A Framework for Formal Probabilistic Risk Assessment Using HOL Theorem Proving

Mohamed Abdelghany, Adnan Rashid(✉), and Sofiène Tahar

Department of Electrical and Computer Engineering, Concordia University, Montreal, QC, Canada
{m_eldes,rashid,tahar}@ece.concordia.ca

Abstract. It is indispensable to identify an occurrence and a possible consequence of any undesirable events in different components/subsystems of safety-critical systems, such as nuclear power plants and smart power grids that may often lead to a catastrophic accident and thus may result into financial losses and severe injuries. This process requires assessing the probabilities of occurrence of these events and is widely known as Probabilistic Risk Assessment (PRA). The frequently used methods for conducting the PRA are Event Tree Analysis (ETA), Functional Block Diagrams (FBD) and Cause Consequence Diagrams (CCD) (using Fault Trees (FT) and Reliability Block Diagrams (RBD)). We propose to use higher-Order-Logic (HOL) theorem proving for performing the formal PRA of safety-critical systems. In this paper, we present recent developments towards this direction and a roadmap towards a successful development of a framework for formal reasoning support for ETs, FBDs and CCDs in a HOL theorem prover.

Keywords: Probabilistic Risk Assessment · Event Tree Analysis · Functional Block Diagram · Cause Consequence Diagram · Higher-Order Logic · Theorem Proving · HOL4

1 Introduction

Probabilistic Risk Assessment (PRA) [21] is a well-known comprehensive methodology adopted by planners/designers for evaluating risks associated with safety-critical systems. It consists of three major steps, according to the risk management standard ISO 31000 [27]. The first step involves the *identification* of any potential risks in subsystems or components of a system causing their failures, which is based on a prior knowledge about working and functionality of each component of the system. This step is followed by the risk and reliability *analysis*, which is a systematic process for understanding the nature and severity of the identified risk. The final step is the risk *evaluation*, i.e., it involves a comparison of the risk analysis performed in the last step with the risk criteria for determining if any action is required [25]. The second step of the PRA is the most

A. Kohlhase and L. Kovács (Eds.): CICM 2024, LNAI 14960, pp. 298–314, 2024.
https://doi.org/10.1007/978-3-031-66997-2_17

crucial from the computational perspective, where the risk is characterized by two major quantities with respect to the possible adverse consequences; (a) the likelihoods of occurrence of each of these consequences; and (b) the severity level of these consequences. The likelihoods of occurrence of these consequences are the frequencies or probabilities of occurrence of these events per unit time that are determined through the reliability evaluations [26]. Therefore, it is utmost important to evaluate the probabilities of occurrence of the risk consequences at the system and subsystem levels during the design process of these systems using appropriate reliability modelling and analysis techniques.

Various reliability modelling approaches have been developed over decades for determining the PRA of systems. These include graph theoretic methods, namely Fault Trees (FTs) [24], Reliability Block Diagrams (RBDs) [35] and Event Trees (ETs) [36]. FTs use the notion of logic gates to develop a graphical model for analyzing the factors causing a complete system failure. Whereas, RBDs present a schematic structure based on various configurations (series/parallel) of system's components, for analyzing their success/reliability relationships ensuring a complete reliable system. In contrast to FTs and RBDs, an Event Tree (ET) is a complete risk tree graphical model providing all possible partial/complete failure and reliability consequence scenarios at the system level *simultaneously*. Moreover, Event Tree Analysis (ETA) is used to associate failure and success events to all components of a system in more complex hierarchical structures, such as Functional Block Diagrams (FBDs) [30]. A Functional Block Diagram (FBD) provides a graphical representation of the detailed functionality of the system and the functional relationship between all its components/subsystems that are captured as Functional Blocks (FBs). More recently, another approach has been proposed to perform ETA in conjunction with FTs and RBDs in order to identify failure and reliability events associated with all components of a safety-critical system and their cascading dependencies on the whole system. This analysis technique method is named as Cause Consequence Analysis (CCA), which uses a combined hierarchical structure of Cause Consequence Diagrams (CCDs) [34]. A CCD is a graphical method used to model the causes of subsystems failures or reliability in a system using FTs or RBDs, respectively. Therefore, the PRA using ETs/CCDs/FBDs can be used for all required system- and subsystem-level improvements, satisfying the reliability demand within acceptable risk levels.

Conventionally, the risk and reliability analysis of systems is performed using analytical [32] and computer-based simulation [20] methods. The analytical approaches are the paper-and-pencil proof methods, where we construct a mathematical model of the given system on paper and manually evaluate the reliability indices from this model. However, when this kind of analysis approach is used for some real-time systems exhibiting complex operating procedures, the resulting analysis may lose its significance due to its nature, i.e., human error-proneness and involvement of the very cumbersome effort to perform the reliability analysis manually. Therefore, many planners/designers utilize the computer-based simulation method, such as the famous Monte-Carlo Simulation (MCS), which

performs faster computations and provides an automation of the analysis. It uses some random algorithms for predicting the real functional behavior of systems and provides an estimated average value of reliability parameters. However, the approximation of results and the involvement of unverified algorithms may compromise the accuracy of the analysis, which is not desired, given the safety-critical nature of systems.

Formal methods, such as petri-nets [33], model checking [14] and theorem proving [22] have been used as complementary approaches for accurately performing the reliability analysis. For example, Nyvlt et al. [28] used Petri nets for ETA by modelling the partial/complete system-level failure and success consequence events. The authors proposed a new method based on P-invariants to obtain a model of cascading dependencies in ETs [28]. However, Petri-nets suffer from the reachability problem [23], which means it is not easy to determine when it is safe to stop. Similarly, Ortmeier et al. [29] developed a model checking framework for deductive CCA. The authors used the SMV model checker [14] for formally verifying CCD proof obligations. However, model checking faces the state space explosion problem [15], specifically, for larger systems. Ahmad et al. [11,12] proposed to use theorem proving for formal reliability analysis. In particular, the authors formalized *multi-level static* FT and RBD structures in the HOL4 theorem prover and used them for formally analyzing various systems, such as smart grids and a satellite solar array [10]. However, the formalization of *static* RBDs and FTs can neither analyze the *dynamic* behavior of the critical system nor determine failure and reliability *simultaneously*. Later, Elderhalli et al. [19] used HOL4 for performing the formal dynamic dependability analysis of safety-critical systems. The authors formalized Dynamic Fault Trees (DFTs) [17] and Dynamic Reliability Block Diagrams (DRBDs) [18], and used them for formally analyzing several systems including a drive by wire system and shuffle-exchange networks [16]. However, their proposed framework enables analyzing either dynamic failure *or* reliability, only, of the given system.

In this paper, we describe an ongoing project for developing a framework for performing formal PRA of systems [9], capturing the failure and success states *simultaneously*, using Higher-order-logic (HOL) theorem proving. The project was initiated in 2018 at the Hardware Verification Group of Concordia University[1]. We opted the HOL4 theorem prover for this project and one of the reasons for this choice was the availability of rich libraries of probability, FTs and RBDs, which are used in our proposed framework. In this regard, we formalized ETs in HOL4 [2], which enable the formal verification of generic expressions of the probability of partial/complete failure and reliability of the whole system and thus are used to perform the formal PRA. Furthermore, we formalized CCDs based on FTs and RBDs [4], which are further used for formally verifying probabilistic expressions used for the failure analysis and identification of any potential areas of poor reliability for *multi-level* subsystems of a complex system, respectively. Similarly, we formalized FBDs [7] and used them for formally verifying probabilistic expressions modeling all consequence scenarios (failure/realiablity)

[1] https://hvg.ece.concordia.ca/projects/prob-it/pr10/.

at the subsystem level that could occur in a complex system. Finally, to facilitate the industrial planners/designers about using our proposed framework, we developed a software tool, named *Functional Block Diagram and Event Tree Modeling and Analysis* (FETMA) in Python [3] that provides a graphical user interface for performing PRA by evaluating the probabilistic assessment of all possible consequence events at the system and subsystem levels using our formalization of ETs/FBDs. It is important to note that the developed formalization of our proposed framework considers only *static* consequence risk failure and reliability events at a specific instant of time t for which the reliability analysis is undertaken. One of our future plans is to integrate the *dynamic* failure and reliability events in our framework, which preserve the history of sudden events occurrence in the required analysis. The current paper mainly summarizes our developed formalization and outlines our plans for the development of a complete framework for the formal PRA of real-world systems using a HOL theorem prover.

The rest of the paper is structured as follows: We present our proposed framework for the formal PRA in Sect. 2. Section 3 provides a brief description of ETs and its formalization in the HOL4 theorem prover. In Sect. 4, we present the formalization of FBDs based on FTs and RBDs. Section 5 presents an overview of CCDs and their formalization in HOL4. We provide the current status of the project and the remaining milestones in Sect. 6. Finally, Sect. 7 concludes the paper.

2 Proposed Framework

Figure 1 depicts the proposed framework for the formal PRA of safety-critical systems using the HOL4 theorem prover. This framework provides the formally verified system/subsystem level generic reliability expressions using our formalized libraries of ETs, CCDs and FBDs developed in HOL4. The designer/reliability engineer provides the system diagrams, descriptions, and subsystems' specifications as input to our framework as shown in Fig. 1. Based on the given input, such as reliability requirements, subsystems FT/RBD or *multi-level* FBDs, our framework allows the user to perform the system or subsystem level reliability analysis by selecting the suitable technique/libraries, such as ETs or CCDs, or FBDs, respectively. For example, to conduct the formal subsystem level reliability analysis using our FBD library, the user is required to provide the FBD *multi-level* model of the underlying system, as depicted by blue arrows in Fig. 1. Using our ET/FBD/CCD libraries, a user can easily verify all possible system/subsystem-level safety classes of complete/partial failure and reliability expressions based on any given probabilistic distribution, like Exponential/Weibull/Poisson, corresponding to the given system description. Moreover, our proposed framework enables us to compute the formally verified risk consequence expressions using Standard Meta Language (SML) functions, which can further be used to determine the significant reliability indices, such as System/Customer Average Interruption Frequency/Duration Index (SAIFI, CAIFI, SAIDI and CAIDI). Lastly, our

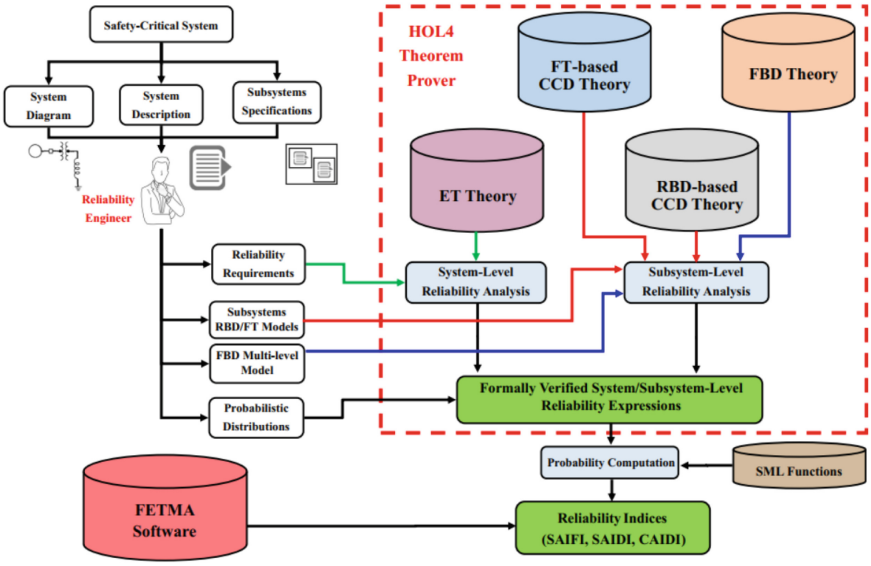


Fig. 1. Formal Probabilistic Risk Assessment

proposed framework incorporates the FETMA software, which provides a user friendly interface facilitating the PRA of real-world systems by performing related system/subsystem-level ET and FBD reliability analysis.

In the next sections, we provide more details of the proposed framework including the current status of the project by presenting the tasks that have been completed and provide some insights about the remaining steps.

3 Event Trees (ETs)

Event Tree Analysis (ETA) is a PRA method that enables modeling components failure and success states *simultaneously* and has been widely adopted for analyzing all possible system-level complete/partial failure and reliability consequence events. Generally speaking, an ET determines the cascading dependencies of different components on the entire system in the form of a tree structure. An ET diagram consists of an *Initiating Node* capturing an event and all possible consequence scenarios of occurrence of a sudden event in a system are drawn as *Branches* connected to *Proceeding Nodes* so that only one of these scenarios can occur, i.e., all possible ET consequence paths are mutually exclusive and distinct. For illustration purposes, we describe the concept of ETs using a small example of a Micro-Grid system that consists of wind turbines power generation (G) and two transmission lines (TL) to supply a certain load X, as depicted in Fig. 2. Moreover, each component of the underlying system has two operational states only, i.e., Success (S) or Failure (F). The four steps ETA is given as follows [31]:

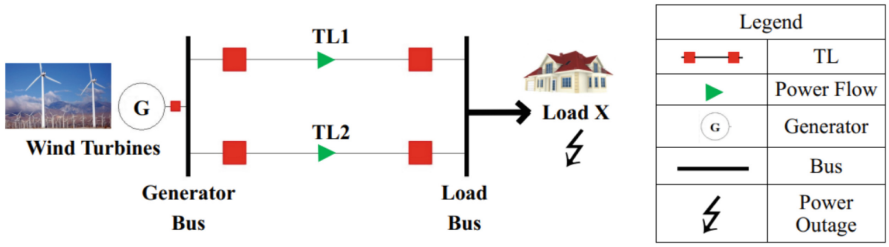


Fig. 2. Schematic of an Example Micro-Grid System

1. **ET Generation:** Draw a complete ET diagram containing all possible scenarios, referred as *Paths*. Each path has a unique sequence of failure/success events. Figure 3(a) presents eight ET consequence paths (Paths 0-7) with all possible scenarios that can occur in the Micro-Grid system.
2. **ET Reduction:** Model the accurate functional behavior of a system to reduce the number of possible test cases. This is achieved by deleting some specific nodes/branches corresponding to the occurrence of certain events, which are known as Complete Cylinders (CCs). These cylinders are ET paths consisting of N failure/success events and are subject to the occurrence of K Conditional Events (CEs) in their respective paths. They are typically known as CCs with respect to K. For the example of the Micro-Grid system, if the wind turbine generation G fails, then the whole grid fails regardless of the status of the transmission lines (TL1 and TL2), as depicted in Fig. 3(b). The ET paths 4-7 are CCs with respect to G_F.
3. **ET Partitioning:** This step is vital as we are only interested in the occurrence of certain failure/success events based on safety requirements of the system. For example, when we focus on the Complete Failure (CF) of the Micro-Grid, the ET Paths 3 and 4 are taken from the reduced ET as depicted in Fig. 3(b). Whereas, in the case of the grid's Complete Success (CS), we only consider Path 0.
4. **ET based Probabilistic Analysis:** This step involves evaluating the probabilities of ET paths based on the occurrence of an accident event in the system. These probabilities represent the likelihood of each scenario that can possibly occur in the system and are obtained by the multiplication of the individual probabilities of all failure/success events associated with the ET path under condition of the mutual independence of all events. For instance, the PRA of the Micro-Grid's CS is mathematically expressed as:

$$Pr(Micro-Grid_{CS}) = Pr(G_S) \times Pr(TL_{1S}) \times Pr(TL_{2S}) \quad (1)$$

where $Pr(X_S)$ is the probability of success of the component X.

We started the ET formalization by formally defining the basic constructors corresponding to a *single event*, *node* and *branch* of an ET and used them to formalize the ET diagram/structure in HOL4. This formalization is then used to

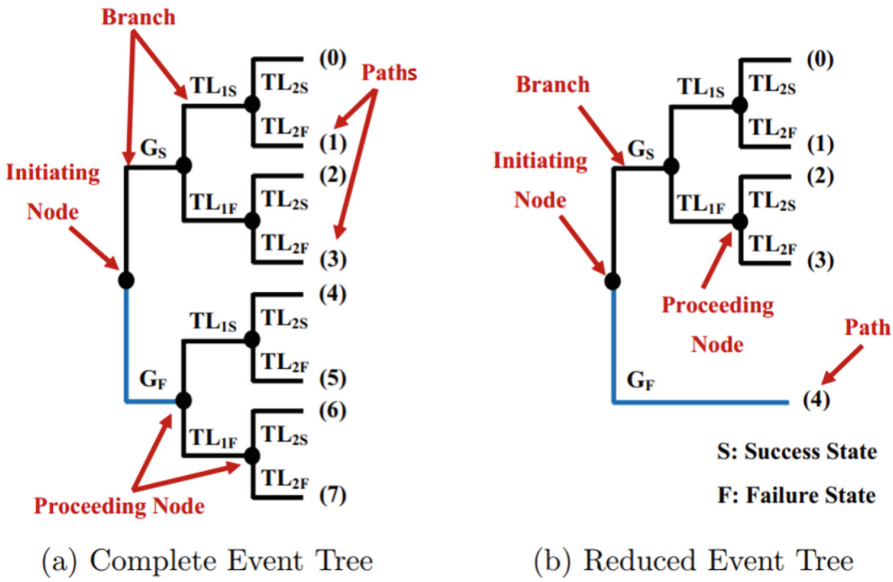


Fig. 3. Event Tree Diagrams of the Micro-Grid System

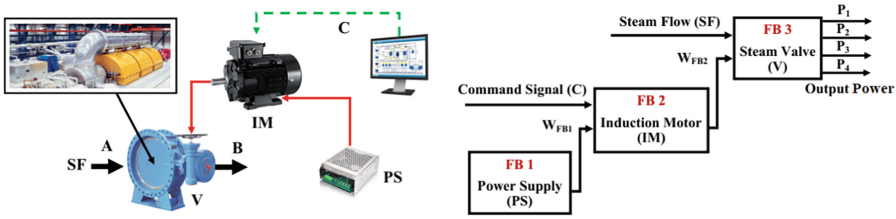
generate a generic ET model of a safety-critical system, which is the first step of ETA. Furthermore, we formalized relevant ET reduction functions that are used to reduce the number of possible test cases in an ET. Next, we formalized the partitioning function in HOL4 to extract a collection of ET paths based on the reliability requirements of the given system. Finally, we used our formalization developed in the previous steps to perform the actual formal ET probabilistic analysis, which mainly includes formal verification of the mathematical expressions providing the probabilities of failure and success events in the ET model of the system. These probabilities expressions are also used for formally verifying various reliability indices, such as SAIFI, CAIFI, SAIDI and CAIDI.

The verification of theorems corresponding to ETA in HOL4 required multiple levels of induction and were mainly based on several HOL4 theories, such as measure, probability, set, list, real and extended real. The complete ET formalization amounts to around 4,000 lines of HOL4 code[2].

4 Functional Block Diagrams (FBDs)

A Functional Block Diagram (FBD) is a more elaborate concept for ETA based PRA technique that involves constructing the hierarchical ET structures to perform subsystem level reliability analysis of complex systems. A Functional Block (FB) is a fundamental constructing element of an FBD graph that captures the

[2] The HOL4 code for the ET formalization can be found at https://github.com/hvg-concordia/ET.



(a) Steam-Turbine Governor System (b) FBD of Steam-Turbine Governor

Fig. 4. Steam-Turbine Governor of a Thermal Power Plant

stochastic behavior of each subsystem in a system. To provide a better understanding of the FBD-based reliability analysis, we consider an example of a Turbine Governor System of a steam power plant, as depicted in Fig. 4(a). This system mainly regulates the steam flow to the turbine by controlling the position of the Steam Inlet Valve (V) and thus controls the output power. The valve operates with an Induction Motor (IM) that is energized by a Power Supply (PS). The main purpose of the valve is to control the Steam Flow (SF) at Point B given the flow situation at Point A and a command signal C dictating the required operation (opening or closing) of the valve. The six step FBD analysis is as follows:

1. **FBD Construction:** Construct an FBD of the system based on the engineering knowledge, which consists of FBs describing the subsystem-level behavior, as depicted in Fig. 4(b).
2. **ET Generation:** Draw a complete ET model corresponding to each subsystem FB. Assuming each subsystem component is represented by two operating states only, i.e., Success (S) or Fail (F). Figure 5 presents all subsystems' complete ETs of the steam-turbine governor, i.e., $ET_{1(Complete)}$, $ET_{2(Complete)}$ and $ET_{3(Complete)}$ corresponding to FB_1, FB_2 and FB_3, respectively.
3. **ET Reduction:** Reduce the ETs by removing some nodes/branches according to the subsystems' functionality. For example, in $ET_{2(Complete)}$ corresponding to FB_2 of the steam-turbine governor, if the power supply FB_1 fails, then the whole IM fails regardless of the status of its other elements, as depicted in Fig. 5. The reduced ETs corresponding to the complete ETs, $ET_{1(Complete)}$, $ET_{2(Complete)}$ and $ET_{3(Complete)}$, are presented as $ET_{1(Reduced)}$, $ET_{2(Reduced)}$ and $ET_{3(Reduced)}$, respectively, in Fig. 5.
4. **ET Composition:** Compose all reduced ETs together considering the functional behavior of the steam-turbine governor to form a complete subsystem-level ET model. Here, $ET_{1(Reduced)}$, $ET_{2(Reduced)}$ and $ET_{3(Reduced)}$ are combined to form a subsystem-level $ET_{Governor}$, as shown in Fig. 5, with all possible complete/partial failure and reliability ET consequence paths that can occur.
5. **ET Partitioning:** This step involves extracting the respective paths of the ETs based on the requirement pertaining to the occurrence of certain events. For example, if we focus on the Complete Failure (CF) of the IM only (Fig. 4(a)), then the ET Paths 3-5 (Fig. 5) are considered from $ET_{Governor}$.

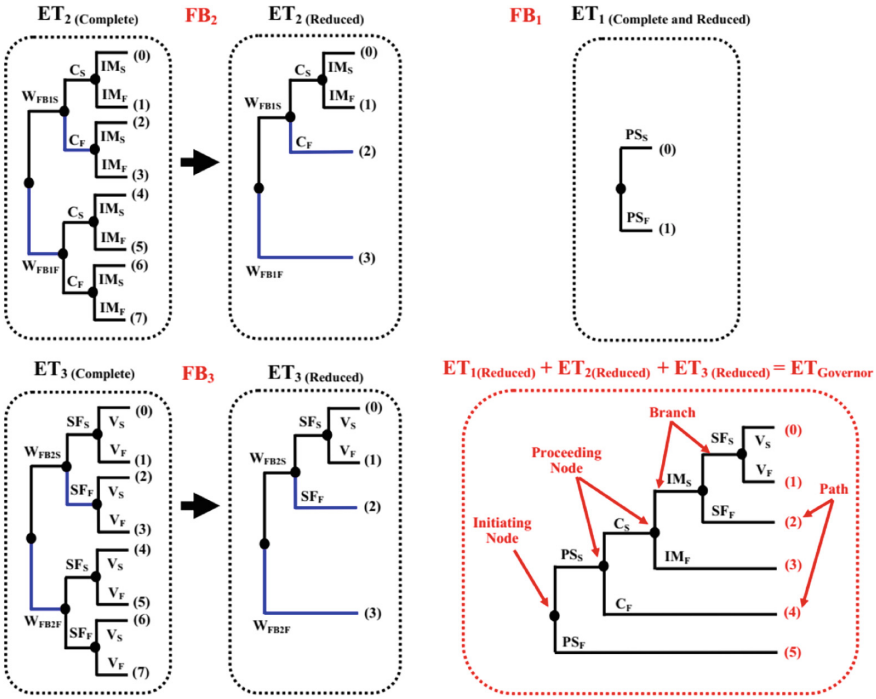


Fig. 5. Functional Block Diagrams of the Steam-Turbine Governor

6. **FBD based Probabilistic Analysis:** This step evaluates the probabilities of all possible safety classes of complete/partial failure and reliability subsystem-level ET paths based on the occurrence of a certain event in the entire system. For example, the probability of CF of the system's component IM, i.e., IM_{CF} event ($ET_{Path3-5}$, Fig. 5) is mathematically expressed as:

$$Pr(IM_{CF}) = Pr(PS_S) \times Pr(C_S) \times Pr(IM_F) + Pr(PS_S) \times Pr(C_F) + Pr(PS_F) \quad (2)$$

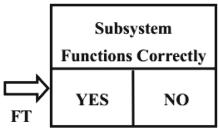
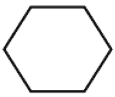
We started the formalization of FBDs by defining a modeling function for its basic element FB in HOL4, which is further used to construct an FBD model consisting of *multi-level* FBs. In the next step of the FBD based reliability analysis, we developed ET models for consecutive *multi-level* FBs formalized in the last step, using our formalization of ETs presented in Sect. 3. The correct construction of these models is ensured by formally verifying some FBD modelling properties, such as splitting, commutativity and associativity of the FBs. To perform the next two steps of the analysis, i.e., model reduction and splitting, we used the corresponding functions developed for ETs (Sect. 3). The last step of the FBD reliability analysis is to determine the probability of each ET consequence risk scenario at the subsystem-level that could occur in a complex system. We used the ET probabilistic theorems (verified as a part of ET analysis in Sect. 3) and the FBD modeling properties to verify the FBD probabilistic properties for

different configurations of FB connections in HOL4. The HOL4 proofs conducted for the FBD based formal analysis mainly required hierarchical levels of induction cases and libraries, such as ET, measure, probability, set, list, real and extended real theories. The proof-script of this formalization amounts to around 3,500 lines of HOL4 code[3].

5 Cause Consequence Diagrams (CCDs)

CCD [34] is a more recent PRA method that is conventionally used to model the causes of subsystem failures or reliability in a system using FTs or RBDs, respectively, and their potential consequences on the entire system, using ETA. The graph theory of CCDs uses three basic constructors *Decision box*, *Consequence Path* and *Consequence Box* for constructing the CCD model. The graphical representation and functionality of these constructors are given in Table 1 [37]. To obtain a better understanding of using FTs in CCDs, we consider a renewable energy solar Photo-Voltaic (PV) system supplying energy to a private house. The PV system mainly comprises of two subsystems namely, a Solar Array (SA) consisting of three series solar PV panels and an Inverter Bridge (IB) that converts Direct Current (DC) to Alternating Current (AC) through Switches and Filters, as depicted in Fig. 6. The four steps CCD analysis for the solar PV system is given as follows [13]:

Table 1. CCD Symbols and Functions

CCD Symbol	Function
	**Decision Box** represents the status of functionality for a component or subsystem (1) **NO Box** describes the subsystem failure operation. A FT of the subsystem is connected to this box that can be used to obtain the failure probability, i.e., $Pr_{NO} = Pr_{FT}$ (2) **YES Box** represents the correct functioning of the subsystem or reliability, which can be determined by simply taking the complement of the failure operation, i.e., $Pr_{YES} = 1 - Pr_{FT}$
	**Consequence Path** models all possible consequence scenarios based on subsystem failure or reliability
	**Consequence Box** models the final outcome due to a particular sequence of events for all subsystems

[3] The HOL4 code for the FBD formalization can be found at https://github.com/hvg-concordia/FBD.

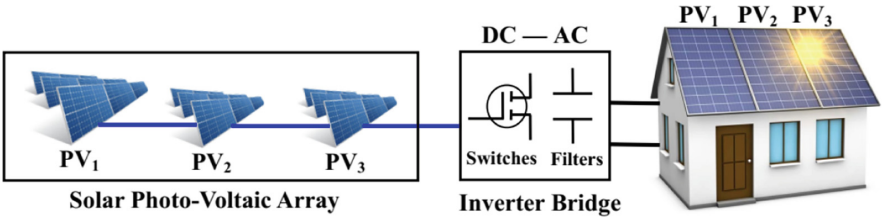


Fig. 6. Solar Photo-Voltaic (PV) System

1. **Components Failure Events:** Assign a FT model to each subsystem (SA and IB) in the solar system, i.e., FT_{SA} (OR connection) and FT_{IB} (OR connection). The mathematical expressions for the probabilities of failure of components SA and IB of the solar system, respectively, are given as:

$$FT_{SA} = 1 - \Big(\big(1 - Pr(PV_{1F})\big) \times \big(1 - Pr(PV_{2F})\big) \times \big(1 - Pr(PV_{3F})\big)\Big) \quad (3)$$

$$FT_{IB} = 1 - \Big(\big(1 - Pr(Switches_F)\big) \times \big(1 - Pr(Filters_F)\big)\Big) \quad (4)$$

2. **Complete CCD Construction:** Construct a complete CCD model of the PV system as shown in Fig. 7(a). Based on the output of the SA decision box that is either YES or NO, the next subsystem IB is taken into consideration. Each consequence path in the CCD analysis ends with either a PV system success (PV_S) or a PV system failure (PV_F).
3. **CCD Reduction:** Reduce the complete CCD model to decrease the number of test cases and model the accurate behavior of the PV system. If the condition of the SA decision box (SA functions correctly) is not satisfied, i.e., if it is a NO box, then the PV fails regardless of the status of the IB. The reduced CCD model of the PV system is presented in Fig. 7(b).
4. **CCD based Probabilistic Analysis:** The PRA of the two consequence boxes PV_S and PV_F at the subsystem level, as shown in Fig. 7(b), are mathematically expressed based on Eqs. (3) and (4) as follows:

$$\begin{aligned} Pr(PV_S) = Pr(SA_{YES}) \times Pr(IB_{YES}) = \\ \Big(\big(1 - Pr(PV_{1F})\big) \times \big(1 - Pr(PV_{2F})\big) \times \big(1 - Pr(PV_{3F})\big)\Big) \times \\ \Big(\big(1 - Pr(Switches_F)\big) \times \big(1 - Pr(Filters_F)\big)\Big) \end{aligned} \quad (5)$$

$$\begin{aligned} Pr(PV_F) = Pr(SA_{YES}) \times Pr(IB_{NO}) + Pr(SA_{NO}) = \\ \Bigg(\Big(\big(1 - Pr(PV_{1F})\big) \times \big(1 - Pr(PV_{2F})\big) \times \big(1 - Pr(PV_{3F})\big)\Big) \times \\ \Big(1 - \big(1 - Pr(Switches_F)\big) \times \big(1 - Pr(Filters_F)\big)\Big)\Bigg) + \\ \Big(1 - \big(1 - Pr(PV_{1F})\big) \times \big(1 - Pr(PV_{2F})\big) \times \big(1 - Pr(PV_{3F})\big)\Big) \end{aligned} \quad (6)$$

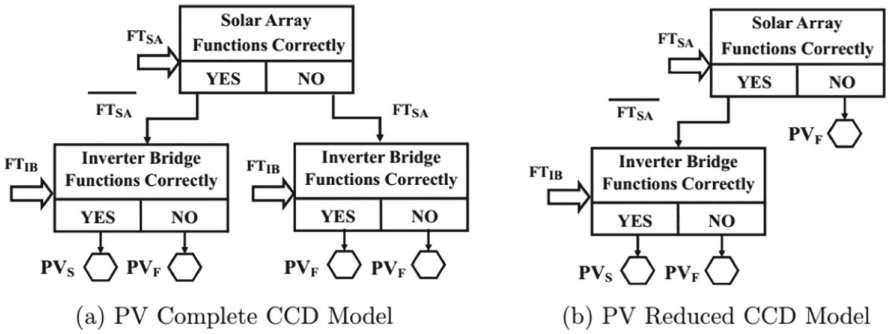


Fig. 7. Cause Consequence Diagrams of the Photo-Voltaic System

where $Pr(X_{NO})$ is the probability of failure for a subsystem X, i.e., the FT_X model, and $Pr(X_{YES})$ is the reliability function or the probability of operating (success), i.e., the complement of the FT_X model.

To perform the CCD-based formal PRA, we used the existing formalization of FTs [10] in HOL4 and our formalization of ETs, developed as a part of this project (Sect. 3). We started the formalization of CCDs by developing formal models of its basic constructing symbols, i.e., CCD *Decision Box*, *Consequence Path* and *Consequence Box*, which are further used to formalize a complete CCD model of a system. Next, this formal model is further reduced using the reduction functions formalized in HOL4. Finally, we formally verified CCD generic probabilistic properties involving different FT configuration connected with *Decision Box* in a CCD-based model. Furthermore, we extended our formalization by performing the *multi-level* CCD reliability analysis for the complex systems, which consist of *multi-level* decision boxes corresponding to connected *multi-level* subsystems, where each subsystem is analyzed by different logic gates associated with an arbitrary list of failure events (*multi-level* FT models) and a *multi-level* ET model. The verification of the related theorems was a bit challenging since we were dealing with all four types of different FT configurations, i.e., AND, NAND, OR and NOR [24], where each type consists of generic n decision boxes and each decision box is associated with generic m events, simultaneously in HOL4. The proof-script of the FT-based CCD formalization work amounts to about $7,000$ lines of HOL4 code[4]. Similarly, we performed the RBD-based CCD reliability analysis in HOL4, using a combination of the existing formalization of RBDs in HOL4 [10] and our formalization of ETs (Sect. 3). The proof-script of the RBD-based CCD formalization work amounts to about $3,500$ lines of HOL4 code[5].

[4] The HOL4 code for the FT-based CCD formalization can be found at https://github.com/hvg-concordia/CCD.

[5] The HOL4 code for the RBD-based CCD formalization can be found at https://github.com/hvg-concordia/CCD_RBD.

6 Current Status and Future Milestones

The ultimate goal of the proposed project is to develop a tool that can be used for formal PRA of safety-critical systems, i.e., to perform the ET, FBD and CCD based risk and reliability analysis considering both *static* and *dynamic* failures as well as reliability events. To achieve this goal, we started with the formalization of ETs [1] in the HOL4 theorem prover. In particular, we formalized the ET structure/diagram based on some basic ET constructs in HOL4, which are further used to conduct the ET probabilistic analysis providing the formal verification of the mathematical expressions capturing the probabilities of *static* failure and success events in the ET model of the system. We illustrated the effectiveness of our proposed formalization of ETs by conducting the formal ET based analysis of the IEEE 39-bus electrical power grid, IEEE 3-Bus bulk power system, HVDC coupling between Quebec and New England, IEEE 118-bus power network [8,9]. Entries 1 and 2 of Table 2 summarize these accomplished tasks about the ET analysis.

Next, we provided a support for formal FBD based analysis [7] in HOL4. In particular, our formalization involves FBD modeling, reduction, partitioning and verification of the expressions of reliability. To illustrate its effectiveness, we performed formal FBD analysis of nuclear power plant with multiple-levels decompositions of the boiling water reactor [6]. These accomplished tasks are listed as Entries 3 and 4 of Table 2. Similarly, we used the formalization of the *static* FTs [10] and RBDs [10] available in HOL4 alongside our formalization of ETs developed in the last step to perform the CCD analysis [4,5]. In particular, we formalized a CCD model of a system based on the formal models of its basic constructing symbols, i.e., CCD Decision Box, Consequence Path and Consequence Box. Next, we reduced the model by applying the reduction functions formalized in HOL4. We also formally verified generic probabilistic properties involving different FT and RBD configurations connected with Decision Box in a CCD-based model. We illustrated the usefulness and utilization of the formalization of CCDs by conducting the reliability analysis of Interconnected Micro-Grids and IEEE 39-bus distributed generation network [8,9]. These accomplished contributions are listed as Entries 5 and 6 of Table 2. Finally, we developed a Python based FETMA software providing a user-friendly interface and computation of the ET and FBD reliability at system/subsystem levels using our formalizations of ET and FBD developed as a part of this project [9]. These contributions are summarized as Entries 7 and 8 of Table 2.

The remaining tasks of this project can be divided into two major categories: 1) the formalization of the mathematical models (Entries 9 and 10 of Table 2); and 2) the development of a comprehensive tool for a complete formal PRA (Entries 11, 12 and 13 of Table 2). For the first category, we need to extend the formalization of the ETs by considering the *dynamic* failure and success events incorporating the history of sudden events occurrence. This requires the formalizations of DFTs [17] and DRBDs [18], which are already available in HOL4. Furthermore, we have to formalize CCDs considering *both* the failure as well as reliability (static) analysis. This formalization mainly requires combining the formalizations of FTs [11] and RBDs [12] available in HOL4 alongside the formal-

ization of ETs developed in Sect. 3. Regarding the implementation of the second category, we plan to extend the FETMA software to incorporate FT/RBD based CCD analysis. Also, we need to integrate the formal system/subsystem-level probabilistic evaluation of systems in HOL4 with the user-friendly graphical interfaces of FETMA software. This requires constructing a Python-HOL4 Parser that provides the translation from FETMA to HOL4 and vice versa. Moreover, based on the type of the input model, the end-user can select the appropriate PRA method (ETs, CCDs or FBDs) for reliability analysis. Next, we intend to use Machine Learning (ML) to automate the formal PRA analysis. This includes classifying the useful theorems and choosing the proper tactics to be used in the verification of the input model. To achieve this step, we need to create training and test datasets from the formalization of ETs, CCDs and FBDs that can be used in developing the proper ML models. Finally, we have to implement the core of the tool that connects the pieces of the framework together to enable the automatic formal PRA. A summary of this roadmap is provided in Table 2.

Table 2. Project Roadmap

Task	Description	Duration
Accomplished Tasks		
1	HOL formalization of ET - ET modeling, ET reduction and ET partitioning [1,2] - Formal ET probabilistic analysis [1,2]	11 Months
2	ET applications - IEEE 39-bus electrical power grid [1,8] - IEEE 3-Bus bulk power system [9] - HVDC coupling between Quebec and New England [9] - IEEE 118-bus power network [9]	3 Months
3	HOL formalization of FBDs - FBD modeling, FBD reduction and FBD partitioning [7] - Formal FBD probabilistic analysis [7]	5 Months
4	FBDs applications - Nuclear power plant with multiple-levels decomposition of the boiling water reactor [6]	1 Month
5	HOL formalization of FT and RBD based CCDs - Subsystems failure/success events modeling, a complete CCDconstruction, CCD reduction [4,5] - Formal CCD probabilistic analysis [4,5]	9 Months
6	CCDs applications - Interconnected Micro-Grids [9] - IEEE 39-bus distributed generation network [4]	1 Month
7	Development of FETMA software - Developing the user-friendly interface ETMA [3] - Computation of ET and FBD reliability at system/subsystem levels using SML [9]	4 Months
8	FETMA applications - Power transmission distance protection scheme [9] - Smart automated substation in a smart power grid [9]	1 Month

(*continued*)

Table 2. (*continued*)

Task	Description	Duration
Future Plan		
9	Formal ET analysis based on *dynamic* failure/reliability events incorporating the history of sudden events occurrence - Formalization of mathematical expressions for multi-states of failure/reliability of multi-level components for connected subsystems - Requires combining three formalizations DFTs, DRBDs and ETs simultaneously	10 Months
10	Formalization of CCD diagrams considering both failure and reliability analysis simultaneously - Requires combining three formalizations FTs, RBDs and ETssimultaneously	6 Months
11	Extension of FETMA software to incorporate FT/RBD based CCD analysis and reliability analysis	2 Months
12	Integration of formal PRA in HOL4 and FETMA - Requires constructing a Python-HOL4 parser providing the automatic translation from FETMA to the HOL4 code and vice versa.	3 Months
13	Using Machine Learning (ML) to automate the formal PRA analysis in HOL4 - Requires creating training and test datasets of HOL4 libraries - Constructing appropriate ML models	4 Months
	Total Time	60 Months

7 Conclusion

This paper proposed a comprehensive framework to perform the formal Probabilistic Risk Assessment (PRA) using HOL theorem proving. We presented the details of the mathematical foundations of each part/component of the proposed framework. The main contribution of this work is the formalization of Event Trees (ETs), Functional Block Diagrams (FBDs) and Cause Consequence Diagrams (CCD). This formalization allows us to perform the formal PRA of many real-world system by considering the failure and reliability event simultaneously. We also described the future milestones to complete the proposed project including the final tool enabling an automation of the analysis.

References

1. Abdelghany, M., Ahmad, W., Tahar, S.: Event tree reliability analysis of electrical power generation network using formal techniques. In: Electric Power and Energy Conference, pp. 1–7. IEEE (2020)
2. Abdelghany, M., Ahmad, W., Tahar, S.: Event tree reliability analysis of safety-critical systems using theorem proving. IEEE Syst. J. **16**(2), 2899–2910 (2021)
3. Abdelghany, M., Ahmad, W., Tahar, S., Nethula, S.: ETMA: an efficient tool for event trees modeling and analysis. In: International Systems Conference, pp. 1–8. IEEE (2020)
4. Abdelghany, M., Tahar, S.: Cause-consequence diagram reliability analysis using formal techniques with application to electrical power networks. IEEE Access **9**, 23929–23943 (2021)

5. Abdelghany, M., Tahar, S.: Formalization of RBD-based cause consequence analysis in HOL. In: Kamareddine, F., Sacerdoti Coen, C. (eds.) CICM 2021. LNCS (LNAI), vol. 12833, pp. 47–64. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-81097-9_4
6. Abdelghany, M., Tahar, S.: Formal probabilistic risk assessment of a nuclear power plant. In: Formal Techniques for Safety-Critical Systems, pp. 80–87 (2022)
7. Abdelghany, M., Tahar, S.: Formalization of functional block diagrams using HOL theorem proving. In: Lima, L., Molnár, V. (eds.) SBMF 2022. LNCS, vol. 13768, pp. 22–35. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-22476-8_2
8. Abdelghany, M., Tahar, S.: Reliability analysis of smart grids using formal methods. In: Fathi, M., Zio, E., Pardalos, P.M. (eds.) Handbook of Smart Energy Systems, pp. 147–163. Springer, Cham (2023). https://doi.org/10.1007/978-3-030-97940-9_81
9. Abdelghany, M.W.E.: Formal probabilistic risk assessment using theorem proving with applications in power systems. Ph.D. thesis, Concordia University, Montreal, QC, Canada (2021)
10. Ahmad, W.: Formal dependability analysis using higher-order-logic theorem proving. Ph.D. thesis, National University of Sciences & Technology, Islamabad, Pakistan (2017)
11. Ahmed, W., Hasan, O.: Towards formal fault tree analysis using theorem proving. In: Kerber, M., Carette, J., Kaliszyk, C., Rabe, F., Sorge, V. (eds.) CICM 2015. LNCS (LNAI), vol. 9150, pp. 39–54. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-20615-8_3
12. Ahmed, W., Hasan, O., Tahar, S.: Formalization of reliability block diagrams in higher-order logic. J. Appl. Log. **18**, 19–41 (2016)
13. Andrews, J.D., Ridley, L.M.: Application of the cause-consequence diagram method to static systems. Reliab. Eng. Syst. Saf. **75**(1), 47–58 (2002)
14. Clarke, E.M.: Model checking. In: Ramesh, S., Sivakumar, G. (eds.) FSTTCS 1997. LNCS, vol. 1346, pp. 54–56. Springer, Heidelberg (1997). https://doi.org/10.1007/BFb0058022
15. Clarke, E.M., Klieber, W., Nováček, M., Zuliani, P.: Model checking and the state explosion problem. In: Meyer, B., Nordio, M. (eds.) LASER 2011. LNCS, vol. 7682, pp. 1–30. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-35746-6_1
16. Elderhalli, Y.: Dynamic dependability analysis using HOL theorem proving with application in multiprocessor systems. Ph.D. thesis, Concordia University, Montreal, QC, Canada (2019)
17. Elderhalli, Y., Ahmad, W., Hasan, O., Tahar, S.: Probabilistic analysis of dynamic fault trees using HOL theorem proving. J. Appl. Logics—IfCoLog J. Logics Appl. **6**(3) (2019)
18. Elderhalli, Y., Hasan, O., Tahar, S.: A formally verified algebraic approach for dynamic reliability block diagrams. In: Ait-Ameur, Y., Qin, S. (eds.) ICFEM 2019. LNCS, vol. 11852, pp. 253–269. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-32409-4_16
19. Elderhalli, Y., Hasan, O., Tahar, S.: A framework for formal dynamic dependability analysis using HOL theorem proving. In: Benzmüller, C., Miller, B. (eds.) CICM 2020. LNCS (LNAI), vol. 12236, pp. 105–122. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-53518-6_7
20. Gardoni, P.: Risk and Reliability Analysis. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-52425-2

21. Haimes, Y.Y.: Risk Modeling, Assessment, and Management. Wiley, Hoboken (2005)
22. Harrison, J., Urban, J., Wiedijk, F.: History of interactive theorem proving. In: Computational Logic, vol. 9, pp. 135–214 (2014)
23. Hasan, O., Tahar, S.: Formal verification methods. In: Encyclopedia of Information Science and Technology, 3rd edn, pp. 7162–7170. IGI Global (2015)
24. Hixenbaugh, A.: Fault Tree for Safety. Seattle: The Boeing Company, D6 53604 (1968)
25. Hutchins, G.: ISO 31000: 2018 Enterprise Risk Management. Greg Hutchins (2018)
26. Kumamoto, H.: Satisfying Safety Goals by Probabilistic Risk Assessment. Springer, London (2007). https://doi.org/10.1007/978-1-84628-682-7
27. Leitch, M.: ISO 31000: 2009-the new international standard on risk management. Risk Anal. **30**(6), 887 (2010)
28. Nỳvlt, O., Rausand, M.: Dependencies in event trees analyzed by petri nets. Reliab. Eng. Syst. Saf. **104**, 45–57 (2012)
29. Ortmeier, F., Reif, W., Schellhorn, G.: Deductive cause-consequence analysis (DCCA). IFAC Proc. Vol. **38**(1), 62–67 (2005)
30. Papazoglou, I.A.: Functional block diagrams and automated construction of event trees. Reliab. Eng. Syst. Saf. **61**(3), 185–214 (1998)
31. Papazoglou, I.A.: Mathematical foundations of event trees. Reliab. Eng. Syst. Saf. **61**(3), 169–183 (1998)
32. Rasmussen, J.: Trends in human reliability analysis. Ergonomics **28**(8), 1185–1195 (1985)
33. Reisig, W.: Petri Nets: An Introduction, vol. 4. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-69968-9
34. Ridley, L.M.: Dependency modelling using fault-tree and cause-consequence analysis. Ph.D. thesis, Loughborough University, UK (2000)
35. Staley, J., Sutcliffe, P.: Reliability block diagram analysis. Microelectron. Reliab. **13**(1), 33–47 (1974)
36. Wall, I.: Probabilistic risk assessment in nuclear power plant regulation. Nucl. Eng. Des. **60**(1), 11–24 (1980)
37. Xin, B., Wan, L., Yu, J., Dang, W.: Basic event probability determination and risk assessment based on cause-consequence analysis method. In: Journal of Physics: Conference Series, vol. 1549, p. 052094. IOP Publishing (2020)