

Light Water Reactor Sustainability Program

Cost Risk Analysis Framework Tool (CRAFT): An Integrated Risk Analysis Tool and its Application in an Industry Use Case



September 2018

U.S. Department of Energy Office of Nuclear Energy

DISCLAIMER

This information was prepared as an account of work sponsored by an agency of the U.S. Government. Neither the U.S. Government nor any agency thereof, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness, of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. References herein to any specific commercial product, process, or service by trade name, trade mark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the U.S. Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the U.S. Government or any agency thereof.

Cost Risk Analysis Framework Tool (CRAFT): An Integrated Risk Analysis Tool and its Application in an Industry Use Case

**D. Mandelli, C. Wang, S. Staples, R. Kunz, P. Zohner, C. Ritter,
C. Rabiti, A. Alfonsi, A. Mack, S. St Germain, C. Smith**

September 2018

**Prepared for the
U.S. Department of Energy
Office of Nuclear Energy**

EXECUTIVE SUMMARY

This report presents the initial application design of the Cost Risk Analysis Framework Tool (CRAFT). At the beginning, the concept of CRAFT was formulated from a need to perform economic risk analysis that can be quickly deployable through the nuclear industry to evaluate the risk associated to capital projects and to perform plant asset management. The design evolved into a framework that can integrate several forms of risk (e.g., not just economic, but also safety related) into a single analysis. In order to reduce the research, development and deployment time, we have decided to use the Risk Analysis Virtual ENvironment (RAVEN) statistical framework as a basis for the CRAFT architecture. In this report we summarize the most recent developments accomplished during the second part of FY18 for the two use cases of the Cost and Risk Categorization Applications path of the RISA project. These two use cases directly target plant health and plant capital SSC management. Regarding the plant health management use case, we have started to investigate methods to extract information from text-based data: text data mining. The Risk Informed Asset management is the main target of this report and several methods are here shown in order to solve the capital SSC replacement issue. This has been performed by presenting: several models designed to determine the effective cost through the lifetime of a capital SSCs, and the optimization algorithms that will be employed to determine the optimal replacement schedule of a given set of capital SSC. The interface between RAVEN and the PRA code SAPHIRE is the first step toward the integration of several plant risk models (e.g., economic and safety) and plant data (e.g., databases from the plant maintenance and diagnostics center) into a single analysis framework. In this respect, RAVEN is the ideal platform to connect several models and to manage data streaming among them.

CONTENTS

EXECUTIVE SUMMARY	iii
ACRONYMS.....	vii
1. INTRODUCTION	1
2. OBJECTIVES.....	1
3. RAVEN	2
3.1 Ensemble Modeling in RAVEN.....	4
4. CRAFT	5
4.1 Applications	8
4.2 Deployment Strategy.....	9
4.3 Example of CRAFT Analysis	9
4.3.1 PRA Data	11
4.3.2 Testing and Maintenance Procedures.....	11
4.3.3 Aging and Degradation Models	11
4.3.4 Plant Operation Data.....	12
4.3.5 SSC Monitoring Data.....	12
4.3.6 SSC RUL Models.....	12
4.3.7 Reliability, Availability and Maintainability Models	12
4.3.8 Plant Performance Risk Models.....	12
4.3.9 Cost Models	13
4.3.10 CRAFT Model	13
4.3.11 Results.....	14
5. INDUSTRY USE CASE: PLANT ASSET MANAGEMENET.....	16
5.1 CRAFT Model	17
5.1.1 Failure Models	17
5.1.2 Consequence Models	17
5.1.3 Cost Models	18
5.2 SSC Replacement Analysis Example.....	18
6. OPTIMIZATION ALGORITHMS	20
7. RAVEN INTERFACE WITH SAPHIRE	21
7.1.1 Time Dependent Probabilistic Risk Assessment.....	22
8. CONCLUSIONS AND FUTURE ACTIVITIES	29
REFERENCES	30
Appendix A - COST MODELS IN CONTINUOUS FORM.....	32
A.1 No Replacement Planned.....	32
A.2 Replacement Planned	33

Appendix B - COST MODELS IN DISCRETE FORM.....	35
B.1 No Replacement Planned.....	35
B.2 Replacement Planned.....	36
Appendix C - NPV DISTRIBUTIONS FOR THE REPLACEMENT USE CASE	38
Appendix D - RAVEN SAPHIRE INTERFACE	39
Appendix E - TEXT DATA MINING	41

FIGURES

Figure 1. RAVEN and CRAFT for RISA applications.....	2
Figure 2. RAVEN abstracted module scheme.	3
Figure 3. Example of an <i>EnsembleModel</i> constituted of 3 sequential sub-models.....	4
Figure 4. <i>EnsembleModel</i> resolving in a non-linear system of equations – Numerical iterations.	5
Figure 5. <i>EnsembleModel</i> data exchange.....	5
Figure 6. Safety related risk analysis (i.e., PRA).....	6
Figure 7. Structure of CRAFT for NPP economic analysis.....	7
Figure 8. NPP cost drivers.	7
Figure 9. Layout of the structure of CRAFT to perform safety, cost and economic analysis.....	8
Figure 10. Scheme of the MFW system.....	10
Figure 11. CRAFT model for the capital components replacement application.....	10
Figure 12. Event Tree description of the consequence model for the P1-P2 system.....	13
Figure 13. Failure probability for V1 and V2 plotted in term of CDF.....	14
Figure 14. Failure probability for SG (left) and P1 (right) plotted in term of CDF.....	15
Figure 15. Histogram of failure probability for P2.	15
Figure 16. Temporal profile of the probabilities associated to the Branches 1 (left) and 2 (right) of the consequence model for the P1-P2 system of Figure 12.....	15
Figure 17. Temporal profile of the cost models for Branch 1 of Figure 12 for the P1-P2 system (left) and for the SG model (right).....	16
Figure 18. Comparison of the NPV distributions for two cases: with (red) and without (blue) planned replacement.	19
Figure 19. Box plots for different SSC failure models: [0,10] (top left), [0,20] (top right), [0,30] (bottom left) and [0,40] (bottom right).....	20
Figure 20. Time-dependent safety related risk analysis with RAVEN and SAPHIRE.....	22
Figure 21. LOSP ET model.	23
Figure 22. CCS FT model.....	23
Figure 23. ECS FT model.....	24

Figure 24. Time dependent uncertainty analysis of ET outcome small-release.....	27
Figure 25. Time dependent uncertainty analysis of ET outcome large-release.	27
Figure 26. Time dependent uncertainty analysis of FT ECS.	28
Figure 27. Time dependent uncertainty analysis of FT CCS.....	28
Figure 28. Plot of plant costs for SSC no replacement case (continuous form).	32
Figure 29. Plot of plant costs for SSC replacement case (continuous form).	33
Figure 30. Plot of plant costs for SSC no replacement case (discrete form).....	35
Figure 31. Plot of plant costs for SSC replacement case (discrete form).....	36
Figure 32. NPV distributions assuming SSC RUL in [0, 10] for three replacement times (5, 10, 15 years).	38
Figure 33. NPV distributions assuming SSC RUL in [0, 20] for three replacement times (5, 10, 15 years).	38
Figure 34. NPV distributions assuming SSC RUL in [0, 30] for three replacement times (5, 10, 15 years).	38
Figure 35. NPV distributions assuming SSC RUL in [0, 40] for three replacement times (5, 10, 15 years).	38
Figure 36. Email data set.	42
Figure 37. Logistic regression coefficients for the email data set.....	43

TABLES

Table 1. PRA data for the test problem.....	11
Table 2. Testing and maintenance data for the test problem.....	11
Table 3. Aging and degradation models for the test problem.	11
Table 4. RUL models for the test problem.....	12
Table 5. Reliability models for the test problem.....	12
Table 6. Plant risk models for the test problem.	13
Table 7. Cost models for the test problem.	13
Table 8. CRAFT reliability models for the test problem.	14
Table 9. CRAFT plant performance risk models for the test problem.....	14
Table 10. CRAFT cost models for the test problem.	14
Table 11. List of possible sources of plant savings.....	18
Table 12. List of possible sources of plant costs.....	18
Table 13. Degrees of freedom initially considered for the replacement problem.....	18
Table 14. Basic failure model employed in ECS/CCS FTs.	24
Table 15. Basic events employed in CCS FT.	25
Table 16. Basic events employed in ECS FT.....	25

ACRONYMS

ADDM	Alternating Direction Method of Multipliers
CCS	Containment Cooling System
CDF	Core Damage Frequency
CRAFT	Cost Risk Analysis Framework Tool
DOE	Department of Energy
ECS	Emergency Cooling System
EPRI	Electric Power Research Institute
ET	Event Tree
FT	Fault Tree
ILCM	Integrated Life Cycle Management
INL	Idaho National Laboratory
LHS	Latin Hypercube System
JH	Jensen Hughes
LOOP	Loss Of Offsite Power
LWRS	Light Water Reactor Sustainability
MCS	Minimal Cut Set
MFW	Main Feedwater System
MOV	Motor Operated Valve
MSO	Maintenance Strategy Optimizer
M&D	Maintenance and Diagnostics
NEI	Nuclear Energy Institute
NPP	Nuclear Power Plant
NPV	Net Present Value
NRC	Nuclear Regulatory Commission
PDEs	Partial Differential Equations
PRA	Probabilistic Risk Assessment
PWR	Pressurized Water Reactor
RAVEN	Risk Analysis Virtual ENvironment
RCP	Reactor Coolant Pump
RISA	Risk Informed Safety Analysis
R&D	Research and Development
RISMC	Risk-Informed Safety Margin Characterization
RUL	Remaining Useful Life
SG	Steam Generator
SLR	Second License Renewal
SSCs	Structures, Systems, and Components
SQA	Software Quality Assurance
T&M	Testing and Maintenance
UQ	Uncertainty Quantification
VBM	Value Based Maintenance

Cost Risk Analysis Framework Tool (CRAFT): An Integrated Risk Analysis Tool and its Application in an Industry Use Case

1. INTRODUCTION

This report presents the initial concept and design of the architecture for the Cost Risk Analysis Framework Tool (CRAFT). The initial request was formulated from a need to perform economic risk analysis that can be quickly deployable through the nuclear industry to evaluate the risk associated to capital projects and asset management.

In the past few years, the Light Water Reactor Sustainability (LWRS) Risk-Informed Safety Margins Characterization (RISMC) [1] project developed tools, methods and algorithms that can be employed to perform Probabilistic Risk Analysis (PRA) in order to analyze the safety risk associated to Nuclear Power Plants (NPPs). These tools, methods and algorithms employ data mining and machine learning models which greatly advance state-of-the-art PRA practices. During FY18, along with the restructuring of the LWRS program, RISMC became RISA (Risk Informed Safety Analysis) with a more specific goal of directly targeting industry relevant applications [2]. As part of this new direction, special attention has been given to economic risk analysis that can be applied to existing NPPs. This started with the idea of developing a risk analysis framework that can encompass several types of performance shortfalls, or “risks”, in a single analysis: CRAFT.

Since the LWRS had invested in PRA methods and it has supported the Risk Analysis Virtual Environment (RAVEN) statistical framework [3], we are employing RAVEN as a base for CRAFT and develop those particular models/algorithms required for CRAFT within RAVEN itself.

In [4] we have described the most relevant elements of CRAFT and how the integrated risk analysis can be set and actually performed. While a large number of the CRAFT elements are already available within RAVEN, it is expected that in the following years more elements will be developed and tested as the use cases mature.

In this report we summarize the most recent developments accomplished during the second part of FY18 for the two use cases of the Cost and Risk Categorization Applications path of the RISA project. These two use cases directly target plant health and the plant asset management. Regarding the plant health management use case, we have started to investigate methods to extract information from text-based data: text data mining (see Appendix E). This capability will be employed when it will be needed to assess SSC health from NPP incident reports.

The risk informed plant asset management is the main target of this report and several methods have been shown in order to solve the capital SSC replacement issue. This has been accomplished by presenting: several models designed to determine the effective cost through the lifetime of a capital SSC, and, the optimization algorithms that will be employed to determine the optimal replacement schedule of a given set of capital SSC. The interface between RAVEN and the PRA code SAPHIRE [5] is the first step toward the integration of several plant risk models (e.g., economic and safety) and plant data (e.g., databases from the plant maintenance and diagnostics center) into a single analysis framework. In this respect, RAVEN is the ideal platform to connect several models and to manage data streaming among them.

2. OBJECTIVES

The main objective of CRAFT targets the determination, the measure and the management of risk under uncertainties. The main CRAFT design requirements/capabilities can be summarized as follows:

- 1) Integrate safety and economic risk

- 2) Integrate heterogeneous models such as: component ageing, reliability and cost models
- 3) Implicit consideration of model/data uncertainties
- 4) Provide analysis and optimization engines
- 5) Explicit consideration of the time variable in the analysis

In this report we show how some of these requirements/capabilities can be reached by employing RAVEN as the main statistical platform.

From a systematic perspective, RAVEN and CRAFT can be considered as two entities strongly coupled together (see Figure 1). RAVEN [3] (see Section 3) provides all statistical analysis methods such as optimization, uncertainty propagation and data analysis methods/algorithms. CRAFT (see Section 4) instead contains all risk models such as the safety or economic ones. As mentioned in Section 1 and Section 4, these models are correlated to each other and, thus, they cannot be considered separately in the overall analysis. In Section 3 we show how it is possible to link models together in order to account for their coupling by employing the RAVEN *EnsembleModel* feature. Sections 4.3 and 5 show how the CRAFT models can be integrated and which type of data will be employed to perform a typical CRAFT analysis. Section 6 presents an optimization algorithm that can be employed to solve the plant asset management use case in order to determine the best capital SSC replacement strategy. Finally, Section 7 shows how the integration between the safety risk model (i.e., SAPHIRE) has been completed by creating an interface between RAVEN and SAPHIRE. This feature will allow the user to link component ageing models to SAPHIRE in order to determine the temporal evolution of the plant safety parameters.

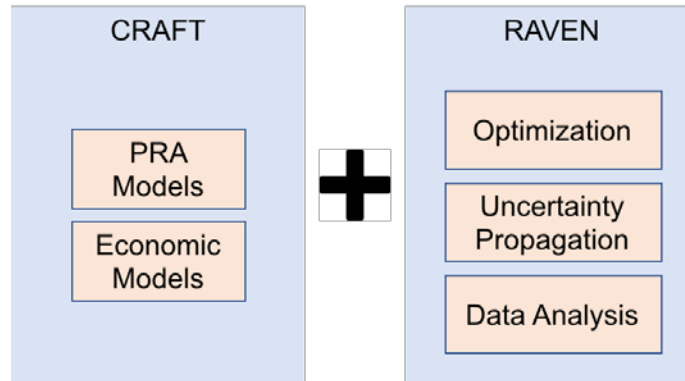


Figure 1. RAVEN and CRAFT for RISA applications.

3. RAVEN

RAVEN [3] is a flexible and multi-purpose Uncertainty Quantification (UQ), regression analysis, PRA, data analysis and model optimization software. Depending on the tasks to be accomplished and on the probabilistic characterization of the problem, RAVEN perturbs (through Monte-Carlo [6], Latin Hypercube [7], reliability surface search [8] sampling methods) the response of the system by altering its parameters. The system is modeled by third party software (e.g., RELAP5-3D [9], MAAP5 [10]) and accessible to RAVEN either directly (software coupling) or indirectly (via input/output files). The data generated by the sampling process is analyzed using classical and more advanced data mining approaches. RAVEN also manages the parallel dispatching (i.e., both on desktop/workstation and large High-Performance Computing machines) of the software representing the physical model. RAVEN heavily relies on artificial intelligence algorithms to construct surrogate models of complex physical systems in order to perform UQ, reliability analysis (limit state surface) and parametric studies.

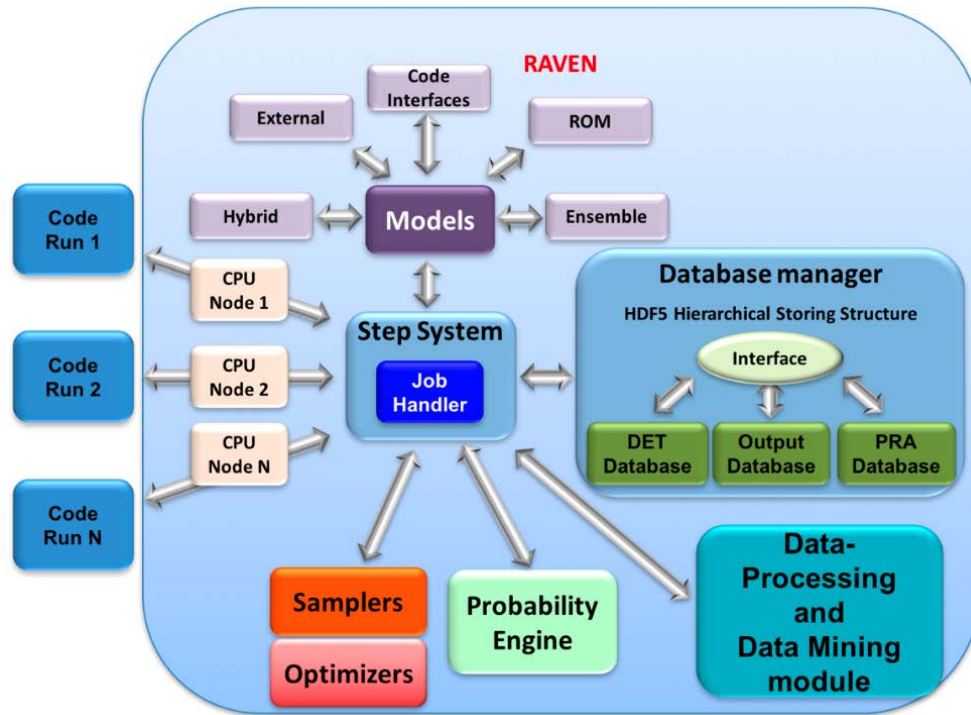


Figure 2. RAVEN abstracted module scheme.

RAVEN's scope is to provide a set of capabilities to build analysis flows based on UQ, PRA, optimization and data analysis techniques to be applied to any physical model(s). The main objective of RAVEN is to assist the engineer/user to:

- Identify the best design (on any physics/model) and its safety impact
- Estimate the likelihood of undesired outcomes (risk analysis)
- Identify main drivers/events to act on for reducing impact/consequences of anomalous dynamic behaviors of the system under analysis
- Construct analysis flows combining multiple physical models and analysis procedures

In other words, the RAVEN software can be employed to perform:

- Uncertainty Quantification
- Sensitivity Analysis / Regression Analysis
- Probabilistic Risk and Reliability Analysis (PRA)
- Data Mining Analysis
- Model Optimization

The RAVEN software employs several novel and unique techniques, based on extensive use of artificial intelligence algorithms, such as adaptive (smart) sampling [11], adaptive branching algorithms (Dynamic Event Tree [12]), time-dependent statistical analysis and data mining. The overall set of algorithms implemented in the RAVEN software are designed to handle highly non-linear systems, characterized by system response discontinuities and discrete variables. These capabilities are crucial for handling complex system models, such as

NPPs. For example, reliability surface analysis, as implemented in RAVEN, is unique and capable to handle non-linear, discontinuous systems, allowing for faster and more accurate assessing of failure risk for complex systems.

Among the different capabilities, RAVEN provides the unique functionality to combine any model (e.g. physical models, surrogate models, data analysis models, etc.) in a single entity (named Ensemble Model) where each model can feedback into others. In the Section 3.1, a more detailed description of this capability is reported.

3.1 Ensemble Modeling in RAVEN

In several cases multiple models need to be interfaced with each other since the initial conditions of some models are dependent on the outcomes of others. In order to solve this problem, RAVEN provides a model entity named *EnsembleModel* [21]. This class is able to assemble multiple models of other categories (i.e., Code, External Model, Reduced Order Models - ROM), identifying the input/output connections, and, consequentially the order of execution and which sub-models can be executed in parallel.

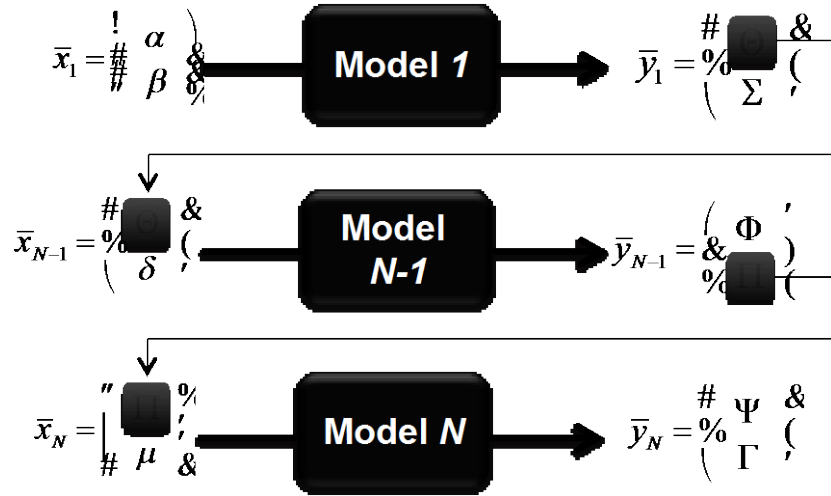


Figure 3. Example of an *EnsembleModel* constituted of 3 sequential sub-models.

Figure 3 shows an example of an *EnsembleModel* that is constituted of 3 sub-models (e.g., Reduced Order Models [13], Codes, or External Models) where:

- *Model 2* is connected with *Model 1* through the variable Θ (Model 1 output and Model 2 input);
- *Model 3* is connected with *Model 2* through the variable Π (Model 2 output and Model 3 input);

In this case, the *EnsembleModel* is going to drive the execution of all the sub-models in sequence, since each model (except the *Model 1*) is dependent on the outcomes of previously executed models.

In several cases, the input of a model depends on the output of another model whose input is the output of the initial model. In this situation, the system of equation is non-linear, and an iterative solution procedure needs to be employed. The *EnsembleModel* entity in RAVEN is able to detect the non-linearity of the sub-models' assembling and activate the non-linear solver: an iterative scheme. Figure 4 shows an example of when the *EnsembleModel* entity activates the iteration scheme, which ends when the residue norm (between an iteration and the other) falls below a certain input-defined tolerance.

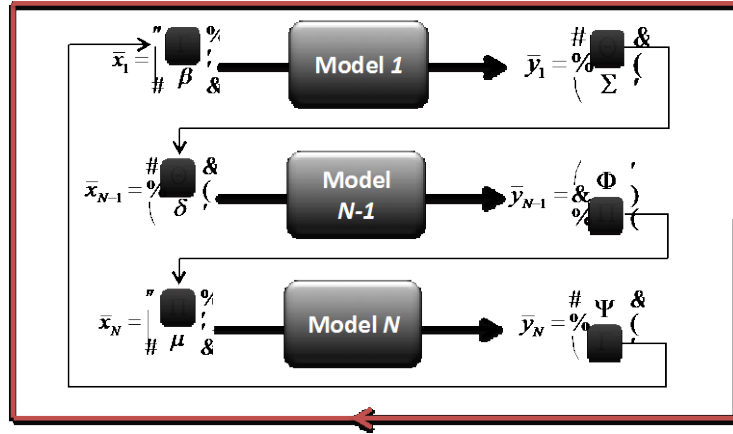


Figure 4. *EnsembleModel* resolving in a non-linear system of equations – Numerical iterations.

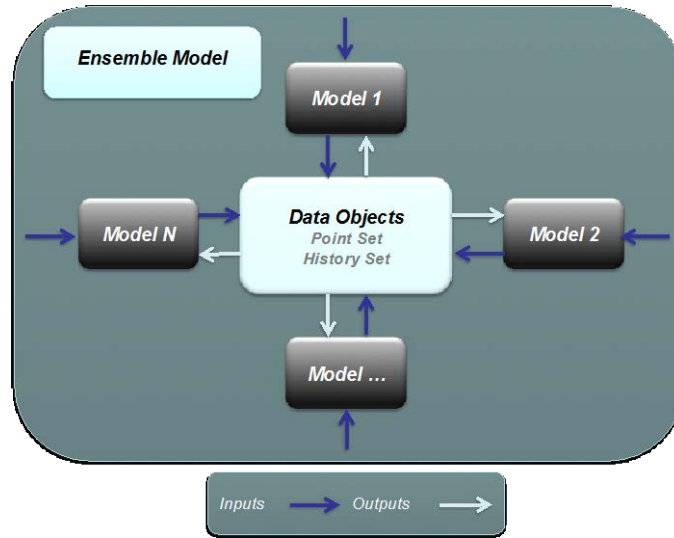


Figure 5. *EnsembleModel* data exchange.

In RAVEN all the models' outputs (e.g. whatever code output, etc.) are collected in internal containers (named *DataObjects*) that store time-series and input/output data relations in a standardized fashion (see Figure 5); in this way, the communication of the output information among different entities (i.e. Models) can be completely independent with respect to the particular type of output generated by a model. The *EnsembleModel* entity fully leverages this peculiarity in order to transfer the data from a Model to the other(s). Based on the Input/Output relations of each sub-models, the *EnsembleModel* entity constructs the order of their execution and, consequentially, the links among the different entities.

4. CRAFT

From a pragmatic perspective, the concept of combining multiple risk analyses is not dissimilar from any engineering problem that combines multiple physics models. In a classical multi-physics problem, multiple physics models are linked together in order to consider their mutual coupling (e.g., hydrodynamic model and heat transfer model). By employing a mathematical terminology applied to the analysis of Partial Differential Equations (PDEs), this coupling can occur either internally or at the initial/ boundary condition levels.

Similar argumentation can be extended to CRAFT-like (i.e., “multi-risk”) problems: multiple risk models can share initial and boundary conditions (typically in a time-dependent domain) or they can be internally coupled. If they are sharing initial/ boundary conditions, then the analysis can be performed by setting such conditions to each model and by running each model independently.

However, if the coupling between models is internal, then the analysis can pose several challenges from a computational perspective. Two situations are possible: loose or tight coupling. Assuming they are loosely coupled, each model can be solved individually, keeping the solutions for the other model fixed. After a solution is obtained for an individual model, it is transferred to other models that depend on it (see Figure 3), and solutions are obtained from those models (with recursion as necessary, see Figure 4). On the other hand, if the models are tightly coupled, a single system of equations is assembled and solved for the full set of coupled models. The nonlinear iterations operate on the full system of model simultaneously, taking into account the interactions between the models for the coupled physics in each iteration.

From the initial formulation of CRAFT, it is expected that most CRAFT models are expected to share initial/ boundary conditions or are to be loosely-coupled. Provided this problem definition, the design of CRAFT has started by considering:

- 1) The objectives and the design requirements
- 2) The structure of the RAVEN framework

The RAVEN framework has been designed to perform stochastic analysis (e.g., UQ and Dynamic PRA) applied to deterministic models (i.e., codes like RELAP5-3D). From a CRAFT perspective, what is needed is to: 1) include the actual models that include all forms of risk, 2) use RAVEN to propagate the uncertainties and introduce the proper aleatory models (i.e., sampling strategies), and 3) develop new optimization algorithms that can applied to specific use cases.

An example of safety related risk model can be a PRA code which determine the frequency (or probability) values of a set of consequences (e.g., Core Damage – CD – or Large Early Release) for a set of initiating events and provided a set of PRA data (i.e., failure rates and failure probabilities). Typically, when performing safety related risk analysis (Figure 6), Event-Tree (ET) and Fault-Tree (FT) based methods are employed to model accident progression while components failure data (or failure probabilities) are used to determine the frequency/probability of each accident sequence.

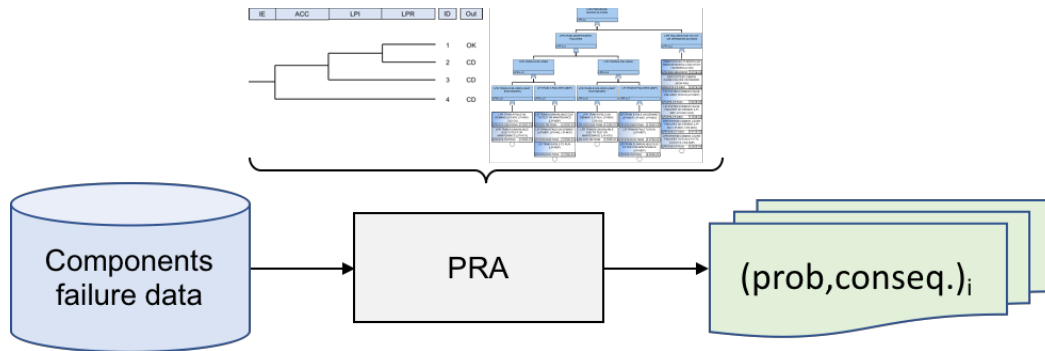


Figure 6. Safety related risk analysis (i.e., PRA).

In our applications, a CRAFT-like analysis combines safety related risk with the economic risk analysis (see Figure 7). The economic portion of the analysis includes information generated by two sub-modules:

- Cost Risk Analysis sub-module: it provides estimates of the plant cost drivers such as (see Figure 8):
 - Operation
 - Testing/Maintenance

- Capital
- Fuel
- Revenue Risk Analysis sub-module: it forecasts plant revenues and associated uncertainties provided energy demand and price

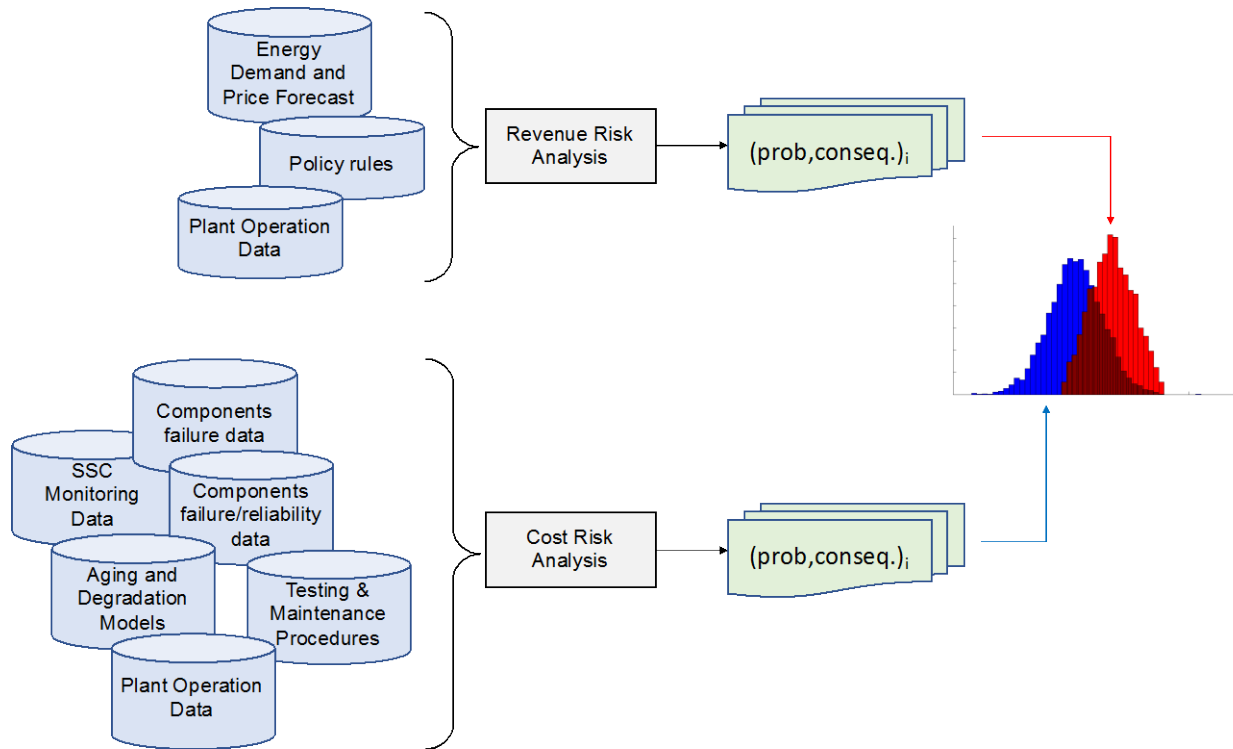


Figure 7. Structure of CRAFT for NPP economic analysis.

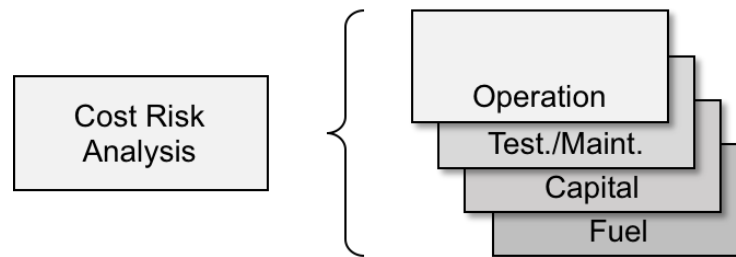


Figure 8. NPP cost drivers.

By combining the safety and the economic risk models in a single analysis, the resulting CRAFT model is shown in Figure 9. Note the following:

1. Safety analysis models (i.e., PRA) can be enhanced by employing Dynamic PRA [22]
2. The cost analysis combines the following information:
 - a. Time dependent PRA
 - b. Time dependent plant performance risk model

Note that while the first one evaluates the time dependent safety risk, the second one evaluates the actual performance of the plant in term of production performance. In other terms, the first one focuses on safety/risk related systems, the second one focuses on systems always involved in the energy production.

Cost models translates the set consequences (e.g., power reduction, unplanned shutdown) into costs.

3. The time dependent PRA and plant performance models requires information of the predicted failure rates (i.e., time dependent models), component monitoring data (diagnostic/prognostic) and predicted plant operation data.

The development and deployment strategies for these sub-modules may vary depending on the type of problem and on the type of capabilities requested; two possible strategies are the following:

- *Option 1:* Interface RAVEN with existing codes (such as SAPHIRE [5], CAFTA [14] and ILCM [15]). This approach is preferable: it would minimize deployment time since it would employ existing codes/models and RAVEN to perform data analysis management.
- *Option 2:* Develop new models within RAVEN specific to individual analysis needs. This approach would require more RD&D efforts, but such models could be tailored and optimized more precisely on the target application.

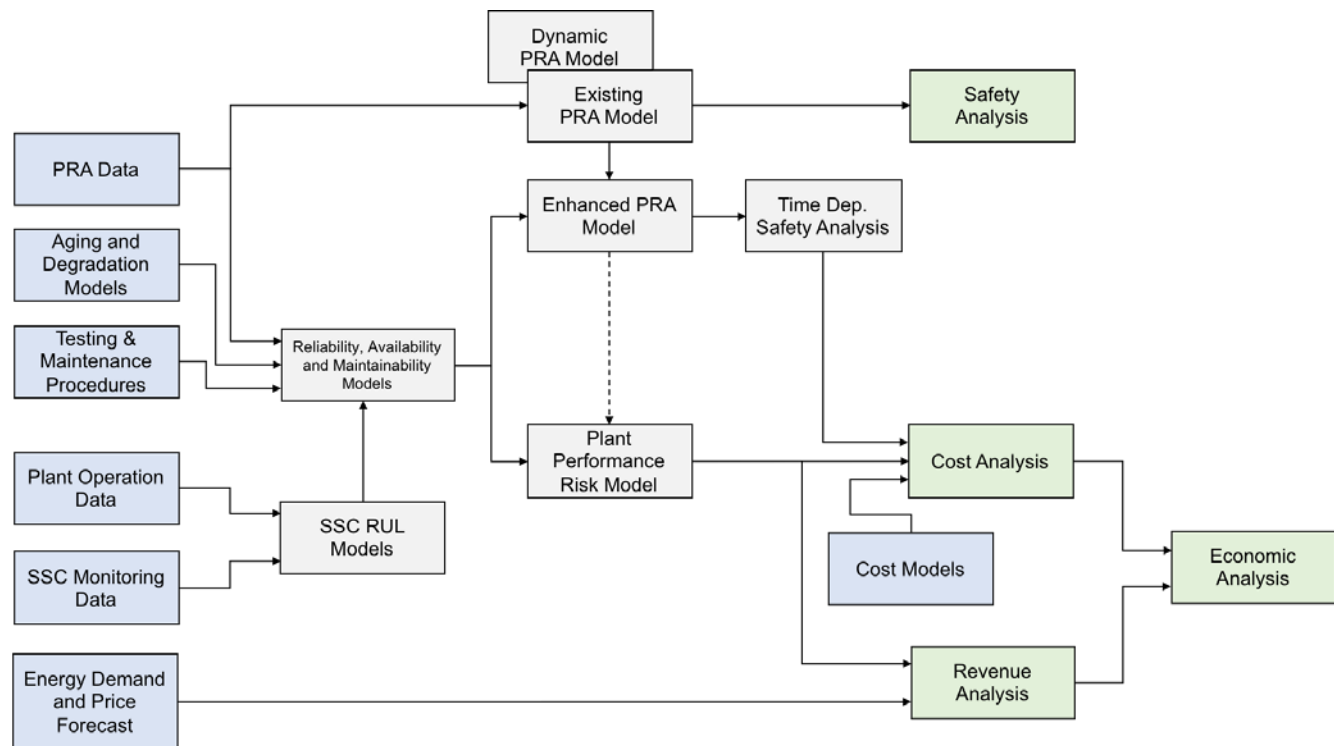


Figure 9. Layout of the structure of CRAFT to perform safety, cost and economic analysis.

4.1 Applications

Even though CRAFT applications initially targeted the nuclear power generation arena, the design of CRAFT has been kept independent of the application environment – it can be easily applied to the oil/gas, space/aeronautics, automotive, and in general to any high-risk industry/environment. The type of applications can be the following:

- 1) Evaluation of the overall risk associated to existing or planned capital projects
- 2) Identification and rank SSCs that are characterized by higher risk and/or higher uncertainties

- 3) Management and optimization of testing and maintenance operations and funds on the most important SSCs
- 4) Integrate large amounts of heterogenous data and models into a single framework

In particular the type of practical applications that we are targeting are the following

- Plant Asset Management
 - Can I reclassify system X to a lower quality class?
 - What is the optimum spare parts posture for system X?
 - Which systems have the highest cost risk of failure?
 - Which systems have the highest maintenance savings potential?
 - What is the true failure cost of component Y?
 - Support for the Nuclear Energy Institute (NEI) Value Based Maintenance (VBM) efficiency bulletin
- Plant Health Management
 - Is continuous online monitoring of system X cost effective?
 - Is investment in data analytics and prognostics for system X cost effective?
 - Is it cost effective to upgrade component Y with a more reliable or cheaper to maintain alternative?
 - Can I extend the maintenance interval on component Y?
 - If so, what is the optimum interval in terms of cost?

4.2 Deployment Strategy

The planned deployment strategy for the models developed in the past years for the RISMC project and for the ones that will be developed under the RISA project (e.g., the CRAFT models and algorithms presented in this report) is to include them in a single RAVEN plug-in. A RAVEN plug-in is a set of RAVEN objects designed for specific type of problems (e.g., risk analysis). Since RAVEN is an open-source code, also the plug-in for the RISMC-RISA projects will be open-source. The design of the RAVEN objects included in this plug-in will be designed so that users (e.g., the industry partners) will be able use their own proprietary information as input data for the models without having such information encoded in the model directly.

As RAVEN has its own repository and Software Quality Assurance (SQA) standards, also the CRAFT plugin will have an independent repository and same SQA standards. From an operation perspective, the user just needs to download RAVEN on his/her local machine, compile RAVEN and then download the RISMC-RISA plugin (see <https://github.com/idaholab/raven/wiki>).

4.3 Example of CRAFT Analysis

The considered example is a simplified Main Feed-Water (MFW) system of a PWR nuclear plant as shown in Figure 10. This system is designed to provide feed-water to the Steam Generator (SG) through the main pump P1 or, alternatively, the stand-by pump P2. Water flow is controlled by two valves V1 and V2.

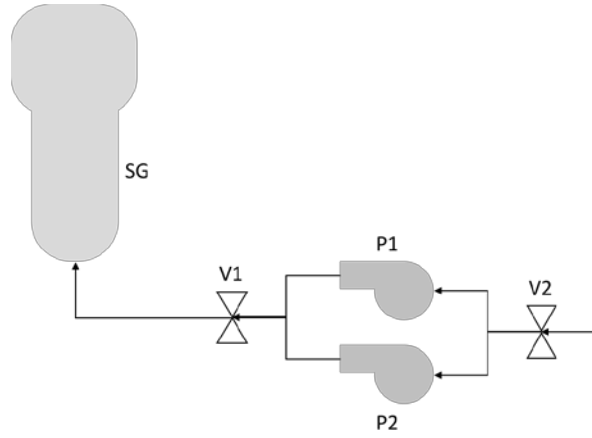


Figure 10. Scheme of the MFW system.

The scope of the CRAFT analysis is to evaluate the economic risk for this system when component ageing and test/maintenance are considered. The following models are included:

- Ageing is affecting valves V1 and V2; such ageing process is modeled as time dependent failure rates $\lambda_{V1}(t)$ and $\lambda_{V2}(t)$
- Main pump P1 and SG aging is modeled by a diagnostic/prognostic method that evaluates component Remaining Useful Life (RUL)
- Stand-by pump P2 is characterized by a constant failure rate λ_{P2} and periodic testing procedures

Given the nature of this example problem, the analysis is performed on a subdomain of the CRAFT framework (see Figure 9). This subdomain is pictured in Figure 11; in the following subsections, each block of Figure 11 is described.

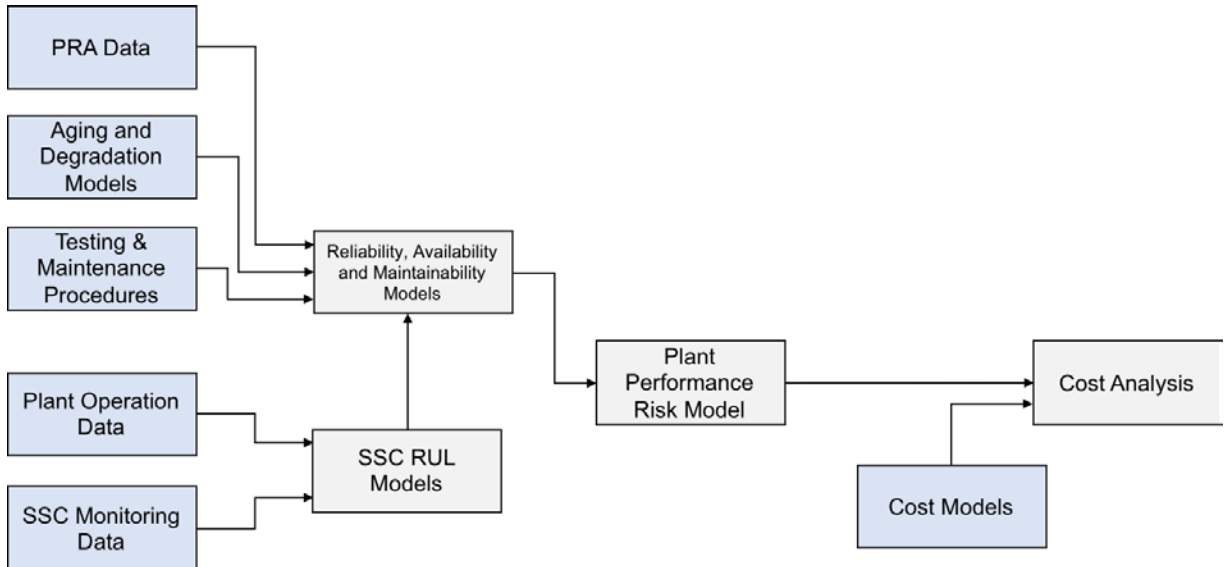


Figure 11. CRAFT model for the capital components replacement application.

4.3.1 PRA Data

This block contains the failure rates or failure probabilities of each component of the considered system that are included in the original PRA and are summarized in Table 1.

Table 1. PRA data for the test problem.

SSC	Failure model	Note	Values
V1	$f(t) = \lambda(t) \cdot \exp \left[- \int_0^t \lambda(\tau) d\tau \right]$	V1 and V2 are subject to ageing	See Section 4.3.2
V2			
P2	Unavailability u_{P2}	P2 subject of scheduled test and maintenance	See Section 4.3.2

4.3.2 Testing and Maintenance Procedures

This block contains information about the type of Testing and Maintenance (T&M) procedures employed for each component. In this case, pump P2 (i.e., stand-by pump) unavailability is modeled by the using the model indicated in [16] where the unavailability for this pump u_{P2} can determined as:

$$u_{P2} = \rho + \frac{\Delta}{T} + z\lambda T$$

where:

- T is the test interval time
- Δ is the downtime and it is a stochastic parameter as indicated in Table 2

Table 2. Testing and maintenance data for the test problem.

SSC	Testing and Maintenance	Note	Values
P2	Performed with interval T and downtime Δ	Δ uncertain parameter	Uniform [0.1,0.6]

4.3.3 Aging and Degradation Models

This block explicitly models how SSC failure rates or failure probabilities change in time due to aging and degradation; hence, they model $\lambda \rightarrow \lambda(t)$. Note that PRA models do not consider explicitly component aging or component degradation [17]. Example are the linear and Weibull component aging models. For the considered example problem, the aging models for the two valves are indicated in Table 3.

Table 3. Aging and degradation models for the test problem.

SSC	Failure model	Note	Values
V1	Failure rate $\lambda(t) = a + t \cdot b$	b is an uncertain parameter	$a=1.5E-5$ b is normal distributed ($\mu =2.5E-9$, $\sigma=2.E-9$)
V2	Failure rate $\log[\lambda(t)] = a + t \cdot b$	a and b are uncertain parameters	$a=-5.0$ b is normal distributed ($\mu =3.75E-5$, $\sigma=1.E-4$)

4.3.4 Plant Operation Data

This block contains the planned temporal profile of the operational status of the plant (e.g., 100% power, shut-down, etc.). Note that plant operational data is not considered in the test problem.

4.3.5 SSC Monitoring Data

This block contains all data generated by the SSC monitoring system. Note that monitoring data is not considered in the test problem.

4.3.6 SSC RUL Models

Provided the data from Sections 2.4 and 2.5, these models determine probabilistically the Remaining Useful Life (RUL) of SSCs. The output is a pdf of the SSCs predicted failure time (see Table 4 for P1 and SG).

Table 4. RUL models for the test problem.

SSC	Aging and Degradation Model	Note	Values
P1	Failure time beta distributed: $B(\alpha, \beta)$	α and β are uncertain parameters	α is uniform distributed [2.0, 2.5] β is uniform distributed [0.2, 0.3]
SG	Failure time normally distributed $N(\mu, \sigma)$	μ and σ are uncertain parameters. These parameters are obtained from the sampled values of α and β	α is uniform distributed [60.0, 70.0] β is uniform distributed [10.0, 15.0]

4.3.7 Reliability, Availability and Maintainability Models

This block generates temporal profiles of actual SSC failure probabilities. These models are summarized in Table 5 for V1, V2 and P2.

Table 5. Reliability models for the test problem.

SSC	Reliability Model	Note
V1	$R(t) = \exp \left[- \int_0^t \lambda(\tau) d\tau \right]$	-
V2		-
P2	Unavailability $u(t) = \rho + \frac{\Delta}{T} + z\lambda T$	Δ and T are uncertain parameters

4.3.8 Plant Performance Risk Models

Provided SSC failure, these models determine the type of consequences associated to such failures. Table 6 summarizes the consequences associated to the failure of the components involved in the test problem. In particular, Figure 12 describes the consequences associated to the two pumps (P1 and P2) using an ET formalism.

Table 6. Plant risk models for the test problem.

SSC	Consequence
V1, V2	When failed \Rightarrow Unplanned plant shut-down
SG	When failed \Rightarrow Unplanned plant shut-down
P1	When failed \Rightarrow Power reduced to 70%
P1 and P2	When failed \Rightarrow Unplanned plant shut-down + regulatory cost

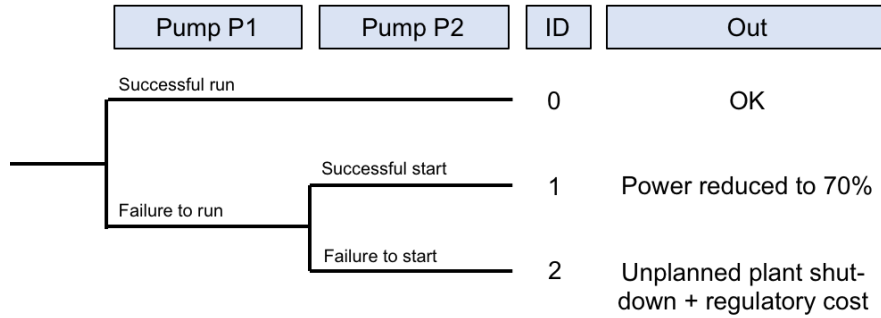


Figure 12. Event Tree description of the consequence model for the P1-P2 system.

4.3.9 Cost Models

These models quantify the economic consequences (in terms of costs) associated to the consequences determined from Section 4.3.8. Table 7 provides the details about the costs associated to the failure induced consequences for the components of the test problem. In our case, the plant costs caused by an unplanned Shut-Down (SD) have been calculated by determining the loss of the revenue of the plant provided the predicted energy price.

Table 7. Cost models for the test problem.

Consequence	Economic cost
Shut-down (V1 or V2)	# days in SD [10,30] Cost linear function of days in SD
Shut-down (SG)	# days in SD [30,60] Cost linear function of days in SD
Shut-down (P1 and P2)	# days in SD [7,21] Cost linear function of days in SD
Power reduced (P1 and P2)	# days in SD [2,7] Cost linear function of days in power reduced
Regulatory cost	Cost linear function of # days in SD

4.3.10 CRAFT Model

The prototype CRAFT model has been built in RAVEN by employing the *EnsembleModel* functionality (see Section 3.1). Tables 8, 9 and 10 provide in detail the structure in terms on input and output variables for the reliability, plant performance risk and cost models for the example problem.

Table 8. CRAFT reliability models for the test problem.

Reliability models	Inputs	Outputs
V1	a, b	Time dependent SSC failure probability
V2	a, b	
P1	α and β	
P2	Δ	
SG	μ, σ	

Table 9. CRAFT plant performance risk models for the test problem.

Plant performance risk models	Inputs	Outputs
V1	Time dependent SSC failure probability	Time dependent consequence probability
V2		
SG		
P1, P2		

Table 10. CRAFT cost models for the test problem.

Cost models	Inputs	Outputs
V1	Time dependent consequence probability	Time dependent costs
V2		
SG		
P1, P2		

4.3.11 Results

The analysis has been performed by performing a Monte-Carlo sampling of the developed CRAFT model. The objective is to show how it is possible to determine the temporal evolution of the economic risk associated this system by using CRAFT-RAVEN. Figure 13 and Figure 14 present the temporal profile of failure probability for the two valves and the SG in term of cumulative distribution function; note that since we are propagating uncertainties in these failure models, these plots include the mean value and the four more important percentiles (5th, 25th, 75th and 95th).

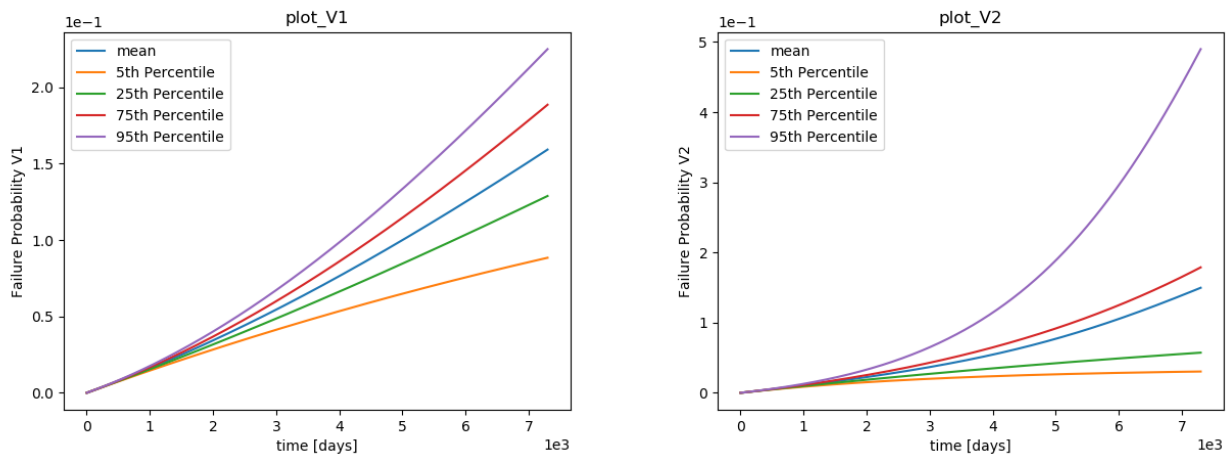


Figure 13. Failure probability for V1 and V2 plotted in term of CDF.

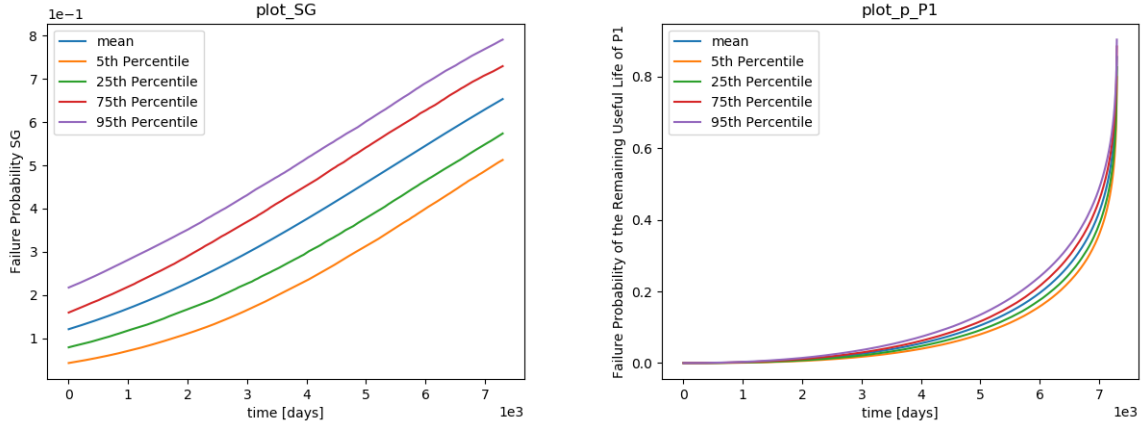


Figure 14. Failure probability for SG (left) and P1 (right) plotted in term of CDF.

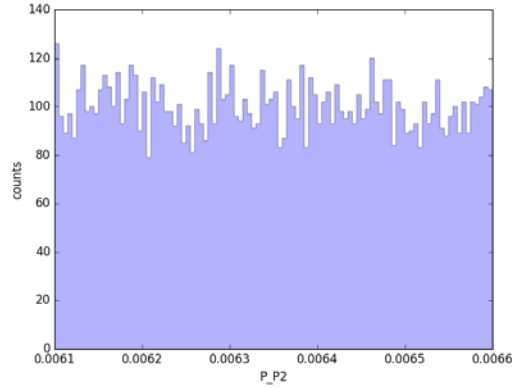


Figure 15. Histogram of failure probability for P2.

Figure 15 presents the plot of failure probability of P2; note that the failure probability of this component is constant through time. Figure 16 plots the temporal profile of the probabilities associated to the Branches 1 (left) and 2 (right) of the consequence model for the P1-P2 system of Figure 12. These probabilities have been determined through a convolution of the probabilities of P1 (see Figure 14 - right) and P2 (see Figure 15).

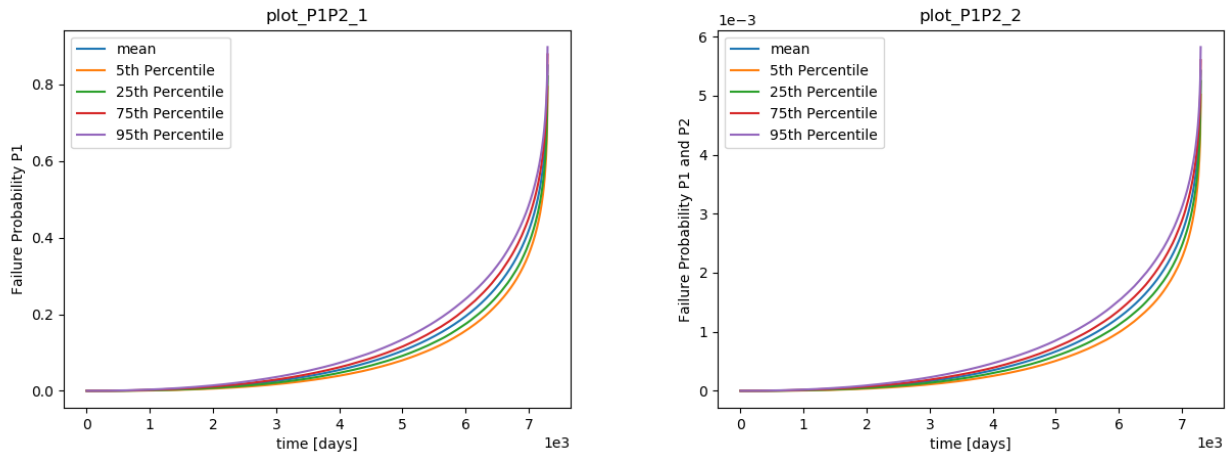


Figure 16. Temporal profile of the probabilities associated to the Branches 1 (left) and 2 (right) of the consequence model for the P1-P2 system of Figure 12.

Figure 17 shows a plot of the temporal profile of the cost models as function time provided the cost models indicated in Section 4.3.9 and the predicted energy cost. Figure 17 plots costs associated to Branch 1 of Figure 12 (left) for the P1-P2 system (left) and for the SG model (right). Note the monthly variability of the energy cost captured in the model.

Once these results are obtained, it is possible to rank SSC and act on the SSC with higher economic risk. These actions could involve the replacing these SSC, improve testing and maintenance procedures, install advance diagnostic/prognostic methods to reduced uncertainty related to the SSC RUL.

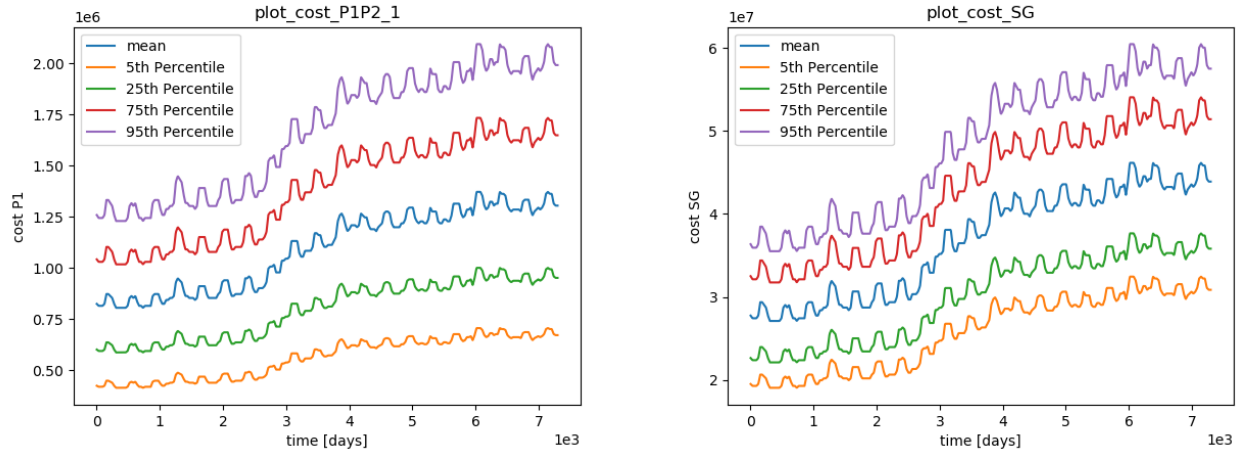


Figure 17. Temporal profile of the cost models for Branch 1 of Figure 12 for the P1-P2 system (left) and for the SG model (right).

5. INDUSTRY USE CASE: PLANT ASSET MANAGEMENT

As commercial NPPs pursue extended plant operation in the form of Second License Renewal (SLR), opportunities exist for these plants to provide capital investments to ensure long term safe and economic performance. At the current time, several utilities have announced an intention to pursue extended operation for one or more of their NPPs via SLR. The goal of this research is to enhance the long term safety and economics of NPPs during the SLR period of operation by providing a structured risk-informed approach to evaluate and prioritize plant capital investments made in preparation for and during the period of extended plant operation.

In this industry application, the partner utility indicated the need to evaluate the costs associated with major capital refurbishments that would occur during the period of the plant's SLR. Specifically, this Use Case application is intended to develop and apply methods and a tool to risk rank major capital refurbishments/replacements of SSCs for SLR with the ability to actively manage these costs between the time of funding approval and implementation to support operation during the period of SLR. The objective of this Use Case application is to risk-inform the capital improvements for SLR that includes assessment of the expected useful life of the SSCs and the likelihood and impacts of unanticipated occurrences that could impact their performance during this operating period. The Use Case application would be used to specify the optimal conditions and timing for replacement/ refurbishment based on anticipated conditions with appropriate assessments performed to address uncertainties.

For this Use Case application, the host utility has developed a list of potential capital improvements to support plant operation throughout the period of SLR. Example candidate plant refurbishment / replacement projects include plant digital I&C upgrades, refurbishment or replacement of Reactor Coolant Pumps (RCPs), replacement of buried piping (including the potential use of reinforced carbon fiber pipes), main generator replacement, and main condenser replacement. The primary objective of this Use Case application is to develop and apply methods and tools that are capable of assessing and managing the risks and likelihood of failure of these SSCs during this

operating period. In addition, secondary objectives for this application are to identify an optimal allocation of the capital expenditures for these SSCs and to manage these expenditures (via re-optimization) as circumstances change from initial approval through the end of the SLR operating period.

Provided the problem statement, two possible solutions can be followed for this use case:

- *Option 1:* Interface RAVEN with the ILCM code [4] developed by EPRI for asset management. This approach would minimize deployment time since it would employ existing codes/models including RAVEN to perform data analysis management.
- *Option 2:* Develop new models within RAVEN specific to needs of the problem. In this option we would need to develop cost models and the failure models for the considered capital SSCs. This approach would require more RD&D effort but such models could be tailored and optimized more precisely on the target application. In Section 5.1 we present a preliminary development of the CRAFT model that could be employed for this use case.

Depending on the evolution of this use case, the overall CRAFT model might include ILCM or the model presented in Section 5.1.

5.1 CRAFT Model

The structure of the CRAFT model for the SSC replacement problem is composed by a set of models, one for each SSC. Each SSC model is structured in three parts:

- *Failure model:* parametrizes in a probabilistic fashion the predicted failure time of the SSC or, similarly, its Remaining Useful Life (RUL)
- *Consequence model:* parametrizes consequence impact of SSC failure on plant operation (e.g., plant shut-down, regulatory consequences, missing production penalties)
- *Cost model:* translates the consequences predicted by the consequence model into an economic value such as the Net Present Value (NPV)

5.1.1 Failure Models

The failure model associated to each particular SSC can be either a stochastic variable in RAVEN with its associated distribution (e.g., exponential or Weibull) or a more complicated and dedicated model which can include the effect of multiple failure modes. It is not unusual that, in fact, large capital SSCs are characterized by several types of failure models each of which is characterized by a particular distribution. Note that each failure model might have an associated specific consequence model (see Section 5.1.2). The design of these models in RAVEN has been kept extremely general and flexible in order to keep them easily adaptable to any future analysis.

5.1.2 Consequence Models

The consequence model translates the failure of the SSC into an operational state of the plant. Table 11 and Table 12 summarize possible operational states of the plant.

Table 11. List of possible sources of plant savings.

Reduced maintenance costs	Decrease in CDF
Improved plant efficiency	Decrease in LERF
Refueling outage reduction	Reduced probability of plant trip
Reduced cost of spare parts (fewer parts or lower quality requirements)	Reduced probability of power reduction
Improved margin to ROP PI	Reduction in CAP entries

Table 12. List of possible sources of plant costs.

Unplanned reactor trip	Reduced plant efficiency
Forced mid-cycle maintenance outage	Increase in CDF
Forced power reduction	Increase in LERF
Refueling outage extension	Capital cost of improvement
Increased risk of industrial accident	Regulatory Costs <ul style="list-style-type: none"> • Reportable event (LER) • CAP entry • Additional NRC inspection • Slipped ROP PI

5.1.3 Cost Models

During FY18 we have developed the cost models that will be employed for the capital SSC replacement problem. These models evaluate the cost associated to the life of a capital SSC in terms of NPV for two scenarios:

- With replacement of the SSC
- Without replacement of the SSC

For each of the two scenarios, the possible degrees of freedom that have been initially considered are summarized in Table 13. Note that the variables listed in Table 13 might not be constant but they might be time-dependent and/or might have uncertainties associated to them. As an example, loss of revenues due to missed energy production caused by an unplanned shut-down could depend on the actual time of the unplanned shut-down (since the electricity price changes over the year). Another example might be associated to the fact that the T&M costs for the actual or new SSC might be subject to variability.

Appendices A and B shows in more detail the mathematical structure of the cost models for the capital SSC replacement use case.

Table 13. Degrees of freedom initially considered for the replacement problem.

Interest rate	Cost of SSC replacement
T&M costs for actual SSC	Penalty cost associated to SSC failure
T&M costs for new SSC	Mission time of the analysis

5.2 SSC Replacement Analysis Example

The scope of this analysis is to test the cost models associated to the SSC replacement use case. The objective is to generate the distribution of the NPV for a determined replacement time for a single SSC and compare it to

the NPV distribution when replacement is not planned. We employed the cost models (continuous case) shown in Appendix A and propagated uncertainties related to the failure time of both the old and the replaced SSC.

The structure of the obtained NPV distribution is shown in Figure 18. In particular, Figure 18 provides a comparison of the NPV distributions for two cases: with (red) and without (blue) planned replacement. Note that given the nature of the problem, the NPV distribution is multi-modal. The first mode (on the right of Figure 18) represents the NPV mode where SSC failure does not occur; this mode is characterized by the cost of testing/maintenance and the unplanned replacement of the SSC.

The second mode located at the middle of Figure 18 represents the NPV associated to the case where the component fails once during the considered time period. This mode is characterized not only by the cost of testing/maintenance of the old and the replaced SSC but also the penalty cost associated to the failure of the SSC.

The third mode located at the left of Figure 18 represents the NPV associated to the case where the component fails twice during the considered time period. This mode is characterized by not only the cost of testing/maintenance of the old and the replaced SSC but also the penalty cost associated to the failures of the considered SSC.

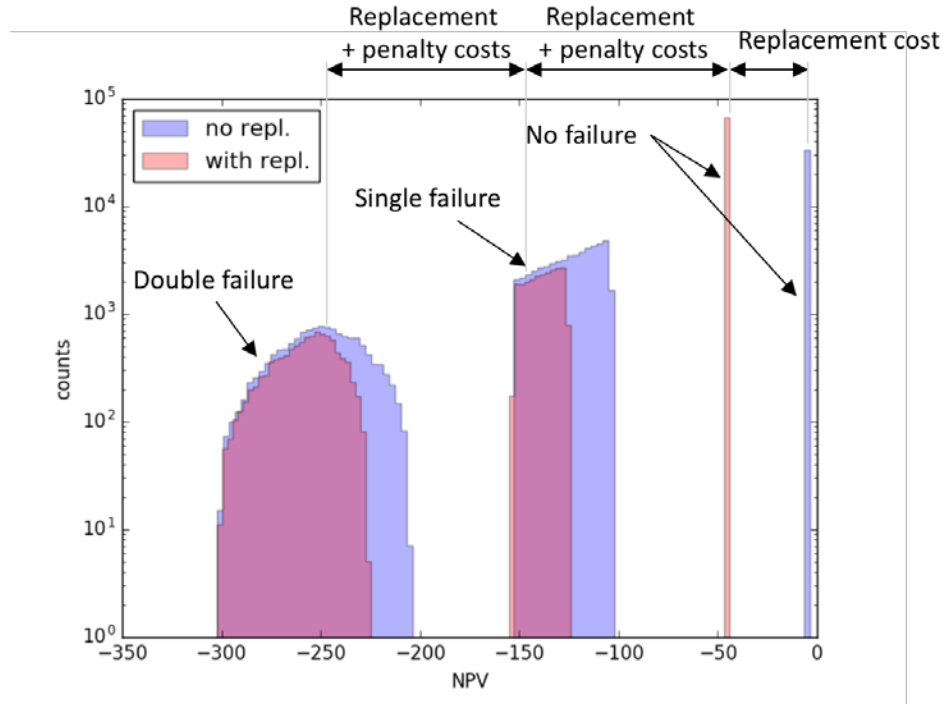


Figure 18. Comparison of the NPV distributions for two cases: with (red) and without (blue) planned replacement.

Then, we have repeated the analysis of Figure 18 for four scenarios. In each scenario we changed the expected failure probability of a single SSC (i.e., maximum expected life of 10, 20, 30 and 40 years) provided the SSC is expected to run for the next 20 years. In this respect, Figure 19 plots the box plot of the NPV for these four scenarios; in each plot it is possible to compare the NPV assuming the component is not replaced with the NPV assuming the component is replaced in 5, 10 and 15 years.

These models will form the basis to identify the optimal strategy of the SSC replacement strategy (replacement or not) and the best replacement time. This optimization problem will involve not a single SSC but multiple SSCs and also multiple constraints (e.g., time, money, SSC availability). Section 6 presents a possible optimization algorithms that will be investigated in the FY19.

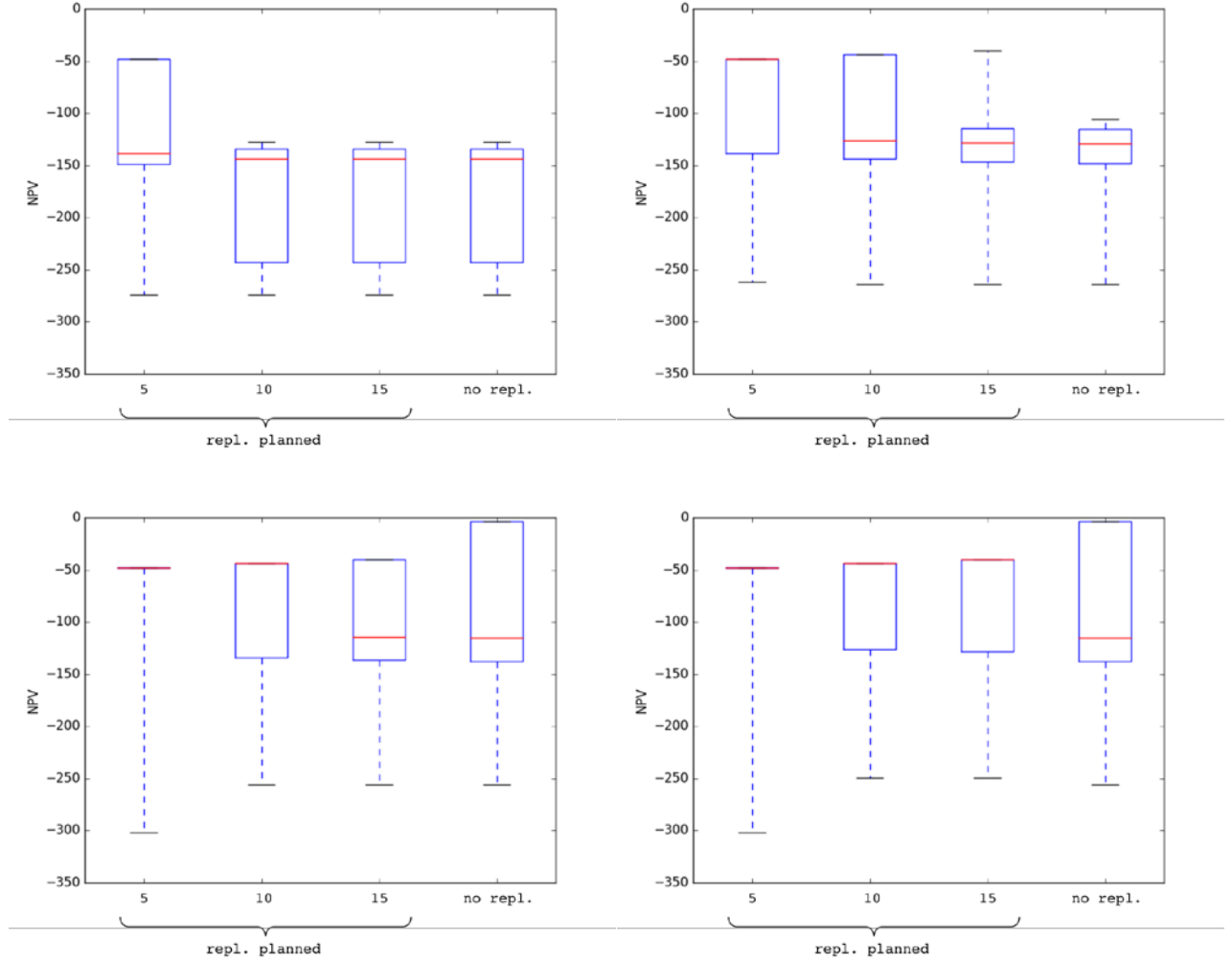


Figure 19. Box plots for different SSC failure models: [0,10] (top left), [0,20] (top right), [0,30] (bottom left) and [0,40] (bottom right).

6. OPTIMIZATION ALGORITHMS

The objective of the problem is to minimize the associated cost when replacing components within a system. From RAVEN, each component will have an associated net present value that may be distributed between several different modes within each component. Only one of the different modes per component must be selected such that the net present value is minimized. This problem may be formed into a variable selection problem where at most one of the modes per component is selected based on the overall maximum cost.

Variable selection is a statistical technique that modifies linear regression such that some of the regression coefficients are zero and therefore the variable is removed from the model. Each variable in this case is a given mode per component. This mode may be sampled numerous times to generate a set of random values of the mode. If this sampling is performed for each mode/component combination, then a large matrix of simulation values may be collected for a given system. More concretely, assume the net present value is normalized to one between components, then the linear regression model is formed as:

$$\|1 - X\beta\|_2^2$$

where \mathbf{X} has n rows as the number of simulations and p columns as the different mode/components, $\boldsymbol{\beta}$ are the regression coefficients that need to be optimized and $\mathbf{1}$ is a vector of 1s. To perform variable selection, the equation above must be altered such that there is a penalty on the regression coefficients reducing some to zero and only one of the modes may be non-zero.

The modification to the linear regression model that makes this possible is called Group Lasso [18]. The regression coefficients are penalized such that within each component only one regression coefficient is allowed to be non-zero. Let l be the total number of components, then the group lasso formulation of the problem is given as:

$$\left\| \mathbf{1} - \sum_l \mathbf{X}_l \boldsymbol{\beta}_l \right\|_2^2 + \lambda_1 \|\boldsymbol{\beta}\|_1 + \sum_l (P(\boldsymbol{\beta}_l), \lambda_2)$$

where λ represents penalty parameters and $P(\cdot)$ is a penalty function. The second term in the above equation ensures that some of the variables are removed while the last term performs individual penalty on each component. The selected penalty for the component will be based on a hard thresholding approach where at most one of the modes will be selected.

The optimization for the selection of the variables is repeated for each different discrete time where an upgrade to the overall system is performed. As such, the mode and component distributions are updated after each time optimization to take account for the recently upgraded component. This means that the simulation must be run again with a new NPV for all components as the upgraded components should reflect a set of new modes and the unchanged components will have the time scale changed within the distribution. In this way, each different time step takes into account the prior time lag effects.

Computationally, the group lasso problem may be solved through Alternating Direction Method of Multipliers (ADMM) [19]. This method is chosen due to its ability to decompose the problem into multiple smaller problems while being fast and scalable. Additionally, ADMM is flexible in the choice of the penalization in the group lasso problem such that the code may be modified for any future optimization problems.

7. RAVEN INTERFACE WITH SAPHIRE

The main objective of CRAFT is to develop an innovative framework that combines physics, risk and cost information of components in a nuclear power plant to enable a risk-and-cost-based decision-making for improving plant reliability and ensuring cost-efficiency. The risk associated to the nuclear power plants can be determined by classical PRA tools, such as SAPHIRE, under the assumption that all failure rates are constant over time. In some circumstances, however, the assumption is rendered invalid since the failure rate of a component may gradually change in a systematic way, for example, because of aging and degradation. In addition, recent data analysis has suggested decreasing values of failure rates for several important components. A PRA that includes these components should employ the time-dependent failure rates in order to accurately estimate the risk at any given time. This time-dependent PRA is achieved in this report by coupling RAVEN with SAPHIRE.

SAPHIRE is a software application, funded by U.S. Nuclear Regulatory Commission (NRC) and developed by Idaho National Laboratory (INL), for performing a complete PRA on a personal computer. SAPHIRE can be used to perform all three levels of risk: 1) model a complex system's response to initiating events, quantify associated damage outcome frequencies, and identify important contributors to this damage (Level 1 PRA); 2) analyze containment performance during a severe accident and quantify radioactive releases (Level 2 PRA); 3) quantify risk in terms of radioactivity release accidents to both the public and the environment (Level 3 PRA). In addition, SAPHIRE can also be used to perform FT/ET uncertainty analysis.

RAVEN is a flexible and multi-purpose framework, also developed by INL, that has been designed to perform uncertainty quantification, regression analysis, probabilistic risk assessment, data mining and model optimization. From CRAFT perspective, the coupling between RAVEN and SAPHIRE can be used to perform the time-

dependent safety analysis with uncertainty quantification, integrating the reliability, availability and maintainability models with the time-dependent PRA models that can be used in the cost analysis, and model the time-dependent failure rate or failure probability distribution with Bayesian inference. The technical detail of the coupling between RAVEN and SAPHIRE can be found in Appendix D.

When performing time-dependent safety related risk analysis (Figure 6), ETs and FTs methods from SAPHIRE are employed to model accident progression while time-dependent components failure data (i.e. failure probabilities and failure rates) generated via RAVEN external model are used to determine the frequency/probability of each accident sequence at particular time.

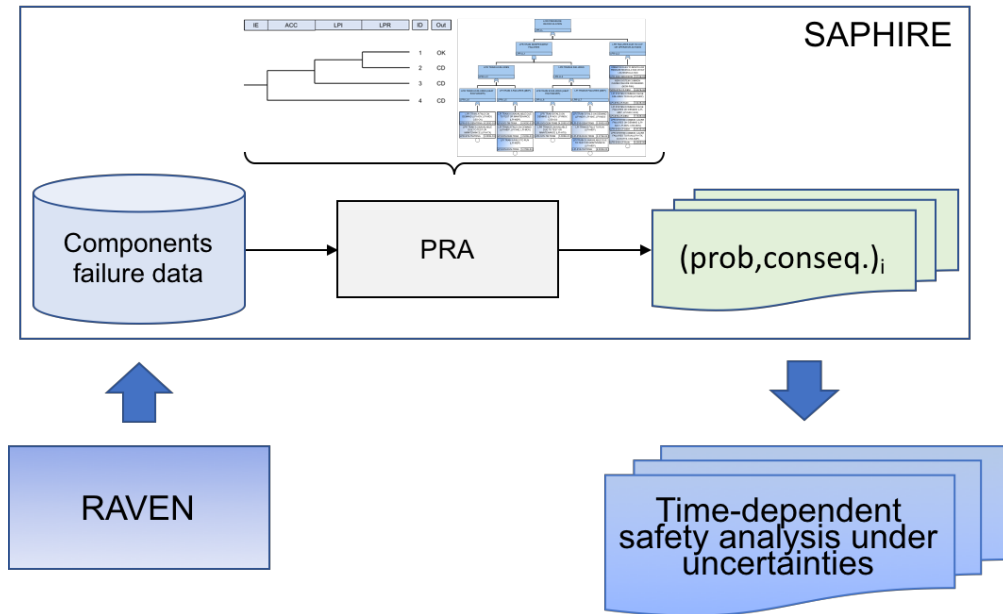


Figure 20. Time-dependent safety related risk analysis with RAVEN and SAPHIRE.

7.1.1 Time Dependent Probabilistic Risk Assessment

The classical PRAs in SAPHIRE are based on static approaches, i.e. average of the real world to approximate the complex NPP behaviors. In SAPHIRE, ETs are used to model accident progression for a specific initiating event. FTs are used to determine the systems' unavailability through a series of logic gates. Both ETs and FTs are static models without the interactions between the evolution of the physical process, the operator actions, and the system configuration. However, it is sometimes the case that the usual static models are rendered invalid since the parameters of interest, i.e. failure rates or failure probabilities, are not constant over time. As mentioned in [5], the failure rates and failure probabilities are decreasing over time for several important components. These time-dependent phenomena can have significant impact on the results of PRAs and on decisions based on the outcomes of such analyses.

The coupling between RAVEN and SAPHIRE (see Appendix D) allow the users to integrate the time-dependent phenomena, such as time-dependent reliability models and time-dependent availability models, with the SAPHIRE PRAs. This can be used to perform the time-dependent safety assessment based on the enhanced PRA models.

In this report, a simple PRA model based on ETs and FTs (shown in Figure 21, Figure 22, and Figure 23) for a single initiating event (e.g. Loss Of Offsite Power - LOOP) is employed to perform the time-dependent risk analysis. The status of each of the two safety systems (i.e., ECS: Emergency Cooling System and CCS: Containment Cooling System) dictates the branching condition of the ET and it is modeled through a set of FTs,

one for each system. SAPHIRE is employed to calculate the most probable combination of basic events that lead to radioactive releases: the Minimal Cut Sets (MCSs). The sum of the probability associated to each MCS determines the probability of radioactive releases given the occurrence of LOSP initiating event. Three outcomes are possible: OK (i.e. no radioactive releases), small-release, and large-release. By observing in Figure 21, radioactive release outcomes can be avoided only if all the two systems are functioning.

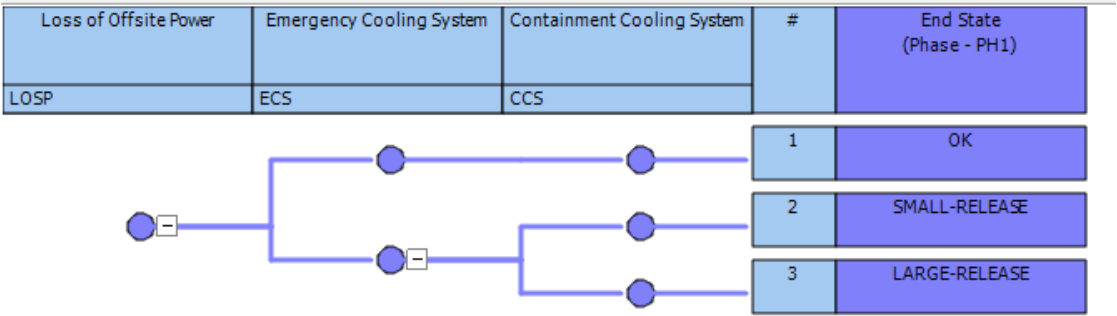


Figure 21. LOSP ET model.

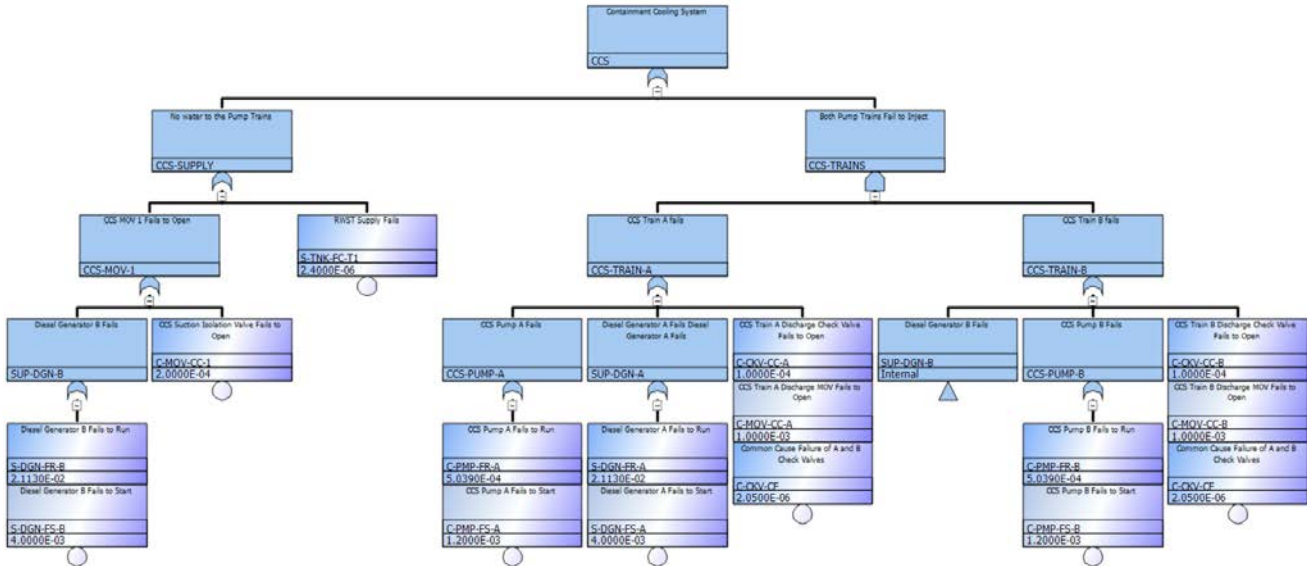


Figure 22. CCS FT model.

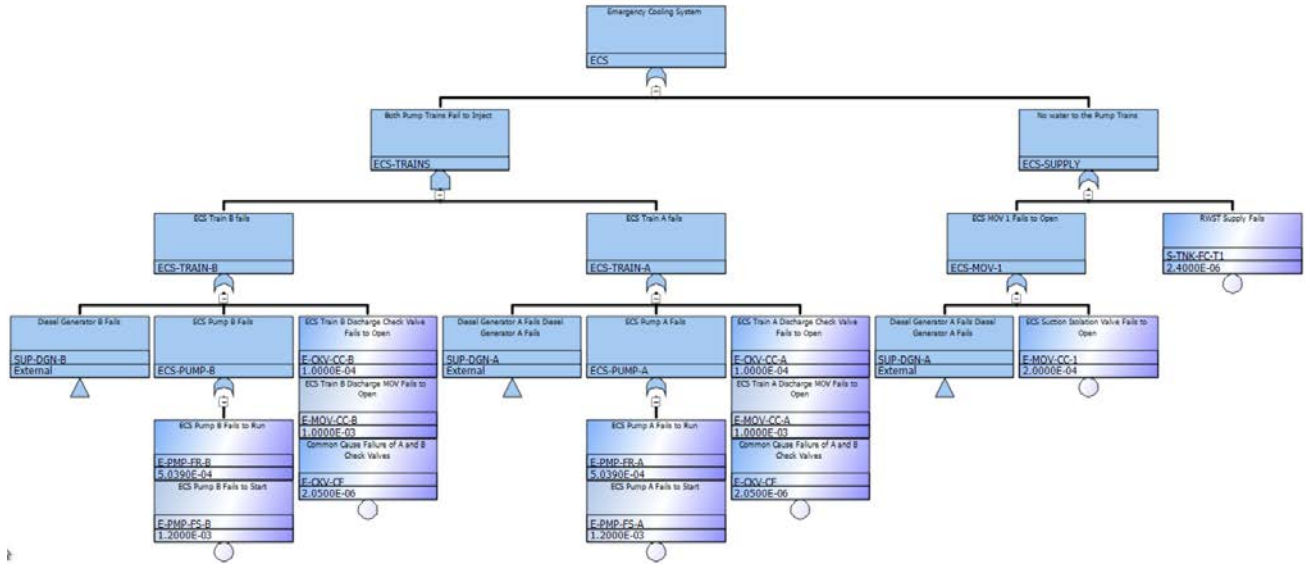


Figure 23. ECS FT model.

The failure model of each basic events in SAPHIRE contains a calculation type selection box. There are two calculation types identified as 1, 3 are employed in this analysis as shown in Table 14.

Table 14. Basic failure model employed in ECS/CCS FTs.

Calculation Type	Equation	Description
1	$P = p$	Mean probability
3	$P = 1 - \exp(-\lambda T_m)$	λ : mean failure rate, T_m : mission time. Failure probability of an operating component without repair.

In this example, the time-dependent models of failure rate and failure probability are employed as shown in the following equations:

- Linear failure probability model: $P(t) = P_0[1 + b(t - t_0)]$
- Exponential failure rate model: $\lambda(t) = \lambda_0 \exp(b(t - t_0))$

Here, t_0 is a value selected by the analyst for convenience. In these two models, $P(t)$ or $\lambda(t)$ is increasing if $b > 0$, is constant if $b = 0$, and is decreasing if $b < 0$. P_0 or λ_0 is the value of failure probability/rate at time $t = t_0$. If t_0 is set to zero, $t - t_0$ is the time measured from the component's installation. The detailed equations that are used in this example can be found in Table 15 and Table 16.

Table 15. Basic events employed in CCS FT.

Basic Event	Type	Failure Probability/Rate	Description
C-CKV-CC-A	1	$P = 1.0E - 5 + 1.45E - 5 * (t - t_0)$	Train A discharge check valve fails to open
C-CKV-CC-B	1	$P = 1.0E - 5 + 1.45E - 5 * (t - t_0)$	Train B discharge check valve fails to open
C-MOV-CC-1	1	$P = 2.0E - 5 + 2.9E - 5 * (t - t_0)$	Suction isolation valve fails to open
C-MOV-CC-A	1	$P = 1.0E - 5 + 1.45E - 5 * (t - t_0)$	Train A discharge MOV fails to open
C-MOV-CC-B	1	$P = 1.0E - 5 + 1.45E - 5 * (t - t_0)$	Train B discharge MOV fails to open
C-PMP-FR-A	3	$\lambda = 2.1E - 6 * \exp(0.17(t - t_0))$	Pump A fails to run
C-PMP-FR-B	3	$\lambda = 2.1E - 6 * \exp(0.17(t - t_0))$	Pump B fails to run
C-PMP-FS-A	1	$P = 1.2E - 4 + 1.74E - 4 * (t - t_0)$	Pump A fails to start
C-PMP-FS-B	1	$P = 1.2E - 4 + 1.74E - 4 * (t - t_0)$	Pump B fails to start
S-DGN-FR-A	3	$\lambda = 8.9E - 5 * \exp(0.17(t - t_0))$	Diesel generator A fails to run
S-DGN-FR-B	3	$\lambda = 8.9E - 5 * \exp(0.17(t - t_0))$	Diesel generator B fails to run
S-DGN-FS-A	1	$P = 4.0E - 4 + 5.8E - 4 * (t - t_0)$	Diesel generator A fails to start
S-DGN-FS-B	1	$P = 4.0E - 4 + 5.8E - 4 * (t - t_0)$	Diesel generator B fails to start

Table 16. Basic events employed in ECS FT.

Basic Event	Type	Failure Probability/Rate	Description
E-CKV-CC-A	1	$P = 1.0E - 5 + 1.45E - 5 * (t - t_0)$	Train A discharge check valve fails to open
E-CKV-CC-B	1	$P = 1.0E - 5 + 1.45E - 5 * (t - t_0)$	Train B discharge check valve fails to open
E-MOV-CC-1	1	$P = 2.0E - 5 + 2.9E - 5 * (t - t_0)$	Suction isolation valve fails to open
E-MOV-CC-A	1	$P = 1.0E - 5 + 1.45E - 5 * (t - t_0)$	Train A discharge MOV fails to open
E-MOV-CC-B	1	$P = 1.0E - 5 + 1.45E - 5 * (t - t_0)$	Train B discharge MOV fails to open
E-PMP-FR-A	3	$\lambda = 2.1E - 6 * \exp(0.17(t - t_0))$	Pump A fails to run
E-PMP-FR-B	3	$\lambda = 2.1E - 6 * \exp(0.17(t - t_0))$	Pump B fails to run
E-PMP-FS-A	1	$P = 1.2E - 4 + 1.74E - 4 * (t - t_0)$	Pump A fails to start
E-PMP-FS-B	1	$P = 1.2E - 4 + 1.74E - 4 * (t - t_0)$	Pump B fails to start
S-DGN-FR-A	3	$\lambda = 8.9E - 5 * \exp(0.17(t - t_0))$	Diesel generator A fails to run
S-DGN-FR-B	3	$\lambda = 8.9E - 05 * \exp(0.17(t - t_0))$	Diesel generator B fails to run
S-DGN-FS-A	1	$P = 4.0E - 04 + 5.8E - 04 * (t - t_0)$	Diesel generator A fails to start
S-DGN-FS-B	1	$P = 4.0E - 04 + 5.8E - 04 * (t - t_0)$	Diesel generator B fails to start

As mentioned, SAPHIRE can be used to perform uncertainty analysis of ETs/FTs via Monte Carlo or Latin Hypercube sampling. In other words, the variability of a FT top event or ET sequence end state probability resulting from uncertainties in the basic event probabilities can be quantified. Monte Carlo sampling strategy is one of the most-used sampling strategies. It approximates an expectation by sample mean of a function of

simulated random variables based on the laws of large numbers. Latin Hypercube sampling (LHS) is a stratified sampling technique, with the random variable distributions divided into equal probability intervals. In this study, LHS is employed setting a limit of 5000 simulations. The following MACRO is used to perform the uncertainty analysis of ETs and FTs:

- Uncertainty analysis of ETs with MACROs:

```
<sequence>
  <unmark></unmark>
  <include>
    <mark event tree mask>LOSP</mark event tree mask>
    <mark logic fault tree>*</mark logic fault tree>
  </include>

  <solve>
    <truncation>1.0E-15</truncation>
  </solve>

  <uncertainty>
    <sample>5000</sample>
    <seed>4321</seed>
    <method>lhs</method>
    <intermediate>csv</intermediate>
    <file name>etuq_im.csv</file name>
  </uncertainty>

  <importance>
    <group>yes</group>
    <seed>4321</seed>
    <sort order>name</sort order>
    <file name>imp-all.csv</file name>
    <report format>csv</report format>
  </importance>
</sequence>
```

- Uncertainty analysis of FTs with MACROs:

```
<fault tree>
  <mark name>CCS</mark name>
  <mark name>ECs</mark name>
  <uncertainty>
    <sample>5000</sample>
    <seed>1234</seed>
    <method>lhs</method>
    <intermediate>csv</intermediate>
    <file name>ftuq_im.csv</file name>
  </uncertainty>

  <report>
    <type>use title</type>
    <title>Uncertainty</title>
    <file name>ft_uq.csv</file name>
    <report format>csv</report format>
  </report>
</fault tree>
```

Grid-based sampling from RAVEN is employed to perform the parametric analysis of ETs/FTs with respect to time. The time-dependent uncertainty analysis is performed by mean of an equally spaced value grid. Now, ET/FT models are run in a time-dependent fashion by evaluating radioactive releases at specific time instants (e.g. 6 months) over a fixed time scale (e.g. 20 years). The time-dependent failure rates and failure probabilities are generated by RAVEN with external functions as provided in Table 15 and Table 16. Figure 24 and Figure 25

illustrate the mean and the 5th/95th percentiles of small radioactive releases and large radioactive releases over 20 years. Figure 26 and Figure 27 show the mean and the 5th/95th percentiles of ECS and CCS over 20 years.

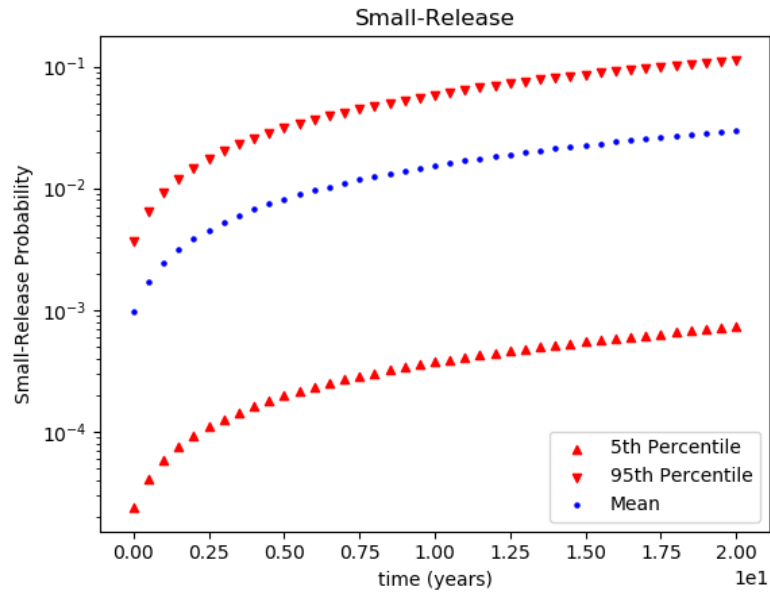


Figure 24. Time dependent uncertainty analysis of ET outcome small-release.

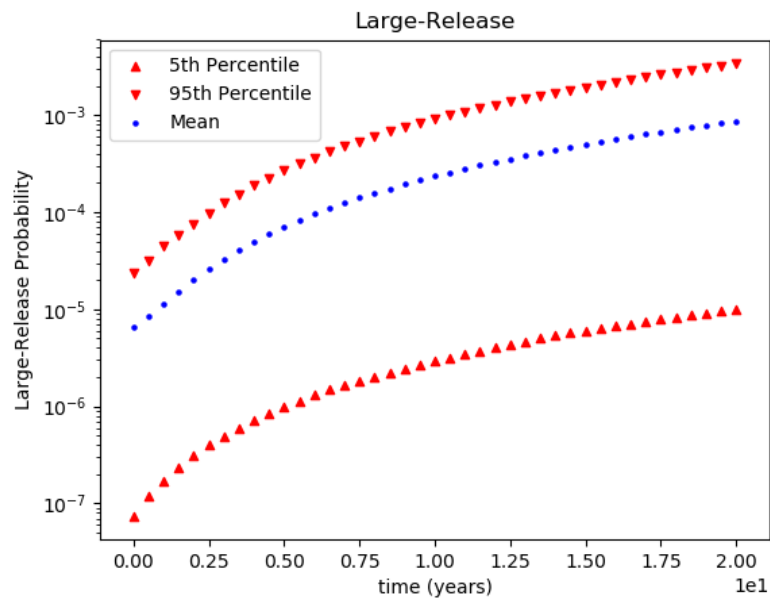


Figure 25. Time dependent uncertainty analysis of ET outcome large-release.

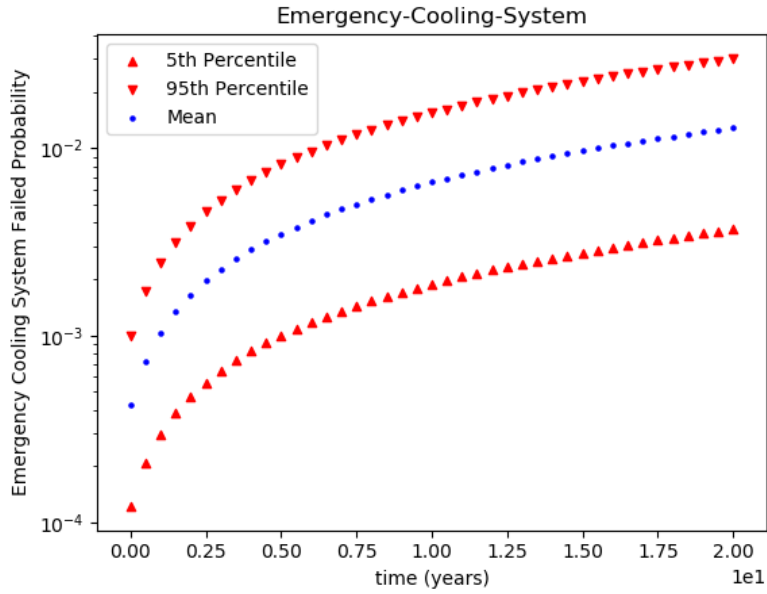


Figure 26. Time dependent uncertainty analysis of FT ECS.

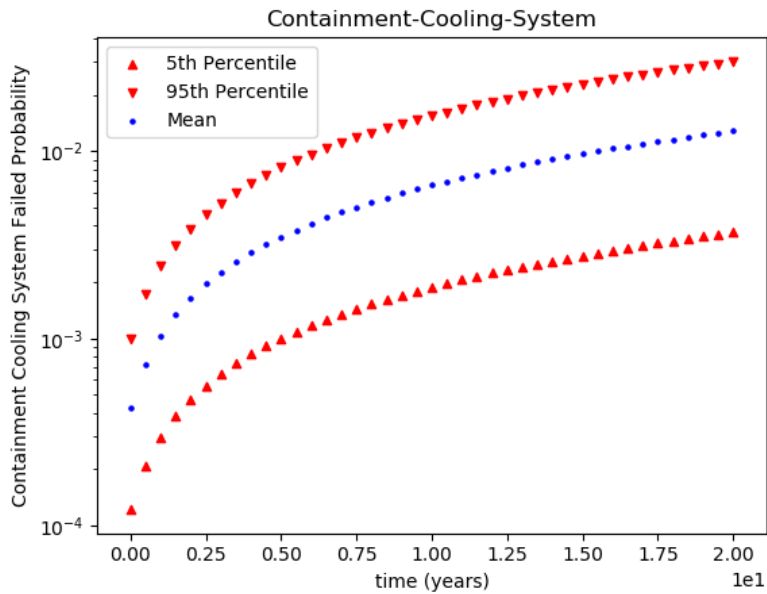


Figure 27. Time dependent uncertainty analysis of FT CCS.

As demonstrated with the simple LOSP PRA model, the coupling of RAVEN and SAPHIRE can be used to perform the time-dependent risk/safety assessment. In the future, the explicit SSC aging or degradation models that explicitly model how the failure rates or failure probabilities change in time can be directly used in PRA of NPP with this coupling capability. The coupling with other commercialized PRA tools, such as CAFTA and RISK SPECTRUM, will also be explored to support the SLR of commercial NPPs.

8. CONCLUSIONS AND FUTURE ACTIVITIES

In light of the development presented in this report we see the following as possible future R&D development items that can be carried out in FY19 in collaboration with two industry partners and Jensen Hughes (JH):

1. Integration of RAVEN with the EPRI ILCM code
2. Deployment of the models and methods developed for the asset management use case in collaboration with the industry partner
3. Integration of the System Health Program and Risk-Informed Applications

In this report we have summarized the most recent developments accomplished during the second part of FY18 for the two use cases of the Cost and Risk Categorization Applications path of the RISA project. These two use cases directly target plant health and plant capital SSC management. Regarding the plant health management use case, we have started to investigate methods to extract information from text-based data (e.g., incident reports): text data mining. This activity will continue in the next fiscal years for these applications:

- *Develop Necessary Linkages between System Health Program and Risk-Informed Applications:* The US commercial nuclear power industry is aggressively pursuing implementation of several risk-informed applications to reduce regulatory burden and operating costs. These applications include Alternative Treatments (10CFR50.69) and Risk-Managed Technical Specification (TSTF-505). To obtain maximum value from these applications, their execution must be strongly integrated with a robust and automated System Health program. This application will likely employ the text data mining capabilities presented in Appendix E.
- *Integration of Equipment Diagnostic/Prognostics to System Health:* we will leverage equipment monitoring activities performed at the host utility's Maintenance and Diagnostics (M&D) Center. The scope is to accelerate the deployment and use of advanced diagnostic/prognostic capabilities to develop a Maintenance Strategy Optimizer (MSO) that is intended to continuously optimize cost and SSC performance.

The Risk Informed Asset management is the main target of this report and several methods have been shown in order to solve the capital SSC replacement issue. This has been performed by presenting:

- Several models designed to determine the effective cost through the lifetime of a capital SSC, and,
- The optimization algorithms that will be employed to determine the optimal replacement schedule of a given set of capital SSC.

The interface between RAVEN and the PRA code SAPHIRE is the first toward the integration of several plant risk models (e.g., economic and safety) and plant data (e.g., database from the plant maintenance and diagnostics center) into a single analysis framework. In this respect, RAVEN is the ideal platform to connect several models and to manage data streaming among them.

REFERENCES

- [1] R. Szilard et al., “RISMC Toolkit and Methodology Research and Development Plan for External hazards Analysis,” INL Technical Report INL/EXT-16-38089 (2016).
- [2] R. Szilard, H. Zhang, S. Hess, J. Gaertner, D. Mandelli, S. Prescott, “RISA Industry Use Case Analysis,” INL Technical Report INL/EXT-18-51012 (2018).
- [3] C. Rabiti, A. Alfonsi, J. Cogliati, D. Mandelli, R. Kinoshita, S. Sen, C. Wang, J. Chen, “RAVEN User Manual,” INL Technical Report INL/EXT-15-34123 (2017).
- [4] D. Mandelli, C. Wang, S. St. Germain, C. Smith, C. Rabiti, A. Alfonsi, “RISMC Toolkit and Methodology Research and Development Plan for External hazards Analysis,” INL Technical Report INL/EXT-18-51214 (2018).
- [5] K. J. Kvarfordt, S. T. Wood, C. L. Smith, “Systems Analysis Programs for Hands-On Integrated Reliability Evaluations (SAPHIRE) Data Loading Manual,” INL Technical Report INL/EXT-05-00643 (2008).
- [6] E. Zio, M. Marseguerra, J. Devooght, P. Labeau, “A Concept Paper on Dynamic Reliability via Monte Carlo Simulation,” *Mathematics and Computers in Simulation*, **47**, pp. 371-382 (1998).
- [7] J. C. Helton, F. J. Davis, “Latin Hypercube Sampling and the Propagation of Uncertainty in Analyses of Complex Systems,” *Reliability Engineering & System Safety*, **81**, 1 (2003).
- [8] D. Mandelli, C. Smith, A. Alfonsi, C. Rabiti, J. Cogliati, “Improved Sampling Algorithms in the Risk-Informed Safety Margin Characterization Toolkit,” INL Technical Report INL/EXT-15-35933 (2015).
- [9] RELAP5-3D Code Development Team, RELAP5-3D Code Manual (2005).
- [10] Fauske and Associates Inc, “MAAP4-Modular Accident Analysis Program for LWR Power Plants,” Vols. 1 to 4, Electric Power Research Institute (1994).
- [11] D. Mandelli, C. Smith, “Adaptive Sampling Using Support Vector Machines,” *Proceeding of American Nuclear Society (ANS)*, San Diego (CA), **107**, pp. 736-738 (2012).
- [12] A. Alfonsi, C. Rabiti, D. Mandelli, J. Cogliati, R. Kinoshita, A. Naviglio, “Dynamic Event Tree Analysis Through RAVEN,” in *Proceedings of International Topical Meeting on Probabilistic Safety Assessment and Analysis (PSA 2013)*, September 22-26, Columbia, SC, USA, (2013).
- [13] H. S. Abdel-Khalik, Y. Bang, J. M. Hite, C. B. Kennedy, C. Wang, “Reduced Order Modeling for Nonlinear Multi-Component Models”, *International Journal On Uncertainty Quantification*, **2** (4), pp. 341-361 (2012).
- [14] EPRI, “CAFTA (Computer-Assisted Fault Tree Analysis) User's Manual: Version 2.0,” EPRI Technical Report EPRI-NP-6296 (1989).
- [15] C. Mengers, “Integrated Life Cycle Management: Status Report,” EPRI Technical Report 1021188 (2010).
- [16] J. K. Vaurio, “Optimization of Test and Maintenance Intervals Based on Risk and Cost,” *Reliability Engineering and System Safety*, **49**, pp. 23-36 (1995).
- [17] J. C. Lee Norman J. McCormick, “Risk and Safety Analysis of Nuclear Systems,” *Wiley ed.* (2011).
- [18] L. Meier, S. Van De Geer, P. Bühlmann, “The group lasso for logistic regression. Journal of the Royal Statistical Society,” *Series B Statistical Methodology*, **70**, pp. 53-71 (2008).
- [19] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Foundations and Trends in Machine Learning*, **3** (1), pp. 1-122 (2011).

- [20] S.A. Eide, T.E. Wierman, C.D. Gentillon, D.M. Rasmuson, C.L. Atwood, "Industry-average performance for components and initiating events at US commercial nuclear power plants," US Nuclear Regulatory Commission NUREG/CR-6928 (2007).
- [21] A. Alfonsi, C. Rabiti, D. Mandelli, "Assembling Multiple Models Within the RAVEN Framework," *Transactions of the American Nuclear Society*, **116**, San Francisco (2017).
- [22] J. Devooght, "Dynamic reliability," *Advances in Nuclear Science and Technology*, **25**, pp. 215-278 (1997).

APPENDIX A - COST MODELS IN CONTINUOUS FORM

A.1 No Replacement Planned

This appendix presents the cost model in continuous form assuming no replacement planned (NRP).

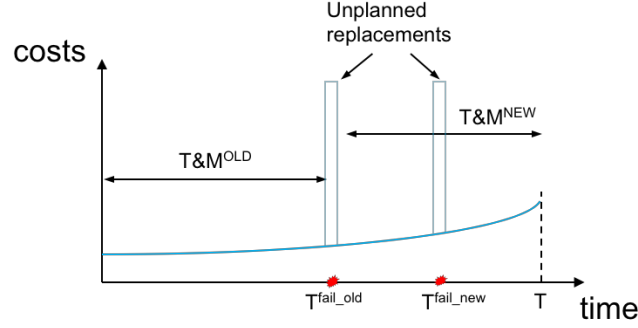


Figure 28. Plot of plant costs for SSC no replacement case (continuous form).

In this case, the corresponding NPV (i.e., NPV^{NRP}) can be calculated as:

$$NPV^{NRP} = \int_0^T pdf^{fail_old}(t) \cdot NPV^{NRP,fail}(t) dt + \left(1 - \int_0^T pdf^{fail_old}(t) dt\right) \cdot NPV^{NRP,no_fail} \quad (1)$$

where $NPV^{NRP,fail}(T^{fail_old})$ is the NPV assuming the component has failed at time T^{fail_old} :

$$\begin{aligned} NPV^{NRP,fail}(T^{fail_old}) &= \int_{T^{fail_old}}^T pdf^{fail_new}(t) \cdot NPV^{NRP,fail_twice}(T^{fail_old}, t) dt \\ &+ \left(1 - \int_{T^{fail_old}}^T pdf^{fail_new}(t) dt\right) \cdot NPV^{NRP,fail_pre}(T^{fail_old}) \end{aligned} \quad (2)$$

Each term in Equation (2) is explicitly shown in the Equation (3) to (5).

$$\begin{aligned} NPV^{NRP,fail_pre}(T^{fail_old}) &= \int_0^{T^{fail}} Cost^{TM,old}(t) \cdot e^{-rt} dt + (Cost^{repl} + Cost^{penalty}) \cdot e^{-rT^{fail}} \\ &+ \int_{T^{fail}}^T Cost^{TM,new}(t) \cdot e^{-rt} dt \end{aligned} \quad (3)$$

$$\begin{aligned}
& NPV^{NRP, fail_twice}(T^{fail_old}, T^{fail_new}) \\
&= \int_0^{T^{fail_old}} Cost^{TM,old}(t) \cdot e^{-rt} dt + (Cost^{repl} + Cost^{penalty}) \cdot e^{-rT^{fail_old}} \\
&+ \int_{T^{fail_old}}^{T^{fail_new}} Cost^{TM,new}(t) \cdot e^{-rt} dt + (Cost^{repl} + Cost^{penalty}) \cdot e^{-rT^{fail_new}} \\
&+ \int_{T^{fail_new}}^T Cost^{TM,new}(t) \cdot e^{-rt} dt
\end{aligned} \tag{4}$$

$$NPV^{NRP, no_fail} = \int_0^T Cost^{TM,old}(t) \cdot e^{-rt} dt \tag{5}$$

A.2 Replacement Planned

This appendix presents the cost model in continuous form assuming no replacement planned (NRP).

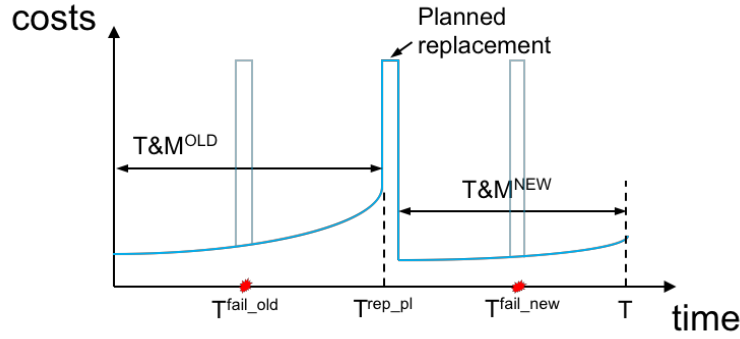


Figure 29. Plot of plant costs for SSC replacement case (continuous form).

In this case, the corresponding NPV (NPV^{RP}) can be calculated as:

$$\begin{aligned}
NPV^{RP} &= \int_0^{T^{rep_pl}} pdf^{failure,old}(t) \cdot NPV^{RP, fail_pre}(t) dt \\
&+ \left(1 - \int_0^{T^{rep_pl}} pdf^{fail,old}(t) dt \right) \int_{T^{rep_pl}}^T pdf^{fail,new}(t) \cdot NPV^{RP, fail_post}(t) dt \\
&+ \left(1 - \int_0^{T^{rep_pl}} pdf^{fail,old}(t) dt \right) \left(1 - \int_{T^{rep_pl}}^T pdf^{fail,new}(t) dt \right) \cdot NPV^{RP, no_fail}
\end{aligned} \tag{6}$$

where:

1. NPV assuming SSC does not fail is:

$$NPV^{RP,no_fail} = \int_0^{T^{rep_pl}} Cost^{TM,old}(t) \cdot e^{-rt} dt + Cost^{repl} \cdot e^{-rT^{rep_pl}} + \int_{T^{rep_pl}}^T Cost^{TM,new}(t) \cdot e^{-rt} dt \quad (7)$$

2. NPV assuming SSC fails prior scheduled replacement time is:

$$NPV^{RP,fail_pre}(T^{fail_old}) = \int_{T^{fail_old}}^T pdf^{fail,new}(t) \cdot NPV^{RP,fail_pre,fail_post}(T^{fail_old}, t) dt + \left(1 - \int_{T^{fail_old}}^T pdf^{fail,new}(t) dt\right) \cdot NPV^{RP,fail_pre,no_fail_post}(T^{fail_old}) \quad (8)$$

3. NPV assuming SSC fails prior scheduled replacement is:

$$NPV^{RP,fail_pre,no_fail_post}(T^{fail_old}) = \int_0^{T^{fail_old}} Cost^{TM,old}(t) \cdot e^{-rt} dt + (Cost^{repl} + Cost^{penalty}) \cdot e^{-rT^{fail_old}} + \int_{T^{fail_old}}^T Cost^{TM,new}(t) \cdot e^{-rt} dt \quad (9)$$

4. NPV assuming SSC fails prior scheduled replacement and also after the replacement time is:

$$NPV^{RP,fail_pre,fail_post}(T^{fail_old}, T^{fail_new}) = \int_0^{T^{fail_old}} Cost^{TM,old}(t) \cdot e^{-rt} dt + (Cost^{repl} + Cost^{penalty}) \cdot e^{-rT^{fail_old}} + \int_{T^{fail_old}}^{T^{fail_new}} Cost^{TM,new}(t) \cdot e^{-rt} dt + (Cost^{repl} + Cost^{penalty}) \cdot e^{-rT^{fail_new}} + \int_{T^{fail_new}}^T Cost^{TM,new}(t) \cdot e^{-rt} dt \quad (10)$$

5. NPV assuming SSC fails after scheduled replacement time is:

$$NPV^{NRP,fail_post}(T^{fail}) = \int_0^{T^{rep_pl}} Cost^{TM,old}(t) \cdot e^{-rt} dt + Cost^{repl} \cdot e^{-rT^{rep_pl}} + \int_{T^{rep_pl}}^{T^{fail}} Cost^{TM,new}(t) \cdot e^{-rt} dt + (Cost^{repl} + Cost^{penalty}) \cdot e^{-rT^{fail}} + \int_{T^{fail}}^T Cost^{TM,new}(t) \cdot e^{-rt} dt \quad (11)$$

APPENDIX B - COST MODELS IN DISCRETE FORM

B.1 No Replacement Planned

This appendix presents the cost model in discrete form assuming no replacement planned (NRP).

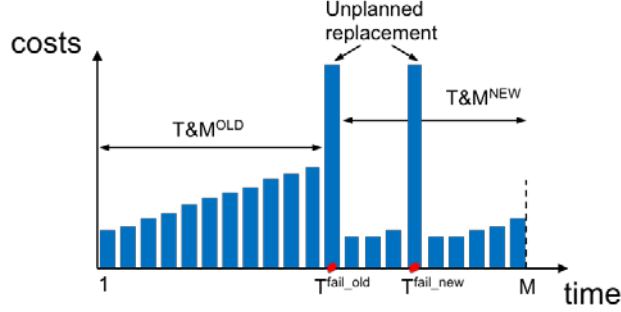


Figure 30. Plot of plant costs for SSC no replacement case (discrete form).

In this case, the corresponding NPV can be calculated as:

$$NPV^{NRP} = \sum_{t=1}^M p^{fail_old@t} \cdot NPV^{fail_old@t} + \left(1 - \sum_{t=1}^M p^{fail_old@t}\right) \sum_{t=1}^M \frac{Cost^{TM,old}(t)}{(1+i)^t} \quad (12)$$

where:

- The NPV assuming the old component fails at T^{fail_old} is:

$$\begin{aligned} NPV^{fail_old@T^{fail_old}} &= \sum_{t=T^{fail_old}+1}^M p^{fail_new@t} \cdot NPV^{fail_twice@T^{fail_old},t} \\ &+ \left(1 - \sum_{t=T^{fail_old}+1}^M p^{fail_new@t}\right) NPV^{fail_once@T^{fail_old}} \end{aligned} \quad (13)$$

- The NPV assuming the old component fails only at T^{fail_old} is:

$$\begin{aligned} NPV^{fail_once@T^{fail_old}} &= \sum_{\tau=1}^{T^{fail_old}} \frac{Cost^{TM,old}(\tau)}{(1+i)^\tau} + \frac{(Cost^{repl} + Cost^{penalty})}{(1+i)^{T^{fail_old}}} + \sum_{\tau=T^{fail_old}+1}^M \frac{Cost^{TM,new}(\tau)}{(1+i)^\tau} \end{aligned} \quad (14)$$

- The NPV assuming the old component fails at T^{fail_old} but also after the replacement at T^{fail_new} :

$$NPV_{fail_twice@T^{fail_old},T^{fail_new}}$$

$$= \sum_{\tau=1}^{T^{fail_old}} \frac{Cost^{TM,old}(\tau)}{(1+i)^\tau} + \frac{(Cost^{repl} + Cost^{penalty})}{(1+i)^{T^{fail_old}}} + \sum_{\tau=T^{fail_old}+1}^{T^{fail_new}} \frac{Cost^{TM,new}(\tau)}{(1+i)^\tau} \quad (15)$$

$$+ \frac{(Cost^{repl} + Cost^{penalty})}{(1+i)^{T^{fail_new}}} + \sum_{\tau=T^{fail_new}+1}^M \frac{Cost^{TM,new}(\tau)}{(1+i)^\tau}$$

B.2 Replacement Planned

This appendix presents the cost model in discrete form assuming replacement planned (RP)

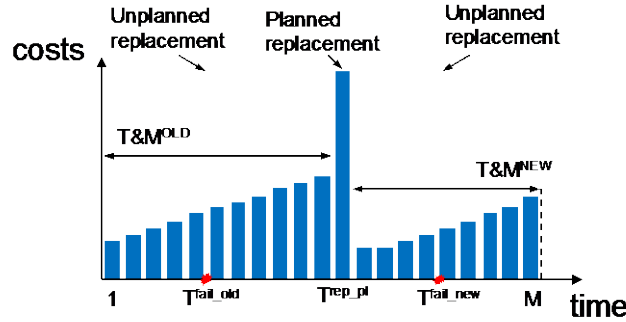


Figure 31. Plot of plant costs for SSC replacement case (discrete form).

In this case, the corresponding NPV can be calculated as:

$$NPV^{RP@T^{rep-pl}} = \sum_{t=1}^M p^{fail_old@t} \cdot NPV^{fail_old@t} + (1 - p^{fail_old}) NPV^{fail_post_T^{rep-pl}} \quad (16)$$

$$+ (1 - p^{fail_old})(1 - p^{fail_new}) NPV^{no_failure}$$

where:

- SSC fails prior replacement at time t:

$$NPV^{fail_old@t} = \sum_{\tau=1}^M p^{fail_new@t} \cdot NPV^{failure_pre@t,failure_post@t} \quad (17)$$

$$+ (1 - p^{fail_new}) NPV^{failure_pre@t,no_failure_post}$$

- SSC fails prior replacement at time t and no failure after replacement:

$$NPV^{failure_pre@t,no_failure_post}$$

$$= \sum_{\tau=1}^t \frac{Cost^{TM,old}(\tau)}{(1+i)^\tau} + \frac{(Cost^{repl} + Cost^{penalty})}{(1+i)^t} + \sum_{\tau=T^{rep-pl}+1}^M \frac{Cost^{TM,new}(\tau)}{(1+i)^\tau} \quad (18)$$

- SSC fails prior replacement at time t and it fails after replacement at time τ :

$$\begin{aligned}
NPV^{failure_pre@t, failure_post@t} = & \\
= & \sum_{\delta=1}^t \frac{Cost^{TM,old}(\delta)}{(1+i)^\delta} + \frac{(Cost^{repl} + Cost^{penalty})}{(1+i)^t} + \sum_{\delta=T^{rep.pl}+1}^t \frac{Cost^{TM,new}(\delta)}{(1+i)^\delta} \\
& + \frac{(Cost^{repl} + Cost^{penalty})}{(1+i)^t} + \sum_{\delta=t}^M \frac{Cost^{TM,new}(\delta)}{(1+i)^\delta}
\end{aligned} \tag{19}$$

- NPV assuming SSC fails after scheduled replacement time:

$$NPV^{fail_post_T^{rep.pl}} = \sum_{t=T^{rep.pl}+1}^M NPV^{failure_post@t} \cdot p^{fail_new@t} \tag{20}$$

- NPV assuming SSC fails at time t after scheduled replacement time:

$$\begin{aligned}
NPV^{failure_post@t} = & \\
= & \sum_{\tau=1}^{T^{rep.pl}} \frac{Cost^{TM,old}(\tau)}{(1+i)^\tau} + \frac{(Cost^{repl})}{(1+i)^{T^{rep.pl}}} + \sum_{\tau=T^{rep.pl}+1}^t \frac{Cost^{TM,new}(\tau)}{(1+i)^\tau} \\
& + \frac{(Cost^{repl} + Cost^{penalty})}{(1+i)^t} + \sum_{\tau=t+1}^M \frac{Cost^{TM,new2}(\tau)}{(1+i)^\tau}
\end{aligned} \tag{21}$$

- NPV assuming SSC does not fail:

$$\begin{aligned}
NPV^{no_failure} = & (1 - p^{fail_old}) \\
& \cdot \left(\sum_{\tau=1}^t \frac{Cost^{TM,old}(\tau)}{(1+i)^\tau} + \frac{(Cost^{repl} + Cost^{penalty})}{(1+i)^t} + \sum_{\tau=t+1}^M \frac{Cost^{TM,new}(\tau)}{(1+i)^\tau} \right)
\end{aligned} \tag{22}$$

APPENDIX C - NPV DISTRIBUTIONS FOR THE REPLACEMENT USE CASE

This appendix contains a more exhaustive summary of the NPV distributions for the capital SSC replacement use case. The objective is to show the NPV distribution for different replacement times and different estimated RUL of a single SSC.

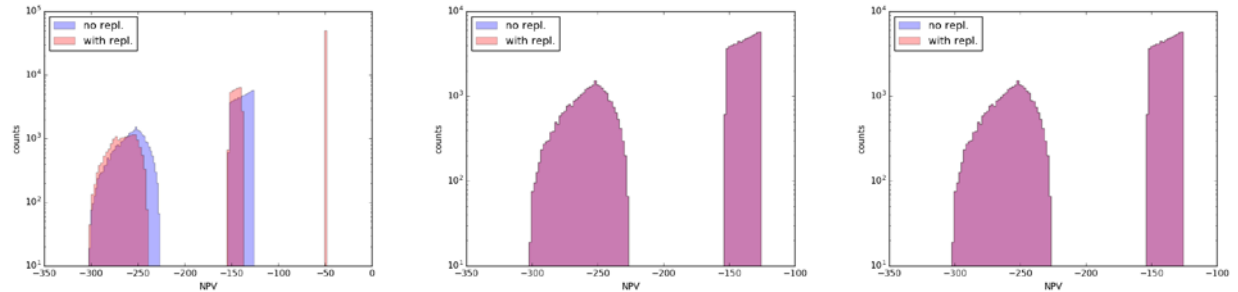


Figure 32. NPV distributions assuming SSC RUL in $[0, 10]$ for three replacement times (5, 10, 15 years).

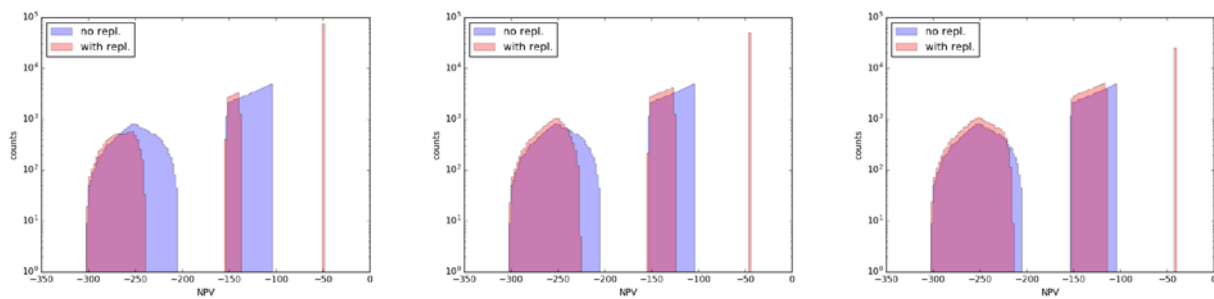


Figure 33. NPV distributions assuming SSC RUL in $[0, 20]$ for three replacement times (5, 10, 15 years).

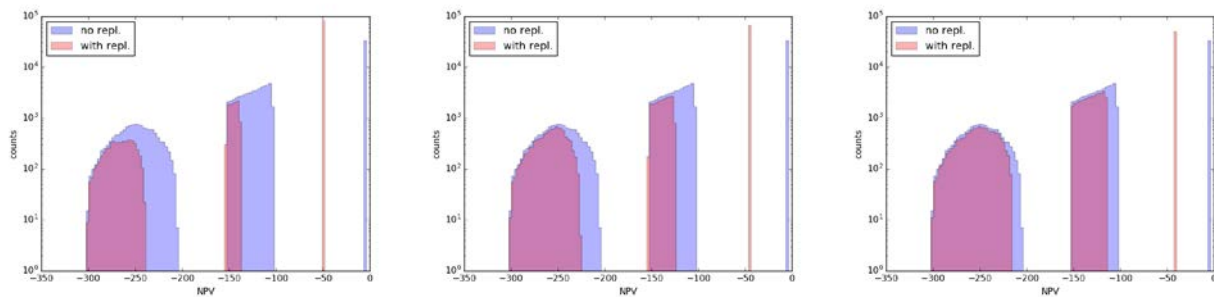


Figure 34. NPV distributions assuming SSC RUL in $[0, 30]$ for three replacement times (5, 10, 15 years).

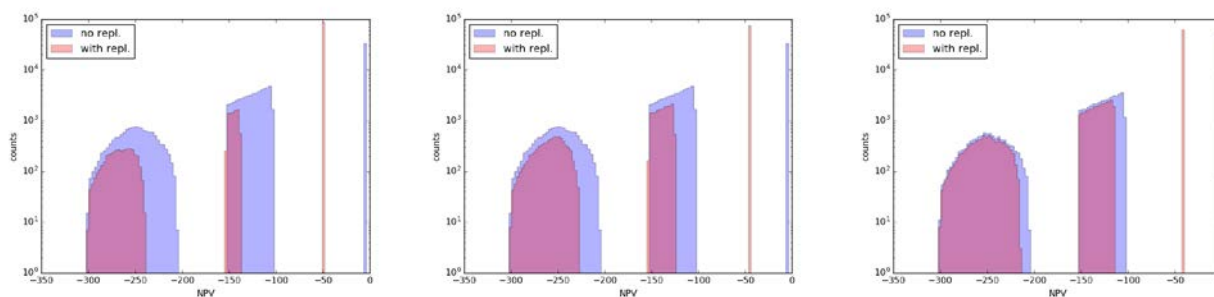


Figure 35. NPV distributions assuming SSC RUL in $[0, 40]$ for three replacement times (5, 10, 15 years).

APPENDIX D - RAVEN SAPHIRE INTERFACE

SAPHIRE version 8 (i.e. the current version) is equipped with a feature that allows the usage of MACROs to run specified analysis steps in a medium outside of SAPHIRE. This feature is useful for tedious and repetitive analysis procedures in several databases, as well as the coupling with RAVEN. It allows RAVEN to modify the parameters of SAPHIRE PRA models and to manage the analysis procedures conducted in SAPHIRE. Based on this feature, we have developed a SAPHIRE code interface that can be used to by RAVEN to drive SAPHIRE simulations.

The procedure of SAPHIRE with RAVEN is a straightforward process. For all the codes currently supported by RAVEN (e.g. RELAP5-3D, MAAP, MELCOR), the coupling is performed through a Python interface that interprets the information coming from RAVEN and translates them into the input of the driven code. The coupling procedure does not require modifying RAVEN itself. Instead, the developer creates a new Python interface that is going to be embedded in RAVEN at run-time (no need to introduce hard-coded coupling statements). The SAPHIRE interface is placed in a folder located in:

Path/to/raven/framework/CodeInterfaces/Saphire/

As mentioned, the MACRO with file extension “.mac” is used to modify the parameters, such as failure rates and failure probabilities, and run specified analysis steps, which is a required input for RAVEN. The other file that is required by RAVEN is SAPHIRE compressed project inputs with file extension “.zip”. The compressed file is used here to make it easier to load the inputs within RAVEN. These two files can be specified within the `<Files>` block in RAVEN XML input file as shown in the following:

```
<Files>
  <Input name="macro" type="">changeSet.mac</Input>
  <Input name="saphireInput" type="">saphireInput.zip</Input>
</Files>
```

All the information that is required to execute the SAPHIRE PRA models should be provided within `<Code>` block in the RAVEN XML input file. The following is an example and the details can be found in RAVEN user manual [3]:

```
<Models>
  <Code name="saphire" subType="Saphire">
    <executable>"C:\Saphire_8\tools\SAPHIRE.exe"</executable>
    <clargs arg="macro" extension=".mac" type="input"
      ↪ delimiter="="/>
    <clargs arg="project" extension=".zip" type="input"
      ↪ delimiter="="/>
    <outputFile>fixed_output.csv</outputFile>
    <codeOutput type="uncertainty">et_uq.csv</codeOutput>
    <codeOutput type="uncertainty">ft_uq.csv</codeOutput>
  </Code>
</Models>
```

This XML block will tell SAPHIRE interface to run the PRA model, i.e. saphireInput.zip, in the following way:

```
"C:\Saphire_8\tools\SAPHIRE.exe" project=path/to/
  ↪ saphireInput.zip macro=path/to/changeSet.mac
```

The output files, such as “et_uq.csv” and “ft_uq.csv”, generated by the SAPHIRE will be combined by the SAPHIRE interface to produce a CSV file named “fixed_output.csv” that can be collected by RAVEN. The perturbation of the parameters in MACRO is performed using the “wild-cards”, i.e. *\$RAVEN-variableName\$*, (placed in the original input files) for injecting the perturbed values. The following is an example of MACRO file using “wild-cards” to perturb the mean failure rates of basic events, i.e. the pumps fails to run:

```
<scenario>
  <start>
    <name>%P-51B</name>
    <description>Class Change Basic Events</description>
    <test>TEST-51B</test>
  </start>
</scenario>

<change set>
  <delete>
    <name>PMP-FR-EVENTS</name>
  </delete>
  <add>
    <name>PMP-FR-EVENTS</name>
    <description>Class change set PMP-FR</description>
    <class>
      <event name>?-PMP-FR-?</event name>
      <calc type>5</calc type>
      <lambda>$RAVEN-PMP-FR-lambda$</lambda>
      <mission time>2.4E01</mission time>
    </class>
  </add>
  <mark name>PMP-FR-EVENTS</mark name>
  <generate></generate>
</change set>

<sequence>
  <unmark></unmark>
  <include>
    <mark event tree mask>LOSP</mark event tree mask>
    <mark logic fault tree>*</mark logic fault tree>
  </include>

  <solve>
    <truncation>1.0E-15</truncation>
  </solve>

  <uncertainty>
    <sample>5000</sample>
    <seed>4321</seed>
    <method>lhs</method>
    <intermediate>csv</intermediate>
    <file name>etuq_im.csv</file name>
  </uncertainty>
</sequence>
<scenario><end></end></scenario>
```


APPENDIX E - TEXT DATA MINING

It is often necessary for a computer to categorize information based on data it is given. Perhaps an example for this would be predicting email as spam or not. If we were given 1,000 emails, with each one labeled as spam or not spam, we could extract certain ‘features’ from the data that might help us describe what does and does not make an email spam. These features could be something simple, such as the frequency of the word “buy” in each email, how long each sentence in the email is, the format of email address that sent it, etc.

Using a classification technique called Logistic Regression we can help the computer understand how to categorize this data. The Logistic Regression algorithm would look at each feature of each email and use the label “spam” or “not spam” to learn which features matter in determining the category of a particular email. Then, once we have ‘trained’ our model in this way, we test it against more labeled data. This time, the algorithm only looks at the features, tries to predict an email into one of the two categories, and we see how well it did. Different features or perhaps different algorithms can help tune the process. Logistic Regression comes from the Artificial Intelligence discipline of Machine Learning. Machine learning is “...the science of getting a computer to act without being explicitly programmed” (Ng, Andrew. Retrieved from: <https://www.coursera.org/learn/machine-learning>).

The main steps to perform the proposed Logistic Regression is shown below:

- Prepping data
 - Tagging
 - Extracting features (bag of words)
 - Normalizing
 - Splitting test/train
- Classification using Logistic Regression
 - Linear Regression hypothesis function
 - Logistic Regression modified hypothesis function
 - Cost function – Finding the right values for θ
- Example – Spam email dataset
 - Prepped data
 - Accuracy/weights
- Other potential algorithms

Raw data is often unstructured or poorly formatted. The first step in Logistic Regression is prepping the data, which will help our model better predict the category the data should belong to. Specifically, we need to tag the data, extract necessary features, normalize, and finally, split the data into testing and training sets.

Logistic Regression is a form of supervised machine learning. This means all the data we want to use to build a model, must be labeled, or tagged, with their specific categories. In the case of the spam email example, this would mean that each email we are going to examine, let’s say 1,000, must be labeled either spam, or not spam. This is done by hand, as we have yet to teach the computer how to do this step.

Next, we must extract necessary features. Features are pieces of the data that we are interested in examining that will be beneficial in helping the model decide whether an email is spam or not. A common feature that is extracted is in the form of a “bag of words” which counts the frequency of words in the sample data. For example, a feature in the spam email set might be the frequency of exclamation points, the frequency of the word “limited”, or the total number of capital letters in a row, etc. We define these features for each email we are going to use to build the model.

The features we extract might vary greatly in their values. This makes it difficult for the model to determine how much ‘weight’ a feature should have in classifying the document. Therefore, to get the best results, the features must be normalized. This makes each value of each feature approximately on the same scale, providing a better ground for the models custom weights that will be applied when we run the logistic regression algorithm.

When the data has been tagged and normalized, it needs to be split into testing and training data. We do our regression on the training data, which builds a model that tries to classify the emails. Then, we run our model against the testing data and see how accurate our model was.

Once the data has been prepped, we are ready to build the model. The following function is the hypothesis function, or, in other words, the function that predicts $h_{\theta}(x)$ based on values (x) and weights (θ) for each feature:

$$h_{\theta}(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \dots + \theta_n x_n$$

This function represents the basic form of regression called Linear Regression, while closely similar, linear and logistic regression share an important difference. The prediction of a linear regression function can be any real number, but the prediction of a logistic regression is constrained to a set. It doesn't make sense for our prediction to output values larger than 1 or smaller than 0. Logistic regression can only output 1 or 0 or eventually more categories. The following change to this function helps fix this issue and makes it a logistic function:

$$z = h_{\theta}(x) \quad g(z) = \frac{1}{1 + e^{-z}}$$

This function will output our prediction within an interval, such as all real values between 0 and 1. Which makes it much easier to classify these values being close to 0 or 1. However, how do we know what the values of the weights are? The cost function below helps us determine what are the best values to use for θ in order to 'fit' the data properly and provide the most accurate solution:

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m (g(x^{(i)}) - y^{(i)})^2$$

Where m is the total number of samples, x is the training data, and $J(\theta)$ is the output which represents the weights to associate to θ in our hypothesis function. After we have run the classification algorithm, we take our x value test data and run it through our model to make predictions with it, then we compare that result to our y test data and see how we did.

To illustrate with a real example, we might go about performing logistic regression in the following way:

1. Get some data, we got this data from <https://archive.ics.uci.edu/ml/datasets/Spambase>. This data has been prepped as explained above in the "prepping data" section. Each column is a feature, each value in each column has been normalized to scale, and each row representing each email is tagged as "spam" or "not spam" (shown here in the Y column as 1.00 or 0.00)

y	word_freq_make	word_freq_internet	word_freq_email	word_freq_credit	word_freq_money	char_freq_!	char_freq_\$	char_freq_#	capital_run_length_longest	capital_run_length_total
1.00	0.21	0.07	0.28	0.00	0.43	0.37	0.18	0.05	101.00	1028.00
1.00	0.06	0.12	1.03	0.32	0.06	0.28	0.18	0.01	485.00	2259.00
1.00	0.00	0.63	0.00	0.00	0.00	0.14	0.00	0.00	40.00	191.00
1.00	0.00	0.63	0.00	0.00	0.00	0.14	0.00	0.00	40.00	191.00
1.00	0.00	1.85	0.00	0.00	0.00	0.00	0.00	0.00	15.00	54.00
1.00	0.00	0.00	0.32	0.00	0.00	0.16	0.05	0.00	4.00	112.00

Figure 36. Email data set.

2. Next, we split the data into train and test data. This dataset has 4,000 samples (not shown). I will do a 75/25 split, train on 3,000, and test on 1,000.
3. Using a library such as scikit-learn with python, the above math equations are provided for us and we can perform the logistic regression on our training data. This builds our model, and learns based on our provided "y" labeled column about each email and the features we have extracted as being relevant or not.

4. After building our model, we now test it against the testing set we set aside earlier, in this example, we get a prediction accuracy of **92.50%**. The θ values (weights) that we calculated are shown in the following graph:

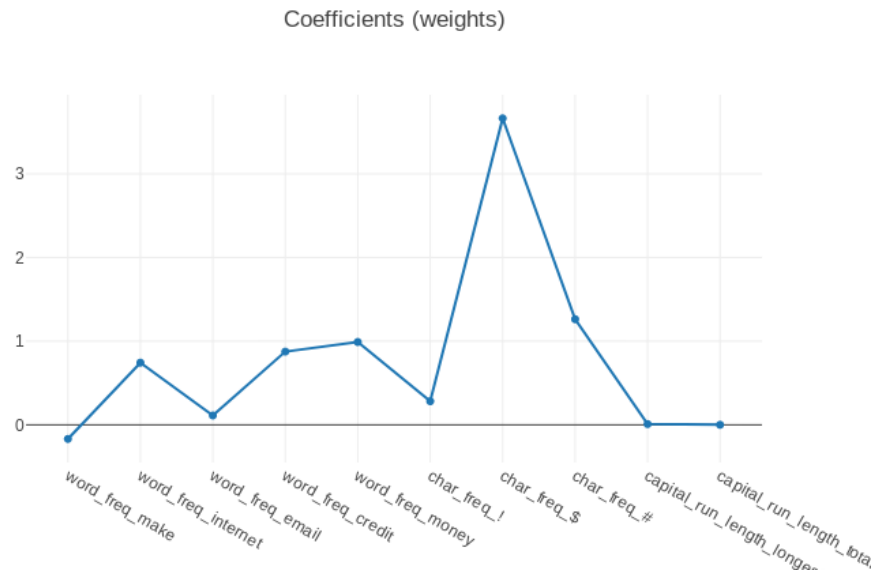


Figure 37. Logistic regression coefficients for the email data set.

Here, we can see that the frequency of the character “\$” has much more to say about an email being spam than the frequency of the word “make”.

While Logistic Regression is a well-documented and widely popular method, other methods are worth investigation as they may provide better results: Random Forest, Naïve Bayes, Support Vector Machines (SVM).