering and the inclusion of CCF

Component State Combination		*Unit States*
Valve-1 state	*Valve-2 state*	(n)
$n_1 = 0$	$n_2 = 0$	1
$n_1 = 0$	$n_2 = 1$	2
$n_1 = 0$	$n_2 = 2$	3
$n_1 = 1$	$n_2 = 0$	4
$n_1 = 1$	$n_2 = 1$	5
$n_1 = 1$	$n_2 = 2$	6
$n_1 = 2$	$n_2 = 0$	7
$n_1 = 2$	$n_2 = 1$	8
$n_1 = 2$	$n_2 = 2$	9

A partial unit failure is considered for the venting unit due to the ICS performance sensitivity to the presence of non-condensable gas fraction. Partial failure implies the vent valves getting fail in any intermediate position (e.g., 50%, 60%, 40%, etc.), which provide a flexibility to observe the system response under varying system configuration. Of course, both the valves failing open or

close will result in a unit failure. The Markov transition diagram for the vent unit is given in *Figure 31*. Note that, V= vent valve 1, and B= vent valve 2.

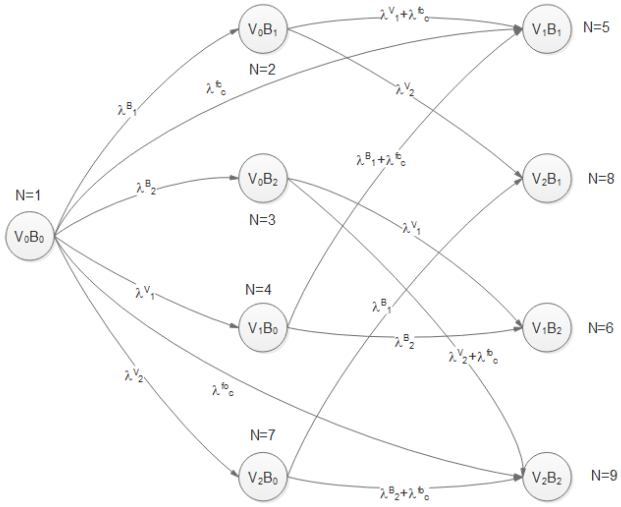


Figure 31: Markov transition diagram of the vent unit

The ODEs obtained from the state transition diagram are:

$$\frac{dP_1(t)}{dt} = -(\lambda_1^B + \lambda_2^B + \lambda_1^V + \lambda_2^V + \lambda_{CCF}^{fc} + \lambda_{CCF}^{fo})P_1(t)$$

$$\frac{dP_2(t)}{dt} = \lambda_1^B P_1(t) - (\lambda_1^V + \lambda_2^V + \lambda_{CCF}^{fc})P_2(t)$$

$$\frac{dP_3(t)}{dt} = \lambda_2^B P_1(t) - (\lambda_1^V + \lambda_2^V + \lambda_{CCF}^{fo})P_3(t)$$

$$\frac{dP_4(t)}{dt} = \lambda_1^V P_1(t) - (\lambda_1^B + \lambda_2^B + \lambda_{CCF}^{fc})P_4(t)$$

$$\frac{dP_7(t)}{dt} = \lambda_2^V P_1(t) - (\lambda_1^B + \lambda_2^B + \lambda_{CCF}^{fo})P_7(t)$$

$$\frac{dP_5(t)}{dt} = \lambda_{CCF}^{fc} P_1(t) + \left(\lambda_1^V + \lambda_{CCF}^{fc}\right)P_2(t) + (\lambda_1^B + \lambda_{CCF}^{fc})P_4(t)$$

$$\frac{dP_8(t)}{dt} = \lambda_2^V P_2(t) + \lambda_1^B P_7(t)$$

$$\frac{dP_6(t)}{dt} = \lambda_1^V P_3(t) + \lambda_2^B P_4(t)$$

$$\frac{dP_9(t)}{dt} = \lambda_{CCF}^{fo} P_1(t) + \left(\lambda_2^B + \lambda_{CCF}^{fo}\right)P_7(t) + (\lambda_2^V + \lambda_{CCF}^{fo})P_3(t)$$

(5-7)

Performing state merging on the solution of the above ODEs with normal unit state at $t = 0$ yields *Table 50*. Note only sample four (4) time steps is shown for illustration purpose.

Table 50: Vent unit states transition probabilities

Unit state	$k = 0$	$k = 1$	$k = 2$	$k = 3$
Normal	1.00	0.998	0.997	0.996
Partial failure	0	0.00	1.97E-11	5.91E-11
Total failure	0	1.21E-03	2.413E-03	3.62E-03

The FT for the vent unit using failure rate evaluation method is performed for two (2) modes, i.e., the unit total failure and partial failure. The total unit failure rate obtained from the FT is:

$$\text{Unit total failure} = \left(\lambda_V^{fc}\lambda_B^{fc} + \lambda_{CCF}^{fc}\right) + \left(\lambda_V^{fo}\lambda_B^{fo} + \lambda_{CCF}^{fo}\right) \quad (5\text{-}8)$$

$$= 1.21 \times 10^{-04}\ per\ yr$$

Whereas, the unit partial failure rate is:

$$\text{Unit partial failure} = \left(\lambda_V^{fc}\lambda_B^{fo}\right) + \left(\lambda_V^{fo}\lambda_B^{fc}\right) \quad (5\text{-}9)$$

$$\text{Unit partial failure} = 1.62 \times 10^{-15}\ per\ yr$$

C. HX Unit

The HX unit consist of a single U-tube heat exchanger immersed in a pool of water. The HX is considered to have the failure modes and rates as shown in *Table 51*.

Table 51: Failure modes and rate of the heat exchanger

Component	*Failure modes*	*Failure rate (per yr)*
Heat exchanger	Normal	–
	Single pipe rupture	2.63×10^{-06}
	Multiple pipe rupture	2.63×10^{-07}
	Single pipe plugging	2.63×10^{-06}
	Multiple pipe plugging	2.63×10^{-07}

The HX creates the interface between the two loops, and is the most critical component of the ICS, in that, heat transfer takes place from the reactor to the IC pool. A change in the HX configuration could significantly affect the ICS performance, and hence the system pressure. Investigation was performed into the possible number of states that the HX can take. Nine (9) possible states for the

HX is considered, and a qualitative outcome of the state combinations and individual failure modes was made (*See Table 52*).

Table 52: Possible states of the HX unit

HX Unit state combination	*Unit states* (n)	*System states*
Normal $(n_1 = 0)$	1	Normal
Single tube rupture $(n_1 = 1)$	2	Partial failure
Multiple tube rupture $(n_1 = 2)$	3	System failure
Single pipe plugging $(n_1 = 3)$	4	Partial failure
Multiple pipe plugging $(n_1 = 4)$	5	System failure
Single tube rupture $(n_1 = 1)$ and Single pipe plugging $(n_1 = 3)$	6	Partial failure
Single tube rupture $(n_1 = 1)$ and Multiple pipe plugging $(n_1 = 4)$	7	System failure
Multiple tube rupture $(n_1 = 2)$ and Single pipe plugging $(n_1 = 3)$	8	System failure
Multiple tube rupture $(n_1 = 2)$ and Multiple pipe plugging $(n_1 = 4)$	9	System failure

This approach enables one to reduce the number of states, and thus to avoid state space explosion phenomenon. For instance, a single tube rupture or plugging may not necessary lead to a total system failure. Hence, a partial system failure is considered, and the individual failure modes are grouped into one state assuming that the consequences of both the failure modes are similar. Merging all possible states based on their consequences, the HX unit is characterized into three (3) possible states: normal, partial failure and total unit failure.

Some of the above unit states are highly unlikely but have non-zero probabilities. For instance, multiple tube rupture and multiple pipe plugging simultaneously occurring is highly unlikely but have occurred in NPPs during the past decade. Note that an independence is considered between tube plugging and rupture, which is realistic. The Markov state transition diagram for the HX unit is given in *Figure 32*. Even though, the HX unit consist only of one component, the state transition

diagram is more complicated. This is due to the possible states as well as the possible state evolution of the HX unit. For instance, a single tube rupture can evolve into a multiple tube rupture state, or a single tube rupture along with single tube plugging can evolve into multiple tube rupture with single tube plugging.

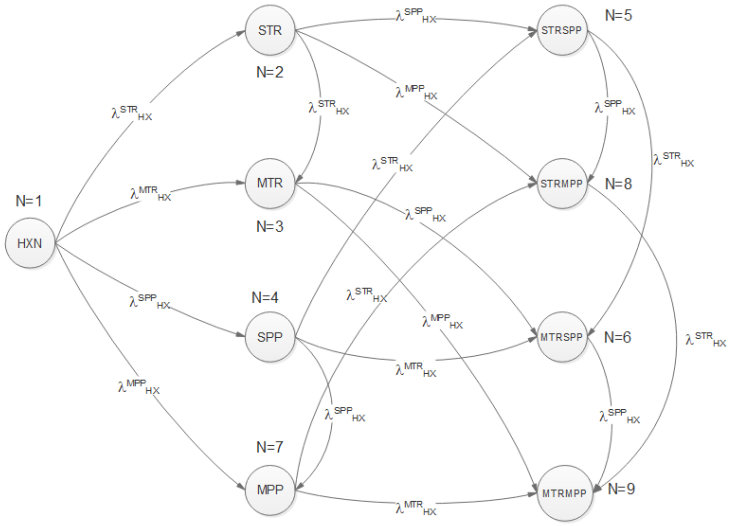


Figure 32: Markov state transition diagram for the HX unit

The resulting ODEs from the state transition diagram are:

$$\frac{dP_1(t)}{dt} = -(\lambda_{HX}^{STR} + \lambda_{HX}^{MTR} + \lambda_{HX}^{SPP} + \lambda_{HX}^{MPP})P_1(t)$$

$$\frac{dP_2(t)}{dt} = \lambda_{HX}^{STR}P_1(t) - (\lambda_{HX}^{STR} + \lambda_{HX}^{SPP} + \lambda_{HX}^{MPP})P_2(t)$$

$$\frac{dP_3(t)}{dt} = \lambda_{HX}^{MTR}P_1(t) + \lambda_{HX}^{STR}P_2(t) - (\lambda_{HX}^{SPP} + \lambda_{HX}^{MPP})P_3(t)$$

$$\frac{dP_4(t)}{dt} = \lambda_{HX}^{SPP}P_1(t) - (\lambda_{HX}^{STR} + \lambda_{HX}^{MTR} + \lambda_{HX}^{SPP})P_4(t)$$

$$\frac{dP_5(t)}{dt} = \lambda_{HX}^{SPP}P_2(t) + \lambda_{HX}^{STR}P_4(t) - (\lambda_{HX}^{STR} + \lambda_{HX}^{SPP})P_5(t)$$

$$\frac{dP_6(t)}{dt} = \lambda_{HX}^{spp}P_3(t) + \lambda_{HX}^{mtr}P_4(t) + \lambda_{HX}^{str}P_5(t) - \lambda_{HX}^{spp}P_6(t)$$

$$\frac{dP_7(t)}{dt} = \lambda_{HX}^{MPP}P_1(t) + \lambda_{HX}^{SPP}P_4(t) - (\lambda_{HX}^{STR} + \lambda_{HX}^{MTR})P_7(t)$$

$$\frac{dP_8(t)}{dt} = \lambda_{HX}^{mpp}P_2(t) + \lambda_{HX}^{spp}P_5(t) + \lambda_{HX}^{str}P_7(t) - \lambda_{HX}^{str}P_8(t)$$

$$\frac{dP_9(t)}{dt} = \lambda_{HX}^{mpp} P_3(t) + \lambda_{HX}^{spp} P_6(t) + \lambda_{HX}^{mtr} P_7(t) + \lambda_{HX}^{str} P_8(t) \quad (5\text{-}10)$$

Performing state merging on the solution of the ODEs for four (4) time steps yields *Table 53*.

Table 53: The overall HX unit state probabilities

Unit state	$k = 0$	$k = 1$	$k = 2$	$k = 3$
Normal	1.00	0.9999	0.9998	0.9998
Partial failure	0	2.89E-05	5.78E-05	8.67E-05
Total failure	0	5.26E-06	1.05E-05	1.58E-05

A failure rate evaluation for partial failure and total failure of the HX is performed to determine the time dependent failure rate that is required to construct and compute the overall ICS state probabilities. Unit state 2, 4 and 6 in *Table 52* implies a partial failure of the unit, hence the failure rate of the unit to be in partially failed state can simply be determined by.

$$\text{Partial unit failure} = \{(n_1 = 1) + (n_1 = 3) + \{(n_1 = 1)\ AND\ (n_1 = 3)\}\} = 5.26 \times 10^{-06}\ per\ yr \quad (5\text{-}11)$$

Unit states 3, 5, 7, 8 and 9 implies a total unit failure. Hence, a total unit failure probability is:

$$\text{Total unit failure} = \{(n_1 = 2) + (n_1 = 4) + \{(n_1 = 1)\ AND\ (n_1 = 4)\} + \{(n_1 = 2)\ AND\ (n_1 = 3)\} + \{(n_1 = 2)\ AND\ (n_1 = 4)\}\} = 5.26 \times 10^{-07}\ per\ yr \quad (5\text{-}12)$$

D. Water Make-up Unit

The water makeup-up unit consist of two redundant and diverse systems as follow:

- Main water make-up system (e.g., condensate)
- Alternate water make-up system (e.g., Firewater)

The two systems are treated as two components connected in parallel with each component have three possible states: (1) Normal; (2) Fail-to-open; and (3) Fail-to-remain-open (See *Table 54*). No common cause failure (CCF) of the system are considered due to the nature of diversity and independence between the two systems. The failure modes and their respective failure rate are:

Table 54: Failure modes and rate of the vent valve

Component	*Failure modes*	*Failure rate (per yr)*
Main system	Fail to open (λ_1^M)	1.2×10^{-03}
	Fail to remain open (λ_2^M)	1.0×10^{-04}
Alternate system	Fail to open (λ_1^A)	1.4×10^{-03}
	Fail to remain open (λ_2^A)	1.0×10^{-04}

The overall water make-up unit is characterized and merged into two (2) possible system states: (1) unit normal; and (2) unit failure. The set of possible water make-up unit states is found by enumerating all possible modes of failure of the main and alternate makeup system (*See Table 55*).

Table 55: Unit state combinations and ordering (notations)

Unit State Combination		*Unit States* (n)
Main valve states	*Alternate valve states*	
$n_1 = 0$	$n_2 = 0$	1
$n_1 = 0$	$n_2 = 1$	2
$n_1 = 0$	$n_2 = 2$	3
$n_1 = 1$	$n_2 = 0$	4
$n_1 = 1$	$n_2 = 1$	5
$n_1 = 1$	$n_2 = 2$	6
$n_1 = 2$	$n_2 = 0$	7
$n_1 = 2$	$n_2 = 1$	8
$n_1 = 2$	$n_2 = 2$	9

The Markov state transition diagram for the water makeup unit is depicted in *Figure 33*.

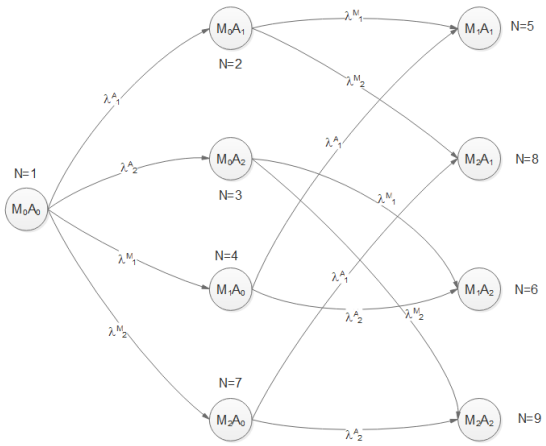


Figure 33: Markov state transition diagram of the water makeup unit

The resulting ODEs from the above state transition diagram are:

$$\frac{dP_1(t)}{dt} = -(\lambda_1^A + \lambda_2^A + \lambda_1^M + \lambda_2^M)P_1(t)$$
$$\frac{dP_2(t)}{dt} = \lambda_1^A P_1(t) - (\lambda_1^M + \lambda_2^M)P_2(t)$$
$$\frac{dP_3(t)}{dt} = \lambda_2^A P_1(t) - (\lambda_1^M + \lambda_2^M)P_3(t)$$
$$\frac{dP_4(t)}{dt} = \lambda_1^M P_1(t) - (\lambda_1^A + \lambda_2^A)P_4(t)$$
$$\frac{dP_5(t)}{dt} = \lambda_1^M P_2(t) + \lambda_1^A P_4(t)$$
$$\frac{dP_6(t)}{dt} = \lambda_1^M P_3(t) + \lambda_2^A P_4(t)$$
$$\frac{dP_7(t)}{dt} = \lambda_2^M P_1(t) - (\lambda_1^A + \lambda_2^A)P_7(t)$$
$$\frac{dP_8(t)}{dt} = \lambda_2^M P_2(t) + \lambda_1^A P_7(t)$$
$$\frac{dP_9(t)}{dt} = \lambda_2^M P_3(t) + \lambda_2^A P_7(t) \quad (5\text{-}13)$$

Analytical solution of the above ODEs is presented below. This was done to check the steady state behavior of the unit states.

$$P_2(t) = \frac{\lambda_1^A}{\lambda_1^A+\lambda_2^A} e^{-(\lambda_1^M+\lambda_2^M)t} - \frac{\lambda_1^A}{\lambda_1^A+\lambda_2^A} e^{-(\lambda_1^A+\lambda_2^A+\lambda_1^M+\lambda_2^M)t} \quad (5\text{-}14)$$

$$P_5(t) = \frac{\lambda_1^A\lambda_1^M}{\lambda_1^A\lambda_1^M + \lambda_1^A\lambda_2^M + \lambda_2^A\lambda_1^M + \lambda_2^A\lambda_2^M} + \frac{\lambda_1^A\lambda_1^M}{\lambda_1^A\lambda_1^M + \lambda_1^A\lambda_2^M + \lambda_2^A\lambda_1^M + \lambda_2^A\lambda_2^M} e^{-(\lambda_1^A+\lambda_2^A+\lambda_1^M+\lambda_2^M)t}$$
$$- \frac{\lambda_1^A\lambda_1^M}{\lambda_1^A\lambda_1^M+\lambda_1^A\lambda_2^M+\lambda_2^A\lambda_1^M+\lambda_2^A\lambda_2^M} e^{-(\lambda_1^A+\lambda_2^A)t} - \frac{\lambda_1^A\lambda_1^M}{\lambda_1^A\lambda_1^M+\lambda_1^A\lambda_2^M+\lambda_2^A\lambda_1^M+\lambda_2^A\lambda_2^M} e^{-(\lambda_1^M+\lambda_2^M)t} \quad (5\text{-}15)$$

$$P_9(t) = \frac{\lambda_2^A \lambda_2^M}{\lambda_1^A \lambda_1^M + \lambda_1^A \lambda_2^M + \lambda_2^A \lambda_1^M + \lambda_2^A \lambda_2^M} + \frac{\lambda_2^A \lambda_2^M}{\lambda_1^A \lambda_1^M + \lambda_1^A \lambda_2^M + \lambda_2^A \lambda_1^M + \lambda_2^A \lambda_2^M} e^{-(\lambda_1^A + \lambda_2^A + \lambda_1^M + \lambda_2^M)t}$$

$$- \frac{\lambda_2^A \lambda_2^M}{\lambda_1^A \lambda_1^M + \lambda_1^A \lambda_2^M + \lambda_2^A \lambda_1^M + \lambda_2^A \lambda_2^M} e^{-(\lambda_1^A + \lambda_2^A)t} - \frac{\lambda_2^A \lambda_2^M}{\lambda_1^A \lambda_1^M + \lambda_1^A \lambda_2^M + \lambda_2^A \lambda_1^M + \lambda_2^A \lambda_2^M} e^{-(\lambda_1^M + \lambda_2^M)t} \quad (5\text{-}16)$$

For analyzing the steady state behavior, take the case for $P_2(t)$ as $t \to \infty$.

$$P_2(t) = \frac{\lambda_1^A}{\lambda_1^A + \lambda_2^A} e^{-\infty} - \frac{\lambda_1^A}{\lambda_1^A + \lambda_2^A} e^{-\infty} \quad (5\text{-}17)$$

$$P_2(t) = 0$$

Again, take the case for $P_5(t)$ as $t \to \infty$.

$$P_5(t) = \frac{\lambda_1^A \lambda_1^M}{\lambda_1^A \lambda_1^M + \lambda_1^A \lambda_2^M + \lambda_2^A \lambda_1^M + \lambda_2^A \lambda_2^M} \quad (5\text{-}18)$$

$$P_5(t) = 0.862 \textit{ (steady state value)}$$

A state merging on the solution of the ODEs yields two unit state transition probabilities. Sample state probabilities for four (4) time steps are given in *Table 56*.

Table 56: Overall makeup water unit states

Unit state	$k = 0$	$k = 1$	$k = 2$	$k = 3$
Normal	1.00	1.0E+00	0.999	0.998
Unit failure	0	0	3.90E-04	1.15E-03

The failure rate evaluation from the FT of the water makeup unit can simply be given by:

$$\text{Unit failure rate} = \lambda_1^M \lambda_1^A + \lambda_2^M \lambda_1^A + \lambda_1^M \lambda_2^A + \lambda_2^M \lambda_2^A \quad (5\text{-}19)$$

$$= 3.21 \times 10^{-08} \; per \; yr$$

E. Envelope failure (EF) or pipe break/rupture

Envelope failure implies a degraded state of the primary IC loop. A failure of IC loop will result in a loss of coolant inventory from the RCS. Envelope failure can be categorized based on the break size and location, however for modelling simplicity, only two state are considered, i.e., normal or failure. Failure rate of the envelope is $\lambda_{EF} = 2.74 \times 10^{-04} \; per \; yr$. Therefore, unit states probability via Markov model is:

$$P_0(t) = e^{-\lambda_{EF} \cdot t} \text{ and } P_1(t) = (1 - e^{-\lambda_{EF} \cdot t}) \quad (5\text{-}20)$$

The probability of EF being in a failed or pipe break state can easily be found using the solution for $P_1(t)$ at any point in time. Note that the mission time considered for the analysis is 72 hours.

5.6. Markov Model of the Benchmark Passive ICS

This sub-section integrates the results to predict the overall ICS reliability. The units are modelled as a single entity or a macro-component thus merging the system states. This significantly reduces the computation time and complexity, without compromising the model and prediction accuracy. The number of possible states that the ICS can take any point in space is first determined from the number of unit possible states. From the previous sections, the individual units have the following number of states:

1. Vent unit: 3 states
2. Makeup water unit: 2 states
3. Drain unit: 2 states
4. System envelope: 2 states
5. HX unit: 3 states

Therefore, the number of possible states that the ICS can take is $5 \times 12 = 72$ discrete system states (*See Table 58*). For ease of identification and analysis purpose, the individual unit's notations and numbering is categorized as shown in *Table 57*.

Table 57: Individual unit notation and numbering

Units	*Unit number*
Condensate Drain Unit	Unit-1
Vent Unit	Unit-2
HX Unit	Unit-3
Makeup Water Unit	Unit-4
Envelope Failure	Unit-5

Table 58: Possible individual unit state combinations

Individual Unit State Combination					*System States* (n)	
Drain unit (n_1)	*Vent unit* (n_2)	*HX unit* (n_3)	*MK unit* (n_4)	*EF* (n_5)		
$n_1 = 0$	$n_2 = 0$	$n_3 = 0$	$n_4 = 0$	$n_5 = 0$	0	$P_0(t)$
$n_1 = 0$	$n_2 = 0$	$n_3 = 0$	$n_4 = 0$	$n_5 = 1$	1	$P_1(t)$
$n_1 = 0$	$n_2 = 0$	$n_3 = 0$	$n_4 = 1$	$n_5 = 0$	2	$P_2(t)$
$n_1 = 0$	$n_2 = 0$	$n_3 = 0$	$n_4 = 1$	$n_5 = 1$	3	$P_3(t)$
$n_1 = 0$	$n_2 = 0$	$n_3 = 1$	$n_4 = 0$	$n_5 = 0$	4	$P_4(t)$
$n_1 = 0$	$n_2 = 0$	$n_3 = 1$	$n_4 = 0$	$n_5 = 1$	5	$P_5(t)$
$n_1 = 0$	$n_2 = 0$	$n_3 = 1$	$n_4 = 1$	$n_5 = 0$	6	$P_6(t)$
$n_1 = 0$	$n_2 = 0$	$n_3 = 1$	$n_4 = 1$	$n_5 = 1$	7	$P_7(t)$
$n_1 = 0$	$n_2 = 0$	$n_3 = 2$	$n_4 = 0$	$n_5 = 0$	8	$P_8(t)$
$n_1 = 0$	$n_2 = 0$	$n_3 = 2$	$n_4 = 0$	$n_5 = 1$	9	$P_9(t)$
$n_1 = 0$	$n_2 = 0$	$n_3 = 2$	$n_4 = 1$	$n_5 = 0$	10	$P_{10}(t)$
$n_1 = 0$	$n_2 = 0$	$n_3 = 2$	$n_4 = 1$	$n_5 = 1$	11	$P_{11}(t)$
$n_1 = 0$	$n_2 = 1$	$n_3 = 0$	$n_4 = 0$	$n_5 = 0$	12	$P_{12}(t)$
$n_1 = 0$	$n_2 = 1$	$n_3 = 0$	$n_4 = 0$	$n_5 = 1$	13	$P_{13}(t)$
$n_1 = 0$	$n_2 = 1$	$n_3 = 0$	$n_4 = 1$	$n_5 = 0$	14	$P_{14}(t)$
$n_1 = 0$	$n_2 = 1$	$n_3 = 0$	$n_4 = 1$	$n_5 = 1$	15	$P_{15}(t)$
$n_1 = 0$	$n_2 = 1$	$n_3 = 1$	$n_4 = 0$	$n_5 = 0$	16	$P_{16}(t)$
$n_1 = 0$	$n_2 = 1$	$n_3 = 1$	$n_4 = 0$	$n_5 = 1$	17	$P_{17}(t)$
$n_1 = 0$	$n_2 = 1$	$n_3 = 1$	$n_4 = 1$	$n_5 = 0$	18	$P_{18}(t)$
$n_1 = 0$	$n_2 = 1$	$n_3 = 1$	$n_4 = 1$	$n_5 = 1$	19	$P_{19}(t)$
$n_1 = 0$	$n_2 = 1$	$n_3 = 2$	$n_4 = 0$	$n_5 = 0$	20	$P_{20}(t)$
$n_1 = 0$	$n_2 = 1$	$n_3 = 2$	$n_4 = 0$	$n_5 = 1$	21	$P_{21}(t)$
$n_1 = 0$	$n_2 = 1$	$n_3 = 2$	$n_4 = 1$	$n_5 = 0$	22	$P_{22}(t)$
$n_1 = 0$	$n_2 = 1$	$n_3 = 2$	$n_4 = 1$	$n_5 = 1$	23	$P_{23}(t)$
$n_1 = 0$	$n_2 = 2$	$n_3 = 0$	$n_4 = 0$	$n_5 = 0$	24	$P_{24}(t)$
$n_1 = 0$	$n_2 = 2$	$n_3 = 0$	$n_4 = 0$	$n_5 = 1$	25	$P_{25}(t)$

$n_1 = 0$	$n_2 = 2$	$n_3 = 0$	$n_4 = 1$	$n_5 = 0$	26	$P_{26}(t)$
$n_1 = 0$	$n_2 = 2$	$n_3 = 0$	$n_4 = 1$	$n_5 = 1$	27	$P_{27}(t)$
$n_1 = 0$	$n_2 = 2$	$n_3 = 1$	$n_4 = 0$	$n_5 = 0$	28	$P_{28}(t)$
$n_1 = 0$	$n_2 = 2$	$n_3 = 1$	$n_4 = 0$	$n_5 = 1$	29	$P_{29}(t)$
$n_1 = 0$	$n_2 = 2$	$n_3 = 1$	$n_4 = 1$	$n_5 = 0$	30	$P_{30}(t)$
$n_1 = 0$	$n_2 = 2$	$n_3 = 1$	$n_4 = 1$	$n_5 = 1$	31	$P_{31}(t)$
$n_1 = 0$	$n_2 = 2$	$n_3 = 2$	$n_4 = 0$	$n_5 = 0$	32	$P_{32}(t)$
$n_1 = 0$	$n_2 = 2$	$n_3 = 2$	$n_4 = 0$	$n_5 = 1$	33	$P_{33}(t)$
$n_1 = 0$	$n_2 = 2$	$n_3 = 2$	$n_4 = 1$	$n_5 = 0$	34	$P_{34}(t)$
$n_1 = 0$	$n_2 = 2$	$n_3 = 2$	$n_4 = 1$	$n_5 = 1$	35	$P_{35}(t)$
$n_1 = 1$	$n_2 = 0$	$n_3 = 0$	$n_4 = 0$	$n_5 = 0$	36	$P_{36}(t)$
$n_1 = 1$	$n_2 = 0$	$n_3 = 0$	$n_4 = 0$	$n_5 = 1$	37	$P_{37}(t)$
$n_1 = 1$	$n_2 = 0$	$n_3 = 0$	$n_4 = 1$	$n_5 = 0$	38	$P_{38}(t)$
$n_1 = 1$	$n_2 = 0$	$n_3 = 0$	$n_4 = 1$	$n_5 = 1$	39	$P_{39}(t)$
$n_1 = 1$	$n_2 = 0$	$n_3 = 1$	$n_4 = 0$	$n_5 = 0$	40	$P_{40}(t)$
$n_1 = 1$	$n_2 = 0$	$n_3 = 1$	$n_4 = 0$	$n_5 = 1$	41	$P_{41}(t)$
$n_1 = 1$	$n_2 = 0$	$n_3 = 1$	$n_4 = 1$	$n_5 = 0$	42	$P_{42}(t)$
$n_1 = 1$	$n_2 = 0$	$n_3 = 1$	$n_4 = 1$	$n_5 = 1$	43	$P_{43}(t)$
$n_1 = 1$	$n_2 = 0$	$n_3 = 2$	$n_4 = 0$	$n_5 = 0$	44	$P_{44}(t)$
$n_1 = 1$	$n_2 = 0$	$n_3 = 2$	$n_4 = 0$	$n_5 = 1$	45	$P_{45}(t)$
$n_1 = 1$	$n_2 = 0$	$n_3 = 2$	$n_4 = 1$	$n_5 = 0$	46	$P_{46}(t)$
$n_1 = 1$	$n_2 = 0$	$n_3 = 2$	$n_4 = 1$	$n_5 = 1$	47	$P_{47}(t)$
$n_1 = 1$	$n_2 = 1$	$n_3 = 0$	$n_4 = 0$	$n_5 = 0$	48	$P_{48}(t)$
$n_1 = 1$	$n_2 = 1$	$n_3 = 0$	$n_4 = 0$	$n_5 = 1$	49	$P_{49}(t)$
$n_1 = 1$	$n_2 = 1$	$n_3 = 0$	$n_4 = 1$	$n_5 = 0$	50	$P_{50}(t)$
$n_1 = 1$	$n_2 = 1$	$n_3 = 0$	$n_4 = 1$	$n_5 = 1$	51	$P_{51}(t)$
$n_1 = 1$	$n_2 = 1$	$n_3 = 1$	$n_4 = 0$	$n_5 = 0$	52	$P_{52}(t)$
$n_1 = 1$	$n_2 = 1$	$n_3 = 1$	$n_4 = 0$	$n_5 = 1$	53	$P_{53}(t)$
$n_1 = 1$	$n_2 = 1$	$n_3 = 1$	$n_4 = 1$	$n_5 = 0$	54	$P_{54}(t)$

$n_1 = 1$	$n_2 = 1$	$n_3 = 1$	$n_4 = 1$	$n_5 = 1$	55	$P_{55}(t)$
$n_1 = 1$	$n_2 = 1$	$n_3 = 2$	$n_4 = 0$	$n_5 = 0$	56	$P_{56}(t)$
$n_1 = 1$	$n_2 = 1$	$n_3 = 2$	$n_4 = 0$	$n_5 = 1$	57	$P_{57}(t)$
$n_1 = 1$	$n_2 = 1$	$n_3 = 2$	$n_4 = 1$	$n_5 = 0$	58	$P_{58}(t)$
$n_1 = 1$	$n_2 = 1$	$n_3 = 2$	$n_4 = 1$	$n_5 = 1$	59	$P_{59}(t)$
$n_1 = 1$	$n_2 = 2$	$n_3 = 0$	$n_4 = 0$	$n_5 = 0$	60	$P_{60}(t)$
$n_1 = 1$	$n_2 = 2$	$n_3 = 0$	$n_4 = 0$	$n_5 = 1$	61	$P_{61}(t)$
$n_1 = 1$	$n_2 = 2$	$n_3 = 0$	$n_4 = 1$	$n_5 = 0$	62	$P_{62}(t)$
$n_1 = 1$	$n_2 = 2$	$n_3 = 0$	$n_4 = 1$	$n_5 = 1$	63	$P_{63}(t)$
$n_1 = 1$	$n_2 = 2$	$n_3 = 1$	$n_4 = 0$	$n_5 = 0$	64	$P_{64}(t)$
$n_1 = 1$	$n_2 = 2$	$n_3 = 1$	$n_4 = 0$	$n_5 = 1$	65	$P_{65}(t)$
$n_1 = 1$	$n_2 = 2$	$n_3 = 1$	$n_4 = 1$	$n_5 = 0$	66	$P_{66}(t)$
$n_1 = 1$	$n_2 = 2$	$n_3 = 1$	$n_4 = 1$	$n_5 = 1$	67	$P_{67}(t)$
$n_1 = 1$	$n_2 = 2$	$n_3 = 2$	$n_4 = 0$	$n_5 = 0$	68	$P_{68}(t)$
$n_1 = 1$	$n_2 = 2$	$n_3 = 2$	$n_4 = 0$	$n_5 = 1$	69	$P_{69}(t)$
$n_1 = 1$	$n_2 = 2$	$n_3 = 2$	$n_4 = 1$	$n_5 = 0$	70	$P_{70}(t)$
$n_1 = 1$	$n_2 = 2$	$n_3 = 2$	$n_4 = 1$	$n_5 = 1$	71	$P_{71}(t)$

Constructing a Markov state transition diagram become quite complex, challenging and unmanageable due to the large number of system states. However, a systematic treatment of the above enumerated system states can be performed using simple algorithm and implementing it in tools such as MATLAB. The system states are organized systematically in layer fashion based on the number of components failure (See *Table 59* to *Table 64*). For instance, Layer 0 implies a normal system states or zero (0) unit failure, Layer 1 implies a single (1) unit failure, Layer 2 implies a two (2) unit failure, and so on.

Table 59: Systematic organization of system states in Layer fashion

Individual Unit State Combination					*System States* (n)	
Drain unit (n_1)	*Vent unit* (n_2)	*HX unit* (n_3)	*MK unit* (n_4)	*EF* (n_5)		
Layer 0 (Number of states with no failure= 1)						
$n_1 = 0$	$n_2 = 0$	$n_3 = 0$	$n_4 = 0$	$n_5 = 0$	0	$P_0(t)$
Layer 1 (Number of states with one= 7)						
$n_1 = 1$	$n_2 = 0$	$n_3 = 0$	$n_4 = 0$	$n_5 = 0$	36	$P_{36}(t)$
$n_1 = 0$	$n_2 = 1$	$n_3 = 0$	$n_4 = 0$	$n_5 = 0$	12	$P_{12}(t)$
⋮	⋮	⋮	⋮	⋮	⋮	⋮
Layer 2 (Number of states with two failures= 19)						
$n_1 = 1$	$n_2 = 1$	$n_3 = 0$	$n_4 = 0$	$n_5 = 0$	48	$P_{48}(t)$
$n_1 = 1$	$n_2 = 2$	$n_3 = 0$	$n_4 = 0$	$n_5 = 0$	60	$P_{60}(t)$
⋮	⋮	⋮	⋮	⋮	⋮	⋮
Layer 3 (Number of states with three failures= 25)						
$n_1 = 1$	$n_2 = 1$	$n_3 = 1$	$n_4 = 0$	$n_5 = 0$	52	$P_{52}(t)$
$n_1 = 1$	$n_2 = 1$	$n_3 = 2$	$n_4 = 0$	$n_5 = 0$	56	$P_{56}(t)$
⋮	⋮	⋮	⋮	⋮	⋮	⋮
Layer 4 (Number of states with 4 failures= 16)						
$n_1 = 1$	$n_2 = 1$	$n_3 = 1$	$n_4 = 1$	$n_5 = 0$	54	$P_{54}(t)$
$n_1 = 1$	$n_2 = 1$	$n_3 = 1$	$n_4 = 0$	$n_5 = 1$	53	$P_{53}(t)$
⋮	⋮	⋮	⋮	⋮	⋮	⋮
Layer 5 (Number of states with five failures= 4)						
$n_1 = 1$	$n_2 = 1$	$n_3 = 1$	$n_4 = 1$	$n_5 = 1$	55	$P_{55}(t)$
⋮	⋮	⋮	⋮	⋮	⋮	⋮

The possible number of transition from one layer to other can be determined systematically. Of course, since no repair of the units are considered, the layer can only progress in an increasing

manner. For instance, the system can make a transition from layer 2 to layer 3, but not vice versa. The number of transition from each layer with their respective failure rates are given in *Table 60*.

Table 60: Transition from Layer 0 to Layer 1

From	*Transition rate*	*To*	*No.*
$n = 0$: $\{(n_1 = 0)(n_2 = 0)(n_3 = 0)(n_4 = 0)(n_5 = 0)\}$	λ_1^{DU}	$n = 36$	1
	λ_1^{VU}	$n = 12$	2
	λ_2^{VU}	$n = 24$	3
	λ_1^{HXU}	$n = 4$	4
	λ_2^{HXU}	$n = 8$	5
	λ_1^{MWU}	$n = 2$	6
	λ_1^{EF}	$n = 1$	7

Table 61: Transition from Layer 1 to Layer 2

From	*Transition rate*	*To*	*No.*
$n = 36$: $\{(n_1 = 1)\ (n_2 = 0)\ (n_3 = 0)\ (n_4 = 0)\ (n_5 = 0)\}$	λ_1^{VU}	$n = 48$	1
	λ_2^{VU}	$n = 60$	2
	λ_1^{HXU}	$n = 40$	3
	λ_2^{HXU}	$n = 44$	4
	λ_1^{MWU}	$n = 38$	5
	λ_1^{EF}	$n = 37$	6
⋮	⋮	⋮	⋮

Table 62: Transition from Layer 2 to Layer 3

From	*Transition rate*	*To*	*No.*
$n = 48$: $\{(n_1 = 1)\ (n_2 = 1)\ (n_3 = 0)\ (n_4 = 0)\ (n_5 = 0)\}$	λ_1^{HXU}	$n = 52$	1
	λ_2^{HXU}	$n = 56$	2
	λ_1^{MWU}	$n = 50$	3
	λ_1^{EF}	$n = 49$	4
⋮	⋮	⋮	⋮

Table 63: Transition from Layer 3 to Layer 4

From	*Transition rate*	*To*	*No.*
$n = 52$: $\{(n_1 = 1)\ (n_2 = 1)\ (n_3 = 1)\ (n_4 = 0)\ (n_5 = 0)\}$	λ_1^{MWU}	$n = 54$	1
	λ_1^{EF}	$n = 53$	2
⋮	⋮	⋮	⋮

Table 64: Transition from Layer 4 to Layer 5

From	*Transition rate*	*To*	*No.*
$n = 54$: $\{(n_1 = 1).\ (n_2 = 1).\ (n_3 = 1)\ (n_4 = 1).(n_5 = 0)\}$	λ_1^{EF}	$n = 55$	1
⋮	⋮	⋮	⋮

The resulting ODEs from the state transition diagram is presented below in a layer fashion.

Layer 0:

$$\frac{dP_0(t)}{dt} = -(\lambda_1^{DU} + \lambda_1^{VU} + \lambda_2^{VU} + \lambda_1^{HXU} + \lambda_2^{HXU} + \lambda_1^{MWU} + \lambda_1^{EF})P_0(t) \tag{5-21}$$

Layer 1:

$$\frac{dP_1(t)}{dt} = \lambda_1^{EF}P_0(t) - (\lambda_1^{MWU} + \lambda_1^{DU} + \lambda_1^{VU} + \lambda_2^{VU} + \lambda_1^{HXU} + \lambda_2^{HXU})P_1(t)$$

$$\frac{dP_2(t)}{dt} = \lambda_1^{MWU}P_0(t) - (\lambda_1^{EF} + \lambda_1^{DU} + \lambda_1^{VU} + \lambda_2^{VU} + \lambda_1^{HXU} + \lambda_2^{HXU})P_2(t)$$

$$\frac{dP_{36}(t)}{dt} = \lambda_1^{DU}P_0(t) - (\lambda_1^{EF} + \lambda_1^{MWU} + \lambda_1^{VU} + \lambda_2^{VU} + \lambda_1^{HXU} + \lambda_2^{HXU})P_{36}(t)$$

$$\frac{dP_{12}(t)}{dt} = \lambda_1^{VU}P_0(t) - (\lambda_1^{EF} + \lambda_1^{MWU} + \lambda_1^{DU} + \lambda_1^{HXU} + \lambda_2^{HXU})P_{12}(t)$$

$$\frac{dP_{24}(t)}{dt} = \lambda_2^{VU}P_0(t) - (\lambda_1^{EF} + \lambda_1^{MWU} + \lambda_1^{DU} + \lambda_1^{HXU} + \lambda_2^{HXU})P_{24}(t)$$

$$\frac{dP_4(t)}{dt} = \lambda_1^{HXU}P_0(t) - (\lambda_1^{EF} + \lambda_1^{MWU} + \lambda_1^{DU} + \lambda_1^{VU} + \lambda_2^{VU})P_4(t)$$

$$\frac{dP_8(t)}{dt} = \lambda_2^{HXU}P_0(t) - (\lambda_1^{EF} + \lambda_1^{MWU} + \lambda_1^{DU} + \lambda_1^{VU} + \lambda_2^{VU})P_8(t)$$

(5-22)

Layer 2:

$$\frac{dP_{48}(t)}{dt} = \lambda_1^{VU}P_{36}(t) + \lambda_1^{DU}P_{12}(t) - (\lambda_1^{HXU} + \lambda_2^{HXU} + \lambda_1^{EF} + \lambda_1^{MWU})P_{48}(t)$$

$$\frac{dP_{60}(t)}{dt} = \lambda_2^{VU} P_{36}(t) + \lambda_1^{DU} P_{24}(t) - (\lambda_1^{HXU} + \lambda_2^{HXU} + \lambda_1^{EF} + \lambda_1^{MWU}) P_{60}(t)$$

$$\frac{dP_{40}(t)}{dt} = \lambda_1^{HXU} P_{36}(t) + \lambda_1^{DU} P_4(t) - (\lambda_1^{VU} + \lambda_2^{VU} + \lambda_1^{EF} + \lambda_1^{MWU}) P_{40}(t)$$

$$\frac{dP_{44}(t)}{dt} = \lambda_2^{HXU} P_{36}(t) + \lambda_1^{DU} P_8(t) - (\lambda_1^{VU} + \lambda_2^{VU} + \lambda_1^{EF} + \lambda_1^{MWU}) P_{44}(t)$$

$$\frac{dP_{38}(t)}{dt} = \lambda_1^{MWU} P_{36}(t) + \lambda_1^{DU} P_2(t) - (\lambda_1^{VU} + \lambda_2^{VU} + \lambda_1^{EF} + \lambda_1^{HXU} + \lambda_2^{HXU}) P_{38}(t)$$

$$\frac{dP_{37}(t)}{dt} = \lambda_1^{EF} P_{36}(t) + \lambda_1^{DU} P_1(t) - (\lambda_1^{VU} + \lambda_2^{VU} + \lambda_1^{MWU} + \lambda_1^{HXU} + \lambda_2^{HXU}) P_{37}(t)$$

$$\frac{dP_{16}(t)}{dt} = \lambda_1^{HXU} P_{12}(t) + \lambda_1^{VU} P_4(t) - (\lambda_1^{DU} + \lambda_1^{MWU} + \lambda_1^{EF}) P_{16}(t)$$

$$\frac{dP_{20}(t)}{dt} = \lambda_2^{HXU} P_{12}(t) + \lambda_1^{VU} P_8(t) - (\lambda_1^{DU} + \lambda_1^{MWU} + \lambda_1^{EF}) P_{20}(t)$$

$$\frac{dP_{14}(t)}{dt} = \lambda_1^{MWU} P_{12}(t) + \lambda_1^{VU} P_2(t) - (\lambda_1^{DU} + \lambda_1^{EF} + \lambda_1^{HXU} + \lambda_2^{HXU}) P_{14}(t)$$

$$\frac{dP_{13}(t)}{dt} = \lambda_1^{EF} P_{12}(t) + \lambda_1^{VU} P_1(t) - (\lambda_1^{DU} + \lambda_1^{MWU} + \lambda_1^{HXU} + \lambda_2^{HXU}) P_{13}(t)$$

$$\frac{dP_{28}(t)}{dt} = \lambda_1^{HXU} P_{24}(t) + \lambda_2^{VU} P_4(t) - (\lambda_1^{DU} + \lambda_1^{MWU} + \lambda_1^{EF}) P_{28}(t)$$

$$\frac{dP_{32}(t)}{dt} = \lambda_2^{HXU} P_{24}(t) + \lambda_2^{VU} P_8(t) - (\lambda_1^{DU} + \lambda_1^{MWU} + \lambda_1^{EF}) P_{32}(t)$$

$$\frac{dP_{26}(t)}{dt} = \lambda_1^{MWU} P_{24}(t) + \lambda_2^{VU} P_2(t) - (\lambda_1^{DU} + \lambda_1^{EF} + \lambda_1^{HXU} + \lambda_2^{HXU}) P_{26}(t)$$

$$\frac{dP_{25}(t)}{dt} = \lambda_1^{EF} P_{24}(t) + \lambda_2^{VU} P_1(t) - (\lambda_1^{DU} + \lambda_1^{MWU} + \lambda_1^{HXU} + \lambda_2^{HXU}) P_{25}(t)$$

$$\frac{dP_6(t)}{dt} = \lambda_1^{HXU} P_2(t) + \lambda_1^{MWU} P_4(t) - (\lambda_1^{DU} + \lambda_1^{EF} + \lambda_1^{VU} + \lambda_2^{VU}) P_6(t)$$

$$\frac{dP_5(t)}{dt} = \lambda_1^{HXU} P_1(t) + \lambda_1^{EF} P_4(t) - (\lambda_1^{DU} + \lambda_1^{MWU} + \lambda_1^{VU} + \lambda_2^{VU}) P_5(t)$$

$$\frac{dP_{10}(t)}{dt} = \lambda_2^{HXU} P_2(t) + \lambda_1^{MWU} P_8(t) - (\lambda_1^{DU} + \lambda_1^{EF} + \lambda_1^{VU} + \lambda_2^{VU}) P_{10}(t)$$

$$\frac{dP_9(t)}{dt} = \lambda_2^{HXU} P_1(t) + \lambda_1^{EF} P_8(t) - (\lambda_1^{DU} + \lambda_1^{MWU} + \lambda_1^{VU} + \lambda_2^{VU}) P_9(t)$$

$$\frac{dP_3(t)}{dt} = \lambda_1^{MWU} P_1(t) + \lambda_1^{EF} P_2(t) - (\lambda_1^{DU} + \lambda_1^{VU} + \lambda_2^{VU} + \lambda_1^{HXU} + \lambda_2^{HXU}) P_3(t)$$

(5-23)

Layer 3:

$$\frac{dP_{52}(t)}{dt} = \lambda_1^{HXU} P_{48}(t) + \lambda_1^{VU} P_{40}(t) + \lambda_1^{DU} P_{16}(t) - (\lambda_1^{EF} + \lambda_1^{MWU}) P_{52}(t)$$

$$\frac{dP_{56}(t)}{dt} = \lambda_2^{HXU} P_{48}(t) + \lambda_1^{VU} P_{44}(t) + \lambda_1^{DU} P_{20}(t) - (\lambda_1^{EF} + \lambda_1^{MWU}) P_{56}(t)$$

$$\frac{dP_{50}(t)}{dt} = \lambda_1^{MWU} P_{48}(t) + \lambda_1^{VU} P_{38}(t) + \lambda_1^{DU} P_{14}(t) - (\lambda_1^{HXU} + \lambda_2^{HXU} + \lambda_1^{EF}) P_{50}(t)$$

$$\frac{dP_{49}(t)}{dt} = \lambda_1^{EF} P_{48}(t) + \lambda_1^{VU} P_{37}(t) + \lambda_1^{DU} P_{13}(t) - (\lambda_1^{HXU} + \lambda_2^{HXU} + \lambda_1^{MWU}) P_{49}(t)$$

$$\frac{dP_{64}(t)}{dt} = \lambda_1^{HXU} P_{60}(t) + \lambda_2^{VU} P_{40}(t) + \lambda_1^{DU} P_{28}(t) - (\lambda_1^{MWU} + \lambda_1^{EF}) P_{64}(t)$$

$$\frac{dP_{68}(t)}{dt} = \lambda_2^{HXU} P_{60}(t) + \lambda_2^{VU} P_{44}(t) + \lambda_1^{DU} P_{32}(t) - (\lambda_1^{MWU} + \lambda_1^{EF}) P_{68}(t)$$

$$\frac{dP_{62}(t)}{dt} = \lambda_1^{MWU} P_{60}(t) + \lambda_2^{VU} P_{38}(t) + \lambda_1^{DU} P_{26}(t) - (\lambda_1^{HXU} + \lambda_2^{HXU} + \lambda_1^{EF}) P_{62}(t)$$

$$\frac{dP_{61}(t)}{dt} = \lambda_1^{EF} P_{60}(t) + \lambda_2^{VU} P_{37}(t) + \lambda_1^{DU} P_{25}(t) - (\lambda_1^{HXU} + \lambda_2^{HXU} + \lambda_1^{MWU}) P_{61}(t)$$

$$\frac{dP_{42}(t)}{dt} = \lambda_1^{MWU} P_{40}(t) + \lambda_1^{HXU} P_{38}(t) + \lambda_1^{DU} P_{6}(t) - (\lambda_1^{EF} + \lambda_1^{VU} + \lambda_2^{VU}) P_{42}(t)$$

$$\frac{dP_{41}(t)}{dt} = \lambda_1^{EF} P_{40}(t) + \lambda_1^{HXU} P_{37}(t) + \lambda_1^{DU} P_{5}(t) - (\lambda_1^{VU} + \lambda_2^{VU} + \lambda_1^{MWU}) P_{41}(t)$$

$$\frac{dP_{46}(t)}{dt} = \lambda_1^{MWU} P_{44}(t) + \lambda_2^{HXU} P_{38}(t) + \lambda_1^{DU} P_{10}(t) - (\lambda_1^{VU} + \lambda_2^{VU} + \lambda_1^{EF}) P_{46}(t)$$

$$\frac{dP_{45}(t)}{dt} = \lambda_1^{EF} P_{44}(t) + \lambda_2^{HXU} P_{37}(t) + \lambda_1^{DU} P_{9}(t) - (\lambda_1^{VU} + \lambda_2^{VU} + \lambda_1^{MWU}) P_{45}(t)$$

$$\frac{dP_{39}(t)}{dt} = \lambda_1^{EF} P_{38}(t) + \lambda_1^{MWU} P_{37}(t) + \lambda_1^{DU} P_{3}(t) - (\lambda_1^{VU} + \lambda_2^{VU} + \lambda_1^{HXU} + \lambda_2^{HXU}) P_{39}(t)$$

$$\frac{dP_{18}(t)}{dt} = \lambda_1^{MWU} P_{16}(t) + \lambda_1^{HXU} P_{14}(t) + \lambda_1^{VU} P_{6}(t) - (\lambda_1^{DU} + \lambda_1^{EF}) P_{18}(t)$$

$$\frac{dP_{17}(t)}{dt} = \lambda_1^{HXU} P_{13}(t) + \lambda_1^{EF} P_{16}(t) + \lambda_1^{VU} P_{5}(t) - (\lambda_1^{DU} + \lambda_1^{MWU}) P_{17}(t)$$

$$\frac{dP_{22}(t)}{dt} = \lambda_1^{MWU} P_{20}(t) + \lambda_2^{HXU} P_{14}(t) + \lambda_1^{VU} P_{10}(t) - (\lambda_1^{DU} + \lambda_1^{EF}) P_{22}(t)$$

$$\frac{dP_{21}(t)}{dt} = \lambda_1^{EF} P_{20}(t) + \lambda_2^{HXU} P_{13}(t) + \lambda_1^{VU} P_{9}(t) - (\lambda_1^{DU} + \lambda_1^{MWU}) P_{21}(t)$$

$$\frac{dP_{15}(t)}{dt} = \lambda_1^{EF} P_{14}(t) + \lambda_1^{MWU} P_{13}(t) + \lambda_1^{VU} P_{3}(t) - (\lambda_1^{DU} + \lambda_1^{HXU} + \lambda_2^{HXU}) P_{15}(t)$$

$$\frac{dP_{30}(t)}{dt} = \lambda_1^{MWU} P_{28}(t) + \lambda_1^{HXU} P_{26}(t) + \lambda_2^{VU} P_6(t) - (\lambda_1^{DU} + \lambda_1^{EF}) P_{30}(t)$$

$$\frac{dP_{29}(t)}{dt} = \lambda_1^{EF} P_{28}(t) + \lambda_1^{HXU} P_{25}(t) + \lambda_2^{VU} P_5(t) - (\lambda_1^{DU} + \lambda_1^{MWU}) P_{29}(t)$$

$$\frac{dP_{34}(t)}{dt} = \lambda_1^{MWU} P_{32}(t) + \lambda_2^{HXU} P_{26}(t) + \lambda_2^{VU} P_{10}(t) - (\lambda_1^{DU} + \lambda_1^{EF}) P_{34}(t)$$

$$\frac{dP_{27}(t)}{dt} = \lambda_1^{EF} P_{26}(t) + \lambda_1^{MWU} P_{25}(t) + \lambda_2^{VU} P_3(t) - (\lambda_1^{DU} + \lambda_1^{HXU} + \lambda_2^{HXU}) P_{27}(t)$$

$$\frac{dP_{33}(t)}{dt} = \lambda_1^{EF} P_{32}(t) + \lambda_2^{HXU} P_{25}(t) + \lambda_2^{VU} P_9(t) - (\lambda_1^{DU} + \lambda_1^{MWU}) P_{33}(t)$$

$$\frac{dP_{7}(t)}{dt} = \lambda_1^{EF} P_6(t) + \lambda_1^{MWU} P_5(t) + \lambda_1^{HXU} P_3(t) - (\lambda_1^{DU} + \lambda_1^{VU} + \lambda_2^{VU}) P_7(t)$$

$$\frac{dP_{11}(t)}{dt} = \lambda_1^{EF} P_{10}(t) + \lambda_1^{MWU} P_9(t) + \lambda_2^{HXU} P_3(t) - (\lambda_1^{DU} + \lambda_1^{VU} + \lambda_2^{VU}) P_{11}(t)$$

(5-24)

Layer 4:

$$\frac{dP_{54}(t)}{dt} = \lambda_1^{MWU} P_{52}(t) + \lambda_1^{HXU} P_{50}(t) + \lambda_1^{VU} P_{42}(t) + \lambda_1^{DU} P_{18}(t) - \lambda_1^{EF} P_{54}(t)$$

$$\frac{dP_{53}(t)}{dt} = \lambda_1^{EF} P_{52}(t) + \lambda_1^{HXU} P_{49}(t) + \lambda_1^{VU} P_{41}(t) + \lambda_1^{DU} P_{17}(t) - \lambda_1^{MWU} P_{53}(t)$$

$$\frac{dP_{58}(t)}{dt} = \lambda_1^{MWU} P_{56}(t) + \lambda_2^{HXU} P_{50}(t) + \lambda_1^{VU} P_{46}(t) + \lambda_1^{DU} P_{22}(t) - \lambda_1^{EF} P_{58}(t)$$

$$\frac{dP_{57}(t)}{dt} = \lambda_1^{EF} P_{56}(t) + \lambda_2^{HXU} P_{49}(t) + \lambda_1^{VU} P_{45}(t) + \lambda_1^{DU} P_{21}(t) - \lambda_1^{MWU} P_{57}(t)$$

$$\frac{dP_{51}(t)}{dt} = \lambda_1^{EF} P_{50}(t) + \lambda_1^{MWU} P_{49}(t) + \lambda_1^{VU} P_{39}(t) + \lambda_1^{DU} P_{15}(t) - (\lambda_1^{HXU} + \lambda_2^{HXU}) P_{51}(t)$$

$$\frac{dP_{66}(t)}{dt} = \lambda_1^{MWU} P_{64}(t) + \lambda_1^{HXU} P_{62}(t) + +\lambda_2^{VU} P_{42}(t) + \lambda_1^{DU} P_{30}(t) - \lambda_1^{EF} P_{66}(t)$$

$$\frac{dP_{65}(t)}{dt} = \lambda_1^{EF} P_{64}(t) + \lambda_1^{HXU} P_{61}(t) + +\lambda_2^{VU} P_{41}(t) + \lambda_1^{DU} P_{29}(t) - \lambda_1^{MWU} P_{65}(t)$$

$$\frac{dP_{70}(t)}{dt} = \lambda_1^{MWU} P_{68}(t) + \lambda_2^{HXU} P_{62}(t) + +\lambda_2^{VU} P_{46}(t) + \lambda_1^{DU} P_{34}(t) - \lambda_1^{EF} P_{70}(t)$$

$$\frac{dP_{69}(t)}{dt} = \lambda_1^{EF} P_{68}(t) + \lambda_2^{HXU} P_{61}(t) + +\lambda_2^{VU} P_{45}(t) + \lambda_1^{DU} P_{33}(t) - \lambda_1^{MWU} P_{69}(t)$$

$$\frac{dP_{63}(t)}{dt} = \lambda_1^{EF} P_{62}(t) + \lambda_1^{MWU} P_{61}(t) + \lambda_2^{VU} P_{39}(t) + \lambda_1^{DU} P_{27}(t) - (\lambda_1^{HXU} + \lambda_2^{HXU}) P_{63}(t)$$

$$\frac{dP_{43}(t)}{dt} = \lambda_1^{EF} P_{42}(t) + \lambda_1^{MWU} P_{41}(t) + \lambda_1^{HXU} P_{39}(t) + \lambda_1^{DU} P_7(t) - (\lambda_1^{VU} + \lambda_2^{VU}) P_{43}(t)$$

$$\frac{dP_{47}(t)}{dt} = \lambda_1^{EF} P_{46}(t) + \lambda_1^{MWU} P_{45}(t) + \lambda_2^{HXU} P_{39}(t) + \lambda_1^{DU} P_{11}(t) - (\lambda_1^{VU} + \lambda_2^{VU}) P_{47}(t)$$

$$\frac{dP_{19}(t)}{dt} = \lambda_1^{EF} P_{18}(t) + \lambda_1^{MWU} P_{17}(t) + \lambda_1^{HXU} P_{15}(t) + \lambda_1^{VU} P_7(t) - \lambda_1^{DU} P_{19}(t)$$

$$\frac{dP_{23}(t)}{dt} = \lambda_1^{EF} P_{22}(t) + \lambda_1^{MWU} P_{21}(t) + \lambda_2^{HXU} P_{15}(t) + \lambda_1^{VU} P_{11}(t) - \lambda_1^{DU} P_{23}(t)$$

$$\frac{dP_{31}(t)}{dt} = \lambda_1^{EF} P_{30}(t) + \lambda_1^{MWU} P_{29}(t) + \lambda_1^{HXU} P_{27}(t) + \lambda_2^{VU} P_7(t) - \lambda_1^{DU} P_{31}(t)$$

$$\frac{dP_{35}(t)}{dt} = \lambda_1^{EF} P_{34}(t) + \lambda_1^{MWU} P_{33}(t) + \lambda_2^{HXU} P_{27}(t) + \lambda_2^{VU} P_{11}(t) - \lambda_1^{DU} P_{35}(t)$$

(5-25)

Layer 5:

$$\frac{dP_{55}(t)}{dt} = \lambda_1^{EF} P_{54}(t) + \lambda_1^{MWU} P_{53}(t) + \lambda_1^{HXU} P_{51}(t) + \lambda_1^{VU} P_{43}(t) + \lambda_1^{DU} P_{19}(t) - \lambda_1^{HXU} P_{55}(t)$$

$$\frac{dP_{59}(t)}{dt} = \lambda_1^{EF} P_{58}(t) + \lambda_1^{MWU} P_{57}(t) + \lambda_2^{HXU} P_{51}(t) + \lambda_1^{VU} P_{47}(t) + \lambda_1^{DU} P_{23}(t) + \lambda_1^{HXU} P_{55}(t)$$

$$\frac{dP_{67}(t)}{dt} = \lambda_1^{EF} P_{66}(t) + \lambda_1^{MWU} P_{65}(t) + \lambda_1^{HXU} P_{63}(t) + \lambda_2^{VU} P_{43}(t) + \lambda_1^{DU} P_{31}(t) - \lambda_1^{HXU} P_{67}(t)$$

$$\frac{dP_{71}(t)}{dt} = \lambda_1^{EF} P_{70}(t) + \lambda_1^{MWU} P_{69}(t) + \lambda_2^{HXU} P_{63}(t) + \lambda_2^{VU} P_{47}(t) + \lambda_1^{DU} P_{35}(t) + \lambda_1^{HXU} P_{67}(t)$$

(5-26)

The above coupled ODEs are solved using finite element method in the computer code Fortran95. The following initial conditions at $t = 0$ is assumed for system analysis:

$$P_n(t = 0) = \begin{cases} 1;\ for\ \ n = 0 \\ 0;\ for\ \ n \neq 0 \end{cases}$$

(5-27)

Sample results obtained from the code is presented in *Table 65*. Only three (3) system state transition probabilities with 10 time steps are presented for comprehension. Moreover, due to the volume of information, the results depiction significantly requires a large number of pages (approximately 30 pages). Sample transition state probabilities is provided in *Appendix V*. It is obvious from the result that the probability of the system state $n = 0$ (all units in normal state) decreases as time progresses, i.e., the failure probability of the unit increases with time, which is intuitive. From a total of 72 systems states, it was found that the states $P_1(t)$, $P_{24}(t)$, $P_{36}(t)$, $P_4(t)$,

$P_8(t)$, $P_{37}(t)$, $P_{25}(t)$ and $P_{60}(t)$ are most likely to occur with the state probabilities within the range 10^{-03} and 10^{-06} for $k = 9$. Of course, the simulation time steps could be increased, however it requires significant amount of time to obtain the solution from the computer code. Among the above-mentioned states, $P_1(t)$ have the highest probability of failure. This implies that the ICS system is likely to fail due to an envelope failure or the pressure boundary being compromised, i.e., a pipe break or rupture as compared to the other system states. This scenario directly challenges the IC loop, and a failure of which could rapidly result to a total system failure. A failure of the ICS to maintain its loop pressure will directly affect the reactor coolant system pressure, which can result in a operation of the safety relief valves. System state $P_1(t)$ is followed by the states $P_{24}(t)$ and $P_{36}(t)$ which are the vent unit and drain unit failure respectively. An occurrence of any of these two states will result to a system failure. For instance, the failure of vent unit to vent non-condensable gases from the IC loop will significantly degrade the heat transfer rate and hence the ICS performance. It is evident that a failure of drain unit will directly result to a system failure since the IC loop will be degraded and the condensate coolant flow path will be affected. The reason for $P_1(t)$ being higher than $P_{24}(t)$ and $P_{36}(t)$ is due to the redundancy provided for the vent and condensate return unit. Whereas, $P_{36}(t)$ being lower than $P_{24}(t)$ is due to the diversity provided in the condensate return valves as compared to the identical vent valves. Regardless, any of the above scenarios will have a direct influence on the IC loop pressure and coolant flow, which in turn will affect the reactor coolant system pressure. The predicted time-dependent stochastic ICS system behavior can be utilized in the next step of the project to couple the model with deterministic system evolution, thus enabling an analyst to predict an integrated determinist and probabilistic system evolution.

Table 65: Sample ICS state transition probabilities with 10 time steps

1. For system state $n = 0$:

k	0	1	2	3	4	5	6	7	8	9
$P_0(t)$	1	0.9995	0.998	0.998	0.997	0.997	0.996	0.996	0.995	0.995

2. For system state $n = 1$:

k	0	1	2	3	4	5
$P_1(t)$	0	2.74E-04	5.48E-04	8.21E-04	1.09E-03	1.37E-03

6	7	8	9
1.64E-03	1.91E-03	2.19E-03	2.46E-03

3. For system state $n = 2$:

k	0	1	2	3	4	5
$P_2(t)$	0	3.21E-08	6.41E-08	9.61E-08	1.28E-07	1.591E-07

6	7	8	9
1.92E-07	2.24E-07	2.55E-07	2.87E-07

Note: State transition probabilities are systematically organized by system configuration. A sample total state transition probabilities for 5 time steps is given in *Appendix V*.

CHAPTER 6: SUMMARY AND CONCLUSIONS

This chapter presents the conclusion of the research work and some recommendations for future research. The conclusion is derived from all the chapters of the thesis. Recommendations for potential future research activities as an extension of the present work as well as ongoing and near-term activities to be performed for the CRP are also noted in this chapter.

6.1. Conclusions

This thesis investigates the application and comparison of classical and dynamic PSA techniques that are utilized for safety analysis of nuclear power plants. The classical techniques include the fault tree, event tree and classical Markov model, whereas the dynamic methodologies include the Markov-CCMT model and Dynamic Flowgraph Method. The capabilities and limitations of the techniques are demonstrated by applying it to a benchmark liquid level control system exhibiting dynamic characteristics. The failure probability of the top events (drained and overflow) are evaluated using each method, and the predicted results are compared. The system reliability analysis using FT/ET are performed using the CAFTA code, whereas states transition probabilities in classical Markov model are executed in Fortran95 code. The DFM model of the benchmark system is developed using the code DYMONDA, and the coupling of Markov-CCMT model is performed using Fortran95 and MATLAB tools. The contribution of this research includes;

- Demonstration of the advantages of dynamic techniques over classical methods for reliability modelling and analysis of system exhibiting dynamic characteristics;
- Formulated the time-dependent reliability analysis of system with multiple failure modes and multi-state components;
- Generation of timed fault trees from DFM model through the backtracking process that can be integrated into an existing conventional PSA model;
- Performed a detailed comparison of the two dynamic PSA methods, i.e., DFM and Markov-CCMT models with regard to time-dependent modelling capability, dynamic accident sequence generation, modelling complexity and computational time required;
- Developed a roadmap for the Coordinated Research Project and a novel approach to model and compute the functional reliability of passive safety systems.

The classical techniques are static in nature except the time-dependent Markov model, and consider an instant system failure for a given minimal cut-set. This leads one to conclude that given the set of possible system failures, the classical approach is a systematic approach linking known anticipated events. However, it has been shown in this research that a failure event does not necessarily result in an instant system failure, rather there is a time delay which is dependent on the evolution of the state variables. The knowledge of time delay is important since it may be the time available for the operators to respond to an initiating event, or the time available for the next frontline system to activate on demand. It is self-evident that classical approach limits itself to the state of the hardware system, whereas the dynamic approach simultaneously considers both the hardware states as well as the magnitude of the state variables. The top events in classical techniques are defined only in terms of failure probability of the hardware state, whereas dynamic methods define the top events in terms of magnitude of state variables. This research thus demonstrates that the dynamic methods can capture the time element and the time-dependent probabilistic accident sequence evolution, and hence provides a more detailed approach to evaluating complex dynamic system behaviour.

However, this comes at the expense of model complexity, and computationally, the time required to process large transition matrix. Additional factors include, the large number of simulation runs and complicated coupling process/algorithm. For instance, consider the Markov-CCMT model. This physical model of the liquid level control system was created using MATLAB and verified via analytical method. The stochastic model was developed systematically via Markov transition diagram, and the generated coupled ODEs was solved in Fortran95 code. State variable discretization, i.e., five (5) computational cells was performed in the MATLAB environment. Based on the number of computational cells and possible systems states from the stochastic model, the number of runs to be performed was determined. The evolution of state variable in the state-space for each run is mapped, and its conditional probabilities computed using the CCMT model. The coupling of the two models i.e., Markov and CCMT, and evaluation of the transition probability matrix was performed in MATLAB. The size of the transition matrix increases with an increase in number of system states and computational cells, and hence evaluation of the matrix become considerably complicated requiring significant computational time.

The impact of state ordering of the accident sequences is also demonstrated such that the dynamic techniques model ordering of failures and thus provides more information of the failure sequence.

DFM and Markov-CCMT model strictly account for failure ordering which is probabilistic in nature and enable probabilistic dynamic system evolution, i.e., simultaneously accounting for failure ordering as well as system dynamics. This is due to the fact that both system state variables and hardware states are accounted for simultaneously. In contrast, FT technique represents a system failure without any particular ordering index; that is, the sequence is fixed apriori by experts undertaking ET analysis. Here, analysis has shown that DFM present advantages in this regard. Furthermore, DFM can account for stochastic accident sequence evolution, in contrast to the classical techniques and Markov-CCMT model where the possible system states are set a priori by the analyst. For instance, with an increase in backtracking steps, basic events in the prime implicants (analogous to minimal cut-sets in fault tree) increases, whereas the possible system state in Markov-CCMT model remains the same except with a deviation in the failure probability. This is of vital importance as the evolution of accident sequence can be observed from the individual timed-prime implicants (time stamped). DFM however lacks the capability to model deviation in component/system failure rate and demand frequency influenced by a prior failure event. The predicted failure probability of the benchmark system for both overflow and drained scenarios show significant deviation from classical and dynamic techniques. The difference in the predicted failure probabilities is attributed to lack of treatment of dynamic interactions among the units through the state variable, and consideration of independence among unit failure modes in the classical approach. In contrast, dynamic methods account for competition within the failure modes and treat all unit modes in an integrated fashion. The classical approach yields conservative results and thus serves to identify logical correctness and establishes conservative relationships between top events and system end states. The viability of integrating the results obtained from a DFM model to an existing static PSA model is depicted by generating timed-fault trees through the backtracking algorithm. we note that even though it is impractical to model the entire plant using DFM, specific system important to safety can be modelled with DFM, and its results can be integrated after post-processing of timed-prime implicants to construct timed-fault trees.

The current research also achieves its objective in developing a preliminary roadmap and selection of a benchmark passive safety system for the coordinated research project. The passive ICS was selected on the basis that: many current generation SMRs concepts implements ICS, availability of research materials and the objective within the international community to determine the functional reliability of system based on natural circulation. A dynamic and integrated approach

for safety assessment of passive systems is developed and presented in Chapter 5. A Markov-CCMT model was selected as the most viable candidate for ICS system modelling due to its nature of tight coupling, and the capability to capture dynamic interactions between systems that results to a deviation in demand frequency of the interacting systems. As a first step towards integrated safety assessment, two classical approaches to reliability modelling of the passive ICS were proposed: (a) small Markov model and large fault tree; and (2) small fault tree and large Markov model. The first approach is largely classical (static in nature) and provide results in terms of cut-sets and is unable to incorporate dynamic characteristics such as state variable evolution and state ordering. The latter approach enables one to model system level time-dependent characteristics as well as merge unit/component states using FT failure rate evaluation method. This results in a reduced system state. This indicates that the latter approach for stochastic system modelling is practical, and subsequently eventual coupling with ICS deterministic model. Reliability of the ICS and hence the system state transition probabilities are predicted using the latter approach. The analysis confirms a high system reliability for the required mission time.

6.2. Recommendations for Future Work

Based on the thesis research activities performed, recommendations and anticipated directions are presented in this sub-section. The recommendations also include the activities to be performed for the research project, and is aligned with the roadmap of the project outlined in Chapter 5. The potential subject of research includes:

1. Development of a mechanized coupling process of stochastic and deterministic system model (best estimate code) through a user-friendly computer code. For instance, the coupling of stochastic model in Fortran95 and physical model in RELAP5 can be realized using Python as the coupling tool.
2. Most, if not all, of the existing NPPs are based on static classical PSA approach. This requires post-processing of results obtained from dynamic techniques in order to integrate into an existing PSA model. Due to the significant volume of information obtained from dynamic methods, post-processing is itself a challenge. This will require time and investigation. Furthermore, there exist no standard technique for post-processing and is thus, analyst dependent. Additional work should be performed to develop a systematic approach for post-processing of results that can be integrated with an existing PSA model.

3. The ICS is a passive reactor pressure depressurization system. However, in almost all current generation reactors, an active system is also dedicated for the same purpose, such as safety relief valves (SRVs) or automatic depressurization systems (ADS). The modelling of complex dynamic interactions between the two-depressurization system through the RCS pressure evolution needs further research. This will enable one to comprehend how dynamic interactions can influence the failure rate, demand frequency and hence the reliability of interacting systems. This may be particularly important for PSSs that function on a small driving force which is highly sensitive to system parameters such as heat loss, presence of non-condensable gases, pipe fouling and oxidation. Unlike the active systems where these parameters are negligible due to the high driving force. These phenomenon in turn influence the state variable (i.e., RCS pressure) evolution, and thus the demand frequency and reliability of interacting systems.

In conclusion, risk assessment of NPPs requires accurate estimates of the frequency of accident scenarios and adequate treatment of dependencies among multiple failure events. The ET/FT methodology currently used is fundamentally well-suited to conservatively treat dependencies that can be expressed in static and logical terms. However, it is not as well-suited to treat dependencies associated with time-dependent processes or continuously varying variables. The methodology does not explicitly carry information concerning the evolution of state variables and operator state that may couple a number of failure events together. the three major severe accidents (TMI, Chernobyl and Fukushima Daiichi) substantiate this observation. The study shows that the two dynamic methods provide a promising pathway to bridging the gap and overcoming the limitations encountered in classical PSA; hence providing risk-significant information and insights into the probabilistic system dynamic evolution. Lastly, dynamic PSA techniques can be used as a complement to the classical approach in that, it provides information on the system dynamic evolution besides providing confirmatory information as that available via classical techniques.

REFERENCES

1. Acosta, C., and N. Siu, “Dynamic event tree analysis method (DETAM) for accident sequence analysis”, Cambridge, MA: Dept. of Nuclear Engineering, Massachusetts Institute of Technology (1991).
2. Acosta, C., and N. Siu, "Dynamic event trees in accident sequence analysis: application to steam generator tube rupture", Reliability Engineering & System Safety 41.2 (1993): 135-154.
3. Aldemir, T., "Computer-Assisted Markov Failure Modeling of Process Control System," IEEE Transactions on Reliability, 36 (1987):1, 133-144.
4. Aldemir, T., "A survey of dynamic methodologies for probabilistic safety assessment of nuclear power plants", Annals of Nuclear Energy 52 (2013): 113-124.
5. Aldemir, T., D. Miller, M. Stovsky, J. Kirschenbaum, P. Bucci, A. Fentiman, L. Mangan, and S. Arndt, "NUREG/CR 6901: Current state of reliability modeling methodologies for digital systems and their acceptance criteria for nuclear power plant assessments", Office of Nuclear Regulatory Research, US Nuclear Regulatory Commission, Washington DC (2006).
6. Aldemir, T., et al., "A benchmark implementation of two dynamic methodologies for the reliability modeling of digital instrumentation and control systems", NUREG/CR-6985, US Nuclear Regulatory Commission, Washington DC (2009).
7. Aldemir, T., et al., "Dynamic reliability modeling of digital instrumentation and control systems for nuclear reactor probabilistic risk assessments", NUREG0CR-6942, US Nuclear Regulatory Commission Washington DC (2007).
8. Amoia, V., G. D. Micheli, and M. Santomauro, "Computer-oriented formulation of transition-rate matrices via Kronecker algebra", IEEE Transactions on Reliability 30.2 (1981): 123-132.
9. ASCA Inc., “Dymonda 7.0 Software Guide,” ASCA Inc., California, USA, 2013.
10. Basharin, G. P., Langville, A. N., and Naumov, V. A., "The life and work of AA Markov", Linear algebra and its applications 386, 3-26, (2004).
11. Beeson, S. C., “Non-coherent Fault Tree Analysis”, PhD Diss., Loughborough University (2002).
12. Belhadj, M., and T. Aldemir. "The Cell to Cell Mapping technique and Chapman-Kolmogorov representation of system dynamics", Journal of Sound and Vibration 181.4 (1995): 687-707.

13. Bucci, P., et al., "Construction of event-tree/fault-tree models from a Markov approach to dynamic system reliability", Reliability Engineering & System Safety 93.11 (2008): 1616-1627.
14. Burgazzi, L., "Open issues associated with passive safety systems reliability assessment", International Conference on Opportunities and Challenges for Water Cooled Reactors in the 21st Century, Vienna (2009).
15. C. J. Garrett, S. B. Guarro and G. E. Apostolakis, “The Dynamic Flowgraph Methodology for Assessing the Dependability of Embedded Software Systems,” IEEE Transactions on Systems, Man, and Cybernetics, vol. 25, pp. 824-840 (1995).
16. Cafaro, G., F. Corsi, and F. Vacca, "Multistate Markov models and structural properties of the transition-rate matrix", IEEE Transactions on Reliability 35.2 (1986): 192-200.
17. Canadian Nuclear Safety Commission, "REGDOC-2.4. 2, Safety Analysis: Probabilistic Safety Assessment (PSA) for Nuclear Power Plants", Ottawa (2014).
18. Canadian Nuclear Safety Commission, "REGDOC-2.5. 2, Design of Reactor Facilities: Nuclear Power Plants", Regulatory Document, Ottawa (2014).
19. Chen, Z., “Global Analysis, Control, and Bayesian Estimation of Nonlinear Dynamic Systems Using Cell-to-cell Mapping”, PhD Dissertation, Cleveland State University (2004).
20. Devooght, J., and C. Smidts, "Probabilistic dynamics as a tool for dynamic PSA", Reliability Engineering & System Safety 52.3 (1996): 185-196.
21. Devooght, J., and C. Smidts, "Probabilistic dynamics as a tool for dynamic PSA", Reliability Engineering & System Safety 52.3 (1996): 185-196.
22. Devooght, J., and C. Smidts, "Probabilistic reactor dynamics—I: the theory of continuous event trees", Nuclear Science and Engineering 111.3 (1992): 229-240.
23. Devooght, J., and C. Smidts. "Probabilistic reactor dynamics—I: the theory of continuous event trees", Nuclear Science and Engineering 111.3 (1992a): 229-240.
24. Devooght, J., and C. Smidts. "Probabilistic reactor dynamics—III. A framework for time-dependent interaction between operator and reactor during a transient involving human error", Nuclear Science and Engineering 112.2 (1992b): 101-113.
25. Dinca, L. G., “A probabilistic approach to parameter estimation towards fault diagnosis in nonlinear dynamic systems”, PhD Dissertation, The Ohio State University (1997).

26. Electric Power Research Institute, “CAFTA Fault Tree Analysis System Version 6.0”, EPRI Inc., Charlotte, NC, USA (2013).
27. Fussell, J. B., "A formal methodology for fault tree construction", Nuclear Science and Engineering 52.4 (1973): 421-432.
28. Garrett, C. J., and G. E. Apostolakis, "Automated hazard analysis of digital control systems", Reliability Engineering & System Safety 77.1 (2002): 1-17.
29. Garribba, S., E. Guagnini, and P. Mussio, "Multiple-valued logic trees: meaning and prime implicants", IEEE Transactions on Reliability 34.5 (1985): 463-472.
30. GE Nuclear Energy, “SBWR Standard Safety Analysis”, MFN 04-138 (1992).
31. Gomes, I. B., P. L. C. Saldanha, and P. F. F. F. Melo, "A cell-to-cell Markovian model for the reliability of a digital control system of a steam generator", Proceedings of the 2013 International Nuclear Atlantic Conference-INAC (2013).
32. Hsu, C. S., "A generalized theory of cell-to-cell mapping for nonlinear dynamical systems", Journal of Applied Mechanics 48.3 (1981): 634-642.
33. Hsu, C. S., "A theory of cell-to-cell mapping dynamical systems", Journal of Applied Mechanics 47.4 (1980a): 931-939.
34. Hsu, C. S., and R. S. Guttalu. "An unravelling algorithm for global analysis of dynamical systems: An application of cell-to-cell mappings", Journal of Applied Mechanics 47.4 (1980b): 940-948.
35. International Atomic Energy Agency, “Probabilistic Safety Assessment”, INSAG-6, a report by the International Nuclear Safety Advisory Group, IAEA, Vienna (1992).
36. Jafari, J., et al. "Reliability evaluation of a natural circulation system." Nuclear Engineering and Design 224.1 (2003): 79-104.
37. Khan, H. J., and U. S. Rohatgi, “Performance characterization of isolation condenser of SBWR”, BNL-NUREG-47960; CONF-921102-25, Brookhaven National Lab., NY (1992).
38. Kirschenbaum, J., et al. "A benchmark system for comparing reliability modeling approaches for digital instrumentation and control systems", Nuclear Technology 165.1 (2009): 53-95.
39. Kirschenbaum, J., et al., "A benchmark system for comparing reliability modeling approaches for digital instrumentation and control systems", Nuclear Technology 165.1 (2009): 53-95.

40. Kumamoto, H. and Ernest J. H., “Probabilistic risk assessment and management for engineers and scientists”, Wiley-IEEE (2000).
41. Kumamoto, H., “Satisfying safety goals by probabilistic risk assessment”, Springer Science & Business Media (2007).
42. Labeau, P., C. Smidts, and S. Swaminathan, "Dynamic reliability: towards an integrated platform for probabilistic risk assessment", Reliability Engineering & System Safety 68.3 (2000): 219-254.
43. Lesanovsky, A., "Multistate Markov models for systems with dependent units", IEEE Transactions on Reliability 37.5 (1988): 505-511.
44. Lorenzo, G., et al., "Assessment of an isolation condenser of an integral reactor in view of uncertainties in engineering parameters." Science and technology of Nuclear Installations (2011).
45. M. Belhadj, M. Hassan and T. Aldemir, "On the need for dynamic methodologies in risk and reliability studies", Reliability Engineering and System Safety, vol. 38, pp. 219-236, 1992.
46. M. Hassan and T. Aldemir, "A Data Base Oriented Dynamic Methodology for the Failure Analysis of Closed Loop Control Systems in Process Plants", Reliability Engineering and System Safety, vol. 27, pp. 275-322, 1990.
47. Mandelli, D., “Reliability Modeling of Digital Control Systems Using the Markov/cell-to-cell Mapping Technique”, MS Dissertation, The Ohio State University (2008).
48. Marques, M., et al. "Methodology for the reliability evaluation of a passive system and its integration into a probabilistic safety assessment", Nuclear Engineering and Design 235.24 (2005): 2612-2631.
49. Marques, M., et al. "Methodology for the reliability evaluation of a passive system and its integration into a probabilistic safety assessment." Nuclear Engineering and Design 235.24 (2005): 2612-2631.
50. Marseguerra, M., and E. Zio, "Monte Carlo approach to PSA for dynamic process systems", Reliability Engineering & System Safety 52.3 (1996): 227-241.
51. Marseguerra, M., et al., "A concept paper on dynamic reliability via Monte Carlo simulation", Mathematics and Computers in simulation 47.2-5 (1998): 371-382.

52. McNelles, P., “Dynamic safety assessment of FPGA-based safety critical systems with applications in nuclear power generation”, PhD Dissertation, University of Ontario Institute of Technology (2016).
53. Mezio, F., et al. "Integration of the functional reliability of two passive safety systems to mitigate a SBLOCA+ BO in a CAREM-like reactor PSA." Nuclear Engineering and Design 270 (2014): 109-118.
54. Nayak, A. K., A. Chandrakar, and G. Vinod, "A review: passive system reliability analysis–accomplishments and unresolved issues", Frontiers in Energy Research 2 (2014): 40.
55. Nayak, A. K., and R. K. Sinha. "Role of passive systems in advanced reactors." Progress in Nuclear Energy 49.6 (2007): 486-498.
56. Nayak, A. K., et al. "Passive system reliability analysis using the APSRA methodology." Nuclear Engineering and Design 238.6 (2008(a)): 1430-1440.
57. Nayak, A. K., et al. "Reliability assessment of passive containment isolation system using APSRA methodology." Annals of Nuclear Energy 35.12 (2008(b)): 2270-2279.
58. Nieuwhof, G. W. E., "An introduction to fault tree analysis with emphasis on failure rate evaluation", Microelectronics Reliability 14.2 (1975): 105-119.
59. Ogunbiyi, E. I., "Application of Decision Tables to Risk Analysis Studies", PhD Dissertation, University of Houston, Texas (1981b).
60. Ogunbiyi, E. I., and E. J. Henley, "Irredundant forms and prime implicants of a function with multistate variables", IEEE Transactions on Reliability 30.1 (1981a): 39-42.
61. Papazoglou, I. A., and E. P. Gyftopoulos, "Markov processes for reliability analyses of large systems", IEEE Transactions on Reliability 26.3 (1977): 232-237.
62. Platz, O., "A Markov model for common-cause failures", Reliability Engineering 9.1 (1984): 25-31.
63. Privault, N., “Understanding Markov chains: examples and applications”, Springer Science & Business Media (2013).
64. Pukite, P., and Pukite, J., “Modeling for Reliability Analysis: Markov Modeling for Reliability, Maintainability, Safety, and Supportability Analyses of Complex Computer Systems”, Wiley-IEEE Press (1998).
65. Ruijters, E., and M. Stoelinga. "Fault tree analysis: a survey of the state-of-the-art in modeling, analysis and tools", Computer Science Review 15 (2015): 29-62.

66. Guarro, S., M. Yau and M. Motamed, “Development of Tools for Safety Analysis of Control Software in Advanced Reactors”, NUREG/CR-6465, U.S. Nuclear Regulatory Commission, Washington DC (1996).
67. Salem, S. L., G. E. Apostolakis, and D. Okrent, "A new methodology for the computer-aided construction of fault trees", Annals of Nuclear Energy 4.9-10 (1977): 417-433.
68. Salem, S. L., G. E. Apostolakis, and D. Okrent, “Computer-oriented approach to fault-tree construction”, No. EPRI-NP-288, California University (1976).
69. Salem, S. L., J. S. Wu, and G. Apostolakis, "Decision table development and application to the construction of fault trees", Nuclear Technology 42.1 (1979): 51-64.
70. Sheskin, T. J., “Markov chains and decision processes for engineers and managers”, CRC Press (2016).
71. Siu, N. "Risk assessment for dynamic systems: an overview", Reliability Engineering & System Safety 43.1 (1994): 43-73.
72. Siu, N., C. Acosta, and N. C. Rasmussen, “Physical dependencies in accident sequence analysis”, Cambridge, MA: Nuclear Engineering Dept., Massachusetts Institute of Technology (1989).
73. Smidts, C. "Probabilistic Reactor Dynamics—IV. An Example of Man/Machine Interaction", Nuclear Science and Engineering 112.2 (1992): 114-126.
74. Smidts, C., and J. Devooght. "Probabilistic reactor dynamics—II: A Monte Carlo study of a fast reactor transient", Nuclear Science and Engineering 111.3 (1992): 241-256.
75. Spek, J. A. W., “Cell mapping methods: modifications and extensions”, PhD Dissertation, Eindhoven University of Technology, Netherlands (1994).
76. Stamatelatos, M., W. Vesely, J. Dugan, J. Fragola, J. Minarick, and J. Railsback, "Fault tree handbook with aerospace applications", Office of Safety and Mission Assurance, NASA HQ, (2002).
77. Taylor, Z., and Ranganathan, S., “Designing High Availability Systems: DFSS and Classical Reliability Techniques with Practical Real Life Examples”, John Wiley & Sons (2013).
78. Tombuyses, B., and T. Aldemir, "Computational efficiency of the continuous cell-to-cell mapping technique as a function of integration schemes", Reliability Engineering & System Safety 58.3 (1997): 215-223.

79. US Nuclear Regulatory Commission, "PRA Procedures Guide", NUREG/CR-2300, Washington DC (1983).
80. US Nuclear Regulatory Commission, “Reactor Safety Study”, WASH-1400 (NUREG-75/014), US Nuclear Regulatory Commission, Washington DC (1975).
81. Vesely, W. E., Goldberg, F. F., Roberts, N. H., & Haasl, D. F., “Fault tree handbook”, NUREG/CR-0492, Nuclear Regulatory Commission, Washington DC (1981).
82. Wang, P., and T. Aldemir, "Some improvements in state/parameter estimation using the cell-to-cell mapping technique", Nuclear Science and Engineering 147.1 (2004): 1-25.
83. Yang, J., and Aldemir, T., "An algorithm for the computationally efficient deductive implementation of the Markov/Cell-to-Cell-Mapping Technique for risk significant scenario identification", Reliability Engineering & System Safety 145 (2016): 1-8.
84. Yau, M., "Dynamic flowgraph methodology for the analysis of software-based controlled systems", PhD Dissertation, University of California, Los Angeles (1997).
85. Yau, M., G. Apostolakis and S. Guarro, “The use of prime implicants in dependability analysis of software controlled systems”, Reliability Engineering and System Safety, 62, 23-32 (1998).
86. Yau, M., M. Motamed and S. Guarro, “Assessment and Integration of Software Risk within PRA”, International Journal of Performability Engineering, 3 (2007): 369-378.
87. Yau, M., S. Guarro, and G. Apostolakis, "Demonstration of the dynamic flowgraph methodology using the Titan II space launch vehicle digital flight control system", Reliability Engineering & System Safety 49.3 (1995): 335-353.
88. Zeliang, C., O. P. Singh, and P. Munshi, "Uncertainty evaluation of reliability of shutdown system of a medium size fast breeder reactor", Nuclear Engineering and Design 308 (2016): 283-296.
89. Zio, E., "Integrated deterministic and probabilistic safety assessment: concepts, challenges, research directions", Nuclear Engineering and Design 280 (2014): 413-419.
90. Zio, E., and N. Pedroni, "Building confidence in the reliability assessment of thermal-hydraulic passive systems", Reliability Engineering & System Safety 94.2 (2009): 268-281.

APPENDIX I

Multi-state modelling with FT and Markov model

To demonstrate the capability of FT and Markov model in modeling multi-state component/system, consider a simple non-repairable system consisting of two components connected in parallel. Assume that the system is fully operational initially and is represented by the system state S_0 (both components in normal state) at $t = 0$. Two distinct cases are considered for analysis and illustration purpose.

1. Components with one failure modes (binary);
2. Components with two failure modes (multi-state).

1. Components with one failure modes

It is considered that each component is independent and has only one failure mode i.e., normal or failed. The state transition probabilities are λ_A and λ_B for components A and B respectively. The Markov state transition diagram for the redundant system is given in *Figure A-I-1*.

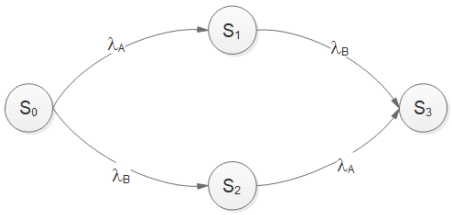


Figure A-I-1: Second-order failures Markov state transition diagram

Figure A-I-1 can be characterized by a set of ODEs:

$$\frac{dS_0}{dt} = -(\lambda_A + \lambda_B)S_0$$
$$\frac{dS_1}{dt} = \lambda_A S_0 - \lambda_B S_1$$
$$\frac{dS_2}{dt} = \lambda_B S_0 - \lambda_A S_2$$
$$\frac{dS_3}{dt} = \lambda_B S_1 + \lambda_A S_2 \qquad \text{(A-I-1)}$$

Solving the set of ODEs with the initial conditions $S_0 = 1$ and $S_1 = S_2 = S_3 = 0$, we obtain the following state probabilities.

$$S_0(t) = e^{-(\lambda_A+\lambda_B)t}$$

$$S_1(t) = e^{-\lambda_B t}(1 - e^{-\lambda_A t})$$

$$S_2(t) = e^{-\lambda_A t}(1 - e^{-\lambda_B t})$$

$$S_3(t) = \left(1 - e^{-\lambda_A t}\right).(1 - e^{-\lambda_B t}) \quad \text{(A-I-2)}$$

Since a system failure occurs when it occupies state S_3 (both component failure), the system failure probability via Markov model is given by:

$$S_3(t) = \left(1 - e^{-\lambda_A t}\right).(1 - e^{-\lambda_B t}) \quad \text{(A-I-3)}$$

The system failure probability (top event) predicted by the FT technique for the redundant system is given by an AND gate as follow:

Failure probability of component A, $P_A = \left(1 - e^{-\lambda_A t}\right)$ (A-I-4)

Failure probability of component B, $P_B = \left(1 - e^{-\lambda_B t}\right)$ (A-I-5)

$$P(Top\ Event) = P_A \wedge P_B = \left(1 - e^{-\lambda_A t}\right).\left(1 - e^{-\lambda_B t}\right) \quad \text{(A-I-6)}$$

Clearly, the top event (failure probability) predicted by FT and Markov model are the same.

2. Components with two failure modes

For the second case, components A and B are considered to have two failure modes $\lambda_1^A, \lambda_2^A, \lambda_1^B$ and λ_2^B respectively with the same system configuration as the first case. The state transition diagram is shown in *Figure A-I-1*.

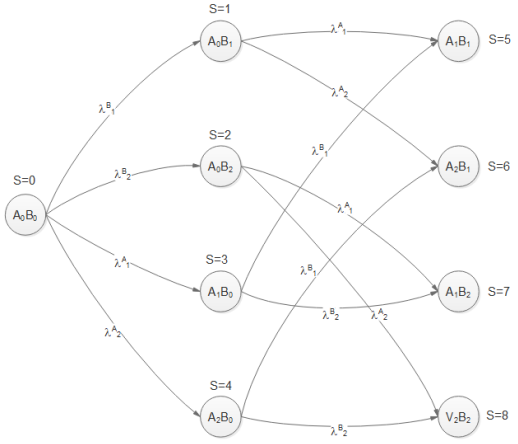


Figure A-I-2: Markov state transition diagram for second case

The set of ODEs associated with the state transition diagram are:

$$\begin{aligned}
\frac{dS_0(t)}{dt} &= -(\lambda_1^A + \lambda_2^A + \lambda_1^B + \lambda_2^B)S_0(t) \\
\frac{dS_1(t)}{dt} &= \lambda_1^B S_0(t) - (\lambda_1^A + \lambda_2^A)S_1(t) \\
\frac{dS_2(t)}{dt} &= \lambda_2^B S_0(t) - (\lambda_1^A + \lambda_2^A)S_2(t) \\
\frac{dS_3(t)}{dt} &= \lambda_1^A S_0(t) - (\lambda_1^B + \lambda_2^B)S_3(t) \\
\frac{dS_4(t)}{dt} &= \lambda_2^A S_0(t) - (\lambda_1^B + \lambda_2^B)S_4(t) \\
\frac{dS_5(t)}{dt} &= \lambda_1^A S_1(t) + \lambda_1^B S_3(t) \\
\frac{dS_6(t)}{dt} &= \lambda_2^A S_1(t) + \lambda_1^B S_4(t) \\
\frac{dS_7(t)}{dt} &= \lambda_1^A S_2(t) + \lambda_2^B S_3(t) \\
\frac{dS_8(t)}{dt} &= \lambda_2^A S_2(t) + \lambda_2^B S_4(t)
\end{aligned} \tag{A-I-7}$$

Considering that the system is operational at $t = 0$, initial condition is given as:

$$S_n(0) = 0; for\ n \neq 0 \tag{A-I-8}$$

Where; $n =$ the number of system states

The resulting solution of the ODEs are:

$$S_0(t) = e^{-(\lambda_1^A + \lambda_2^A + \lambda_1^B + \lambda_2^B)t}$$

$$S_1(t) = \frac{\lambda_1^B}{\lambda_1^B + \lambda_2^B}\left(e^{-(\lambda_1^A + \lambda_2^A)t} - e^{-(\lambda_1^A + \lambda_2^A + \lambda_1^B + \lambda_2^B)t}\right)$$

$$S_2(t) = \frac{\lambda_2^B}{\lambda_1^B + \lambda_2^B}\left(e^{-(\lambda_1^A + \lambda_2^A)t} - e^{-(\lambda_1^A + \lambda_2^A + \lambda_1^B + \lambda_2^B)t}\right)$$

$$S_3(t) = \frac{\lambda_1^A}{\lambda_1^A + \lambda_2^A}\left(e^{-(\lambda_1^B + \lambda_2^B)t} - e^{-(\lambda_1^A + \lambda_2^A + \lambda_1^B + \lambda_2^B)t}\right)$$

$$S_4(t) = \frac{\lambda_2^A}{\lambda_1^A + \lambda_2^A}\left(e^{-(\lambda_1^B + \lambda_2^B)t} - e^{-(\lambda_1^A + \lambda_2^A + \lambda_1^B + \lambda_2^B)t}\right)$$

$$S_5(t) = \frac{\lambda_1^A \lambda_1^B}{\lambda_1^A \lambda_1^B + \lambda_1^A \lambda_2^B + \lambda_2^A \lambda_1^B + \lambda_2^A \lambda_2^B}\left(1 + e^{-(\lambda_1^A + \lambda_2^A + \lambda_1^B + \lambda_2^B)t} - e^{-(\lambda_1^B + \lambda_2^B)t} - e^{-(\lambda_1^A + \lambda_2^A)t}\right)$$

$$S_6(t) = \frac{\lambda_2^A \lambda_1^B}{\lambda_1^A \lambda_1^B + \lambda_1^A \lambda_2^B + \lambda_2^A \lambda_1^B + \lambda_2^A \lambda_2^B}\left(1 + e^{-(\lambda_1^A + \lambda_2^A + \lambda_1^B + \lambda_2^B)t} - e^{-(\lambda_1^B + \lambda_2^B)t} - e^{-(\lambda_1^A + \lambda_2^A)t}\right)$$

$$S_7(t) = \frac{\lambda_1^A \lambda_2^B}{\lambda_1^A \lambda_1^B + \lambda_1^A \lambda_2^B + \lambda_2^A \lambda_1^B + \lambda_2^A \lambda_2^B}\left(1 + e^{-(\lambda_1^A + \lambda_2^A + \lambda_1^B + \lambda_2^B)t} - e^{-(\lambda_1^B + \lambda_2^B)t} - e^{-(\lambda_1^A + \lambda_2^A)t}\right)$$

$$S_8(t) = \frac{\lambda_2^A \lambda_2^B}{\lambda_1^A \lambda_1^B + \lambda_1^A \lambda_2^B + \lambda_2^A \lambda_1^B + \lambda_2^A \lambda_2^B}\left(1 + e^{-(\lambda_1^A + \lambda_2^A + \lambda_1^B + \lambda_2^B)t} - e^{-(\lambda_1^B + \lambda_2^B)t} - e^{-(\lambda_1^A + \lambda_2^A)t}\right) \quad \text{(A-I-9)}$$

The system unreliability is determined by summing all the second order failures.

$$\bar{R}(t) = \sum_{n=5}^{8} S_n(t) \quad \text{(A-I-10)}$$

$$\bar{R}(t) = \frac{\left(1 + e^{-\left(\lambda_1^A + \lambda_2^A + \lambda_1^B + \lambda_2^B\right)t} - e^{-\left(\lambda_1^B + \lambda_2^B\right)t} - e^{-\left(\lambda_1^A + \lambda_2^A\right)t}\right)}{\lambda_1^A \lambda_1^B + \lambda_1^A \lambda_2^B + \lambda_2^A \lambda_1^B + \lambda_2^A \lambda_2^B}\left(\lambda_1^A \lambda_1^B + \lambda_2^A \lambda_1^B + \lambda_1^A \lambda_2^B + \lambda_2^A \lambda_2^B\right) \quad \text{(A-I-11)}$$

Some differences between FTA and Markov model can be observed. To highlight one key difference, consider the system state $S_8(t)$, where both component A and B are in a failed state with modes λ_2^A and λ_2^B respectively. The individual failure probabilities of the components are:

$$P_{A_2} = \left(1 - e^{-\lambda_2^A t}\right) \text{ and } P_{B_2} = \left(1 - e^{-\lambda_2^B t}\right) \quad \text{(A-I-12)}$$

Therefore, system failure probability (top event) predicted by FT technique is computed as:

$$\text{Top Event} = P_{A_2} \wedge P_{B_2} = \left(1 - e^{-\lambda_2^A \boldsymbol{t}}\right).\left(1 - e^{-\lambda_2^B \boldsymbol{t}}\right) \quad \text{(A-I-13)}$$

Whereas, the failure probability of the top event computed from Markov model is given in *Equation (A-I-9)*. Clearly, it can be observed that the failure probability predicted by the FT and Markov model are different. For illustration purpose, consider that the component A have two failure modes with failure rates $\lambda_1^A = 0.005\, per\ hr$ and $\lambda_2^A = 0.007\, per\ hr$, and component B have two failure modes with failure rates of $\lambda_1^B = 0.005\, per\ hr$ and $\lambda_2^B = 0.007\, per\ hr$ respectively. The two cases are simulated for comparison purpose.

- Components A and B with binary mode (normal and failed)
- Components A and B with multiple modes, and system failure occurs when A and B are in λ_2^A and λ_2^B respectively.

It may be observed from *Figure A-I-3* that the top event probabilities computed from the two methods are different. The top event computed from Markov model is smaller than the one from the FT technique (green and red curve). This is due to the fact that Markov model accounts for competition between failure modes of a component, while the FT treats the two failure modes independently. For the explicit mode consideration, top event from FT is simply given by an AND gate between λ_2^A and λ_2^B, whereas in Markov model, $S_8(t)$ represents the top event.

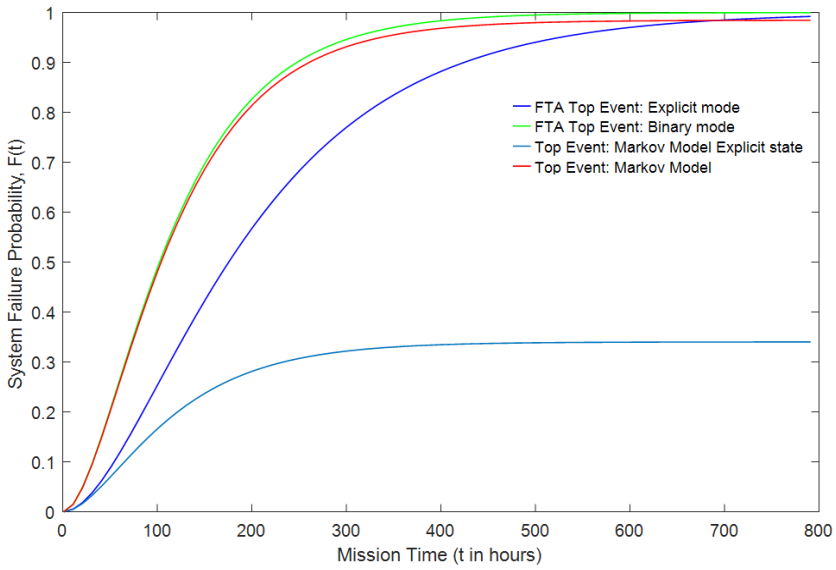


Figure A-I-3: Top event comparison between FT and Markov model

The asymptotic behavior of the system predicted by FT tends to '1' as $(t \to \infty)$, whereas the system state $S_8(t)$ from the Markov model as $(t \to \infty)$ is given by:

$$S_8(t \to \infty) = \frac{\lambda_2^A \lambda_2^B}{\lambda_1^A \lambda_1^B + \lambda_1^A \lambda_2^B + \lambda_2^A \lambda_1^B + \lambda_2^A \lambda_2^B} \tag{A-I-14}$$

i.e., $S_8(t \to \infty) = 0.3403$ (limiting state probability), as opposed to '1' predicted by the FT technique. These properties of Markov model accounting for failure modes in an integrated fashion provide a superior way to model components/systems with multiple failure modes or states.

APPENDIX II

```
! Chireuding Zeliang/Major: Nuclear Engineering
! Faculty of Energy Systems and Nuclear Science: University of Ontario Institute of Technology,
Oshawa, Ontario, 2018
Program Benchmark_Liquid_Level_Control_System
! Declaration of variables
Real :: M(27,27), A(27,27), P(11,27), dt, t, q(27), sum
Integer :: i, j, k
! Initialization of variables
i=0
j=0
k=0
dt=1.
do i= 1, 27
        do j= 1, 27
                M(i,j)=0.
                A(i,j)=0.
                if (i.eq.j) then
                A(i,j)=1.       ! Identity matrix
                end if
        end do
end do
do k=1,11       ! The number of time steps
        do i=1,27
                P(k,i)=0.
                q(i)=0.         ! Initial transition matrix
        end do
end do
P(1,1)=1.       ! Initial state of the system
! Coefficient of the coupled Ordinary Differential Equations
```

M(1,1):	− 0.0134	M(2,1):	0.0023	M(2,2):	−0.0088
M(3,1):	0.0023	M(3,3):	−0.0088	M(4,1):	0.0029
M(4,4):	−0.0077	M(5,1):	0.00285	M(5,5):	−0.0077
M(6,1):	0.0016	M(6,6):	−0.0103	M(7,1):	0.00156
M(7,7):	−0.0103	M(8,2):	0.0028	M(8,4):	0.0023
M(8,8):	0.0031	M(9,2):	0.0028	M(9,5):	0.0023
M(9,9):	−0.0031	M(10,2):	0.0016	M(10,7):	0.0023
M(10,10):	−0.0057	M(11,2):	0.0016	M(11,6):	0.0023
M(11,11):	−0.0057	M(12,3):	0.0029	M(12,4):	0.0023
M(12,12):	−0.0031	M(13,3):	0.0029	M(13,5):	0.0023
M(13,13):	−0.0031	M(14,3):	0.00156	M(14,6):	0.0023
M(14,14):	−0.0057	M(15,3):	0.00156	M(15,7):	0.0023
M(15,15):	−0.0057	M(16,4):	0.00156	M(16,6):	0.00285
M(16,16):	−0.0046	M(17,4):	0.00156	M(17,7):	0.00285
M(17,17):	−0.0046	M(18,5):	0.00156	M(18,6):	0.00285
M(18,18):	−0.0046	M(19,5):	0.00156	M(19,7):	0.00285
M(19,19):	−0.0046	M(20,8):	0.00156	M(20,10):	0.00285
M(20,16):	0.0023	M(21,8):	0.00156	M(21,11):	0.00285
M(21,17):	0.0023	M(22,9):	0.00156	M(22,10):	0.00285
M(22,18):	0.0023	M(23,9):	0.00156	M(23,9):	0.0016
M(23,11):	0.00285	M(23,19):	0.0023	M(24,12):	0.0016
M(24,14):	0.00285	M(24,16):	0.0023	M(25,12):	0.0016
M(25,15):	0.00285	M(25,17):	0.0023	M(26,13):	0.0016
M(26,14):	0.00285	M(26,18):	0.0023	M(27,13):	0.0016
M(27,15):	0.00285	M(27,19):	0.0023		

```
! Computation of transition state probabilities
do k=1,10
    t=(k-1)*dt
    do i=1,27
        do j=1,27
            q(i)= (A(i,j) + M(i,j)*dt) * P(k,j)
```

```
                    P(k+1, i)= P(k+1,i) + q(i)
                end do
            end do
        end do
        ! Results
        write(1,*)"Markov model of the benchmark system"
        write(1,*)""
        write(1,*)"Sort by time"
        write(1,*)""
        do k=1,5
            t=(k-1)*dt
            write(1,*)"t=",t
            sum=0
                do i= 1,27
                    sum= sum + P(k,i)
                write(1,*)"P(",i,")=", P(k,i)
                end do
        write(1,*)"sum=",sum
        write(1,*)""
        end do
        write(1,*)""
        write(1,*)"Systematically arrange by system status"
        write(1,*)""
        do i=1,27
            write(1,*)"n=", i
            do k=1,5
            write(1,*) P(k,i)
            end do
        write(1,*)""
        end do
        end
```

APPENDIX III

Timed Fault Tree generation from DFM Model

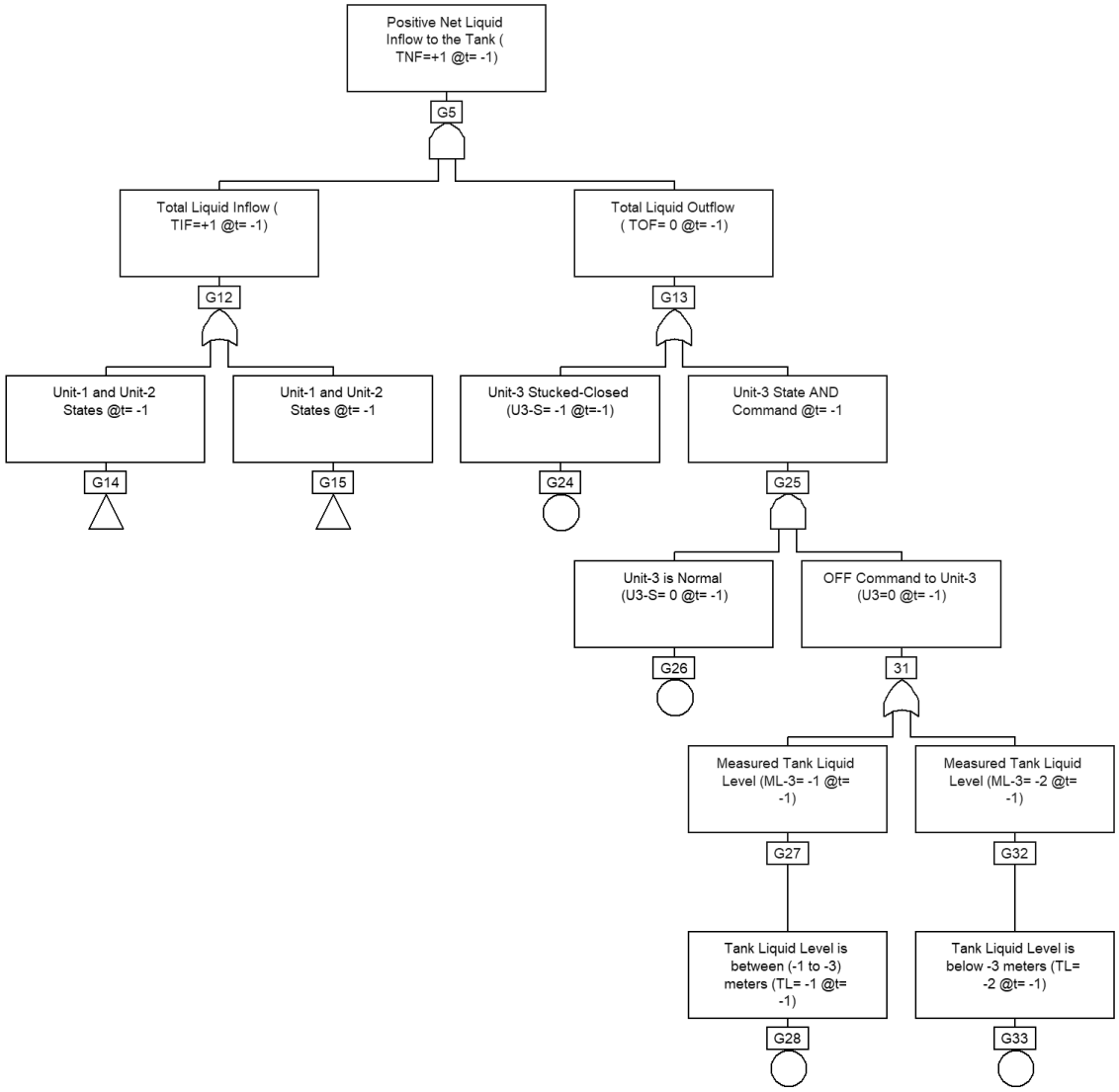


Figure A-III-1: Timed-fault tree for transfer gate G5

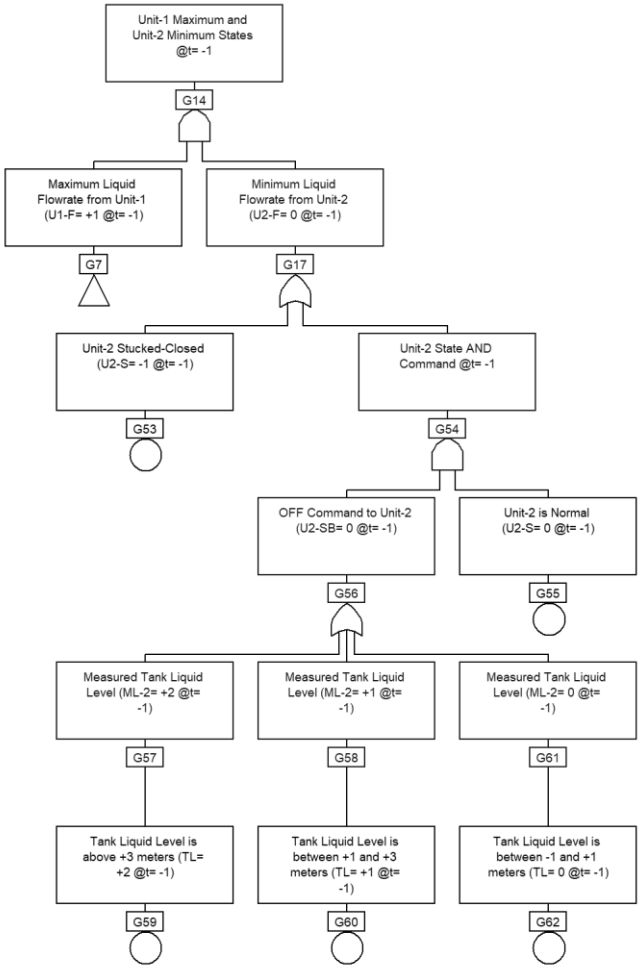


Figure A-III-2: Timed-fault tree for transfer gate G14

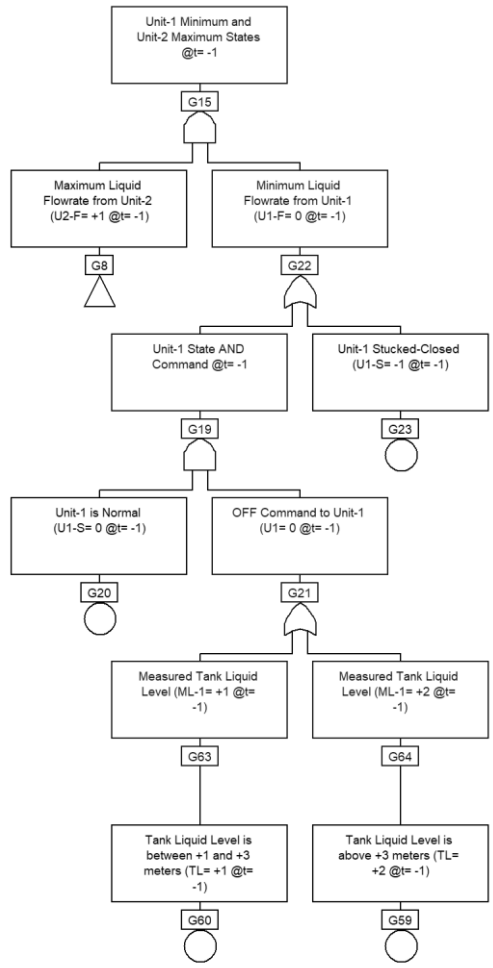


Figure A-III-3: Timed-fault tree for transfer gate G15

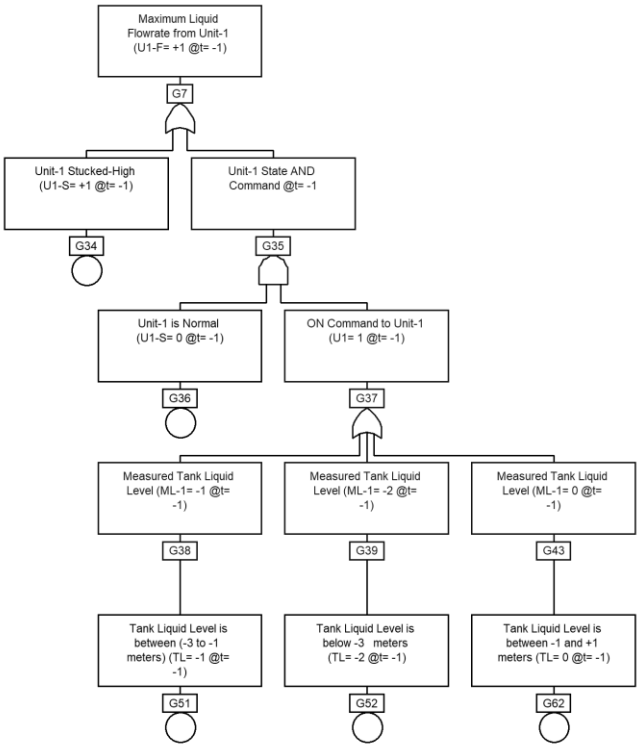


Figure A-III-4: Timed-fault tree for transfer gate G7

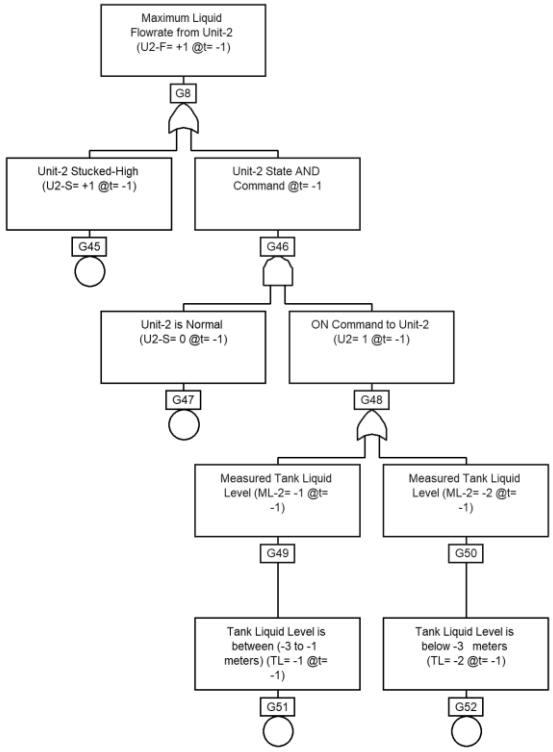


Figure A-III-5: Timed-fault tree for transfer gate G8

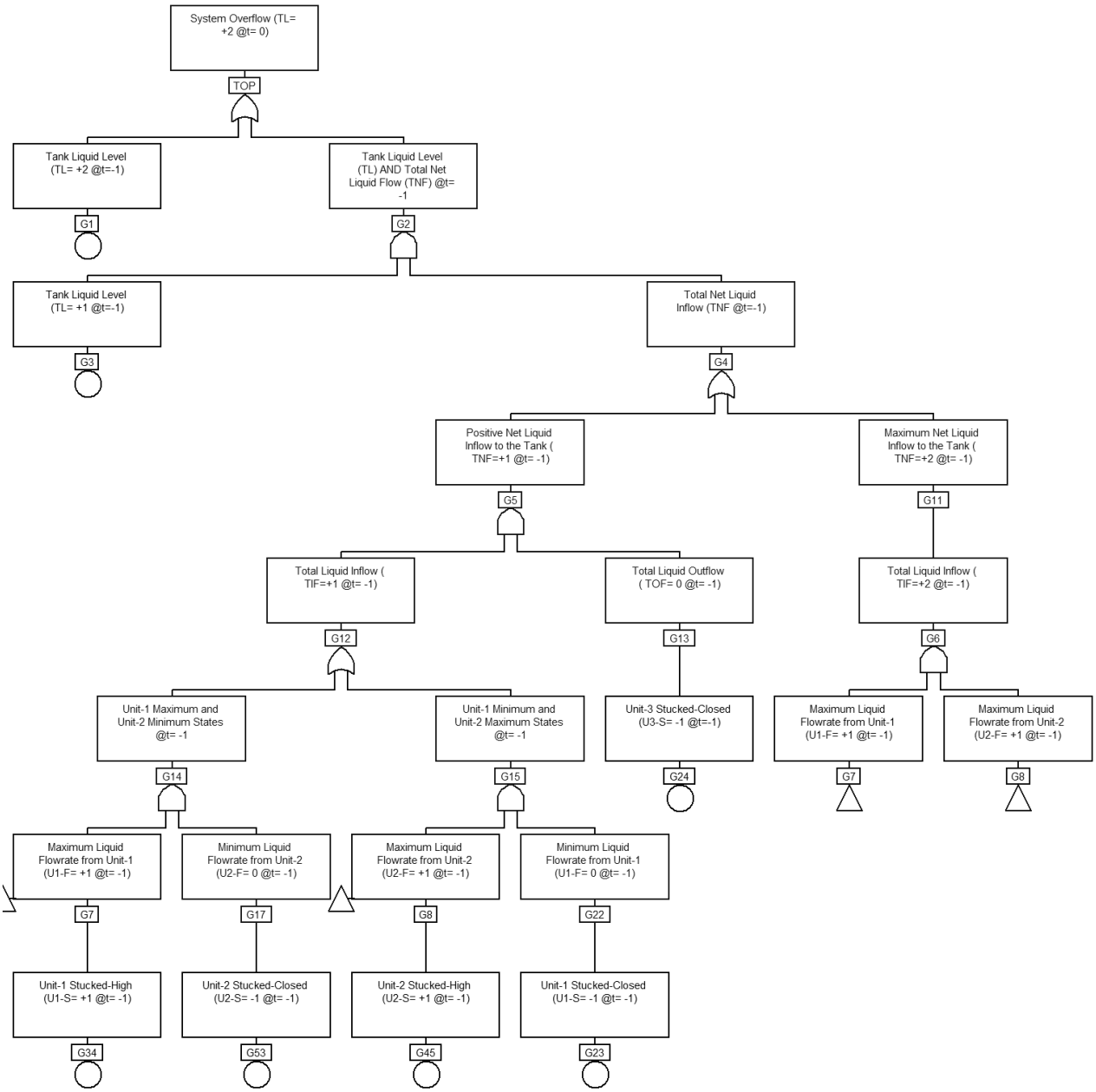


Figure A-III-6: Reduced timed-fault tree after consistency check

APPENDIX IV

State-space Discretization and computation of cell-to-cell transition probabilities.

```
clear all;
clc;

% State-Space Discretization Algorithm
syms VR Vj1 Vj2 Vj3 Vr1 Vr2 Vr3 r Vr x(t) dt f f1 z J1 J2 J3
VR_L= -3; % Control space lower bound
VR_U= +3; % Control space upper bound
VR= [-3:+3]; % Overall Control Space
VR= [-3:2:+3]; %Discretized control regions: analyst choice
for VR= -3:2:+3
        if (VR<=-3)
                Vr1_L=VR;
        elseif (VR<=-1)
                Vr1_U=VR;
                Vr2_L=Vr1_U;
        elseif (Vr2_L<VR<1)
                Vr2_U=VR;
                Vr3_L=Vr2_U;
        elseif (VR>=+3)
                Vr3_U=VR;
         end
end
r=[1 2 3]; %Number of control region
Vr1= [-3 -1]; % Space for control region-1
Vr2= [-1 +1]; % Space for control region-2
Vr3= [+1 +3]; % Space for control region-3
r1= range(Vr1); %Range of control region-1
cw1= range(Vr1)/3; %computational cell width in region-1
r2= range(Vr2); %Range of control region-2
cw2= range(Vr2)/3; %computational cell width in region-2
r3= range(Vr3); %Range of control region-3
cw3= range(Vr3)/3; %computational cell width in region-3
```

```
% Discretization scheme for Control region-1
a_bar= -3; % Lower bound of control space Vr1
alpha_1= -1; % Upper bound of control space Vr1
J1=3; % Number of sub-cells in control space Vr1
for j=1:J1
    delta_x1= ((alpha_1-a_bar)/J1); %sub-cell size/volume
          a(j)= a_bar + (j-1)*delta_x1; % Lower bound of the sub-cells
          b(j)= a_bar + j*delta_x1; % Upper bound of the sub-cells
       dp1(:,j)= ((a(:,j)+b(:,j))/2); %departure points from the sub-cells of control region-1
    Vd(j)=-3;
end
% Discretization scheme for Control region-2
alpha_1= -1; % Lower bound of control space Vr2
alpha_2= +1; % Upper bound of control space Vr2
J2=3; % Number of sub-cells in control space Vr2
for j=1:J2
          delta_x2= ((alpha_2-alpha_1)/J2); %sub-cell size/volume
          a(j)= alpha_1 + (j-1)*delta_x2; % Lower bound of the sub-cells
          b(j)= alpha_1 + j*delta_x2; % Upper bound of the sub-cells
          dp2(:,j)= ((a(:,j)+b(:,j))/2); %departure points from the sub-cells of control region-2
end
% Discretization scheme for Control region-3
alpha_2= +1; % Lower bound of control space Vr3
b_bar= +3; % Upper bound of control space Vr3
J3=3; % Number of sub-cells in control space Vr3
for j=1:J3
          delta_x3= ((b_bar-alpha_2)/J3); %sub-cell size/volume
          a(j)= alpha_2 + (j-1)*delta_x3; % Lower bound of the sub-cells
          b(j)= alpha_2 + j*delta_x3; % Upper bound of the sub-cells
          dp3(:, j)= ((a(:,j)+b(:,j))/2); %departure points from the sub-cells of control region-3
          Vo(j)=+3;
end
```

```
DP= [dp1; dp2; dp3]; %Departure points from all the sub-cells of the three regions
x(t)=DP;
%System dynamics is defined by (dx/dt)=f
x1_bar= 1; %flowrate from unit-1
x2_bar= 1; %flowrate from unit-2
x3_bar= 1; %flowrate from unit-3
N=8; % component state combination
for n=1:8;
        for dt=1 %the time step
        f= (x1_bar + x2_bar - x3_bar)*dt; %Liquid level evolution as a function of N
                if n==1 % system at component state combination n=1, 4 and 6
                        z1=f; %z1 is the "dx" i.e., the small increment in level
                        y1=[x(t)+z1]; % y1= x(t+dt) is the liquid level in the next time step
                        y11= vpa(y1);
                else if n==2 % system state at n=2
                        z2=2*f; %The net flowrate at n=2
                        y2=[x(t)+z2]; % y2=x(t+dt) is the liquid level in the next time step
                        y22= vpa(y2); %Fraction to decimal
                elseif n==3 % system state at n=3, 5 and 8
                        z3=0;
                        y3=[x(t)+z3];
                        y33=vpa(y3);
                elseif n==7 % system state at n=7
                        z4=-f;
                        y4=[x(t)+z4];
                        y44=vpa(y4);
                        end
                end
        end
end
```

```
p=3; %number of rows and column of y11, y22, y33 and y4
B=zeros(5,5);
B(1,1)=-5;
        for i=1:5
                B(5,i)=+5;
        end
        for i=1:p
                for j=1:p
                B(i+1,j+1)=y11(i,j);
                end
        end
J= 5; %number of discretized space
h=5; % number of groups/control range
k(J,h)=zeros; % k is a group matrix
% Each row and column or elements of matrix 'k' represents the number of arrival trajectories in j from j'
        for j=1:J %Number of columns of g(j/j',n',dt)
                for i=1:J %Number of rows g(j/j',n',dt)
                        if B(i,j)<-3
                        k(i,1)=k(i,1)+1;
                        end
                        if B(i,j)>=-3 && B(i,j)<=-1
                        k(i,2)=k(i,2)+1;
                        end
                        if B(i,j)>-1 && B(i,j)<=+1
                        k(i,3)=k(i,3)+1;
                        end
                        if  B(i,j)>+1 && B(i,j)<=+3
                        k(i,4)=k(i,4)+1;
                        end
                        if B(i,j)>+3
                        k(i,5)=k(i,5)+1;
                end
        end
end
```

APPENDIX V

Sample Results: State Transition Probabilities of the Passive Isolation Condenser Systems

Transition State Probabilities for $k = 5$:

$P_1 = 0.997$	$P_2 = 1.37E\text{-}03$	$P_3 = 1.59E\text{-}07$
$P_4 = 1.75E\text{-}10$	$P_5 = 2.621E\text{-}05$	$P_6 = 2.88E\text{-}08$
$P_7 = 3.36E\text{-}12$	$P_8 = 2.77E\text{-}15$	$P_9 = 2.62E\text{-}06$
$P_{10} = 2.87E\text{-}09$	$P_{11} = 3.36E\text{-}13$	$P_{12} = 2.77E\text{-}16$
$P_{13} = 0.00$	$P_{14} = 4.44E\text{-}18$	$P_{15} = 5.19E\text{-}22$
$P_{16} = 5.689E\text{-}25$	$P_{17} = 8.51E\text{-}20$	$P_{18} = 9.33E\text{-}23$
$P_{19} = 1.09E\text{-}26$	$P_{20} = 6.73E\text{-}30$	$P_{21} = 8.51E\text{-}21$
$P_{22} = 9.33E\text{-}24$	$P_{23} = 1.09E\text{-}27$	$P_{24} = 6.73E\text{-}31$
$P_{25} = 6.02E\text{-}04$	$P_{26} =6.61E\text{-}07$	$P_{27} = 7.73E\text{-}11$
$P_{28} = 6.36E\text{-}14$	$P_{29} = 1.27E\text{-}08$	$P_{30} = 1.04E\text{-}11$
$P_{31} = 1.22E\text{-}15$	$P_{32} = 6.68E\text{-}19$	$P_{33} = 1.27E\text{-}09$
$P_{34} = 1.04E\text{-}12$	$P_{35} = 1.22E\text{-}16$	$P_{36} = 6.68E\text{-}20$
$P_{37} = 6.02E\text{-}04$	$P_{38} = 6.61E\text{-}07$	$P_{39} = 7.73E\text{-}11$
$P_{40} = 6.36E\text{-}14$	$P_{41} = 1.27E\text{-}08$	$P_{42} = 1.04E\text{-}11$
$P_{43} = 1.22E\text{-}15$	$P_{44} = 6.68E\text{-}19$	$P_{45} = 1.27E\text{-}09$
$P_{46} = 1.04E\text{-}12$	$P_{47} = 1.22E\text{-}16$	$P_{48} = 6.68E\text{-}20$
$P_{49} = 1.95E\text{-}18$	$P_{50} = 2.14E\text{-}21$	$P_{51} = 2.51E\text{-}25$
$P_{52} = 1.55E\text{-}28$	$P_{53} = 4.11E\text{-}23$	$P_{54} = 2.54E\text{-}26$
$P_{55} = 2.97E\text{-}30$	$P_{56} = 8.67E\text{-}34$	$P_{57} = 4.11E\text{-}24$
$P_{58} = 2.54E\text{-}27$	$P_{59} = 2.97E\text{-}31$	$P_{60} = 8.67E\text{-}35$
$P_{61} = 2.91E\text{-}07$	$P_{62} = 2.39E\text{-}10$	$P_{63} = 2.80E\text{-}14$
$P_{64} = 1.54E\text{-}17$	$P_{65} = 4.59E\text{-}12$	$P_{66} = 2.52E\text{-}15$
$P_{67} = 2.95E\text{-}19$	$P_{68} = 8.07E\text{-}23$	$P_{69} = 4.59E\text{-}13$
$P_{70} = 2.52E\text{-}16$	$P_{71} = 2.95E\text{-}20$	$P_{72} = 8.07E\text{-}24$
$Sum = 1.00$		

APPENDIX VI

Dynamic Probabilistic Safety Assessment using Dynamic Flowgraph Method and Markov-Cell-to-cell Mapping Technique

Chireuding Zeliang
MASc Candidate

Supervisor:
Prof. Akira Tokuhiro
(Prof. Lixuan Lu)

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

24th July 2018

Acknowledgements

- My sincere gratitude to **Prof. Akira Tokuhiro**
- My appreciation to the Committee members: **Prof. Glenn Harvel and Dr. Salam K. Ali**
- **Dr. Hadid Subki**, Nuclear Power Technology Development Section, International Atomic Energy Agency (**IAEA**)

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Objectives

- Demonstrate the advantages and applicability of dynamic PSA methods over classical techniques
- Formulate the coupling of Markov-CCMT model for dynamic PSA
- Illustrate generation of timed-fault trees from DFM model for its integration into an existing classical PSA model
- Development of a novel approach for integrated reliability assessment of passive safety system in iPWR-type Small Modular Reactors (IAEA-UOIT Coordinated Research Project)

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Introduction

- Classical Probabilistic Risk Assessment (PRA)
  - Event/Fault Tree techniques
  - Classical Markov chain
- Dynamic PRA
  - Dynamic Flowgraph Method
  - Coupled Markov/Cell-to-cell Mapping Technique
- Application and demonstration of methodologies
- Development of an integrated approach for Dynamic Reliability Assessment of passive safety systems in iPWR-type SMRs

NUREG-6901: "Current State of Reliability Modeling Methodologies for Digital Systems and Their Acceptance Criteria for Nuclear Power Plant Assessments" *(2016)*

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Classical PSA techniques

- Well-established techniques, and a systematic and comprehensive approach for Risk Assessment
  - Fault/Event Tree Technique ***(WASH-1400, NUREG-75/014, 1975)***
  - Classical Markov chain ***(Norris, 1997)***
- *Static logic* based technique
- Fault trees are series of Boolean Algebraic equations to determine the failure probability of an undesired event

- *WASH-1400, NUREG-75/014, "Reactor Safety Study: An assessment of accident risks in U.S. Commercial Nuclear power plants", US Nuclear Regulatory Commission, 1975.*
- *Norris, J. R., "Continuous-time Markov chains I and II", 1997.*

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Limitations of Classical PSA

- Classical PRA techniques lacks the ability to account for *system dynamics*
- Lacks the ability to capture *dynamic interactions*
  - Component failure modes and probability as a function of state variables (e.g., interactions of control units through liquid level)
- Classical techniques are neither developed nor intended to capture *time element*
- *States ordering in ET/FT* are fixed/no particular ordering index
- FT (*binary logic*) has limited capability in *multi-state modelling*

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Dynamic PSA

- Dynamic PRA techniques enables one to *couple system model with risk analysis code (as early as 1986)*
- *Simultaneously accounting* for system time-dependent phenomenological model and its stochastic behavior
- *Explicit* consideration and treatment of time-element
- A better *treatment of dynamic interactions* and scenario development
  - For systems with multiple top events, the final state is dependent on magnitude of state variables, order and timing of failure events
- Allows for systematic and more *complete coverage of possible failure scenarios*
- *State-space explosion phenomenon*, highly time-dependent, modelling complexity and computational time requirements

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Dynamic Flowgraph Method (DFM)

- An integrated approach to modelling system logical and time-dependent behavior
- Multi-valued logic model
- Dynamic behaviors are represented as a series of discrete state transitions
- A single model can analyze arbitrary number of possible system conditions (i.e., any top event of interest)
- A three step process:
  1. Model development
  2. Model analysis (deductive/inductive)
  3. Quantification of the prime implicants.

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Prime Implicants

- A prime implicant is a *conjunction of basic events* that is sufficient to cause a top event, but does not contain any shorter conjunction of events that is sufficient to cause the top event
- Analogous to minimal cut-set in FT, but is time-stamped (*e.g., A_i @t=-1: component 'A' is in state "i" at time "t=-1"*)

  <Variable A=2 *@t=-1* ***AND*** Variable A=3 *@t=-2*>
- Complete base is the set of prime implicants logically equivalent to the FT TOP event function
- Complete base is obtained by evaluation of transition table using algebraic laws including:
  - Absorption, Merging, Reduction and Absorption-Merging law (*Yau, 1997*)
  - Generalized Consensus Method *(Garribba et al., 1985)*

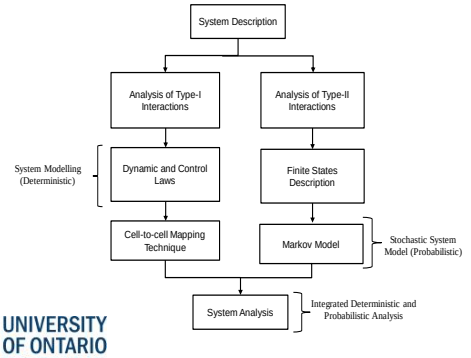
UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Timed FT generation from DFM model

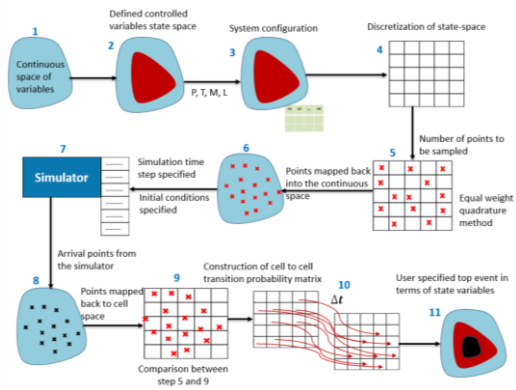
- *Integration of results* obtained from DFM model to an existing classical plant PRA model
- *Sequences of static FTs* at different time steps representing evolution of logical combinations of events leading to a top event
- Consistency Rules:
  - *Physical consistency*: A process variable with different state cannot occur in the same time step. e.g., valve 'A' open AND valve 'A' closed @t= -1
  - *Dynamic consistency*: A process variable can change its states in a certain direction or by amount (dictated by system dynamics) in a single time step. e.g., liquid level variation from 0% to 90% in a single time step

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Coupled Markov-CCMT Model



UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY



The Algorithm

- The probability $P_{n,j}(t)$ of finding a state variable in cell V_j at time $t=k\Delta t$ for a given component state combination 'n' can be recursively found:

$$P_{n,j}(k+1) = \sum_{n'=1}^{N} \sum_{j'=1}^{J} g(j/j', n', \Delta t).\, h(n/n', j' \to j, \Delta t).\, P_{n',j'}(k)$$

$$P_{n,j}(k+1) = \sum_{n'=1}^{N} \sum_{j'=1}^{J} q_{n,j}^{n',j'}.\, P_{n',j'}(k)$$

$$q_{n,j}^{n',j'} = g(j/j', n', \Delta t).\, h(n/n', j' \to j, \Delta t)$$

$\mathbf{g(j/j', n', k\Delta t)}$: Conditional probability of state variables in cell j at time $(k+1)\Delta t$ given that it was in cell j' at time t

$\mathbf{h(n/n', j' \to j, k\Delta t)}$: Conditional probability of component state combination in state n at time $(k+1)\Delta t$ given that it was in state n' at time $k\Delta t$, and the state variables move from cell j' to cell j during $k\Delta t \le t \le (k+1)\Delta t\}$

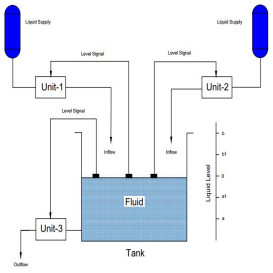
UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Application of Methodologies

1. Fault Tree
2. Event Tree
3. Classical Markov model
4. Dynamic Flowgraph Method
5. Markov-CCMT model

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

The Benchmark System (BS)



Control Laws

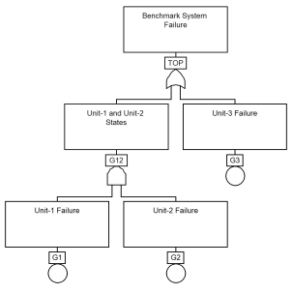
Liquid Level (x)	Control Unit State		
	Unit-1	Unit-2	Unit-3
$x > b$	-	-	-
$b_1 < x$	OFF	OFF	ON
$a_1 \le x \le b_1$	ON	OFF	ON
$x < a_1$	ON	ON	OFF
$x < a$	-	-	-

Unit discrete states (i)

- ON/OFF
- Failed-open
- Failed-closed

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Fault Tree Model: Binary

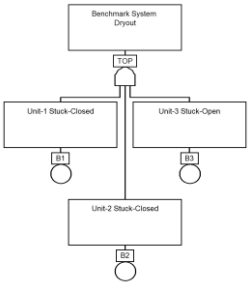


Top Event	
MCS	$G_1G_2 + G_3$
Failure Probability	3.1511×10^{-3}

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Ref: Electric Power Research Institute, "CAFTA Fault Tree Analysis System Version 6.0", EPRI Inc., NC, 2013.

Fault Tree Model: Multiple Modes

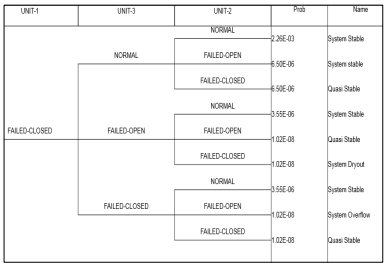


Top Event: Overflow	
MCS	$B_1B_2 + B_1B_3 + B_2B_3$
Failure Probability	1.4555×10^{-05}

Top Event: Dryout	
MCS	$B_1B_2B_3$
Failure Probability	1.0192×10^{-08}

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Event Tree Model



System Dryout
3.06×10^{-08}

System Overflow
2.91×10^{-05}

Note: Rounded to two decimal point for simplicity

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Ref: Electric Power Research Institute, "CAFTA Fault Tree Analysis System Version 6.0", EPRI Inc., NC, 2013.

Classical Markov model

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

19

Markov chain

- Enables one to model a sequence of random variables which corresponds to discrete system states
- System state at any point in time is dependent only on the previous state, i.e., conditional state probabilities

$$\mathbb{P}(\boldsymbol{X_{n+1}} = \boldsymbol{j} | \boldsymbol{X_n} = \boldsymbol{i}, \boldsymbol{X_{n-1}} = \boldsymbol{i_{n-1}}, \dots, \boldsymbol{X_0} = \boldsymbol{i_0}) = \mathbb{P}(\boldsymbol{X_{n+1}} = \boldsymbol{j} | \boldsymbol{X_n} = \boldsymbol{i})$$

- System dynamics represented by a set of coupled linear differential equations (*Chapman-Kolmogorov*)

$$\frac{dP_i(t)}{dt} = -\sum_{\substack{j=1 \\ j \neq i}}^{n} a_{ij} P_i(t) + \sum_{\substack{j=1 \\ j \neq i}}^{n} a_{ji} P_j(t)$$

- State probabilities can be determined by solving the set of coupled ordinary differential equations

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

20

Enumeration of system states

$N = (\textbf{Possible failure modes})^{(\textbf{number of units})}$

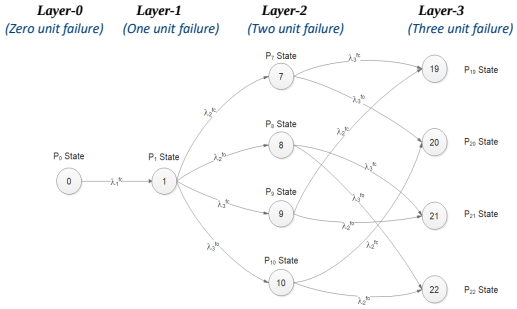
$N = 27$ discrete system states

Individual Unit States			System States (n)
Unit-1	Unit-2	Unit-3	
$n_1 = 0$	$n_2 = 0$	$n_3 = 0$	$P_0(t)$
$n_1 = 1$	$n_2 = 0$	$n_3 = 0$	$P_1(t)$
:	:	:	:
:	:	:	:
$n_1 = 2$	$n_2 = 2$	$n_3 = 2$	$P_{26}(t)$

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

21

Markov state transition diagram



UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

22

Transition State Probabilities

$$\frac{dP_0(t)}{dt} = -(\lambda_1^{fc} + \lambda_1^{fo} + \lambda_2^{fc} + \lambda_2^{fo} + \lambda_3^{fc} + \lambda_3^{fo}) P_0(t)$$

$$\frac{dP(t)}{dt} = A.P(t)$$

$$P(t + \Delta t) = (I + A.\Delta t).P(t)$$

Initial conditions:

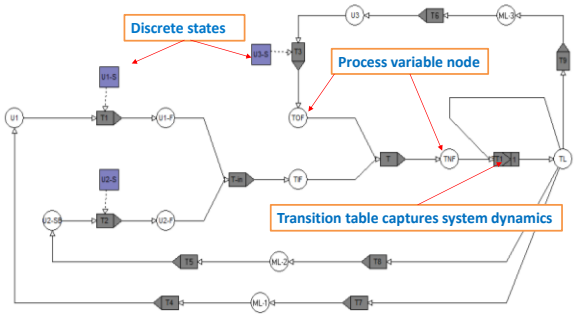
$$P_n(t = 0) = \begin{cases} 1; \ for \ n = 0 \\ 0; \ for \ n \neq 0 \end{cases}$$

Sample Results ($k = 3$)	
System overflow:	8.6805×10^{-05}
System dryout:	6.1155×10^{-08}

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

23

DFM Model of the Benchmark System



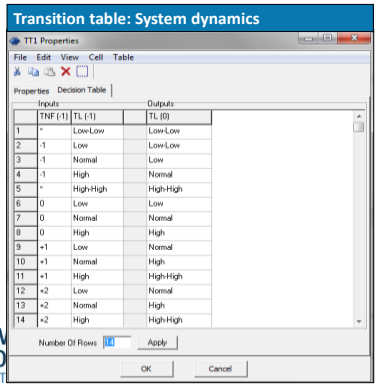
OF ONTARIO INSTITUTE OF TECHNOLOGY

Ref: DYMONDA 7.0, ASCA Inc., USA, 2013

24

Transition and Decision Table

Transition table: System dynamics



Prime Implicants for 'system overflow'

#	Prime Implicants	Time	Logic	Probability
1.	Tank Liquid Level was at +3 meters	@t= -1		-
2.	Tank Liquid Level was between +1 and +3 meters	@t= -1	AND	6.5232×10^{-6}
	Unit-1 stucked-open	@t= -1	AND	
	Unit-3 stucked-closed	@t= -1		
3.	Tank Liquid Level was between +1 and +3 meters	@t= -1	AND	4.4643×10^{-6}
	Unit-2 stucked-open	@t= -1	AND	
	Unit-3 stucked-closed	@t= -1		
4.	Tank Liquid Level was between +1 and +3 meters	@t= -1	AND	3.5673×10^{-6}
	Unit-1 stucked-open	@t= -1	AND	
	Unit-2 stucked-open	@t= -1		
	Top Event Probability			1.45548×10^{-5}

PIs for 'system overflow': 2 backtracking depth

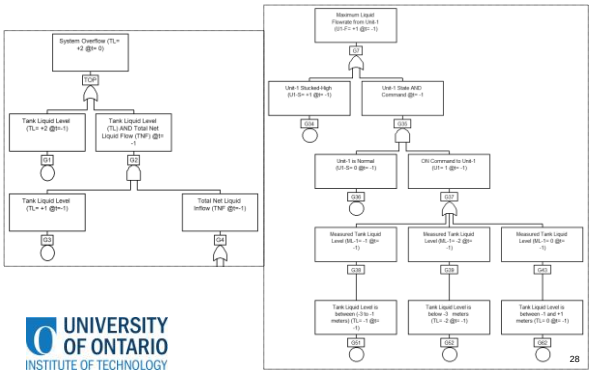
State variable evolution depicting system dynamics

#	Prime Implicant	Time	Logic
1.	Tank Liquid Level was between -1 and +1 meters	@t= -2	AND
	Unit-1 is Normal	@t= -2	AND
	Unit-2 Stucked-High	@t= -2	AND
	Unit-3 is Normal	@t= -2	AND
	Unit-3 Stucked-Low	@t= -1	
2.	Tank Liquid Level was between +1 and +3 meters	@t= -2	AND
	Unit-1 Stucked-High	@t= -2	AND
	Unit-3 is Normal	@t= -2	AND
	Unit-3 Stucked-Low	@t= -1	

State ordering/sequence

Scenario history

Generation of timed fault tree



Final MCS via Backtracking process

For 5 discretized intervals

#	Minimal cut-sets	Logic
1.	TL= +2 @t= –1	OR
2.	(TL= +1 AND U1-S= +1 AND U2-S= +1) @t= –1	OR
3.	(TL= +1 AND U1-S= +1 AND U3-S= –1) @t= –1	OR
4.	(TL= +1 AND U2-S= +1 AND U3-S= –1) @t= –1	

Markov-CCMT Model of the BS

- State space discretized into 5 cells, ($J = 5$)
  - 3 computational cells
  - 2 sink cells
- Computational cells are further discretized into 3 sub-cells
- Total number of system states, ($N = 27$)
- $N \times J = 135$; $q_{n,j}^{n',j'}(k\Delta t) = 135 \times 135$ square matrix
- Sink cells:

$$\text{Top Event} = \begin{cases} j = 5; & Overflow \\ j = 1; & Dryout \end{cases}$$

Predicted system state probabilities

Time step	$P_{n,1}(k\Delta t)$	$P_{n,5}(k\Delta t)$	$P_{OD}(k\Delta t)$
$k=1$	0	0	0
$k=2$	0	1.019×10^{-08}	1.019×10^{-08}
$k=3$	1.857×10^{-07}	9.173×10^{-05}	9.192×10^{-05}
$k=4$	2.1496×10^{-06}	5.114×10^{-04}	5.135×10^{-04}

n = System states/configuration
j = Number of discretized state variables
OD = Total failure probability (Overflow + Dryout)

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

IAEA-UOIT Coordinated Research Project (*iPWR-type Small Modular Reactors*)

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

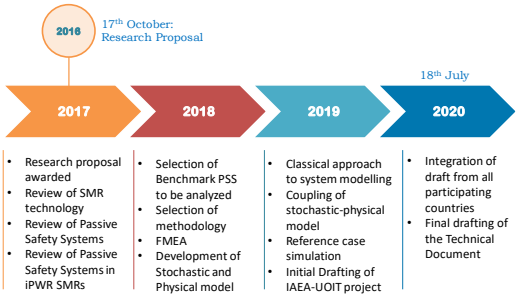
IAEA-UOIT Coordinated Research Project

Research proposal made:	**17th October 2016**
Technical contract awarded:	**13th October 2017**
Title:	Application of DFM and FTA for Reliability and Risk Assessments of Passive Safety System in an Integral-PWR type SMRs

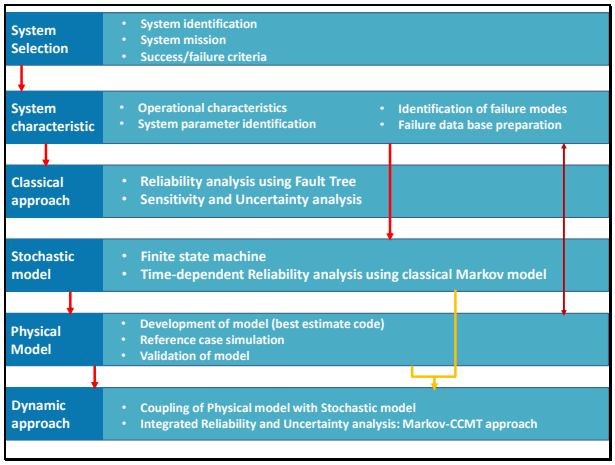
- Presentations made at two (2) IAEA technical meetings:
  1. First Technical Meeting *(30th Oct.- 2nd Nov. 2017)*
     - Passive Safety Features in iPWR-type SMRs:
     - Overview and Planning of Technical Contract No. 21051
  2. Second Technical Meeting *(7-10th May 2018)*
     - Planning and Progress of the Research Project

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

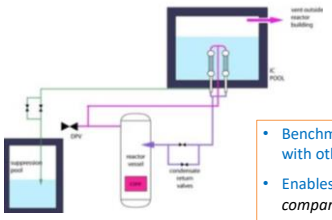
Preliminary CRP Roadmap



UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY



The Benchmark Isolation Condenser System



- Benchmark system is selected in *alignment* with other participating countries
- Enables one to *validate the model* and *compare* the predicted system reliability
- Some generic *experimental data* available *(Argentina and India)*

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Dynamic PRA for SMRs

- Dynamic PRA can be more suitable for integral pressurized water reactor type SMRs:
  - Simplified Design
  - Reduced number of components (e.g., No reactor coolant pumps)
  - Reduced number of accident initiators (e.g., No large brake loss-of-coolant accident)
  - Less reliance on active power source

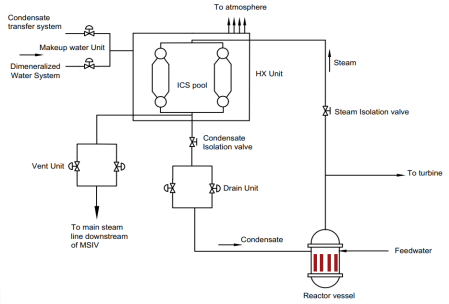
UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

ICS Mission

- Primary objectives of the ICS includes:
  - Reactor pressure vessel depressurization without the loss of primary coolant inventory
  - Remove sensible and decay heat from the reactor for 72 hours (mission time)
  - Maintain fuel peak cladding temperature within design limits (1200 ^{o}C typically for PWR).
- Advantages of implementing ICS includes:
  - Prevents unnecessary activation of safety relief valves (SRVs)
  - Eliminate/mitigate potential LOCA that may result from SRVs failure
  - Potential elimination/delay of high pressure coolant injection system operation

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

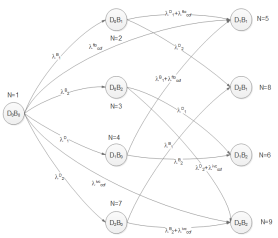
Benchmark System Characterization



OF ONTARIO INSTITUTE OF TECHNOLOGY

Condensate Drain Unit: Sample

State Transition Diagram



Unit state ordering

Unit State Combination		CCF of the valves	Unit States (n)
Main valve states	Bypass valve states		
$n_1 = 0$	$n_2 = 0$	-	1
$n_1 = 0$	$n_2 = 1$	-	2
$n_1 = 0$	$n_2 = 2$	-	3
$n_1 = 1$	$n_2 = 0$	-	4
$n_1 = 1$	$n_2 = 1$	CCF	5
$n_1 = 1$	$n_2 = 2$	-	6
$n_1 = 2$	$n_2 = 0$	-	7
$n_1 = 2$	$n_2 = 1$	-	8
$n_1 = 2$	$n_2 = 2$	CCF	9

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Predicted State Transition Probabilities

- The analysis shows that $P_1(t)$ has the highest state transition probability. This implies that the ICS is mostly likely to fail due to an envelope failure/pressure boundary being compromised
- $P_1(t)$ is followed by $P_{24}(t)$ and $P_{36}(t)$ which are the vent and drain unit failure respectively. This can be due to the redundancy provided for the vent and drain unit
- $P_{36}(t)$ being lower than $P_{24}(t)$ is due to the diversity provided in the drain valves as compared to the identical vent valves.

CNSC, REGDOC- 2.5.2: Safety system failure probability requirement: $< \mathbf{10^{-03}}$ *(2014)*

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Conclusion

- Dynamic techniques can provide useful insight and significant information in risk assessment compared to classical techniques
- Detail analysis of the benchmark system shows that dynamic techniques can capture:
  - System dynamics
  - Time element
  - State ordering
  - A more realistic treatment of system/components with multiple top events and failure modes respectively
- Markov-CCMT model provide a more rigorous treatment of time element (real-time) or tightly coupled
- DFM have the advantage of providing a scenario history and probabilistic accident sequence evolution

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Conclusions (cont')

- The issue of state space explosion can be systematically treated by:
  - System state merging
  - Hybrid approach (small FT and Large Markov model)
  - Re-arranging and solving the matrix in Canonical form
- Classical approach yields conservative results, and is suitable for identifying logical correctness and establishing conservative relationships between top and basic events
- The integrated approach presented in this research provide a framework for dynamic risk assessment of passive safety systems that has never been employed.

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Future Works

- Development of *ICS physical model* (deterministic model) using best-estimate system code RELAP5
- *Mechanized coupling* of stochastic and deterministic system model through a user-friendly computer code.
  e.g., coupling of stochastic model (*Fortran95*) and physical model (*RELAP5*) in *Python* code
- *Integrated modelling* of automatic depressurization system (ADS) and ICS to capture the dynamic interactions, and their influence on demand frequency and predicted failure probability of the two systems

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY

Thank You

"Towards an Integrated Risk Assessment"

@C.Zeliang, 2018

UNIVERSITY OF ONTARIO INSTITUTE OF TECHNOLOGY