THESIS

A MODEL-BASED RISK AND RELIABILITY ASSESSMENT OF A SECOND-LIFE EV BATTERY ENERGY STORAGE SYSTEM

Submitted by

Hector Miguel Hernandez Ramirez

Department of Systems Engineering

In partial fulfillment of the requirements

For the Degree of Master of Science

Colorado State University

Fort Collins, Colorado

Spring 2026

Master's Committee:

Advisor: Vincent P. Paglioni

Jason Quinn
Indrajit Ray

ABSTRACT

A MODEL-BASED RISK AND RELIABILITY ASSESSMENT OF A SECOND-LIFE EV BATTERY ENERGY STORAGE SYSTEM

Lithium-ion batteries are widely used in energy storage applications because they are flexible, efficient, and scalable. However, as the electric vehicle market continues to grow, many batteries reach the end of their vehicle service life, raising concerns about significant waste generation. Some procedures allow these batteries to be repurposed for second-life applications. One viable approach is reusing these batteries in energy storage systems to provide a more sustainable solution. However, ensuring that these second-life systems operate safely is essential to prevent risks to people and the environment. To ensure operational integrity, diverse risk assessment frameworks can be employed to identify vulnerabilities and evaluate system reliability. Therefore, identifying the components that are most vulnerable to failure is essential for mitigating risks in these systems. This work presents a multi-stage analytical framework that unifies risk and reliability analyses within a model-based systems engineering (MBSE) architecture to identify and assess potential system risks. This project presents the procedure used to apply these methodologies and describes how the process evolved from manual qualitative assessments to an automated, model-based approach. Using this framework, the analysis suggests that the most significant risks reside in the auxiliary systems that maintain optimal operating conditions. Therefore, prioritizing the reliability and monitoring of these components is essential for the safe and sustainable deployment of second-life battery energy storage systems.

ACKNOWLEDGEMENTS

In this work, I would like to express my sincere gratitude to the Systems Engineering Department for the opportunity to pursue my master's degree within this program. In particular, I would like to thank Debra Dandaneau and Thomas Bradley for opening the door to this opportunity, for taking the time to explain the program to me, and for encouraging me to begin this new academic challenge by introducing me to my advisor.

I would also like to extend my deepest appreciation to my advisor, Vincent Paglioni, for welcoming me into the 3RC Lab and allowing me to be part of such a collaborative and inspiring research environment. Being part of the lab has given me the opportunity to work alongside outstanding colleagues and researchers who are conducting fascinating and meaningful work.

I would also like to thank the professors who taught the courses I took throughout this journey, which has not always been easy. As someone who has sometimes found it challenging to learn certain topics in a classroom environment, several of these professors demonstrated exceptional ability in communicating complex ideas clearly. Their dedication and teaching methods greatly contributed to my academic growth, and the knowledge gained from these courses contributed to the development of this work.

I am grateful to everyone who has supported me throughout this journey, whether through academic guidance, collaboration, or encouragement. Their support has played an important role in making this achievement possible.

I would like to acknowledge use of artificial intelligence (AI) for assisting with translating my writing and improving my grammar, as well as helping with some of the finer points of LaTeX coding.

Finally, this work was funded in part by the U.S. Department of Energy through a project led by Smartville, Inc. under program DE-FOA-0002680.

DEDICATION

To my beloved wife my best friend and the most important person in my life

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

Energy security and affordability have become central priorities across the world, yet nations are pursuing different strategies to achieve these goals [5]. Many fuel-importing countries increasingly rely on renewable energy sources (e.g., wind and solar) and efficiency improvements, while others continue to emphasize securing adequate supplies of traditional fuels. Energy now lies at the center of global economic and geopolitical discussions, as long-standing risks to fuel supply (e.g., foreign policies, supply chain logistics, etc.) are compounded by emerging constraints on critical minerals. At the same time, the electricity sector, which is essential to modern economies, faces growing vulnerabilities from cyber threats, operational challenges, and extreme weather events [5].

Despite rapid progress in clean energy deployment, fossil fuels continue to dominate global electricity generation. In May 2024, approximately 55% of net electricity generation capacity in the United States was derived from coal, petroleum, and natural gas [6]. Meanwhile, energy systems are undergoing rapid transformation. Renewables achieved record deployment for the 23rd consecutive year in 2024, while consumption of oil, natural gas, coal, and nuclear energy also reached historic highs [5]. These trends coincide with a sharp rise in energy demand driven by economic growth and the expansion of energy-intensive technologies, such as artificial intelligence [7]. Addressing these challenges requires energy systems that not only deliver sufficient and reliable power to meet escalating demand, but also align with the broader objective of transitioning to sustainable and environmentally friendly economies.

In this global context, novel energy systems continue to draw the attention of researchers and industry professionals. These novel technologies are a promising means to solving some of the challenges associated with extant energy sources, but are also associated with their own novel risks. This thesis aims to smooth the transition to new energy systems by developing a robust risk assessment methodology that is applicable even to early-stage designs.

1.1 Motivation

The variability of renewable energy sources such as wind and solar presents a significant challenge to achieving reliable and scalable energy systems [8]. Because these resources generate electricity only under favorable weather conditions, they are insufficient on their own to consistently support large-scale electrical grids and are limited in their effectiveness for off-grid applications. Moreover, addressing both short-term and seasonal variability requires the availability of flexible resources that can respond to fluctuations in supply on weekly, monthly, and seasonal timescales. One effective solution is to combine renewable generation with Battery Energy Storage Systems (BESS) [9]. By storing excess energy during periods of low demand and releasing it during peak demand (or in order to close gaps in generation), these systems allow for effective energy storage and control, helping to reduce intermittency and unpredictability. BESSs enhance energy resilience by providing backup power during grid disruptions, alleviating grid stress, and reducing the need for costly infrastructure upgrades [9].

Lithium-ion (Li-ion) batteries are among the most widely used technologies for energy storage due to their versatility and scalability [10]. However, the rapid growth of the Electric Vehicle (EV) market has raised concerns about the relatively short lifespan of Li-ion batteries in vehicles. The average Li-ion battery in EV applications has a lifespan of 10-15 years, and are no longer considered suitable for EV use when their state of health falls to 70% (equivalent to a 30% reduction in capacity) [11]. The large-scale retirement of EV batteries poses a significant challenge to environmental protection and resource recovery, since these batteries are usually replaced well before the end of their usable life, resulting in substantial battery waste [12]. To mitigate this issue, second-life applications have been developed, re-purposing retired EV batteries for use in complex energy storage systems [10].

This approach extends battery life and supports climate change mitigation efforts. However, second-life Li-ion batteries also present certain risks that must be carefully managed to ensure safety and reliability, particularly in critical energy storage applications. While the reconditioning process can restore some performance characteristics, it cannot fully reverse the structural and

chemical degradation that occurs during the battery's initial use [10]. This degradation can lead to increased impedance, lithium consumption, reduced conductivity, and electrical shorts in the anode-electrolyte interface, especially at low temperatures [2]. Additionally, corrosion can accelerate self-discharge, further compromising battery performance [13].

BESSs present an opportunity to use second life batteries to provide reliable and clean power. However, installations that incorporate physical and chemical safety mechanisms, along with control-based algorithms, have been associated with various operational risks, including premature shutdowns, fires, and system damage that can lead to cascading failures [14]. Such incidents are often caused by short circuits resulting from overloading, overheating, or mechanical abuse. Regardless of the specific application, unplanned shutdowns pose significant threats, ranging from loss of power to critical infrastructure and grid instability to damage to other generation equipment. The potential failure modes and degraded operation in second-life applications, underscore the need for rigorous safety protocols and reliable management systems to mitigate risks to human health and the environment [12].

Because BESS installations involve complex interactions among hardware components, control algorithms, and operating conditions, component-level reliability approaches (e.g., establishing the reliability of specific components) is not sufficient to ensure system safety [15]. A systems-level approach to risk and reliability assessment is necessary to properly understand and manage potential vulnerabilities. Reliability engineering and risk analysis offer structured methods to identify, evaluate, and reduce failure mechanisms that may lead to hazardous events [16]. These methods support the prevention, mitigation, response, and recovery from system failures, helping to maintain overall system integrity.

Such approaches are especially important for second-life battery applications, where degraded performance and cascading failures are more likely to result in events that can compromise public safety, worker health, and the environment.

1.2 Technical Gaps in Risk Assessment for BESS Applications

Despite the availability of established methodologies that help improve system robustness, many current approaches remain fragmented or component-focused and cannot fully capture the complex interactions and cascading effects inherent to complex systems [17] (not only BESS installations). A systems-level risk assessment can help prevent or mitigate losses arising from system vulnerabilities that may be insufficiently identified or evaluated [16]. Such assessments consider both technical and environmental threats, including risks from intentional actions as well as unintentional events such as natural disasters, operational errors, and component failures. These limitations highlight critical gaps in existing risk assessment practices when used in isolation, underscoring the need for more integrated, systems-level frameworks that provide a complete view of the entire system.

Addressing the complexity involved in identifying and preventing risks associated with complex systems requires comprehensive systems-level risk assessments. Combining various risk analysis and reliability tools offers significant advantages by enabling a more thorough and systematic approach to uncovering, assessing, and prioritizing potential failure modes.

The first methodology used here was Failure Mode and Effects Analysis (FMEA), which is normally applied to conduct early-stage risk assessments. This approach offers several benefits throughout product development, allowing the identification and resolution of potential risks before they escalate into costly problems [18], and facilitating the development of countermeasures to mitigate potential failures. The analysis is complemented by the use of an online tool, the Reliability Online Automated Databook System (ROADS), which provides data on the types and probabilities of component failures and their corresponding failure modes. This combined approach supports risk assessment focused on the most critical system components, identified based on the severity of their failure.

While FMEA is a valuable tool for providing qualitative insights into system design and operation, it also has inherent limitations due to its relatively simplified nature [16]. In particular, FMEA by itself offers limited probabilistic representation of system reliability, and the quality

of its analyses can vary widely depending on the level of effort invested. Additionally, because FMEA evaluates one failure mode at a time, it can become repetitive and time-consuming [16] for systems with numerous components.

This single-failure focus restricts its effectiveness in analyzing systems where multiple failures may occur simultaneously or where significant redundancy and diversity are present. To address these limitations and strengthen early-stage risk assessment beyond purely qualitative analysis, FMEA was complemented by Fault Tree Analysis (FTA) and Bayesian Network (BN) modeling to support more effective risk management strategies.

FTA and BNs share conceptual similarities, as both represent causal relationships and support reasoning about system behavior. The combined FTA–BN with FMEA approach enables structured scenario analysis and quantitative evaluation of failure likelihoods, resulting in more robust predictions of factors influencing system-level failures [19].

A failure mode can be viewed as an abstract representation of system failure, whereas faults represent the underlying causal events leading to that failure [20]. FTA is a top-down, deductive failure analysis method in which an undesired system state is analyzed using Boolean logic to combine contributing lower-level events [20]. In this context, when faults and failures are examined at a consistent level of granularity, faults may be interpreted as collections of causes associated with a specific hazard. By constructing cause-and-effect logical diagrams, FTA enables the systematic exploration of relationships among faults across different system layers, using failure effect information derived from FMEA. At the same time, integrating FMEA with BNs provides a probabilistic framework for mitigating system-specific risks and enhancing risk management effectiveness [21].

BNs are used to analyze the most critical failure causes among the high-priority items identified through FMEA [22], enabling the modeling of failure propagation throughout the system and offering deeper insight into potential cascading effects. By incorporating BN-based inference, the impact and interdependencies of individual failure causes can be evaluated, allowing the identification of those that exert the greatest influence on overall system behavior. As a result, the

likelihood and severity of system-wide damage due to interacting failure modes and causes can be significantly reduced.

Building on the complementary strengths of FTA and BNs, their integration enhances risk analysis by combining logical fault modeling with probabilistic reasoning under uncertainty. When further combined with FMEA, this integrated FMEA–FTA–BN framework supports a more comprehensive evaluation of mechanical and electronic failure scenarios by linking systematic failure identification from FMEA with causal modeling in FTA and probabilistic inference in Bayesian Networks. This combined approach enhances insight into critical failure mechanisms, captures failure interactions and uncertainties, and ultimately strengthens decision-making for overall system safety and reliability.

Reliability analysis are essential for guiding system design, operation, and maintenance decisions [23]. However, increasing system complexity has driven the need for more automated and integrated reliability analysis methods. There are several challenges associated with current approaches to creating, applying, and managing large volumes of information for critical systems, which are often labor-intensive and increasingly complex, to the point where automation becomes necessary [23]. Traditional approaches rely heavily on textual reports, large data tables, specialized analyses, and informal communication, which can be ambiguous, inconsistent, and difficult to maintain as system designs evolve [24].

Model-Based Systems Engineering (MBSE) addresses many of these challenges by improving the organization and consistency of design information. MBSE can help manage system complexity by maintaining and synchronizing all information about the system in a consistent and complete way [23]. Although some tools support reliability analysis, external tools are sometimes required to complement the work done in MBSE software. The development of critical systems requires closer integration between design, safety, and reliability activities, as these processes occur concurrently and directly influence design decisions [24].

This work thus combines FMEA, FTs and BNs within an MBSE environment to create a novel, feasible, and robust risk assessment approach that is applicable across the system life-cycle. Cru-

cially, this approach allows for easy scalability through the incorporation of an MBSE architecture, allowing models to easily grow and flex with the system design. The use of FMEA supports early-stage risk insights without the need for detailed system data, while FTs and BNs facilitate more a robust quantitative understanding of risk.

1.3 Impact

One of the initial impacts of this research was the identification of specific failure modes and root causes associated with both new and reconditioned lithium-ion batteries for second-life applications, through a detailed examination of their operating principles, degradation mechanisms, and performance limitations. By analyzing these degradation mechanisms and their underlying causes, this work provides a clearer understanding of how they contribute to power fade and capacity fade. Furthermore, this research clarifies the distinctions between these phenomena and identifies the principal factors influencing battery behavior and reliability when reused in energy storage systems.

Another key impact of this research was the identification of potential failure modes and causes within second-life BESS components, along with the assessment of the risks associated with these failure modes. By employing a catalog of component failures and integrating it into the well-known FMEA methodology, this approach enables an easier calculation of failure rate parameters and the ranking of issues based on their severity, likelihood, and detectability, ensuring that safety-critical components receive appropriate attention.

This work also simplifies the process of tracking failures and understanding failure propagation within a complex systems. By modeling dependencies and interactions among system components, the proposed approach helps determine whether an individual failure could trigger cascading failures elsewhere in the system. This is particularly important for BESS applications, where interactions between electrical, thermal, and control subsystems can amplify the consequences of different faults.

Another important contribution of this research is the creation of an automated process that connects MSOSA (Magic Systems of Systems Architect) reliability tool outputs with the GeNIe Python library to automatically build Bayesian Networks. When using tools such as FMEA and FTA in MSOSA, there was no straightforward way to perform detailed probabilistic analysis using Bayesian Networks. This research developed a method to extract the basic, intermediate, and top events, along with their associated probabilities, from Fault Tree Analysis conducted in MSOSA, and automatically generate Bayesian Network models in a software to visualize these Networks.

1.4 Thesis Organization

This work describes the methodologies, findings, and solutions involved in assessing the reliability of second-life EV battery systems. Chapter 2 introduces the BESS and establishes the theoretical framework for Li-ion batteries, including their functionality and the various degradation mechanisms that affect second-life applications. Building on this technical foundation, the chapter then reviews the risk analysis methods used to evaluate the system, highlighting how their specific features guide the overall assessment.

Next, Chapter 3 describes the steps carried out to complete this work. First, it explains how each of the methodologies used complements the others, detailing how the data were initially collected and processed throughout the study. This chapter outlines the sequence of steps followed to perform each methodology and the transition from one analysis tool to the next, explaining the role of each tool within the overall framework. It provides a clear overview of the analytical process, allowing the reader to understand both the rationale behind the chosen approach and how the different techniques were integrated to support the final results. Chapter 4 presents the results obtained in this work. The results are organized according to the different stages of the project, showing how each stage progressed and contributed to the overall outcomes.

Finally, Chapter 5 presents the higher-level insights derived from the results. This chapter analyzes the implications of the findings, discusses their limitations. Chapter 6 concludes this work by summarizing the main contributions of the study, highlighting its practical and theoretical im-

portance, and outlining recommendations for future research. It identifies potential improvements to the proposed framework and suggests directions for further development.

Chapter 2

Background

Many technologies used today to produce clean (low or zero-emission) electrical power can operate only during specific periods of time during the day. This is the case for solar and wind energy, which rely on environmental conditions to produce electricity [25]. These systems exhibit variability in power generation due to their intermittent nature and are typically used to complement load demand by supplying energy when environmental conditions are favorable. While they contribute to grid stability through auxiliary services such as frequency and voltage regulation [26], their variable nature can cause problems for applications requiring continuous power. However, this can be mitigated by allowing excess energy that is not consumed immediately, to be stored for later use.

This requires dedicated energy storage systems, such as Battery Energy Storage Systems (BESS). These systems can improve energy resilience by providing backup power during grid disruptions and can reduce the need for costly infrastructure upgrades [27]. Due to their complex design, they incorporate both physical and chemical mechanisms associated with different operational risks (e.g., system damage leading to cascading effects such as premature shutdowns or fires), often triggered by short circuits resulting from overloading, overheating, or mechanical stress [14].

BESS offer a variety of essential grid services, including peak demand reduction, load balancing, frequency stabilization, and emergency backup power during outages [27]. However, the rapid expansion of renewable energy generation has placed increased pressure on existing transmission networks, which often cannot expand quickly enough to accommodate new capacity. This mismatch can cause congestion in regions where energy demand is high but transmission capability is limited. In such areas, integrating BESS can be challenging because the charging process itself increases grid load. Therefore, proactive identification and mitigation of associated operational risks are required to support safe and reliable BESS planning and operation.

2.1 Battery Energy Storage Systems

A BESS is a technology-focused system that stores electricity and delivers it when needed [28]. It operates similarly to a large-scale rechargeable battery, capturing surplus energy during periods of low demand and discharging it as consumption increases. BESS installations range from small residential units to large utility-scale systems. In power systems with a high penetration of variable renewable energy sources, BESS support grid stability by "flattening" the variability. When combined with advanced control strategies, BESS provides essential grid services including frequency regulation, voltage support, and power smoothing, thereby enabling reliable operation of modern power networks [29]. Within smart grid architectures, BESS are increasingly integrated with real-time, two-way communication and interoperable control systems to improve coordination between loads and distributed generation, particularly solar photovoltaic (PV) resources. At the core of these systems are electrochemical batteries, with lithium-ion technology being the most widely adopted due to its high performance and scalability.

2.1.1 Lithium Ion Batteries

Lithium-ion batteries (LIBs)serve as the primary power source for modern electric vehicles and are also widely used in devices that require an energy storage component capable of delivering high power. This rechargeable technology offers a higher energy density than most other commercially available batteries. Each battery is composed of multiple individual units known as cells, which are connected through conductive surfaces that enable the flow of electric current [30].

Like other types of batteries, LIBs contain two electrodes: a positive electrode (cathode) and a negative electrode (anode). The cathode is typically made of highly purified lithium metal oxide, while the anode is generally composed of graphite, a form of carbon with a layered structure that facilitates efficient charge storage and energy transfer [30]. Each of these electrodes contains particles of active material (the material that allows ions to move between the anode and cathode) and a carbon-doped polymer binder, whose main role is to maintain physical contact between the active material particles and ensure good adhesion to the current collector [31].

Another important component is the Separator. In Li-ion batteries, this element transports ions between the cathode and anode while keeping the positive and negative terminals physically separated. The separator is typically a porous membrane (permeable to Li ions) that enables ionic conduction between the terminals while preventing electrical short circuits. Figure 2.1 illustrates the main components of a Li-ion battery and provides a simplified overview of how the electrochemical process enables the storage and release of electrical energy.

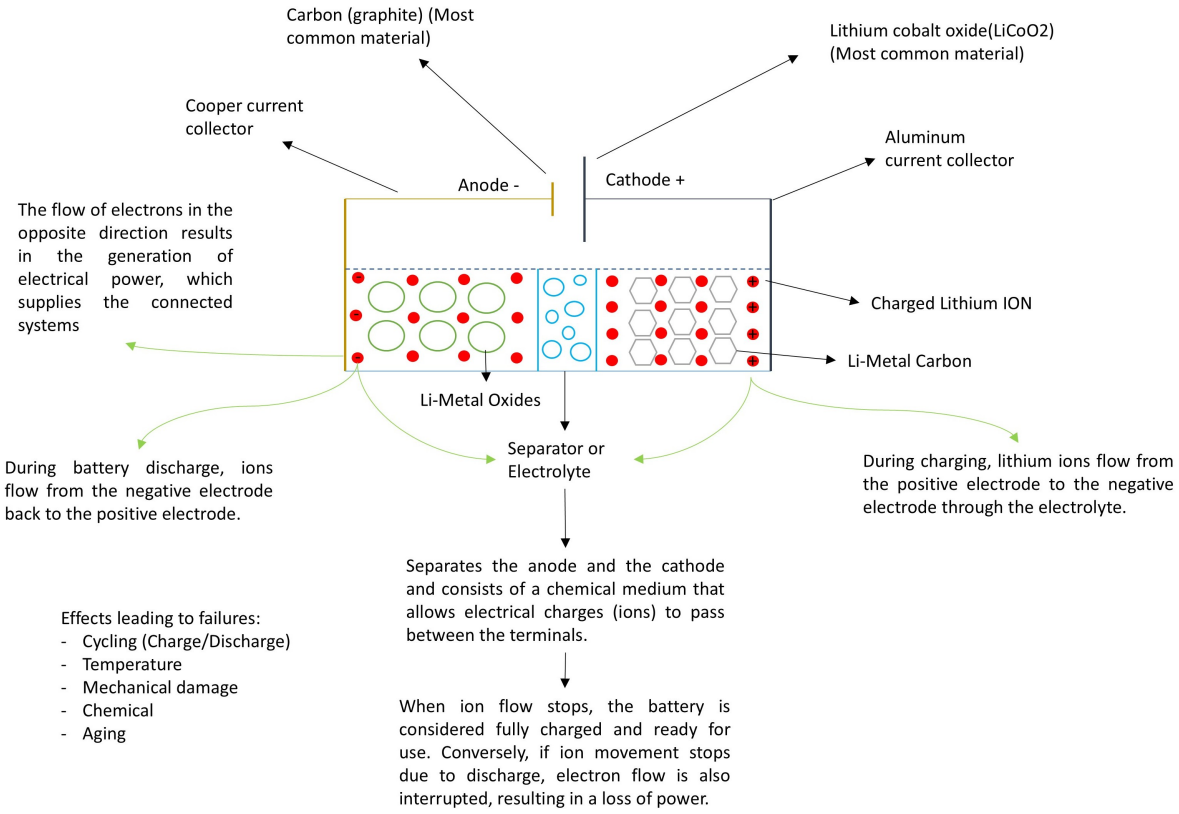


Figure 2.1: General view of Li-ion Batteries

2.1.2 Li-ion Batteries Degradation Mechanisms

Numerous studies have examined specific aspects of individual battery components, and several issues can lead to cell failure in LIBs. These failures may result from operating conditions, physical damage, or defects in cell design, materials, or manufacturing processes [32]. Over time, Li-ion batteries also deteriorate due to repeated charge and discharge cycles, which gradually reduces

their ability to retain energy, and thus their utility for many applications. Additionally, operating a cell at excessively high charge or discharge rates (high C-rates) can accelerate degradation and trigger other failure mechanisms. The physical consequences of these degradation mechanisms can be classified into three main modes [2]:

- Loss of Lithium Inventory (LLI) occurs when lithium ions are no longer available to move between the positive and negative electrodes during charging and discharging. The main reason is that some lithium becomes trapped or consumed by unwanted side reactions, such as the growth of surface films (like the solid electrolyte interphase, or SEI), lithium plating, or material decomposition.

- Loss of Active Material of the Negative Electrode (LAMNE) occurs when parts of the NE (anode) can no longer store lithium effectively. The main causes include cracking of electrode particles, loss of electrical contact within the material, or the formation of resistive surface layers that block lithium from entering active sites.

- Loss of active material of the Positive Electrode (LAMPE) is the degradation of the PE (cathode), where some of the active material becomes unable to participate in lithium insertion and extraction. This can occur due to structural damage within the material, cracking of particles, or a loss of electrical connection.

These degradation modes are closely linked to a variety of underlying degradation mechanisms, each triggered by different causes such as temperature, current load, and/or mechanical stress. As illustrated in Figure 2.2, the combination of these causes can initiate and progressively aggravate degradation mechanisms, contributing to the degradation modes that result in two principal outcomes (failures): capacity fade and power fade.

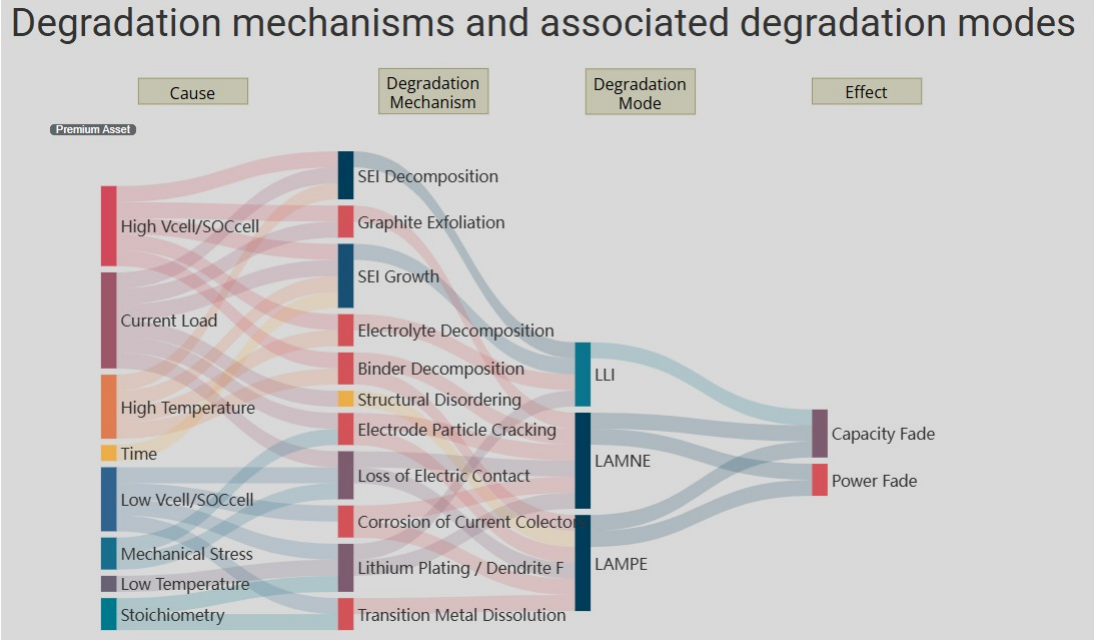


Figure 2.2: Degradation mechanisms and associated degradation modes with causes and effects adapted from [2]

2.1.3 Describing degradation Mechanisms

The two main failure effects experienced by Li-ion batteries are the inability to effectively store energy and the inability to deliver energy. Capacity fade refers to the reduction in the amount of charge the battery can store, which mainly occurs due to three factors: solid–electrolyte interphase (SEI) growth, mechanical damage, and changes in the positive electrode [33]. Power fade, on the other hand, represents the decline in the battery's ability to deliver energy efficiently. This phenomenon is primarily associated with an increase in internal impedance as the cell ages, often caused by repeated cycling. Factors such as resistive film formation, loss of electrical contact, and structural degradation within the electrodes contribute to higher resistance, which limits the flow of current and reduces the battery's instantaneous power output [34, 35]. To provide a clearer understanding of these relationships, the different degradation mechanisms shown in Figure 2.2 are described below.

Interface Degradation Mechanisms

The first degradation mechanism is the formation of the solid–electrolyte interphase (SEI), also known as the passivation layer. This layer is produced when the electrolyte reacts with the negative electrode during charging, forming a thin protective film on its surface [35]. The SEI acts as a protective barrier that allows lithium-ion transport while preventing further direct contact between the electrolyte and the electrode. It is one of the most important components in LIBs because it strongly influences battery performance, including capacity degradation, safety, calendar life, and cycle life [36]. However, the SEI layer is not completely stable and can degrade under certain operating conditions, such as elevated temperatures and continuous cycling, leading to SEI decomposition [37]. This degradation mechanism occurs when the SEI reacts with the electrolyte, causing continual growth and reformation of the layer. This repeated decomposition and regeneration consumes lithium ions and electrolyte, increases internal resistance, and ultimately degrades battery performance.

Although capacity loss in LIBs is often directly linked to side reactions at the negative electrode, it can also result indirectly from processes occurring at the positive electrode, particularly electrolyte decomposition [38]. The electrolyte in LIBs is not completely stable when the battery operates at high voltages. Under these conditions, it can break down through chemical reactions with both electrodes. At the negative electrode, the electrolyte reacts with lithium ions and electrons, producing compounds that form or modify the SEI. At the positive electrode, the electrolyte can also decompose, forming resistive surface films and other unwanted products, especially at elevated temperatures [38].

Another degradation mode is Structural Disordering. Battery materials are made of crystal structures, which are highly organized arrangements of atoms where lithium-ions can stay, move around, and react. The performance of a battery depends heavily on how well organized these crystal structures are [39]. When the crystal is well ordered, lithium can move easily, enabling repeated charge and discharge cycles. Many cathode materials used today work well because they have highly ordered structures. In these materials, lithium ions and transition-metal (TM) ions

each have their own well-defined positions within the crystal structure, and this separation enables fast and stable lithium diffusion [39].

However, when lithium is removed during charging, the material passes through metastable states, which are states that are not thermodynamically stable. In these states, the structure becomes stressed. Because of this instability, the material often undergoes structural disordering [39]. This means the original atomic arrangement breaks down due to TM ions migrating into the lithium layers, the structure changing into a different form, or oxygen being released at high voltage. These changes disrupt the pathways that lithium ions use to move. Once this disorder sets in, lithium can no longer move freely, causing the material to lose capacity and reducing battery life. Preventing this disordering has been a major challenge in battery research, and tremendous efforts have been dedicated to understanding and inhibiting it [39].

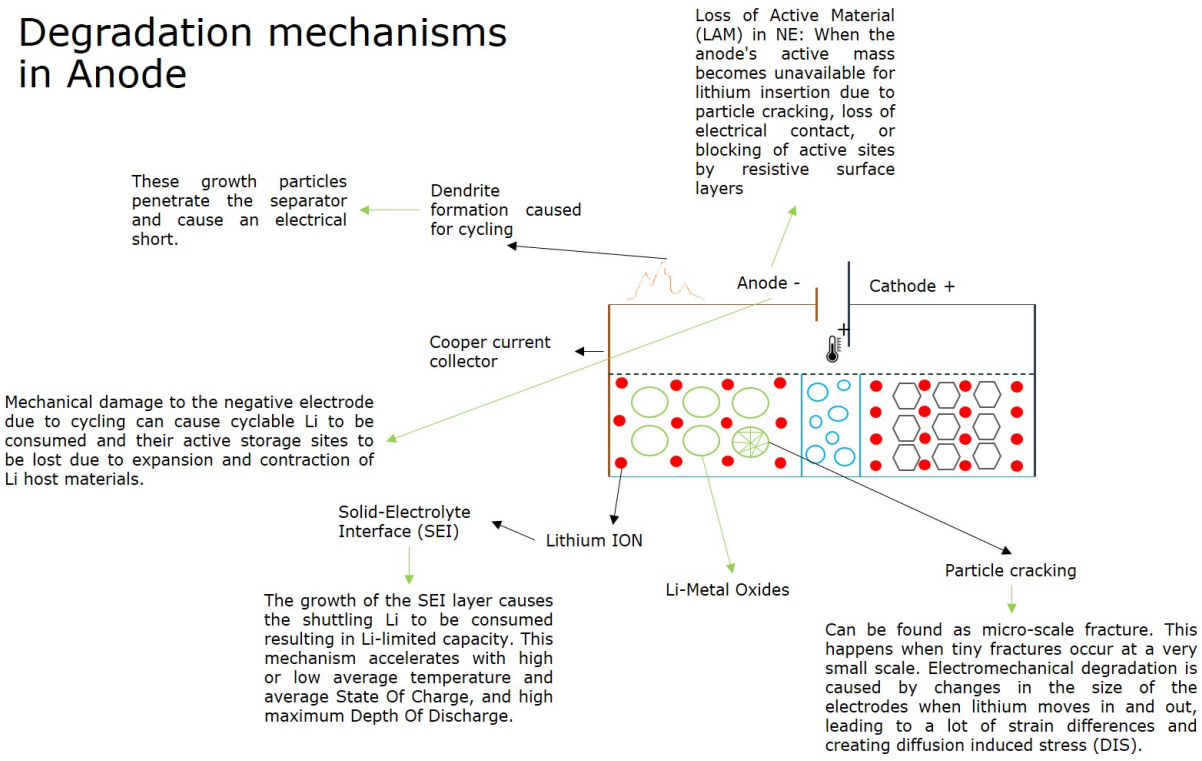


Figure 2.3: Example of Degradation mechanisms in Anode

Chemical Degradation Mechanisms

Lithium plating, also called dendrite formation, is another degradation process (similar to SEI growth) that occurs at the negative electrode of a lithium-ion battery. Lithium plating happens when the movement of lithium ions from the electrolyte into the graphite, where they normally intercalate between the graphite layers, slows down or becomes blocked. Instead of entering the graphite, lithium ions are forced to deposit on the surface as metallic lithium resulting in the consumption of active lithium, reduction of battery capacity, and formation of structures called dendrites, which may cause short circuits [40].

Lithium plating becomes more likely under several conditions [41]:

- Low temperatures: Lithium ions cannot move into the graphite quickly. As a result, they pile up on the surface as metal.
- Fast charging: Charging the battery too quickly raises the anode potential and creates conditions that favor plating.
- Fully lithiated graphite surface: When the graphite is already full of lithium, new ions cannot intercalate and instead plate on the surface.
- Design issues: Low anode-to-cathode ratio, manufacturing defects, or clogged pores can also increase the risk of creating this degradation mode.

Research indicates that lithium plating can happen in two distinct phases when a battery is charged at low temperatures. The first phase begins soon after charging starts at –20°C, while the second phase takes place later in the charging process. Lithium that accumulates on the surface of the negative electrode during low-temperature charging can move back into the graphite once the battery is warmed to room temperature. However, this diffusion does not occur at –20°C. This suggests that the slow movement of lithium within the graphite at low temperatures is the main factor that limits how quickly lithium-ion cells can be safely charged in cold conditions [42].

Corrosion of the current collector is another chemical degradation mechanism that can affect lithium-ion battery performance and safety. This process occurs in both aluminum and copper

current collectors [43], which are commonly used in battery electrodes. Corrosion is a spontaneous reaction between the metal and its environment, and it can be accelerated by the harsh chemical conditions inside a battery [44]. Aluminum, typically used as the current collector for the positive electrode, is particularly vulnerable to corrosion due to the high operating voltages of the cathode [44]. Copper, which is commonly used on the negative side, is more prone to corrosion when the battery is exposed to moisture contamination or when lithium comes into direct contact with the copper surface [43]. In both cases, reactions between the metal and the electrolyte can produce unwanted byproducts and weaken the metal structure.

This degradation mechanism can also contribute to self-discharge. When aluminum is coated with active cathode material, metal dissolution can occur alongside reduction reactions at the electrode surface, allowing internal currents to flow even when the battery is not in use. Over time, this reduces available energy and accelerates aging. In addition, corrosion can weaken the mechanical connection between the current collector, electrode coating, and external tabs, increasing electrical resistance and, in severe cases, leading to venting or safety failures [44]. Like other degradation mechanisms, current collector corrosion is influenced by the battery's operating conditions and materials. High voltage, electrolyte impurities, and long-term exposure to reactive environments increase corrosion rates. Although surface treatments and protective coatings are commonly used to improve corrosion resistance, degradation of the current collector remains an important contributor to capacity loss, increased resistance, and reduced battery safety over the cell's lifetime [43, 44].

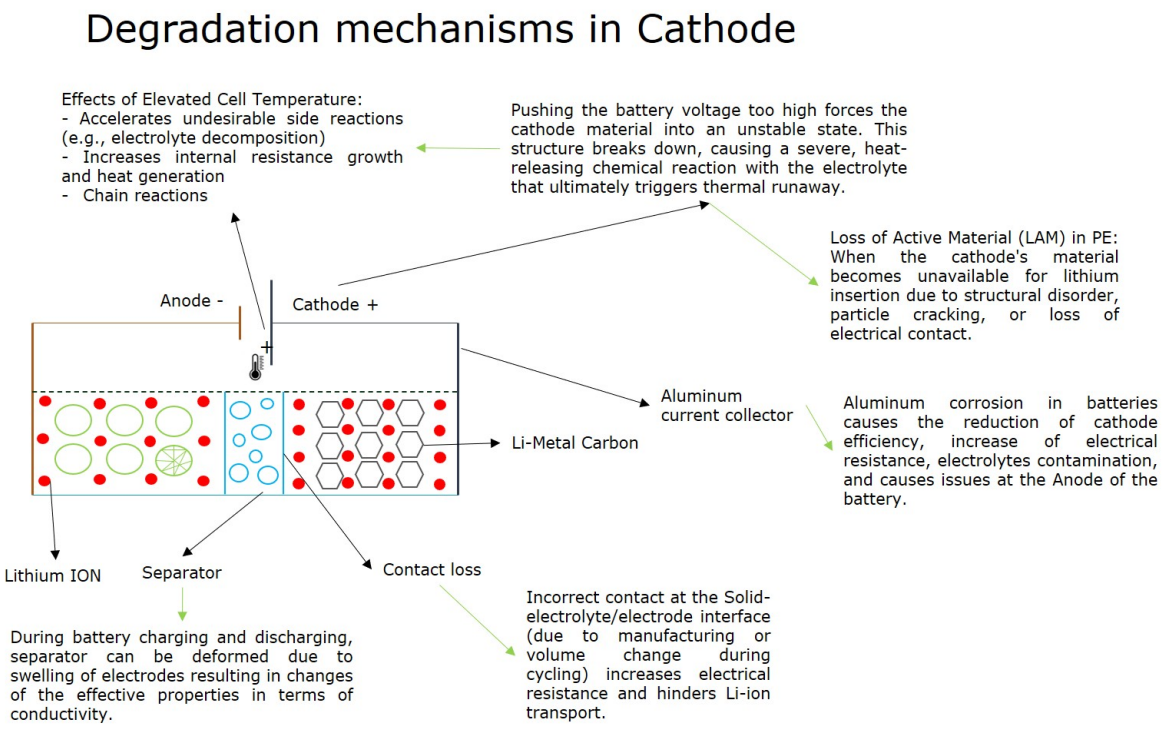


Figure 2.4: Example of Degradation mechanisms in Cathode

Electrode Degradation Mechanisms

While lithium plating and current collector corrosion represent chemical failure mechanisms, the loss of electrical contact results from *mechanical* degradation of the electrode's internal structure. Loss of contact can occur at the interface between the active material and the current collector or within the active material itself, either between active particles or between particles and the conductive additive [45].

Mechanical degradation of the internal electrode structure is mainly caused by diffusion-induced stress (DIS). As lithium moves in and out of the electrode during charging and discharging, the material repeatedly expands and contracts. The current collector limits this movement, which creates mechanical stress at the interface and can lead to cracking or separation [45]. In addition, electrochemical reactions can slow lithium diffusion, reducing stress and making it easier for the active layer to detach from the current collector. These effects become worse under high current con-

ditions, such as fast charging or high-power discharge, because rapid volume changes increase fatigue at material interfaces [46] [47].

Operating at low state of charge or low cell voltage, which induce strong contractions in the electrode structure, also contributes to damage as the electrical connections fatigue and gradually break [47].As a result, some regions of the active material lose electrical contact and become inactive. The stress is often unevenly distributed, causing mechanical strain that negatively impacts cycle aging, especially in cells with curved geometries, where bending stresses vary with the state of charge [47].

Repeated mechanical stress during electrochemical cycling in LIBs can also generate particle cracking. The movement of lithium ions into and out of the electrode causes repeated expansion and contraction of the material [40]. For example, graphite particles expand during lithiation and contract during delithiation [48]. These cycles generate concentration gradients within the particles, leading to diffusion-induced stress and initiating micro-fracturing of the particles [40]. Particle cracking becomes more severe under high current loads and in electrodes composed of larger particles, where lithium concentration gradients are more pronounced.

As cycling continues and the battery approaches end of life, micro-cracks can propagate and lead to complete particle fracture [40]. This damage is often observed near the separator, where local reaction current densities are higher during operation. Improper current loading and aggressive cycling conditions further accelerate this electromechanical degradation. The formation of cracks exposes fresh particle surfaces to the electrolyte, which promotes rapid growth of SEI. SEI formation on these newly exposed surfaces occurs faster than on pre-existing surfaces, increasing irreversible lithium consumption. In addition, particle cracking disrupts electrical pathways within the electrode, increasing resistance and contributing to capacity fade. The combined effects of mechanical fracture, accelerated SEI growth, and enhanced side reactions such as lithium plating significantly reduce battery performance and lifetime [40].

Degradation mechanisms specific to the positive electrode are not completely understood, but they are known to include irreversible structural phase changes [40]. Metals like Manganese (Mn)

and Nickel (Ni), commonly found in the cathode, can gradually dissolve into the electrolyte. Once dissolved, these metal ions can travel through the electrolyte and deposit on the negative electrode in a manner similar to lithium plating [49]. When Mn and Ni accumulate on the graphite surface, they interfere with normal electrode reactions and promote excessive growth of SEI accelerating capacity fade.

This so-called transition metal dissolution becomes more severe at high state of charge, elevated temperatures, and long storage times [49]. As the process continues, the cathode gradually loses active material, while the electrolyte and electrode surfaces accumulate decomposition products. These changes also contribute to self-discharge and further electrolyte breakdown, creating a feedback loop that accelerates battery aging. Overall, transition metal dissolution interacts closely with other degradation mechanisms [40] to contribute to battery failure through multiple pathways. Like particle cracking leads to the loss of active material and structural damage within the electrode. Similar to SEI growth and lithium plating, it consumes active lithium and degrades electrode interfaces [49].

Structural Degradation Mechanisms

The binder in LIBs can also decompose due to chemical or electrochemical breakdown of the polymer material under harsh operating conditions, such as high temperature, overpotential, or prolonged cycling [50]. Reactions induced by elevated temperature or high cell voltage, as well as the mechanical stresses generated during repeated charge–discharge cycles, can further degrade the binder. This degradation affects directly the integrity of the electrode and its cycle life, resulting in increased impedance and capacity fading [51].

In addition to binder decomposition, the structural stability of the graphite anode can also be compromised through exfoliation processes. Graphite exfoliation can be described as an electrochemical process occurring in exfoliation-sensitive graphite particles, caused by differences in voltage and current across the electrode densities during lithium insertion [52]. It can be suppressed when the local current density attributed to exfoliation exceeds a threshold, either by reducing the

amount of exfoliation-prone graphite or by increasing the total current density, resulting in more stable, passivated graphite behavior.

Second-life EV Battery Degradation

Lithium-ion batteries begin to degrade as soon as they are first used. Although reconditioning can restore some performance through testing, balancing the cells, and replacing damaged parts, cannot completely reverse the internal aging of the battery. Second-life batteries are still affected by their previous usage, and reconditioning does not fully restore them to their original condition. Some types of damage, such as microscopic cracks inside the battery or corrosion of parts, remain even after reconditioning. These hidden issues can worsen over time, hastening capacity loss, especially when the operating context changes, such as deploying EV batteries in stationary energy storage systems. Additionally, resistance within the battery is often uneven, causing some areas to heat up more than others.

Although second-life batteries experience similar aging processes as new batteries, their prior use and reconditioning influence how these processes affect them. The principal degradation mechanisms for second-life LIBs identified in this research are described below.

- SEI Layer Growth: The protective layer inside the battery keeps getting thicker over time, making it harder for the battery to work efficiently and reducing the amount of lithium that can move around inside [2].

- Lithium Plating and Dendrite Formation: Some lithium can form unwanted deposits during fast charging in the battery's first life. These remaining deposits can promote further accumulation during the battery's second life, raising the risk of short circuits, especially under cold temperatures or high-power demands [41].

- Particle cracking: Stress from temperature changes in the first life can cause tiny cracks in the battery's internal materials. These cracks can grow over time, even with gentler use, leading to loss of active material and allowing harmful substances inside [10].

- Corrosion of Current Collectors: Degradation of the positive current collector increases contact resistance and can lead to electrolyte contamination. This accelerates self-discharge rates, a critical concern for stationary storage where energy retention over time is paramount [13].

Understanding these degradation processes is essential for accurately assessing battery health and performance over time. This knowledge is also critical for determining the optimal point at which a battery should be retired from its first-life application in electric vehicles, thereby ensuring its safe and efficient transition into a second-life use [10], where it can continue to deliver functional and economic value.

2.1.4 Second Life Lithium Ion Batteries

The high market cost of new LIBs, combined with the rapid growth of the EV market, limits their widespread adoption for new applications. Batteries from EVs typically have a lifespan ranging from 5 to 15 years, during which they can experience a capacity reduction of up to 20% [12]. Although around 80% of their capacity remains, safety, performance, and regulatory concerns prevent their continued use in vehicles. This limitation has prompted the exploration of new opportunities, such as second-life applications for EV batteries. A second-life battery refers to the re-manufacturing, re-purposing, and reuse of batteries once they reach the end of their primary service life [53]. By extending their operational lifespan, these second-life applications add economic value to batteries while promoting the principles of the circular economy and sustainability [12].

To enable these second-life applications, retired EV batteries must first go through a reconditioning process that prepares them for reuse. This process involves a series of specialized treatments, as described in [10] and [12]. It begins with the dismantling of battery packs, during which some components are recycled or repurposed for other uses. The EV battery packs are then visually and electrically inspected, and sorted according to their specifications (e.g., battery chemistry, voltage range, energy capacity, state of health, remaining useful life, etc.) [12]. Next, the packs are disassembled into modules, which are subsequently reassembled to form new battery products.

In addition, the usage history is analyzed, and further tests are performed or inferred using model-based methods to estimate the remaining lifetime of the batteries. Batteries with similar first-life cycling profiles and health statuses are clustered to ensure consistency in performance. Based on parameters such as power capability, energy capacity, service life, and depth of discharge (DOD), suitable stationary applications for these reconditioned batteries are then identified [12]. Figure 2.5 summarizes the steps described.

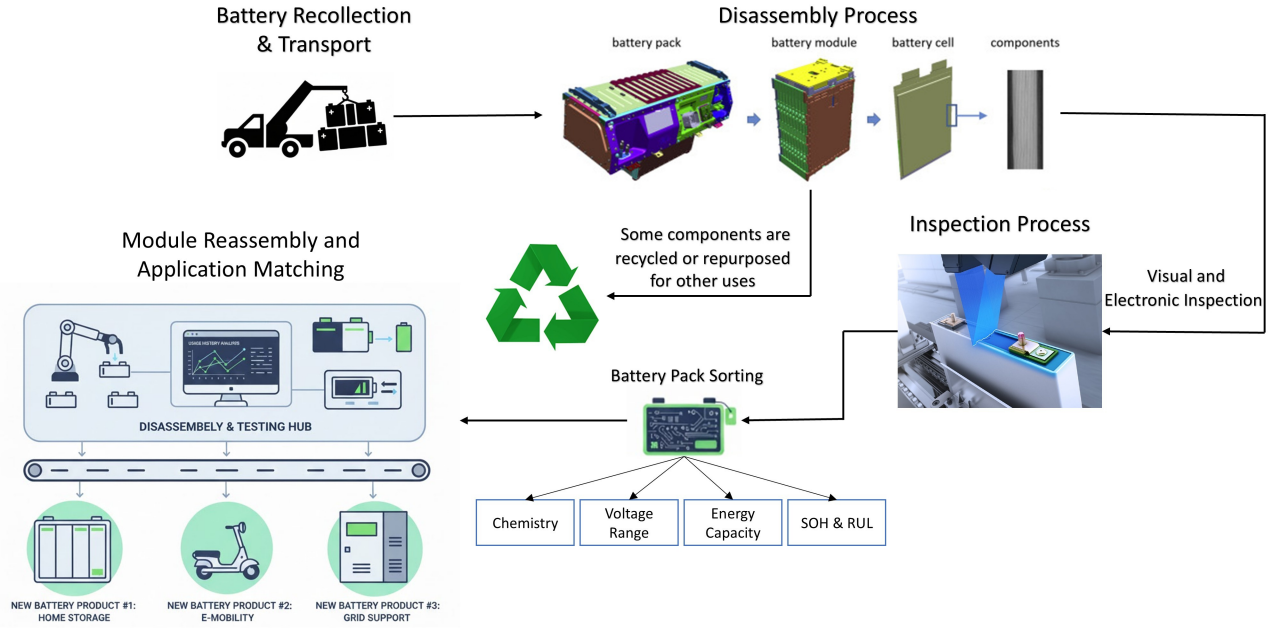


Figure 2.5: Second-life Li-ion Reconditioning Process

Despite the challenges associated with reconditioned batteries, battery energy storage systems (BESS) represent a significant opportunity for their utilization. These systems can play an important role in power grids, emergency power supplies, telecommunications [12, 54], and other stationary applications. When integrated into low-voltage (LV) networks, BESS can participate in peak shaving and help prevent excess energy from renewable sources from flowing into the transmission system. Combining BESS with second-life batteries can reduce the global environmental footprint while providing cost-effective energy storage alternatives [55]. Ongoing research and development efforts aim to enhance the performance and reliability of BESS that incorporate

second-life LIBs. Current studies focus on understanding and mitigating degradation mechanisms, optimizing clustering algorithms for efficient battery pairing, and developing advanced thermal management systems to ensure safe and stable operation.

2.1.5 Additional BESS Components and Subsystems

Energy storage allows power-generation sources to operate at optimal efficiency by absorbing demand fluctuations. Different grid applications require storage systems of vastly different sizes, ranging from seasonal storage that holds energy for months to fast-response systems that stabilize the grid for only minutes [56]. Because these systems must perform reliably across diverse time and power scales, effective management is essential to ensure safe and optimal operation. In addition to this, other components are required to support system functionality and reliability, including control units, interface hardware, thermal regulation mechanisms, sensing devices, and power-conversion equipment [26]. Together, these subsystems enable a Battery Energy Storage System (BESS) to monitor its internal state, maintain safe operating conditions, manage energy flow efficiently, and interact seamlessly with the electrical grid.

- Battery Management System (BMS) Controller: This is a component used to control how the storage system operates [56]. By incorporating advanced physics-based models, it enables significantly more reliable and efficient operation of the storage system. Since BESS rely on this component to continuously monitor and ensure the safe, optimal performance of each battery pack, and because battery packs degrade over time through cycling, with degradation intensified by factors such as aggressive charging protocols, elevated temperatures (both ambient and operating), overcharging or undercharging, advanced BMS can be employed to mitigate degradation mechanisms, enhance overall system performance, and do more than merely manage battery packs to meet immediate power demands [56].

- Battery Interface Box (BIB): Hardware protection and control device used to interface a high-voltage lithium-ion battery bank with the rest of the electrical system, such as a DC bus, load, or charger [57]. Its primary function is to safely connect or disconnect a battery

or battery bank in order to protect the batteries from improper use and unsafe operating conditions. The BIB communicates with each battery's BMS to verify battery availability and ensure operation within specified limits. By continuously monitoring all connected batteries, the BIB can detect alarms or communication losses reported by the BMS. If one or more batteries indicate a fault or become unavailable on the bus, the BIB automatically disconnects the affected battery bank from the DC bus to prevent damage or hazardous conditions. The BIB is intended for use in applications such as off-grid power systems, marine and industrial power supplies, and renewable energy storage systems, but it is not suitable for medical or aviation applications [57].

- Cooling systems are a critical component of BESS designed to dissipate heat generated during battery operation [58]. Many BESS utilize liquid-cooling systems to maintain batteries within an optimal thermal window, a critical factor for maximizing performance and service life [59]. By providing a closed-loop environment, these systems ensure battery compartments remain cool, clean, and dry, effectively isolating sensitive cells from external stressors like solar radiation and extreme ambient temperatures. This specialized thermal management is normally used in outdoor installations, where uncontrolled thermal stress can lead to accelerated degradation, reduced capacity, and system malfunction [59].

- Sensors: These components are used continuously to monitor key operational and environmental conditions. Sensors measure parameters such as temperature, humidity, and the presence of off-gassing, among others, providing critical data needed to optimize system performance and prevent failures [60]. Lithium-ion batteries are particularly sensitive to temperature, making precise thermal monitoring essential for maintaining optimal operating conditions. As previously mentioned, elevated temperatures can accelerate battery degradation and increase the risk of thermal runaway, a hazardous chain reaction in which damaged or defective cells rapidly overheat, releasing flammable gases and potentially leading to fires or explosions [61]. Advanced sensor technologies, including off-gas detectors, are capable of identifying electrolyte vapors emitted by batteries in the early stages preceding thermal

runaway. Early detection allows corrective actions, such as reducing charge rates or stopping operation, to be taken before a critical failure occurs. By enabling continuous environmental monitoring, early fault detection, and data-driven system optimization, sensors are essential for improving design efficiency and ensuring the safe, reliable, and long-term operation of BESS [60].

- AC-DC Converters / Inverters: AC–DC converters enable the integration of renewable energy sources, batteries, and the electrical grid. Many renewable energy technologies generate electricity in direct current (DC) form, which is also the form required for battery storage [62], while most electrical grids and end-use equipment operate using alternating current (AC). AC–DC converters and inverters allow electricity to be efficiently converted between these two forms so that energy produced by Renewables can be stored in batteries and later delivered to loads or fed back into the grid. BESS can be configured using either AC-coupled or DC-coupled architectures, depending on how renewable generation, storage, and grid connections are arranged [62–64]. Advances in power electronics and inverter technology have improved the performance and availability of both configurations. Because each energy conversion step introduces additional complexity and potential losses, AC–DC converters in BESS are designed to maximize efficiency and reliability in order to reduce overall system costs and ensure dependable operation [63, 64].

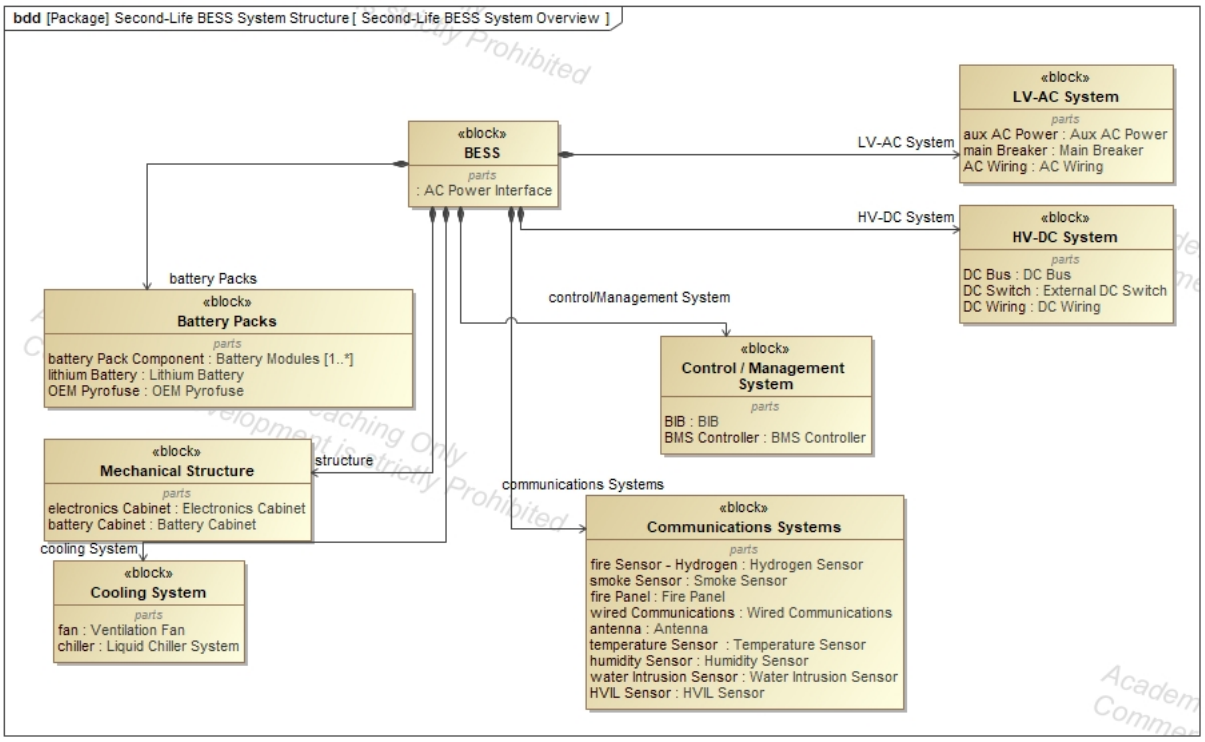


Figure 2.6: Block Definition Diagram of Second-Life BESS General Structure

2.2 Risk Analysis Methods

Although the risks associated with BESS are well known and mitigation strategies are widely implemented in grid-scale systems, established and standardized risk management schemes and models remain less mature compared to those in industries such as chemical processing, aviation, nuclear power, and petroleum [29]. Since 2018, multiple incidents involving battery storage facility fires and explosions have been reported annually, resulting in injuries and significant financial losses due to asset damage and operational downtime [29]. While BESS deployments include a range of physical, chemical, and control-based protection mechanisms, they continue to face notable operational risks. These risks may present as unplanned outages, thermal events, fires, or equipment failures that can propagate into wider system disturbances. In many cases, such incidents are initiated by short circuits arising from overload conditions, elevated temperatures, or mechanical damage [14].

To understand and manage risks related to BESS as well as other types of systems, a variety of reliability assessment techniques are available. Each method offers distinct strengths for evaluating system performance and helping to uphold the integrity of engineered systems. These methods help explain how system component configurations influence overall behavior and support logic based modeling of complex interactions. Collectively, they inform strategies to prevent failures, mitigate their impacts, support effective response, and enhance system recovery [16].

Compared to standard series parallel configurations, reliability analysis of complex systems is more challenging. This increased complexity arises from uncertainties in how component failures propagate through the system and result in cascading effects, since the performance of one component often depends on the operating state of others. These dependencies, together with multiple possible failure modes, require the use of more than one risk analysis method, as no single approach can fully capture system behavior. By combining different reliability analysis methodologies, it becomes possible to estimate risk likelihood more accurately, identify critical failure points, and reduce modeling assumptions that may overlook important interactions between components. The following subsections will discuss each of the different reliability analyses used in this work, how they were applied, and how each of these approaches also improves coverage across different system levels, supports both qualitative and quantitative evaluations, and explain how these approaches improve coverage across different system levels and inform more effective system design and operational strategies.

2.2.1 Failure Modes and Effects Analysis (FMEA)

Failure modes and effects analysis (FMEA), which in this work served as the first step in system reliability assessment, is commonly used for early-stage risk assessments. FMEA is a well-defined and structured method that offers several advantages throughout product development by enabling the identification of potential failure modes and their effects before they evolve into costly issues [18]. The semi-quantitative approach of FMEA (or FMECA, failure modes, effects

and criticality analysis) allow analysts to identify system risks with minimal data, and the method itself is easily adaptable to changing system designs.

This structured approach facilitates the development of countermeasures to mitigate or prevent failures and is typically carried out through a formal process involving a multidisciplinary team. In practice, FMEA can be applied at various stages of system development, from conceptual design to implementation and operation, and may be conducted at different levels of abstraction, including system functions, subsystems, and individual components.

The analysis assumes the occurrence of a failure and subsequently evaluates its potential effects and underlying causes, with the objective of reducing the likelihood of occurrence or recurrence. While FMEA is an essential tool in many system design and reliability engineering processes, it remains a simplified technique and provides limited insight into the probabilistic characterization of overall system reliability [16]. Figure 2.7 describes the flow process of the FMEA that was followed in this work.

FMEA has been widely adopted in a range of domains, including renewable energy systems, aerospace engineering, automotive manufacturing, and medical devices. The core elements of FMEA include:

- **Failure Mode** – A potential way in which a component, subsystem, or system may fail to perform or deliver its intended function.
- **Effect of Failure** – The impact of the failure mode on the intended function
- **Cause of Failure** – The factors within the design process that might allow the failure to occur, typically expressed in terms of variables that can be corrected or controlled.
- **Severity** – The degree to which the failure impacts system functionality.
- **Occurrence** – The likelihood of the failure occurring.
- **Detection** – The ability to detect the failure before it affects the system

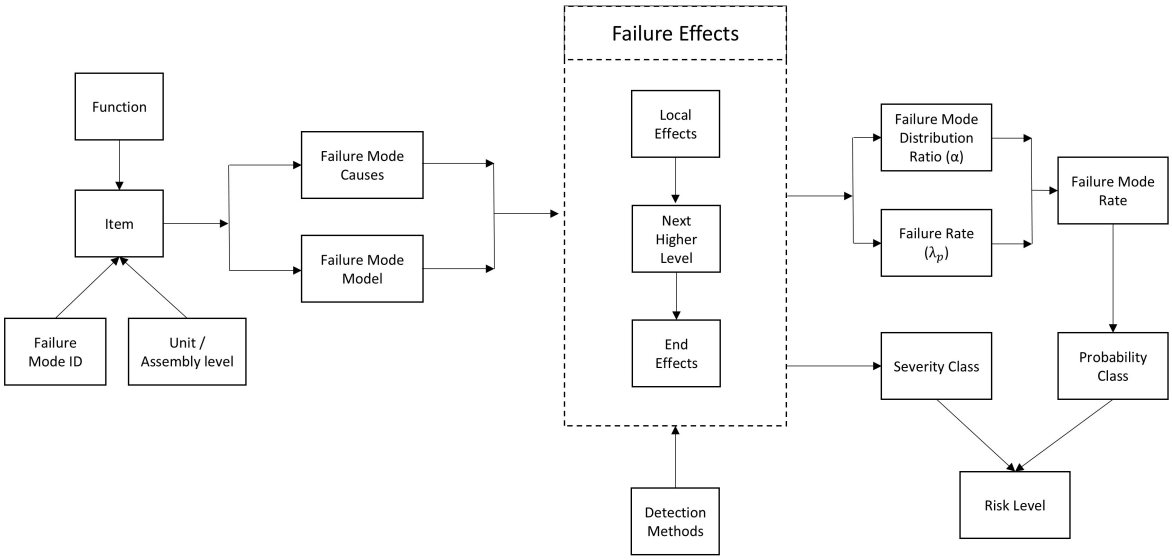


Figure 2.7: FMEA flow process in this work.

When performing a system reliability analysis, several aspects must be considered, including the reliability of individual components, the physical configuration of the system and its components, and the failure modes of the items under consideration [16]. Establishing the system's functional configuration effectively partitions it into specific units and assemblies, building the causal logic of the system. This hierarchical approach enables each component to be mapped to its parent subsystem, the individual items within those subsystems to be defined, and the specific functions performed by each component to be clearly identified.

Once the system elements have been defined and represented, the next step is to systematically identify the critical failure modes for each component and evaluate their effects on both the immediate system functions and the overall mission objectives. A given failure mode may arise from multiple underlying causes; therefore, each failure mode and its associated cause(s) must be explicitly identified and documented [16].

Modern FMEA software tools typically provide libraries containing predefined components and corresponding failure modes. This improves the quantitative strength of the overall risk assessment and facilitates the identification of the most critical system components based on the severity and impact of their failures, without requiring significant data collection from the early-stage system itself. The identification of failure modes and their associated causes, the effects of each failure mode on item operation are analyzed and documented [16]. This information is structured into three levels to track failure effects across the system:

- **Local Effects:** Describe the immediate consequences of a failure mode on the operation and function of the specific item or component under analysis.
- **Next Higher Level Effects:** Capture the impact of a failure mode as it propagates from the affected item to its parent subsystem or assembly.
- **End Effects:** Represent the ultimate consequences of a failure mode on the overall system operation, mission objectives, and system status.

Local effects describe the immediate impact of a postulated failure mode on the function and performance of the specific item under consideration. In some cases, no additional local effects can be identified beyond the failure mode itself; however, the influence of the failure on the item's output, including any secondary consequences, is documented when applicable. As failures propagate beyond the item level, their consequences manifest at higher levels, influencing the operation and functionality of the parent subsystem and, ultimately, the overall system [16]. The analysis of end effects captures this propagation, recognizing that system-level impacts may arise from the interaction or combination of multiple failure modes occurring across different components.

Once the failure modes and their respective effects have been established, the analysis shifts to identifying operational safeguards and quantifying the overall risk [16]. This process begins by identifying failure detection methods, which are elements within the system designed to prevent failures or to detect them before they propagate (e.g., alarms, diagnostic tests, and system processes that limit failure propagation). Once these controls are identified, the risk associated

with each failure mode is evaluated as the convolution of consequence severity and the probability of occurrence. There are multiple ways to develop the probability, from linguistic judgments to data-driven assessments of the true failure probability.

While FMEA is a valuable tool for gaining qualitative insights into system design and operation, it is an approach that focuses on individual component failures. As a result, it often fails to account for complex scenarios in which multiple, independent faults must occur simultaneously to trigger a system-level failure [16]. Additionally, the depth and accuracy of the analysis are highly sensitive to the resources and effort allocated. Since the process requires evaluating each failure mode in isolation, it can become excessively repetitive and labor-intensive for systems with a high density of components.

To address these gaps, FTA can be employed to identify the precise logical, physical, or interaction-based root causes of specific events [65]. Unlike the linear nature of FMEA, FTA maps the causal logic of complex systems and provides the quantitative specificity needed to calculate overall system failure probability.

2.2.2 Fault Tree Analysis (FTA)

Fault Tree Analysis (FTA) is a well-established method for evaluating system reliability by graphically representing how component failures and environmental conditions can combine to cause a system-level failure [65]. This methodology includes both qualitative analysis, which identifies the minimal cut sets" representing critical failure combinations, and quantitative analysis, using Boolean algebra to determine the system failure probability and identify critical components.

The FTA analysis begins with a top event and works backward (deductively) to identify its root causes. Fault trees break down events to explain why a failure occurs and how its probability is determined. By decomposing the top event into intermediate and basic events, fault trees clarify the mechanisms leading to failure and how the overall probability is constructed from contributing events. One of the primary advantages of FTA is its ability to quantitatively evaluate the probability of the top event using Boolean logic, while maintaining a graphical structure that is relatively easy

to read and interpret. That is, the failure events identified in the FMEA can be more rigorously assessed for probability by using the system structure established in FTA. The following describes the event elements of a fault tree:

- **Top Event:** The final undesired outcome being analyzed, representing the system-level failure of interest (e.g., complete loss of system or system unavailability, etc).
- **Intermediate Event:** An event that links lower-level causes to higher-level failures to provide logical structure to the fault tree.
- **Basic Event:** The most fundamental identified failure causes that cannot (or will not) be further developed.
- **External Event:** An event originating outside the system boundary that is assumed to occur under normal conditions.
- **Undeveloped Event:** An event that is not expanded further due to limited impact on the analysis.

Another important element to consider is the logic itself, which is specified through Boolean logic gates. These are the operators that define the logical relationships between events within the fault tree. These gates determine how input events combine to produce an output event.

- **AND gate:** The output occurs if and only if every input event takes place
- **OR gate:** The output occurs when one or more of the input events take place.
- **NOR gate (NOT OR):** The output occurs when only one of the input events occurs.
- **NAND gate (NOT AND):** The output occurs when at least one input event does not occur
- **XOR gate (Exclusive OR):** The output occurs when only one of the input event occurs.

Combining FMEA and FTA creates a synergistic approach that overcomes the individual limitations of each method, resulting in a more robust and comprehensive analysis. FMEA identifies where errors occur and their effects on the system, while FTA offers a more rigorous, top-down investigation into the root causes of undesired top events. This integration enhances error detection beyond what either method can achieve alone, providing a clearer and more complete risk assessment [66]. Although FTA details failure causes, it often lacks a mechanism to evaluate risk severity, which is a gap that FMEA fills by assessing criticality and risk levels. Additionally, this combined approach offers structural clarity by transparently linking error modes with system-level failures, a necessary feature for developing the subsequent parts of this project. Despite their relative advantages, neither method is particularly suited to tracking changes as the system evolves over time, or to communicating risk effectively among multiple stakeholders. Model-based systems engineering can be used to better incorporate these approaches into the rest of the systems engineering pipeline.

2.3 Model Based Systems Engineering (MBSE)

Systems architecture aims to maintain a complete view of a project so that the main objectives are not overshadowed by detailed technical issues or isolated subsystems [67]. The growing complexity of modern systems requires more rigorous and structured engineering methodologies. Models and simulations provide a means to represent and analyze system behavior prior to physical development, offering a clear and consistent design framework for those responsible for implementation, testing, deployment, and long-term evolution. By enabling the early identification of limitations and incompatibilities, these activities reduce the risk of costly redesigns and schedule delays, particularly during the operational phase [68].

Systems Engineering is commonly defined as the process of transforming customer needs into effective and affordable solutions, and it has traditionally relied on a document-based approach [69]. Throughout the system life cycle, numerous documents are produced by different stakeholders to record decisions and engineering outcomes. However, this document-centered

method is often inefficient, costly, and difficult to maintain, as information is repeatedly updated across multiple files and formats, increasing the risk of inconsistencies and errors. These limitations have driven the increasing adoption of Model-Based Systems Engineering (MBSE), where a centralized model replaces scattered documentation and serves as the foundation of the engineering process, providing a clear, consistent, and shared definition of the system [69].

The MBSE methodology is the formalized application of modeling to support system requirements, design, analysis, verification, and validation activities, beginning in the conceptual design phase and continuing throughout system development and later life-cycle stages. By leveraging system models as the primary means of capturing and managing system information, MBSE enhances the ability to represent, analyze, share, and maintain product specifications in a consistent and structured manner [68]. MBSE improves product quality by providing precise and unambiguous system models that can be evaluated for consistency, correctness, and completeness. The use of standardized modeling practices and abstraction mechanisms also promotes knowledge capture and reuse, leading to reduced development cycle time and lower maintenance costs [68].

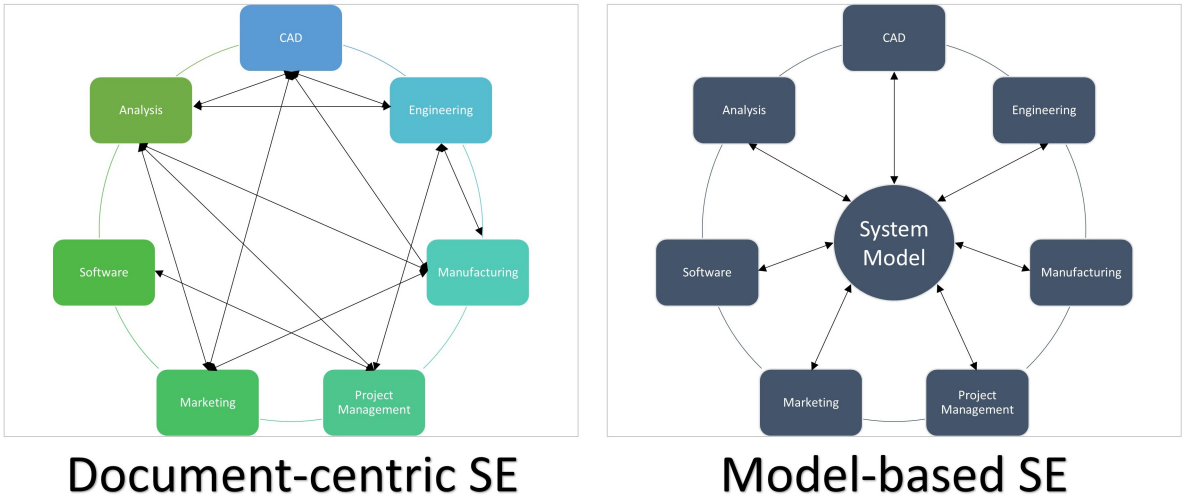


Figure 2.8: Document Centric SE vs MBSE adopted from [3]

2.3.1 System Modeling Language (SysML)

Although conceptual model development does not require the use of specific modeling language or software tool, a widely adopted practice is to apply the Unified Modeling Language (UML), a standardized method for describing system designs that originated in the software engineering community, or its systems engineering counterpart, the Systems Modeling Language (SysML). UML is a well-established industry standard that offers a consistent notation for describing system elements, their interactions, and their behavior through a set of structured diagrams [70]. SysML is a multipurpose modeling language for systems engineering that provides a standardized yet customizable toolkit for mapping interactions among hardware, software, and human elements, enabling engineers to model complex systems that integrate technology, personnel, and physical infrastructure [4].

The balance between structure and adaptability, along with the wide variety of available diagram types, makes these languages well suited for modeling system-of-systems architectures, operational contexts, and system behavior [70]. Their ability to present information from multiple dimensions allows stakeholders to focus on the specific perspectives most relevant to their roles, while still ensuring that all views remain consistent with one another. SysML has emerged as an important modeling language for systems engineering because it supports core systems engineering activities, including requirements management, structural and functional modeling, and multiple forms of allocation, while also enabling basic testing and preliminary trade studies during the specification and design phases [4, 67]. By applying object-oriented methodologies, SysML ensures that architectural principles are consistently maintained throughout the design process. Figure 2.9 presents a general overview of the diagrams that constitute SysML.

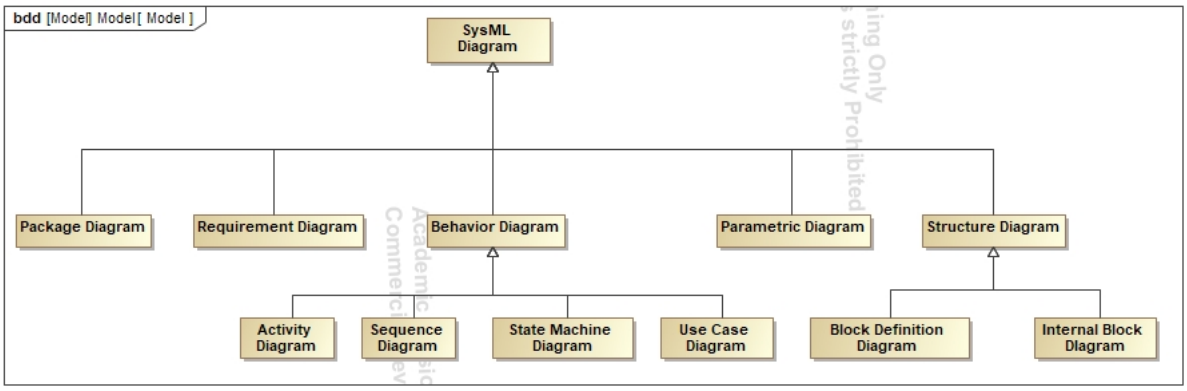


Figure 2.9: SysML Diagrams General Overview adopted from [4]

The SysML framework includes different diagrams, as shown in Figure 2.9. The following section provides a brief description of each diagram and highlights its relationship to corresponding UML representations [4, 68]:

- **Package diagram:** Similar to UML, this diagram illustrates the organization of the model through packages that group related model elements. This structure supports model navigation and promotes model re-usability, an effective access and change management.
- **Requirement diagram:** A SysML specific addition (not found in UML) that bridges the gap between text-based requirements and the system model. It establishes direct traceability between requirements, design components, and verification test cases.
- **Behavior diagrams:** Represent how a system functions over time in response to various inputs. They integrate critical aspects such as concurrency, execution logic, and state information to provide a comprehensive view of both system timing and operational behavior. These diagrams include Activity, Sequence, State Machine, and Use Case diagrams.
- **Activity diagram:** Adapted from UML, this diagram describes system behavior by modeling the transformation of how inputs are converted into outputs through a coordinated sequence of actions triggered by specific control conditions.

- **Sequence diagram:** Same as UML sequence diagram, this diagram focuses on interaction, illustrating system behavior as an ordered sequence of message exchanges between systems or system components.

- **State machine diagram:** Describes the behavior of a system element by modeling its transitions between defined states in response to events. Consistent with the UML state machine diagram, this diagram tracks how an entity transitions between different states in response to specific events, including those that occur during state entry and exit events.

- **Use case diagram:** Represents system functionality from the perspective of external entities, known as actors. It provides a description of how users or external entities interact with the system to achieve specific goals. This diagram is used in the same way as the UML use case diagram.

- **Parametric diagram:** This diagram type is unique in SysML to model system constraints for analytical purposes. These constraints often involve mass properties, reliability, and performance. Additionally, SysML supports integration with specialized engineering models to facilitate automated analysis.

- **Structure diagram:** Adapted from the UML, Structure diagram illustrates the relationships and interfaces between Parts within a Block. These diagrams include both Block Definition and Internal Block diagrams.

- **Block definition diagram:** It serves as the primary tool for defining system hierarchy, classifications, and structural characteristics using blocks. Blocks are introduced as the primary structural elements and are used to describe their properties, relationships, and composition. This diagram is a modification of the UML class diagram.

- **Internal block diagram:** Illustrates the internal structure of a system by showing how the parts of a block are interconnected through ports and connectors. This diagram is a modification of UML composite structure diagram.

2.3.2 Reliability Analysis using MBSE

As systems integrate new and advanced technology, it is important to integrate these elements early in the design phase to maintain consistency, efficiency, and overall performance. Reliability Engineers need to move beyond document-based approaches and work directly within an MBSE environment. This allows failure risk assessments to be part of the design and development process, rather than being performed only as retrospective checks, and ensures these activities add value throughout the design, development, and operational phases of the system life cycle [71]. To support this approach, a standardized UML profile for safety and reliability has been applied in MBSE, which allows for incorporating external risk/reliability analysis tools, and ensures traceability to model elements such as requirements, design components, parametric models, test cases, and test results [24].

The Risk Analysis and Assessment Modeling Language (RAAML) specification introduces extensions to SysML that support safety and reliability analysis, providing the capabilities needed for tools that use a model-based approach while still allowing traditional representations (e.g., trees, tables, etc.). This specification supports a wide range of analysis methods, including Failure Modes, Effects, and Criticality Analysis (FMECA), Limited Life Analysis (LLA), Fault Tree Analysis (FTA), maintainability and availability analysis, and Probabilistic Risk Assessment (PRA). These techniques are supported by international standards, including IEC 61025 for FTA and IEC 60812 for FMEA and FMECA, which help ensure consistency and broad applicability within the modeling profile. The specification also enables direct connections to the SysML system model, allowing analyses to be integrated with and traced back to relevant model elements [24, 72]. When FMEA is implemented within an MBSE environment using SysML, failure information can be directly tied to system models, making the analysis more consistent and easier to review. Modeling FMEA elements allows failure modes, effects, and associated system components to be traced to systems, subsystems, and requirements, improving visibility and accountability [1, 73].

Integrating FTAs within an MBSE framework can substantially improve system architecture by offering greater insight into potential failure paths and required protective measures. Through

simulation-based analysis, the probabilities of various failure events can be estimated, allowing designers to better understand and prioritize risk drivers. By associating requirement value properties with the calculated probabilities of intermediate and top-level fault tree events, engineers can evaluate whether system requirements are satisfied and identify opportunities to improve the allocation and management of failure probabilities [1].

FMEA and FTA safety analyses enable the identification of early design flaws, safety-critical functions, and associated elements at a qualitative level during the conceptual design phase of a complex technical system. However, both approaches are limited in their ability to capture intricate interactions and interdependencies among system components, as they typically assume that failures occur independently and are characterized by static probabilities [74]. This limitation becomes more pronounced when they are implemented within an MBSE environment, where system behavior is inherently interconnected and influenced by complex dependencies. These challenges are particularly significant when identifying functional independence and when failure data are incomplete, creating difficulties in parameterizing and updating the model.

BNs are well-suited for representing uncertainty and modeling dependencies between components, thereby improving the accuracy and adaptability of risk assessments. They also support probabilistic inference and allow system risk to be updated as new information becomes available [74]. By integrating traditional reliability analysis with BN, a more robust risk assessment framework can be achieved, providing a better-structured decomposition of failure pathways while capturing complex dependencies and evolving system behavior.

2.4 Bayesian Networks (BN)

FMEA and FTA are widely used industry standards for reliability assessment, but they have limitations when applied to complex systems with uncertainty. Although their results are valuable because they reflect expert knowledge and historical data, they are typically static and have difficulty representing multi-state variables or changing relationships between failures. When used together, however, FMEA and FTA can provide a strong foundational knowledge base. By translat-

ing the qualitative insights from FMEA and the logical structure of FTA into Bayesian Networks, a more robust system-level diagnosis approach can be developed. Unlike traditional methods, BNs are well suited to handling uncertainty and failure dependencies, allowing static expert knowledge to be transformed into a dynamic probabilistic framework that supports real-time diagnosis [75].

Bayesian networks are a subclass of probabilistic graphical models (PGMs) that represent structured knowledge about a system [76]. Rather than treating failures as isolated events, BNs function as causal frameworks that illustrate how various internal and external risk-influencing factors, such as human error, environmental conditions, or external interference, interact and eventually propagate into system failures. In the field of risk and safety engineering, these models are valued for their capability to formally compute the joint probability distribution over all system variables. This enables analysts to move beyond deterministic, rule-based reasoning and instead perform quantitative inference on system behavior, accounting for the combined and evolving influence of multiple interdependent causal factors on failure outcomes [21].

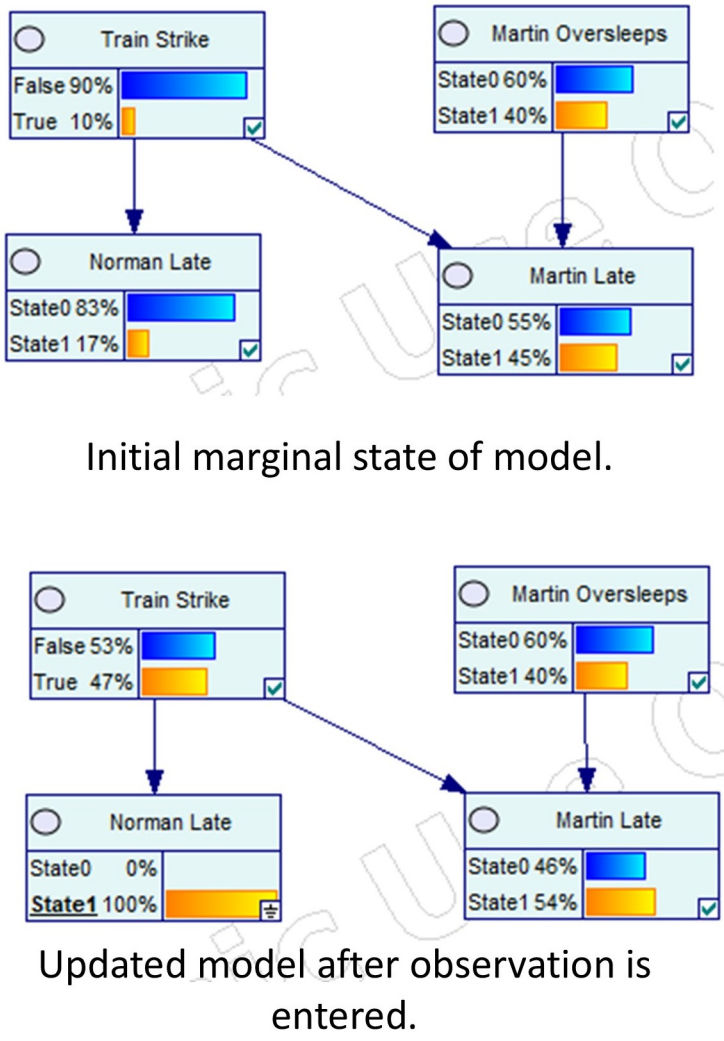


Figure 2.10: Example of BN model using GeNIe Software

BNs use directed acyclic graphs to represent the relationships among variables in a system [77]. In these graphs, each node represents a variable, and the arrows between nodes indicate direct causal dependencies; The "acyclic" requirement means that feedback loops are not allowed, one of the drawbacks of BN modeling. The strength of the causal relationships is quantified through conditional probability values. This structure allows the joint probability distribution of the system to be decomposed into a product of local conditional probabilities. The joint probability is calculated based on the network structure, expressed in Equation 2.1.

$$p(x) = \prod_{j=1}^{n} p(x_j \cap a_j) = \prod_{j=1}^{n} p(x_j|a_j) \cdot p(a_j)\,, \tag{2.1}$$

where x represents the random vector that includes all variables x_1, x_2, x_3 and x_4 and $\mathbf{a}_j$ denotes the set of parent nodes for variable $\mathbf{x}_j$. For the model shown in Figure 2.10, the complete joint probability distribution of the variables is obtained by multiplying the conditional probability of each variable given its parents, and is formulated as shown in Equation 2.2:

$$p(x1, ..., x4) = p(x_1)p(x_2)p(x_3|x_1)p(x_4|x_1, x_2), \tag{2.2}$$

Dependencies among variables are described using the definition of sets of parents and children [77]. For example Figure 2.10 the set $\{x_1, x_2\}$ contains the parent of $x4$, while the set of $\{x_3, x_4\}$ contains the children of x_1. This structural model also makes possible to analyze the effects of interventions on the system. An intervention consists of fixing the value of a variable and observing how this change affects the rest of the network [77].

When a variable is set to a specific value, the links from its parent variables are effectively ignored during the probability calculation, because the variable's value is no longer determined by its usual causal influences. Continuing with the example of Figure 2.10 if the variable x_2 is fixed to a particular state (e.g., $x_2 = 1$) its parent x_1 is effectively removed in the calculation when computing the joint probability distribution. This allows the joint PDF to be evaluated under specific assumptions about the system.

A primary challenge in implementing BNs is the assessment of conditional probabilities for every node. Each variable requires a Conditional Probability Table (CPT) defining its probability across all possible combinations (for example 1 or 0) of its parents states. The size of these tables increases combinatorially as the number of parent nodes increases, resulting in substantial complexity for large-scale systems.

To overcome these challenges, modern development environments enable the rapid creation of graphical decision models and simplified causal representations. These tools are especially useful

for strategic planning problems that are too complex to be handled intuitively. By automating much of the model-building process, these software tools eliminate the need for manually constructing large sets of equations. As a result, they act as an important intermediate step by reducing initial development time and providing a structured foundation that can later be refined into fully specified quantitative systems [78].

Chapter 3

Methodology

Reliability analysis of complex systems often requires multiple complementary methods, as individual methodologies may possess inherent weaknesses when applied in isolation. For this reason, many studies combine FMEA, FTA, and BNs to take advantage of their respective strengths. Modern engineering systems and subsystems are highly interconnected, so a fault in one component can quickly propagate through the entire system. FMEA identifies potential failure modes and their causes, FTA offers a structured logical path of system failures, and BNs provide a system-level diagnostic approach that explicitly handles uncertainty. Together, this combined framework enables a more comprehensive analysis of failure dependencies and mitigation strategies, which is crucial for understanding and managing the complex causal relationships present in large-scale systems.

Combining FMEA, FTA and BNs for risk analysis provides an effective way to manage the complexity of modern system risks and their interdependencies. Integrating these analyses within a model-based environment simplifies the development and review process, allowing engineers to address questions directly through the system model rather than relying on separate documents. The results support informed updates to the system architecture by guiding corrective actions and highlighting critical failure probabilities. As the system evolves in response, the changes are easily propagated through the linked FMEA, FTA and BN models. This integrated approach offers a unified view of system reliability, promotes consistency, reuse, and traceability across analyses, and helps ensure the system remains robust and adaptable as the design evolves [1].

This section explains the methodology used in this work, detailing how the data were initially collected and how they were processed throughout the study. It outlines the sequence of steps followed to transition from one analysis tool to the next, explaining the role of each tool within the overall framework. The objective of this section is to provide a clear overview of the analytical

process, allowing the reader to understand both the rationale behind the chosen approach and how the different techniques were integrated to support the final results.

3.1 FMEA

3.1.1 System Definition and Data Collection

Prior to conducting the FMEA and beginning the data collection process, it was first necessary to define the system to be analyzed. To properly define the system, it was divided into block diagrams in order to identify the unit and assembly (subsystem) levels. Based on this decomposition, a Functional Block Diagram (FBD) was developed to establish a clear understanding of the system's physical architecture.

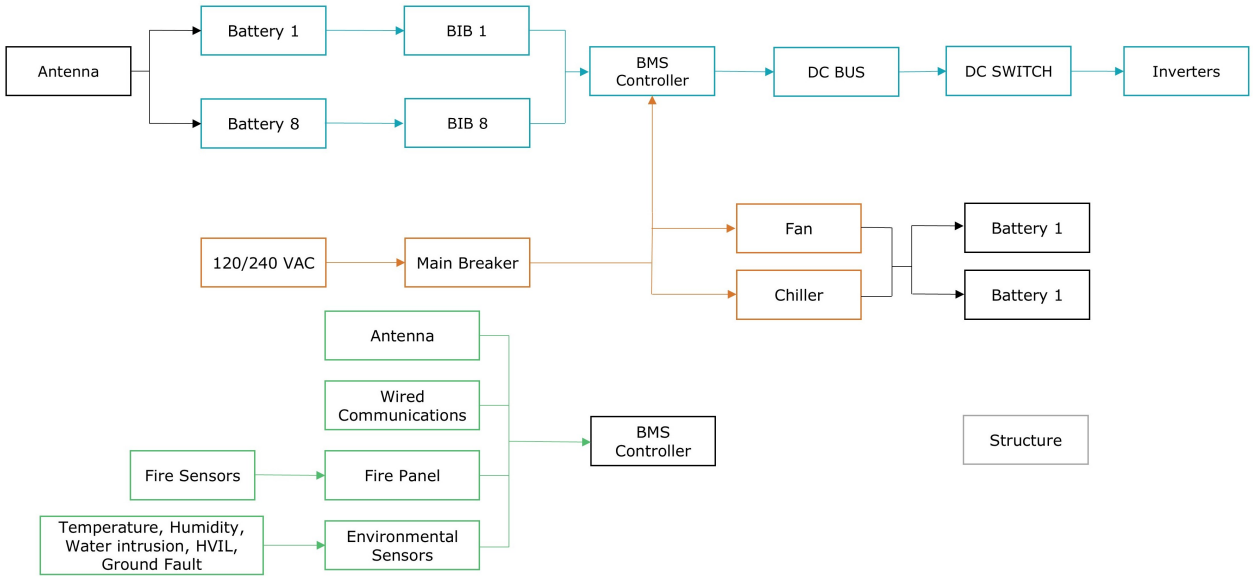


Figure 3.1: Functional Block Diagram of SV 360 BESS

During the development of the FBD, each major subsystem was identified, including the battery packs, high-voltage direct current (HV-DC) and alternating current (HV-AC) systems, communications/instrumentation & controls, cooling, logic management, and physical infrastructure. The major subsystems formed the unit level, with a unique identifier 1-7, and the components within

each subsystem were assigned unique assembly-level identifiers. This hierarchy became a structured foundation for understanding the functions performed by individual components and their roles within the overall system. Table 3.1 describes the main individual components whose coordinated operation is required for proper system functionality. The system logic was defined from information provided by an unnamed industry collaborator.

Table 3.1: System Hierarchical approach: Subsystems to Components

Unit Level	Assembly Level	Subsystem	Component
1	1.1	Battery pack	Battery pack
	1.2	Battery pack	Lithium Battery
2	2.1	HV-DC system	DC Bus
	2.2	HV-DC system	DC Switch
	2.3	HV-DC system	DC Wiring
3	3.1	HV-AC system	Aux AC Power
	3.2	HV-AC system	Main Breaker
	3.3	HV-AC system	AC Wiring
4	4.1	Communication	Fire sensor - Hydrogen
	4.2	Communication	Fire sensor - Smoke
	4.3	Communication	Fire Panel
	4.4	Communication	Wired Communications
	4.5	Communication	Antenna
	4.6	Communication	Sensor - Temperature
	4.7	Communication	Sensor - Humidity
	4.8	Communication	Sensor - Water Intrusion
	4.9	Communication	HVIL
5	5.1	Cooling system	Fan
	5.2	Cooling system	Chiller
6	6.1	Management	BMS Controller
	6.2	Management	BIB
7	7.1	Structure	Battery Cabinet
	7.2	Structure	Electronic Cabinet

Developing the FBD at the beginning of the process provided a detailed understanding of the system and its components. This structured representation helped identify the key elements that make up the system and clarified their relationships within the overall architecture. As a result, it facilitated the initial development and implementation of the FMEA worksheets, which are described in the next section.

3.1.2 FMEA Analysis

Once the system was decomposed into subsystems (units) and components (assemblies), the FMEA proceeded with the identification of component failure modes. Due to the limited existing knowledge the initial research started with understanding the operation of lithium-ion batteries, their degradation mechanisms, and the associated challenges involved in repurposing Li-ion batteries for second-life applications, as described in Section 2.1.2. This process provided most of the failure modes assigned to the battery pack and cell components.

To document these findings, a structured FMEA worksheet was developed. Each identified failure mode was represented by a single row, with columns organized to capture critical data points, including potential causes, local and system-level effects, and the quantitative parameters required to calculate the risks associated with each component.

For the remaining mechanical and electrical components, several sources could be used to obtain the necessary information, including standardized failure rate databases, manufacturer technical documentation, relevant standards and guidelines, maintenance records, and prior experience with the system or similar systems.

In this study, the Reliability Online Automated Databook System (ROADS) database was selected as the primary source. Using the observed failure modes identified in ROADS produced a more robust set of failure modes than the subjective ideation processes common in FMEA. ROADS database also simplified the estimation of several parameters required for calculating the failure mode rate. The identified components were listed in a "Failure Rate Source," which documents the components referenced from the failure catalog lists. If a component was not directly found in the catalog, an analogous component with equivalent functionality was selected, and its failure data were utilized for the FMEA.

A key component of this evaluation was estimating the likelihood that a specific failure mode will occur within a system component. This likelihood is commonly estimated using the Failure Mode Rate (λ), also known as the Failure Mode Criticality Number (Cm). This parameter represents the expected occurrence rate of a given failure mode and provides a quantitative basis for

assessing its contribution to the overall system risk. λ is normally calculated using the following equation:

$$\lambda = \alpha\beta t\lambda_p \tag{3.1}$$

Where:

- λ_p = the failure rate for all failure modes of a specific component.
- α (Failure Mode Ratio) = the fraction of component failures corresponding to the failure mode (the probability that the item fails in the identified failure mode). In the absence of data, it is often assumed that all identified failure modes are equally likely. Thus, for n failure modes, $\alpha = 1/n$.
- β (Failure Effect Probability) = the conditional probability that the failure effect with the specified criticality classification will occur, given that the failure mode occurs.
- t (Operation time) = the operating time in hours or the number of operating cycles.

The ROADS failure mode catalog provided the data required to determine α, providing the probability of specific failure modes occurring in a component. Meanwhile, λ_p was obtained by dividing the number of failed units reported in the catalog by the total number of tested hours (typically 1,000,000 hours). For this study, both the Failure Effect Probability and the Operation Time were assumed to be equal to 1 (i.e., the failure effect is certain), with β interpreted as the actual loss of the unit and t as a normalized operational duration. Once the value of λ was determined, it was used to classify each potential failure mode of an element into qualitative categories (e.g., Low, Medium–Low, Medium–High, and High) based on the probability of occurrence and the severity of its consequences.

Each failure mode was evaluated for its worst potential consequence, and a corresponding severity classification category was assigned based on criteria defined in [79], which quantifies

the extent to which these failure effects degrade system functionality. Multiple classifications of severity are possible, but this work used the following definitions:

- **Catastrophic:** A failure which can cause death or system loss
- **Critical:** A failure which can cause severe injury, major property damage, or major system damage which will result in mission loss
- **Marginal:** A failure which may cause minor injury, minor property damage, or minor systems damage which will result in delay/loss of availability or mission degradation.
- **Minor:** A failure not serious enough to cause injury, property damage, or system damage but which will result in unscheduled maintenance/repair.

The next step was to determine the overall risk level of failure of the component, which was accomplished by comparing the assigned severity class with corresponding failure rate value (λ). The chain of effects described in the Failure Effects section was used to establish the severity class, with the end effects serving as the basis for assigning the final severity rating in this column. The combination of the Severity Class and the Failure Mode Rate resulted in the evaluation of the risk level of failure for each component within the system, categorized as 1 for high, 2 for medium-high, 3 for medium-low, and 4 for low risk. Figure 3.2 illustrates the risk level matrix used to assess the failure risk of each component within the system.

Likelihood	Severity				
		Level 4	**Level 3**	**Level 2**	**Level 1**
	High	Medium	Medium-High	High	High
	Medium-High	Medium-Low	Medium	Medium-High	High
	Medium-Low	Low	Medium-Low	Medium	Medium-High
	Low	Low	Low	Medium-Low	Medium

Figure 3.2: FMEA Risk Matrix Level

The final step was to calculate the failure probability using the exponential distribution. This distribution is a standard selection for modeling the time-to-failure of non-repairable components, particularly when aging effects are not the primary focus of the initial analysis [16]. Given the assumption of a constant λ the probability that a specific component fails within a normalized operational duration t was calculated using the CDF of the exponential distribution. The CDF of the exponential distribution was applied using the following equation:

$$F(t) = 1 - e^{-\lambda t} \tag{3.2}$$

This model assumes a constant failure rate, consistent with the λ values previously extracted from the component failure catalog. The resulting probability provided the quantitative input necessary to populate the leaf nodes of a Fault Tree or the conditional probability tables within a Bayesian Network, allowing to continue to the next stage of this analysis.

3.2 MBSE Environment Implementation

3.2.1 FMEA in MBSE

Methods for system risk and failure identification, analysis, and planning are supported by a variety of tools. However, as noted in [80], there are several challenges with the application of these tools to real systems. These include the fact that the analyses can be labor-intensive and time-consuming, leading to increased costs and, in some cases, reduced motivation among the teams involved. Additionally, certain failure scenarios may be overlooked, or the identification of issues may appear subjective or arbitrary. The results are often highly dependent on the experience, skills, and background of the analysis team, and the overall process may not always be perceived as fully systematic in capturing all potential system failure risks. This study minimized these issues by using the ROADS database to identify component failure modes and structuring the FMEA and FTA within the MBSE model.

However, before conducting these analyses, it is necessary to first establish the system architecture. This process involves creating a structured framework that defines multiple architectural levels [1]. For this work, architecture development for the MBSE model began by integrating system documentation along with the requirements (Table 3.2) and translating them into an FBD to establish the structural logic of the system. Based on this functional representation, a BDD (Figure 2.6) along with an IBD (Figure 4.6) were created in order to illustrate the general system structure, the system elements and their interconnections.

Some MBSE software tools enable the performance of reliability analyses directly on system models. To conduct the reliability analysis within the MSOSA framework, the Cameo Safety and Reliability Analyzer, ISO 26262, and Cameo Simulation Toolkit plugins were installed in the MSOSA Software. These tools enable the generation of FMEA tables and FTA diagrams among other safety analysis directly within the MBSE environment, leveraging SysML and RAAML specifications. To support effective model organization, multiple package diagrams were developed, beginning with a dedicated system structure package that includes blocks representing the sys-

Table 3.2: SV 360 System Requirements

High Level Requirement	#	Low Level Requirement	Comment
1 - System Architecture Requirements	1.1	Batteries	8 batteries
	1.2	BIB	8 MANA BIBs
	1.3	Controller	1 Controller
	1.4	HVDB	1 DC Bussing Solutions
	1.5	Inverters	DC output
	1.6	Chiller	Liquid circulation sized up to 8 packs
	1.7	DC Switch	External DC Switch rated to 400 A
	1.8	Fire Detection	Fire Detection
	1.9	Circulation Fans	Circulation Fans - 240 VAC.
2 - High Level Electrical System Requirements	2.1	Voltage Rating	300 - 400 VDC
	2.2	Current Rating	C/4 continuous from battery (4 hr charge rate)
	2.3	Capacity	480 kWh per rack (correlates to 8 battery packs)
	2.4	Aux Power	480 VAC, three phase
3 - High Level Mechanical System Requirements	3.1	Approximate Dimensions	Whole system below 9.5 preferred. Max allowable width of 8.5 feet.
	3.2	Footprint	Small as possible
	3.3	Access	Front doors (2 or 3), removable panels, rear door for equipment rack, No walk in accessibility
	3.4	Installation	front load to install 8 packs
	3.5	Structural Design	Designed for ability and compliance to ship battery packs already installed in enclosure.
	3.6	Certification (Designed to)	IP 54/NEMA 3R, NFPA 69 and NFPA 855, Fire and installation standards for ESS, UL9540 UN 38.3 and UN 3481 shipping subsection.
	3.7	Environmental Controls	Racking of Batteries (20 - 45 deg C). Equipment Rack (-20 - 60 deg C)
4 - High Level Software System Requirements	4.1	BMS	Single enclosure level BMS to control all battery interface boxes and monitor all sensors throughout the enclosure.
	4.2	Control	Wireless or Ethernet connection

tem, subsystems, and components, along with a requirements package to enable the conversion of unstructured system documentation into a well-organized system representation.

Once the system architecture was established and organized within the MBSE environment, the next step involved integrating the FMEA information directly into the model. Causes of Failure are hereafter referred to as CF, Failure Modes as FM, Local Effects of Failure as LEF, and Final

Effects of Failure as FEF. To incorporate these elements, a dedicated Reliability Analysis package diagram was first developed, which incorporated the CF, FM, LEF, and FEF, and severity for each component. Failure modes were assigned to blocks, part properties, requirements, operations, and activities within the model [1], and the corresponding CF associated with the blocks representing individual system components were defined and allocated within the FMEA table. Figure 3.3 presents an FMEA table implemented in an MBSE environment for an offshore wind farm system using the Magic Systems of Systems Architect (MSOSA) software to illustrate an example for this approach.

#	Id	Subsystem	Item	Cause Of Failure	Failure Mode	Local Effect Of Failure	SEV
1	1.11	Battery Packs	Battery Modules	Improper Output	Component Functional Failure	Degraded Output	4
2	1.13	Battery Packs	Battery Modules	Expended	Component Functional Failure	Loss of Power Degraded Output	4
3	1.14	Battery Packs	Battery Modules	Electrical Failure not Determined	Component Electrical Failure	Short-Circuit Discontinuities Discharges across battery terminals	3
4	1.12	Battery Packs	Battery Modules	Broken	Component Mechanical Failure	Physical Damage	3
5	1.21	Battery Packs	Lithium Battery	Shorted	Component Electrical Failure	Short-Circuit Discontinuities Discharges across battery terminals	3
6	1.22	Battery Packs	Lithium Battery	Opened	Component Electrical Failure	Loss of electrical connections Discontinuities	3
7	1.23	Battery Packs	Lithium Battery	Expended	Component Functional Failure	Degraded Output	3
8	1.24	Battery Packs	Lithium Battery	Degraded Operation	Component Functional Failure	Degraded Output	3
9	1.25	Battery Packs	Lithium Battery	Unknown	Component Unknown	Short-Circuit Discontinuities Discharges across battery terminals Failure to Operate	4
10	1.31	Battery Packs	OEM Pyrofuse	Failure to Open	Component Functional Failure	Failure to Operate	3
11	1.32	Battery Packs	OEM Pyrofuse	Shorted	Component Electrical Failure	Voltage Fluctuation	3
12	1.33	Battery Packs	OEM Pyrofuse	Opened	Component Electrical Failure	Power Flow Interruption	3
13	1.34	Battery Packs	OEM Pyrofuse	Out of Specification	Component Functional Failure	Improper Isolation	3
14	1.35	Battery Packs	OEM Pyrofuse	Induced Failure	Component Functional Failure	Operation Outside Specification	3
15	1.36	Battery Packs	OEM Pyrofuse	Mechanical Failure not Determined	Component Mechanical Failure	Physical Damage Degraded Output	3
16	1.37	Battery Packs	OEM Pyrofuse	Worn	Component Environmental Effects	Degraded Output Physical Damage	3
17	1.38	Battery Packs	OEM Pyrofuse	Workmanship	Component Process Failure	Manufacture Errors	3
18	1.39	Battery Packs	OEM Pyrofuse	Unknown	Component Unknown	Not Determined	3
68	4.71	Communications Systems	Humidity Sensor	Degraded Operation	Component Functional Failure	Not able to detect Humidity inside Failure to Operate	2
70	4.73	Communications Systems	Humidity Sensor	No Operation	Component Functional Failure	Not able to detect Humidity inside	2
95	5.23	Cooling System	Liquid Chiller System	Failure to Operate	Component Functional Failure	Improper temperature regulation in	2
97	5.25	Cooling System	Liquid Chiller System	Unknown	Component Unknown	Improper temperature regulation in	2
107	7.11	Mechanical Structure	Battery Cabinet	Cracked Loose	Component Mechanical Failure	Difficulties in opening/close the cal	4

Figure 3.3: Example of an FMEA table in MBSE

To link all levels of the system architecture within the FMEA analysis in an MBSE environment, a set of linking rules, summarized in Table 3.3, was developed in accordance with the procedure established by [1]. These rules, combined with the inherent bottom-up logic of FMEA, were followed to create the FMEA table from the system diagrams. First, the component level was

Rule #	Source Element (Lower Level)	Target Element (Higher Level)
1	Subsystem Local Effect of Failure	System Failure Mode
2	Subsystem Final Effect of Failure	System Local Effect of Failure
3	Subsystem Failure Mode	System Cause of Failure
4	Component Local Effect of Failure	Subsystem Failure Mode
5	Component Final Effect of Failure	Subsystem Local Effect of Failure
6	Component Failure Mode	Subsystem Cause of Failure

Table 3.3: Rules for linking system architecture level in FMEA analysis adapted from [1]

linked to the subsystem level by mapping each component cause of failure to the corresponding failure mode at the subsystem level (CF-FM). Second, the local effect of failure at the component level was linked to the failure mode at the subsystem level (LEF–FM). And finally, the final effect of failure at the component level was linked, through decomposition, to the local effect of failure at the subsystem level (FEF–LEF).

These relationships enable the systematic linkage of the different FMEA levels to the system architecture, improving traceability across the model and making it easier to identify which elements are affected when updates are introduced, whether to requirements, functions, components, or other aspects of the system. Furthermore, this structured logic facilitates the integration of FMEA with FTA within the MBSE framework, providing a natural transition to the next section.

3.2.2 FTA in MBSE

FTA diagrams were built on the UML Composite Structure Diagrams. To create the system FTAs, a package element was created within the Reliability Analysis package to store both the FTA diagrams and the associated events (basic, intermediate, and top). As a top-down methodology, the analysis originates with the top event; in this work, the top events were defined to represent the total failure of each subsystem in the second-life BESS. Top events were further decomposed into intermediate events, corresponding to the total failures of components within the subsystems. The intermediate events were then decomposed as logical functions of component-level basic events. Thus, the basic events directly map to the causes of failure for each component as defined through the FMEA, and logically build up the system-level failures.

The use of UML Composite Structure Diagrams allows the specification of probabilities for basic events. Accordingly, once the FTAs were created, the next step was to assign failure probabilities to each basic event based on the exponential distribution values obtained during the FMEA analysis. Using the Cameo Simulation Toolkit, the probabilities of intermediate and top events are automatically computed when the FTA simulation is executed. Fault trees can be analyzed directly in their original form or instantiated to support multiple simulation runs. Figure 3.4 presents an example of a Fault Tree Analysis without assigned probabilities for the battery pack failure structure within the MBSE environment.

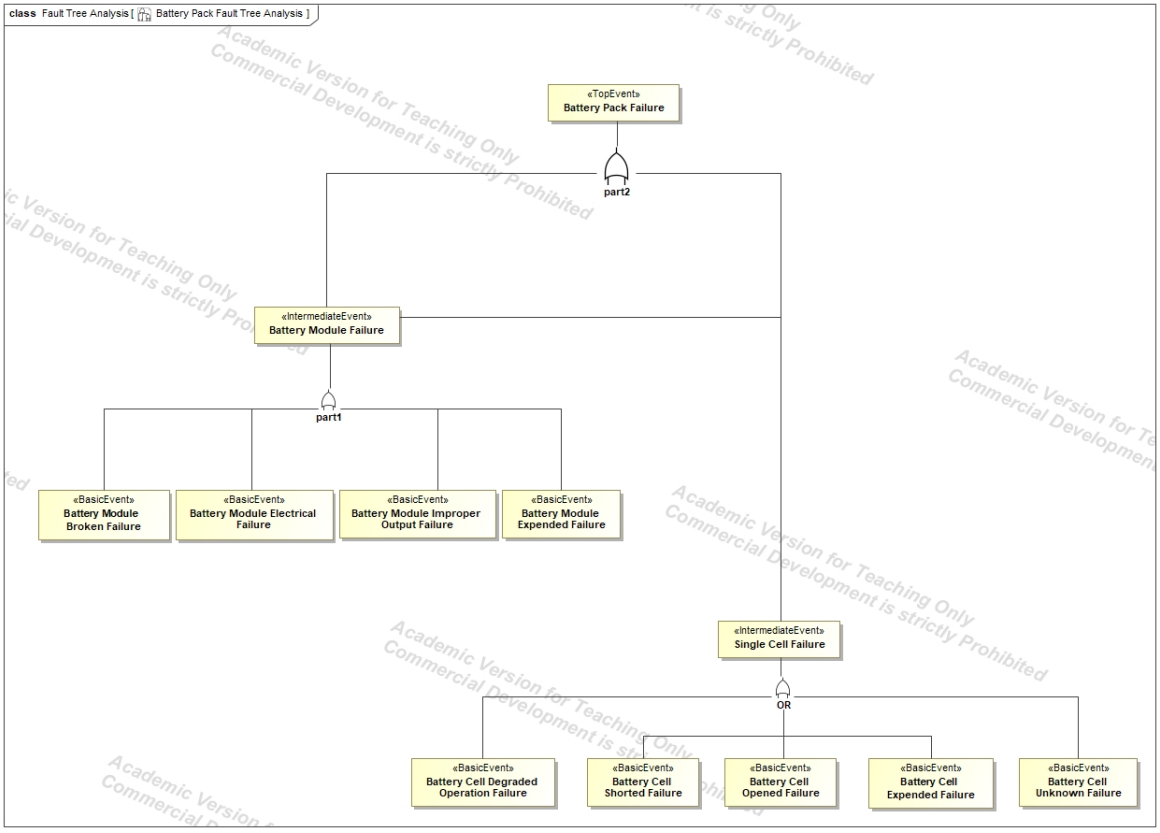


Figure 3.4: FTA example in MSOSA

When creating an MBSE model *Dependency Matrices* are used to represent dependencies between different elements in the model (e.g. UML relationships, SysML relationships, tags, etc) [81]. This tool is useful for visualizing and analyzing relationships within complex systems

in a compact and structured way. The *RelevantTo* relationship, as defined in the Risk Analysis and Assessment Modeling Language (RAAML), is used to link situations to system model elements in order to establish context and relevance for the situation [82]. FTA diagrams and the system model elements can be can interconnected with the help of these relationships.

In this model, dependency matrices are used to extend the relationships between FMEA and FTA by linking FTA basic events with corresponding causes of failures. Through this mapping, the FMEA inherits quantitative and probabilistic information derived from the FTA, such as failure likelihoods. The traceability enabled by the *RelevantTo* relationships ensures that updates in the FTA, such as changes in probability distributions, are automatically reflected in the associated FMEA elements, maintaining consistency between analytical risk models and failure analysis elements.

By identifying which system components or functions are impacted by a given failure event represented in the FTA, these relationships allow failures events to be explicitly tied to the elements they degrade or threaten, providing a bridge between behavioral risk models and the system architecture.

3.3 BN Analysis

As stated earlier in this section, the FTA structure can be mapped directly onto a BN. This transition allows uncertainty to be addressed through probabilistic reasoning and enhances the model's capability to capture complex dependencies that exceed the limitations of a traditional fault tree structure. Additionally, it supports automatic probability updating and provides a more compact representation of system failure behavior.

Several software programs provide tools for creating Bayesian networks for risk modeling. In this work, GeNIe Modeler was used. GeNIe Modeler is a graphical user interface that serves as a visual interface for the SMILE Engine (Structural Modeling, Inference, and Learning Engine), a C++ library designed for probabilistic reasoning and decision-making [78]. Within this software,

the elements required to build a BN, such as different types of nodes, arcs to connect them, and other modeling components can be created, as shown in Figure2.10.

Although the tool is designed to enable model development in a less time-consuming manner, building the models manually can still be labor-intensive and time-consuming when creating new networks. To mitigate these challenges, this work developed a Python-based automation script to facilitate the transfer of FTA models from the MBSE environment into the Bayesian network framework. This approach establishes a structured method for translating FTA models into a BN-compatible format through the use of dependency matrices and general table diagrams, combined with a Bayes Fusion Python wrapper that allows its functionality to be easily integrated, thereby automating the bridge between the MBSE environment and the BN framework, significantly reducing manual effort and improving integration between these software platforms.

Dependency matrices were used to represent the structural relationships defined in the FTA model. To use this tool in a way that clearly displays the elements composing the FTA diagram, specific criteria had to be established. This table configuration required defining a $RowElementType$, then selecting the $ColumnElementTypes$, and specifying the $Dependencycriteria$, which represents how these elements are related. In this case, the dependency criterion was defined through *connectors*.

Three distinct dependency matrices were created, each capturing a specific level of the fault tree structure:

- **Basic Event Matrix** - Maps basic events to their respective logic gates. Basic events were assigned to the $RowElementType$, while logic gates were assigned to the $ColumnElementTypes$.
- **Intermediate Event Matrix** - Maps intermediate events to the gates associated with both basic events and top events. In this matrix, logic gates were placed in the $RowElementType$, and intermediate events were placed in the $ColumnElementTypes$.

- **Top Event Matrix** - Maps top-level failure events to their primary governing gates, following the same logic as the intermediate event matrix. Logic gates were placed in the $RowElementType$, and top events were placed in the $ColumnElementTypes$.

The general table within the MBSE environment was used to extract the probabilities associated with each event and integrate them into the BN during node creation. The MSOSA software enables the automatic calculation of intermediate and top event probabilities when running the FTA interface simulation. During the simulation, these probability values can be obtained through instances, which are used for configuration purposes to extract the probability values assigned to value properties.

By instructing the software to export the instances to a specified location, they are automatically generated with the relevant information. To use the data from the instances in a general table (similar to dependency matrices), it is necessary to define the $ElementType$, which in this case is Instance Specification, and specify the $Scope$ as Instances. Once selected, all elements contained within the instances are displayed in the table. Then the data must be filtered to display only the events and their associated information. Finally the table can then be exported for subsequent processing and use.

All tables were exported as tabular data in $.csv$ format from the MBSE environment. The pandas library was then used to pre-process and adapt the data, producing a cleaner table for each case. This step was necessary to facilitate the use of the information from each table in creating the BN model.

The Python wrapper provides extensive documentation and tutorials that facilitate the development of scripts for network construction. To begin the script, it was necessary to import the $pysmile$ module, which contains the different commands for creating networks. It is important to note that a valid license is required to use this module.

With the tables prepared and the license installed, the next step was to construct an adjacency dictionary to capture the parent–child relationships using the information from the dependency matrices between events at different levels of the fault tree. When the tables are exported and

cleaned, the symbol <> (automatically generated during export) is used to indicate that an event is connected to a specific logic gate. Based on this notation, adjacency dictionaries were generated for basic, intermediate, and top events by identifying shared gates across rows and columns. These dictionaries encode the structural dependencies of the FTA and were subsequently used to create the corresponding BN nodes and define their hierarchical relationships among all events.

To create the nodes, a bottom-up approach was used, beginning with the basic events. The unique identifier for each event was extracted from the index of the corresponding dependency matrix to initialize a new node. While the SMILE library supports various node types [83] depending on the required analysis, this work used the Conditional Probability Table (CPT) nodes. These CPT nodes allow for the definition of discrete states, enabling the integration of multi-state events and specific probability distributions.

For each Basic Event, two distinct outcomes were defined *"Failure"* and *"Success"*. Where the *"Failure"* state represents the prior probability of a component malfunction, the *"Success"* state represents the probability of normal operation, calculated as the complement:

$$p(Success) = 1 - p(Failure), \tag{3.3}$$

During the instantiation process, the model dynamically populates these tables by cross-referencing the node identifiers with a dedicated probability dataset stored in the $.csv$ file ensuring that each node in the BN accurately reflects the failure data of each event. Once this was completed, the nodes for the intermediate events were created following a similar logic, establishing the structural dependencies to allow the identification of which basic nodes are related to which intermediate nodes. To better organize the network, sub-models were created to group and store the basic events associated with the corresponding intermediate event.

After the lower-level components were created, each Top Event was added and connected to its corresponding Intermediate Events based on the relationships defined earlier in the dependency matrices. These connections were represented by linking the elements together using $arcs$ between nodes in a way that preserved the structure of the original fault tree. These links show how one

event can influence another. In simple terms, they allow the model to track how failures in smaller components can contribute to larger system failures, following the logic defined in the fault tree.

In addition, a final manual step is required to implement the logic of the FTA within the BN for the child nodes. For an OR gate, the CPT is configured such that the child node results in a "Failure" state ($P = 1.0$) if any of its parent nodes are in a "Failure" state. Conversely, the child node achieves a "Success" state ($P = 1.0$) only if all parent nodes are functioning correctly. In the case of an AND gate, the child node reaches a "Failure" state only if all contributing parent nodes fail simultaneously; if even one parent remains in a "Success" state, the child node remains in "Success". The corresponding logic should be updated in case any other type of gate is used in the model.

Once the network structure was completed, the model was exported and reviewed in the GeNIe software. This step involved checking that each node correctly represented failure and success conditions, and that the probability values matched those from the original data. Reviewing the model in a graphical interface helped confirm that the network was built correctly and was ready for further analysis.

Chapter 4

Results

This chapter presents the results of the risk and reliability assessment derived from the methodologies described in Section 3. This section includes visualizations of the FMEA tables, the system architecture developed using the MBSE model, including its components, interconnections, and relevant system information, and the data used to run the models, as well as the diagrams, data structures and scripts used to complete this work.

4.1 FMEA Results

The first results of this work were the FMEA tables developed to describe the failure modes of batteries and the Second Life BESS in an FMEA format. These tables were first created in excel using the format described in [16]. Figure 4.1, Figure 4.2 and Figures 4.3 present the information from Section 2.1.2 in the FMEA format within the Cameo MSOSA environment. These figures are created from the MBSE system model.

bdd [Package] FMEA Original Table [FMEA Li-ion Batteries - Lithium Degradation Modes]

							Failure Effects			
Failure Mode ID	Unit Level	Assembly Level	Item/ Functional Identification	Function	Failure Modes and Causes	Failure Mode Model	Local Effects	Next Higher Level	End Effects	Failure Detection Method
1.211	1	2	Lithium	Save / Release Electrons	Time	Loss of Lithium Inventory (LLI)	SEI growth	Consumption of Lithium	Capacity Fade	Consumption of Lithum, increace of impedance, electrolyte breakdown and formation of SEI layer on the graphite surface
1.212	1	2	Lithium	Save / Release Electrons	High Temperature	Loss of Lithium Inventory (LLI)	SEI growth	Consumption of Lithium	Capacity Fade	Reduction/Consumption of Lithum, increace of impedance, electrolyte breakdown and formation of SEI layer on the graphite surface
1.213	1	2	Lithium	Save / Release Electrons	High Vcell/SOCcell	Loss of Lithium Inventory (LLI)	SEI growth	Consumption of Lithium	Capacity Fade	Reduction/Consumption of Lithum, increace of impedance, electrolyte breakdown and formation of SEI layer on the graphite surface
1.214	1	2	Lithium	Save / Release Electrons	Current Load	Loss of Lithium Inventory (LLI)	SEI growth	Consumption of Lithium	Capacity Fade	Reduction/Consumption of Lithum, increace of impedance, electrolyte breakdown and formation of SEI layer on the graphite surface
1.215	1	2	Lithium	Save / Release Electrons	High Temperature	Loss of Lithium Inventory (LLI)	SEI Decomposition	Consumption of Lithium	Capacity Fade	Reactive reactions in Li-ions (leading to a reduction on lithium) and SEI formation in anode electrode
1.216	1	2	Lithium	Save / Release Electrons	High Vcell/SOCcell	Loss of Lithium Inventory (LLI)	SEI Decomposition	Consumption of Lithium	Capacity Fade	Reactive reactions in Li-ions (leading to a reduction on lithium) and SEI formation in anode electrode
1.217	1	2	Lithium	Save / Release Electrons	Current Load	Loss of Lithium Inventory (LLI)	SEI Decomposition	Consumption of Lithium	Capacity Fade	Reactive reactions in Li-ions (leading to a reduction on lithium) and SEI formation in anode electrode
1.218	1	2	Lithium	Save / Release Electrons	High Temperature	Loss of Lithium Inventory (LLI)	Electrolyte decomposition	Generate resistive films	Capacity Fade	Electrolyte oxidation (Production of CO2, CO gas, and H2O), Loss of performance, SEI formation and increase of impedance
1.219	1	2	Lithium	Save / Release Electrons	High Vcell/SOCcell	Loss of Lithium Inventory (LLI)	Electrolyte decomposition	Chemical reactions inside the batteries	Capacity Fade	Electrolyte oxidation (Production of CO2, CO gas, and H2O), Loss of performance, SEI formation and increase of impedance

Figure 4.1: Li-ion Battery FMEA - Lithium Degradation Modes and Effects

bdd [Package] FMEA Original Table [FMEA Li-ion Batteries - Cathode and Anode Degradation Modes - Parrt 1]

							Failure Effects			
Failure Mode ID	Unit Level	Assembly Level	Item/ Functional Identification	Function	Failure Modes and Causes	Failure Mode Model	Local Effects	Next Higher Level	End Effects	Failure Detection Method
1.219	1	2	Lithium	Save / Release Electrons	High Vcell/SOCcell	Loss of Lithium Inventory (LLI)	Electrolyte decomposition	Chemical reactions inside the batteries	Capacity Fade	Electrolyte oxidation (Production of CO2, CO gas, and H2O), Loss of performance, SEI formation and increase of impedance
1.241	1	2	Anode / Cathode	High-energy capacity / Storage or Accept and Release lithium ions repeatedly and quickly	High Temperature	Loss of Active Matherial (LAM)	Binder decomposition	Exothermal Reactions / Thermal Runaway	Capacity Fade / Power fade	Mechanical instability and Loss of Lithium
1.242	1	2	Anode / Cathode	High-energy capacity / Storage or Accept and Release lithium ions repeatedly and quickly	High Vcell/SOCcell	Loss of Active Matherial (LAM)	Binder decomposition	Structure Destabilization	Capacity Fade / Power fade	Mechanical instability and Loss of Lithium
1.221	1	2	Anode	High-energy capacity / Storage	High Vcell/SOCcell	Loss of Active Matherial (LAM)	Graphite exfoliation	Material Cristalization	Capacity Fade / Power fade	Gas formation and changes/desintegration of structure in graphite electrode
1.222	1	2	Anode	High-energy capacity / Storage	Current Load	Loss of Active Matherial (LAM)	Graphite exfoliation	Material Cristalization	Capacity Fade / Power fade	Gas formation and changes in structure in graphite electrode
1.231	1	2	Cathode	Accept and Release lithium ions repeatedly and quickly	Current Load	Loss of Active Matherial (LAM)	Structural disordering	Chemical reactions with compounds in the battery cells.	Capacity Fade / Power fade	Heat increase inside the battery due to oxigen release
1.251	1	2	Anode / Lithium	High-energy capacity / Storage or Save / Release Electrons	Low Temperature	Loss of Active Matherial (LAM) / Loss of Lithium Inventory (LLI)	Lithium plating/dendrite formation	Aging due to Li Metal Formation on the graphite surface	Capacity Fade / Power fade	Electrical Shorts in the Anode/Electrolyte zone by charging batteries at low temperatures
1.252	1	2	Anode / Lithium	High-energy capacity / Storage or Save / Release Electrons	Stoichiometry	Loss of Active Matherial (LAM) / Loss of Lithium Inventory (LLI)	Lithium plating/dendrite formation	Aging due to Li Metal Formation on the graphite surface	Capacity Fade / Power fade	Electrical shorts in the anode/electrolyte zone due to chemical reactions that fully lithiate the graphite surface
1.253	1	2	Anode / Lithium	High-energy capacity / Storage or Save / Release Electrons	Low Vcell/SOCcell	Loss of Active Matherial (LAM) / Loss of Lithium Inventory (LLI)	Lithium plating/dendrite formation	Aging due to Li Metal Formation on the graphite surface	Capacity Fade / Power fade	Electrical Shorts in the Anode/Electrolyte zone due to changes in charging conditions (high charging rate and overcharging)
1.243	1	2	Anode / Cathode	High-energy capacity / Storage or Accept and Release lithium ions repeatedly and quickly	Current Load	Loss of Active Matherial (LAM)	Loss of electric contact	Disconnection & Contact Reduction	Capacity Fade / Power fade	Contact loss in the composite porous electrode between the current collector and the active layer or within the active layer itself. Contact loss between the active particles or between the electrode particles and the conductive additive.

Figure 4.2: Li-ion Battery FMEA - Cathode and Anode Degradation Modes and Effects. Part 1

bdd [Package] FMEA Original Table [FMEA Li-ion Batteries - Cathode and Anode Degradation Modes - Part 2]

							Failure Effects			
Failure Mode ID	Unit Level	Assembly Level	Item/ Functional Identification	Function	Failure Modes and Causes	Failure Mode Model	Local Effects	Next Higher Level	End Effects	Failure Detection Method
1.244	1	2	Anode / Cathode	High-energy capacity / Storage or Accept and Release lithium ions repeatedly and quickly	Mechanical Stress	Loss of Active Matherial (LAM)	Loss of electric contact	Disconnection & Contact Reduction	Capacity Fade / Power fade	Contact loss in the composite porous electrode between the current collector and the active layer or within the active layer itself. Contact loss between the active particles or between the electrode particles and the conductive additive.
1.245	1	2	Anode / Cathode	High-energy capacity / Storage or Accept and Release lithium ions repeatedly and quickly	Low Vcell/SOCcell	Loss of Active Matherial (LAM)	Loss of electric contact	Disconnection & Contact Reduction	Capacity Fade / Power fade	Contact loss in the composite porous electrode between the current collector and the active layer or within the active layer itself. Contact loss between the active particles or between the electrode particles and the conductive additive.
1.246	1	2	Anode / Cathode	High-energy capacity / Storage or Accept and Release lithium ions repeatedly and quickly	Current Load	Loss of Active Matherial (LAM)	Electrode particle cracking	Strain differences and Diffusion Induced Stress (DIS)	Capacity Fade / Power fade	SEI growth, causing further loss of lithium inventory. Loss in electronic conductivity. Complete fracture or complete detachment from the binder
1.247	1	2	Anode / Cathode	High-energy capacity / Storage or Accept and Release lithium ions repeatedly and quickly	Mechanical Stress	Loss of Active Matherial (LAM)	Electrode particle cracking	Strain differences and Diffusion Induced Stress (DIS)	Capacity Fade / Power fade	SEI growth, causing further loss of lithium inventory. Loss in electronic conductivity. Complete fracture or complete detachment from the binder
1.232	1	2	Cathode	Accept and Release lithium ions repeatedly and quickly	Stoichiometry	Loss of Active Matherial (LAM)	Transition metal dissolution	Reduction of metals in batteries	Capacity Fade / Power fade	Oxidation of electrolytes
1.233	1	2	Cathode	Accept and Release lithium ions repeatedly and quickly	Low Vcell / SOCcell	Loss of Active Matherial (LAM)	Transition metal dissolution	Reduction of metals in batteries	Capacity Fade / Power fade	Oxidation of electrolytes
1.248	1	2	Anode / Cathode	High-energy capacity / Storage or Accept and Release lithium ions repeatedly and quickly	Low Vcell / SOCcell	Loss of Active Matherial (LAM)	Corrosion of current collectors	Cathode Efficiency Reduction, Electrical Resistance Increase, Electrolyte Contamination, and Anode issues	Capacity Fade / Power fade	Reduction of cathode efficiency, increase of electrical resistance inside the battery, electrolytes contamination, self-discharge of the battery and deficiencies in Anode due to de corrosion generated in current collectors.

Figure 4.3: Li-ion Battery FMEA - Cathode and Anode Degradation Modes and Effects. Part 2

Due to the number of components in a system such as the BESS, it was decided to display only the subset of components contributing to the highest risks, as shown in Figure 4.4. Although the BESS uses second-life batteries, the highest risks are associated with critical components that support system operation and maintain optimal conditions. These components have more than one high-risk failure mode identified. Therefore, attention should be paid to the safety-critical components responsible for these functions.

The complete set of FMEA tables can be found in Appendix A.

Failure Mode ID	Unit Level	Assembly Level	Item/ Functional Identification	Function	Failure Modes and Causes	Failure Effects			Severity Class	Item Failure Rate λp	Failure Mode Distribution Ratio α	Failure Mode Rate λ	Prob. Class	Risk Matrix Level	Exponential Distribution Value (CDF)
						Local Effects	Next Higher Level	End Effects							
1.13	1	1	Battery pack	Store and Develop Power	Electrical Failure	Short circuits; Increase in internal resistance	Damage to batteries and internal components; Increased temperature in batteries	Capacity fade and Power fade on affected batteries	2	2.99E-05	0.286	8.56E-06	Medium-Low	3	5.73E-03
4.11	4	1	Fire Sensors - H2	Detect Hydrogen	Failure to detect hydrogen	Unable to detect hydrogen inside the battery cabinet	Fire/explosion	Loss of system; Threat to human health	1	5.03E-05	0.50	2.52E-05	Medium-High	1	3.33E-02
4.21	4	2	Fire Sensors - Smoke	Detect Smoke	Shorted	Unable to detect Smoke inside the battery cabinet	Fire propagation inside the battery cabinet	Loss of system; Threat to human health	1	2.19E-03	1.00	2.19E-03	High	1	7.71E-01
4.81	4	8	Water Intrusion Sensor	Measure the water level inside the Battery Chamber	Improper Output	Degraded operation of component	Inoperability; Unable to detect water intrusion leading to possible short circuits	System Unsafe Operation; Possible fire inside the battery cabinet	2	1.37E-05	0.552	7.54E-06	Medium-Low	3	5.05E-03
4.91	4	9	High Voltage Interlock Loop (HVIL)	Disables the system in response to safety-critical events (Open door, External stop, Physical remote disconnect)	Improper Output	System operates in unsafe condition	Short circuits; Unsafe electrical environment	Personnel injury; Uncontrolled operation; Unsafe shutdown	2	6.31E-05	0.525	3.32E-05	Medium-High	2	2.77E-02
5.21	5	2	Chiller	Cools batteries	Leakage	Loss of coolant, reducing heat dissipation	Water intrusion into the cabinet, leading to a short circuit; Reduction of chiller performance	Increased battery temperature, potentially leading to a possible fire in the system	1	7.31E-04	0.44	3.18E-04	Medium-High	1	1.92E-01
5.22	5	2	Chiller	Cools batteries	Failed to Operate	Improper temperature regulation leading to an increase in temperature in the cabinet	Increased temperature in battery cells; Increase of LLI and LAM in affected batteries; Thermal runaway	Capacity fade and power fade in affected batteries; Potential fire in the cabinet	1	7.31E-04	0.109	7.97E-05	Medium-High	1	5.21E-02
6.21	6	2	BMS Controller	Regulates battery state of charge (SOC), temperature, and voltage during charging and discharging processes to enhance battery performance and lifespan	Improper Output	Fails to manage SOC during charging/discharging; Improper management of temperature	Excessive battery charging or discharging; Battery pack overheating, resulting in reduced battery lifespan; Increased LLI and LAM in impacted batteries	Capacity fade and Power fade on affected batteries	2	3.00E-04	0.35	1.05E-04	Medium-High	2	6.82E-02
6.23	6	2	BMS Controller	Regulates battery state of charge (SOC), temperature, and voltage during charging and discharging processes to enhance battery performance and lifespan	Electrical Failure	System operates in unsafe condition	Unsafe electrical environment; Improper management of system parameters	System unsafe shutdown; Damage to other components; Reduction of system efficiency	2	3.00E-04	0.35	1.05E-04	Medium-High	2	6.82E-02

Figure 4.4: FMEA results showing the highest-risk components

4.2 MBSE Models

This section describes the MBSE models developed using the MSOSA tool, illustrating the system's structure, behavior, and interactions. These diagrams provide a visualization of the system architecture, including its components, subsystems, and interconnections. Additionally, this section includes the results of the reliability analysis, such as FMEA tables and FTA diagrams, along with the corresponding tables and matrices used in the subsequent stages of the project.

4.2.1 SV 360 General Structure MBSE Model

This first part illustrates the system's physical architecture using BDD and IBD diagrams. Figure 4.5 shows the overall physical architecture of the system, including all subsystems and components, while Figure 4.6 presents the internal physical architecture, highlighting the interconnections between components and subsystems. In addition, the SV 360 requirements, as described in

Table 3.2, were also incorporated into the model using Requirements Diagram and summarized with Requirements Table in the MBSE model (Figure 4.7).

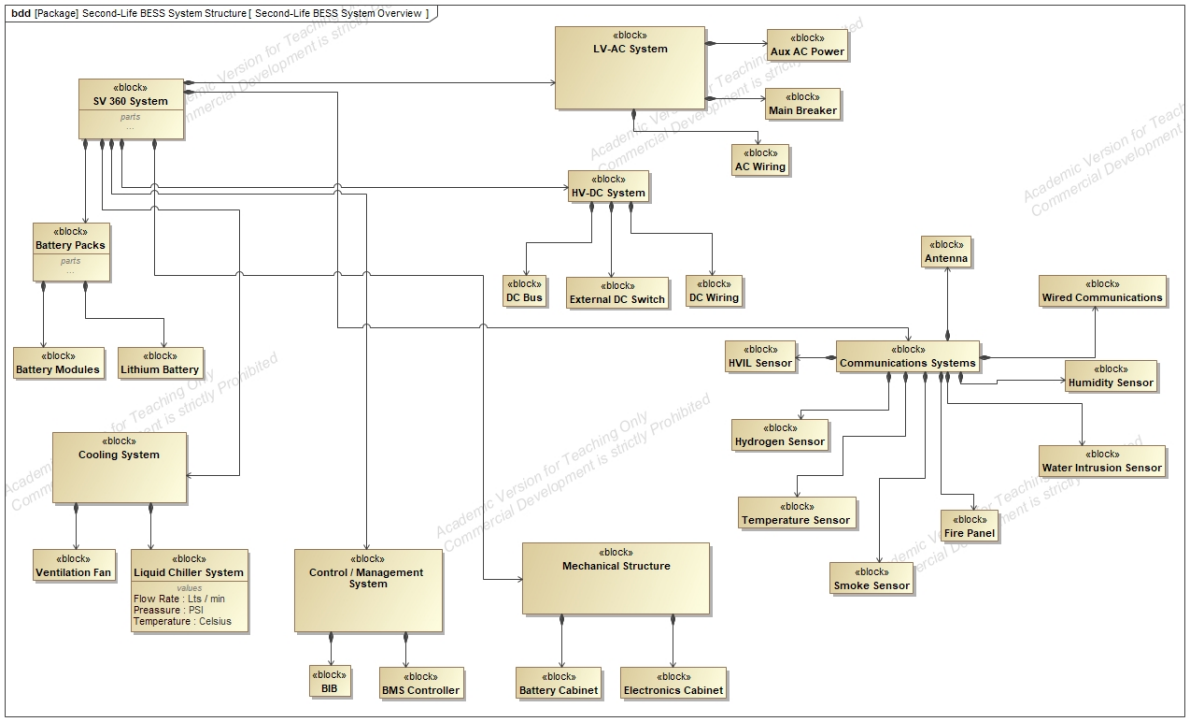


Figure 4.5: BDD of SV 360 BESS with all Subsystems and Components

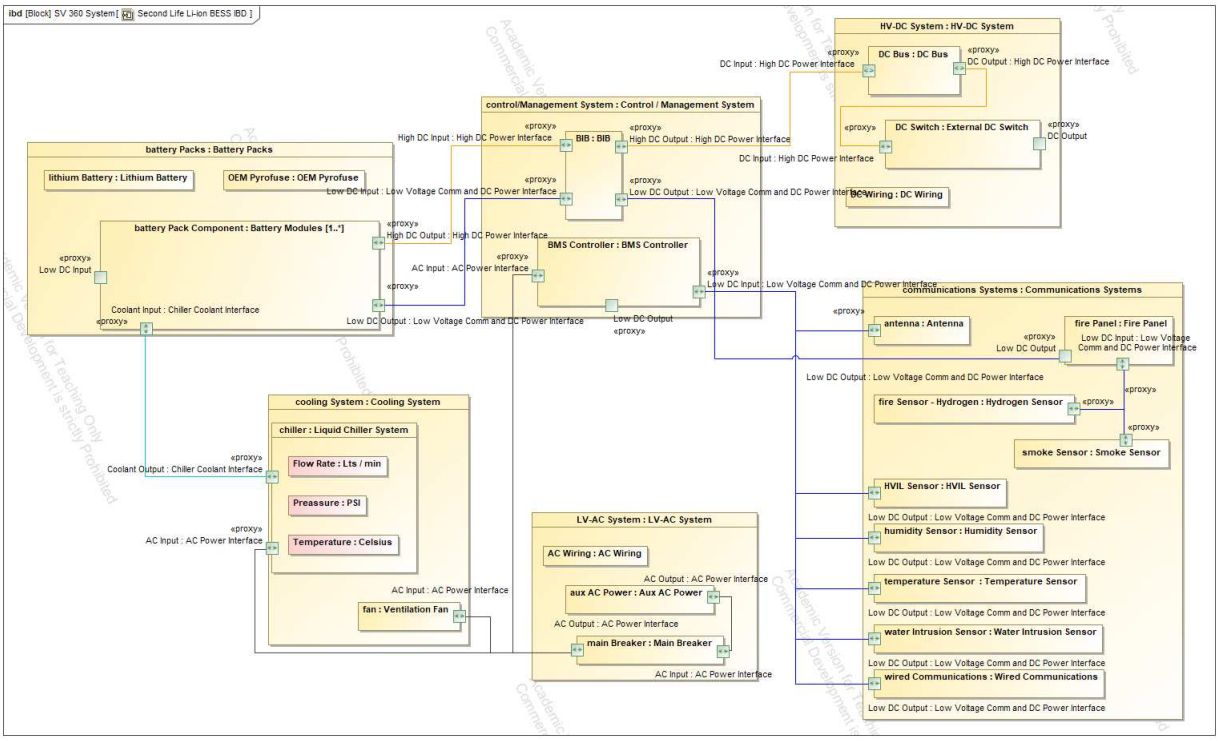


Figure 4.6: IBD of SV 360 BESS with all Subsystems and Components

#	Name	Text
1	1 System Architecture Requirements	High Level Architecture Component Requirements for the SV360s V2.2 BESS to be designed.
2	1.1 Batteries	8 batteries
3	1.2 BIBs	8 MANA BIBs
4	1.3 Controller	1 Controller with New iteration
5	1.4 HVDB	1 DC Bussing Solutions possibly off the shelf PDU or iteration of HVDB
6	1.5 Inverters	2 Sinexcel inverters removed, DC output.
7	1.6 Chiller	replaced HVAC with liquid chiller/heater, liquid circulation sized up to 8 packs
8	1.7 DC Switch	External DC Switch rated to 400 A
9	1.8 Fire Detection	Fire detection added
10	1.9 Circulation Fans	Circulation Fans changed from 480 VAC to 240 VAC.
11	2 High Level Electrical System Requirements	High Level Electrical Requirements for the SV360s V2.2 BESS to be designed.
12	2.1 Voltage Rating	300 - 400 VDC
13	2.2 Current Rating	C/4 continuous from battery (4 hr charge rate)
14	2.3 Capacity	480 kWh per rack (correlates to 8 Tesla)
15	2.4 Aux Power	480 VAC, three phase
16	3 High Level Mechanical System Requirements	High Level Mechanical Requirements for the SV360s V2.2 BESS to be designed.
17	3.1 Approximate Dimensions	Max allowable height of 9.5 feet of batteries. Whole system below 9.5 preferred. Max allowable width of 8.5 feet.
18	3.2 Footprint	Small as possible, while satisfying other requirements
19	3.3 Access	Front doors (2 or 3), removable panels elsewhere, rear door for equipment rack. No walk in accessibility
20	3.4 Installation	Open to any method to install 8 packs, front load
21	3.5 Structural Design	- 8 battery packs. - Designed for ability and compliance to ship battery packs already installed in enclosure.
22	3.6 Certification (Designed to)	- IP 54/NEMA 3R weather rating for the battery environment. - NFPA 69 and NFPA 855 - Fire and installation standards for ESS. - UL9540 - ESS Requirements - UN 38.3 and the subsection UN 3481 for shipping.
23	3.7 Environmental Controls	- Part 1: Racking of Batteries (20 - 45 deg C). - Part 2: Equipment Rack (-20 - 60 deg C).
24	4 High Level Software System Requirements	High Level Software Requirements for the SV360s V2.2 BESS to be designed.
25	4.1 BMS	A single enclosure level BMS will be implemented to control all battery interface boxes and monitor all sensors throughout the enclosure.
26	4.2 Control	High Level Control Requirements Communication Methods: Wireless or ethernet connection

Figure 4.7: SV 360 Systems Requirements

4.2.2 Example of FMEA table results in MBSE

This section shows the FMEA tables for the Battery Pack subsystem, including components, failure effects, and the calculated severity levels for each item. The information in these tables

is derived from the FMEA results presented in the first part of this chapter. The FMEA tables containing information on the other subsystems can be found in the Appendix A.

#	Id	Subsystem	Item	Cause Of Failure	Failure Mode	Local Effect Of Failure	SEV
1	1.11	Battery Packs	Battery Modules	Improper Output	Component Functional Failure	Degraded Output	4
2	1.13	Battery Packs	Battery Modules	Expended	Component Functional Failure	Loss of Power Degraded Output	4
3	1.14	Battery Packs	Battery Modules	Electrical Failure not Determined	Component Electrical Failure	Short-Circuit Discontinuities Discharges across battery terminals	3
4	1.12	Battery Packs	Battery Modules	Broken	Component Mechanical Failure	Physical Damage	3
5	1.21	Battery Packs	Lithium Battery	Shorted	Component Electrical Failure	Short-Circuit Discontinuities Discharges across battery terminals	3
6	1.22	Battery Packs	Lithium Battery	Opened	Component Electrical Failure	Loss of electrical connections Discontinuities	3
7	1.23	Battery Packs	Lithium Battery	Expended	Component Functional Failure	Degraded Output	3
8	1.24	Battery Packs	Lithium Battery	Degraded Operation	Component Functional Failure	Degraded Output	3
9	1.25	Battery Packs	Lithium Battery	Unknown	Component Unknown	Short-Circuit Discontinuities Discharges across battery terminals Failure to Operate	4
10	1.31	Battery Packs	OEM Pyrofuse	Failure to Open	Component Functional Failure	Failure to Operate	3
11	1.32	Battery Packs	OEM Pyrofuse	Shorted	Component Electrical Failure	Voltage Fluctuation	3
12	1.33	Battery Packs	OEM Pyrofuse	Opened	Component Electrical Failure	Power Flow Interruption	3
13	1.34	Battery Packs	OEM Pyrofuse	Out of Specification	Component Functional Failure	Improper Isolation	3
14	1.35	Battery Packs	OEM Pyrofuse	Induced Failure	Component Functional Failure	Operation Outside Specification	3
15	1.36	Battery Packs	OEM Pyrofuse	Mechanical Failure not Determined	Component Mechanical Failure	Physical Damage Degraded Output	3
16	1.37	Battery Packs	OEM Pyrofuse	Worn	Component Environmental Effects	Degraded Output Physical Damage	3
17	1.38	Battery Packs	OEM Pyrofuse	Workmanship	Component Process Failure	Manufacture Errors	3
18	1.39	Battery Packs	OEM Pyrofuse	Unknown	Component Unknown	Not Determined	3

Figure 4.8: SV 360 FMEA MBSE Table for Battery Pack System

4.2.3 Example of FTA diagram in MBSE

This section illustrates the Fault Tree Analysis Diagram for the Cooling Subsystem. The basic events are shown with their respective assigned failure probabilities, derived from the calculated values of the exponential distribution. As depicted in the figure, each of the intermediate and top events currently has a probability of zero. These are value properties that will be updated once the model is executed and later stored in an instance. The FTA diagrams with information on the other subsystems can be found in the Appendix B.

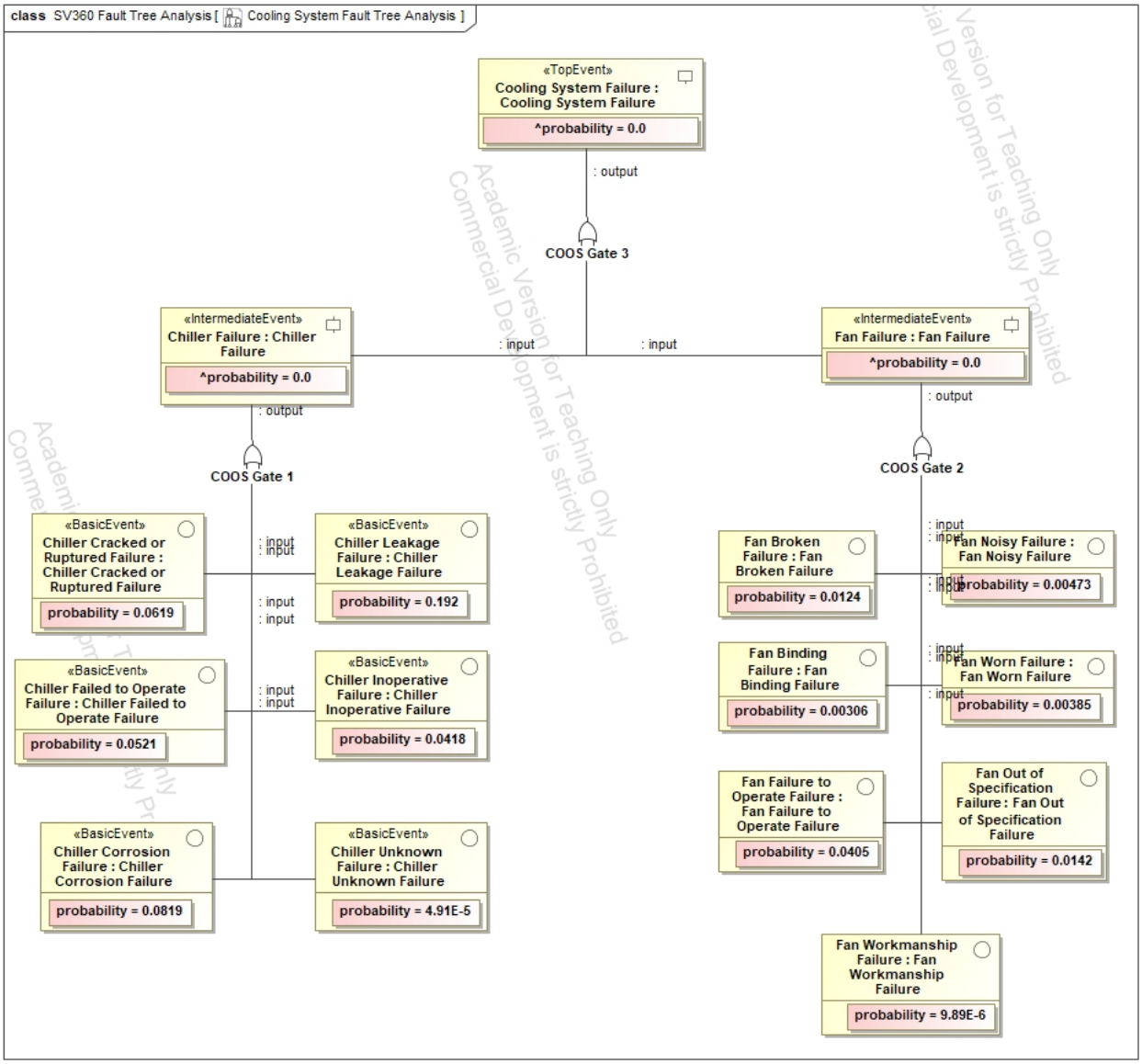


Figure 4.9: SV 360 MBSE Cooling System FTA

4.2.4 Relationship Tables MBSE

This section presents the dependency matrix that illustrate the connections between intermediate events and model elements. Although the nature of these tables results in a condensed visual format here, the high-resolution output generated by the MSOSA software when saving images allows users to zoom in and clearly examine individual cells and event labels without any loss of clarity. Figures 4.10 illustrates the dependency matrix that maps intermediate events to their cor-

responding logic gates. The dependency matrices and relationship tables generated from this work can be found in Appendix C.

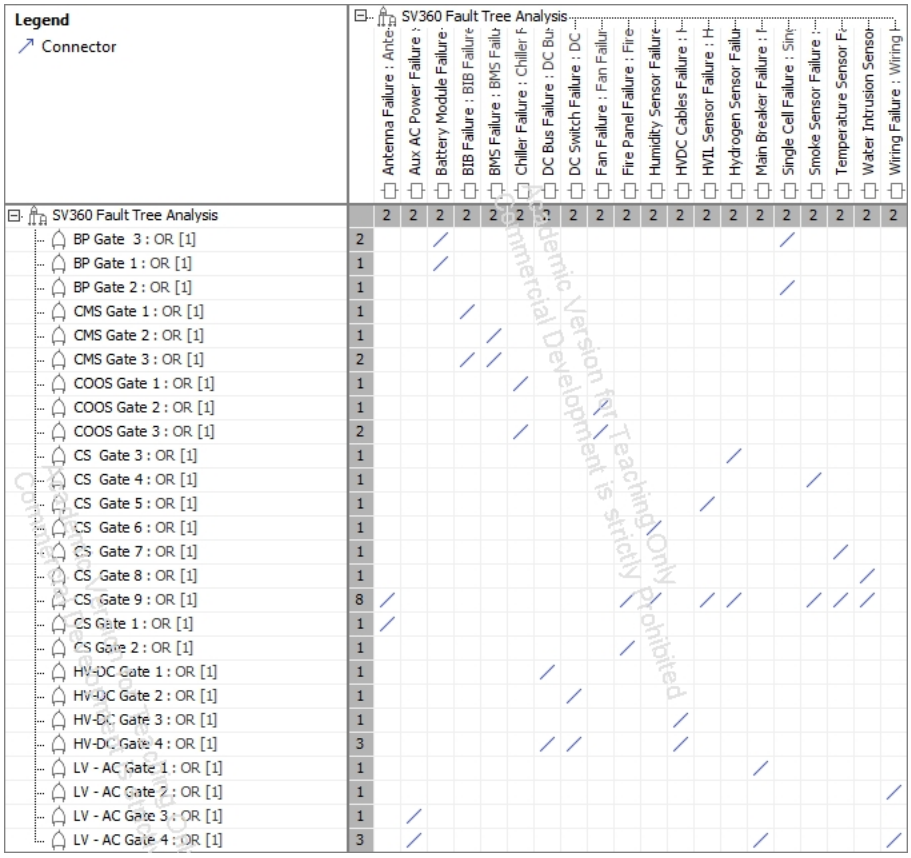


Figure 4.10: Dependency Matrix for Intermediate FTA Events

4.2.5 MBSE-BN Integration

A critical novelty of this work is the fusion of MBSE and Bayesian Networks, including the development of a Python script that automatically creates Bayesian Networks from MBSE models. This section presents the resulting BN, illustrated in Figure 4.11, once the elements are connected and properly incorporated. The scripts developed to perform the automation can be found in Appendix D.

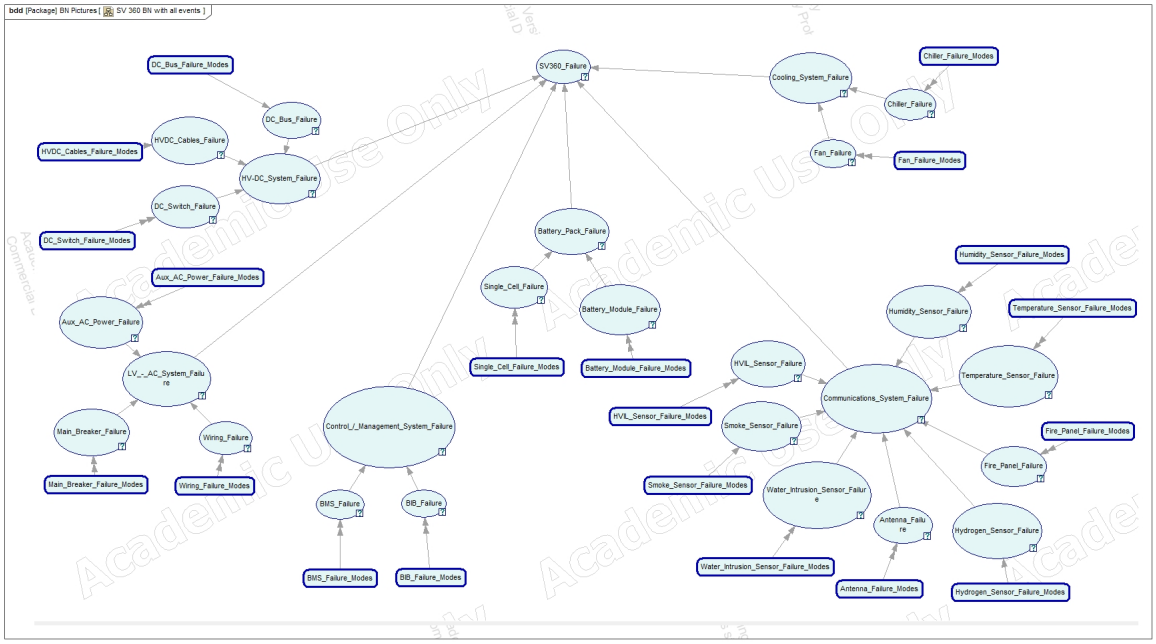


Figure 4.11: SV 360 Bayesian Network with all events

4.3 BN Assessment

igure 4.12 illustrates the probability distribution of events within the SV-360 network. The BN models the top event, defined as the failure of the SV-360 system, as a function of its subsystem failures. The blue-outlined nodes represent submodels containing basic events. These are linked via arcs to intermediate nodes which, as detailed in Chapter 3, utilize CPTs to define Failure and Success states. This structure recreates the logic of the Fault Tree Analysis (FTA) from Section 4.2.3 while providing a framework that can be easily updated with non-linear dependencies or new data as the system design matures.

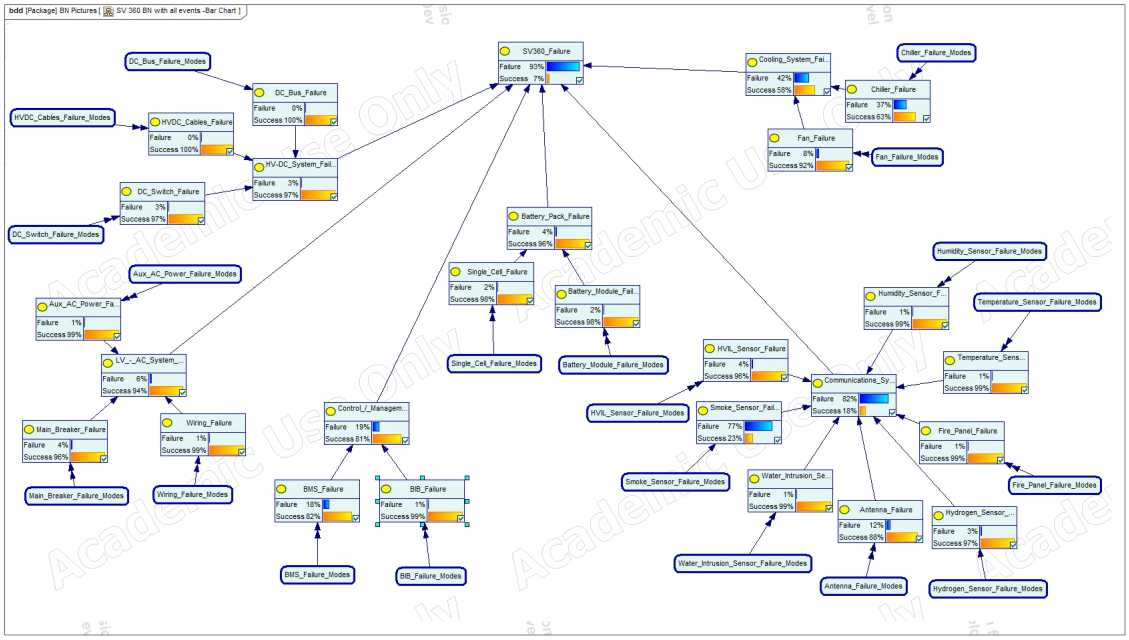


Figure 4.12: SV 360 Bayesian Network with all events - Bar chart

Chapter 5

Discussion

This research not only demonstrated the risk and reliability characteristics of a specific BESS implementation, it also developed a novel approach to the application of risk analysis through MBSE and connected approaches.

5.1 Results Discussion

As previously mentioned, even though the BESS uses second-life batteries, the FMEA results indicate that the components with the highest risk are those responsible for ensuring proper operation and maintaining suitable environmental conditions. Particular attention should be given to safety-critical components such as chillers, the BMS controller, and various sensors. These components were found to exhibit multiple high-risk failure modes. Therefore, it is necessary to implement effective detection methods to help reduce overall failure risks.

Detection and mitigation strategies aimed at preventing these failures or reducing their likelihood should be further investigated. These results do not imply that battery packs require less attention, as they remain among the elements with the highest risk of failure, only that non-battery components should not be assumed reliable or lower priority. FMEA results are largely confirmed by the subsequent FTA and BN analyses, which provide additional detail on the causal relationships between component-level and system-level failures.

Building on the FMEA, the FTA models show how failures can spread through the system. OR gates are used to represent a structure in which the system fails if any individual component fails. This creates a single-point-of-failure condition, which provides a sensitive baseline for identifying the system's most vulnerable elements. Under this logic, the Communications and Cooling subsystems appear as the main contributors to system unreliability, although for different reasons.

In the Communications System, the risk is primarily related to system complexity. Although most components have low failure probabilities, the large number of components means that these

probabilities accumulate and increase the likelihood of subsystem failure. This effect is further influenced by the relatively higher failure probability of the smoke sensor. In contrast, the risk in the Cooling System is mainly driven by the chiller. Although the chiller has a relatively high probability of failure, the overall subsystem risk is less influenced by the accumulation of component failures because the subsystem contains fewer components. As identified in the FMEA, both the smoke sensor and the chiller have relatively high risk profiles and therefore represent critical components whose failure modes can significantly affect overall system reliability.

The vulnerabilities identified in both the FMEA and the FTA can also be observed in BNs. Converting a Fault Tree into a BN involves creating conditional probability distributions that represent the logic of the OR gates. Initially, the network is still binary, meaning that a component either works or fails. However, BNs allow analysis in both directions - both prognostic and diagnostic. This means that if the system is set to a failed state, the model can trace back to estimate which components are most likely responsible. Other scenarios can also be handled by moving beyond simple yes/no states, making it possible to represent partial failures or degraded performance rather than just total system failure. The BN can also handle more complex logical relationships than fault trees, including multi-dependent components. With additional design detail and operational data from the system, the BN becomes a significantly more attractive option for risk and reliability analysis.

5.2 Process and Approach Discussion

Developing the Functional Block Diagram (FBD) of the subsystems and component elements prior to conducting any analysis or modeling helped achieve a better understanding of the Second-Life BESS. By visualizing the system architecture early in the process, the FBD enabled a structured decomposition of functionality, interfaces, and dependencies across components, providing clarity on how system element interactions influence overall system behavior. As a result, the modeling process became more efficient and targeted, helping to prevent possible misinterpretations of failure pathways and enabling the identification of critical elements essential to system operation.

Although the information on degradation mechanisms is summarized in the background section, some of them remain difficult to interpret due to the highly interconnected nature of the processes. By integrating these mechanisms into an FMEA table, the study organized these complex processes into a structured format that links each failure mode to its primary affected component. Furthermore, this approach helped organize the data, improved the overall understanding of the table's structure, and served as an essential first step for the broader system-wide analysis using FMEA tables.

The use of FMEA in this work allowed for the identification of potential failure modes for a Second life Battery Energy Storage System, the assessment of the risks associated with these failure modes, the ranking of issues in terms of importance, and the identification of some of the detection methods within the system to address the most serious concerns.

Combining FMEA with tools like ROADS enables the identification and use of the probabilities that a component could fail via a specific mode and cause, facilitating the acquisition of some parameters required to calculate the Failure Mode Rate. Using the observed failure modes identified in this database produced a more robust set of failure modes than the subjective approaches commonly used in FMEA.

Implementing an MBSE model significantly enhanced the ability to capture, analyze, and manage system information in a structured and centralized way. One of the key benefits of this approach was that all relevant system knowledge, including supporting documentation, system structure, and safety and reliability analyses (FMEA and FTA) were integrated within a single digital environment. This reduced the need to rely on multiple external tools and improved traceability of different types of elements within the system. In addition, implementing an MBSE model facilitated a more automated and integrated reliability assessment approach, enabling more consistent analysis and better management of complex system interactions as the model evolved.

With the use of MSOSA Software calculated failure probabilities were automatically captured and saved, which reduces manual work and saves time. These results are stored in model-linked packages, allowing the probabilities for basic, intermediate, and top events to be preserved for

future use. This means the outputs are not just one-time results, but reusable data that can support the next stages of the project.

Using dependency matrices helped organize and visualize the connections between the system design and the safety analyses in a clear and compact way. Beyond just showing relationships these matrices serve as a critical validation layer allowing for the rapid identification of redundant traces or unnecessary data that remained after design changes. This is not directly seen in the results, but this made it possible to remove unused or disconnected elements and keep the model well-structured. By doing this, the matrices helped maintain consistency between the safety analysis and the system architecture.

The MSOSA software had the capability to export FMEA and FTA data into standard formats. This was important for the next stage of the project because it allowed the data to be used in external reliability tools and automated reporting processes. By moving information from the model into an external data workflow, the project was able to support more advanced analyses. This helped ensure that the safety data remained up to date and could continue evolving along with changes in the system design.

As explained in Section 3.3, the script automatically creates nodes for basic, intermediate, and top events and connects them based on the relationships defined in the input data that can be opened in GeNIe to visually display the generated network. While the script correctly builds the mathematical structure and relationships between events, it does not automatically arrange the network for visualization. Because of this, the network needs to be manually organized to achieve the clear hierarchical layout shown in Figure 4.11. In addition, it is required to manually define the specific connections that lead to the complete system failure "SV 360 Failure" top event so the network fully represents how faults propagate through the system.

5.3 Limitations

The main challenge of this analysis was the limited access to system data. Because the models were constructed using a small set of documents and without direct collaboration from the design

team, some assumptions had to be made about how the system works, effectively treating some subsystems as a “black box." This led to focusing mostly on individual components at the beginning of the analysis rather than focusing on the system as a whole. Comparative research on similar BESS systems was also used to help address some of the lack of design details, and certain assumptions had to be made such as the physical interconnections between components or failure mode effects. These assumptions made it possible to create a working SysML model and FTA framework. However, because the exact documentation was not available, the current models do not represent the exact system architecture and should not be considered an exact digital copy of the system.

The last identified limitation was the lack of information about internal safety and recovery features. Without knowing the details of the system’s diagnostics, such as the control logic that responds to faults or triggers automatic shutdowns, the FTA and BN models cannot fully capture the system’s true resilience. As a result, the failure probabilities reported here should be seen as worst-case scenarios, since they do not include the potential effects of these active safety measures.

5.4 Broader Impacts

As mentioned throughout this work, both standard FMEA and FTA analyses often fail to account for the inherent uncertainties in failure data, which can lead to misleading reliability estimates that do not meet the demands of modern, complex systems. By automating the translation of FTA diagrams into Bayesian Networks within an MBSE environment and incorporating failure information captured in FMEA, the gap between static design documentation and dynamic system behavior can be bridged.

This integration allows detailed component level insights from FMEA and FTA to inform the probabilistic structure in a more realistic way, enabling Bayesian updates that reflect uncertainties quickly and consistently as new information becomes available. As systems grow in complexity, manual translation becomes both a bottleneck and a source of error. An automated tool set ensures

that reliability analyses remain structurally aligned with current system operations, providing a scalable approach to maintaining accurate and up to date system reliability assessments over time.

Another broader impact of this work is its potential use in high-consequence domains such as nuclear energy. Previous studies have shown that Bayesian Networks are useful for modeling cause-and-effect relationships and estimating accident probabilities in nuclear safety analysis [84]. Having a tool that combines BNs with traditional reliability methods such as FMEA and FTA in an MBSE environment, contributes to a more comprehensive and robust framework for risk assessment in systems with very strict safety requirements. In these industries, certification and safety processes often depend on extensive documentation. MBSE models can also help modernize these processes by replacing large document sets with a centralized digital model, making it easier to manage data gaps and missing information. This work lays the foundation for developing more robust risk and reliability assessments within MBSE, which helps promote its use for critical applications.

Small organizations, which often lack the resources for extensive manual reliability assessments, can particularly benefit from this work. By providing automated and reusable analysis methods that are integrated with existing tools, this framework makes rigorous risk management more accessible. It also makes it easier to identify which parts of the system and which analyses need to be updated as requirements change, helping maintain consistency throughout the system life cycle and enabling system design improvements over time.

Finally, an important long-term impact of this research is that it creates a framework that can be expanded to support other reliability and safety analyses in the future. This work demonstrates a practical workflow that connects the MBSE environment to other modeling paradigms using Python-based tools and then brings the results back into the MBSE model. This approach can serve as a guide for integrating other analysis software as well, such as physics-based simulations and probabilistic models. The two-way exchange of information ensures that the system model remains the main and most reliable source of information, while external tools can still be used to run more

advanced analyses. As a result, this framework makes it easier to add other reliability methods in the future and supports a more connected and comprehensive engineering design process.

Chapter 6

Conclusions

This project involved a reliability analysis of a Second Life Battery Energy Storage System, combining various assessment tools. First, extensive research was conducted on lithium ion batteries to understand their operation and the degradation mechanisms that can affect these components. The information gathered was used to create the initial FMEA table that summarizes the main effects on battery components. These tables focused solely on the causes of each failure mode, as well as the resulting failure effects, without including any quantitative analysis. This approach helped summarize the available information and supported the initial use of the FMEA tool.

The next step was to create a Functional Block Diagram to identify the subsystems and main components of the Second Life BESS. This provided a clear understanding of the system's physical architecture, enabling the FMEA to proceed. With the system elements identified, the FMEA was performed using a catalog of failures from various components and integrating it into this well known methodology. Incorporating ROADS into the analysis facilitated the identification of failure modes and causes for each component, allowing their potential effects to be determined. Furthermore, the ROADS database simplified the estimation of several parameters required to calculate the failure mode rate, which represents the likelihood of a specific failure mode occurring within a system component.

Each failure mode was evaluated for its worst potential consequence, and a corresponding severity classification category was assigned to determine the overall risk level of failure for each component within the system. With this information, it was possible to identify and summarize the components that present the highest risks to the system, revealing that those with the highest risk are primarily responsible for ensuring proper operation and maintaining suitable environmental conditions.

Compared to FMEA developed in Excel, using FMEA tables integrated within an MBSE environment improves both data accuracy and reporting efficiency. Instead of managing safety in-

formation manually, the data was better organized within model-linked folders specific to each CF, FM, LFE, or FEF for every subsystem, which remain connected to the system design. This allows the FMEA tables to be updated automatically as the design evolves. As shown in the results from Section A, the large number of elements involved can make manual organization difficult and error-prone. Adapting this approach to an MBSE framework reduced this risk by keeping all information in a centralized and structured environment. As a result, reporting became more reliable and provided a clearer understanding of how individual failures affect the overall system.

To address the labor-intensive and time-consuming nature of traditional reliability methods, the study was transitioned into aa Model Based Systems Engineering environment. Using an MBSE software in this case MSOSA helped simplify the analysis by bringing different tools together in one place. The process started by defining the system architecture using the information from the FBD and once the architecture was established, the FMEA tables were created and linked directly to the different levels of the system within the model. This integrated approach made it much easier to develop the Fault Tree Analysis diagrams. Since the failure causes were already defined in the FMEA, these elements could be used to build the logical relationships that show how failures at the component level can lead to larger system level failures. By using the automated calculation features for FTA in MSOSA, failure probabilities were stored in model-linked packages saving the probability data for basic, intermediate, and top events and remain accessible for future reuse.

Since FMEA and FTA cannot fully measure uncertainty, Bayesian Networks were added to the study to improve the analysis. However, building Bayesian Networks by hand takes a significant amount of time and can be difficult to keep updated as the system model changes. To address this, an automated process was developed using GeNIe's Python wrapper and relationship tables from MSOSA. This process extracted the system connections directly from the MBSE model into an external data workflow that could support Bayesian probability analysis.

The Python scripts cleaned the exported data by removing unnecessary information and extracting probability values from the generated tables. After the data was prepared, the script created and displayed the Bayesian Network for the Second Life BESS. With only minor manual

adjustments, the resulting network provides a high-fidelity tool for evaluating system risks under uncertain conditions. This approach provides a first step toward better understanding how failures in specific components and external factors can spread through the system. It opens the door to a more detailed view of failure probabilities and their root causes.

6.1 Future Work

There are several next steps that can be taken to continue improving this work. The first relates to the implementation of FMEA within MBSE. If we compare the FMEA tables created in Excel with those generated in MSOSA, a key difference can be observed. While Excel allows for manual, multi-variable data entry, the tables currently produced in MSOSA primarily capture failure modes and severity levels at the component level. Numerical values such as specific probabilities (λ_p, α, λ) or other types of quantitative information cannot be integrated into MSOSA tables with the same flexibility as in a spreadsheet because the software treats these entries as model properties rather than simple text cells.

This limitation exists because MBSE prioritizes data traceability over manual entry. In a spreadsheet, a number is simply a static value. In MSOSA, however, a numerical value must be tied to a specific element attribute or to a requirement defined within the system architecture. future work should focus on developing custom stereotypes or tagged values within the model to store these type of parameters directly at the component level to address this limitation. This would allow quantitative reliability data to remain traceable while improving the usability of the FMEA tables and enabling a more seamless integration between qualitative failure descriptions and quantitative risk assessments.

This project focused on identifying failure modes only for Block elements. A valuable next step for future work would be to expand this analysis to include Part Properties, Requirements, Operations, and Activity Diagrams. Moving the FMEA from physical structure to system behavior helps provide a deeper understanding of where the system may be vulnerable. By linking failure modes to specific operations or activities, the model can show how failures might occur during real

system use. This helps ensure that safety goals are still met even when certain requirements are not fully satisfied. In the end, this detailed approach allows both Fault Trees and Bayesian Networks to be adjusted using data that reflects real operating conditions, helping transform the model from a static representation into a more practical predictive tool.

As illustrated in Section 4.2.3 the current diagrams utilize OR gates exclusively. While this logic provided a helpful starting point for learning how to use the modeling software and its capabilities, it provides a simplified view of how failures happen in the system. An important next step is to improve these FTAs by adding more advanced logic gates that better represent the complex relationships within a Second-Life BESS. By connecting these gates to real system operations, the model can show situations where failure only happens if both primary and backup safety layers fail at the same time. Adding this more realistic logic will improve the conditional probability tables used in the Bayesian Networks, allowing the FTA to move beyond a basic diagnostic tool and become a more accurate representation of how the BESS performs in real-world conditions.

Another important goal for future development is to fully automate the creation of Bayesian Networks from FTA diagrams in order to remove all manual work that is currently required. At the moment, the structure of the generated networks must be adjusted by hand to make them easy to read. By using PySmile's automatic layout features, the scripts could be improved to organize the nodes on their own and avoid overlaps. In addition, the model should move beyond using only standard Conditional Probability Tables. Adding different types of nodes, such as deterministic nodes for system logic or equation nodes for continuous variables, would allow the Bayesian Network to better represent how the BESS behaves in real conditions, making the MBSE to BN process easier to scale and apply to larger systems.

Finally, it is necessary to review all the results obtained with the system designers in order to receive feedback and ensure that the work being done is meaningful and aligned with the system's needs. In addition, making full use of the MBSE tool should remain an important focus for future work. Beyond the methods used in this study, the model can become much more powerful by connecting it with other software tools, since these platforms offer different capabilities. By linking

the system architecture to tools that simulate power electronics or monitor thermal behavior in real time, it becomes possible to run combined simulations. This approach would transform the model from a descriptive framework into a dynamic representation that supports system optimization and helps predict important data that comply with the needs for the Second-Life BESS.

Appendix A

Additional FMEA Figures

This section shows the FMEA tables including all components, failure effects, and severity levels. The information of this tables is derived from the FMEA results from the first part of this chapter.

#	Id	Subsystem	Item	Cause Of Failure	Failure Mode	Local Effect Of Failure	SEV
49	4.11	Communications Systems	Hydrogen Sensor	Failure to Operate Failure to detect Hydrogen	Component Functional Failure	Hydrogen not detected	1
50	4.21	Communications Systems	Smoke Sensor	Shorted Failure to Operate	Component Functional Failure	Smoke not detected	1
51	4.31	Communications Systems	Fire Panel	Loose	Component Mechanical Failure	Intermittent Operation Delayed Detections	2
52	4.32	Communications Systems	Fire Panel	Worn	Component Environmental Effects	Delayed Detections	2
53	4.33	Communications Systems	Fire Panel	Broken	Component Mechanical Failure	System Shutdown	2
54	4.41	Communications Systems	Wired Communications	Loose	Component Mechanical Failure	Intermittent Operation	3
55	4.42	Communications Systems	Wired Communications	Worn	Component Environmental Effects	Failure to Operate Degraded Output Disconnections	3
56	4.43	Communications Systems	Wired Communications	Broken	Component Mechanical Failure	Failure to Operate	3
57	4.51	Communications Systems	Antenna	Broken	Component Mechanical Failure	Loss of Server Connection	4
58	4.52	Communications Systems	Antenna	Loose	Component Mechanical Failure	Intermittent Operation	4
59	4.53	Communications Systems	Antenna	Delamination	Component Environmental Effects	Degraded Output Failure to Operate Disconnections	4
60	4.54	Communications Systems	Antenna	Worn	Component Environmental Effects	Intermittent Operation	4
61	4.55	Communications Systems	Antenna	Electrical Failure not Determined	Component Electrical Failure	Loss of Server Connection Loss of Electrical System Control	4
62	4.56	Communications Systems	Antenna	Leakage	Component Electrical Failure	Signal Degradation - Loss of Signal	4
63	4.57	Communications Systems	Antenna	Improper Output	Component Functional Failure	Signal Degradation - Loss of Signal	4
64	4.61	Communications Systems	Temperature Sensor	No Operation	Component Functional Failure	Operation Outside Specification Sensor fails to detect within limits	2
65	4.62	Communications Systems	Temperature Sensor	Functional Failure not Determined	Component Functional Failure	Sensor fails to detect within limits Operation Outside Specification	2
66	4.63	Communications Systems	Temperature Sensor	Shorted	Component Functional Failure	Physical Damage	2
67	4.64	Communications Systems	Temperature Sensor	Degraded Operation	Component Electrical Failure	Operation Outside Specification Sensor fails to detect within limits	2
68	4.71	Communications Systems	Humidity Sensor	Degraded Operation	Component Functional Failure	Not able to detect Humidity inside Failure to Operate	2
69	4.72	Communications Systems	Humidity Sensor	Improper Output	Component Functional Failure	Incorrect Measurements	3
70	4.73	Communications Systems	Humidity Sensor	No Operation	Component Functional Failure	Not able to detect Humidity inside	2
71	4.74	Communications Systems	Humidity Sensor	Functional Failure not Determined	Component Functional Failure	Failure to Operate Intermittent Operation Degraded Operation	2
72	4.75	Communications Systems	Humidity Sensor	Opened	Component Electrical Failure	Inoperation	3
73	4.76	Communications Systems	Humidity Sensor	Shorted	Component Electrical Failure	Short-Circuit	3
74	4.81	Communications Systems	Water Intrusion Sensor	Improper Output	Component Functional Failure	Incorrect Measurements	1
75	4.82	Communications Systems	Water Intrusion Sensor	Failure to Operate	Component Functional Failure	Unable to detect water inside the ca	1
76	4.83	Communications Systems	Water Intrusion Sensor	Intermittent Operation	Component Functional Failure	Degraded Operation	1
77	4.84	Communications Systems	Water Intrusion Sensor	Broken	Component Mechanical Failure	Unable to detect water inside the ca	2
78	4.85	Communications Systems	Water Intrusion Sensor	Electrical Failure not Determined	Component Electrical Failure	Unable to detect water inside the ca	1
79	4.86	Communications Systems	Water Intrusion Sensor	Unknown	Component Unknown	Unable to detect water inside the ca Degraded Operation Intermittent Operation	2
80	4.91	Communications Systems	HVIL Sensor	No Operation Failure to Operate	Component Functional Failure	System operates in unsafe conditio	2
81	4.92	Communications Systems	HVIL Sensor	Physical Damage	Component Mechanical Failure	System operates in unsafe conditio	3
82	4.93	Communications Systems	HVIL Sensor	Improper Output Degraded Operation	Component Functional Failure	System operates in unsafe conditio	3
83	4.94	Communications Systems	HVIL Sensor	Leakage	Component Electrical Failure	System operates in unsafe conditio	3
84	4.95	Communications Systems	HVIL Sensor	Unknown	Component Unknown	System operates in unsafe conditio	3

Figure A.1: SV 360 FMEA MBSE Table for Communications System

#	Id	Subsystem	Item	Cause Of Failure	Failure Mode	Local Effect Of Failure	SEV
99	6.11	Control / Management System	BIB	Failure to Operate	Component Functional Failure	System operates in unsafe condition	4
100	6.12	Control / Management System	BIB	Improper Output	Component Functional Failure	System operates in unsafe condition	3
101	6.21	Control / Management System	BMS Controller	Improper Output	Component Functional Failure	Fails to manage SOC charging/disc Fails to manage Temperature	2
102	6.22	Control / Management System	BMS Controller	Failure to Operate	Component Functional Failure	Fails to manage SOC charging/disc Fails to manage Temperature	3
103	6.23	Control / Management System	BMS Controller	Intermittent Operation	Component Functional Failure	Fails to manage SOC charging/disc Fails to manage Temperature	3
104	6.24	Control / Management System	BMS Controller	Binding	Component Mechanical Failure	System operates in unsafe condition	3
105	6.25	Control / Management System	BMS Controller	Cracked	Component Mechanical Failure	Potential exposure to unsafe condit	3
106	6.26	Control / Management System	BMS Controller	Electrical Failure not Determined	Component Electrical Failure	System operates in unsafe condition	2

Figure A.2: SV 360 FMEA MBSE Table for Control or Management System

#	Id	Subsystem	Item	Cause Of Failure	Failure Mode	Local Effect Of Failure	SEV
85	5.11	Cooling System	Ventilation Fan	Broken	Component Mechanical Failure	No ventilation in Electronics cabine Degraded Operation	4
86	5.12	Cooling System	Ventilation Fan	Noisy	Component Mechanical Failure	Noise Pollution	4
87	5.13	Cooling System	Ventilation Fan	Worn	Component Environmental Effects	Operation Outside Specification	4
88	5.14	Cooling System	Ventilation Fan	Binding	Component Mechanical Failure	Degraded Operation	4
89	5.15	Cooling System	Ventilation Fan	Failure to Operate	Component Functional Failure	Electronics cabinet overheating No ventilation in Electronics cabine	4
90	5.16	Cooling System	Ventilation Fan	No Operation	Component Functional Failure	No ventilation in Electronics cabine Electronics cabinet overheating	4
91	5.17	Cooling System	Ventilation Fan	Out of Specification	Component Functional Failure	Degraded Operation	4
92	5.18	Cooling System	Ventilation Fan	Workmanship	Component Procedure Failure	Failure to Operate	4
93	5.21	Cooling System	Liquid Chiller System	Cracked Ruptured	Component Mechanical Failure	Water intrusion into electronics cab	2
94	5.22	Cooling System	Liquid Chiller System	Leakage	Component Mechanical Failure	Operation Outside Specification	3
95	5.23	Cooling System	Liquid Chiller System	Failure to Operate	Component Functional Failure	Improper temperature regulation in	2
96	5.24	Cooling System	Liquid Chiller System	No Operation	Component Functional Failure	Temperature regulation off	3
97	5.25	Cooling System	Liquid Chiller System	Unknown	Component Unknown	Improper temperature regulation in	2
98	5.26	Cooling System	Liquid Chiller System	Corrosion	Component Environmental Effects	Coolant component faults	1

Figure A.3: SV 360 FMEA MBSE Table for Cooling System

#	Id	Subsystem	Item	Cause Of Failure	Failure Mode	Local Effect Of Failure	SEV
19	2.11	HV-DC System	DC Bus	Worn	Component Environmental Effects	Degraded Output Failure to Operate	3
20	2.21	HV-DC System	External DC Switch	Opened	Component Electrical Failure	Current stops flowing through circu	4
21	2.22	HV-DC System	External DC Switch	Out of Specification	Component Functional Failure	Degraded Output	4
22	2.23	HV-DC System	External DC Switch	Mechanical Failure not Determined	Component Mechanical Failure	Physical Damage	4
23	2.24	HV-DC System	External DC Switch	Shorted	Component Electrical Failure	Physical Damage	4
24	2.31	HV-DC System	DC Wiring	Mechanical Failure not Determined	Component Mechanical Failure	Physical Damage	4
25	2.32	HV-DC System	DC Wiring	Worn	Component Environmental Effects	Degraded Output Discontinuities Increase of Impedance	4
26	2.33	HV-DC System	DC Wiring	Vendor Defect	Component Manufacture Failure	Discontinuities Operation Outside Specification Physical Damage	4
27	2.34	HV-DC System	DC Wiring	Workmanship	Component Procedure Failure	Operation Outside Specification	4
28	2.35	HV-DC System	DC Wiring	Induced Failure	Component Procedure Failure	Operation Outside Specification Short-Circuit	4
29	2.36	HV-DC System	DC Wiring	Unknown	Component Unknown	Current stops flowing through circu Degraded Output Discontinuities Failure to Operate Loss of Power Loss of electrical connections Operation Outside Specification Physical Damage Short-Circuit	4
30	2.37	HV-DC System	DC Wiring	Shorted	Component Electrical Failure	Short-Circuit Component overheating	4
31	2.38	HV-DC System	DC Wiring	Opened	Component Electrical Failure	Current stops flowing through circu	4
32	2.39	HV-DC System	DC Wiring	Arching Sparking	Component Electrical Failure	Component overheating Operation Outside Specification Physical Damage Short-Circuit	3

Figure A.4: SV 360 FMEA MBSE Table for HV-DC System

#	Id	Subsystem	Item	Cause Of Failure	Failure Mode	Local Effect Of Failure	SEV
33	3.11	LV-AC System	Aux AC Power	Drift	Component Functional Failure	Miscalculation of the SOC	3
34	3.12	LV-AC System	Aux AC Power	No Operation	Component Functional Failure	Failure to Operate	3
35	3.13	LV-AC System	Aux AC Power	Shorted	Component Electrical Failure	Component overheating	4
36	3.14	LV-AC System	Aux AC Power	Vendor Defect	Component Manufacture Failure	Operation Outside Specification Voltage Fluctuation	4
37	3.15	LV-AC System	Aux AC Power	Unknown	Component Unknown	Voltage Fluctuation	4
38	3.21	LV-AC System	Main Breaker	No Operation Failure to Open	Component Functional Failure	Damage to chiller	2
39	3.22	LV-AC System	Main Breaker	Degraded Operation	Component Functional Failure	Power Flow Interruption	3
40	3.23	LV-AC System	Main Breaker	Opened	Component Electrical Failure	Power Flow Interruption	4
41	3.24	LV-AC System	Main Breaker	Opens without Command	Component Functional Failure	Unexpeted Activation Power Flow Interruption	4
42	3.25	LV-AC System	Main Breaker	Intermittent Operation	Component Functional Failure	Operation Outside Specification Failure to Operate	3
43	3.26	LV-AC System	Main Breaker	Mechanical Failure not Determined	Component Mechanical Failure	Loss of Electrical System Control	3
44	3.31	LV-AC System	AC Wiring	Worn Corrosion	Component Environmental Effects	Power Flow Interruption Degraded Output Failure to Operate	3
45	3.32	LV-AC System	AC Wiring	Improper Output	Component Functional Failure	Failure to Operate Degraded Output	3
46	3.33	LV-AC System	AC Wiring	Broken	Component Mechanical Failure	Failure to Operate Short-Circuit Open Circuit	4
47	3.34	LV-AC System	AC Wiring	Loose	Component Mechanical Failure	Intermittent Operation	4
48	3.35	LV-AC System	AC Wiring	Termination Failure	Component Procedure Failure	Open Circuit Degraded Output	4

Figure A.5: SV 360 FMEA MBSE Table for LV-AC System

Appendix B

Additional FTA Figures

This section illustrates the Fault Tree Diagrams for the rest of the subsystems in the Second Life BESS. The basic events are shown with their respective assigned failure probabilities, derived from the calculated values of the exponential distribution. As depicted in the figure, each of the intermediate and top events currently has a probability of zero. These are value properties that are updated once the model is executed and later stored in an instance.

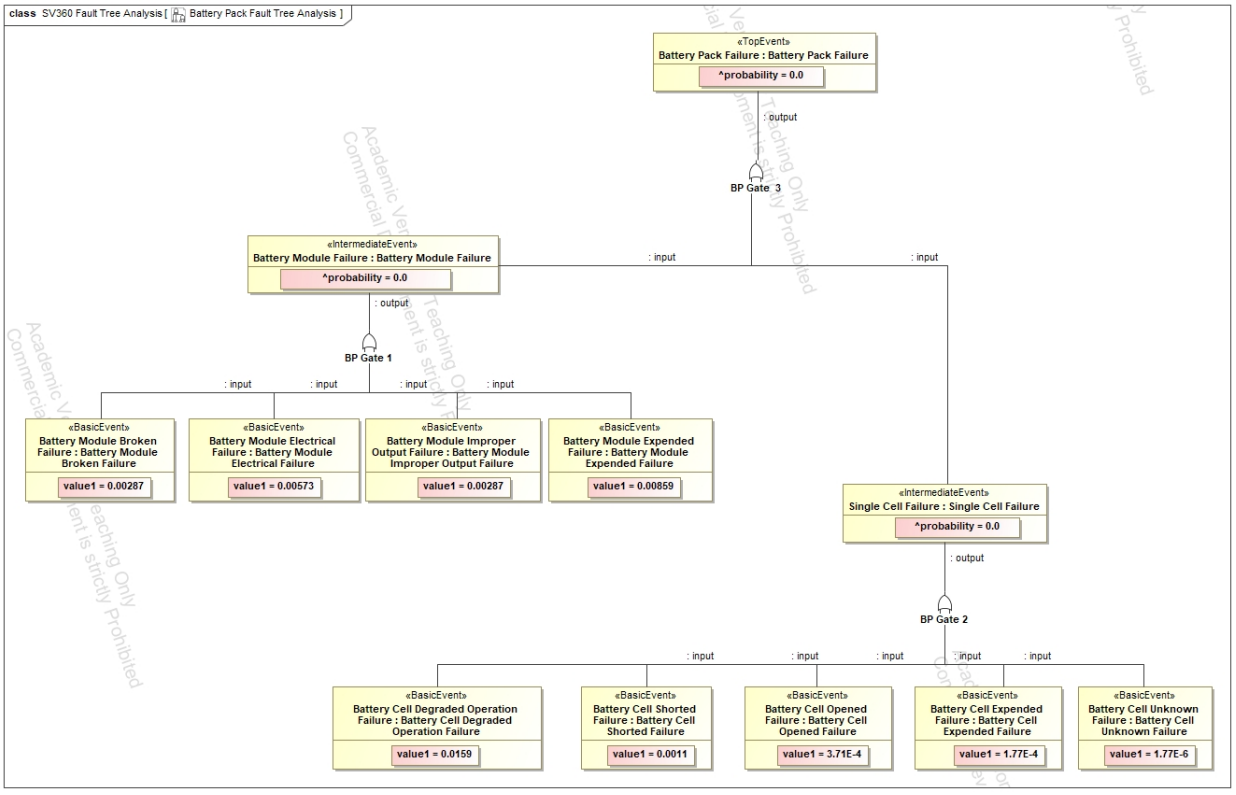


Figure B.1: SV 360 MBSE Battery Pack System FTA

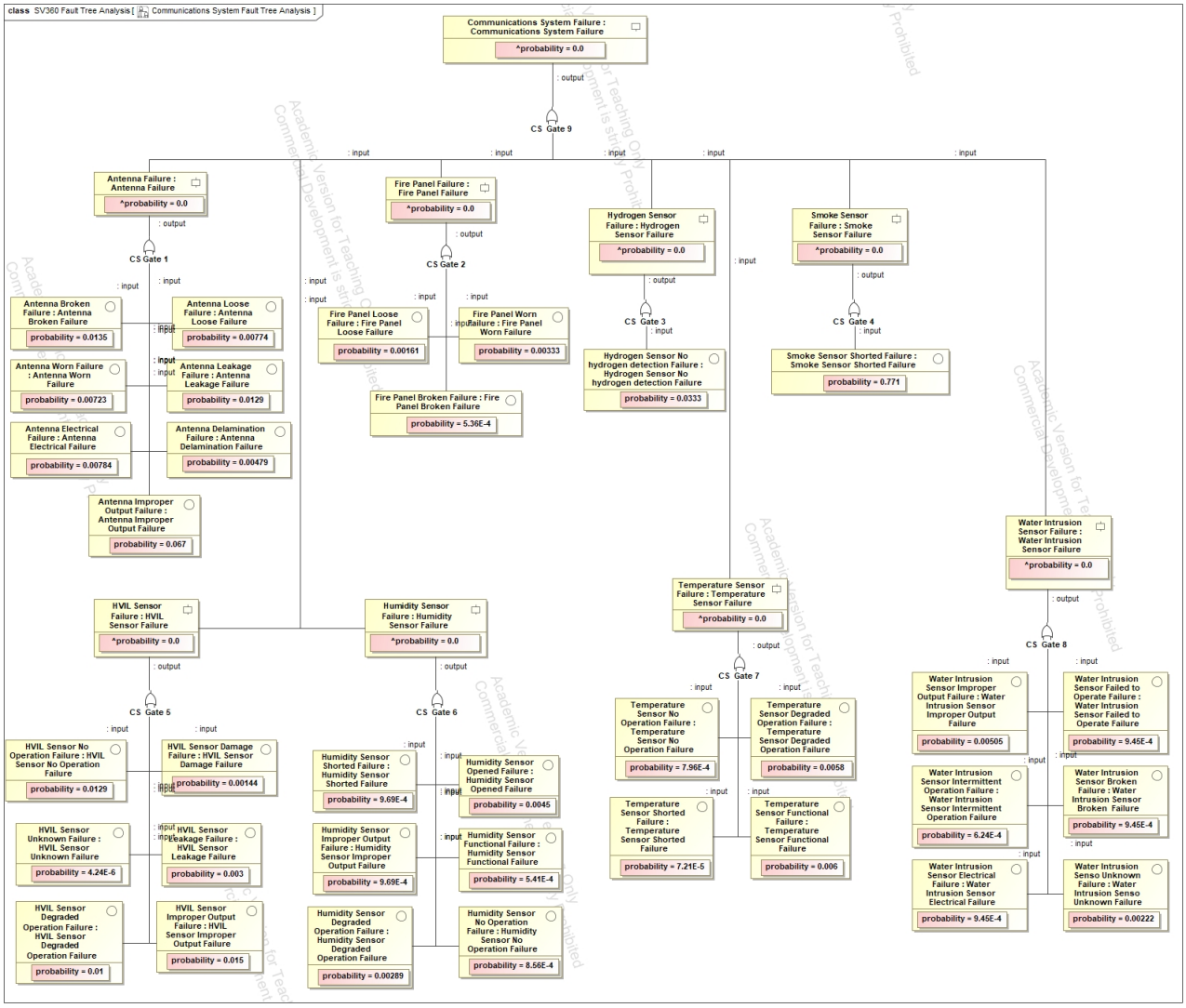


Figure B.2: SV 360 MBSE Communications System FTA

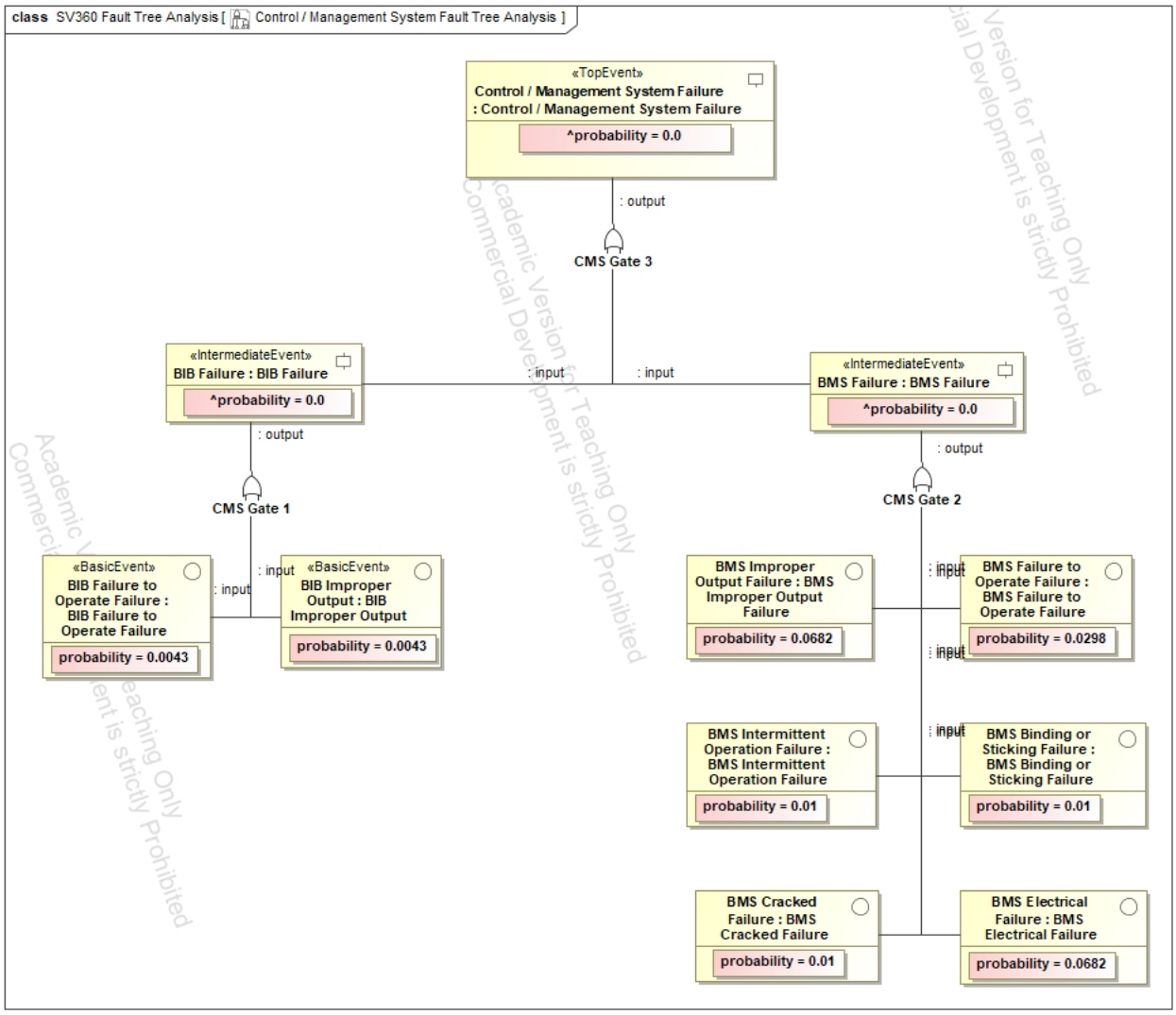


Figure B.3: SV 360 MBSE Control/Management System FTA

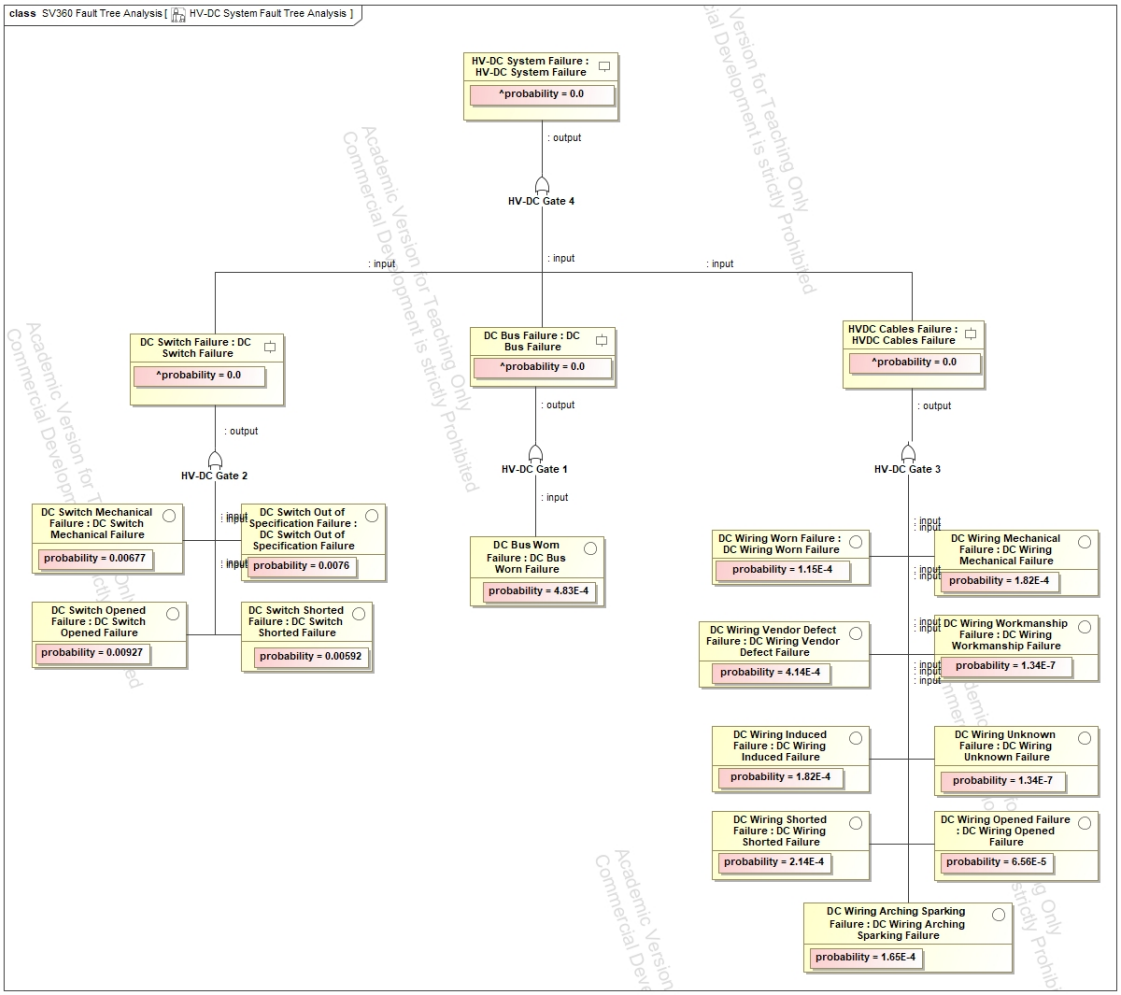


Figure B.4: SV 360 MBSE HV-DC System FTA

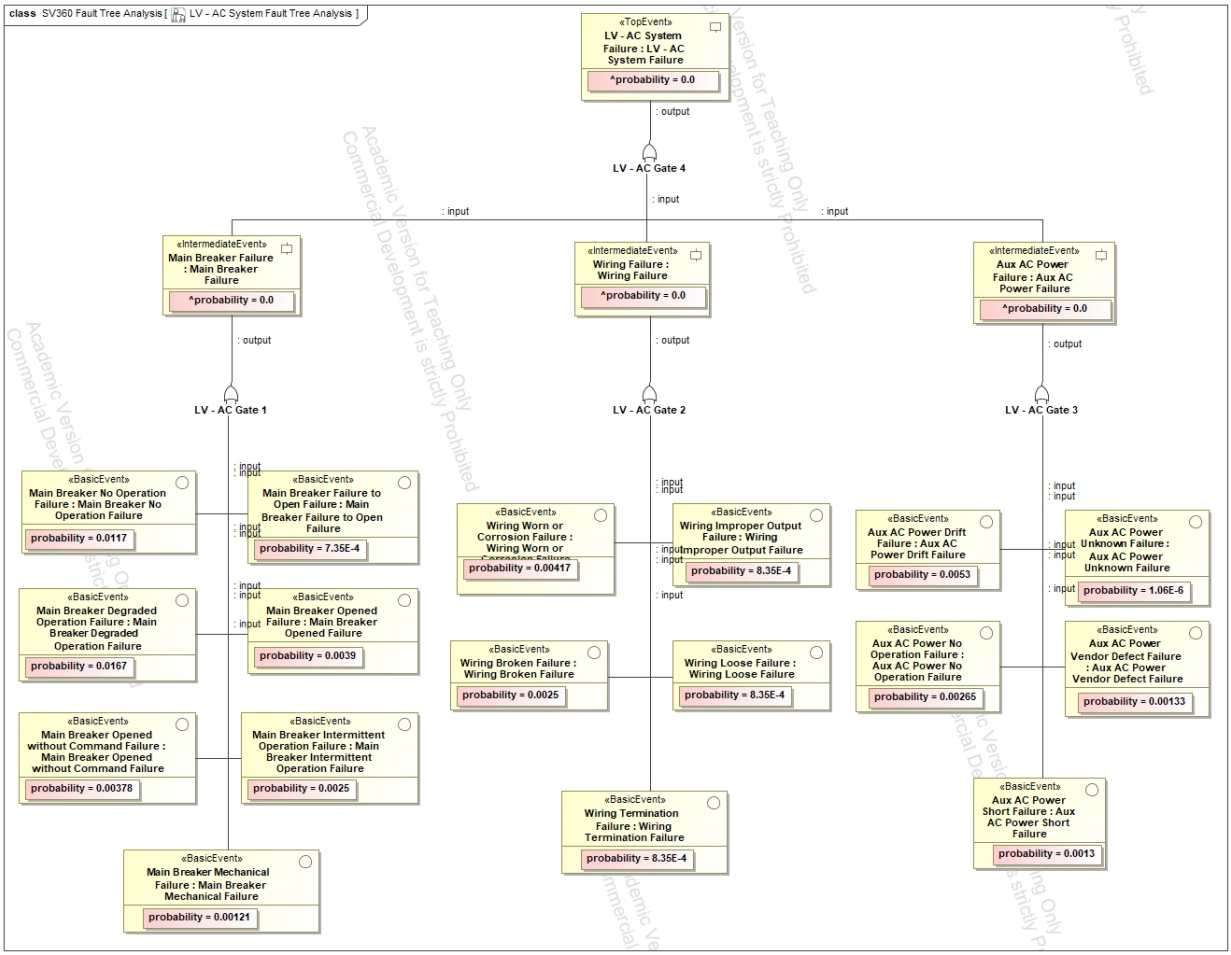


Figure B.5: SV 360 MBSE LV-AC System FTA

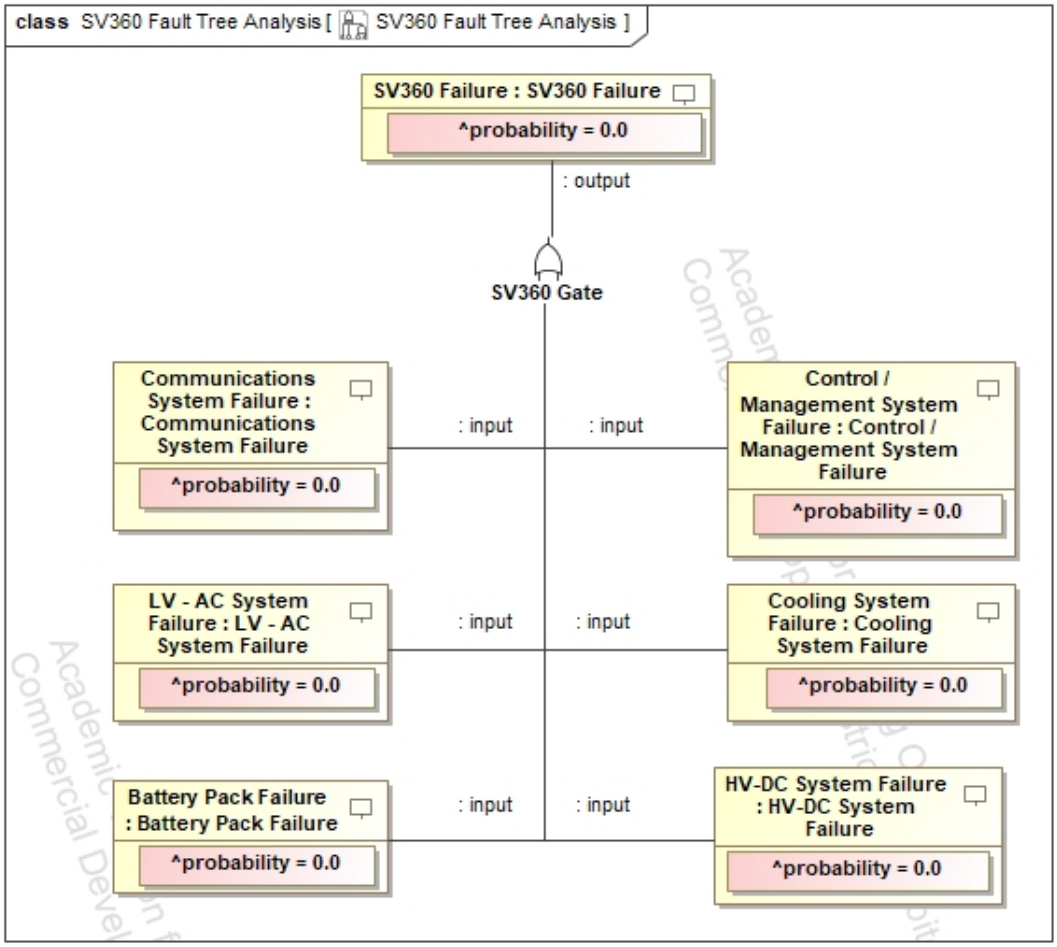


Figure B.6: SV 360 MBSE SV 360 System FTA

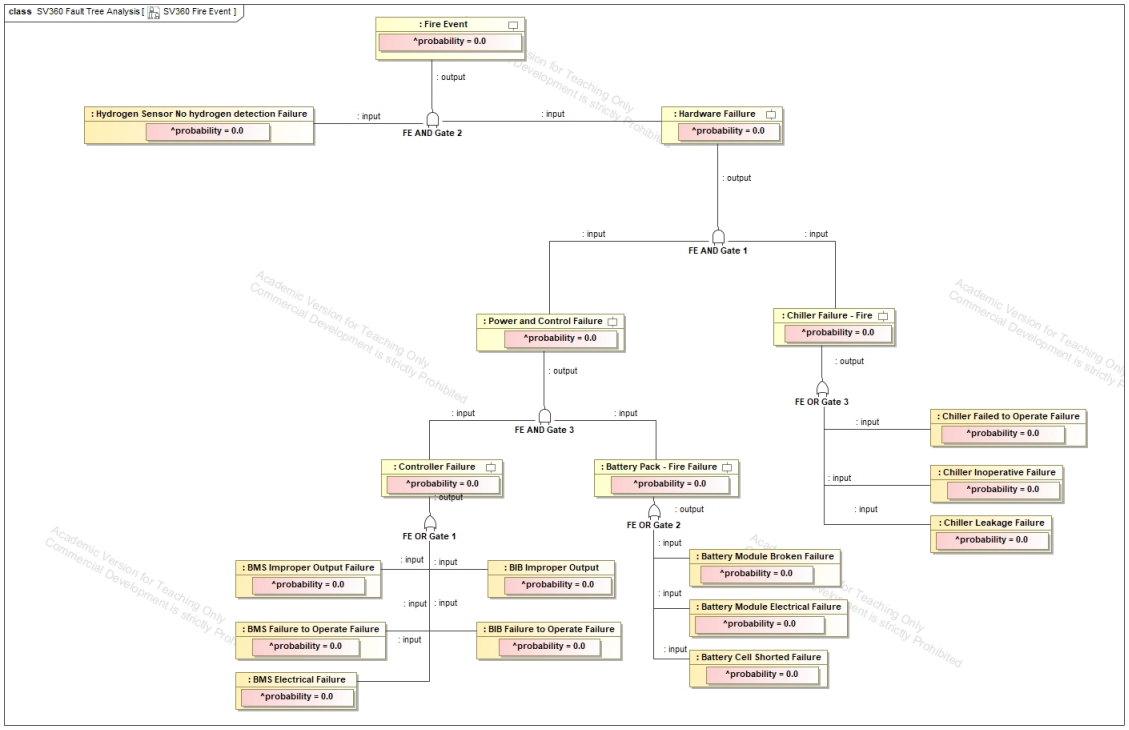


Figure B.7: SV 360 MBSE Fire Event FTA

Appendix C

Additional Relationship Tables

This section presents the traceability matrices and relationship tables that illustrate the connections between model elements. Figure C.1 provides an example of how a dependency matrix can be used to visualize the relationships between FMEA and FTA elements. Figure C.2 presents the resulting general table, which includes the events and their respective failure probabilities. Although the nature of these tables results in a condensed visual format here, the high-resolution output generated by the MSOSA software when saving images allows users to zoom in and clearly examine individual cells and event labels without any loss of clarity. Figure C.3 illustrates the dependency matrix that maps basic events to their corresponding logic gates. Figure C.4 provide the dependency matrix for intermediate and top events, respectively.

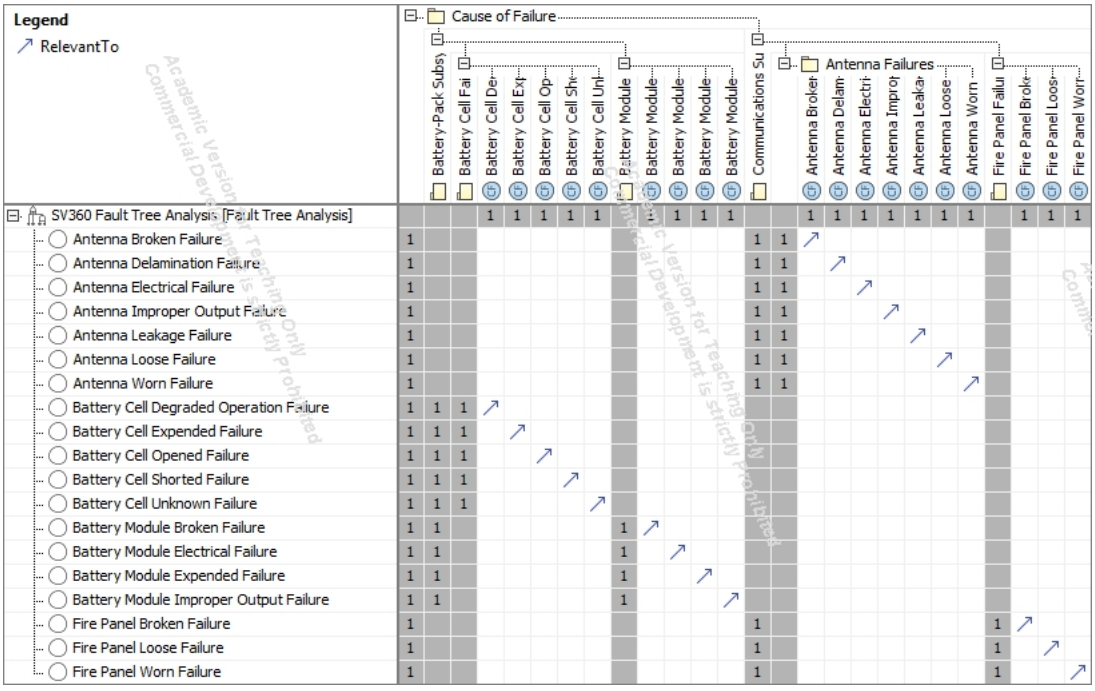


Figure C.1: Example of Relationship table between FMEA and FTA elements

Figure C.2: General Table with all FTA Events MBSE

Figure C.3: Dependency Matrix for Basic FTA Events

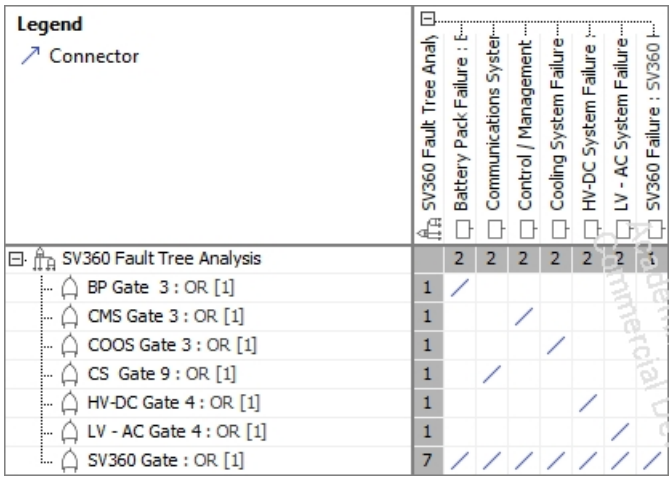


Figure C.4: Dependency Matrix for Top FTA Events

Appendix D

Scripts

A critical novelty of this work is the fusion of MBSE and Bayesian Networks, and within that the development of a python script that automatically creates Bayesian Networks from the MBSE models. This section presents the Python scripts used to automate the construction of Bayesian Networks from the MBSE models directly. Figure D.1 shows the script used to clean the dependency matrices and the general table by removing unnecessary information. Figures D.2, D.3, and D.4 show the different parts of the script that use the cleaned tables and the logic explained in Section 3.3 to automatically create the BN in the GeNIe software. Figure 4.11 shared the resultant BN once the elements were accommodated.

bdd [Package] BN Pictures [Data Cleaning Script]

```python
import pandas as pd
import os

########## CSV Data Clean ##########

########## Dependecy matrix Data Cleaning ##########
# Load CSV exported from dependency matrix
df1 = pd.read_csv(r"C:\Users\aylin\Desktop\Masters in Systems Engineering\Thesis topics\Smart Ville Project\Z - Thesis final\Tests\CSVs\Original Files\Dependecy Matrix Basic Event - Gate.csv", index_col=0)
df2 = pd.read_csv(r"C:\Users\aylin\Desktop\Masters in Systems Engineering\Thesis topics\Smart Ville Project\Z - Thesis final\Tests\CSVs\Original Files\Dependency Matrix Gate - Intermediate Event.csv", index_col=0)
df3 = pd.read_csv(r"C:\Users\aylin\Desktop\Masters in Systems Engineering\Thesis topics\Smart Ville Project\Z - Thesis final\Tests\CSVs\Original Files\Dependency Matrix Gate - Top Event.csv", index_col=0)


# Keep rows that contain <>
df1 = df1.reset_index()
df2 = df2.reset_index()
df3 = df3.reset_index()

suffix_remove = " : FTA Library::Gates::OR [1] [Reliability Analysis::Fault Tree Analysis::SV360 Fault Tree Analysis]"
# Clean Column Headers and replace spaces with underscores for df1
df1.columns = (df1.columns.str.replace(suffix_remove, "", regex=False).str.replace(" ", "_"))

# Clean Column Headers and replacing spaces with underscores for df2 and df3
for df in (df2, df3):
    df.columns = [col if col == 'index'
        else col.split(':')[0].strip()
        for col in df.columns
    ]

    df.columns = df.columns.str.replace(" ", "_")


# Clean Row Names
# df1
df1['index'] = (df1['index'].astype(str).str.split(':').str[0].str.strip().str.replace(" ", "_"))

for df in (df2, df3):
    if 'index' in df.columns:
        df['index'] = (df['index'].astype(str).str.replace(suffix_remove, "", regex=False).str.replace(" ", "_"))

# Create new cleaned csvs
df1.to_csv(r"C:\Users\aylin\Desktop\Masters in Systems Engineering\Thesis topics\Smart Ville Project\Z - Thesis final\Tests\CSVs\Cleaned DFs\Basic_Event_DM_Cleaned.csv", index=False)
df2.to_csv(r"C:\Users\aylin\Desktop\Masters in Systems Engineering\Thesis topics\Smart Ville Project\Z - Thesis final\Tests\CSVs\Cleaned DFs\Intermediatte_Event_DM_Cleaned.csv", index=False)
df3.to_csv(r"C:\Users\aylin\Desktop\Masters in Systems Engineering\Thesis topics\Smart Ville Project\Z - Thesis final\Tests\CSVs\Cleaned DFs\Top_Event_DM_Cleaned.csv", index=False)

########## Event Probability CSV Data Cleaning ##########
# Load CSV exported from General Table
df4 = pd.read_csv(r"C:\Users\aylin\Desktop\Masters in Systems Engineering\Thesis topics\Smart Ville Project\Z - Thesis final\Tests\CSVs\Original General Table\Reliability Analysis.csv", index_col=0)

# Clean Data and replace spaces for underscores
df4['Event Probability Value'] = (df4['Event Probability Value'].str.extract(r'=\s*([\d\.E+-]+)$')[0].astype(float))
df4['Classifier'] = (df4['Classifier'].str.replace(" ", "_"))

df4.to_csv(r"C:\Users\aylin\Desktop\Masters in Systems Engineering\Thesis topics\Smart Ville Project\Z - Thesis final\Tests\CSVs\Original General Table\Reliability_Analysis_Cleaned.csv", index=False)
```

Figure D.1: Script to Clean Tables from MBSE

bdd [Package] BN Pictures [FTA_to_BN_Script_Part_1]

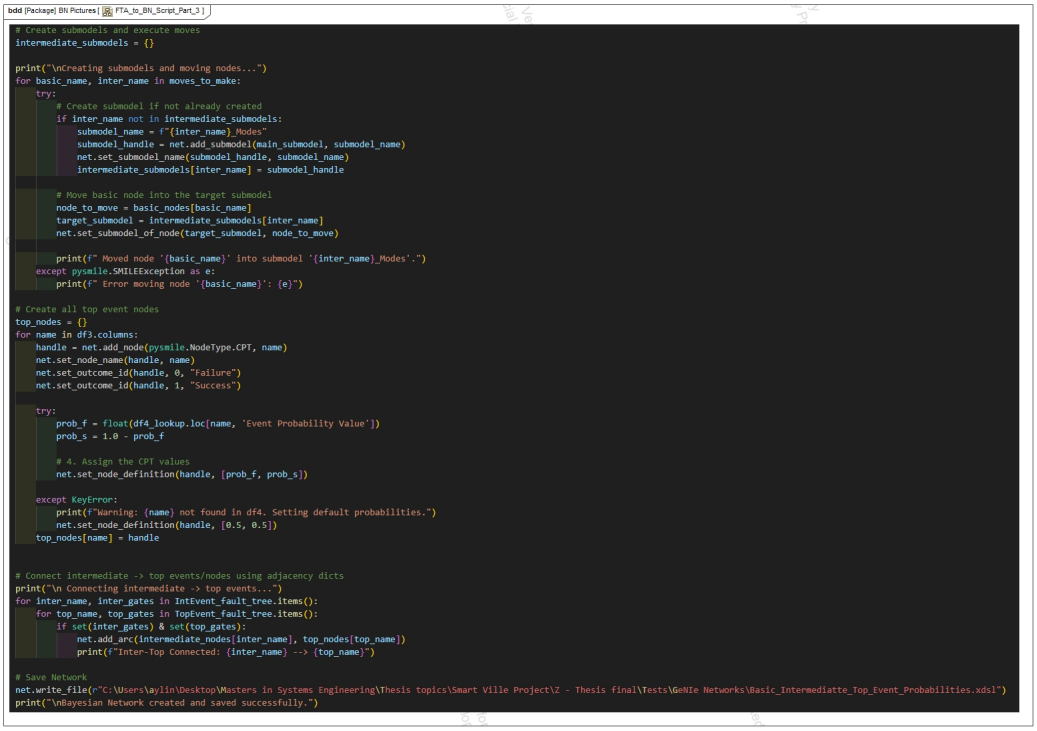
```
########## CSV Data Clean ##########

########## Dependecy matrix Data Cleaning ##########
# Load CSV exported from dependency matrix
df1 = pd.read_csv(r"C:\Users\aylin\Desktop\Masters in Systems Engineering\Thesis topics\Smart Ville Project\Z - Thesis final\Tests\CSVs\Cleaned Data\Basic_Event_DM_Cleaned.csv", index_col=0)
df2 = pd.read_csv(r"C:\Users\aylin\Desktop\Masters in Systems Engineering\Thesis topics\Smart Ville Project\Z - Thesis final\Tests\CSVs\Cleaned Data\Intermediatte_Event_DM_Cleaned.csv", index_col=0)
df3 = pd.read_csv(r"C:\Users\aylin\Desktop\Masters in Systems Engineering\Thesis topics\Smart Ville Project\Z - Thesis final\Tests\CSVs\Cleaned Data\Top_Event_DM_Cleaned.csv", index_col=0)
df4 = pd.read_csv(r"C:\Users\aylin\Desktop\Masters in Systems Engineering\Thesis topics\Smart Ville Project\Z - Thesis final\Tests\CSVs\Cleaned Data\Even_Probability.csv", index_col=0)

########## Event Probability CSV Data Cleaning ##########

# Build adjacency dictionary for Top Events
TopEvent_fault_tree = {}
for event in df3.columns: # loop over rows
    TopEvent_fault_tree[event] = [row for row in df3.index if df3.loc[row, event] == "<>"]

# Build adjacency dictionary for Intermediate Events
IntEvent_fault_tree = {}
for event in df2.columns:  # loop over rows
    IntEvent_fault_tree[event] = [row for row in df2.index if df2.loc[row, event] == "<>"]

# Build adjacency dictionary for Basic Events using Rows
BasicEvent_fault_tree = {}
for event in df1.index:  # loop over columns
    BasicEvent_fault_tree[event] = [col for col in df1.columns if df1.loc[event, col] == "<>"]


########## Create Bayesian Network ##########
net = pysmile.Network()

df4_lookup = df4

# Create all basic event nodes
basic_nodes = {}

for name in df1.index:
    handle = net.add_node(pysmile.NodeType.CPT, name) # CPT - Discrete chance node with conditional probability table
    net.set_node_name(handle, name)
    net.set_outcome_id(handle, 0, "Failure")
    net.set_outcome_id(handle, 1, "Success")

    try:
        prob_f = float(df4_lookup.loc[name, 'Event Probability Value'])
        prob_s = 1.0 - prob_f

        # 4. Assign the CPT values
        net.set_node_definition(handle, [prob_f, prob_s])

    except KeyError:
        print(f"Warning: {name} not found in df4. Setting default probabilities.")
        net.set_node_definition(handle, [0.5, 0.5])

    basic_nodes[name] = handle
```

Figure D.2: Script to convert FTA table into BN Part 1

bdd [Package] BN Pictures [FTA_to_BN_Script_Part_2]

```
# Create all intermediate event nodes
intermediate_nodes = {}
intermediate_submodels = {}

main_submodel = net.get_main_submodel()

for name in df2.columns:
    # Create intermediate event node
    handle = net.add_node(pysmile.NodeType.CPT, name)
    net.set_node_name(handle, name)
    net.set_outcome_id(handle, 0, "Failure")
    net.set_outcome_id(handle, 1, "Success")

    try:
        prob_f = float(df4_lookup.loc[name, 'Event Probability Value'])
        prob_s = 1.0 - prob_f

        # 4. Assign the CPT values
        net.set_node_definition(handle, [prob_f, prob_s])

    except KeyError:
        print(f"Warning: {name} not found in df4. Setting default probabilities.")
        net.set_node_definition(handle, [0.5, 0.5])

    intermediate_nodes[name] = handle

# Identify which basic events should move to which intermediate
print("\nIdentifying nodes to move into submodels...")
moves_to_make = []

for basic_name, basic_gates in BasicEvent_fault_tree.items():
    for inter_name, inter_gates in IntEvent_fault_tree.items():
        if set(basic_gates) & set(inter_gates):
            moves_to_make.append((basic_name, inter_name))
            break  # assign each basic event to only one intermediate


########################################################## This makes the program fail, need to verify the logic ###################################
# Add missing connections from basic to intermediate events
print("\nConnecting basic -> intermediate events...")
for basic_name, basic_gates in BasicEvent_fault_tree.items():
    for inter_name, inter_gates in IntEvent_fault_tree.items():
        # If they share a gate, they are connected
        if set(basic_gates) & set(inter_gates):
            net.add_arc(basic_nodes[basic_name], intermediate_nodes[inter_name])
            print(f"Basic-Inter Connected: {basic_name} --> {inter_name}")
```

Figure D.3: Script to convert FTA table into BN Part 2



```
# Create submodels and execute moves
intermediate_submodels = {}

print("\nCreating submodels and moving nodes...")
for basic_name, inter_name in moves_to_make:
    try:
        # Create submodel if not already created
        if inter_name not in intermediate_submodels:
            submodel_name = f"{inter_name}_Modes"
            submodel_handle = net.add_submodel(main_submodel, submodel_name)
            net.set_submodel_name(submodel_handle, submodel_name)
            intermediate_submodels[inter_name] = submodel_handle

        # Move basic node into the target submodel
        node_to_move = basic_nodes[basic_name]
        target_submodel = intermediate_submodels[inter_name]
        net.set_submodel_of_node(target_submodel, node_to_move)

        print(f" Moved node '{basic_name}' into submodel '{inter_name}_Modes'.")
    except pysmile.SMILEException as e:
        print(f" Error moving node '{basic_name}': {e}")

# Create all top event nodes
top_nodes = {}
for name in df3.columns:
    handle = net.add_node(pysmile.NodeType.CPT, name)
    net.set_node_name(handle, name)
    net.set_outcome_id(handle, 0, "Failure")
    net.set_outcome_id(handle, 1, "Success")

    try:
        prob_f = float(df4_lookup.loc[name, 'Event Probability Value'])
        prob_s = 1.0 - prob_f

        # 4. Assign the CPT values
        net.set_node_definition(handle, [prob_f, prob_s])

    except KeyError:
        print(f"Warning: {name} not found in df4. Setting default probabilities.")
        net.set_node_definition(handle, [0.5, 0.5])
    top_nodes[name] = handle


# Connect intermediate -> top events/nodes using adjacency dicts
print("\n Connecting intermediate -> top events...")
for inter_name, inter_gates in IntEvent_fault_tree.items():
    for top_name, top_gates in TopEvent_fault_tree.items():
        if set(inter_gates) & set(top_gates):
            net.add_arc(intermediate_nodes[inter_name], top_nodes[top_name])
            print(f"Inter-Top Connected: {inter_name} --> {top_name}")

# Save Network
net.write_file(r"C:\Users\aylin\Desktop\Masters in Systems Engineering\Thesis topics\Smart Ville Project\Z - Thesis final\Tests\GeNIe Networks\Basic_Intermediatte_Top_Event_Probabilities.xdsl")
print("\nBayesian Network created and saved successfully.")
```

Figure D.4: Script to convert FTA table into BN Part 3

Appendix E

Networks

This section presents a more detailed view of the networks generated in GeNIe software. Due to the large number of nodes in the Network Base, only selected examples are shown. Figure 4.12 illustrates the overall distribution of events within the SV-360 network using a bar chart representation. Figure E.1 shows the basic events within their corresponding submodels. Figure E.2 presents the CPT of a basic event node with the corresponding assigned failures and success probabilities. Figure E.3 displays the CPT of an intermediate event node of an intermediate event, showing how multiple parent failure modes contribute to the probability of the intermediate event. Finally, Figure E.4 shows the CPT of the top event, illustrating how higher-level failures combine to determine the probability of the Top event Failure.

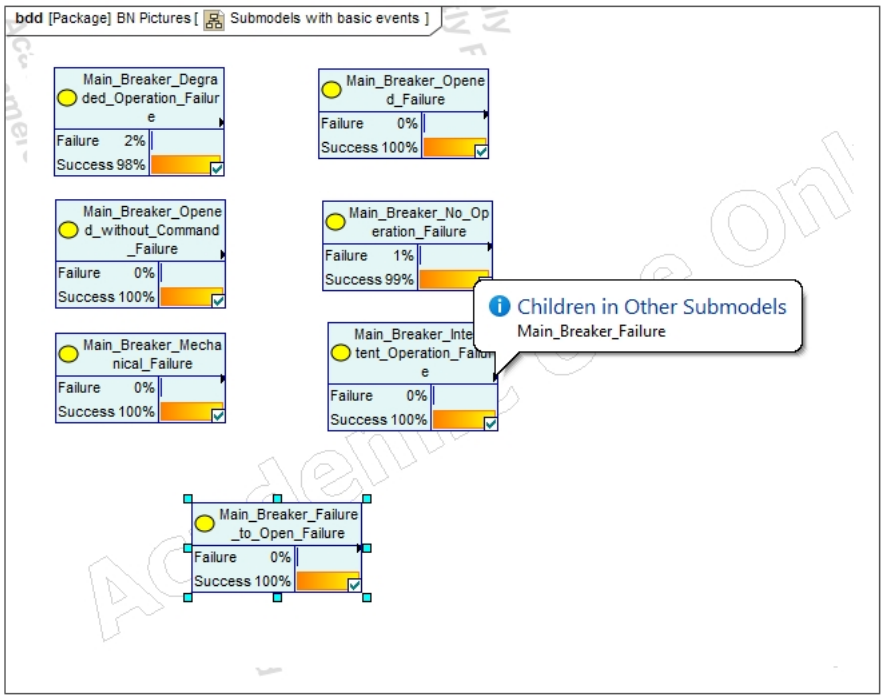


Figure E.1: Submodels with assigned basic events

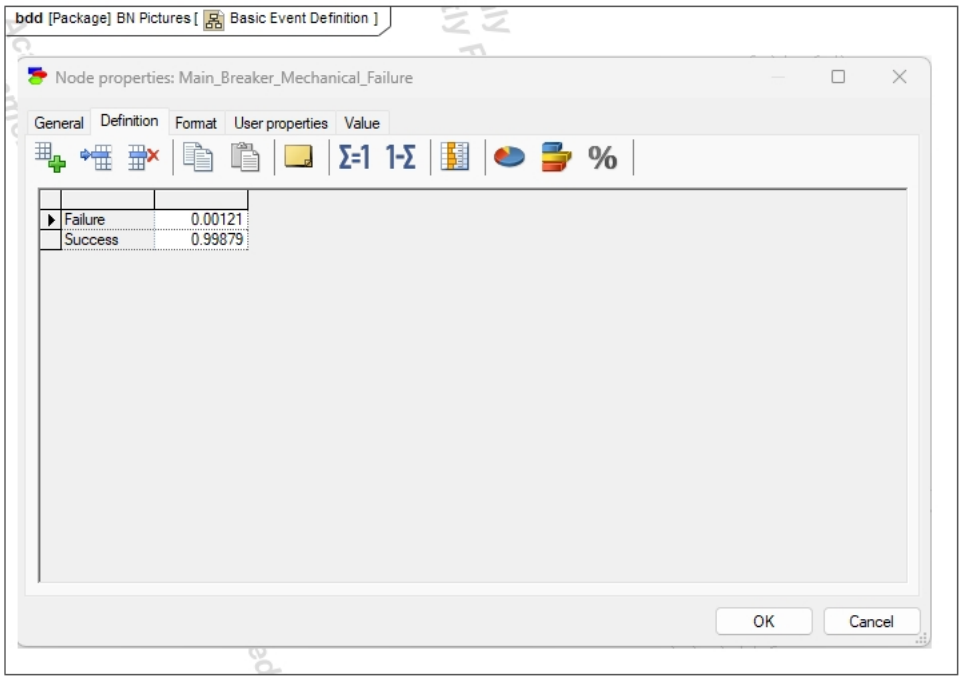


Figure E.2: GeNIe Basic Event Definition

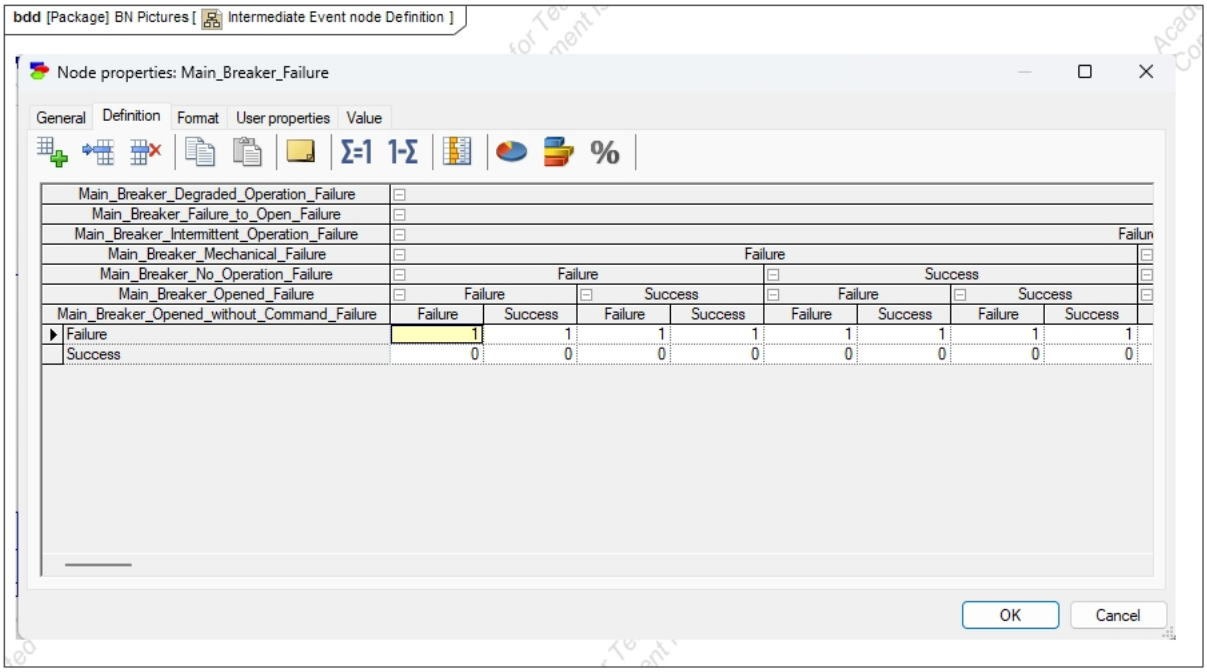


Figure E.3: GeNIe Intermediate Event Definition

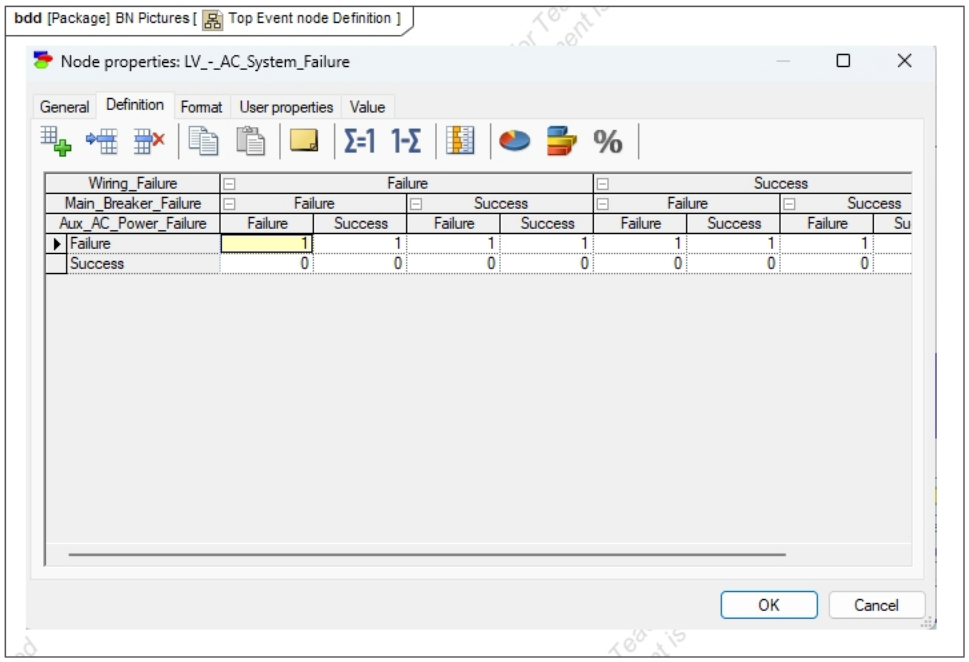


Figure E.4: GeNIe Top Event Definition

Bibliography

[1] B. Pepper, H. H. Arifin, S. Pavalkis, and K. Post, "Systematic risk analysis: Fmea and fta approaches for multi-level system architectures," in *INCOSE International Symposium*, vol. 35, pp. 573–594, Wiley Online Library, 2025.

[2] C. R. Birkl, M. R. Roberts, E. McTurk, P. G. Bruce, and D. A. Howey, "Degradation diagnostics for lithium ion cells," *Journal of Power Sources*, vol. 341, pp. 373–386, 2017.

[3] D. Herber, "Syse 567: Systems engineering architecture session 1," tech. rep., Colorado State University, 2024.

[4] S. Friedenthal, A. Moore, and R. Steiner, *A practical guide to SysML*. Morgan Kaufmann, 3rd edition ed., 2015.

[5] I. E. Agency, "World energy outlook 2025," tech. rep., IEA, 2025. Licence: CC BY 4.0 (report); CC BY-NC-SA 4.0 (Annex A).

[6] U. E. I. Administration, "Electricity data browser - net generation for all sectors," tech. rep., U.S. Energy Information Administration, 2024.

[7] G. Sachs, "Ai is poised to drive 160

[8] C. R. Service, "Variable renewable energy: An introduction," tech. rep., Congressional Research Service, 2025.

[9] N. Guru, S. Patnaik, M. R. Nayak, and A. K. Barisal, "Renewable energy sources and battery storage integrated microgrid energy management for customer benefit with reduced emission," in *2024 IEEE International Conference on Smart Power Control and Renewable Energy (ICSPCRE)*, pp. 1–6, IEEE, 2024.

[10] M. Shahjalal, P. K. Roy, T. Shams, A. Fly, J. I. Chowdhury, M. R. Ahmed, and K. Liu, "A review on second-life of li-ion batteries: Prospects, challenges, and issues," *Energy*, vol. 241, p. 122881, 2022.

[11] M. Etxandi-Santolaya, L. C. Casals, and C. Corchero, "Extending the electric vehicle battery first life: Performance beyond the current end of life threshold," *Heliyon*, vol. 10, no. 4, p. e26066, 2024.

[12] X. Hu, X. Deng, F. Wang, Z. Deng, X. Lin, R. Teodorescu, and M. G. Pecht, "A review of second-life lithium-ion batteries for stationary energy storage applications," *Proceedings of the IEEE*, vol. 110, no. 6, pp. 735–753, 2022.

[13] A. Gabryelczyk, S. Ivanov, A. Bund, and G. Lota, "Corrosion of aluminium current collector in lithium-ion batteries: A review," *Journal of Energy Storage*, vol. 43, p. 103226, 2021.

[14] J. Conzen, S. Lakshmipathy, A. Kapahi, S. Kraft, and M. DiDomizio, "Lithium ion battery energy storage systems (bess) hazards," *Journal of Loss Prevention in the Process Industries*, vol. 81, p. 104932, 2023.

[15] R. Aalund and V. Paglioni, "Systems Engineering Approach to Design for Reliability (DfR)," in *Proceedings of ESREL SRA-E 2025*, (Stavanger, Norway), June 2025.

[16] M. Modarres and K. Groth, *Reliability and risk analysis*. CRC Press, 2023.

[17] R. Aalund and V. P. Paglioni, "Enhancing reliability in embedded systems hardware: A literature survey," *IEEE Access*, 2025.

[18] K. D. Sharma and S. Srivastava, "Failure mode and effect analysis (fmea) implementation: a literature review," *Journal of Advance Research in Aeronautics and Space Science*, vol. 5, no. 1, pp. 1–17, 2018.

[19] B. Göksu, C. Şakar, and O. Yüksel, "A probabilistic assessment of ship blackout incident with fault tree analysis into (fta) bayesian network (bn)," *Journal of Marine Engineering & Technology*, vol. 24, no. 1, pp. 54–69, 2025.

[20] X. Han and J. Zhang, "A combined analysis method of fmea and fta for improving the safety analysis quality of safety-critical software," in *2013 IEEE International Conference on Granular Computing (GrC)*, pp. 353–356, IEEE, 2013.

[21] A. Ruiz-Tagle, A. D. Lewis, C. A. Schell, E. Lever, and K. M. Groth, "Bantera: a bayesian network for third-party excavation risk assessment," *Reliability Engineering & System Safety*, vol. 223, p. 108507, 2022.

[22] S. Rastayesh, S. Bahrebar, F. Blaabjerg, D. Zhou, H. Wang, and J. Dalsgaard Sørensen, "A system engineering approach using fmea and bayesian network for risk analysis—a case study," *Sustainability*, vol. 12, no. 1, p. 77, 2019.

[23] A. H. de Andrade Melani and G. F. M. de Souza, "Obtaining fault trees through sysml diagrams: A mbse approach for reliability analysis," in *2020 Annual reliability and maintainability symposium (RAMS)*, pp. 1–5, IEEE, 2020.

[24] G. Biggs, T. Juknevicius, A. Armonas, and K. Post, "Integrating safety and reliability analysis into mbse: overview of the new proposed omg standard," in *INCOSE International Symposium*, vol. 28, pp. 1322–1336, Wiley Online Library, 2018.

[25] J. F. Da Mata, R. O. Neto, A. Z. Mesquita, *et al.*, "Comparison of the performance, advantages and disadvantages of nuclear power generation compared to other clean sources of electricity," tech. rep., Associação Brasileira de Energia Nuclear (ABEN), Rio de Janeiro, RJ (Brazil), 2017.

[26] D. Anggraini, "Reliability and cost-benefit analysis of the battery energy storage system," 2023.

[27] A. Bakeer, A. Chub, Y. Shen, and A. Sangwongwanich, "Reliability analysis of battery energy storage system for various stationary applications," *Journal of Energy Storage*, vol. 50, p. 104217, 2022.

[28] T. Holleran, "The best of the bess: The role of battery energy storage systems in grid reliability," 2025. Accessed: October 28, 2025.

[29] E. H. Y. Moa and Y. I. Go, "Large-scale energy storage system: safety and risk assessment," *Sustainable Energy Research*, vol. 10, no. 1, p. 13, 2023.

[30] K. S. Boparai and R. Singh, "Electrochemical energy storage using batteries, superconductors and hybrid technologies," *Elsevier*, 2020.

[31] J. M. Foster, X. Huang, M. Jiang, S. J. Chapman, B. Protas, and G. Richardson, "Causes of binder damage in porous battery electrodes and strategies to prevent it," *Journal of Power Sources*, vol. 350, pp. 140–151, 2017.

[32] S. Berg, "Battery failure analysis and characterization of failure types," *Process Safety Progress*, vol. 41, no. 3, pp. 419–422, 2022.

[33] M. Rasheed, M. Kamel, H. Wang, R. Zane, and K. Smith, "Investigation of active life balancing to recondition li-ion battery packs for 2 nd life," in *2020 IEEE 21st Workshop on Control and Modeling for Power Electronics (COMPEL)*, pp. 1–7, IEEE, 2020.

[34] Y. Zheng, X. Han, L. Lu, J. Li, and M. Ouyang, "Lithium ion battery pack power fade fault identification based on shannon entropy in electric vehicles," *Journal of Power Sources*, vol. 223, pp. 136–146, 2013.

[35] J. R. Belt, C. D. Ho, T. J. Miller, M. A. Habib, and T. Q. Duong, "The effect of temperature on capacity and power in cycled lithium ion batteries," *Journal of power sources*, vol. 142, no. 1-2, pp. 354–360, 2005.

[36] L. Alzate-Vargas, S. M. Blau, E. W. C. Spotte-Smith, S. Allu, K. A. Persson, and J.-L. Fattebert, "Insight into sei growth in li-ion batteries using molecular dynamics and accelerated chemical reactions," *The Journal of Physical Chemistry C*, vol. 125, no. 34, pp. 18588–18596, 2021.

[37] C. L. Campion, W. Li, and B. L. Lucht, "Thermal decomposition of lipf6-based electrolytes for lithium-ion batteries," *Journal of The Electrochemical Society*, vol. 152, no. 12, p. A2327, 2005.

[38] B. L. Rinkel, J. P. Vivek, N. Garcia-Araez, and C. P. Grey, "Two electrolyte decomposition pathways at nickel-rich cathode surfaces in lithium-ion batteries," *Energy & environmental science*, vol. 15, no. 8, pp. 3416–3438, 2022.

[39] Y. Zhou, J. Huang, and B. Li, "Cation order and disorder in cathode materials for li-ion batteries," *Next Materials*, vol. 6, p. 100441, 2025.

[40] S. E. O'Kane, W. Ai, G. Madabattula, D. Alonso-Alvarez, R. Timms, V. Sulzer, J. S. Edge, B. Wu, G. J. Offer, and M. Marinescu, "Lithium-ion battery degradation: how to model it," *Physical Chemistry Chemical Physics*, vol. 24, no. 13, pp. 7909–7922, 2022.

[41] Q. Liu, C. Du, B. Shen, P. Zuo, X. Cheng, Y. Ma, G. Yin, and Y. Gao, "Understanding undesirable anode lithium plating issues in lithium-ion batteries," *RSC advances*, vol. 6, no. 91, pp. 88683–88700, 2016.

[42] J. Fan and S. Tan, "Studies on charging lithium-ion cells at low temperatures," *Journal of The Electrochemical Society*, vol. 153, no. 6, p. A1081, 2006.

[43] K. Cho, J. Baek, C. Balamurugan, H. Im, and H.-J. Kim, "Corrosion study of nickel-coated copper and chromate-coated aluminum for corrosion-resistant lithium-ion battery lead-tab," *Journal of Industrial and Engineering Chemistry*, vol. 106, pp. 537–545, 2022.

[44] X. Shi, H. Zhang, Y. Zhang, J. Liu, J. Zhang, and L. Li, "Corrosion and protection of aluminum current collector in lithium-ion batteries," *The Innovation Materials*, vol. 1, no. 2, pp. 100030–1, 2023.

[45] A. S. Mussa, G. Lindbergh, M. Klett, P. Gudmundson, P. Svens, and R. W. Lindström, "Inhomogeneous active layer contact loss in a cycled prismatic lithium-ion cell caused by the jelly-roll curvature," *Journal of Energy Storage*, vol. 20, pp. 213–217, 2018.

[46] T. N. L. Doan and I. Taniguchi, "Cathode performance of limnpo4/c nanocomposites prepared by a combination of spray pyrolysis and wet ball-milling followed by heat treatment," *Journal of Power Sources*, vol. 196, no. 3, pp. 1399–1408, 2011.

[47] M. Ecker, N. Nieto, S. Käbitz, J. Schmalstieg, H. Blanke, A. Warnecke, and D. U. Sauer, "Calendar and cycle life study of li (nimnco) o2-based 18650 lithium-ion batteries," *Journal of Power Sources*, vol. 248, pp. 839–851, 2014.

[48] Y. Zhang, C. Zhao, and Z. Guo, "Simulation of crack behavior of secondary particles in li-ion battery electrodes during lithiation/de-lithiation cycles," *International Journal of Mechanical Sciences*, vol. 155, pp. 178–186, 2019.

[49] N. P. Pieczonka, Z. Liu, P. Lu, K. L. Olson, J. Moote, B. R. Powell, and J.-H. Kim, "Understanding transition-metal dissolution behavior in lini0. 5mn1. 5o4 high-voltage spinel for lithium ion batteries," *The Journal of Physical Chemistry C*, vol. 117, no. 31, pp. 15947–15957, 2013.

[50] R. Stockhausen, L. Gehrlein, M. Müller, T. Bergfeldt, A. Hofmann, F. J. Müller, J. Maibach, H. Ehrenberg, and A. Smith, "Investigating the dominant decomposition mechanisms in lithium-ion battery cells responsible for capacity loss in different stages of electrochemical aging," *Journal of Power Sources*, vol. 543, p. 231842, 2022.

[51] Atomfair, "Binder degradation and electrode disintegration," 2025. Accessed November 10, 2025.

[52] D. Goers, M. E. Spahr, A. Leone, W. Märkle, and P. Novák, "The influence of the local current density on the electrochemical exfoliation of graphite in lithium-ion battery negative electrodes," *Electrochimica Acta*, vol. 56, no. 11, pp. 3799–3808, 2011.

[53] K. Chirumalla, I. Kulkov, F. Vu, and M. Rahic, "Second life use of li-ion batteries in the heavy-duty vehicle industry: Feasibilities of remanufacturing, repurposing, and reusing approaches," *Sustainable Production and consumption*, vol. 42, pp. 351–366, 2023.

[54] G. Lacey, G. Putrus, and A. Salim, "The use of second life electric vehicle batteries for grid support," in *Eurocon 2013*, pp. 1255–1261, IEEE, 2013.

[55] T. Steckel, A. Kendall, and H. Ambrose, "Applying levelized cost of storage methodology to utility-scale second-life lithium-ion battery energy storage systems," *Applied Energy*, vol. 300, p. 117309, 2021.

[56] M. T. Lawder, B. Suthar, P. W. Northrop, S. De, C. M. Hoff, O. Leitermann, M. L. Crow, S. Santhanagopalan, and V. R. Subramanian, "Battery energy storage system (bess) and battery management system (bms) for grid-scale applications," *Proceedings of the IEEE*, vol. 102, no. 6, pp. 1014–1030, 2014.

[57] Super B Lithium Power B.V., *User Manual: Battery Interface Box (BIB)*. Super B Lithium Power B.V., Europalaan 202, 7559 SC Hengelo (Ov), The Netherlands, version 1.2 ed., 2023.

[58] C. Multiphysics, *Liquid-Cooled Battery Energy Storage System*. COMSOL Application Gallery, 2024.

[59] B. Kreeley and S. Coulton, "Enclosure cooling keeps battery energy storage systems going & going," whitepaper, Kooltronic, Inc., 2022.

[60] A. A. Sensors, "Battery energy storage systems safety: Protecting power," feature article, Amphenol Corporation, 2025.

[61] F. Global, "Why this technology is a 'big deal' for data centers, battery energy storage systems and more," feature article, FM Global, 2025.

[62] R. Power, "Ac vs dc-coupled bess: the pros and cons," blog article, Rated Power, 2023.

[63] R. Kumar, "Dc-ac power electronics converters for battery energy storage," technical article, EEPower, 2023.

[64] M. Stecca, L. R. Elizondo, T. B. Soeiro, P. Bauer, and P. Palensky, "A comprehensive review of the integration of battery energy storage systems into distribution networks," *IEEE Open Journal of the Industrial Electronics Society*, vol. 1, pp. 46–65, 2020.

[65] S. Kabir, "An overview of fault tree analysis and its application in model based dependability analysis," *Expert Systems with Applications*, vol. 77, pp. 114–135, 2017.

[66] P. Renosori, H. Oemar, and S. R. Fauziah, "Combination of fta and fmea methods to improve efficiency in the manufacturing company.," *Acta Logistica (AL)*, vol. 10, no. 3, 2023.

[67] J. M. Borky and T. H. Bradley, *Effective model-based systems engineering*. Springer, 2018.

[68] INCOSE, *INCOSE systems engineering handbook*. John Wiley & Sons, 2023.

[69] J. L. Fernandez and C. Hernandez, *Practical model-based systems engineering*. Artech house, 2019.

[70] J. S. Topper and N. C. Horner, "Model-based systems engineering in support of complex systems development," *Johns Hopkins APL technical digest*, vol. 32, no. 1, pp. 419–432, 2013.

[71] N. J. Lindsey, M. Alimardani, and L. D. Gallo, "Reliability analysis of complex nasa systems with model-based engineering," in *2020 Annual reliability and maintainability symposium (RAMS)*, pp. 1–8, IEEE, 2020.

[72] O. M. Group, *RAAML — Risk Analysis and Assessment Modeling Language*, 2023.

[73] J. Hummell, "How to model your failure mode and effects analysis (fmea) with sysml," tech. rep., MBSE Solutions, 2016.

[74] L. Dai and B. Kantarci, "Advancing autonomous vehicle safety: A combined fault tree analysis and bayesian network approach," in *2025 IEEE Engineering Reliable Autonomous Systems (ERAS)*, pp. 1–7, IEEE, 2025.

[75] D. Ma, Z. Zhou, Y. Jiang, and W. Ding, "Constructing bayesian network by integrating fmea with fta," in *2014 Fourth International Conference on Instrumentation and Measurement, Computer, Communication and Control*, pp. 696–700, 2014.

[76] N. Fenton and M. Neil, *Risk Assessment and Decision Analysis with Bayesian Networks*. CRC Press, Taylor & Francis Group, 2nd ed., 2018.

[77] C. S. Kulkarni, M. Corbetta, and E. I. Robinson, "Systems health monitoring: Integrating fmea into bayesian networks," in *42nd International IEEE Aerospace Conference*, no. 20205006373, 2021.

[78] L. BayesFusion, *GeNIe Modeler User's Manual*. BayesFusion, LLC, version 5.0r2 ed., 2024.

[79] S. ARP5580, "Recommended failure modes and effects analysis (fmea) practices for non-automobile applications," *Warrendale: Society of Automotive Engineers*, 2020.

[80] W. D. Schindel, "Failure analysis: Insights from model-based systems engineering," *INSIGHT*, vol. 27, no. 5, pp. 44–49, 2024.

[81] CATiA, *Cameo Safety and Reliability Analyzer 2026x User Guide*. Dassault Systèmes company, 2025.

[82] R. Analysis and A. M. L. (RAAML), *Risk Analysis and Assessment Modeling Language (RAAML) Libraries and Profiles*. OMG Standards Development Organzation, 2021.

[83] L. BayesFusion, *SMILE Wrappers - Programmer's Manual*. BayesFusion, LLC, version 2.4.r1 ed., 2025.

[84] K. Li, L. Chen, X. Cai, C. Xu, Y. Lu, S. Luo, W. Wang, L. Jiang, and G. Wu, “A bayesian network approach to predicting severity status in nuclear reactor accidents with resilience to missing data,” *Energies*, vol. 18, no. 11, p. 2684, 2025.