Risk Modeling of Variable Probability External Initiating Events in a Functional Modeling Paradigm

Article · August 2018

CITATIONS
2

READS
45

4 authors:

Douglas Lee Van Bossuyt
Naval Postgraduate School
66 PUBLICATIONS **168** CITATIONS
SEE PROFILE

Jose Dempere
Colorado School of Mines
2 PUBLICATIONS **4** CITATIONS
SEE PROFILE

Nikolaos Papakonstantinou
VTT Technical Research Centre of Finland
47 PUBLICATIONS **254** CITATIONS
SEE PROFILE

Bryan O'Halloran
Naval Postgraduate School
48 PUBLICATIONS **108** CITATIONS
SEE PROFILE

Risk Modeling of Variable Probability External Initiating Events in a Functional Modeling Paradigm

Jose Dempere
Nikolaos Papakonstantinou
Bryan O'Halloran
Douglas L. Van Bossuyt
(Corresponding Author)

Summary & Conclusions

As component engineering has progressively advanced over the past 20 years to encompass a robust element of reliability, a paradigm shift has occurred in how complex systems fail. While failures used to be dominated by 'component failures,' failures are now governed by other factors such as environmental factors, integration capability, design quality, system complexity, built-in testability, etc. Of these factors, environmental factors are some of the most difficult to predict and assess. While test regimes typically encompass environmental factors, significant design changes to the system to mitigate any potential failures is not likely to occur due to the cost. The early stages of the systems engineering design process offer significant opportunity to evaluate and mitigate risks due to environmental factors.

Systems that are expected to operate in a dynamic and changing environment have significant challenges for assessing environmental factors. For example, external failure initiating event probabilities may change with respect to time, and new discovered external initiating events may also be expected to have varying probabilities of occurrence with respect to time. While some industry standard methods such as Probabilistic Risk Assessment (PRA) [3] and Failure Modes and Effects Analysis (FMEA) [4] can partially address a time-dependent external initiating event probability, current methods of analyzing system failure risk during conceptual system design cannot.

We have developed the Time Based Failure Flow Evaluator (TBFFE) to address the need for a risk analysis tool that can account for variable probabilities in initiating events over the duration of a system's operation.

This method builds upon the Function Based Engineering Design (FBED) [19] method of functional modeling and the Function Failure Identification and Propagation (FFIP) [9] failure analysis method that is compatible with FBED. Through the development of TBFFE, we have found that the method can provide significant insights into a design that is to be used in an environment with variable probability external initiating events. We present a case study of the conceptual design of a nuclear power plant's spent fuel pool experiencing a variety of external initiating events that vary in probability based upon the time of year. The case study illustrates the capability of TBFFE by identifying how seasonally variable initiating event occurrences can impact the probability of failure on a monthly timescale that otherwise would not be seen on a yearly timescale. Changing the design helps to reduce the impact that time-varying initiating events have on the monthly risk of system failure.

1 Background

There are several methods required to understand TBFFE and its background; one commonality all of those methods have is that they do not easily model failure probability shifts caused by time-based initiating event probabilities. This section reviews related methods and demonstrates the novelty of the TBFFE method presented in this paper.

1.1 Functional Modeling

Functional modeling connects a series of inputs, outputs, flows, and functions together that transform the inputs into outputs [19]. This tool is useful for modeling systems at a variety of fidelities. A common functional modeling implementation is FBED [19]; we use FBED as the basis for our method. Functional modeling's robustness makes it a useful tool for many kinds of systems modeling [12, 13, 14]. However, functional modeling is not useful for the stated goal of this method because it does not model failure.

1.2 Function Failure Design Method

The Function Failure Design Method (FFDM) is the groundwork upon which TBFFE is built [7]. Within FFDM, all functions are given a list of potential failures, and the probability of that failure is then cataloged for every function in the functional model. Next, the probability of a functional model failing can be calculated in much the same way as a failure is calculated in PRA—via cutset development and calculation. The limitation of FFDM is that there is no way to modify a risk's chance of occurring over time without creating a new functional model.

1.3 Function Failure Identification and Propagation

FFIP is an extension of the functional modeling theory underlying FFDM [9]. Instead of utilizing a table to quantify the possible failures, FFIP analysis iterates through failures of possible functions and follows the failure flow until it exits the system as an output. This method allows for a user to see how a failure state transmits across a complex system and whether or not it ultimately poses a major risk to the system. While the addition of flows adds the concept of failure propagation to a system modeled using functional modeling, FFIP does not include time-based failure probabilities.

1.4 Related Functional Model-Based Methods

There are various authors in multiple fields that have attempted to address the subject of applying time-variable risk analysis, but few of them have addressed functional modeling. For instance, Hutcheson et. al. fit failure modes to functions during prototypes through a time-informed lens [7]. While Hutcheson's method creates a certain amount of flexibility for modeling various stages of a mission when a system may be in different configurations, the method does not encapsulate different rates of change [6]. Another related method is a semi-functional nonparametric

analysis by Aneiros-Perez that attempts to use a series of past values as predictors for later behavior [1]. This method does not match the needs of functional modeling because its nonparametric analysis methods are well beyond the scope of a basic functional model or FFDM methods. Dynamic risk assessment techniques, such as those described by Siu, are applicable to multiple engineering systems, but they lack a functional framework and focus instead on the use of PRA and similar systems [15].

The final related method discussed here was developed by Woltjer et. al. and presents a functional analysis meant to react to shifting airplane conditions [19]. However, there are issues with this method because it does not account for multiple potential failure conditions and uses velocity components to resolve a time-based issue so that planes enter in the right order to a flight pattern rather than utilizing time to modify the failure velocity. In essence, Woltjer et. al.'s analysis method is, much like Hutcheson's, meant to change dynamically with time rather than take into account time from a risk analysis perspective so that probabilistic risk of failure may be determined.

1.5 Probabilistic Risk Assessment

PRA is a method that exists outside of the functional flow modeling methods that have been discussed so far. PRA usually is implemented at the component level rather than the functional level [6]. The focus of the PRA method is to create a series of cutsets based on initiating events to create a series of potential failure pathways and their associated probabilities.

PRA's use in nuclear power plants has resulted in its modification to deal with time-dependent issues unique to that specific use case [11]. In particular, the exploration of core damage frequency as a surrogate measure to reach particular safety goals is of interest when discussing time-varying risk mitigation methods. This method focuses on whether or not changes to a reactor are allowed by evaluating the frequency of reactor core damage.

1.6 Related Physics-of-Failure and Parametric Analysis Methods

Despite the limitations of the previously discussed methods, engineers do have various tools to evaluate time-dependent failure probabilities in a variety of contexts—however, the usefulness of these methods to an engineering team analyzing time-varying external initiating events is debatable. One such discipline is physics-of-failure mathematics. Physics-of-failure mathematics represents the identification and analysis of the physical causes for the failure of a particular component and then modeling the resulting data to develop a probability density function along a system's lifetime [8]. While this approach is useful for looking at the probabilities of failure that happen within any particular component, the approach does not contain a methodology for design teams to act on the data. It is primarily a statistical method, not a design method.

Another methodology used to determine risk of failure is parametric analysis which is a statistical technique used when the unknown parameters of a particular component's longevity are populated with random values and a distribution is made of the resultant variables. This approach has many implementations, and has been integrated both in PRA analysis as well as in time-dependent probabilities of failure [5, 2]. However, there are certain issues with this approach—parametric analyses require the population of a data set to be developed through a tool such as Monte Carlo analysis, which can lead to a high computational expense when the methodology is applied to a more complex functional framework.

2 Methodology

TBFFE is a risk quantification method used to analyze complex, cyber physical systems during systems engineering design. This method focuses primarily in early systems engineering design where systems have the opportunity for significant configuration changes with minimal cost (both monetary and

development time). The goal of this method is to inform the system designer and systems engineer on predominate risks that may be realized during the system's life cycle, and thus, the method systematically analyzes all known or foreseeable risks to the system.

The results of TBFFE can be used to enhance a system design by reducing its risk of failure. The type of results produced by TBFFE include the systems' functional risk, which combines failure probability with the loss in functional health, tied to an initiating event. The results are presented across time (e.g., per month in the following case study) to represent heightened risk during specific time periods. An example of solutions to an unacceptable risk could include a design configuration change where the failure propagation has a behavior that renders the system less susceptible to the specific initiating events being analyzed.

A key difference from other risk methods surveyed earlier in this paper is that TBFFE uses discrete time-based failure probabilities with short time scales to more accurately model the external initiating events caused by natural environmental cycles and related factors. Typical failure probabilities are modeled on an annual basis; however, we analyze failure probabilities either monthly (as in the case study), daily, or even hourly. The discretization depends on the fidelity of the data used to build the probability values. For example, when assessing the risk of failures due to storms, failure probabilities would depend on the occurrence of storms in the local environment. If data is recorded in storms per month, the discretized failure probabilities will be monthly.

TBFFE is a process-oriented methodology that has several well-defined steps. These steps are meant to guide a design team from a basic functional model to a complete time-dependent representation of all potential points of failure that can affect a system.

2.1 Step 1: Functional Model Creation

The first step is to create a functional model based on the initial system architecture. The initial system architecture is usually a body of work developed by the system design team or a system architect and ranges across design requirements, sketches, blueprints, flowcharts, reliability block diagrams, as well as piping and instrumentation diagrams. In the absence of such existing work, an experienced system design team might instead opt to generate a native FBED [8] from scratch based on the desired system inputs and outputs given to the team.

2.2 Step 2: Initiating Event Identification

After creating a functional model, the team must note potential initiating events that may cause a failure. Finding initiating events in TBFFE is similar to the method used in PRA. For TBFFE the, designers are encouraged to consider external and internal initiating events. External events are those which originate outside the system, such as weather or debris. After compiling a list of external events, designers then go through the system and identify potential internal initiating events, such as mechanical wear, fire, internal flooding, or an electrical bus failure.

2.3 Step 3: Time-Dependent Initiating Event Identification

After developing a list of initiating events, the design team then classifies each event as time-dependent or independent. Those events which are based on seasonal phenomena, weather events, or events of variable strength such as storms are considered time-dependent.

For each time-dependent initiating event, there must be a particular profile to how the probability of the initiating event occurring increases or decreases over the course of a year (or other time increment that is normally analyzed across for the specific system in question). Each initiating event should be analyzed for how the probability of the initiating event occurring increases or decreases. A practitioner can choose to model certain initiating events either through a continuous function or through a discretized, step-wise function. Contin-

uous functions best serve events like storms where there is an identifiable period of peak intensity followed by a gradual drop-off. Typically, a systems design team is limited by the discretization of available data. For instance, engineers may have access to thorough meteorological data for their region that covers several days, or they may only know that storms occur more frequently over a particular range of months out of the year. Another important aspect to cover is how a system's probability of failure is affected by long-term or short-term forecasts. As an example, the engineering team may know that a seasonally-affected failure is only possible during certain hours of the day (such as the position of the sun affecting certain sensors only certain months of the year). As a rule of thumb, systems design teams are encouraged to account for short-term forecasts if they represent a change in probability greater than a standard deviation from their given probability of risk. Changes of less than one standard deviation will likely be inconsequential as compared to other factors within the risk analysis during the conceptual design process such as design, model, and data uncertainty. The result of the data acquired by the engineers will be similar to a Bayesian statistical model, but dependent on time.

In the TBFFE method, the design team is presumed to have existing probability of failure (per year or unit of time used for the particular system) as well as more discrete and detailed data for initiating events. The existing probability of failure divided by the unit of time for the overlay data is the baseline probability. If a design team has a yearly probability of failure by storm, and monthly values for frequency of storms in their region, then the design team will divide the yearly probability by twelve. The probability data that the design team has will then resolve itself into a function that shows frequency over time. In the case of the storm example, the systems design team will be able to chart the per-month frequency of storms over the year. As a verification step, the overlay data can be scaled such that, when combined, its value equals the existing yearly frequency of occurrence of the initiating event.

Step 4: Analyze Failure Propagation in the System

Once the various initiating events have been given a time-dependent profile, a probability of failure for each function within the model can be constructed. Both time-dependent and time-independent initiating events represent causes of failure that can map to particular functions. The systems design team can assign various causes of failure to individual functions. A function's probability of failure is the OR probability of any particular event occurring. By calculating the function's probability of failure at each time step, the design team develops a time-dependent failure profile for each function. Doing this for all functions allows the design team to have a FFIP model that the team can look at through a temporal lens, which allows the team to identify peak risks for particular functions.

Step 5: Design Iteration or Retrofit the System

By analyzing the various probabilities of failure present at particular time steps, the design team can begin to optimize the system design from a risk-of-failure perspective. Starting with functions crucial to the system's operation, the designers can look at local maxima of failure probabilities for a given function. Functions that exhibit the highest risk across any period of time can then be marked as at-risk. By identifying the initiating event or events responsible for this heightened state of risk, systems designers can focus on mitigating the probability of those initiating events happening, potentially by including specific functions or components that are used specifically in times of heightened risk. Starting with the highest risk functions, designers can continue to mitigate risks within the constraints of time, complexity, cost, etc. Once optimizations are complete, they can then iterate through the previous step to compare how their design has improved.

3 Case Study

In this section, we present a case study that demonstrates TBFFE and its capabilities. A representative example was created of the potential applications in a nuclear power context specifically for this paper. Note that we have intentionally fictionalized probability data and plant design, and explicitly do not recommend using the results of this case study in a real-world application. The case study is demonstrative of the method and is intentionally not directly applicable to a specific nuclear power plant. In this example, a new nuclear power plant somewhere on the East coast of the United States is being designed. The engineers are working on designing a spent fuel pool where spent fuel rods will be housed until the rods are cool enough for dry cask storage and disposal. Consequently, the fuel pool's main purpose is to cycle hot water to a system of heat exchangers to continuously maintain the temperature of the water in the pool at acceptable levels. The region the plant is being constructed in is prone to stormy weather, as well as seasonal algae blooms. The plant is planned to have one internal loop of water exchanging heat to the ocean. The design team decides to utilize TBFFE to anticipate and mitigate time-variant risks due to these unique conditions.

3.1 Step 1: Functional Model Creation

The systems engineering team first creates a functional model (see Figure 1) using the FBED functional modeling method. This functional model represents the baseline design prior to any iteration or redesign. At this stage, the team has already decided that they wanted to include multiple redundant systems; three sets of motors that power three sets of pumps that move the water in the primary pool, and three different heat exchangers are available to remove heat from the pool and route the heat to the ocean. The pumps are designed to begin operation one after the other, in the event of failure, while the heat exchangers are designed such that any one of the heat exchangers can transfer heat efficiently enough to keep the system

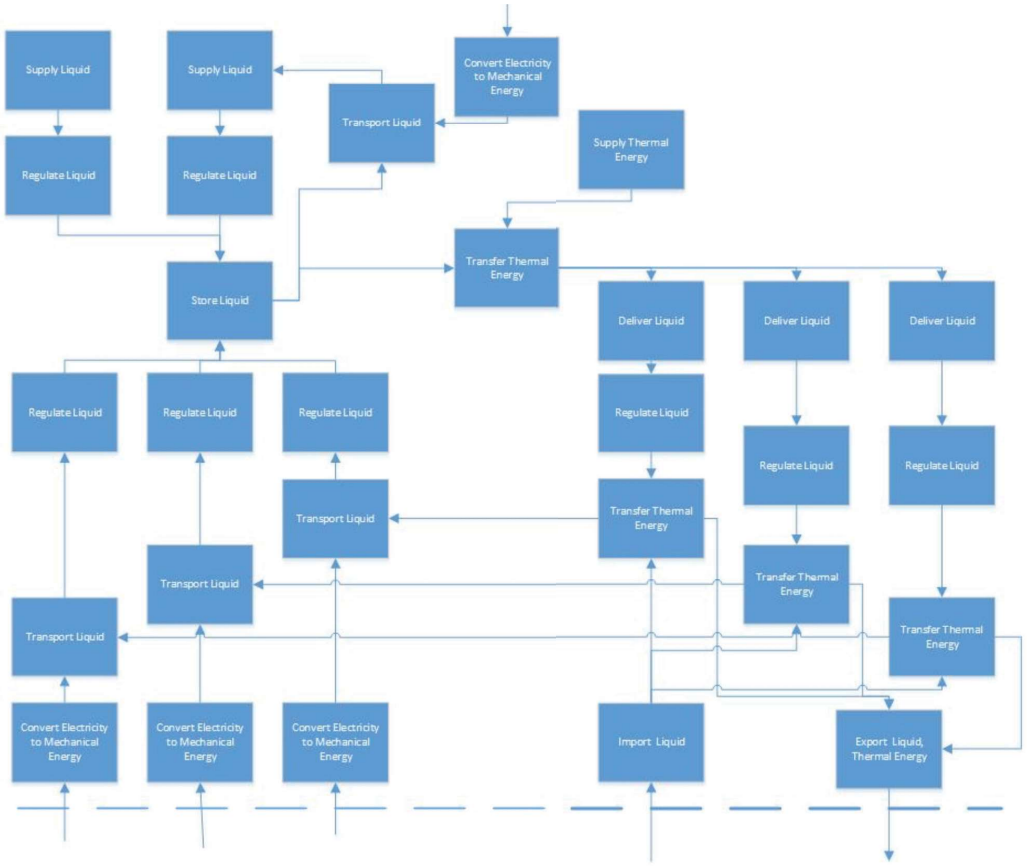

Figure 1. Functional model of a spent fuel pool.

operating nominally.

Initiating Event	Prob/Year
IE_MechanicalFailureCondenser	0.003
IE_Algae	0.004
IE_MechanicalFailureValve	0.001
IE_MechanicalFailurePump	0.005
IE_MechanicalFailureMotor	0.002
IE_Storm1	0.003
IE_MechanicalFailureValve	0.001
IE_MechanicalFailurePipe	0.003
IE_Storm2	0.002
IE_MechanicalFailureTank	0.0005

Table 1. List of initiating events for spent fuel pool used in the case study. Note that items that are italicized are time variant in their probabilities. The initiating events are presented here as an averaged yearly probability statistic. See Tables 2 and 3 for further details on these events.

3.2 Step 2: Initiating Event Identification

Once the functional model is complete, the systems engineers next consider potential initiating events and research the probabilities of occurrence until the systems engineers have a list of initiating events they feel is complete (see Table 1). The systems engineers first identify external failures like electrical storms shorting out motors or algae blooms clogging up the intake ocean water to the heat exchangers. Next identified are the internal failures such as mechanical wear within the machines.

3.3 Step 3: Time-Dependent Initiating Event Identification

Once the systems design team has a list of potential initiating events, the team can go through each initiating event and classify them as either time-variant or time-invariant. The team quickly recognized that algae blooms, electrical failures due to storms affecting the pumps, and heat exchanger failures due to storms affecting the intake of secondary water as significant initiating events that have a time-varying probability of occurrence. The reason for this is simple: both algae blooms and storms are events that occur seasonally, with significant change in event frequency occurring depending on the month.

Having identified which initiating events are time-dependent, the team next determines what data is available to better characterize the identified initiating events from a time basis. At this point, the team has finished researching initiating events and are able to use yearly probabilities of occurrence for all of the initiating events, as well as on-demand failure probabilities for all components. To acquire more detailed information on the behavior of the spent fuel cooling pool system, the team realizes that a monthly time step is appropriate for the external initiating event analysis. For storms, the systems engineering team determines the monthly number of storms that have historically occurred in the area where the nuclear reactor and its spent fuel cooling pool will be built. On the other hand, when researching the propagation of the algae the design team knows to be problematic, the team is restricted to data from marine biologists that forecast algae blooms to be most prevalent in the months of July to October and otherwise not present in the area. Knowing this, the design team creates a set of Boolean values corresponding to each month. Knowing the yearly probability of failure for the heat exchangers being installed for the spent fuel cooling pool due to storms and algae (as well as pumps failing due to storms), the team is able to develop the information found in Tables 2 and 3 to calculate the monthly probability of each initiating event occurring. Specifically, the annual failure rate for algae is evenly distributed across the months of May to October. Similarly, the annual failure rate for storms is proportionately distributed across the year based on the number of storms in each month (Storm for January: 0.03 x 10/526 = 5.7E–05 fails/month).

3.4 Step 4: Analyze Failure Propagation in the System

The team then generates cutsets based on the functional model of what possible failures could

occur based on the propagation of certain components failing when called upon to function. Table 4 shows the yearly probabilities of failure for ten generated example cutsets. Of the ten cutsets, four involve time-variant initiating events. Table 5 is then generated to track the monthly probabilities of failure. Tables 4 and 5 are generated based on the probability of the component failures required to cause the system to fail.

Available Monthly Frequencies

Month	Storm	Algae
1	10	0
2	5	0
3	7	0
4	12	0
5	22	1
6	35	1
7	77	1
8	110	1
9	96	1
10	80	1
11	50	0
12	22	0

Table 2. Monthly frequency values for storms as well as yes/no values for algae presence in the oceans. This data is utilized by the engineering team to produced scaled monthly probabilities, shown in Table 3.

Monthly Probabilities (probability of failure per month)

Month	Storm	Algae
1	5.7E-05	0
2	2.85E-05	0
3	3.99E-05	0
4	6.84E-05	0
5	0.000125	0.000667
6	0.0002	0.000667
7	0.000439	0.000667
8	0.000627	0.000667
9	0.000548	0.000667
10	0.000456	0.000667
11	0.000285	0
12	0.000125	0
Total	**0.003**	**0.004**

Table 3. Monthly Initiating Event Probabilities of Occurrence.

3.5 Step 5: Design Iteration or Retrofit the System

Based on analyzing the available data, the design team notices some statistically significant spikes in the probability of failure. For example, the team discovers from the cutsets that the probability of heat exchanger failure due to storms is highest in August, as is the probability of an electrical failure. Consequently, the team realizes that the system could be redesigned to mitigate the risk of failure during those months. For example, the team may decide that from July to October, the system could use a cooling pond rather than directly using the ocean to prevent both algae blooms and flotsam created by storms from clogging up the heat exchangers. Similarly, the design team realizes that backup generators could be kept on hot standby during the high risk months to lessen the risk of an outage caused by electrical storms. Beyond these specific seasonal improvements, the team also notices that an emergency cooling water pipe could be implemented that goes from the water tanks to the secondary loop to ensure that heat removal can continue in the case of an inlet water pipe clog. From there, the team is able to

Available Monthly Frequencies

Cutset No.	Prob (freq)/ year	Cutset
1	4.00E-03	IE_Algae, Import Liquid, Transfer Thermal Energy, Transfer Thermal Energy, Transfer Thermal Energy, Export Liquid/Thermal Energy
2	3.00E-09	IE_Storm1, Transfer Thermal Energy Transfer Thermal Energy, Transfer Thermal Energy, Export Liquid/Thermal Energy
3	1.80E-08	IE_Storm2, Convert Mechanical Energy to Electrical Energy, Transport Liquid, Regulate Liquid, Transport Liquid, Regulate Liquid, Transport Liquid, Regulate Liquid, Store Liquid, Transfer Thermal Energy
4	3.20E-08	IE_MechanicalFailureMotor, Convert Mechanical Energy to Electricity, Convert Mechanical Energy to Electricity, Convert Mechanical Energy to Electricity, Transport Liquid, Transport Liquid, Transport Liquid, Store Liquid, Transfer Thermal Energy
5	6.00E-08	IE_MechanicalFailurePump, Transport Liquid, Regulate Liquid, Convert Mechanical Energy to Electricity, Transport Liquid, Regulate Liquid, Transport Liquid, Regulate Liquid, Store Liquid, Transfer Thermal Energy
6	1.60E-08	IE_MechanicalFailureValve, Regulate Liquid, Convert Mechanical Energy to Electricity, Transport Liquid, Regulate Liquid, Convert Mechanical Energy to Electricity, Transport Liquid, Regulate Liquid, Store Liquid, Transfer Thermal Energy
7	1.00E-09	IE_MechanicalFailureValve, Regulate Liquid, Deliver Liquid, Transfer Thermal Energy, Deliver Liquid, Transfer Thermal Energy, Transfer Thermal Energy
8	3.00E-09	IE_MechanicalFailurePipe, Deliver Liquid, Transfer Thermal Energy, Deliver Liquid, Transfer Thermal Energy, Deliver Liquid, Transfer Thermal Energy
9	6.00E-09	IE_MechanicalFailureExchangers, Transfer Thermal Energy, Regulate Liquid, Deliver Liquid, Transfer Thermal Energy, Deliver Liquid, Transfer Thermal Energy
10	6.00E-09	IE_Storm1, Transfer Thermal Energy, Deliver Liquid, Transfer Thermal Energy, Regulate Liquid, Deliver Liquid, Transfer Thermal Energy

Table 4. Cutsets for yearly failure probabilities.

Cutset 1

Month	Monthly Probability
1	0
2	0
3	0
4	0
5	6.67E-04
6	6.67E-04
7	6.67E-04
8	6.67E-04
9	6.67E-04
10	6.67E-04
11	0
12	0
Yearly Probability	**4.00E-03**

Table 5. Monthly probability of algae bloom creating a failure event as described in Cutset 1.

create new, lowered monthly probability profiles and iterate further through the system design to achieve a desired risk profile for the system.

4 Results & Discussion

To better demonstrate the full capabilities of the TBFFE method when applied to iterative design, the authors of this paper developed a tool based on a Universal Modeling Language (UML) [10] backend that does the work of generating cutsets automatically based on the functional model and using the TBFFE method. By implementing functional modeling in UML, defining critical functions that cannot be interrupted, and providing a per-month list of the probability of initiating events, the tool runs through all cutsets that result in the failure of the system and then calculates the overall risk of

failure applied on the given timescale. This tool enables designers to use TBFFE even in scenarios that involve rapid iteration. In the spent fuel pool case study, the critical function was defined as the transfer of heat out of the water in the spent fuel pool. Figure 2 shows the resultant UML functional model, and Figure 3 shows the results of the overall risk analysis.

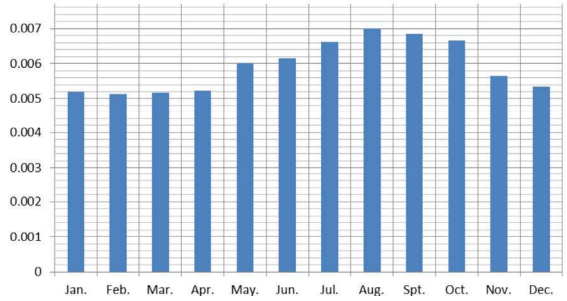

Figure 2. TBFFE Tool Implementation.

From these results, it is possible to see how the resolution in the risk of failure afforded by creating monthly probabilities is useful to systems design teams. Risk profiles can spike depending on the month; however, sometimes yearly probabilities are all that is available to an engineering team in databases for nuclear power plant failure events, and seasonal occurrences like storms or algae blooms are often unique to a region. The systems design team is best served by fitting local data to yearly probabilities that might be otherwise useful to their facility.

By utilizing the UML-based tool, iterative designs can be performed as described in the previous section. Cutsets have been generated (similar to those found in Table 4 – the baseline design cutsets) on a modified functional model from the months of July to October that uses a cooling pond as a cooling water intake source. Based on this model, the peak of risk of system failure is reduced significantly—the probability of a failure is reduced by 10% in August, and consistent decreases along similar months are observed. Figure 4 displays the new set of probabilities—the new risk profile is significantly flatter, and showcases potential avenues that the design team can take to improve the risk profile of their fuel pool.

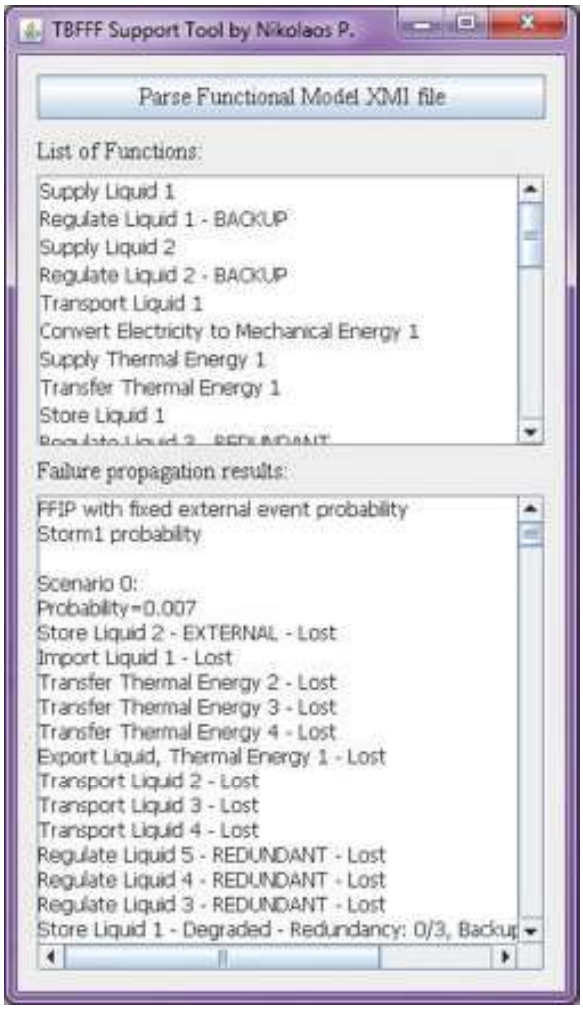

Figure 3. Probability of System Failure per Month.

From these results, the design team can note new avenues of development—increased risk of system failure occurs in May consistent with the heightened risk of the storm initiating event. The design team can then focus on mitigating that form of system failure by creating redundancies in the power supply such as waterproofing the motor system as well as potentially investigating redundant backup generators. The

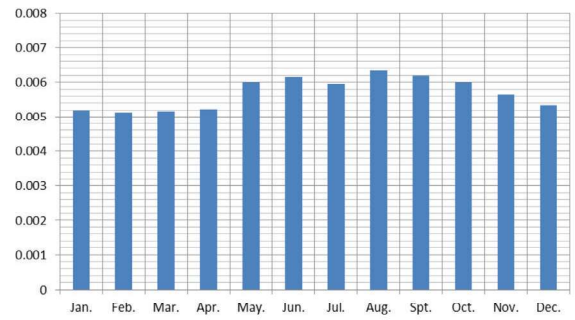

Figure 4. Probability of System Failure per Month After Iterative Design Using Insights Gained from TBFFE.

team may also run more TBFFE iterations and generate a new risk assessment based on the previously mentioned system design improvements with the aim of creating a very flat risk profile throughout the year. The iterative design potential is the main draw of TBFFE, permitting systems engineering teams, systems designers, and systems architects to rapidly identify and mitigate areas of concern for their systems that would go unnoticed without access to time-based failure evaluation methods.

The main benefit of time-dependent analysis of risk of system failure is that increased granularity of failure probabilities with respect to time over which the probabilities are analyzed allows engineers to mitigate risk in a more optimal way, thereby focusing on spending resources in times of heightened system failure risk. TBFFE allows practitioners to bring together risk data that operates on non-uniform timescales to create overall profiles of risk that provide insight which otherwise would be obscured by the commonly used yearly timescales of PRA and other risk analysis techniques.

One limitation of the TBFFE method is the need for more granular initiating event data as an input to the method. TBFFE is useful when the design team already knows that the system is going to be impacted by time-variant initiating events – in the example of the spent fuel cooling pool, the designers already knew that algae and storms had been problems in previous nuclear reactors and were able to account for this within their design by using the TBFFE method. TBFFE is a method best suited to characterizing known information with greater granularity—unknowns are harder for the system to deal with and frequently can be as opaque to the design team as they would be had they only used a method such as FFIP or PRA. An extension of this limitation is that TBFFE requires the design team to bring in data beyond what they might get from existing engineering databases to create distinct probability of occurrence data for initiating events. Depending on the initiating event, this may require assumptions on the part of the design team that possibly will not be borne out by reality.

By understanding these weaknesses, it becomes clear that TBFFE is best suited to those scenarios where designers wish to integrate data that is specific to their use case into a larger framework of existing probabilities in their system analysis. Examples include specific scenarios such as nuclear reactors or spacecraft, where there is a plurality of information available to an engineering team but where the details born of location or purpose are unique to a particular project.

References

1. German Aneiros-Perez and Philippe Vieu. Nonparametric time series prediction: A semi functional partial linear modeling. Journal of Multivariate Analysis, 99(5):834{857, 2008.
2. Corwin L Atwood. Parametric estimation of time-dependent failure rates for probabilistic risk assessment. Reliability engineering & system safety, 37(3):181{194, 1992.
3. US Nuclear Regulatory Commission et al. Regulatory Guide 1.174: An Approach for Using Probabilistic Risk Assessment in Risk-informed Decisions on Plant-specific Changes to the Licensing Basis. US Nuclear Regulatory Commission, Office of Nuclear Regulatory Research, 2002.
4. Warren Gilchrist. Modelling failure modes and effects analysis. International Journal of Quality & Reliability Management, 10(5), 1993.
5. PL Hall and JE Strutt. Probabilistic physics-of-failure models for component reliabilities using Monte carol simulation and weibull analysis: a parametric study. Reliability Engineering & System Safety, 80(3):233{242, 2003.
6. Julie Hirtz, Robert B Stone, Daniel A McAdams, Simon Szykman, and Kristin L Wood. A functional basis for engineering

design: reconciling and evolving previous efforts. Research in engineering Design, 13(2):65{82, 2002.

7. Ryan S Hutcheson, Daniel A McAdams, Robert B Stone, and Irem Y Tumer. A function-based methodology for analyzing critical events. In ASME 2006 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pages 1193{1204. American Society of Mechanical Engineers, 2006.
8. K Kimseng, M Hoit, N Tiwari, and M Pecht. Physics-of-failure assessment of a cruise control module. Microelectronics Reliability, 39(10):1423{1444, 1999.
9. Tolga Kurtoglu and Irem Y Tumer. FFIP: A framework for early assessment of functional failures in complex systems. In The International Conference on Engineering Design, ICED, volume 7, 2007.
10. Craig Larman and UML Applying. Patterns: An introduction to object-oriented analysis and design and iterative development. 2004.
11. Nuclear Regulatory Commission et al. Severe accident risks: an assessment for five us nuclear power plants. Technical report, Nuclear Regulatory Commission, 1991.
12. Bryan M O'Halloran, Nikolaos Papakonstantinou, and Douglas L Van Bossuyt. Modeling of function failure propagation across uncoupled systems. In 2015 Annual Reliability and Maintainability Symposium (RAMS), pages 1{6. IEEE, 2015.
13. Bryan M O'Halloran, Nikolaos Papakonstantinou, and Douglas L Van Bossuyt. Cable routing modeling in early system design to prevent cable failure propagation events. In 2016 Annual Reliability and Maintainability Symposium (RAMS), pages 1{6. IEEE, 2016.
14. Nikolaos Papakonstantinou, Markus Porthin, M O'Halloran, and L Van Bossuyt. A model-driven approach for incorporating human reliability analysis in early emergency operating procedure development. In 2016 Annual Reliability and Maintainability Symposium (RAMS), pages 1{6. IEEE, 2016.
15. N Siu. Risk assessment for dynamic systems: an overview. Reliability Engineering & System Safety, 43(1):43{73, 1994.
16. Michael Stamatelatos. Probabilistic risk assessment: What is it and why is it worth performing it? NASA Office of Safety and Mission Assurance, 4(05):00, 2000.
17. R.B. Stone, I.Y. Tumer, and M. Van Wie. The function-failure design method. Journal of Mechanical Design, 127(3):397{407, 2005.
18. Robert B Stone and Kristin L Wood. Development of a functional basis for design. Journal of Mechanical design, 122(4):359{370, 2000.
19. Rogier Woltjer and Erik Hollnagel. Functional modeling for risk assessment of automation in a changing air traffic management environment. In Proceedings of the 4th International Conference Working on Safety, volume 30, 2008.