HYBRID DYNAMIC EVENT TREE SAMPLING STRATEGY IN RAVEN CODE

A.Alfonsi**, C. Rabiti, D. Mandelli, J. Cogliati, R. Kinoshita

2525 Fremont Ave, Idaho Falls, ID 83402, United States,
*** andrea.alfonsi@inl.gov*

The RAVEN code has been under development at the Idaho National Laboratory since 2012. Its main goal is to create a multi-purpose platform for the deploying of all the capabilities needed for Probabilistic Risk Assessment, uncertainty quantification and data mining analysis. RAVEN is currently equipped with three different sampling strategies: Once-through samplers (Monte Carlo, Latin Hyper Cube, Stratified and Grid Sampler), Adaptive Samplers (Adaptive Point Sampler) and Dynamic Event Tree samplers (Traditional and Adaptive Dynamic Event Trees).

The main subject of this paper is about the development of a Dynamic Event Tree (DET) sampler named "Hybrid Dynamic Event Tree" (HDET). As other authors have already reported, among the different type of uncertainties, it is possible to discern two principle types: aleatory and epistemic uncertainties. The classical Dynamic Event Tree is in charge of treating the first class (aleatory) uncertainties; the dependence of the probabilistic risk assessment and analysis on the epistemic uncertainties are treated by an initial Monte Carlo sampling (MCDET). From each Monte Carlo sample, a DET analysis is run (in total, N trees). The Monte Carlo employs a pre-sampling of the input space characterized by epistemic uncertainties. The consequent Dynamic Event Tree performs the exploration of the aleatory space.

In the RAVEN code, a more general approach has been developed, not limiting the exploration of the epistemic space through a Monte Carlo method but using all the once-through sampling strategies RAVEN currently employs. The user can combine a Latin Hyper Cube, Grid, Stratified and Monte Carlo sampling in order to explore the epistemic space, without any limitation. From this pre-sampling, the Dynamic Event Tree sampler starts its aleatory space exploration.

I. INTRODUCTION

RAVEN [1,2,3,4] (Risk Analysis and Virtual control ENviroment), under the support of the Nuclear Energy Advanced Modeling and Simulation (NEAMS) program, is increasing its capabilities to perform probabilistic analysis of stochastic dynamic systems. This supports the goal of providing the tools needed by the Risk Informed Safety Margin Characterization (RISMC) path-lead [5] under the Department of Energy (DOE) Light Water Reactor Sustainability program. In particular, the development of RAVEN in conjunction with the thermal-hydraulic code RELAP-7 [6], will allow the deployment of advanced methodologies for nuclear power plant (NPP) safety analysis at the industrial level. The investigation of accident scenarios in a probabilistic environment for a complex system (i.e. NPPs) is not a minor task. The complexity of such systems, and a large quantity of stochastic parameters, lead to demanding computational requirements (several CPU/hour). Moreover, high consequence scenarios are usually located in low probability regions of the input space, making even more computational demands of the risk assessment process.

This extreme need for computational power leads to the necessity to investigate methodologies for the most efficient use of available computational resources, either by increasing effectiveness of the global exploration of input space, or by focusing on regions of interest (e.g. failure/success boundaries, etc.). Several publications [7,8,9,10] of the same author described the capability of RAVEN to perform the exploration of the uncertain domain (probabilistic space) through the support of the well-known Dynamic Event Tree (DET) approach and its evolution, the Adaptive Dynamic Event Tree (ADET) method.

This paper will focalize on a newer sampler strategy now available in the RAVEN framework: the Hybrid Dynamic Even Tree (HDET) approach.

This methodology and its implementation, in the RAVEN code, are discussed in this paper. In order to show the effectiveness of this methodology, a Station Black Out (SBO) scenario for a Pressurized Water Reactor, using RELAP-7 code, has been employed. The HDET approach will be used to focus the exploration of the input space toward the computation of the failure probability of the system (i.e. clad failure), under the presence of epistemic uncertainties.

This paper is organized in five additional sections. Section II provides a brief overview of RELAP-7 and RAVEN codes. Section III recalls the concept of the DET methodology. Section IV reports how the newer developed algorithm is employed. Section V is focused on the analysis performed on the PWR SBO, and, section VI draws the conclusions.

II. RAVEN AND RELAP-7 CODES

As already mentioned, the Hybrid Dynamic Event Tree method has been developed within the RAVEN code and it has been tested using the newer developed system code RELAP-7.

This section briefly reports a general description of the two codes.

II.A. RAVEN

RAVEN has been developed in a highly modular and pluggable way in order to enable easy integration of different programming languages (i.e., C++, Python) and, as already mentioned, coupling with any system code.
RAVEN is composed of three main software systems that can operate either in coupled or stand-alone mode:

- Control Logic System
- Graphical User Interface
- Probabilistic and Parametric framework

The control logic system and the Graphical User Interface were not crucial for the deployment of the methodology subject of this paper. For this reason, attention is focused on the probabilistic and parametric framework.

II.A.1. Probabilistic and Parametric framework

The probabilistic and parametric framework represents the core of the RAVEN analysis capabilities. The main idea behind the design of the system is the creation of a multi-purpose framework characterized by high flexibility with respect to the possible performable analysis. The framework must be capable of constructing the analysis/calculation flow at run-time, interpreting the user-defined instructions and assembling the different analysis tasks following a user specified scheme.
In order to achieve such flexibility, combined with reasonably fast development, a programming language naturally suitable for this kind of approach was needed: Python.
Hence, RAVEN is coded in Python and is characterized by an object-oriented design. The core of the analysis performable through RAVEN is represented by a set of basic components (objects) the user can combine, in order to create a custom analysis flow. A list of these components and a summary of their most important functionalities are reported as follows:

- *Distribution*: In order to explore the input/output space, RAVEN requires the capability to perturb the input space (initial conditions of a system code). The initial conditions, that represent the uncertain space, are generally characterized by probability distribution functions (PDFs), which need to be considered when a perturbation is applied. In this respect, a large library of PDFs is available.
- *Sampler*: A proper approach to sample the input space is fundamental for the optimization of the computational time. In RAVEN, a "sampler" employs a unique perturbation strategy that is applied to the input space of a system. The input space is defined through the connection of uncertain variables and their relative probability distributions.
- *Model:* A model is the representation of a physical system (e.g. Nuclear Power Plant); it is therefore capable of predicting the evolution of a system given a coordinate set in the input space.
- *Reduced Order Model (ROM):* The evaluation of the system response, as a function of the coordinates in the input space, is very computationally expensive, especially when brute-force approaches (e.g. Monte Carlo methods) are chosen as the sampling strategy. ROMs are used to lower this cost, reducing the number of needed points and prioritizing the area of the input space that needs to be explored. They can be considered as an artificial representation of the link between the input and output spaces for a particular system.
- *Postprocessor:* In order to analyze the data generated from the exploration of the uncertain domain, post-processing capabilities are needed. Under this category, RAVEN collects all the statistical tools, data mining algorithms and uncertainty quantification capabilities.

The list above is not comprehensive of all the RAVEN framework components (visualization and storage infrastructure).

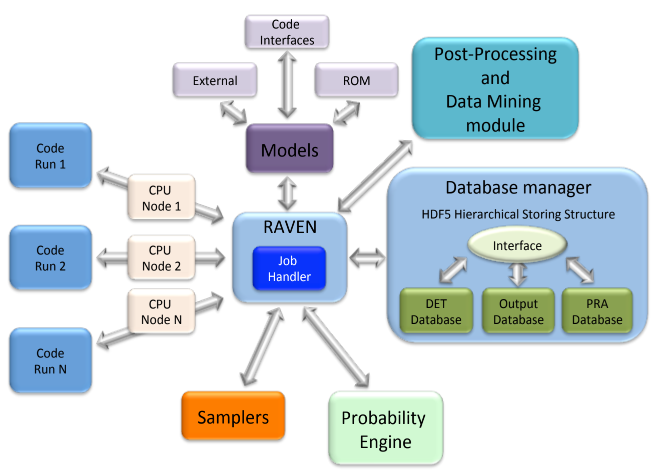

Figure 1 - RAVEN statistical framework layout.

Figure 1 shows a general overview of the elements that comprise the RAVEN statistical framework, including the ones not explained above.

II.B. RELAP-7

The RELAP-7 code [6] is the new nuclear reactor system safety analysis codes being developed at the Idaho National Laboratory (INL). RELAP-7 is designed to be the main reactor system simulation toolkit for the RISMC Pathway of the Light Water Reactor Sustainability (LWRS) Program). The RELAP-7 code development is taking advantage of the progress made in the past several decades to achieve simultaneous advancement of physical models, numerical methods, and software design. RELAP-7 uses the INL's MOOSE (Multi-Physics Object-Oriented Simulation Environment) framework [11] for solving computational engineering problems in a well planned, managed, and coordinated way. This allows RELAP-7 development to focus strictly on systems analysis-type physical modeling and gives priority to retention and extension of RELAP5-3D's multidimensional system capabilities [12].

A real reactor system is very complex and may contain hundreds of different physical components. Therefore, it is impractical to preserve real geometry for the whole system. Instead, simplified thermal hydraulic models are used to represent (via "nodalization") the major physical components and describe major physical processes (such as fluid flow and heat transfer). There are three main types of components developed in RELAP-7: (1) one-dimensional (1-D) components, (2) zero-dimensional (0-D) components for setting a boundary, and (3) 0-D components for connecting 1-D components.

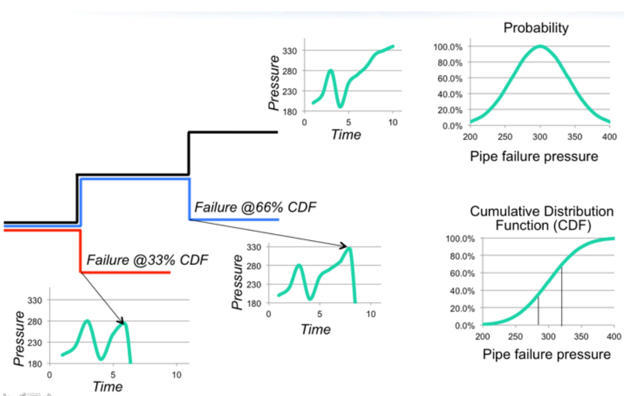

Figure 2 - Dynamic Event Tree conceptual flow.

III. THE DYNAMIC EVENT TREE METHOD

The DET method has been developed in order to solve the issues connected with traditional event tree based methodologies and the heavy computational power request of the most used technique to explore the uncertain space, Monte Carlo method. Conventional ET based methodologies are extensively used as tools to perform reliability and safety assessment of complex and critical engineering systems. One of the disadvantages of these methods is that timing/sequencing of events and system dynamics is not explicitly accounted for in the analysis. In order to overcome these limitations a "dynamic" approach is needed. The DET technique brings several advantages [7,8], among which the fact that it simulates probabilistic system evolution in a way that is consistent with the severe accident model.

In DET, event sequences are run simultaneously starting from a single initiating event. The branches occur at user specified times and/or when an action is required by the operator and/or the system, creating a deterministic sequence of events based on the time of their occurrence (see Fig. 1). This leads to a more realistic and mechanistically consistent analysis of the system taken in consideration. The DPRA, in general, and the DET methodologies, in particular, are designed to take the timing of events explicitly into account, which can become very important especially when uncertainties in complex phenomena are considered. The main idea of this methodology is to let a system code (e.g., RELAP-7, etc.) determine the pathway of an accident scenario within a probabilistic "environment". Figure 1 schematically shows the DET logic. As already mentioned, the accident sequence starts with an initiating event. Based on an user defined branching logic, driven by Probabilistic Distribution Functions (PDFs), an event occurs at a certain time instant. The simulation spoons n different branches. In each of them, the branching event determines a different consequence (carrying on associated probabilities). Each sequence continues until another event occurs and a new set of branching is spooned. The simulation ends when an exit condition or a maximum mission time is reached.

IV. THE HYBRID DYNAMIC EVENT TREE METHOD

In the uncertainty quantification field, the uncertainties are generally collected in two main types:

- *Aleatory uncertainties:* uncertainty due to inherent variation or randomness and can occur among members of a population or due to spatial or temporal variations. Aleatory uncertainty is generally characterized by a probability distribution, most commonly as either a probability density function (PDF) – which quantifies the probability density at any value over the range of the random variable – or a cumulative distribution function (CDF) – which quantifies the probability that a random variable will be less than or equal to a certain value;
- *Epistemic uncertainties:* uncertainty that arises due to a lack of knowledge on the part of the analyst, or team of analysts, conducting the modeling and simulation. If knowledge is added (through experiments, improved numerical approximations, expert opinion, higher fidelity physics modeling, etc.) then the uncertainty can be reduced. If sufficient knowledge is added, then the epistemic uncertainty can, theoretically, be eliminated. Epistemic

uncertainty is traditionally represented as either an interval with no associated probability distribution or a probability distribution, which represents degree of belief of the analyst, as opposed to frequency of occurrence (aleatory uncertainty).

Currently, the “simulation based” exploration of the uncertain domain (aleatory and epistemic) is generally performed through Monte-Carlo or Latin Hyper Cube methodologies. Indeed, from a practical point of view, the usage of these methodologies allows avoiding the necessity to discriminate between epistemic and aleatory uncertainties, since those are treated following the same approach (i.e., random sampling).

As already mentioned, Monte-Carlo based methodologies are extremely computational expensive. In order to overcome the computational burden of these methods, the Dynamic Event Tree approach has been developed. As understandable from previous section, the DET methodologies are particularly indicated to treat uncertainties that lead to state transitions of the analyzed system. In other words, events characterized by a time and a system state change, each of which may be either deterministic or aleatory and discrete, can be treated by the DET methods.

As already stated, the epistemic uncertainties represent the lack of knowledge of the analyst that performs the modeling and simulation. From a practical point of view, these uncertainties generally affect the accuracy of the parameters that represent the model of the system is going to be analyzed. A typical parameter affected by epistemic uncertainty, in the modeling of the thermal-hydraulic systems, is the friction factor of the piping system, which is generally dependent on the flow regime and computed through empirical correlations.

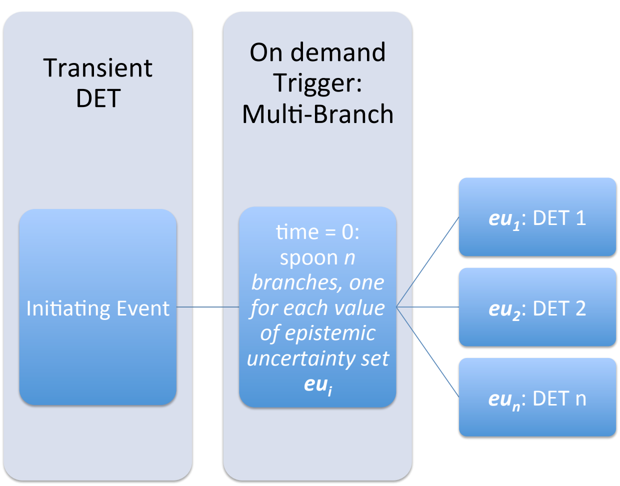

Figure 3 - Epistemic treatment with basic DET method.

As already mentioned, the DET method is particularly indicated for the treatment of the aleatory uncertainties, since they can represent events that might happen during an accident/transient scenario. On the other hand, the epistemic uncertainties systematically affect all the calculations, being translated in bias of the parameters using in the modeling of the system. From a DET point of view, the epistemic uncertainties can already be treated, by the basic DET approach, requesting a multi-branch trigger on demand: As shown in Figure 3, at the begin of the simulation (i.e. time = 0.0), the DET is triggered by an user-inputted on-demand multi-branch on the uncertain (epistemic) parameters' vector ***eu*** (*n* branches with a different values of **eu**: eu_i). From each branch, a standard DET (see Section III) begins. Hence, the DET methodology already let the user treat the epistemic uncertainties using the already available capabilities. Obviously, this “by-hand” approach can become cumbersome when a large number of epistemic uncertainties need to be taken in account: the method needs to be automatized.

In order to make the DET capable to handle these types of uncertainties automatically, the method needs to be upgraded embedding point sampling strategies (e.g. Monte-Carlo, etc.). This upgrading determined the creation of the sampling strategy that has been named, by the author, “Hybrid Dynamic Event Tree” (HDET) method.

The HDET method represents an evolution of similar methodologies (e.g. MCDET [13]) for the simultaneous exploration of the epistemic and aleatory uncertain space. In these methods the uncertainties are generally treated employing a Monte-Carlo sampling approach (epistemic) and DET methodology (aleatory). The HDET methodology, developed within the RAVEN code, can reproduce the capabilities employed by this approach, but provides additional sampling strategies to the user. The epistemic or epistemic-like uncertainties can be sampled through the following strategies:

- Monte-Carlo;
- Grid sampling;
- Stratified (e.g., Latin Hyper Cube).

Figure 4 schematically shows how conceptually the HDET methodology works. The user defines the parameters that need to be sampled by one or more different approaches. The HDET module samples those parameters creating a N-Dimensional Grid characterized by all the possible combinations of the input space coordinates coming from the different sampling strategies. Each coordinate in the input space represents a separated and parallel standard Dynamic Event Tree exploration of the uncertain domain.

The HDET methodology allows the user to completely explore the uncertain domain employing one methodology. The addition of Grid sampling strategy among the approaches usable, allow the user to perform a discrete parametric study, under aleatory and epistemic uncertainties.

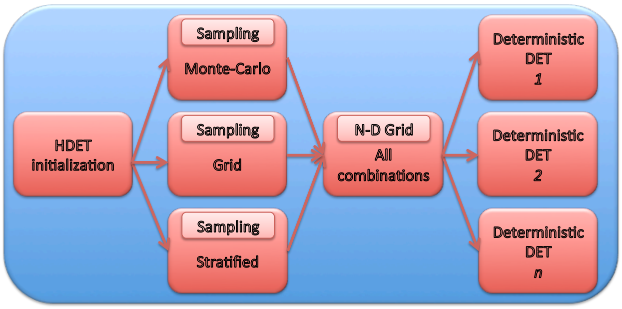

Figure 4 - Hybrid Dynamic Event Tree scheme.

IV. PROOF OF CONCEPT: PRA ANALYSIS ON A SIMPLIFIED PWR-LIKE MODEL

IV.A. PWR SYSTEM

A PWR simplified model has been set up based on the parameters specified in the OECD main steam line break (MSLB) benchmark problem [14]. The reference design for the OECD MSLB benchmark problem is derived from the reactor geometry and operational data of the TMI-1 Nuclear Power Plant (NPP), which is a 2772 MW two loop pressurized water reactor (see the system scheme shown in Fig. 5).

In order to simulate a SBO initiating event, the following electrical systems have been considered (see Fig. 6):

- Primary and auxiliary power grid lines (500 KV and 161 KV) connected to the respectively switchyards
- Set of 2 diesel generators (DGs), DG1 and DG2, and associated emergency buses
- Electrical buses: 4160 V (step down voltage from the power grid and voltage of the electric converter connected to the DGs) and 480 V for actual reactor components (e.g., reactor cooling system)
- DC system, which provides power to instrumentation and control components of the plant. It consists of these two sub-systems: battery charger and AC/DC converter and DC batteries.

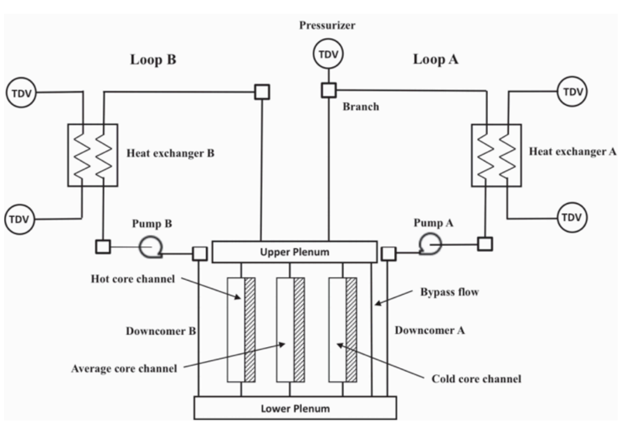

Figure 5 - Scheme of the TMI PWR benchmark.

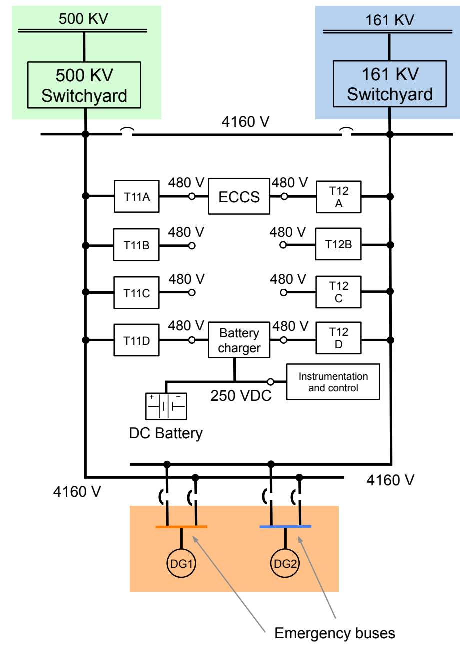

Figure 6 - Scheme of the electrical system.

IV.B. PLANT MECHANISTIC MODELING

As already mentioned, the analysis has been performed using RELAP-7 as system code. The modeling of the nuclear power plant has been performed using different basic components (see Fig. 7). The reactor vessel model consists of the Down-comers, the Lower Plenum, the Reactor Core Model and the Upper Plenum. Three Core-Channels (components with a flow channel and a heating structure) were used to describe the reactor core. Each Core-Channel is representative of a region of the core (from one to thousands of real cooling channels and fuel rods).

In this analysis, the core model consists of three parallel Core-Channels (hot, medium and cold) and one bypass flow channel. Respectively they represent the inner and hottest zone, the mid and the outer and colder zone of the core. The Lower Plenum and Upper Plenum are modeled with Branch models.

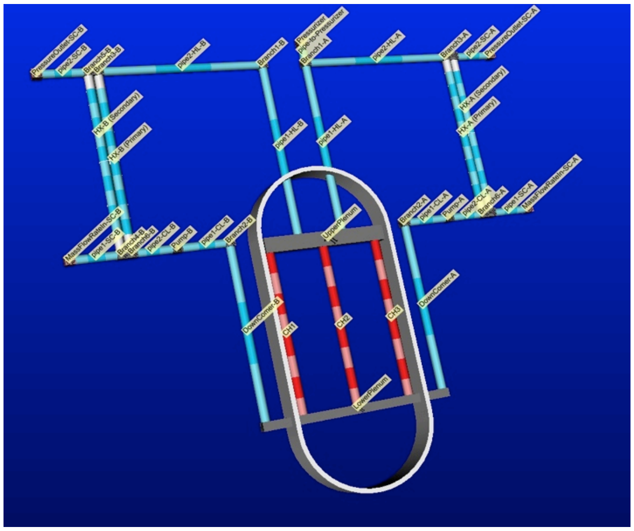

Figure 7 - PWR nodalization.

There are two primary loops in this model – Loop A and Loop B. Each loop consists of the Hot Leg, a Heat Exchanger and its secondary side pipes, the Cold Leg and a primary Pump. A Pressurizer is attached to the Loop-A piping system to control the system pressure. Since a complex Pressurizer model has not been implemented yet in the current version of RELAP-7 code, a time dependent volume (pressure boundary conditions) is used.

IV.C SBO SCENARIO

The scenario considered is a loss of off-site power (LOOP) initiating event caused by an earthquake (in proximity of the NPP), followed by tsunami induced flooding. The wave height is such that it causes water to enter into the air intake of the DGs and temporary disable the DGs themselves. In more detail, the scenario is the following:

1. An external event (i.e., earthquake) causes a LOOP due to damage of both 500 kV and 161 kV lines; the reactor successfully scrams and, thus, the power generated in the core follows the characteristic exponential decay curve
2. A tsunami wave hits the plant causing flooding of the plant itself. The wave causes the DGs to fail and may also flood the 161 kV switchyard. Hence, conditions of SBO are reached (4160 V and 480 V buses are not energized); all core cooling systems are subsequently off-line (including the ECCS system)
3. Without the ability to cool the reactor core, its temperature starts to rise
4. In order to recover AC electric power on the 4160 V and 480 V buses, three strategies are followed:
   - A plant recovery team is assembled in order to recover one of the two DGs
   - The power grid owning company is working on the restoration of the primary 161 kV line
   - A second plant recovery team is also assembled to recover the 161 kV switchyard if flooded
5. When the 4160 kV buses are energized (through the recovery of the DGs or 161 kV line), the auxiliary cooling system (i.e., ECCS system) is able to cool the reactor core and, thus, core temperature decreases

IV.D STOCHASTIC PARAMETERS

As mentioned in the previous section, in this scenario, the cooling of the reactor, through the ECCS system, is ensured when either the DGs or the 161 kV line are restored. Since the purpose of this paper is to demonstrate the development of the HDET methodology in the RAVEN code, both the DGs and the 161 kV line recovery times are collapsed in one single stochastic parameter (t_{ECCS}).

The failure of the plant occurs when the cladding in the core reaches its failure temperature. The clad failure temperature value ($T_{failure}$) is considered stochastic as well.

IV.E EPISTEMIC PARAMETERS

In order to show the HDET capabilities and its way to treat epistemic or "epistemic-like" uncertainties by the combination of its main sampling strategies (i.e. Monte-Carlo, Stratified and Grid samplers), 3 different epistemic or "epistemic-like" parameters have been considered.

As already mentioned, one of the most common epistemic uncertainties taken in account in such analysis are the friction factors of the piping network. As proof of concept, in the demo case here presented, the friction factors of the three core channels have been considered affected by uncertainties.

Another important source of uncertainties, in this kind of scenarios, is represented by the "accuracy" of the model used to simulate, right after the "scram" of the reactor, the decay heat generation. If it is not computed by appropriated Burn-Up codes [15], its evolution, right after the scram, is generally modeled by user-input approximated exponential decay curves (Decay Power vs. time). In RELAP-7 (i.e., in its module CROW), a set of predefined curves is already implemented. For this analysis, a curve characterized by the following equation has been employed:

$$\begin{aligned} P_{decay}(t) = P_0 * \alpha \\ &* \Big\{(t + t_{start} + 10.0)^{-0.2} \\ &- \left(t + t_{op} + 10.0\right)^{-0.2} - 0.87 \\ &* \big[(t + t_{start} + 10^7)^{-0.2} \\ &- (t + t_{start} + t_{op} + 2 * 10^7)^{-0.2}\big]\Big\} \end{aligned} \quad (1)$$

where, P_0 is the initial power, α is the power coefficient (% of decay power with respect to the nominal power), t_{start} is the scram time, t_{op} is the time the NPP has been at P_0 power level. In order to investigate the effects of the approximations in this modeling strategy, the power coefficient α has been considered as an epistemic uncertainty.

Finally, the third parameter that has been considered is one that can be included among the so-called "epistemic-like" uncertain factors: the operational power. Even if it is not a real epistemic uncertainty, it is interesting to investigate, in a parametric fashion, the effects of perturbations on the operational power level.

IV.F PROBABILITY DISTRIBUTION FUNCTIONS AND APPLIED SAMPLING STRATEGIES

As stated in sections IV.D and IV.E, 5 sources of uncertainty (epistemic and aleatory) have been considered in this analysis.

Regarding the parameters affected by aleatory uncertainty, the following probability distribution functions have been used:

- ECCS recovery time (t_{ECCS}):
  - **Normal distribution,**
    - Mean: 3125 seconds
    - Sigma: 850 seconds
- Clad Failure Temperature ($T_{failure}$):
  - **Triangular distribution,**
    - Peak: 1477.59 K
    - Lower: 1255.37 K
    - Upper: 1699.81 K

The recovery of the ECCS system has been considered being "faster" that the reality (mean ~hours) since the transient timing has been shrunk. This was needed because the computational time of RELAP-7 does not allow simulating too long transient yet.

The effects of both aleatory parameters have been explored, by the DET part of the HDET method, using a grid in probability characterized by the following thresholds: 0.01, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.85, 0.9, 0.95. This means that a branch occur when either the clad temperature or the time corresponds to a CDF equal or bigger than the probability threshold imposed by the sampler.

On the other hand, regarding the parameters affected by aleatory uncertainty, the following probability distribution functions have been used:

- Friction scaling factor ($f_{scaling}$):
  - **Truncated Normal distribution,**
    - Mean: 1.0 (-)
    - Sigma: 0.2 (-)
    - Lower: 0.5 (-)
    - Upper: 1.5 (-)
- Decay heat curve power coefficient ($\alpha_{scaling}$):
  - **Truncated Normal distribution,**
    - Mean: 1.0 (-)
    - Sigma: 0.2 (-)
    - Lower: 0.5 (-)
    - Upper: 1.5 (-)
- Power scaling factors ($P_{scaling}$):
  - **Uniform distribution,**
    - Lower: 1.0 (-)
    - Upper: 1.2 (-)

As can be inferred from above, the same "sampled" multiplier $f_{scaling}$ scales the friction factors of the three core channels.

The parameters above have been perturbed through the following strategies (Table 1):

Table 1 - Epistemic Sampling settings.

Parameter	Sampler	Grid Size	Samples
$f_{scaling}$	Grid	0.33 CDF	4
$\alpha_{scaling}$	MonteCarlo	-	4
$P_{scaling}$	Stratified	0.33 CDF	4

This approach led to a 3-Dimensional grid of 64 combinations and, thus, 64 parallel DET simulations spooned.

IV.G RESULTS

The HDET analysis has been performed, obtaining ~2100 branches resulting in ~1600 completed histories.

Figure 8 shows the clad temperature evolution for all the histories the HDET has simulated. The different colors represent the distinctive branches that have been simulated. It can be noticed that the temperature starts raising right after the initiating event and the raising paths are slightly diverging (the slope coefficients change). This behavior is connected to the sampling of the initial power level and decay curve coefficient.

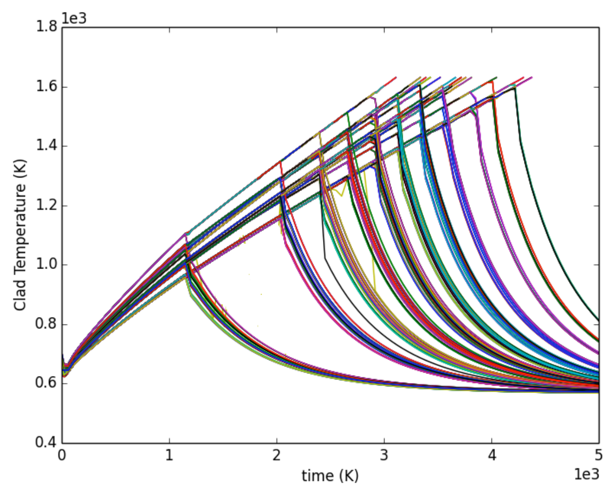

Figure 8 - Clad Temperature evolution.

Figures 9 and 10 show the flow velocity evolution in the hot and cold legs, respectively. In both figures it can be noticed the velocity oscillations right after the back up of the ECCS system. These oscillations are determined since the sudden insertion of the system. Indeed, for this analysis, the ECCS effect has been modeled by setting the pumps' head to 5% of the nominal one, without any ramp up approach.

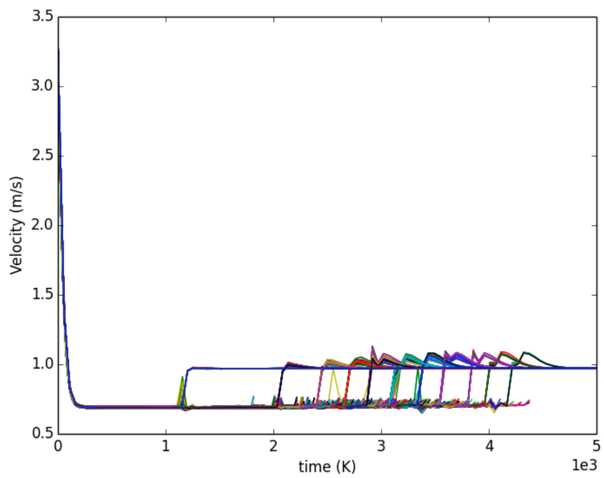

Figure 9 - Hot Leg flow velocity evolution.

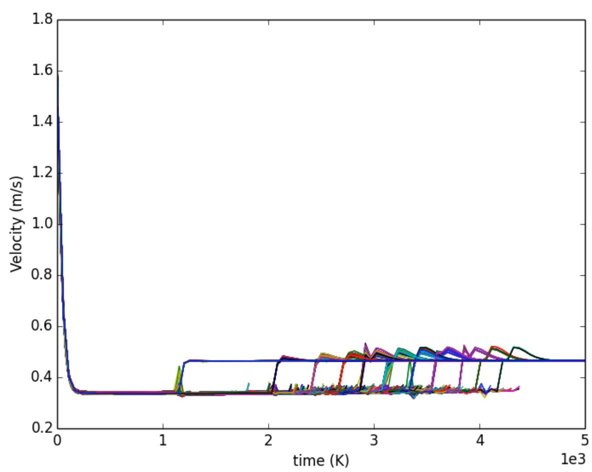

Figure 10 - Cold Leg flow velocity evolution.

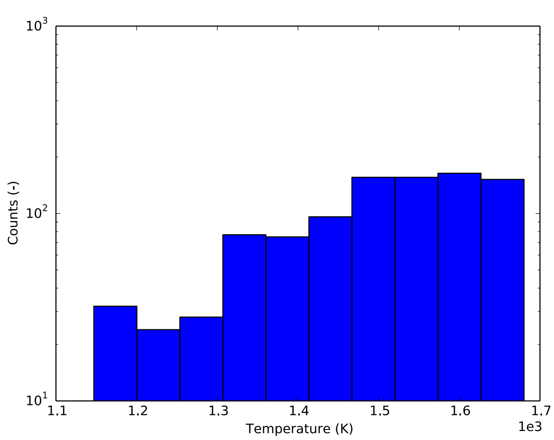

Figure 11 - Histogram maximum fuel temperature.

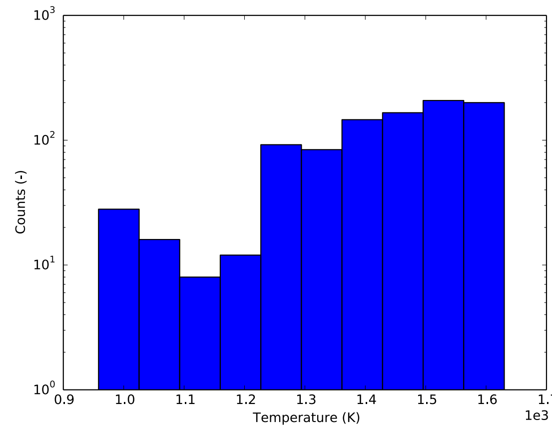

Figure 12 - Histogram maximum clad temperature.

Figures 11 and 12 show the histograms of the maximum temperatures reached by the fuel and clad, respectively. It can be noticed that the most populated bins are those at high temperatures, since, the clad failure temperature is a stochastic parameter, that is translated into a forward movement of the clad failure temperature threshold.

When such analyses are performed, they are generally concluded with the computation of the global probability of failure under the analyzed uncertainty. In our case, the probability of failure of this system is *9.33E-04*.

V. CONCLUSIONS

This paper presented RAVEN as a tool to perform dynamic PRA through the newer Hybrid Dynamic Event Tree methodology. The addition of this method improves the capability of the Dynamic Event Tree to explore the uncertain domain, under epistemic uncertainties and for parametric studies.

In particular, the software concept and all the main components that are involved in the computation have been described, including the used system simulator (e.g., RELAP-7). A proof of concept of a PRA analysis has been also shown for a SBO scenario for a simplified PWR loop. The description of the implementation for such case demonstrates how the flexibility of the software framework provides state-of-the-art tools to perform Dynamic PRA, uncertainty quantification and plant control.

The Hybrid Dynamic Event Tree methodology is part of an heavy development around the concept of DET and goal-oriented exploration of the uncertain domain [7,8,9,10].

The implementation of this methodology determines the basis for a future class of algorithms, under development by the author, that are designed to exploit the intrinsic characteristics of DET based methods.

ACKNOWLEDGMENTS

This work is supported by the U.S. Department of Energy, under DOE Idaho Operations Office Contract DE-AC07-05ID14517. Accordingly, the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes.

REFERENCES

1. A. Alfonsi, C. Rabiti, D. Mandelli, J. Cogliati, and R. Kinoshita, "Raven as a tool for dynamic probabilistic risk assessment: Software overview," in Proceeding of M&C2013 International Topical Meeting on Mathematics and Computation, CD-ROM, American Nuclear Society, LaGrange Park, IL (2013).
2. C. Rabiti, A. Alfonsi, D. Mandelli, J. Cogliati, R. Martinueau, C. Smith, "Deployment and Overview of RAVEN Capabilities for a Probabilistic Risk Assessment Demo for a PWR Station Blackout," Idaho National Laboratory report: INL/EXT-13-29510 (2013).
3. A. Alfonsi, C. Rabiti, D. Mandelli, J. Cogliati, R. Kinoshita, and A. Naviglio, "RAVEN and dynamic probabilistic risk assessment: Software overview," in Proceedings of ESREL European Safety and Reliability Conference (2014).
4. A. Alfonsi, C. Rabiti, D. Mandelli, J. Cogliati, R. Kinoshita, "Performing Probabilist Risk Assessment Through RAVEN", Proceedings American Nuclear Society 2013 Annual Meeting "Next Generation Nuclear Energy: Prospects and Challenges", Atlanta, GA (2013).
5. C. Smith, C. Rabiti, and R. Martineau, "Risk Informed Safety Margins Characterization (RISMC) Pathway Technical Program Plan", Idaho National Laboratory INL/EXT-11-22977 (2011).
6. David, R. Berry, D. Gaston, R. Martineau, J. Peterson, H. Zhang, H. Zhao, L. Zou, "RELAP-7 Level 2 Milestone Report: Demonstration of a Steady State Single Phase PWR Simulation with RELAP-7," Idaho National Laboratory: INL/EXT-12-25924 (2012).
7. A. Alfonsi, C. Rabiti, D. Mandelli, J. Cogliati, R. Kinoshita, A. Naviglio, "Dynamic Event Tree Analysis Through RAVEN", International Topical Meeting on Probabilistic Safety Assessment and Analysis (PSA 2013), September 22-26, Columbia, SC, USA, (2013).
8. A. Alfonsi, C. Rabiti, D. Mandelli, J. Cogliati, R. Kinoshita, "RAVEN: Dynamic Event Tree Approach. Level III Milestone", Idaho National Laboratory, Idaho Falls, Idaho, INL/EXT-13- 30332, (2013).
9. A. Alfonsi, C. Rabiti, D. Mandelli, J. Cogliati, and R. Kinoshita, "RAVEN: Development of the adaptive dynamic event tree approach," Tech. Rep. INL/MIS-14-33246, Idaho National Laboratory (INL), (2014).
10. A. Alfonsi, C. Rabiti, D. Mandelli, J. Cogliati, and R. Kinoshita, "Adaptive Dynamic Event Tree in RAVEN code," Proceedings American Nuclear Society 2014 Winter Meeting "Nuclear-The Foundation of Clean Energy", Anaheim, CA (2014).
11. D. Gaston, C. Newman, G. Hansen and D. Lebrun-Grandi, "MOOSE: A parallel computational framework for coupled systems of nonlinear equations," Nuclear Engineering Design, 239, pp. 1768-1778, (2009).
12. The RELAP5-3D© Code Development Team, "RELAP5-3D© code manual volume I: code structure, system models, and solution methods", Idaho National Laboratory Report , INEEL-EXT-98-00834, (2005).
13. M. Kloos, J. Peschke, "MCDET: A Probabilistic Dynamics Method Combining Monte Carlo Simulation with the Discrete Dynamic Event Tree Approach," Nuclear Science and Engineering, 153,137-156, (2006)
14. "Pressurized Water Reactor Main Steam Line Break (MSLB) Benchmark", Volume I: Final Specifications, NEA/NSC/DOC(99)8.
15. A. Alfonsi, C. Rabiti, A. Epiney, Y. Wang, J. Cogliati, "PHISICS Toolkit: Multi-Reactor Transmutation Analysis Utility – MRTAU", Proc. PHYSOR 2012 Advances in Reactor Physics Linking Research, Industry, and Education, on CD-ROM, American Nuclear Society, LaGrange Park, Ill (2012), Knoxville, Tenn., April 15-20, 2012.