Discrete-Time Bayesian Networks Applied to Reliability of Flexible Coping Strategies of Nuclear Power Plants

Elvan Sahin

Thesis submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Master of Science

in

Nuclear Engineering

Juliana P. Duarte, Chair

Yang Liu

Mark Pierson

May 7, 2021

Blacksburg, Virginia

Keywords: FLEX,Probabilistic Risk Assessment, Bayesian Networks.

Discrete-Time Bayesian Networks Applied to Reliability of Flexible Coping Strategies of Nuclear Power Plants

Elvan Sahin

(ABSTRACT)

The Fukushima Daiichi accident prompted the nuclear community to find a new solution to reduce the risky situations in nuclear power plants (NPPs) due to beyond-design-basis external events (BDBEEs). An implementation guide for diverse and flexible coping strategies (FLEX) has been presented by Nuclear Energy Institute (NEI) to manage the challenge of BDBEEs and to enhance reactor safety against extended station blackout (SBO). To assess the effectiveness of FLEX strategies, probabilistic risk assessment (PRA) methods can be used to calculate the reliability of such systems. Due to the uniqueness of FLEX systems, these systems can potentially carry dependencies among components not commonly modeled in NPPs. Therefore, a suitable method is needed to analyze the reliability of FLEX systems in nuclear reactors. This thesis investigates the effectiveness and applicability of Bayesian networks (BNs) and Discrete-Time Bayesian Networks (DTBNs) in the reliability analysis of FLEX equipment that is utilized to reduce the risk in nuclear power plants. To this end, the thesis compares BNs with two other reliability assessment methods: Fault Tree (FT) and Markov chain (MC). Also, it is shown that these two methods can be transformed into BN to perform the reliability analysis of FLEX systems. The comparison of the three reliability methods is shown and discussed in three different applications. The results show that BNs are not only a powerful method in modeling FLEX strategies, but it is also an effective technique to perform reliability analysis of FLEX equipment in nuclear power plants.

Discrete-Time Bayesian Networks Applied to Reliability of Flexible Coping Strategies of Nuclear Power Plants

Elvan Sahin

(GENERAL AUDIENCE ABSTRACT)

Some external events like earthquakes, flooding, and severe wind, may cause damage to the nuclear reactors. To reduce the consequences of these damages, the Nuclear Energy Institute (NEI) has proposed mitigating strategies known as FLEX (Diverse and Flexible Coping Strategies). After the implementation of FLEX in nuclear power plants, we need to analyze the failure or success probability of these engineering systems through one og the existing methods. However, the existing methods are limited in analyzing the dependencies among components in complex systems. Bayesian net- works (BNs) are a graphical and quantitative technique that is utilized to model dependency among events. This thesis shows the effectiveness and applicability of BNs in the reliability analysis of FLEX strategies by comparing it with two other reliability analysis tools, known as Fault Tree Analysis and Markov Chain. According to the reliability analysis results, BN is a powerful and promising method in modeling and analyzing FLEX strategies.

Dedication

To my dear family and lovely wife.

Acknowledgments

First of all, I would like to thank the Ministry of National Education of the Turkish Government for offering me this chance with the scholarship I was granted. With this scholarship, I could have the opportunity to broaden my horizon socially and academically. I would like to thank my advisor and mentor throughout my graduate studies, Professor Juliana Duarte. Her guidance and support have helped me to better realize my potential academically, and she has always been willing to put in the effort to assist me with any problems I might be having while still allowing this thesis to be my own work. I would like to thank Victor C. Leite, who assisted me in understanding the general idea of the work as well as creating the structure of Matlab code. I would like to thank my committee members, Dr. Yang Liu and Dr. Mark Pierson, for their time and contributions. Finally, I must express gratitude to my family and friends for providing me with unfailing support and continuous encouragement throughout my years of study. This accomplishment would not have been possible without them. Thank you.

Contents

List of Figures

List of Tables

List of Abbreviations

λ The failure rate of a component

F Fail

n Time Granularity

Q Unreliability

R Reliability

T and t Mission Time

W Work

$\triangle$ Interval Length

AC Alternating Current

AOP Abnormal Operating Processes

BBN Bayesian Belief Network

BDBEE Beyond-Design-Basis External Event

BN Bayesian Network

BNCC Bayesian Network with Causality Constraint

CBDTM Cause-Based Decision Tree Method

CDF Core Damage Frequency

CP Conditional Probability

CPD Conditional Probability Distribution

CPT Conditional Probability Table

CS Cut Set

CSP Cold Spare Gate

DAG Directed Acyclic Graph

DE Dependent Event

DFT Dynamic Fault Tree

DTBN Discrete-Time Bayesian Network

ELAP Extended Loss of Coolant Accident

EOP Emergency Operating Processes

ET Event Tree

FLEX Diverse and Flexible Strategies

FTA Fault Tree Analysis

GTG Gas Turbine Generator

HEP Human Error Probability

LOCA Loss of Coolant Accident

MC Markov Chain

MCS Minimal Cut Set

MTBF Mean Time Between Failures

MTTR Mean Time To Repair

NEI Nuclear Energy Institute

NLP Natural Language Processing

NPP Nuclear Power Plant

NRC Nuclear Regulatory Commission

PIF Performance Influencing Factor

PRA Probabilistic Risk Analysis

RABN Reliability Assessment through Bayesian Network

SBO Station Blackout

SEFT State-Event Fault Tree

SFT Static Fault Tree

TE Top Event

THERP Technique for Human Error Prediction

Chapter 1

Introduction

After the accident at the Fukushima Daiichi nuclear power plant, the nuclear community faced new challenges to mitigate the consequences of Beyond-Design-Basis External Events (BDBEEs) [11]. Implementation guidelines for Diverse and Flexible Coping Strategies (FLEX) have been introduced by the Nuclear Energy Institute (NEI) in order to better respond to the challenge conditions of BDBEEs and to improve the reactor safety when extended events, such as station blackout (SBO) occurs in the reactor.

FLEX systems include mitigating strategies to reduce risk situations due to BDBEEs. The main purpose of utilizing FLEX is to develop an indefinite coping capacity to minimize the risk of damage to the fuel in the reactor core and preserve the reactor containment by utilizing both on-site and off-site mobile devices. To reduce the accident consequences in nuclear reactors, the risk assessment of FLEX strategies needs to be performed by utilizing reliability analysis tools. However, there are some challenges in quantifying the risk assessment of FLEX strategies: human-based errors, a dependency between system components, and having large and complex systems.

Risk analysis for nuclear power plants has been discussed for more than four decades since the release of the first study published in 1975 [9]. An essential method established during this period was the Probabilistic Risk Analysis (PRA) for the study of core meltdown accidents in 1990 [7]. According to the U.S. Nuclear Regulatory Commission (NRC), PRA is utilized to estimate risk by calculating real numbers to identify what can fail, what is the probability of

this failure, and what are the consequences. Therefore, PRA gives awareness of the strengths and weaknesses of nuclear reactor safety. In a nuclear power plant, the method consists of evaluating three levels of risk. Level 1 evaluates core damage frequency (CDF). Level 2 evaluates the radioactivity release from a nuclear reactor containment. Level 3 evaluates the damage to people and the environment. A PRA example is displayed in Fig. 1.1, and PRA levels are shown in Fig. 1.2.

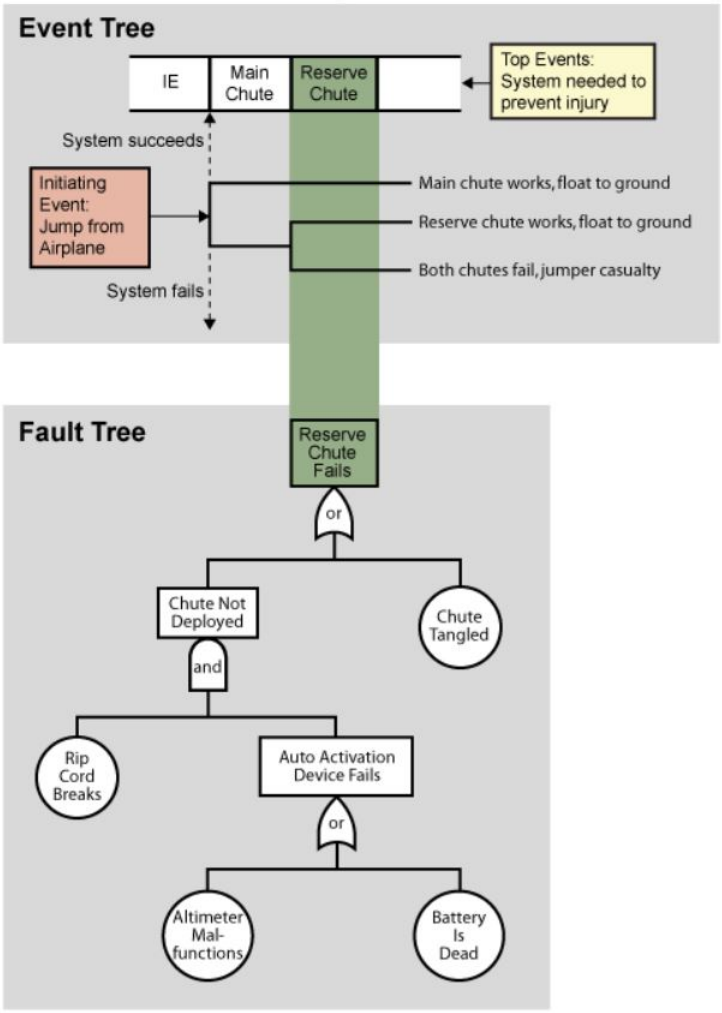


Figure 1.1: Sample PRA [7]

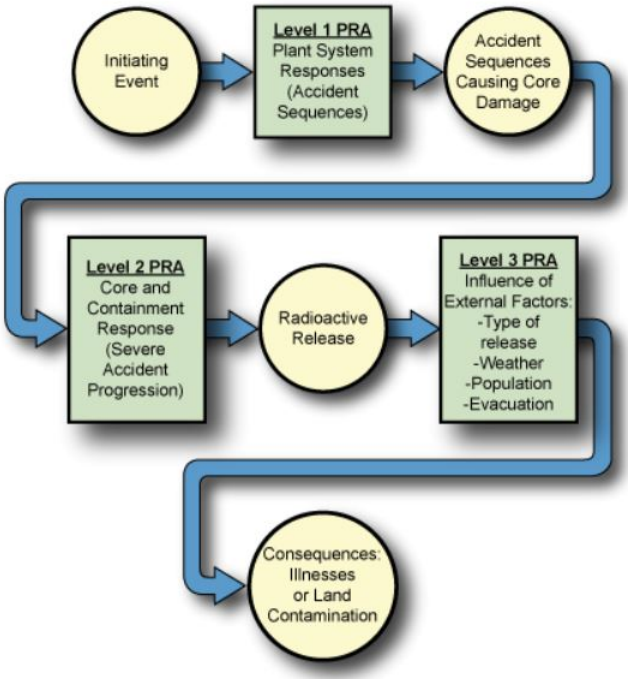


Figure 1.2: PRA Levels [7]

The first step of the PRA is the accident frequency analysis, referred to as PRA Level 1 analysis or front-end analysis. PRA Level 1 analysis includes the construction of event trees (ETs) and static fault trees (SFTs) of events that could lead to a core meltdown. These models must reflect the event sequences that may lead to core damage and estimate the CDF. Such SFT models should consider the faults of the components in each system included in the event trees, usually at the level of detail of valves, pumps, and other related components, including human actions. The main data source for the nuclear industry to develop ETs and SFTs is the NUREG-6928 [45] and the NUREG-0492 [58] for guidelines on the construction of SFTs.

SFT analysis might not be an appropriate technique to model some events, such as switching devices, spare component failures, and fault error recovery [13]. However, there are other safety analysis tools discussed in the literature for modeling these cases, such as dynamic fault trees (DFTs), Markov chains (MCs), and discrete-time Bayesian networks (DTBNs) [31]. This thesis proposes to use Bayesian networks (BNs) as an effective method that should be considered for PRA level 1 analysis, especially when including diverse and flexible strategies (FLEX) [11].

According to previous studies, the utilization of BNs has some advantages in conducting risk analysis. One advantage is that BNs can utilize Bayesian inferences to update failure probabilities of any system or component when having new evidence or information. Bayesian inferences allow us to model event sequences that might lead to core damage while updating the failure probabilities of upcoming events based on previous events in nuclear operations. One more benefit of utilizing BNs is that this reliability tool can include all types of accident sequences and scenarios. Hence, this method enables the modeling of FLEX strategies in the case of BDBEEs, such as earthquakes, floods, and high winds.

To determine the ultimate failure probability of FLEX devices applied to nuclear reactors, the success or failure of these tools can be broken down into a sequence of events from which Bayesian inference can be made. When extreme external events occur, preventive actions are taken at all facilities utilizing FLEX tools for the protection and deployment of stated equipment in the reactor. The occurrence of an extreme external event not only impacts the on-site facility but also have impact on either off-site operations and human reliability. Despite the fact that all of these elements has an impact on the ultimate failure probability of the FLEX strategies, the implementation of BNs enables their accurate consideration in real-time.

The primary reason why reliability studies generally depend on SFT evaluation is that SFT analysis is a standard technique with a clear and objective goal released nearly four decades ago. SFT develops a visual record of the system revealing the logical relationships between events that result in a failure [58]. However, SFT cannot be used for more complex systems, which may include dependent events, because of its intrinsic modeling limitation [31].

Also, according to Khadzad [31], the modeling limitations of SFT analysis could be overcome through the use of dynamic fault trees (DFTs), which account for dependencies by using dynamic logic gates. Although DFTs are able to deal with dependent events, they are not suitable for accounting for unknown uncertainties, which is dealt with using Bayes inference. Another problem regarding the use of DFTs in reliability studies is that it is not possible to use conventional SFT algorithms when dealing with several dynamic gates.

Markov chains (MC) are another alternative that allows for the modeling of multiple dynamic behaviors in systems, but the construction of these models is tedious and error-prone [13]. Each node of an MC represents a combination of all components in one system, which means that the space-state grows exponentially with the number of components in the system, a process often called a space-state explosion. Another limitation of modeling with MC is that it gets too complex with components that follow a non-exponential failure time distribution [38].

Moreover, BNs are a convenient approach to model reliability problems. Bobbio's study [3] mapped FTs into BNs, establishing a clear parallel between FTs and BNs. For the present thesis, a program called Reliability Assessment through Bayesian Networks (RABN) was developed in Matlab that allows the user to model the logic operations (e.g., "AND" gate and "OR" gate), which are used for SFT models. RABN can use operations in series or parallel, which is suitable for modeling complex systems. The source code of this program was developed following the formulations and equations in Boudali [5] and Khadzad [31]

studies and is an open-source program, allowing further development by anyone. Another program used in the thesis is Netica [42] that is based on belief networks and belief diagrams and enables users to create nodes and arcs to analyze system reliability by using the Bayesian network technique.

This thesis covers three different applications of the Bayesian network in order to show the effectiveness of the technique in dealing with the issues encountered in other safety analysis tools, i.e., Fault Tree and Markov Chain. In the first application, an FT including static gates (*i.e.,* OR and AND gates) is transformed into BN. After this conversion, the unreliability results of both tools are compared. The next application shows the transformation of an MC analysis that includes a cold spare gate into a discrete-time Bayesian Network (DTBN) to analyze a system having a dynamic gate. In the last application, a FLEX system designed for nuclear reactors and analyzed by FT method is mapped into BN to show that the effectiveness of BNs in the reliability analysis of FLEX strategies and, finally, a DTBN for the FLEX system is proposed in the case study. This thesis will first go through a review of the relevant literature needed to fully understand the current breadth of knowledge on this topic in chapter 2, and will then follow into a detailed outline of the methodology of reliability analysis tools in chapter 3. The thesis will then discuss the results obtained by the computational study in chapter 4, followed by conclusions and future work on this topic, which are discussed briefly in chapter 5.

Chapter 2

Review of Literature

FLEX strategies have been proposed to mitigate Beyond Design Basis External Events (BDBEEs). This chapter presents a literature review on some of those proposed strategies and methodologies used to estimate the reliability of those systems. Particularly, three methods are discussed: fault tree analysis, Markov chain, and Bayesian Networks. Fault tree analysis is a graphical method that uses a Boolean algorithm to quantify system components and top event reliability. The other technique reviewed in this thesis is the Markov chain which is a stochastic method that defines a sequence of possible events, and the probability of each event does not depend on the state of the previous event. This thesis focuses on Bayesian network which depends on Bayes' law to quantify system reliability. Although this chapter does not examine every study in the academic and industrial fields in detail, it provides sufficient information to understand the subject studied in this thesis and the future importance of the Bayesian network in reliability analysis.

2.1 Diverse and Flexible Coping Strategies (FLEX) at Nuclear Power Plants

The Fukushima Daiichi accident that occurred in east Japan on March 11, 2011, was a significant turning point for improvements in nuclear power plants' critical safety elements.

A severe earthquake caused the extended loss of alternating current (AC) power, and a tsunami followed by the earthquake led to the loss of the diesel generators leading to the loss of the core cooling. The accident also caused considerable risk in the containment of the building reactor.

After this severe accident in the Fukushima nuclear reactor, the U.S. Nuclear Regulatory Commission (NRC) released EA-12-049 [8], recommending mitigation approaches for beyond-design basis external events (BDBEE). Among the main lessons gained from this severe accident, we can highlight the challenges imposed by a loss of safety-critical systems complying with BDEBEEs. To reduce the negative impacts of BDBEEs, the Nuclear Energy Institute (NEI) prepared an instrument [11], that consists of recommendations regarding the utilization and implementation of diverse and flexible (FLEX) strategies to improve the capability of mitigating the consequences arising from BDBEEs.

FLEX strategies primarily intend to supply controlled as well as programmatic techniques to mitigate BDBEE by using mobile devices. Mobile tools that supplement set up systems, such as high-pressure pumps, can certainly make it possible for critical safety functions to be kept regardless of a proposed extended loss of normal AC power and the loss of normal access to the ultimate heat sink.

There are three steps in the FLEX methodology proposed by NEI-12-06 [11] in order to reduce the effect of BDBEEs. In the first step, installed devices and also resources are utilized. The middle stage requires supplying of installed plant equipment as well as consumables to preserve essential functions until they can be accomplished with resources carried from the off-site. The last stage depends on whether will be enough off-site sources to maintain those features for sufficient time.

When an accident has proceeded beyond the range of a design basis event, leading the plant

to lose its alternating current power or core cooling system, FLEX strategies will certainly be triggered to prevent the reactor from being harmed and improve the capacity of defense-in-depth. Figure 2.1 demonstrates the ability of FLEX strategies.

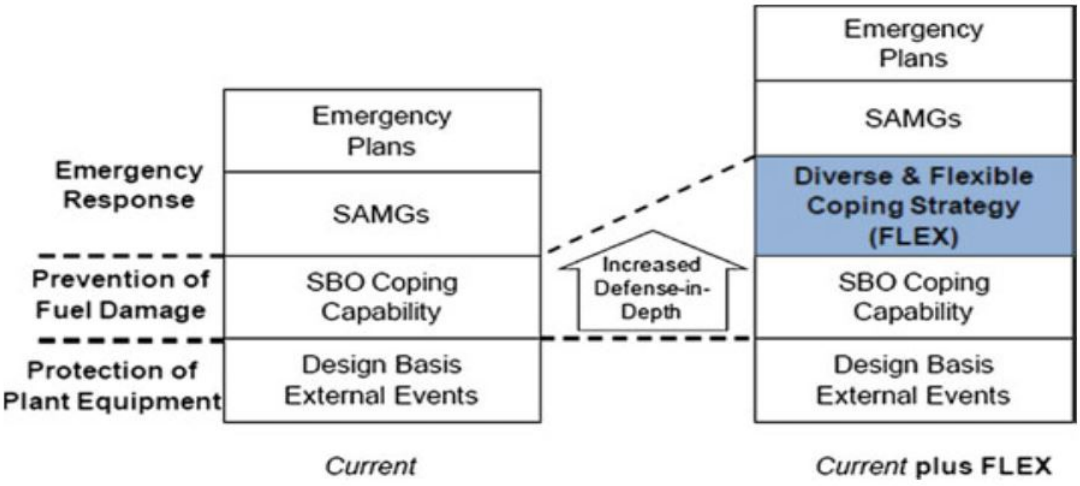


Figure 2.1: FLEX improves Defense-in-Depth [11]

FLEX approaches are highlighted in 2.1 for the prevention of fuel damage, and they can provide support to mitigate consequences following fuel damage. FLEX application will involve coordination with Severe Accident Management Guidelines (SAMGs) and supporting procedures. SAMGs are procedures that supply approaches to handle the effects of a severe accident which would be applied, upon particular plant parameter values, a sign of imminent or real damage to the fuel. SAMGs supply support to operators and staff to act in case of an accident condition progresses [11].

According to NEI [11], FLEX contains the following four aspects:

1. Mobile tools supplying power as well as water to preserve or restore vital safety features for all reactors.

2. Affordable setting up and protection against BDBEEs appropriate to a site.

3. Techniques and guidance for applying FLEX approaches. FLEX standards will supply pre-planned FLEX approaches for achieving particular tasks on behalf of Emergency Operating Processes (EOP) and also Abnormal Operating Processes (AOP) to enhance the capability of handling BDBEEs.

4. Programmatic management that ensures the ongoing feasibility and reliability of the FLEX methods.

The FLEX techniques contain not only an onsite element utilizing tools stored at the plant site but also an off-site element providing extra products and devices for long-term action.

2.1.1 Implementation of FLEX Strategies

Lim [35] investigates the use of two portable gas turbine generators (GTGs), a small and large one, as FLEX strategies during an extended Station Blackout (SBO). In the study, the event trees of both scenarios were created by adding FLEX equipment. Fault tree analyses were also conducted to estimate FLEX reliability. The cause-based decision tree method (CBDTM) and technique for human error rate prediction (THERP [55]) method were utilized to analyze the cognitive and execution part, respectively, for the human error probability (HEP). As a result of the study, when utilizing FLEX equipment, the core damage frequency of SBO is more efficiently minimized, and using a small mobile GTG is better than a large GTG.

Son and Lim [50] studied the effectiveness of FLEX strategies under long-term SBO by using PRA methods. In their studies, a mobile GTG and portable pump were utilized as FLEX equipment in a nuclear reactor. The main goal of these FLEX equipment is to prevent damage to the reactor and maintain safety functions. ETs and FTs were used to quantify CDF before and after the implementation of FLEX equipment in the reactor for the SBO

event. As a result of both PRA, it was concluded that FLEX strategies are an effective way to reduce CDF and mitigate risky situations caused by BDBEEs.

Xiao and Wang [62] have simulated specific scenarios to examine reactor cooling systems under the extended loss of AC power (ELAP) by utilizing a severe accident code called MAAP5.0.3 [INER-9482] to examine reactor cooling system when there is a seal leakage and the unavailability of emergency core cooling system. There are two different scenarios; base case utilizing installed devices as a FLEX model and plant-specific adding external devices for FLEX strategy to the primary and the secondary side when a BDBEE occurs. As a result, adding FLEX mobile devices on the primary and the secondary side enhanced the safety system of the nuclear power plant under BDBEEs.

Rahman and Shohag [47] propose an outside core cooling system and batteries to mitigate the consequence of the extended SBO. The research especially focuses on utilizing core cooling water injection and enhancement of the usable time of FLEX batteries throughout the SBO situation. The results indicate that this mitigation strategy is capable of preserving the reactor cooling system during the existence of BDBEEs.

According to Vaibhav and Biersdof's study [63], there are some advantages to the implementation of FLEX strategies to Nuclear Power Plants (NPPs) during not only the normal operation but also in the plant risk assessment. These advantages are:

- Direct and indirect financial saving when there is a shutdown situation,
- The reduced financial effect of element failure and the maximum efficiency in electric generation,
- Extension of allowed outage time, i.e., the time needed for portable tools be taken out of service without shifting to shutdown to the time for launch technical specification needed to shutdown.

2.2 Fault Tree Analysis

In numerous sectors, such as aerospace, automobile, healthcare, and power, safety-critical methods are commonly employed. Failures that may occur in these industries can lead to devastating impacts on both people and the ecosystem. Therefore, system reliability analysis is a powerful tool of reliability engineering that can be utilized to prevent frequent and severe failures.

There are several commonly used risk assessment methodologies that are designed to support safety analysts in order to perform reliability analysis of systems. One of the most common methods is the Fault Tree Analysis (FTA) [58]. FTA is a graphical method that describes how failures spread through the system, i.e., how the failure of a component causes a system failure. A Fault Tree is a visual depiction of the Boolean failure logic related to the advancement of a top event for a specific system. FTA builds a bridge between mathematics and related sectors to find solutions. FTA can be implemented deductively in its own nature, which means that the evaluation begins with the top event (system failing) and goes backward from the top to the bottom to identify the source of the top event. Figure 2.2 demonstrates a representative example for FTA.

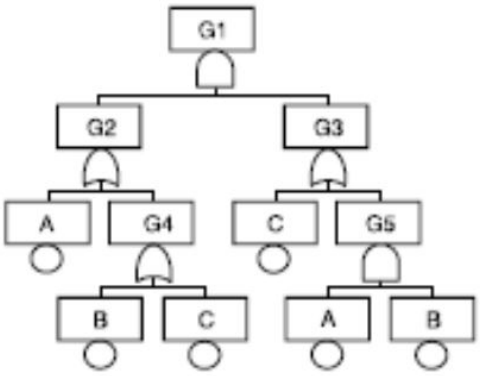


Figure 2.2: FTA example [15]

The analysis results show how individual parts or environmental factors can be combined to cause a system failure. There are two levels to constructing a fault tree: a qualitative level that describes the structure of the Fault Tree and a quantitative level that calculates probability values, for example, failure probabilities. At the qualitative level, Cut sets determining the element failures, or event combinations that can result in the undesirable event are one of the essential products from FTA. At this level, the main goal is to reduce fault trees' minimal cuts sets (MCSs) which are the minimal combinations of primary events that are required and enough to lead to the undesired top event.

The quantitative level analysis measures the top event probability as well as the probability of individual system components mathematically. By means of quantitative analysis, system reliability can be computed, and it also helps to determine which parts of the system are more critical. Therefore, the analysts may place more importance on essential elements or equipment by taking the required steps, e.g., to include redundant components in the system model.

In Fussell [18], component reliability characteristics are thoroughly defined in terms of their time-dependent failure rate (hazard rate) and time-dependent repair rate. In reality, failure and repair rates are frequently believed to be constant. In order to compute the reliability of a system, a few definitions are needed [18]:

- **Availability** is the probability that there will be no system failure at any specified point in the future. In contrast, **unavailability** is known as the probability that there will be a failure in the system. The unavailability, numerically, is equal to unity minus the availability. The availability can be formulated as

$$Availability = (MTBF)/(MTBF + MTTR) \tag{2.1}$$

where MTBF = mean time between failures, and MTTR = mean time to repair.

In contrast, the unavailability

$$Unavailability = 1 - Availability = 1 - (MTBF)/(MTBF + MTTR) \tag{2.2}$$

- **Reliability** is defined as the probability that there are no failures in the system at time t. **Unreliability** is described as the probability that the system has at least one failure to time t and mathematically is equal to one minus the reliability. The reliability can be formulated as

$$Reliability = e^{-\lambda * t} \tag{2.3}$$

where λ = the failure rate of a given system and t = mission time.

On the other hand, the unreliability

$$Unreliability = 1 - Reliability = 1 - e^{-\lambda * t} \tag{2.4}$$

These measurements are vital in deciding if the system satisfies the expectations for the reliability or whether additional measures are required. Availability is the percentage of time that a system is operating under normal working conditions. Reliability determines the probability that the system will satisfy certain performance standards and produce correct output for a given time.

SFTs employed in reliability analysis are an informative and straightforward formalism. Even though SFT is widely used in reliability calculations, it can only be implemented for static systems. The dynamic fault tree (DFT) is an expansion of SFT analysis and was developed particularly in order to overcome the drawbacks of SFT by enabling the reliability evaluation

of dynamic systems. Modern large-scale and dynamic systems can, however, work in several stages, e.g., a pump may have three different failures: pump fails to operate, the pump fails to start, and pump operator fails. There are a number of dynamic failure features such as functional dependent events and failure event priorities that an SFT is not capable of capturing. Another advantage is that DFT can be utilized for multi-state systems and is able to model interactions between system parts and variables.

Due to the limitations of the SFT, new methodologies have been developed and implemented in some studies. In Dugan's study [13], DFTs are able to solve SFT limitations, such as sequence-dependent failures and fault-error recovery. Kaiser [29] has developed a new model named State-Event Fault Trees (SEFTs), which enable the creation of state-based submodels in an FTA. The method combines FTA and State charts elements to incorporate finite-state designs with fault trees. In these previous works, DFT is one of the most extensively utilized dynamic expansions of the SFT, and it can capture sequence-dependent behavior, the behavior of dependent system parts as well as the priorities of the events. Chiacchio [6] developed a unique method called SHyFTA, which is a technique that integrates DFT as well as the Stochastic Hybrid Robot [1] methods to execute dynamic integrity analysis.

SFTs are the most basic reliability method, created in the 1960s in Bell Laboratories by H.A. Watson, with the support of M. A. Mearns to analyze a ballistic missile [14]. After the development of the FTA, it has been widely utilized in many application areas such as nuclear power plants and aerospace [28], [60]. Stamatelatos [53] introduced FTA as a method for the undesirable event, and the system was analyzed to discover all realistic methods in which the unwanted event (top event) can happen. However, due to its limitations, other reliability methods are described in the following sections.

2.3 Markov Chain Method in Reliability Analysis

Stochastic models are widely utilized in measuring the reliability of critical systems in thermodynamics, statistical mechanics, physics, chemistry, economics, finance, and energy. In these techniques, a state-space model is applied to a system to quantify the system reliability. MC is a reliability evaluation method used to define a series of possible events in a specific configuration or design. MCs are utilized throughout a wide variety of applications to represent a "memoryless" stochastic procedure. The procedure is composed of random variables that present the development of the process via different states. "Memoryless," also known as the Markov model, is that the probability of the present state depending only on the information in the current step, which means it does not depend upon any previous step. Figure 2.3 shows the general representation of a Markov chain. In the Figure, the A, B, and C represent the current states, and the arcs mean the transition from one state to another state.

In reliability analysis, MC represents different states that a system can be in at any time. These states connecting with transitions describe probabilities that the system will transfer from one state to another at a given particular time.

Recently, MC has become widespread to perform reliability analysis of systems consisting of critical components. There are several studies conducting MC as a reliability evaluation method in many industries such as automotive, digital systems, and energy, especially in nuclear power plants.

The MC method has been extensively explained in the studies of Howard [23] and Kemeny and Snell [30]. In Papazoglou and Gyftopoulos' study [43], an MC model was developed to analyze the reliability of the reactor shutdown system of the Clinch River Breeder Reactor. The authors also examined the uncertainties for the probability of loss of coolable core

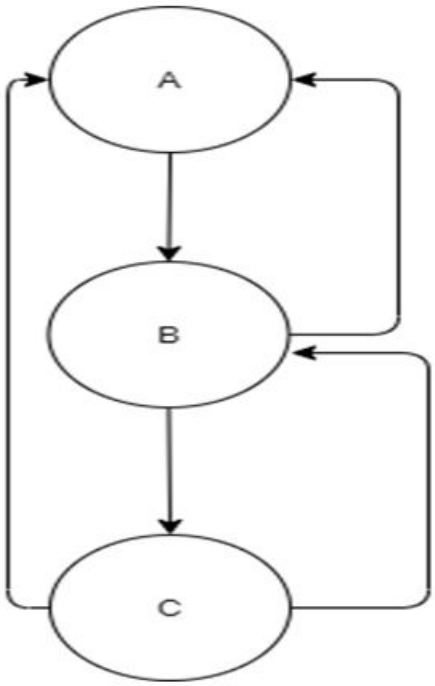


Figure 2.3: Visualization of Markov chain model

geometry of the Clinch River Breeder Reactor. According to the author, the utilization of the Markov model allows the following aspects:

1. Inspection and maintenance dependency upon the state of the system consisting of human actions;
2. Common cause failures by permitting connections between states of elements and failure rates;
3. Connections between shutdown unavailability and the transient occurrence.

Fleming [17] applied an MC model as a new reliability analysis to determine the failure probabilities of the piping system in NPPs. The main purpose of the work was to provide a new reliability model using MC and create new databases to determine the frequencies of pipe failures in the nuclear reactor so that the failure probabilities can be reduced prior to

cause a damage due to pipe leakages and cracks in the reactor. According to the result of the study, the Markov chain model is a useful technique to examine the effect of alternative approaches for inspection and leakage detection in the piping system of NPPs. In spite of the large uncertainties in evaluating the reliability of passive elements, the MC model and existing data together also have the capacity to sustain risk-informed decisions.

Jiang [27] employed the Markov chain method to monitor the digital control room of nuclear reactors. According to the authors, using the MC is an effective method to minimize unusual events since the occurrence of human factor-dependent events recently raised in the monitoring process. The model primarily computes the probability of the next monitoring component utilizing the existing plant information. As a result of the study, the human factors-dependent events can be significantly reduced by the MC techniques. The study also shows that the MC method can be employed to evaluate transition probability for digital human-machine interface in the nuclear reactors' monitoring system, which is a current and important subject of an investigation by the nuclear industry and regulatory body.

In Kumar's study [32], a method was suggested for reliability evaluation of safety-critical systems (SCSs) to show the coherent optimization MC and, the method has been confirmed on several safety systems of a nuclear reactor. In this study, a shutdown system (SDS)-2 which adds a liquid called gadolinium nitrate to quickly end reaction inside the reactor, was selected for reliability analysis. According to the author, although MC is a powerful method in the reliability prediction of SCS, there is a challenge called state-space explosion. The state-space explosion means that when the number of states is increased, the computing mechanism becomes more complex. In order to solve the problem, the work utilized an optimized MC technique that combines states based upon transition probability.

2.4 Bayesian Networks Applications for Risk Analysis

BNs are another powerful tool for reliability studies. Considering the background of Bayes' theorem, it is seen that the term "Bayes" was first presented by Thomas Bayes and was named after him. The theory presents the relationship between one conditional probability and its inverse. It gives a mathematical expression for updating an estimate by using experience and observation. Bayes' law explains the probability of an event-based upon conditions that could be associated with the event. For two events occurring in a system:

$$P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)} \tag{2.5}$$

where X and Y = events, P(X) and P(Y) = probabilities of X and Y events, P(X|Y) and P(Y|X) = conditional probabilities.

In system reliability analysis, BNs are utilized in more relevant areas such as medicine, data analysis, automotive, and energy generation since it has an advantage in dealing with complex systems and dependencies.

BNs can be described as a directed acyclic graph (DAG) which consists of nodes standing for the variables of the system and arcs depicting conditional dependencies between parent and daughter nodes. A basic example of a BN structure is depicted in Figure 2.4.

Figure 2.4: Visual representation of a BN

BN is a powerful tool to deal with risk assessment in the complex system, as discussed in further studies below. In Jensen's [25] and Pearl's [44] studies, the main characteristic of the BN technique was primarily clarified. Basically, in the BN method, the distribution proba-

bilities for some variables are calculated by using the observation results of some variables and their prior knowledge. Similarly to the FTA, Jensen [26] describes two primary parts in the BN method:

1. Qualitative approach: figures such as nodes and arcs define the relationship between system variables visually.

2. Quantitative approach: the probability computations between parent and daughter nodes are generated, and also conditional probability distribution (CPD) is produced by means of these computed probabilities.

By making use of these two approaches (qualitative and quantitative), all required probability information of a system is calculated and obtained. Spiegelhalter and Lauritzen [52] explained how to calculate the CPD for a given graphical model (qualitative) of the system.

BNs are commonly identified as being a reliable and durable decision-making structure for issues consisting of probabilistic reasoning and uncertainty. When utilizing BN, there are two types of reasoning algorithms, which are predictive and diagnostic [38]. Modifying prior probabilities by making use of real observations of events can be executed in a diagnostic algorithm. The computing of the probabilities of any elements by using conditional dependencies and probability distributions in a system can be given as an example of predictive reasoning.

The other advantage of BNs is to allow the utilization of information from different kinds of sources such as expert opinion, historical and experimental data. There is a wide range of applications that are modeled by implementing BNs. Neil [40] has studied the calculation of reliability of military vehicles by utilizing TRACS software that is a Bayesian belief network (BBN) based tool. The study presents that BBNs provides two benefits; enhancing the

reliability predictions in the early stage of the design and influencing system reliability in the positive direction. Moreover, "learning" BBN has been developed to learn subelements' probability distribution.

In Hu's study [24], a model was developed by utilizing BNs with causality constraints (BNCC) in order to obtain the risk assessments software for project design. The implementation of BNs on design software has some benefits: BNs can (1) integrate the information from different sources such as expert opinion, historical data, and experimental data; (2) graphically show the cause-effect connections to assist in determining risky situations; (3) give probabilistic calculations and design uncertainties.

In Gran and Helminen [21], the BN method was utilized to assess the reliability of nuclear power plants. In many areas, such as power systems and military equipment, BNs have been applied as a risk analysis method ([64], [10], [65]). According to Sigurdsson [49], the necessary actions to construct and use a BN are depicted in Figure 2.5.

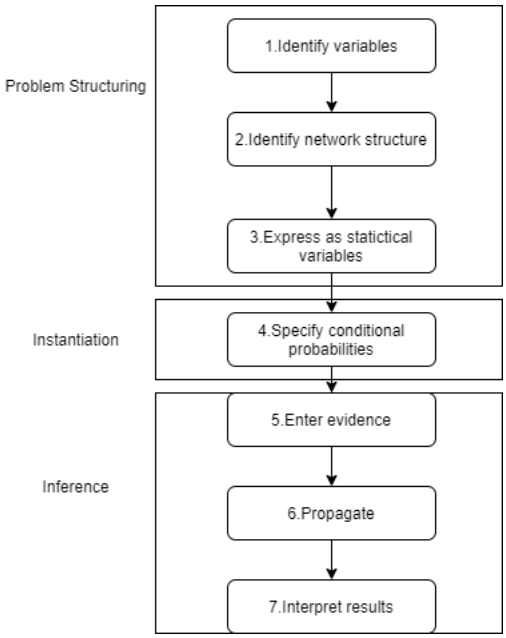


Figure 2.5: Necessary steps to create and use BN [49]

Gomes [20] worked on the required steps when there is a human-based error in radiotherapy procedures by determining possible human failures. BN was utilized to model the most related stages of teletherapy as well as brachytherapy. Additionally, when there is a lack of data source in the system, the expert opinion was utilized. The main purpose of the study is to show how accident elimination affects system reliability and follow the connections between required steps. As a result of the study, BN is a powerful method to demonstrate the connections between stages in brachytherapy and teletherapy, which include human failures.

Doguc [12] suggested a technique for automated construction of BN for the design of a system in order to deal with the limitation of specialists. The technique automatically constructs BN using historical data to evaluate system reliability. In the method, the K2 algorithm, which is a data mining method, was applied to produce raw system data to utilize instead of expert opinion. As a result, it can be concluded that BN can be automatically constructed and utilized in need of an expert.

A variety of current researches have attempted to utilize BNs to offer an integrated structure for reliability modeling and assessment of the complicated system. Specifically, BNs have been utilized as an alternate design of static fault trees. BNs allow the modeling of a diverse set of scenarios, such as common cause failures, multi-state variables, noisy gates, and simple dependent failures that have the power to increase system reliability analysis ([3]; [33]; [46]; [48]; [57]).

2.5 Comparison among Reliability Analysis Methods

This section consists of the comparison of modeling capacities between BN and two classic techniques of reliability analysis: FT and MC.

FTs are based upon the theory of Boolean depiction of primary events. The probability calculation in an FT is effectively solved by binary decision diagrams allowing a precise calculation, as described in Chapter 3.

FT is a really intriguing model considering that it enables the calculation of the probability of the top event based on the failure probabilities of various types of sources, such as organizational, technical, and decisional, to acquire a complete safety evaluation.

However, this technique requires the representation of more than one state variable when elements of the system are exposed to more than one failure. In this case, the utilization of FT is not appropriate. An additional restriction is that the FT technique is restricted to analyze simply one top event. On the other hand, BN enables the same capacities to the FT with the benefits of multi-state variable modeling and the capability to analyze numerous outcome variables in the same system.

The MC is an appropriate technique for the reliability analysis of a given system. When dependencies exist among system elements, the MC enables evaluation of the precise failure probabilities. Moreover, it enables the combination of various kinds of knowledge in order to express multi-state variables. Nevertheless, the utilization of MC becomes more complicated with the multitude of variables and states in order to describe actions and causalities of the system. MC can also become very complex to deal with non-constant failure rates. With the utilization of BN, the state-space explosion can be solved since the variables in CPTs of a BN is significantly lower compared to an MC.

Chapter 3

Methodology

The main purpose of this section is to examine the mathematics, rules, and processes of the three safety analysis tools, *i.e.*, Fault Tree, Markov Chain, and Bayesian Network studied in this thesis. A combination of quantitative and qualitative approaches is utilized in the reliability analysis of these three methods. The critical safety systems are visualized using qualitative approaches. In quantitative analysis, the reliability calculations of the system and system components are explained in detail and summarized in tabular form. As discussed before, FTA was used to compare with the results of BNs because of its wide range of applicability. However, when dealing with a simple dynamic problem, *i.e.*, a small number of components, and constant failure rates, MC can provide a fast analytical solution to verify the results of the DTBN. The results are presented in chapter 4.

3.1 Fault Tree Analysis

FTA is the system evaluation method utilized to identify the root causes and the event probability of a defined undesirable occasion. FTA is used to examine large, complicated dynamic systems to comprehend and avoid possible issues. FTA enables the system expert to special model combinations of fault events that can create an undesirable event [15]. The undesirable event can be defined as a system hazard or a mishap that is under accident investigation [15].

FTs are visual designs utilizing logic gates and fault events to create the cause-effect connections associated with leading to the undesirable event. In the FTA, the visual model can be converted into a mathematical version in order to calculate the probabilities of failure and system importance measures. Considering a reliability analysis tool, FT leads to a logical and visual depiction of the different combinations of possible occasions, such as working and faulty happening in the system.

In the FTA, there are some basic goals described by Ericson [15]:

1. Identify the root causes of an undesirable event.
2. Develop the root causes of an accident that has happened and avoid them from reoccurring.
3. Determine the undesirable event-causal aspect combinations and their probabilities.
4. Establish risky fault paths and their structures.
5. Determine risk importance measures for system elements.
6. Provide a probabilistic risk analysis of the system.

3.1.1 Methodology

In the FTA, the methodology consists of eight basic stages shown in Figure 3.1. These stages are the actions needed to execute a precise and complete FTA. Some of these steps can be combined or extended, but the general process basically proceeds in this way [15]:

1. Learn the system design and how it operates. Obtain existing design information.
2. Specify the issue and develop the appropriate undesirable event for evaluation.

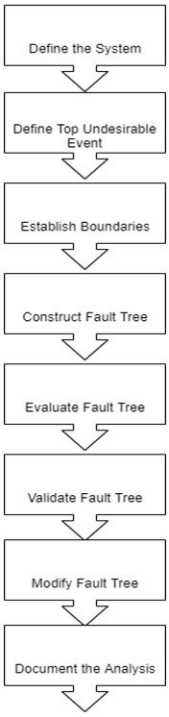


Figure 3.1: FTA process [15]

3. Determine analysis rules and limits. Scope the issue and document all guidelines.

4. Follow construction process, guidelines, and logic to construct FT design of the system.

5. Create cuts sets and the probabilities of the system. Determine weak spots and safety issues in the model.

6. Examine-in the case that the FT design proper, complete, and precisely shows the system design.

7. Change the FT as required throughout validation or because of the system model modifications.

8. Record the whole analysis with supporting information. Supply as client product or maintain for future referral.

Figure 3.2 shows the basic symbols for a standard event, condition event, and transfer event, and also their meanings in the analysis. Figure 3.3 displays the gate event signs, meanings, and probability computation equations. It is by means of the gates that the FT logic is built and the framework grows in size and depth.

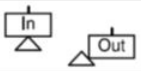
Symbol	Type	Description
	Node Text Box	Contains the text for all FT nodes. Text goes in the box, and the node symbol goes below the box.
	Primary Failure (BE)	A basic component failure; the primary, inherent, failure mode of a component. A random failure event.
	Secondary Failure (BE)	An externally induced failure or a failure mode that could be developed in more detail if desired.
	Normal Event (BE)	An event that is expected to occur as part of normal system operation.
	Condition (CE)	A conditional restriction or probability.
	Transfer (TE)	Indicates where a branch or sub-tree is marked for the same usage elsewhere in the tree. In and Out or To/From symbols.

Figure 3.2: FT symbols for basic events, conditions, and transfers [15]

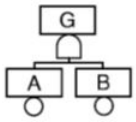
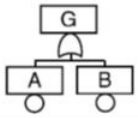
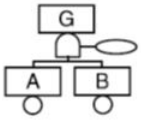
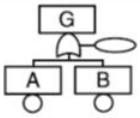
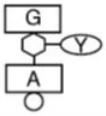
Symbol	GateType	Description
	AND Gate	The output occurs only if all of the inputs occur together. $P = P_A \bullet P_B = P_A P_B$ (2 input gate) $P = P_A \bullet P_B \bullet P_C = P_A P_B P_C$ (3 input gate)
	OR Gate	The output occurs only if at least one of the inputs occurs. $P = P_A + P_B - P_A P_B$ (2 input gate) $P = (P_A + P_B + P_C) - (P_{AB} + P_{AC} + P_{BC}) + (P_{ABC})$ (3 input gate)
	Priority AND Gate	The output occurs only if all of the inputs occur together, and A must occur before B. The priority statement is contained in the Condition symbol. $P = (P_A P_B) / N!$ Given $\lambda_A \approx \lambda_B$ and N = number of inputs to gate
	Exclusive OR Gate	The output occurs if either of the inputs occurs, but not both. The exclusivity statement is contained in the Condition symbol. $P = P_A + P_B - 2(P_A P_B)$
	Inhibit Gate	The output occurs only if the input event occurs and the attached condition is satisfied. $P = P_A \bullet P_Y = P_A P_Y$

Figure 3.3: FT symbols for gate events [15]

3.1.2 Block Diagrams

In FTA, there are nodes connected to each other in a treelike framework. The nodes represent failure modes and are connected together by Boolean logic and signs. These symbols are used to create the fundamental building blocks of FTA and include four general groups [15]:

- Basic events
- Gate events
- Conditional events
- Transfer events

3.1.3 Cut Sets

In FTA, there are key elements named cut sets utilizing in order to determine the system element failures and combinations of events that can result in the top undesirable event happen. Another purpose of a CS is to give a mechanism to compute the probabilities of a system and its components. The minimal cut set is an important term that is utilized to minimize the number of events that lead to the undesirable top event happening.

Figure 3.4 demonstrates an example FT with its resulting CSs noted on the right. According to the behavior of a CS, each CSs can lead to an undesirable top event happen. CSs are produced via Boolean mathematics, and many kinds of algorithms exist for producing CSs.

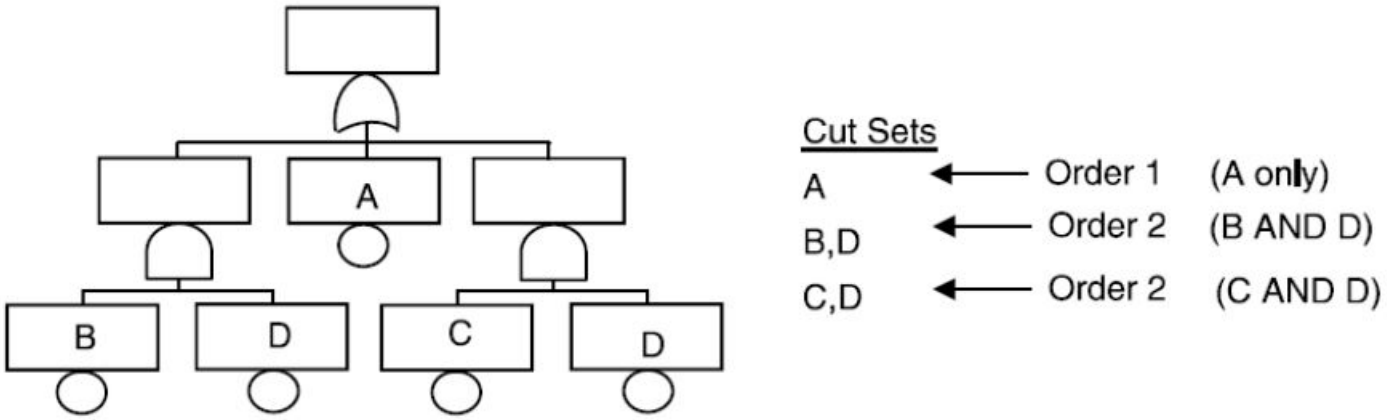


Figure 3.4: Cut sets example [15]

3.1.4 Mathematics

There are some mathematical expressions, such as Boolean algebra, probability, and reliability theories, in the background of FTA that allow the analysis to be performed. The following are simplified explanations for mathematical expressions used in FTA [15].

- **Probability of success:** Reliability (R) of a system element, which is computed by

$$R = e^{-\lambda * T} \tag{3.1}$$

  where λ=system element failure rate and T = system element exposure time.

- **Probability of failure:** Unreliability (Q) is defined as component failure probability, where

$$R + Q = 1 \tag{3.2}$$

  and

$$Q = 1 - R = 1 - e^{-\lambda * T} \tag{3.3}$$

- **Boolean rules for FTA:** The following Boolean equations employ directly to FTA for a decrease of CS to their minimal elements.

  $a * a = a$(AND gate, a variable itself is always equal to the same. For example, if ideally, the same two components are in working condition, the result is working for AND gate.)

  $a + a = a$(OR gate, a variable itself is always equal to the same.)

  $a + ab = a(1 + b) = a * 1 = a$(OR gate, b variable with 1 is equal to 1.)

  $a(a + b) = a * a + a * b = 1 + a * b = a(1 + b) = a * 1 = a$

- **AND gate probability expansion:** The probability computation for the AND gate, as shown in Fig.3 for two and three components, is

$$P = P_A * P_B * P_C * P_D * P_E, ..., P_N \tag{3.4}$$

  N = number of inputs.

- **OR gate probability expansion:** Probability calculation for an OR gate is

$$P = (P_A + P_B + P_C) - (P_{AB} + P_{AC} + P_{BC}) + (P_{ABC}) \tag{3.5}$$

  for 3-input OR gate.

These are the basic mathematical expressions and definitions to conduct FTA in this thesis. A more comprehensive and detailed discussion can be found in Ericson [15].

3.2 Markov Chain Analysis

MC analysis is a modeling strategy that is extensively valuable for the reliability evaluation of complicated systems. The MC is rather beneficial for modeling complex operation systems having dynamic behavior and also repairs models [34]. Actually, it is commonly utilized to execute reliability and availability evaluation of a given system that has constant failure rates.

Prior to beginning the Markov analysis formulation, all the components of the system and their possible states are determined. A state is explained to be a specific combination of working and failed components.

Table 3.1 shows all possible states of a system, including three components: a, b, and c. As it can be seen in the table, there are eight different combinations of the system components. In Table 3.1, W represents an operational element, and F is an element having a fault. In the system, if there are N components, the system will have 2^N possible states. It must be noted that the number of states grows much quicker than the number of elements.

States								
Component	1	2	3	4	5	6	7	8
a	W	F	W	W	F	F	W	F
b	W	W	F	W	F	W	F	F
c	W	W	W	F	W	F	F	F

Table 3.1: Three-Component States in MC [34]

In the MC analysis, the states of the system change depending on the configuration of the components. In the three-component system, there can be three different arrangements, as shown in Figure 3.5. In case the system components are connected to each other in series, even failure occurring in only one of the components will cause the entire system to fail. On the other hand, in a system with parallel-connected redundant components, all the components need to fail in order for the system to fail.

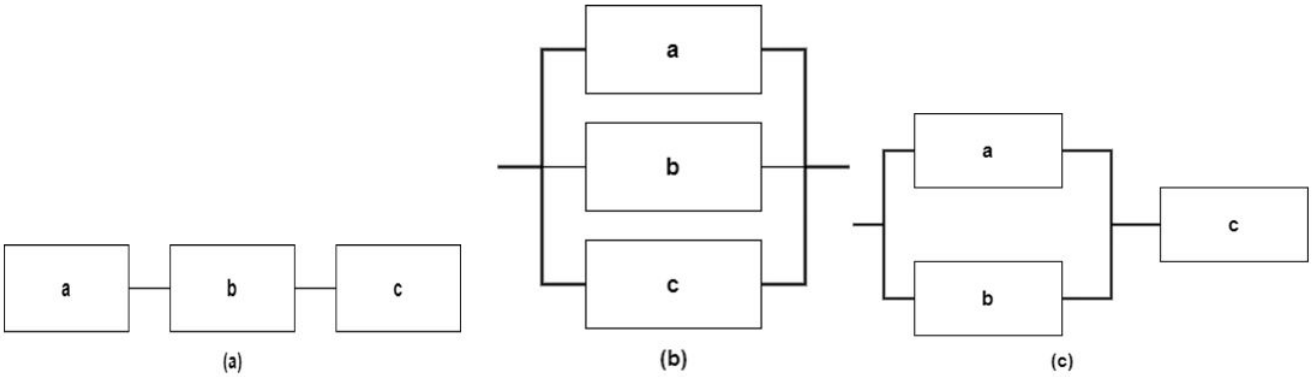


Figure 3.5: Three-component block diagrams [34]

In the MC, the main purpose is to compute $P_i(t)$, representing the probability that the system has not failed in state i at time t. After calculating this probability, the reliability of a system is computed as a function of time, as shown in equation 3.6, where the sum shows all the working states.

$$R(t) = \sum_{i \in W} P_i(t) \tag{3.6}$$

As an alternative way, the reliability of the system can be computed from equation 3.7 with the summation of faulty states in the system.

$$R(t) = 1 - \sum_{j \in F} P_j(t) \tag{3.7}$$

In the MC analysis, state 1 is usually assigned as the state for which all the elements are working, and, therefore, at time t=0 the system is working. Hence,

$$P_1(0) = 1, \tag{3.8}$$

and,

$$P_i(0) = 0, \qquad i \neq 1. \tag{3.9}$$

Any component in the system can only exist in one state at any time. Therefore, the sum of all possible states of the component is always equal to 1, as shown in equation 3.10.

$$\sum_i P_i(t) = 1 \tag{3.10}$$

3.2.1 Two Independent Elements

Table 3.2 shows all possible states for a two-component system. The logic of the state changes from each of the possible scenarios is shown in Figure 3.6 using the transition diagram. In Fig. 3.6, the probability of both components to failure from state 1 to state 4 is ignored.

States				
Component	1	2	3	4
a	W	F	W	F
b	W	W	F	F

Table 3.2: Two-Components States in MC [34]

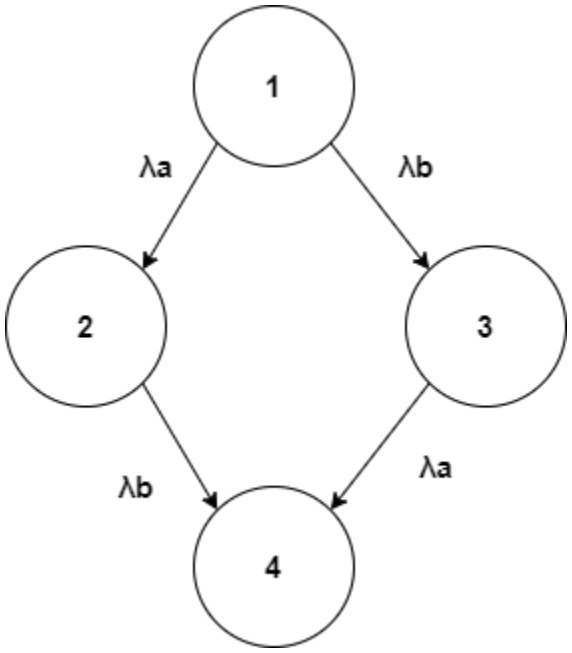


Figure 3.6: State transition diagram [34]

We are assuming, that the components a and b have constant failure rates, λ_a and λ_b, respectively, the probabilities of each state are calculated by solving equations 3.11 to 3.14.

$$\frac{d}{dt}P_1(t) = -\lambda_a P_1(t) - \lambda_b P_1(t) \tag{3.11}$$

$$\frac{d}{dt}P_2(t) = \lambda_a P_1(t) - \lambda_b P_2(t) \tag{3.12}$$

$$\frac{d}{dt}P_3(t) = \lambda_b P_1(t) - \lambda_a P_2(t) \tag{3.13}$$

$$\frac{d}{dt}P_4(t) = \lambda_b P_2(t) + \lambda_a P_3(t) \tag{3.14}$$

Using Eqs. 3.8 and 3.9 as initial conditions, one can obtain the following solution:

$$P_1(t) = e^{-(\lambda_a+\lambda_b)t} \tag{3.15}$$

$$P_2(t) = e^{-(\lambda_b)t} - e^{-(\lambda_a+\lambda_b)t} \tag{3.16}$$

$$P_3(t) = e^{-(\lambda_a)t} - e^{-(\lambda_a+\lambda_b)t} \tag{3.17}$$

$$P_4(t) = 1 - \sum_{i=1}^{3} P_i(t) \tag{3.18}$$

$$P_4(t) = 1 - e^{-(\lambda_a)t} - e^{-(\lambda_b)t} - e^{-(\lambda_a+\lambda_b)t} \tag{3.19}$$

The system of the equation above (Eqs. 3.15 - 3.18) can be solved analytically or using computational software, such as Mathematica Wolfram [61]. After the calculation of $P_i(t)$, the reliability of the system can be computed depending on the configuration of system components. Here, only two components are considered; there are only two possible configurations in the system: series and parallel. Therefore, for two components in series

$$R_s(t) = P_1(t) \tag{3.20}$$

$$R_s(t) = e^{-(\lambda_a+\lambda_b)t} \tag{3.21}$$

If the components are connected in parallel, the system will failure when both components failure. Therefore,

$$R_p(t) = P_1(t) + P_2(t) + P_3(t) \tag{3.22}$$

$$R_p(t) = 1 - P_4(t) \tag{3.23}$$

$$R_p(t) = e^{-(\lambda_a)t} + e^{-(\lambda_b)t} - e^{-(\lambda_a+\lambda_b)t} \tag{3.24}$$

A more comprehensive description and calculation of Markov chain analysis can be found in Lewis' study [34].

3.3 Bayesian Network Analysis

3.3.1 Fundamentals and Predictive Approach

A Bayesian network (BN) can be defined as a probabilistic graphical method that has a group of variables and their conditional probabilities through a directed acyclic graph (DAG). BNs are a powerful tool in understanding the dependency among events and assigning probabilities so that the probabilities of possible known causes that are contributing factors can be predicted when an event occurred. Before explaining what the Bayesian network exactly is, it is important to define some terms related to the probability theorem.

- A joint probability distribution means a probability distribution for more than one variable. Let's assume, $X_1, X_2, ...X_n$ are some events, the joint probability distribution is,

$$P(X_1, X_2, ..., X_n) = P(X_1).P(X_2|X_1).P(X_3|X_2, X_1)...P(X_n|X_1, X_2, ..., X_n) \quad (3.25)$$

- Conditional probability can be defined as the probability of an outcome happening based upon the occurrence of a previous outcome. The conditional probability of X given Y, shown by P(X|Y),

$$P(X|Y) = \frac{P(X \cap Y)}{P(Y)} \quad (3.26)$$

- Lets say $X_1, ..., X_n$ are disjoint events. The total probability of any event Y is,

$$P(Y) = P(X_1 \cap Y) + ... + P(X_n \cap Y) = P(X_1)P(Y \cap X_1) + ... + P(X_n)P(Y \cap X_n) \quad (3.27)$$

According to Murphy [39], a BN exemplifies the causal probabilistic relationship among some arbitrary variables and their dependencies. Also, it supplies a compact depiction of a joint probability distribution. A BN includes two essential elements: a directed acyclic graph and conditional probability distribution [33]. For the directed acyclic graph, nodes that exhibit a set of random variables are connected to each other by arcs representing dependencies among nodes to create the BN. A conditional probability distribution is determined for each variable (node) in the graph. To put it another way, the conditional probability distribution of a variable (node) is specified for each possible outcome of the preceding causal node.

The dependencies throughout the nodes are calculated utilizing Bayes' law [54], specified mathematically in equation 3.28, defining only two events, X and Y.

$$P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)} \tag{3.28}$$

According to equation 3.28, P(X|Y) represents the conditional probability of event X happening considered that event Y has actually happened. P(Y|X), the CP of event B occurring given that X has occurred, P(X) and P(Y), which are the probabilities of X and Y happening, respectively. In reliability analysis, these nodes can represent different things, such as component states in a given system or a human error probability.

BNs are a practical method to handle various relationships among its random variables (nodes). Figure 3.7 depicts a BN with events X and Y, where event Y is dependent on event X.

In Figure 3.7, X, known as the parent node, and Y, called the child node, represent random variables that are the probabilities of its events occurring [51]. The arch in the figure shows a connection that represents the probability of Y is influenced by the probability of event X through Bayes' law and the total probability theorem. Equation 3.29 explains how to

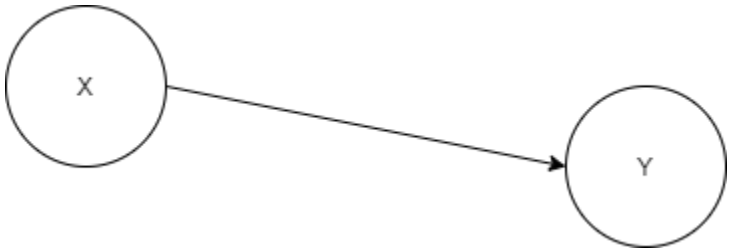


Figure 3.7: BN of event Y dependent on event X

calculate the probability of event Y. Node X does not receive an arch from any type of node as it is a parent node. Therefore, the probability of event X is just marginal prior probability P(X), and,

$$P(Y) = P(Y|X)P(X) \tag{3.29}$$

In equation 3.29, the probability value of event Y depends on the CP of P(Y|X) and the prior probability of event X, P(X). When analyzing with the BN method, the conditional probability table (CPT) is specified for a set of random variables to show conditional probabilities of individual variables relative to the others. Table 3.3 represents the CPT of node Y that was described in Figure 3.7.

Dependent Event	Conditional Probability of node Y
X	P(Y|X)

Table 3.3: Conditional Probability Table of node Y

Each child node (variable) in the BN has a CPT. The size of the CPT depends on the number of parent nodes connected to the child node. If more arches are connected to the same child node, more terms will be used in Eq. 3.29. For example, if N parent nodes are connected to the child node below, the total probability is given by

$$P(Y) = \sum_{i=1}^{N} P(Y|X_i)P(X_i) \tag{3.30}$$

3.3.2 Discrete-Time Bayesian Network

The BN described above is intrinsically static; however, one of the challenges encountered in reliability analysis is modeling dynamic systems. Boudali and Dugan [5] have established a discrete-time Bayesian network (DTBN) in order to handle the dynamic behavior of systems. They defined a mission time (T), dividing the total period of interest into n discrete time intervals plus one more interval that accounts for the period [T,+∞]. Figure 3.8 shows the timeline intervals for each node.

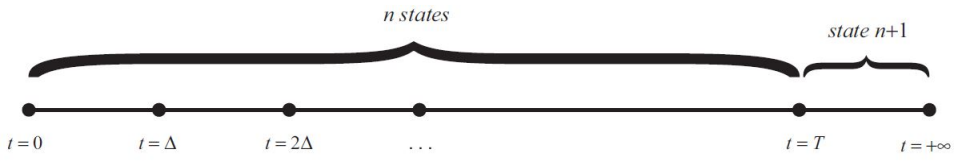


Figure 3.8: Timeline intervals [5]

Assuming that there is an interaction at any time (t) between the states of two or even more events of BN throughout the period T, its probability can be computed by populating the CPTs with values that calculate these event probabilities.

According to the formalism defined in the study Boudali and Dugan [5], the node X specifies to be in the state i ($1 \leq i \leq n$), or X=i, which means X failed in the ith interval. Hence, the marginal probability values of component X can be obtained for each time interval by using equation 3.31.

$$P(X=i) = P((i-1)\triangle < t_X < i\triangle) = \int_{(i-1)\triangle}^{i\triangle} f_X(t)dt = F_X(i\triangle) - F_X((i-1)\triangle) \quad (3.31)$$

where t_X defines the time of event X, F_X symbolizes the cumulative probability distribution, $\triangle$ shows the interval length $\triangle = T/n$, and n is the time granularity [5].

In the same way, when X in the state n+1, which indicates X has endured the mission time T.

$$P(X=n+1) = P(t_X > T) = \int_{T}^{\infty} f_X(t)dt = 1 - F_X(T) \quad (3.32)$$

Figure 3.9 depicts a BN with events C and D, which are independent, and connected to each other with an OR gate. The visual transformation of a basic OR gate into BN is represented on the right-hand side. After the visual transformation, the next step is the probability calculation of child and parent nodes. Figure 3.9 shows the fault tree of a two-component system that is connected by a static OR gate and its equivalent BN. It is noted in Figure 3.9, the basic events C and D have constant failure rates.

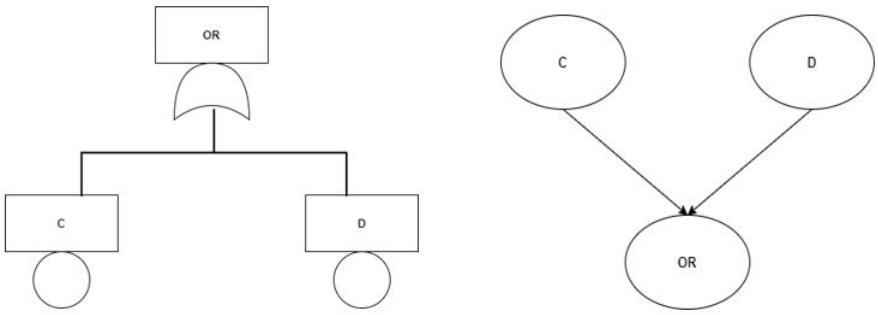


Figure 3.9: OR gate and its transformation into BN [5]

Tables 3.4 and 3.5 show the marginal probability table of events C and D for n=2, respectively. The probability values of time interval]0, t[and] t,T[can be calculated by using equation 3.31, and $P_3 = 1 - P_1 - P_2$.

C	]0,$\triangle t$]	]$\triangle t, T$]	]T, $+\infty$[
P(C)	$P_1(C)$	$P_2(C)$	$P_3(C)$

Table 3.4: Marginal probability table of event C [5]

D	]0,$\triangle t$]	]$\triangle t, T$]	]T, $+\infty$[
P(D)	$P_1(D)$	$P_2(D)$	$P_3(D)$

Table 3.5: Marginal probability table of event D [5]

Table 3.6 shows the CPT of an OR gate for n=2. It is a unit matrix since the OR gate and component D fail at the same time. In Table 3.6, "1"s represent the failure at the given time interval, and "0"s show the working condition of the component at that time interval. For example, when the components C and D fail at the first time interval (]0, $\triangle t$]), the OR gate fails at the same time interval, so Table 3.6 shows "1" in the first column and other columns are "0"s.

C		]0, $\triangle$t]			]$\triangle$t, T]			]T, $+\infty$[		
D		]0, $\triangle$t]	]$\triangle$t, T]	]T, $+\infty$[	]0, $\triangle$t]	]$\triangle$t, T]	]T, $+\infty$[	]0, $\triangle$t]	]$\triangle$t, T]	]T, $+\infty$[
OR	]0, $\triangle$t]	1	1	1	1	0	0	1	0	0
	]$\triangle$t, T]	0	0	0	0	1	1	0	1	0
	]T, $+\infty$[	0	0	0	0	0	0	0	0	1

Table 3.6: Conditional probability table of OR gate [5]

3.3.3 Static and Dynamic Gates

Considering that it is not easy to utilize static OR gates and AND gates along with dynamic gates to design a complicated system, these gates also require to be designed as discrete-time gates to adapt with the remainder of DTBN. When modeling a discrete-time BN system, having more components means more multi-dimensional CPTs. In order to minimize the sizes of CPTs, the neutral dependency method [31] is employed in Figure 3.10. In the neutral dependency method, the parent nodes are connected to each other from left to right so that the visual representation of the transformation looks like a line as shown in the figure, and the number of CPTs is effectively reduced.

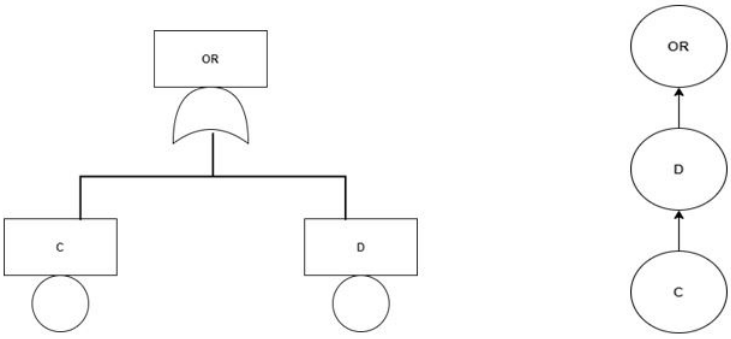


Figure 3.10: OR gate structured in DTBN [31]

The conditional probabilities of the D component are calculated by utilizing Eq. 3.33 (as shown in Table 3.7) and Eq.3.34 (as shown in Table 3.8) for OR and AND gates, respectively. Tables 3.7 and 3.8 show the probability values in each cell. It should be noted that the summation of probability values in each row is every time same and equal to 1.

D	]0, △t]	]△t, T]	]T, +∞[
]0, △t]	$P_1 + P_2 + P_3$	P_1	P_1
]△t, T]	0	$P_2 + P_3$	P_2
]T, +∞[	0	0	P_3

Table 3.7: CPT of D for an OR gate [31]

D	]0, △t]	]△t, T]	]T, +∞[
]0, △t]	P_1	0	0
]△t, T]	P_2	$P_1 + P_2$	0
]T, +∞[	P_3	P_3	$P_1 + P_2 + P_3$

Table 3.8: CPT of D for an AND gate [31]

- CPT of D event for OR gate

$$P(D = i|C = j) = \begin{cases} 0 & \text{if } i > j \\ \sum_{k=i}^{n+1} P(D = k) & \text{if } i = j \\ P(D = i) & \text{if } i < j \end{cases} \tag{3.33}$$

- CPT of D event for AND gate

$$P(D = i|C = j) = \begin{cases} P(D = i) & \text{if } i > j \\ \sum_{k=1}^{i} P(D = k) & \text{if } i = j \\ 0 & \text{if } i < j \end{cases} \tag{3.34}$$

3.3.4 Cold Spare Gate

A spare gate is a dynamic gate utilized to model standby systems that include two elements: primary and standby components. Spare gates are very common in NPPs for redundant components. When the primary component has a failure, it is replaced by its standby element immediately. According to Dugan [13], the standby system only fails if all the components fail or become unavailable.

Figure 3.11 shows a CSP gate having two components.

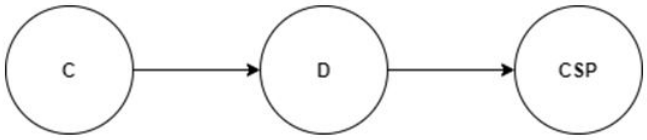


Figure 3.11: BN of event D dependent on event C with a CSP gate [31]

Table 3.9 shows the marginal probability table of event C. One of the most vital tasks in the CSP gate is to obtain the logic of how to calculate conditional probabilities P_{ij} in Table 3.10. Table 3.10 represents the probability of D to fail in the ith interval, considering that C has stopped working in the jth interval.

C	]0, △t]	]△t, T]	]T, +∞[
P(C)	P_1	P_2	P_3

Table 3.9: Marginal probability table of event C [31]

C	]0, △t]	]△t, T]	]T, +∞[
]0, △t]	0	0	0
]△t, T]	P_{21}	0	0
]T, +∞[	P_{31}	1	1

Table 3.10: Conditional probability table of spare D [31]

Finally, Table 3.11 shows the CPT of the CSP gate below. In addition, entries of unity in the identity matrix showed in Table 3.11 indicate that the failures of the CSP gate and standby component D happen at the same time. For example, when component D has a failure in $]0, \triangle t]$, the CSP gate fails in $]0, \triangle t]$.

D	]0, △t]	]△t, T]	]T, +∞[
]0, △t]	1	0	0
]△t, T]	0	1	0
]T, +∞[	0	0	1

Table 3.11: Conditional probability table of CSP gate [31]

Let's assume the failure rates are, the conditional probabilities of cold spare component D are computed using Eqs. from 3.35 to 3.37.

$$P(D = i|C = j) = P((i-1)\triangle < t_D < i\triangle | (j-1)\triangle < t_C < j\triangle) \tag{3.35}$$

$$P(D = i|C = j) = P(D = i|C > j) = \frac{P(D = i \cap D > j)}{P(D > j)} = \begin{cases} \frac{P(D=i)}{P(D>j)} & \text{if } i > j \\ 0 & \text{otherwise} \end{cases} \tag{3.36}$$

Since event D cannot have failed before the jth interval.

$$P(D = i|C = j) = \frac{\int_{(i-1)\triangle}^{i\triangle} \lambda e^{-\lambda t}\, dt}{\int_{(j-1)\triangle}^{j\triangle} \lambda e^{-\lambda t}\, dt} = e^{-\lambda\triangle(i-j)}(e^{\lambda\triangle} - 1) \tag{3.37}$$

3.3.5 Transformation of Static Fault Trees into Bayesian networks

In this section, the transformation of FT into BN is explained step by step. Before the transformation steps are explained, it should be remembered FT has some basic assumptions [3]:

1. Events have binary states (work/fail);
2. There is no dependency between components;
3. The components were connected by AND or OR gates;
4. The undesirable top event (TE) is analyzed.

Depending on the conversion rules for static gates, it is easy to convert an FT into a binary BN. There are some steps in order to transform standard FTA into BN, *i.e.*, a BN has two values: faulty and working. [37]:

1. Perform qualitative analysis of nodes for the fault tree depiction.
2. Specify the top event to be examined; clearly show all the various connections that are required to cause the top event.
3. In order to evaluate the probability of failure, utilize the existing data.
4. Assess the probability of failure of events.
5. Obtain the visual framework of BN through transforming SFT into a BN according to the recommended approach.

The transformation of fault tree for static OR gate and AND gate into BN is presented in 3.12 visually.

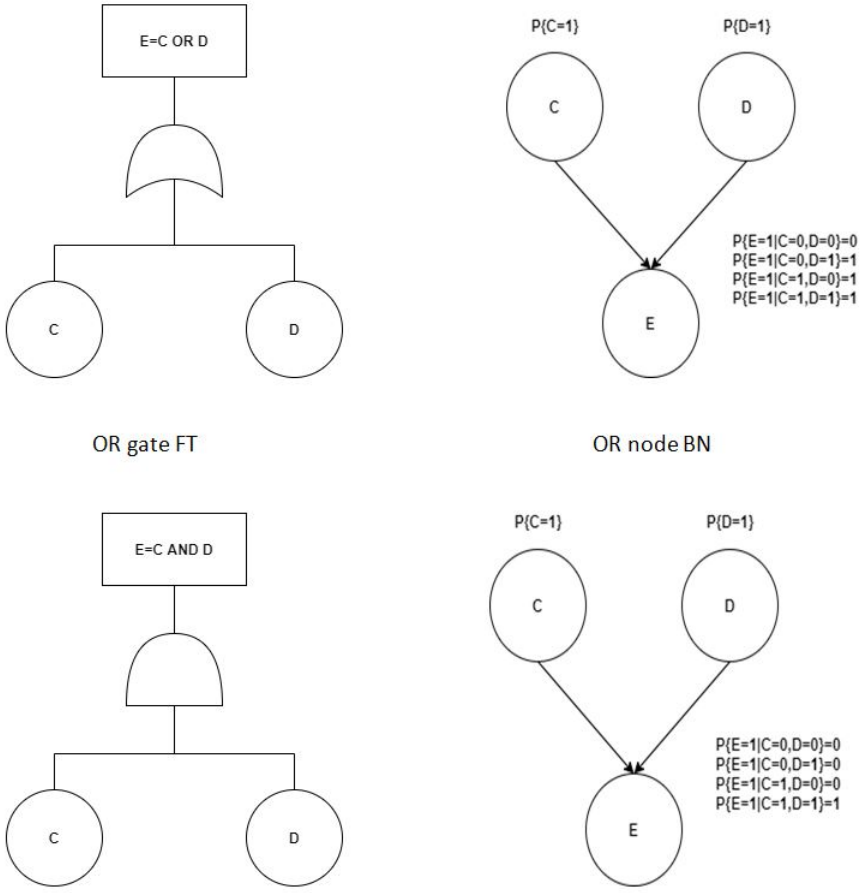


Figure 3.12: Transformation of static gates (OR and AND) into BN [3]

Throughout this chapter, the reliability analysis tools (*i.e.*, FT, MC and BN) utilized in the thesis were analyzed methodologically. It was also shown how these methods work mathematically and how their probability values are calculated. The contents of the probability tables created as a result of probability calculations are explained. Finally, how FT and MC methods are converted to BN and the necessary steps for the conversion are presented. The information explained in this section will be applied in the next section.

3.3.6 The Reliability Assessment by Bayesian Networks (RABN)

The RABN is a program established in C++ and MATLAB that enables the user to model a DTBN for many systems' features such as "OR" and "AND" logic gates, cold spare gate (CSP), and systems displaying switching components in order to develop a level 1 PRA. The RABN includes three steps to perform a reliability analysis described below:

- Creating parent and child nodes: in this step, the required nodes are created. There are two types of nodes; "component nodes", which represent the component's reliability, and "operator nodes", which establish relationships of dependencies among the components.
- Connecting nodes by arrows: a user can connect created nodes by arrows to model the entire system.
- Showing results: the user can see different results of the modeled system.

The RABN program allows users to simply model any given system with some useful features. By changing time steps in DTBN for a dynamic gate, more accurate results can be obtained at the end of the reliability analysis. The RABN program for modeling dynamic gates has some advantages compared to Netica software. Modeling large and complicated systems in Netica software takes a long time since every input must manually be entered in CPTs. Another limitation of Netica, or other commercially available software, is that when the number of time steps is increased, these time steps must also be entered manually. In contrast, RABN enables similar capabilities to the Netica with the advantages of a neutral dependency algorithm utilized to reduce the complexity of CPTs and to model a system in a short time.

Chapter 4

Reliability Applications

In this section, applications will be analyzed with the help of the RABN code (Appendix A) developed in MATLAB software using the reliability analysis mechanisms explained in the previous chapter, and the results will be presented. In the first application, the two cases analyzed by the SFT method are converted to the BN by following the necessary steps, and the results are compared. The main purpose of this application is to show that SFT analysis can be easily converted to the BN method, so the problems encountered in SFT analysis are eliminated with this conversion process. In the second application, a system including a cold spare gate is analyzed using Markov chain and BN methods. MC gives the exact solution for problems with constant failure rates, and they are easier to solve than dynamic fault trees for simple examples as presented in this chapter; however, this method has some limitations, such as the state-space explosion, and it is error-prone. The main purpose here is to show that the state-space explosion and error-prone problems encountered in MC analysis can be solved effectively by using the BN method. In addition, it is shown that more complex results can be obtained thanks to the BN analysis. In the last case of the study, a FLEX system developed to mitigate BDBEE in a nuclear power plant is examined. The safety analysis of this system has been done by the Fault tree method [19]. In this application, SFT analysis is transformed into BN, and the results are presented. For static problems, the results from SFT and BN are exactly the same. In the next step, DTBN analysis is performed using failure rates of system elements obtained from various sources, and the results are shown.

4.1 Conversion of Fault Trees into Bayesian Networks

The specific objective of the section is to assess static FT for AND and OR gates and show how to convert them into equivalent BN. For this aim, two different examples were created in both gates, and the unreliability results of SFT and equivalent BN were compared.

4.1.1 AND Gate Transformation

As discussed in chapter 3, in FTA, the AND gate is utilized to demonstrate that the top event has failed only if all basic components have failed. The AND gate and its converted version in BN are presented in Figure 4.1 for the first system. In the figure, A and B nodes represent two basic components that are connected by the AND gate. The failure rates of A and B components are shown in Table 4.1 with the unit of failure/hour (f/h).

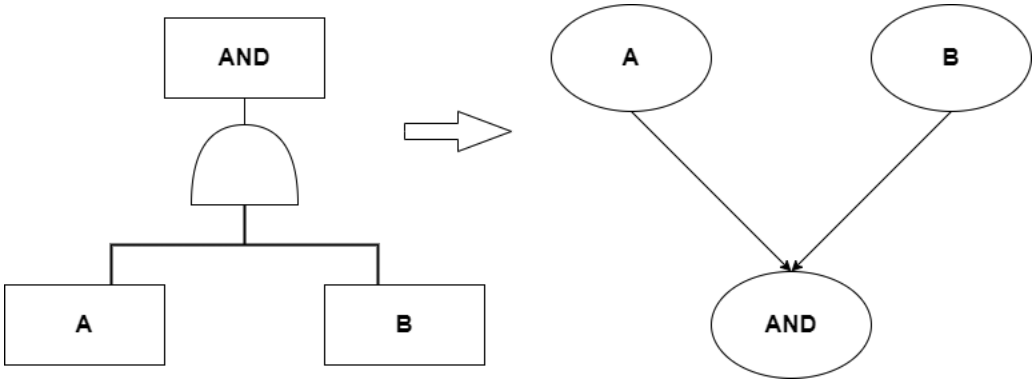


Figure 4.1: The AND gate and its equivalent BN

Basic component	Failure rate ($10^{-6} f/h$)
A	1
B	2

Table 4.1: Basic component failure rates or AND gate

There are some basic assumptions of the SFT that are utilized in the transformation mechanism from FT to BN:

1. Events have binary states (work/fail);
2. There is no dependency between components;
3. The components were connected by AND or OR gates;
4. The undesirable top event (TE) is analyzed.

The following convention is applied to use in FTA probability calculations. The components of the system that have binary status, e.g., if A, B = 1, the components are faulty, but if A, B = 0, then the components are working. In the quantitative analysis of FT, the probability value of each basic component needs to be computed. The analysis of the system components is performed at a provided mission time, so the failure probabilities of basic components at this time are calculated. The failure probability of a given system component is calculated as $P(A = 1, T) = 1 - e^{-\lambda_A T}$, where λ_A is the constant failure rate of component A. The following calculations show the failure probabilities of A and B components and the failure probability result of the AND gate at a given mission time, T=100,000 h. According to the results, the failure probability of the system is approximately 1.72 %.

$$P_A(failure) = 1 - e^{-\lambda_A T} = 1 - e^{-(10^{-6})100000} = 0.095162 \tag{4.1}$$

$$P_B(failure) = 1 - e^{-\lambda_B T} = 1 - e^{-(2*10^{-6})100000} = 0.181269 \tag{4.2}$$

$$P_{AND}(failure) = P_A.P_B = (0.095162)x(0.181269) = 0.017250 \tag{4.3}$$

For the AND gate conversion into the BN shown in Fig. 4.1, the prior probabilities of A and B components are displayed in both Tables 4.2 and 4.3 below for the working and failures statements. Table 4.4 provides the conditional probability table of the AND gate. As can be seen in Table 4.4, all inputs are 1s or 0s because of deterministic causal relationships that are represented by standard gates, *i.e.*, AND and OR.

A	Work	Fail
Probability	0.904838	0.095162

Table 4.2: Marginal probability table of node A

B	Work	Fail
Probability	0.818731	0.181269

Table 4.3: Marginal probability table of node B

A		Work		Fail	
B		Work	Fail	Work	Fail
AND	Work	1	1	1	0
	Fail	0	0	0	1

Table 4.4: Conditional probability table of AND gate

The probability result of the equivalent BN is presented in Figure 4.2 below by using the transformation rules and the CPT of the AND gate. Fig. 4.2 shows the BN created using Netica Norsys software version 5.18. Looking at Figure 4.2, it is shown that the failure probability of the child node is 1.72 %.

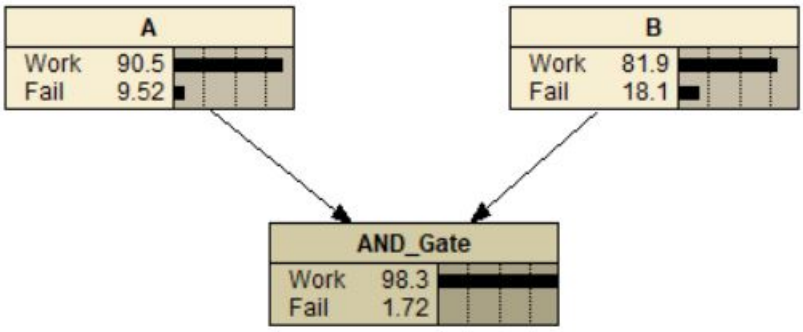


Figure 4.2: The BN for AND gate

Table 4.5 compares the top event unreliability results of FTA and its equivalent BN below.

	Top Event Unreliability
Fault tree for AND gate	0.017250
Equivalent BN	0.017250

Table 4.5: The comparison of FT and BN unreliability results for AND gate example

4.1.2 OR Gate Transformation

In FTA, the OR gate is utilized to demonstrate that the top event has failure only if all at least one basic component has a failure in the system. The OR gate and its converted version in BN are shown in Figure 4.3 for an example system. In figure 4.3, C and D nodes represent two basic components that are connected by the OR gate. The failure rates of C and D components are shown in Table 6 with the unit of f/h.

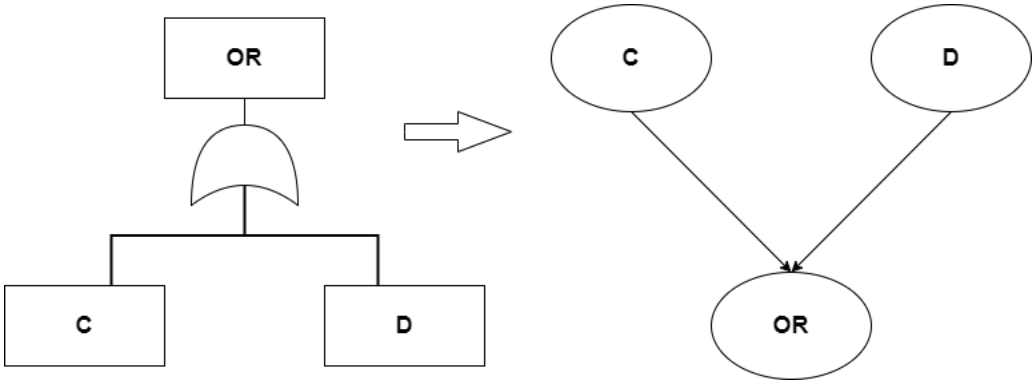


Figure 4.3: The OR gate and its equivalent BN

Basic component	Failure rate $(10^{-6} f/h)$
C	4
D	5

Table 4.6: Basic component failure rates for OR gate

The computations below explain the failure probabilities of C and D components and the failure probability result of the OR gate at a given mission time, T=100,000 h. According to the result, the failure probability of the system is approximately 59.3 %.

$$P_C(failure) = 1 - e^{-\lambda_C T} = 1 - e^{-(4*10^{-6})100000} = 0.329679 \tag{4.4}$$

$$P_D(failure) = 1 - e^{-\lambda_D T} = 1 - e^{-(5*10^{-6})100000} = 0.393469 \tag{4.5}$$

$$P_{OR}(failure) = P_C + P_D - P_C.P_D = 0.329679 + 0.393469 - (0.329679).(0.393469) = 0.593429 \tag{4.6}$$

In the BN, basic components C and D tables are specified marginal probabilities showing working and faulty situations, as shown in Tables 4.7 and 4.8. The CPT of the OR gate or child node in the BN is summarized in Table 4.9.

C	Work	Fail
Probability	0.670321	0.329679

Table 4.7: Marginal probability table of node C

D	Work	Fail
Probability	0.606531	0.393469

Table 4.8: Marginal probability table of node D

As shown in Fig. 4.4, the system fails (OR = 1) when either C or D fails. The results of the failure probability for equivalent BN were calculated in Netica and can be seen in Figure 4.4 below. The failure probability of the child node is 59.3 %.

The unreliability results of FT and equivalent BN can be compared in Tables 4.5 and 4.10

C		Work		Fail	
D		Work	Fail	Work	Fail
OR	Work	1	0	0	0
	Fail	0	1	1	1

Table 4.9: Conditional probability table of OR gate

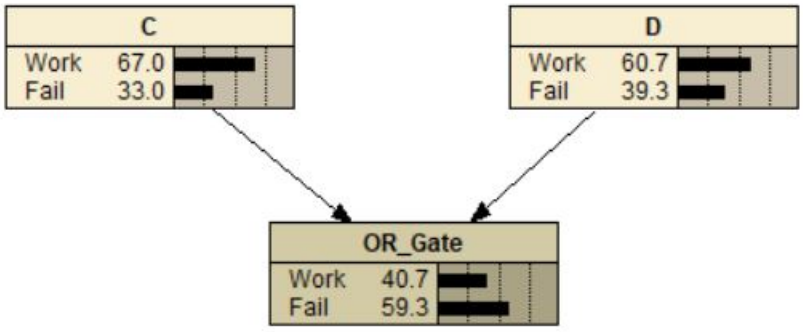


Figure 4.4: The BN for OR gate

for AND and OR gates, respectively. According to the results shown in both tables, it is clear that an SFT can be transformed into a BN by applying basic conversion rules. This can also be demonstrated using mathematic equations.

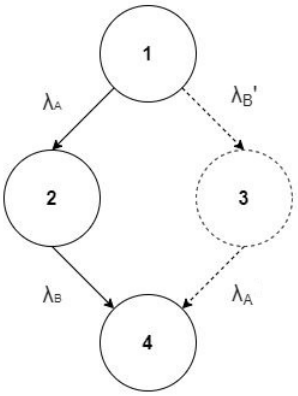
	Top Event Unreliability
Fault tree for OR gate	0.593429
Equivalent BN	0.593429

Table 4.10: The comparison of FT and BN unreliability results for a fictitious OR gate

4.2 Cold Spare Gate Example

Spare gates are utilized to design subsystems with redundant elements in which a stopped working main element is quickly changed by its spares, one after another. In this application, a system containing a CSP gate is analyzed using MC and BN methods at a given mission time. The reliability probabilities of the two methods are compared in a table for the components having the same failure rates, i.e., 0.1, 0.01, and 0.001 h^{-1}.

Figure 4.5 below illustrates an MC and its possible states for a CSP gate that has one primary and one standby component. In the example, there are two components, so the system will have $2^2 = 4$ possible states, as can be seen in the below table.



	A	B
State 1	Work	Work
State 2	Fail	Work
State 3	Work	Fail
State 4	Fail	Fail

Figure 4.5: Markov chain and possible states for CSP gate [31]

In order to compute the reliability of the MC system, the probability of state 4 is calculated:

$$P_4 = 1 - e^{-\lambda T} - \lambda T e^{-\lambda T} \tag{4.7}$$

Then, the system reliability is computed:

$$R_S(T) = 1 - P_4 = e^{-\lambda T} + \lambda T e^{-\lambda T} \tag{4.8}$$

where $\lambda = \lambda_A = \lambda_B$ =the failure rate of component, since failure rates of two components are the same. $\lambda_B^{'}$ = failure rate of standby component = 0. State 3 does not exist in practice since we can disregard the probability of failure during standby and T=mission time. In the example, we will use T=10 h, $\lambda = 0.1, 0.01$, and 0.001 h^{-1}.

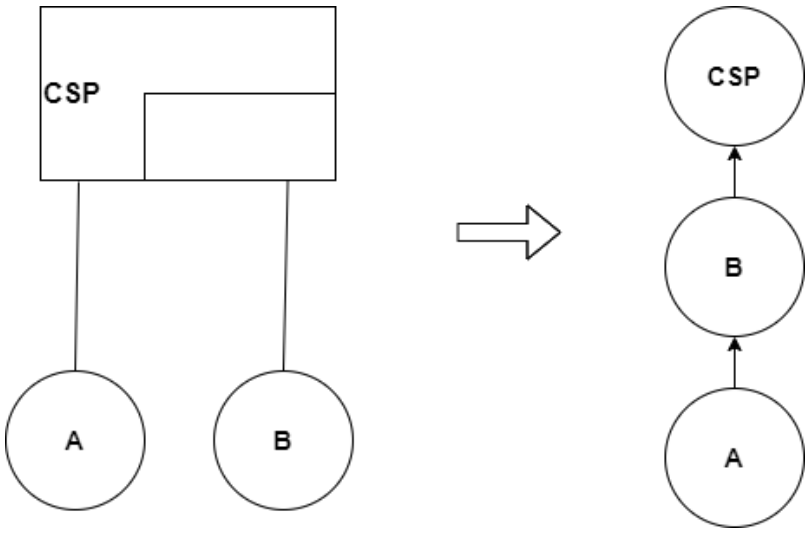


Figure 4.6: Transformation of CSP gate into DTBN [5]

After CSP gate mapping into DTBN, the failure probabilities of primary and standby components are calculated by using the equations below to generate a marginal probability table and CPTs. The below equation shows component A failed in the lth interval and $t_A \in](l-1)\triangle, l\triangle]$:

$$P(A=l) = P((l-1)\triangle < t_A < l\triangle) = \int_{(l-1)\triangle}^{l\triangle} f_A(t)\,dt = F_A(l\triangle) - F_A((l-1)\triangle) \tag{4.9}$$

where t_A = element A failure time, F_A = cumulative distribution, $\triangle$ = interval length, and n = time granularity [13]. For the n+1 state, which means the component did not fail during the mission time:

$$P(A=n+1) = P(t_a > T) = \int_{T}^{\infty} f_A(t)\,dt = 1 - F_A(T) \tag{4.10}$$

The prior probabilities of component A is indicated in Table 4.11 for T=10 h, $\triangle = 2$ so that the time granularity, $n = T/\triangle = 10/2 = 5$.

A	]0, 2]	]2, 4]	]4, 6]	]6, 8]	]8, 10]	]10, ∞[
P(A)	0.1813	0.1484	0.1215	0.0995	0.0814	0.3679

Table 4.11: Prior probability table of component A

Table 4.12 shows the conditional probabilities P_{kl} representing that component B fails in the kth interval given that A failed in the lth interval. It should be noted that the probabilities shown in Table 4.12 belong to two components systems having the same λ (failure rate) = 0.1 h^{-1}.

$$P_{kl} = P(B = k|A = l) = P((k-1)\triangle < t_B < k\triangle | (l-1)\triangle < t_A < l\triangle) \tag{4.11}$$

The probability calculation of component B is explained in the equation below. The spare component B cannot have a failure before the main component A. Hence,

$$P(B = k|A = l) = P(B = k|B > l) = \frac{P(B = k \cap B > l)}{P(B > l)} = \begin{cases} \frac{P(B=k)}{P(B>l)} & \text{if } k > l \\ 0 & \text{otherwise} \end{cases} \tag{4.12}$$

If the computation continues for k > l:

$$P(B = k|A = l) = P(B = k|B > l) = \frac{P(B = k \cap B > l)}{P(B > l)} = \frac{\int_{(k-1)\triangle}^{k\triangle} f_B(t)\,dt}{\int_{l\triangle}^{\infty} f_B(t)\,dt} = \frac{F_B(k\triangle) - F_B((k-1)\triangle)}{1 - F_B(l\triangle)} \tag{4.13}$$

$$P(B = k|A = l) = \frac{\int_{(k-1)\triangle}^{k\triangle} \lambda e^{-\lambda t}\,dt}{\int_{l\triangle}^{\infty} \lambda e^{-\lambda t}\,dt} = e^{-\lambda\triangle(k-l)}(e^{\lambda\triangle} - 1) \tag{4.14}$$

A	]0, 2]	]2, 4]	]4, 6]	]6, 8]	]8, 10]	]10, ∞[
]0,2]	0	0.1813	0.1484	0.1215	0.0995	0.4493
]2,4]	0	0	0.1813	0.1484	0.1215	0.5488
]4,6]	0	0	0	0.1813	0.1484	0.6703
]6,8]	0	0	0	0	0.1813	0.8187
]8,10]	0	0	0	0	0	1
]10,∞]	0	0	0	0	0	1

Table 4.12: CPT of spare B

The conditional probability table of CSP gate is provided in Table 4.13. It can be seen from Table 4.13 that the CSP gate has a unit matrix since the spare component B and the CSP gate have failure at the same time.

B	]0, 2]	]2, 4]	]4, 6]	]6, 8]	]8, 10]	]10, ∞[
]0,2]	1	0	0	0	0	0
]2,4]	0	1	0	0	0	0
]4,6]	0	0	1	0	0	0
]6,8]	0	0	0	1	0	0
]8,10]	0	0	0	0	1	0
]10,∞]	0	0	0	0	0	1

Table 4.13: CPT of CSP gate

Figure 4.7 displays DTBN reliability analysis results calculated by $Netica^{TM}$ software for three different failure rates at the given mission time. In Netica software, all probability values within marginal and conditional probability tables must be entered manually. It is one of the main drawbacks of using this software since care should be taken every time when entering these values. Another limitation of this software is that modeling a large and complex system with DTBN may take a long time and effort. In contrast, the RABN code allows similar capabilities to the Netica software with modeling DTBN for complex systems since it enables to change time steps, the number of components, and system configuration in seconds. Moreover, the RABN code utilizes a neutral dependency algorithm so that the number of parameters within CPTs can be reduced for large systems. Nevertheless, it can be

seen from the data in Figure 4.7 that with the decreasing of component failure rates, system reliability will increase at the given mission time.

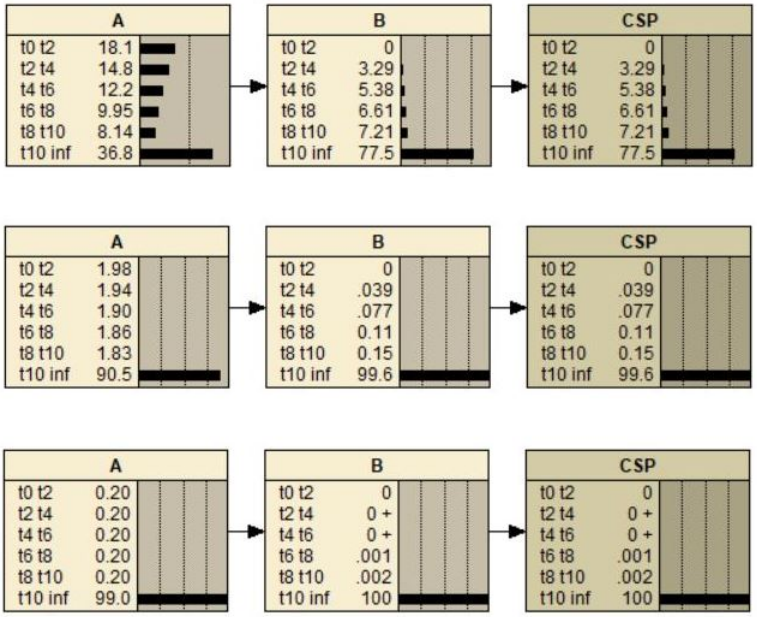


Figure 4.7: DTBN results for two-component system having $\lambda = 0.1(top), 0.01(middle)$, and $0.001(bottom)h^{-1}$

The comparison of DTBN and MC reliability results for the CSP gate is summarized in Table 4.14 for three different failure rates, i.e. $\lambda = 0.1$, 0.01, and 0.001 h^{-1}. One can see in Table 4.14, the results of the DTBN analysis as developed by Boudali and Dugan [5] approach MC results with the increasing time granularity. It is worth highlighting that the relative error is significantly smaller for lower failure rates that are closer to real cases.

n(time granularity)=T/△ = 10/2 = 5			
$\lambda(h^{-1})$	DTBN	MC	Relative Error (%)
0.1	0.77512705	0.73575888	5.3507
0.01	0.99623205	0.99532115	0.0915
0.001	0.99996023	0.99995033	9.9076e-4
n(time granularity)=T/△ = 10/1 = 10			
$\lambda(h^{-1})$	DTBN	MC	Relative Error (%)
0.1	0.75478162	0.73575888	2.5855
0.01	0.99577509	0.99532115	0.0456
0.001	0.99995528	0.99995033	4.9521e-4
n(time granularity)=T/△ = 10/0.1 = 100			
$\lambda(h^{-1})$	DTBN	MC	Relative Error (%)
0.1	0.73760442	0.73575888	0.2508
0.01	0.99536641	0.99532115	0.0045
0.001	0.99995082	0.99995033	4.9507e-5

Table 4.14: Comparison of DTBN and MC reliability results for CSP gate

From Table 4.14 and the graphs are below shown in Figure 4.8, 4.9, and 4.10, we can see that DTBN allows the designer or analyst to change n (time granularity) so that more accurate results can be obtained by increasing the time steps. Figures 4.8 - 4.10 present the comparison of DTBN and MC visually for n=5 and 10. The sensitivity analysis of the system is obtained by increasing the number of time steps. As can be seen in the graphs, more time steps mean the reliability result of DTBN approximates to the reliability result of MC.

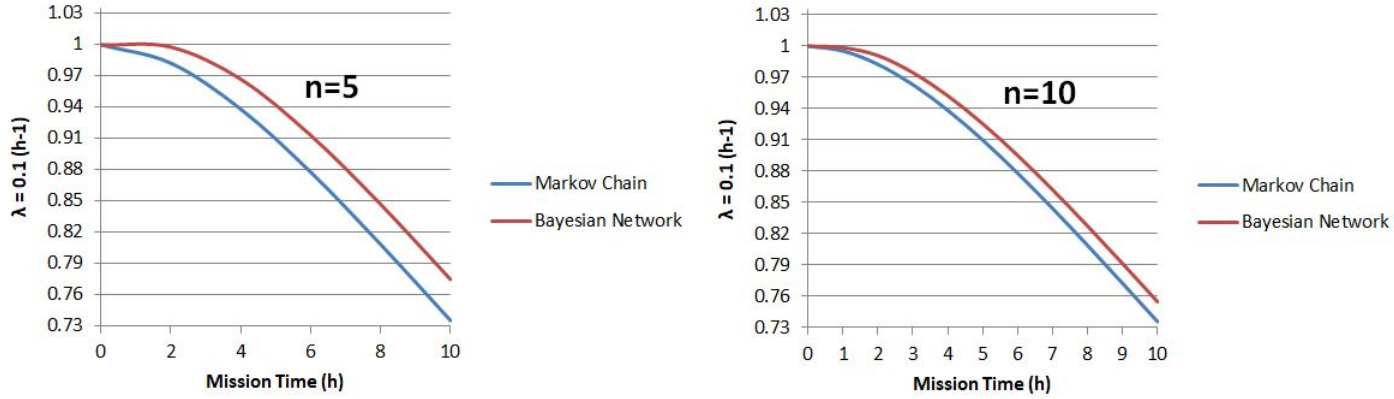


Figure 4.8: The comparison of DTBN and MC reliability results graphically for $\lambda = 0.1h^{-1}$

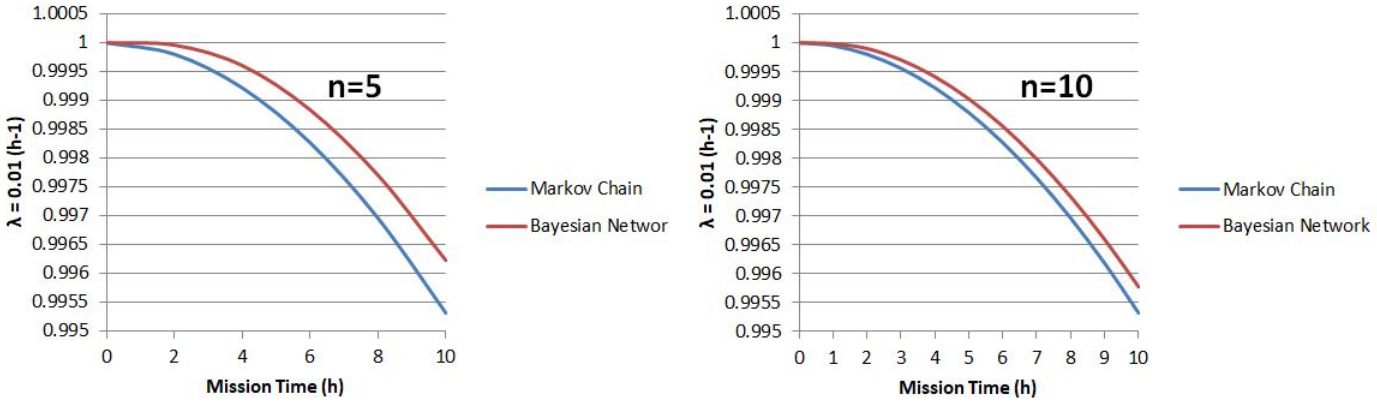


Figure 4.9: The comparison of DTBN and MC reliability results graphically for $\lambda = 0.01h^{-1}$

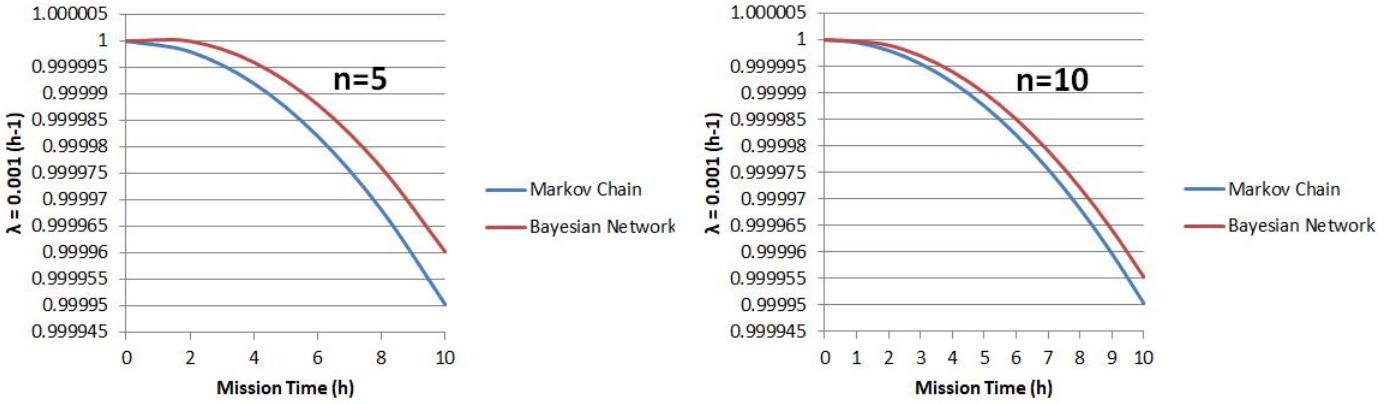


Figure 4.10: The comparison of DTBN and MC reliability results graphically for $\lambda = 0.1$, 0.01, and 0.001 h^{-1}

One of the limitations of utilizing DTBN is that performing a reliability analysis can be computationally expensive compared to FT and MC methods. As seen in Table 4.15, the computation time of DTBN increases more than the computation of MC with the increasing time steps. In order to utilize DTBN effectively, a neutral dependency algorithm reducing computation time as much as possible is applied in the next case study for a FLEX system.

n (time granularity)	Markov Chain computation time (s)	DTBN computation time (s)
100	0.004	0.073
1000	0.006	1.555
5000	0.008	65.437

Table 4.15: Comparison of Markov chain and DTBN computation times

One advantage of DTBN, as mentioned in the previous chapters, is the ability to model a system consisting of dependencies between system components. To exemplify it, Table 4.16 shows the CPT of the CSP gate that was modified from in the previous application. As seen in Table 4.16, there are probability values in each time interval instead of "1s" and "0s" since these probability values represent the dependency between the primary and spare components. These numbers were randomly created to show how the dependency between system components can be implemented in the system reliability. Fig. 4.11 shows the reliability analysis of the system.

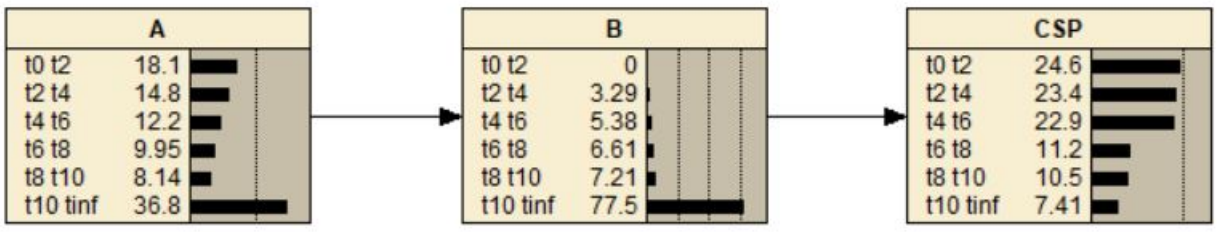


Figure 4.11: DTBN results for a two-component system having dependency $\lambda = 0.1h^{-1}$

A/B	]0, 2]	]2, 4]	]4, 6]	]6, 8]	]8, 10]	]10, ∞[
]0,2]	0.2	0.1	0.25	0.2	0.15	0.1
]2,4]	0.3	0.15	0.15	0.05	0.25	0.1
]4,6]	0.1	0.2	0.3	0.15	0.1	0.15
]6,8]	0.4	0.1	0.1	0.1	0.1	0.2
]8,10]	0.15	0.25	0.1	0.25	0.1	0.15
]10,∞]	0.25	0.25	0.25	0.1	0.1	0.05

Table 4.16: CPT of CSP gate

4.3 Case Study: Implementation of Bayesian Network into FLEX Strategy

The FLEX system chosen for the case study was designed to reduce the damage to the reactor core due to two important events occurring in nuclear reactors, i.e., the station blackout (SBO) and large loss of coolant accident (LOCA). SBO event happens because of complete loss of alternating current (AC) power and has a huge effect on nuclear reactor safety [59]. Another event that has a significant contribution to the NPP risk is the LOCA, which happens due to the loss of coolant in the reactor core [22]. However, it should be noted that only the SBO event is examined here, and its FLEX strategy is analyzed by transforming the givens FT into BN. Then in section 4.3.2, a DTBN is proposed.

4.3.1 Static Fault Tree Analysis and Equivalent Bayesian Network Solution

In this case study, an auxiliary water storage unit used for water injection into the primary and secondary loops was proposed as a FLEX strategy. Fig. 4.12 shows the additional water storage system's schematic representation, including a water supplier, pipes, a portable

electric pump, valves, and a boron tank that fulfills the conditions necessary for proper operation [19].

As shown in Fig. 4.12, two main lines are connected to the system of a nuclear reactor. Line 1 is connected to the secondary loop and is utilized for mitigation of the SBO event; Line 2 is connected to the primary loop and is utilized to mitigate the large LOCA. Hence, the supplementary water storage unit works in both ways, i.e., SBO mitigation and large LOCA mitigation. Fig. 4.13 displays the FT for the SBO mode, with the top event reflecting the failure of the developed framework to supply water to the Steam Generator.

In the case study, it was shown that the fault tree of the proposed system could be smoothly converted into a Bayesian Network. One can notice that the FT of the proposed FLEX system consists of only OR gates. Therefore, following the steps described before, the BN for this system was created, as displayed in Fig. 4.14.

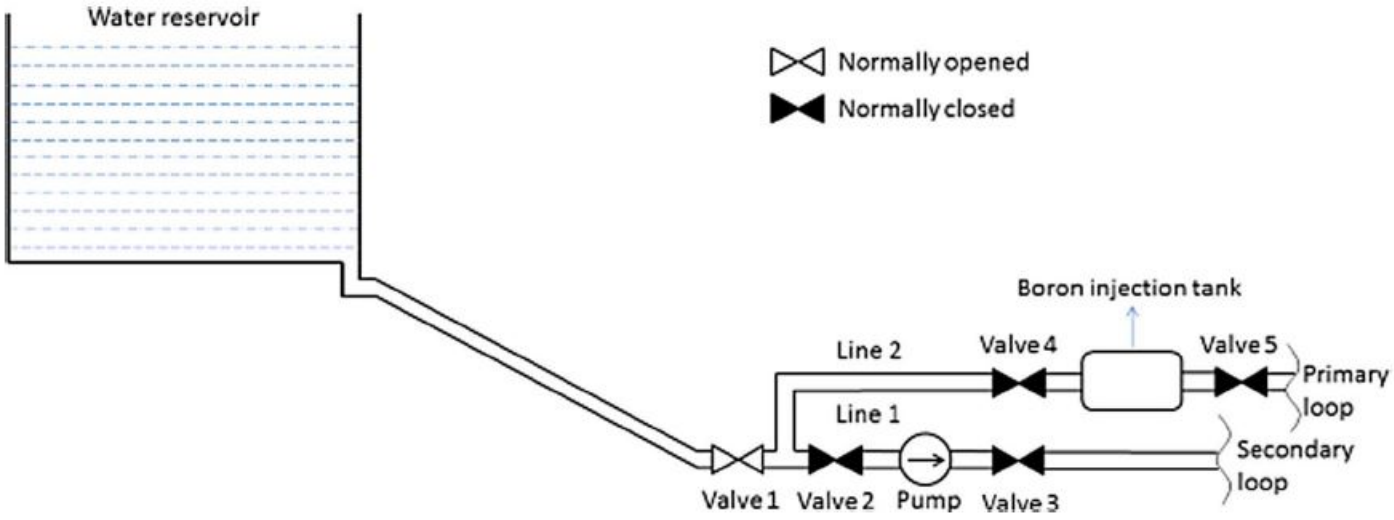


Figure 4.12: Schematic of supplemental water storage system [19]

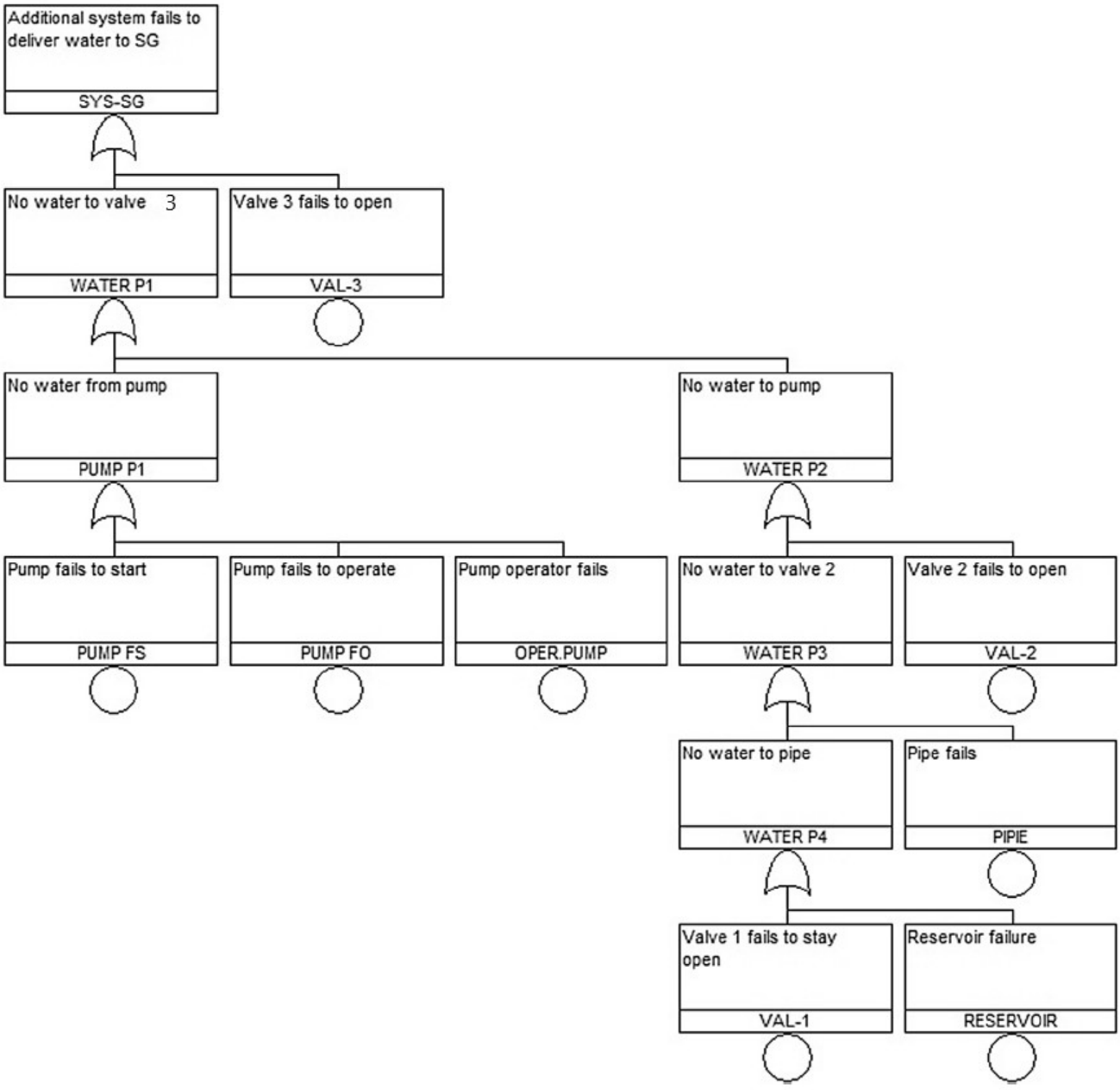


Figure 4.13: Qualitative analysis by fault tree for the proposed system adapted from [19]

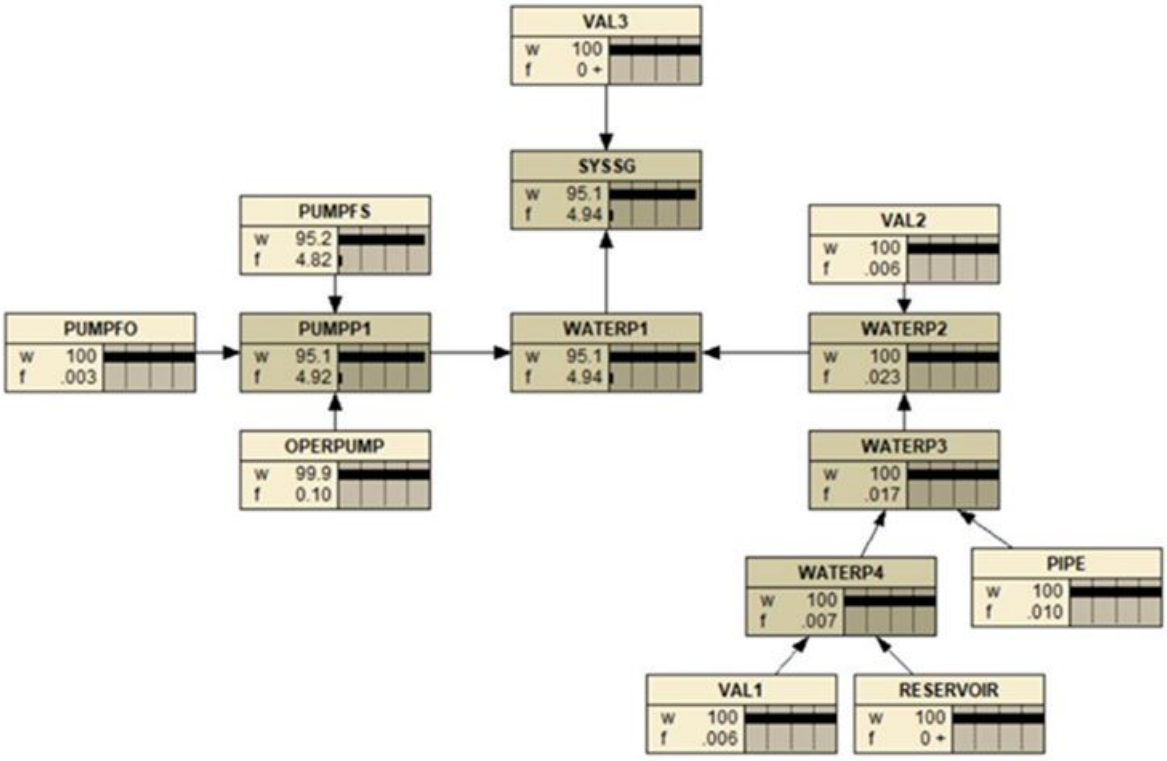


Figure 4.14: Bayesian network of the water storage system used as FLEX

As shown in Table 4.17, water storage system data were used in order to compare the results of Fault Tree and Bayesian Network.

Component failure mode	Unreliability
Pump failure to start	4.82E-2
Pump failure to run	3.00E-5
Pump operator failure	1.00E-3
Valve failure to stay open	6.18E-5
Valve failure to close	4.24E-6
Reservoir failure	1.00E-5
Pipe failure	1.00E-4

Table 4.17: Reliability parameters used in the fault tree and Bayesian network [19]. The sources of the unreliability values are given by Ref. [19], *apud.*

Table 4.18 shows the comparison of Fault Tree and Bayesian Network for SBO mitigation. According to the results of FT and BN for the proposed system, a Fault Tree can be transformed into a Bayesian Network, obtaining the exact same results.

	Top Event Unreliability
Fault tree for SBO mitigation	4.94E-02
Bayesian network for SBO mitigation	4.94E-02

Table 4.18: The Comparison of FT and BN Results for the Proposed System

4.3.2 Discrete-Time Bayesian Network Analysis

Because of its intrinsic modeling restriction, Static Fault Tree cannot be used for more complex structures that may involve dependent events. Therefore, Static BN also cannot model dynamic gates. For the next step, the RABN code that includes Discrete-Time Bayesian Network is implemented by using component failure rates that are stemmed from developed NPP-specific data sources, articles, and research papers in order to compute the reliability of the top event over time.

The quantification of the DTBN involves the calculation of a probability for each component. Because the simulation is carried out at a specified mission time t, so the failure probabilities of the primary elements at the time are calculated by using the failure rates of each element provided. The fundamental theory is that basic element failures are exponentially distributed, the unreliability of basic component is

$$P(C = faulty, t) = 1 - e^{-\lambda_c t} \tag{4.15}$$

where

- $P(C = faulty, t)$ = the unreliability of component C
- $\lambda_c(\frac{f}{h})$ = failure rate
- $t(h)$ = mission time

In order to perform reliability analysis of the case study by using DTBN, the methods explained in studies of Boudali and Dugan [5] and Khadzad [31] are applied to the FLEX system. In the case study, the system only includes static OR gates in the given fault tree. It is not easy to utilize static gates to design these complicated FLEX systems, so these gates are also required to be designed as discrete-time gates and be adapted into DTBN. Moreover, the CPTs of static gates could become very complex in the foregoing steps. Due to the complexity of large CPTs, the neutral dependency methodology (Fig. 4.15) is also used to minimize the number of multi-dimensional CPTs [31]. It ought to be mentioned that during the neutral dependency mechanism, because of modifications in CPT, the outcomes from probability updating (i.e., posterior probabilities) may not show the system's real behavior. Therefore, a neutral dependency algorithm can be taken into consideration to lessen the dimensions of CPT if DTBN is implemented to get the unreliability of the system, which is true for the present case.

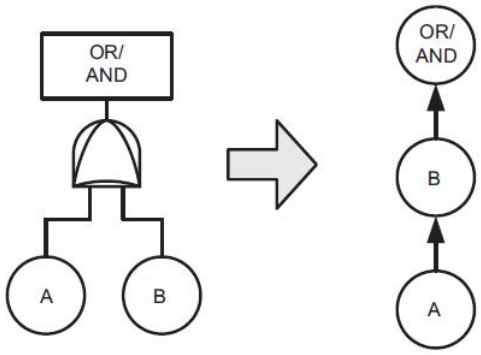


Figure 4.15: The conversion of static gates into BN utilizing neutral dependency [31]

All of the water storage system components have an exponential reliability function since they have constant failure rates. The failure rates of all basic elements comprising the FLEX system are displayed in Table 4.19. For the failure rates of pipe failure and pump failure to run, we modify and use the values shown in the table.

Component failure mode	Failure rate (f/h)	Source
Pump failure to start	1.0E-6	IAEA [56]
Pump failure to run	8.0E-6	IAEA [56]
Pump operator failure	1.00E-7	Bertucio [2]
Valve failure to stay open	2.37E-6	IEEE [4]
Valve failure to close	2.37E-6	IEEE [4]
Reservoir failure	1.0E-9	Yan and Lan [16]
Pipe failure	4.45E-6	SKI [36]

Table 4.19: Reliability parameters used in DTBN

To solve the reliability of FLEX equipment shown in Fig. 4.12, the SFT is transformed into the corresponding DTBN structure in Fig. 4.15. As can be seen in Fig. 4.14, due to the complexity of CPTs, static OR gates along the right and left branches are used to develop the DTBN in Fig. 4.16 by means of neutral dependency formalism. Further, the standard divorcing method [41] is implemented to connect WATER-P2 and PUMP-P1 to the system.

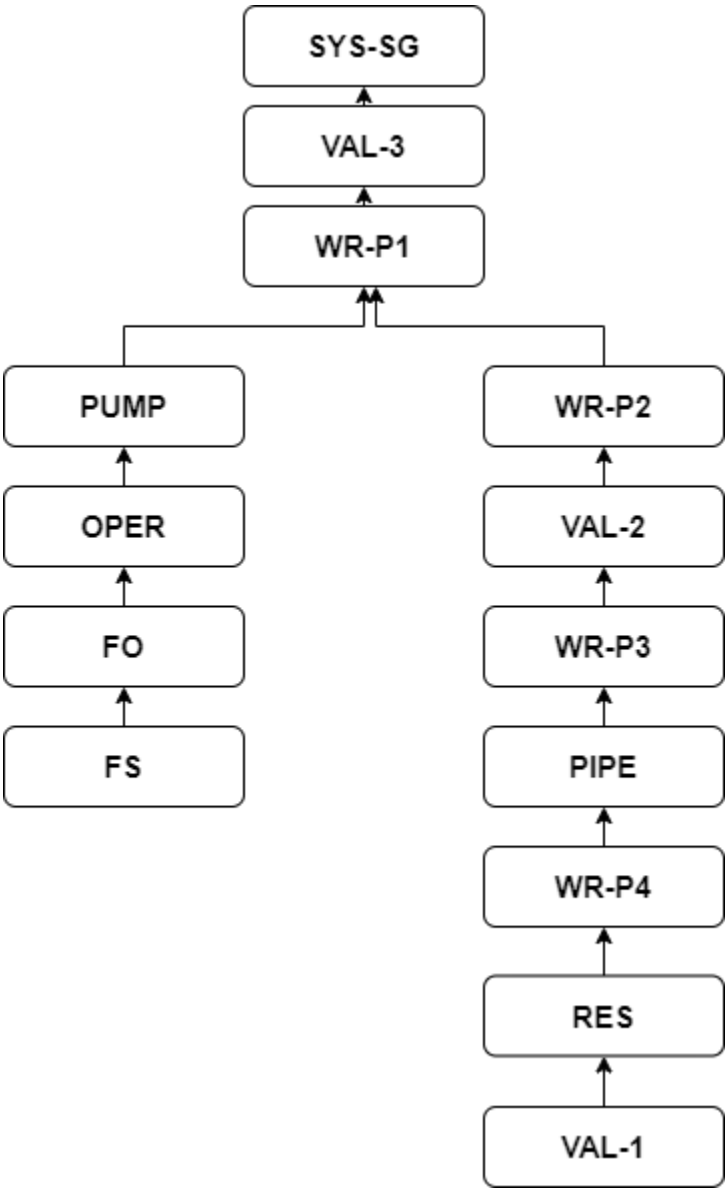


Figure 4.16: Bayesian network of the water storage system

The DTBN results in Table 4.20 have been set for different time granularities (n) (i.e., the number of intervals in that the timeline is divided) and T (mission time) = 10000 h. The reliability of the system refers to the top event (i.e., node SYS-SG) success probability at the given mission time. Additionally, the final result is provided graphically in Figure 4.17 as a curve that suggests the system reliability as a feature of the given mission time. As shown in

Table 4.20, the increase in time granularity does not change the final reliability result since the FLEX system in the case study consists of only static gates (i.e., OR gate).

n (time granularity)	Computation time (s)	Reliability
1	0.908	0.813337
100	48.353	0.813337
250	215.058	0.813337
500	1183.250	0.813337

Table 4.20: The Water Storage System Reliability Results

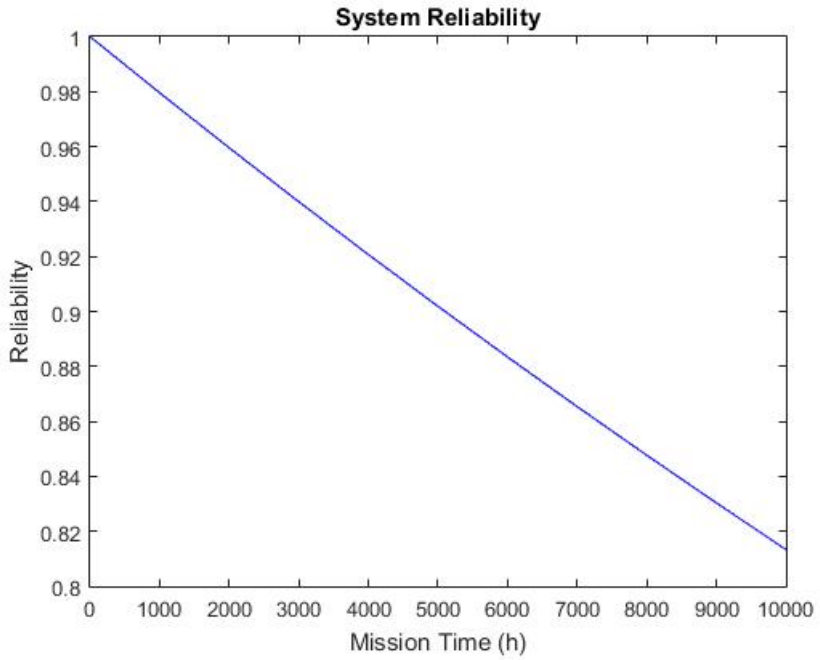


Figure 4.17: System reliability curve at given mission time

In the further step, a spare valve is connected to the FLEX system by using a CSP gate as shown in Fig. 4.18 so that the system reliability can be improved. It should be noted that the primary and spare valves have the same failure rates in this application. When DTBN is applied to a system including dynamic gates (CSP gate), the increase in time step affects the final reliability result so that more accurate results can be calculated for

the system by changing the time step, as shown in Table 4.21. Furthermore, although all system components are independent in this application, the CPTs of DTBN allow us to add dependency between system components.

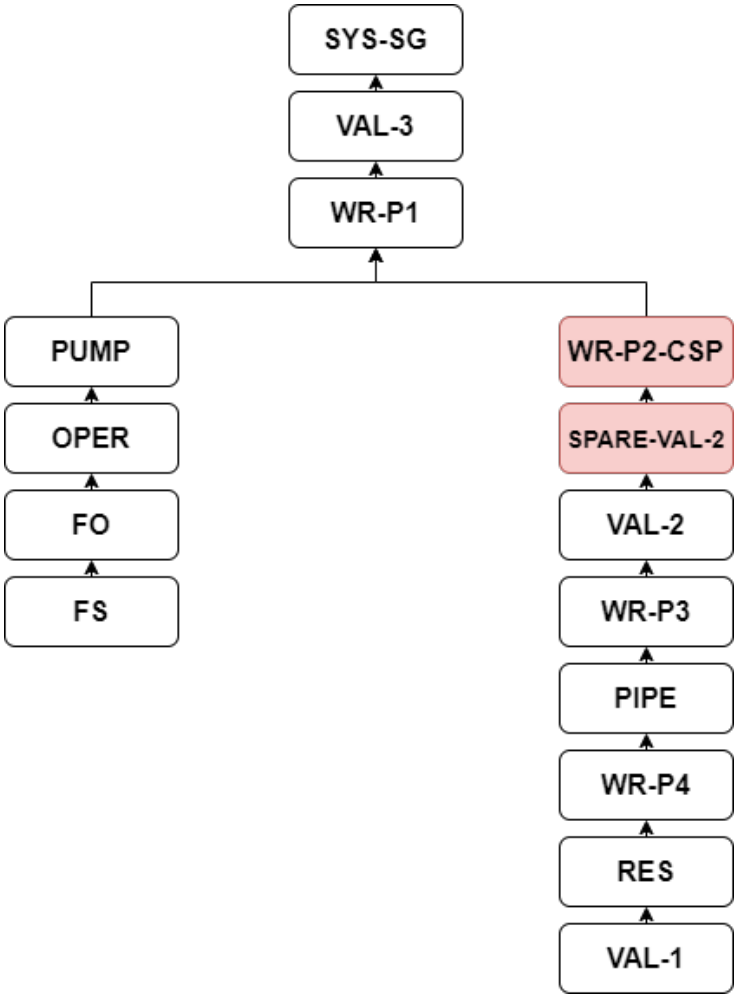


Figure 4.18: Bayesian network of the modified water storage system

n (time granularity)	Computation time (s)	Reliability
1	0.599	0.976578
100	36.288	0.974620
250	146.385	0.974608
500	486.364	0.974605

Table 4.21: The Modified Water Storage System Reliability Results

4.3.3 Sensitivity Analysis

Sensitivity analysis enables the analyst to measure the importance of each of the system's parts as well as the effect on the improvement of element reliability will affect the top event reliability. In the case study, the Fussell-Veselly importance factor will be quantified to understand system components' contributions to total reliability.

The Fussell-Veselly importance factor can be calculated by using the below equation,

$$FV_i = P_i/P_t \tag{4.16}$$

where, P_i is the basic event unreliability result, and P_t is the system unreliability result. FV has values between 0 and 1.

As seen in Table 4.22, the pipe has the biggest importance factor, so it is the system component that makes the greatest contribution to the system unreliability.

Basic Component	I (component)
VAL-1	0.148869
RESERVOIR	6.281E-5
PIPE	0.793406
VAL-2	0.148869
PUMP-FS	0.062814
PUMP-FO	0.502513
OPER-PUMP	0.006281
VAL-3	0.148869

Table 4.22: Fussell-Vesely importance factor

In this chapter, first of all, the FT reliability analysis of a system having two components was transformed to BN for AND and OR gates, respectively. The assumptions and probability calculations used during this transformation are explained. According to the unreliability results obtained, an FT with static gates can easily be converted to BN. In the second application, the conversion of the MC method used for the reliability analysis of dynamic systems to DTBN is examined. The reliability results of the two-element system with cold spare gate are obtained and compared by two methods. In this application, DTBN allows us to approximate the real case obtained by MC analysis by increasing time granularity for the given failure rates. In the last application, a system designed as a FLEX strategy in nuclear reactors and whose reliability analysis is made by the FT method is examined. The same unreliability result was obtained for this system converted to the BN method. The main purpose of these applications is to overcome the problems encountered in FT and MC methods using the BN method and to show that this method is an efficient tool for the reliability analysis of FLEX strategies used in nuclear reactors.

Chapter 5

Conclusions and Future Work

In this thesis, the reliability analysis of FLEX strategies developed and applied to reduce the risky situations caused by BDBEEs in nuclear reactors was examined. It was explained the advantages and disadvantages of some of the tools (*i.e.*, FT, MC, and BN) used for reliability analysis. Among these investigated reliability tools, it was shown that the BN method could be a more suitable tool for performing the reliability analysis of FLEX equipment since this method has some advantages compared to the other two methods. This thesis shows the effectiveness and applicability of the Bayesian network in a wide range of areas, especially in nuclear power plants, to conduct the reliability analysis.

BN method can be preferred in the reliability analysis of FLEX strategies when compared to the FT and MC methods. FT is a powerful and widely utilized technique in reliability evaluation; however, the technique has some drawbacks such as having insufficient capacity to model large systems that consist of dependencies between system components, and inability to effectively deal with the basic components' failure distributions. MC analysis is another popular method for reliability evaluations of systems. However, this technique has two major concerns; state-space explosion and error-prone. With the increasing number of system components, it could be really hard to analyze system reliability by utilizing MC. In this thesis, BN is preferred to analyze the reliability of FLEX components since BN is a powerful method to overcome the drawbacks of FT and MC and has some advantages. One of these

advantages is that BN can update probabilities of each component in a system since it is easily integrated with Bayesian inference. Another advantage of using BN is that BN can model large and complex systems even if there are dependencies between system components.

In Chapter 4, it was shown that FT and MC methods could be transformed into BN by following the conversion rules. In the first application, the FTA of two-component system that has static gates converted into BN and the same reliability results were obtained in reliability analysis. It can be concluded that by converting FT into BN, the problems encountered in FTA can be easily solved. In the second application, a system that has a cold-spare gate was analyzed by using MC and mapped into DTBN. This application showed that DTBN allows us to overcome the drawbacks of MC, such as the state-space explosion and error-prone, and to approximate real case reliability results by changing time steps. In the last application, FLEX equipment analyzed by using FT was mapped into BN, and the same reliability results were obtained at the end of the analysis. As a result of this application, BN is an appropriate method to model large and complex FLEX systems since this model allows us to add neutral dependency between components to reduce the complexity of any system. A Discrete-Time Bayesian Network (DTBN) was then proposed to further develop the reliability studies of FLEX systems. The DTBN can be easily modified to include dependencies, non-constant failure rates, or other more complex dynamic interactions.

Additionally, FLEX equipment represents new challenges in the modeling of Performance Influencing Factors (PIFs) of a critical task, increasing the probability of a human error. The human error probability (HEP) associated with transportation, placement, connection, and operation of the FLEX equipment (e.g., water tank, portable pump, power supplier) needs to be incorporated in the reliability calculation of these systems. BNs are a powerful tool to model the HEP using PIFs and then are suggested in future work.

Another challenge faced in NPPs is common cause failures. A severe external event can affect

not only nuclear reactor equipment but also FLEX systems that are designed to decrease risky events for reactor safety. As an example, we can assume in the FLEX system, explained in the previous chapter that, all valves in the system were powered by the same battery. If the battery failure due to an external event, the system will fail due to a common cause. Modeling of common cause failure in FLEX systems using DTBN is also suggested in future work.

Bibliography

[1] Jean-François Aubry and Nicolae Brînzei. "Stochastic hybrid automaton". In: *Systems Dependability Assessment: Modeling with Graphs and Finite State Automata* (2015), pp. 105–120.

[2] Robert C Bertucio, Jeffrey A Julius, and WR Cramond. *Analysis of core damage frequency: Surry, Unit 1 internal events*. Tech. rep. Nuclear Regulatory Commission, 1990.

[3] Andrea Bobbio et al. "Improving the analysis of dependable systems by mapping fault trees into Bayesian networks". In: *Reliability Engineering & System Safety* 71.3 (2001), pp. 249–260.

[4] LE Booth. "IEEE Guide to the collection and presentation of Electrical, Electronic, Sensing Component and Mechanical Equipment Reliability data for nuclearpower generating stations". In: *IEEE Std 500-1984* (1983).

[5] Hichem Boudali and Joanne Bechta Dugan. "A discrete-time Bayesian network reliability modeling and analysis framework". In: *Reliability Engineering & System Safety* 87.3 (2005), pp. 337–349.

[6] Ferdinando Chiacchio et al. "SHyFTA, a Stochastic Hybrid Fault Tree Automaton for the modelling and simulation of dynamic reliability problems". In: *Expert Systems with Applications* 47 (2016), pp. 42–57.

[7] Nuclear Regulatory Commission et al. *Severe accident risks: An assessment for five US nuclear power plants: Appendices A, B, and C*. Tech. rep. Nuclear Regulatory Commission, 1990.

[8] US Nuclear Regulatory Commission et al. *Issuance of Order To Modify Licenses With Regard To Requirements For Mitigation Strategies For Beyond-design-basis External Events*. Tech. rep. EA-12-049, March 12, 2012.

[9] US Nuclear Regulatory Commission. *Reactor safety study: An assessment of accident risks in US commercial nuclear power plants*. Vol. 2. US Nuclear Regulatory Commission, 1975.

[10] T Daemi, A Ebrahimi, and M Fotuhi-Firuzabad. "Constructing the Bayesian network for components reliability importance ranking in composite power systems". In: *International Journal of Electrical Power & Energy Systems* 43.1 (2012), pp. 474–480.

[11] NEI Diverse. "flexible coping strategies (FLEX) implementation guide". In: *Nuclear Energy Institute* (2012).

[12] Ozge Doguc and Jose Emmanuel Ramirez-Marquez. "A generic method for estimating system reliability using Bayesian networks". In: *Reliability Engineering & System Safety* 94.2 (2009), pp. 542–550.

[13] Joanne Bechta Dugan, Salvatore J Bavuso, and Mark A Boyd. "Dynamic fault-tree models for fault-tolerant computer systems". In: *IEEE Transactions on reliability* 41.3 (1992), pp. 363–377.

[14] Clifton A Ericson. "Fault tree analysis". In: *System Safety Conference, Orlando, Florida*. Vol. 1. 1999, pp. 1–9.

[15] Clifton A Ericson et al. *Hazard analysis techniques for system safety*. John Wiley & Sons, 2015.

[16] Yan Fei and Lan Yucheng. "Probabilistic slope stability analysis: The case study of a deposit slope in hydropower reservoir". In: *2013 Fourth International Conference on Digital Manufacturing & Automation*. IEEE. 2013, pp. 948–951.

[17] Karl N Fleming. "Markov models for evaluating risk-informed in-service inspection strategies for nuclear power plant piping systems". In: *Reliability Engineering & System Safety* 83.1 (2004), pp. 27–45.

[18] JB Fussell. "A review of fault tree analysis with emphasis on limitations". In: *IFAC Proceedings Volumes* 8.1 (1975), pp. 552–557.

[19] Blaže Gjorgiev, Andrija Volkanovski, and Giovanni Sansavini. "Improving nuclear power plant safety through independent water storage systems". In: *Nuclear Engineering and Design* 323 (2017), pp. 8–15.

[20] EC Gomes, JP Duarte, and PF Frutuoso e Melo. "Human reliability modeling of radiotherapy procedures by Bayesian networks and expert opinion elicitation". In: *Nuclear Technology* 194.1 (2016), pp. 73–96.

[21] Bjørn Axel Gran and Atte Helminen. "A Bayesian belief network for reliability assessment". In: *International Conference on Computer Safety, Reliability, and Security*. Springer. 2001, pp. 35–45.

[22] FT Harper. *Analysis of core damage frequency from internal events: Surry, Unit 1*. Tech. rep. Sandia National Labs., 1986.

[23] Ronald A Howard. "Dynamic probabilistic systems, volume 1: Markov models". In: *John Wiley and Sons, Inc* 197.1 (1971), pp. 189–246.

[24] Yong Hu et al. "Software project risk analysis using Bayesian networks with causality constraints". In: *Decision Support Systems* 56 (2013), pp. 439–449.

[25] Finn V Jensen et al. *An introduction to Bayesian networks*. Vol. 210. UCL press London, 1996.

[26] Finn V Jensen. "Causal and Bayesian networks". In: *Bayesian networks and decision graphs*. Springer, 2001, pp. 3–34.

[27] Jian-jun Jiang et al. “Markov reliability model research of monitoring process in digital main control room of nuclear power plant”. In: *Safety science* 49.6 (2011), pp. 843–851.

[28] Sohag Kabir et al. “Reliability analysis of automated pond oxygen management system”. In: *2015 18th International Conference on Computer and Information Technology (ICCIT)*. IEEE. 2015, pp. 144–149.

[29] Bernhard Kaiser, Catharina Gramlich, and Marc Förster. “State/event fault trees—A safety analysis model for software-controlled systems”. In: *Reliability Engineering & System Safety* 92.11 (2007), pp. 1521–1537.

[30] John G Kemeny and J Laurie Snell. “Finite markov chains. d van nostad co”. In: *Inc., Princeton, NJ* (1960).

[31] Nima Khakzad, Faisal Khan, and Paul Amyotte. “Risk-based design of process systems using discrete-time Bayesian networks”. In: *Reliability Engineering & System Safety* 109 (2013), pp. 5–17.

[32] Pramod Kumar, Lalit Kumar Singh, and Chiranjeev Kumar. “An optimized technique for reliability analysis of safety-critical systems: A case study of nuclear power plant”. In: *Quality and Reliability Engineering International* 35.1 (2019), pp. 461–469.

[33] Helge Langseth and Luigi Portinale. “Bayesian networks in reliability”. In: *Reliability Engineering & System Safety* 92.1 (2007), pp. 92–108.

[34] Elmer Eugene Lewis. *Introduction to reliability engineering*. Tech. rep. 1987.

[35] Hak Kyu Lim. “A conceptual comparative study of FLEX strategies to cope with Extended Station Blackout (SBO)”. In: *14th International Topical Meeting on Probabilistic Safety Assessment and Management (PSAM14)*. 2018.

[36] BOY Lydell. *International databases on piping failures: Do they exist-are they needed?* Tech. rep. 1997.

[37] Medkour Malika et al. “Transformation of fault tree into Bayesian Network Methodology for Fault Diagnosis”. In: *Mechanics* 23.6 (2017), pp. 891–899.

[38] David Marquez, Martin Neil, and Norman Fenton. “Improved reliability modeling using Bayesian networks and dynamic discretization”. In: *Reliability Engineering & System Safety* 95.4 (2010), pp. 412–425.

[39] Kevin Murphy. “A Brief Introduction to Graphical Models and Bayesian Networks2”. In: *http://www. cs. ubc. ca/~ murphyk/Bayes/bnintro. html* (1998).

[40] Martin Neil et al. “Using Bayesian belief networks to predict the reliability of military vehicles”. In: *Computing & Control Engineering Journal* 12.1 (2001), pp. 11–20.

[41] Thomas Dyhre Nielsen and Finn Verner Jensen. *Bayesian networks and decision graphs*. Springer Science & Business Media, 2009.

[42] Norsys Software Corp. *Netica 5.0*. 1995. URL: https://www.norsys.com/netica.html.

[43] Ioannis A Papazoglou and Elias P Gyftopoulos. “Markovian reliability analysis under uncertainty with an application on the shutdown system of the Clinch River Breeder Reactor”. In: *Nuclear Science and Engineering* 73.1 (1980), pp. 1–18.

[44] J Pearl. “Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference Morgan Kaufmann, San Mateo, California 19882”. In: *Guttman EA Suchman PF Lazarfeld SA Star and JA Classen Wiley New York 1966* (1988).

[45] Nuclear Power Plants. *Industry-average performance for components and initiating events at US commercial nuclear power plants*. 2007.

[46] Luigi Portinale and Andrea Bobbio. “Bayesian networks for dependability analysis: an application to digital control reliability”. In: *arXiv preprint arXiv:1301.6734* (2013).

[47] M Mizanur Rahman and MB Shohag. “FLEX Strategy to Cope with Extended SBO for APR1400”. In: ().

[48] Ross D Shachter and Mark A Peot. "Simulation approaches to general probabilistic inference on belief networks". In: *Machine Intelligence and Pattern Recognition*. Vol. 10. Elsevier, 1990, pp. 221–231.

[49] JH Sigurdsson, LA Walls, and JL Quigley. "Bayesian belief nets for managing expert judgement and modelling reliability". In: *Quality and Reliability Engineering International* 17.3 (2001), pp. 181–190.

[50] Hyun Joon Son and Hak Kyu Lim. "Study of the FLEX Effectiveness of Strategies Under the Long-Term SBO by PRA". In: ().

[51] David J Spiegelhalter. "Bayesian graphical modelling: a case-study in monitoring health outcomes". In: *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 47.1 (1998), pp. 115–133.

[52] David J Spiegelhalter and Steffen L Lauritzen. "Sequential updating of conditional probabilities on directed graphical structures". In: *Networks* 20.5 (1990), pp. 579–605.

[53] Michael Stamatelatos et al. "Fault tree handbook with aerospace applications". In: (2002).

[54] Alan Stuart et al. *Kendall's advanced theory of statistics*. Wiley, 1994.

[55] Alan D Swain. "Handbook of human reliability analysis with emphasis on nuclear power plant applications". In: *NUREG/CR-1278, SAND 80-0200* (1983).

[56] IAEA TECDOC. "478. Component reliability data for use in probabilistic safety assessment". In: *Vienna: International Atomic Energy Agency* (1988).

[57] José Gerardo Torres-Toledano and Luis Enrique Sucar. "Bayesian networks for reliability analysis of complex systems". In: *Ibero-American Conference on Artificial Intelligence*. Springer. 1998, pp. 195–206.

[58] William E Vesely et al. *Fault tree handbook*. Tech. rep. Nuclear Regulatory Commission Washington DC, 1981.

[59] Andrija Volkanovski and Andrej Prošek. "Extension of station blackout coping capability and implications on nuclear safety". In: *Nuclear Engineering and Design* 255 (2013), pp. 16–27.

[60] Martin Walker and Yiannis Papadopoulos. "Qualitative temporal analysis: Towards a full implementation of the Fault Tree Handbook". In: *Control Engineering Practice* 17.10 (2009), pp. 1115–1125.

[61] Wolfram Research Inc. *Mathematica 8.0*. 2010. URL: http://www.wolfram.com.

[62] Bo-Bin Xiao and Te-Chuan Wang. "An Investigation of FLEX Implementation in Maanshan NPP by Using MAAP 5". In: *International Confernece Pacific Basin Nuclear Conference*. Springer. 2016, pp. 81–96.

[63] Vaibhav Yadav and John M Biersdorf. *Utilizing FLEX Equipment for O&M Cost Reduction in Nuclear Power Plants*. Tech. rep. Idaho National Lab.(INL), Idaho Falls, ID (United States), 2019.

[64] Zhu Yongli et al. "Bayesian network based time-sequence simulation for power system reliability assessment". In: *2008 Seventh Mexican International Conference on Artificial Intelligence*. IEEE. 2008, pp. 271–277.

[65] David C Yu, Thanh C Nguyen, and Peter Haddawy. "Bayesian network model for reliability assessment of power systems". In: *IEEE transactions on power systems* 14.2 (1999), pp. 426–432.

Appendices

Appendix A

Reliability Analysis through Bayesian Network Matlab Code

A.1 Cold Spare Gate Matlab Code

```
clc
clear
close all

lambda_A=0.001;
lambda_B=0.001;
key_work=1;
for m=10:10

delta_t=1;
tf=10;

A_fail_=0;
B_fail_=0;
```

```
n=tf/delta_t;

fun=@(tf)lambda_A*exp((-lambda_A)*tf);

A's Propability Table fails in a delta_t time interval
A_fail_total=0;

for j=1:n
   %A_fail_(j)=1-exp(-lambda_A*delta_t);
   A_fail_(j)=integral(fun,(j-1)*delta_t,j*delta_t);
   A_fail_total=A_fail_total+A_fail_(j);
end
A_fail_(n+1)=1-A_fail_total;

B's conditional probability table fails, given that A failed at an earlier time
for j=1:n
   B_fail_total=0;
    for i=1:n
        if i<=j
            B_fail_(j,i)=0;
        else
            B_fail_(j,i)=exp(-lambda_B*(i-j)*delta_t)*(exp(lambda_B*delta_t)-1);
        end
        B_fail_total=B_fail_total+B_fail_(j,i);
    end
   B_fail_(j,n+1)=1-B_fail_total;
```

```
   B_fail_(j+1,n+1)=1-B_fail_total;


end
The system will fail if at the same time that B fails, that is,
in this case, the total probability of B is calculated.
System_fail_total=0;
for i=1:n


    for j=1:n
        System_fail_total=System_fail_total+B_fail_(j,i)*A_fail_(j)*key_work;
        if i==j
            System_fail_total=System_fail_total+1*A_fail_(j)*(1-key_work);
        end


        Gate_work_(i)=1-System_fail_total;
    end
    Time_Int(i)=(i)*(delta_t);
    Reliability(i)=Gate_work_(i);
    if lambda_A == lambda_B
        Gate_markov(i) =1;
       Gate_markov(i) = exp(-lambda_A*i*delta_t)*(1+key_work*lambda_A*i*delta_t);
    else
        Gate_markov(i) = exp(-lambda_A*i*delta_t)+(key_work*lambda_A/(lambda_A-lambda_B)
    end
    MarkRel(i)=Gate_markov(i);
```

```
    relative_error=100*(Gate_work_(i)-Gate_markov(i))/Gate_markov(i);
end
end
close all
figure
plot(Time_Int,Reliability)
hold on
plot(Time_Int,MarkRel)
hold off
title('System Reliability (delta t = 0.1)')
xlabel('Mission Time')
ylabel('System Reliability')
legend('Bayesian Network','Markov Chain')
disp(Gate_work_(i))
disp(Gate_markov(i))
disp(relative_error)
```

A.2 The Case Study of FLEX Equipment Matlab Code

```
 clc
clear
close all
```

```
lambda_VAL1=2.37e-6;
lambda_RES=1.00e-9;
lambda_PIPE=4.45e-6;
lambda_VAL2=2.37e-6;
lambda_PUMPFS=1.00e-6;
lambda_PUMPFO=8.00e-6;
lambda_OPER=1.00e-7;
lambda_VAL3=2.37e-6;

delta_t=1000;
tf=10000;

VAL1_fail_=0;
RES_fail_=0;

n=tf/delta_t;

fun1=@(tf)lambda_VAL1*exp((-lambda_VAL1)*tf);
fun2=@(tf)lambda_RES*exp((-lambda_RES)*tf);
fun3=@(tf)lambda_PIPE*exp((-lambda_PIPE)*tf);
fun4=@(tf)lambda_VAL2*exp((-lambda_VAL2)*tf);
fun5=@(tf)lambda_PUMPFS*exp((-lambda_PUMPFS)*tf);
fun6=@(tf)lambda_PUMPFO*exp((-lambda_PUMPFO)*tf);
fun7=@(tf)lambda_OPER*exp((-lambda_OPER)*tf);
fun8=@(tf)lambda_VAL3*exp((-lambda_VAL3)*tf);
```

```
PUMPFS_fail_total=0;


for j=1:n


   PUMPFS_fail_(j)=integral(fun5,(j-1)*delta_t,j*delta_t);
   PUMPFS_fail_total=PUMPFS_fail_total+PUMPFS_fail_(j);
end
PUMPFS_fail_(n+1)=1-PUMPFS_fail_total;


for j=1:n+1
   PUMPFO_fail_total=0;
    for i=1:n+1
        if i>j
            PUMPFO_fail_(j,i)=0;
        elseif i==j
            syms k
         PUMPFO_fail_(1,1)=1;
         if i>1
         PUMPFO_fail_(j,i)=1- PUMPFO_fail_(j,i-1);
         end
         if i>2
           PUMPFO_fail_(j,i)=1-sum(PUMPFO_fail_(j,[1:i-1]));
         end
        elseif i<j
            PUMPFO_fail_(j,i)=integral(fun6,(i-1)*delta_t,i*delta_t);
        end
```

```
    end
end
 PUMPFO_VEC=0;

        PUMPFO_VEC=PUMPFS_fail_*PUMPFO_fail_;

for j=1:n+1
   OPER_fail_total=0;
    for i=1:n+1
        if i>j
            OPER_fail_(j,i)=0;
        elseif i==j
            syms k
         OPER_fail_(1,1)=1;
         if i>1
         OPER_fail_(j,i)=1- OPER_fail_(j,i-1);
         end
         if i>2
           OPER_fail_(j,i)=1-sum(OPER_fail_(j,[1:i-1]));
         end
        elseif i<j
            OPER_fail_(j,i)=integral(fun7,(i-1)*delta_t,i*delta_t);
        end
    end
end
 OPER_VEC=0;
```

```
        OPER_VEC=PUMPFO_VEC*OPER_fail_;

OR_4=eye(n+1);

        OR_VEC4=OPER_VEC*OR_4;

        for j=1:n+1
   VAL1_fail_total=0;
    for i=1:n+1
        if i>j
            VAL1_fail_(j,i)=0;
        elseif i==j
            syms k
         VAL1_fail_(1,1)=1;
         if i>1
          VAL1_fail_(j,i)=1- VAL1_fail_(j,i-1);
         end
         if i>2
             VAL1_fail_(j,i)=1-sum(VAL1_fail_(j,[1:i-1]));
         end
        elseif i<j
            VAL1_fail_(j,i)=integral(fun1,(i-1)*delta_t,i*delta_t);
        end
    end
end
```

```
 VAL1_VEC=0;


        VAL1_VEC=OR_VEC4*VAL1_fail_;


for j=1:n+1
   RES_fail_total=0;
    for i=1:n+1
        if i>j
            RES_fail_(j,i)=0;
        elseif i==j
            syms k
         RES_fail_(1,1)=1;
         if i>1
          RES_fail_(j,i)=1- RES_fail_(j,i-1);
         end
         if i>2
            RES_fail_(j,i)=1-sum(RES_fail_(j,[1:i-1]));
         end


        elseif i<j
            RES_fail_(j,i)=integral(fun2,(i-1)*delta_t,i*delta_t);
        end
    end
end
 RES_VEC=0;
```

```
      RES_VEC=VAL1_VEC*RES_fail_;

      OR_1=eye(n+1);

      OR_VEC1=RES_VEC*OR_1;

      Rel=0;

      for j=1:n+1
  PIPE_fail_total=0;
   for i=1:n+1
      if i>j
          PIPE_fail_(j,i)=0;
      elseif i==j
          syms k
       PIPE_fail_(1,1)=1;
       if i>1
       PIPE_fail_(j,i)=1- PIPE_fail_(j,i-1);
       end
       if i>2
         PIPE_fail_(j,i)=1-sum(PIPE_fail_(j,[1:i-1]));
       end

      elseif i<j
          PIPE_fail_(j,i)=integral(fun3,(i-1)*delta_t,i*delta_t);
      end
```

```
    end


end
 PIPE_VEC=0;


        PIPE_VEC=OR_VEC1*PIPE_fail_;


        OR_2=eye(n+1);


        OR_VEC2=PIPE_VEC*OR_2;


          for j=1:n+1
   VAL2_fail_total=0;
    for i=1:n+1
        if i>j
            VAL2_fail_(j,i)=0;
        elseif i==j
            syms k
         VAL2_fail_(1,1)=1;
         if i>1
        VAL2_fail_(j,i)=1- VAL2_fail_(j,i-1);
         end
         if i>2
           VAL2_fail_(j,i)=1-sum(VAL2_fail_(j,[1:i-1]));
         end
```

```
        elseif i<j
            VAL2_fail_(j,i)=integral(fun4,(i-1)*delta_t,i*delta_t);
        end

    end

end
 VAL2_VEC=0;

        VAL2_VEC=OR_VEC2*VAL2_fail_;

        OR_3=eye(n+1);

        OR_VEC3=VAL2_VEC*OR_3;

         OR_5=eye(n+1);

        OR_VEC5=OR_VEC3*OR_5;

          for j=1:n+1
   VAL3_fail_total=0;
    for i=1:n+1
        if i>j
            VAL3_fail_(j,i)=0;
        elseif i==j
            syms k
```

```
         VAL3_fail_(1,1)=1;
         if i>1
        VAL3_fail_(j,i)=1- VAL3_fail_(j,i-1);
         end
         if i>2
            VAL3_fail_(j,i)=1-sum(VAL3_fail_(j,[1:i-1]));
         end

        elseif i<j
            VAL3_fail_(j,i)=integral(fun8,(i-1)*delta_t,i*delta_t);
        end
    end

end
 VAL3_VEC=0;

        VAL3_VEC=OR_VEC5*VAL3_fail_;

        OR_6=eye(n+1);

        OR_VEC6=VAL3_VEC*OR_6;

        ORSUM=cumsum(OR_VEC6);

      for i=1:n+1
         Time_Int(i)=(i-1)*(delta_t);
```

```
        Rel(1)=1;
         if i>1
             Rel(i)=1-ORSUM(i-1);
         end
         end

figure
plot(Time_Int,Rel,'b')
hold on
title('System Reliability')
xlabel('Mission Time (h)')
ylabel('Reliability')
```