Discrete Dynamic Event Tree Uncertainty Quantification in the ADS-IDAC Dynamic PSA Software Platform

Mihai A. Diaconeasa[*] and Ali Mosleh
The B. John Garrick Institute for the Risk Sciences
Department of Mechanical Engineering
University of California, Los Angeles, USA

Abstract: ADS-IDAC stands for the Accident Dynamics Simulator coupled with the Information, Decision and Action in a Crew context. It contains both a cognitive crew model and a nuclear power plant thermal-hydraulic model to simulate their response behavior and interactions given abnormal conditions and generate a discrete dynamic event tree (DDET). When the probabilities of the initiating events and branching or non-branching events in a DDET are subject to uncertainty, the probabilities can be considered to be random variables described by some probability distribution. The form of their probability distribution depends on the type of events (e.g., hardware failure, human activity, etc.) Therefore, the probability of the end state events in such a DDET will also be a random variable, and the form of its probability distribution will depend both on the DDET structure and the probability distributions of the events. In this paper, the various sampling techniques (i.e., Monte Carlo, Latin Hypercube, quasi-Monte Carlo) that are implemented in an updated version of ADS-IDAC are summarized. They are used for the propagation of uncertainties in the DDET generated by ADS-IDAC. These Monte Carlo methods are used to obtain a probability distribution of the end state events in a DDET using available information on the tree structure and the assumed probability distributions of its top events. The same methods can be applied to simulate the fault trees (FTs) used to represent frontline and support systems. For these accident sequences, the propagation of uncertainties is performed on the combined structure of DDET and FTs.

Keywords: Dynamic PSA, Uncertainty Quantification, Simulation HRA.

1. INTRODUCTION

The Accident Dynamics Simulator coupled with the Information, Decision, and Action in a Crew context (ADS-IDAC) simulation platform falls into the category of discrete dynamic probabilistic safety assessment techniques in which the time-dependent changes in the system elements are simulated to create scenarios by stepping forward in time or branching to new sequences at discrete time steps following a relatively small set of generic branching rules. These model-based branching rules have been developed to not only constrain the simulation into a realistic solution space, but also to avoid the sequence explosion phenomenon given the large number of system states. The latest version of ADS-IDAC is limited to point-estimate quantification, thus its events are not subjected to uncertainty.

The two major types of uncertainties are aleatory and epistemic uncertainties. Aleatory uncertainties account for the randomness in the behavior of a system or crew, while epistemic uncertainties arise from a lack of knowledge of the systems, processes, or mechanisms. Dealing with the aleatory uncertainties is straightforward when data is available and allows probabilistic characterization. Probability-based approaches such as Monte Carlo simulations are typically used to describe uncertainties in input data and propagate them through fault trees (FTs) or event trees (ETs). This technique requires empirical input data information or expert judgement in the form of probability density functions of relevant parameters. In some cases, this information may not be available. As an alternative to empirical data that can be difficult to obtain, expert judgment is typically used. Moreover, it is known that Monte Carlo simulations can be computationally expensive, requiring efficient sampling techniques.

[*] Corresponding author's email address: diacon@ucla.edu.

Given the motivation for this work, the following tasks were completed: appropriate probability distributions for the human failure events (HFE) were selected, capability to select various probability distributions for the hardware failure events was created, and algorithms to propagate the uncertainties through the generated DDET were implemented. Moreover, the Hybrid Causal Logic (HCL) library, which is internally coupled into ADS-IDAC, had to be extended with new quantification features. These features include multiple types of basic event quantification models, sampling methods, and time-dependent importance measures.

2. OVERVIEW OF ADS-IDAC

To safely operate Nuclear Power Plants (NPPs), the crew are required to closely interact with the system and between themselves – especially under abnormal conditions. Although the NPPs are equipped with automated safety systems, the crew still need to perform complex activities during highly dynamic accident conditions. Their performance could be impaired by the inability to obtain information about the systems or the lack of time available to safely recover the NPP. The safety of NPPs can be improved by predicting, quantifying, and mitigating the conditions that could result in the crew making inappropriate decisions based on conflicting information or committing erroneous actions. Conventionally, the methodologies and techniques used to predict and quantify the HFEs (i.e., THERP) are static in time and include limited information about the cognitive context in which the crew perform. ADS-IDAC was and remains revolutionary for being the only simulation-based HRA method that not only attempts to quantify the time-dependent crew behavior, but also transparently predict it by simulating both the crew and NPP behavior. It is one of the most mature dynamic platforms with an evolution that spans more than 25 years as illustrated in Figure 1.

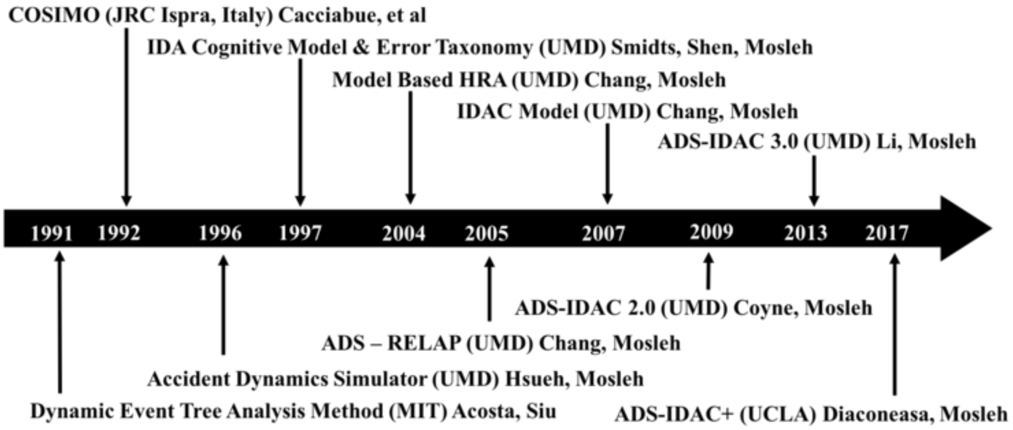
COSIMO (JRC Ispra, Italy) Cacciabue, et al
IDA Cognitive Model & Error Taxonomy (UMD) Smidts, Shen, Mosleh
Model Based HRA (UMD) Chang, Mosleh
IDAC Model (UMD) Chang, Mosleh
ADS-IDAC 3.0 (UMD) Li, Mosleh
1991
1992
1996
1997
2004
2005
2007
2009
2013
2017
ADS-IDAC 2.0 (UMD) Coyne, Mosleh
ADS – RELAP (UMD) Chang, Mosleh
Accident Dynamics Simulator (UMD) Hsueh, Mosleh
Dynamic Event Tree Analysis Method (MIT) Acosta, Siu
ADS-IDAC+ (UCLA) Diaconeasa, Mosleh

Figure 1 ADS-IDAC Development History [1-7]

ADS-IDAC is a simulation engine that includes a scheduler module, a hardware reliability module, a control panel module, and the IDAC operator response model coupled with the RELAP5/MOD3.3 thermal hydraulic code (the system model) to generate DDETs containing contextually rich scenarios that could occur given an initiating event. Its modular structure and the flow of information between modules are shown in Figure 2. A scheduler module coordinates the interactions between all the other modules and generates the DDETs. The probability of each scenario/sequence is calculated as the product of conditional probabilities of its constituent branches (as is the case for conventional ETs). The indicator module simulates the control panel indicators' states driven by information from the system module. The hardware reliability module simulates the failure probabilities of the system's and control panel's components. The IDAC operator response model requires either the heuristic cognitive engine or the reasoning module to guide operator decision-making. The cognitive engine forms a situational assessment from perceived information to identify and select suitable goals and strategies based on the situational assessment and to prioritize and resolve conflicts among the selected goal and strategy sets [6]. In the reasoning module, the operators' knowledge-based behavior is augmented through an embedded attention mechanism in information perception channels, better capturing cognitive resource limitations, and top-down attention control [7]. This module also simulates an operator 'making sense' of perceived information, connecting different pieces of information to form a comprehensive mental picture of the plant situation, and making accident diagnoses.

One of the new features that was implemented into the simulation engine was support systems integration by dynamically linking FTs [8]. This was achieved by linking the Control Panel Module to the HCL Module (Figure 2). In turn, this allowed the frontline systems to be also modeled with dynamic FTs when they could not be included into the thermal hydraulic model.

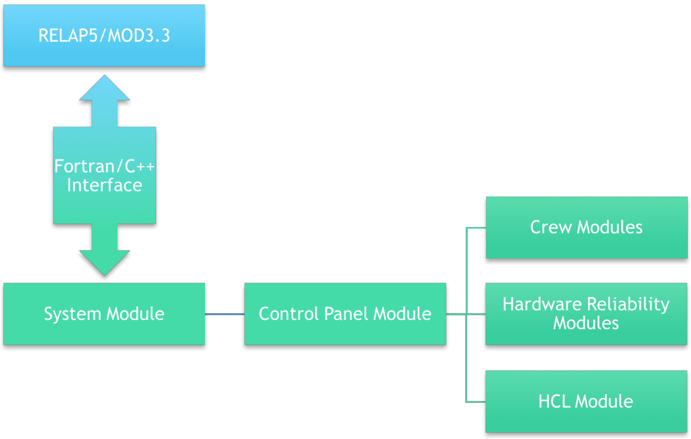

Figure 2 ADS-IDAC architecture

This coupling gave access to HCL's capability for creating, updating, and quantifying of FTs. Also, HCL supports creating, updating, and evaluating of Bayesian Networks (BNs). A BN does not require complete knowledge of the cause-and-effect relations between the random variables. HCL offers a natural platform for creating a BN for all the PSFs when more empirical data becomes available. Overall, the modeling of the impact of the situational and cognitive factors on the operator's behavior was improved.

However, both ADS-IDAC and the HCL library had to be expanded to support uncertainty propagation, even if the coupled HCL library was used as the basis for this work.

3. NEW QUANTIFICATION FEATURES

In this chapter, the new features implemented into the HCL library are described. New quantification models were included to allow the modelling of more advanced hardware failures. For uncertainty propagation, a number of sampling methods have been included into the HCL library. A range of importance measures have been implemented that can be used to assess the relative importance of different components in a system modeled with a FT.

3.1. Basic event quantification models

The following quantification models were implemented into the HCL library to expand the types of system failures that ADS-IDAC can simulate and provide the necessary probability distributions for uncertainty quantification: Uniform, Normal, Lognormal, Gamma, Noncentral chi-squared, Cauchy, Student's t, Exponential, and Weibull.

Time-dependent reliability results from third-party advanced models of component failure mechanisms (e.g., MATLAB simulation) may be used in the system analysis. If the results cannot be fitted to the common statistical distributions, they can be included in the analysis in the form of a nonparametric model given by the time-dependent reliability cumulative distribution function for each component. The data should be included in a '.csv' file containing two rows for time and reliability data pairs.

The first developmental task pursued was to create a generic expression parsing module with added distribution recognition and global variable functionalities by linking the C++ Mathematical Expression Parsing And Evaluation Library (called 'exprtk' and available for free use under the MIT license [http://partow.net/programming/exprtk/]). The exprtk library was extended to include the dictionary of the following commonly-used distribution functions: "uniform", "normal", "lognormal", "exponential", "weibull". These can be individually used to model uncertain failures on demand or be part of expressions defining complex failure mechanisms including time-to-failure or other user-defined global variables.

For example, temperature could be a global variable for an Arrhenius degradation model. A user could define the temperature to be sampled from a distribution or fix it to a certain value through the assignment operator. In this example, it is assumed the component time to failure is exponentially distributed with the failure rate given by an Arrhenius degradation model. The input XML snippet for this case is:

```
<expression expressionLiteral="T:=uniform(223.2,353.2);
1-exp(-(1/(5.57e-6*exp(8566/T)))*t)" testInterval="500.0"/>
```

In other component definitions that require the use of the same temperature variables, the user would not need to redefine it. In other words, each global variable needs to be defined only once, otherwise it will be overwritten.

3.2. Sampling methods

The sampling techniques presented here were used in HCL to perform uncertainty analysis on the discrete DDETs. All the sampling methods that have been implemented into this version of ADS-IDAC fall into the category of forward samplers because they do not learn from information gathered during the sampling of the system.

The HCL library includes Monte Carlo sampling (MCS) and three variants of Latin Hypercube sampling (LHS). The HCL library offers more advanced forward sampling strategies that use low-discrepancy sequences called quasi-Monte Carlo sampling (QMCS). The difference between these sampling methods can be seen in Figure 3.

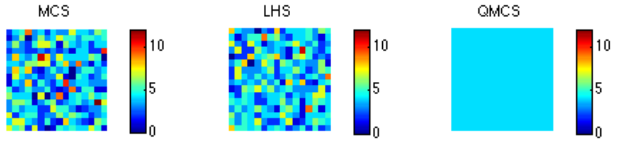

Figure 3 Density plot of Monte Carlo sampling (MCS), Latin hypercube sampling (LHS) and quasi-Monte Carlo sampling (QMCS) on a 16x16 grid with 1024 samples [9]

Monte Carlo simulation leverages the law of large numbers (amongst other things) to estimate the expectation using the sample mean of a function of a set of sampled random variables. To initialize the sampler, a priori knowledge of the needed number of samples or the number of dimensions is not required. Forward sampling generates pseudorandom numbers using the Mersenne Twister algorithm without considering the previously generated sample points. Monte Carlo is arguably the most used sampling strategy across multiple fields [10].

LHS requires the number of samples at initialization. The number of samples is then used to stratify the domain space into Latin squares such that each sample's location must be remembered to not be explored at future iterations. The following LHS methods have been implemented:

- LHC Center determines each subsquare's center that is equidistant from the square's corners or apexes,

- LHS Random determines each subsquare's center that is randomly located within the square's corners or apexes; and
- Improved Distributed LHS finds a set of samples that are optimally spread out in the domain space [11].

QMCS employs a quasi-random number generator. A quasi-random or low-discrepancy sequence (such as the Faure, Halton, Hammersley, Niederreiter or Sobol sequences) is less 'random' than a pseudorandom number sequence, but more useful for tasks such as uncertainty quantification in higher dimensions. This is because low discrepancy sequences tend to explore the space more evenly than random numbers as successive samples are generated in a position as far as possible from the previous samples. Low-discrepancy sequences avoid clustering of samples. However, careful review of each sequence's limitation needs to be given as each has its strengths and weaknesses, especially in terms of the maximum number of dimensions allowed.

The raw result of an uncertainty quantification using any of the sampling strategies described above is a median with confidence interval centered about it. During the simulation setup, the user is asked for a confidence level larger than 0.5 and smaller than 1.0. Cumulative distribution function (CDF) uncertainty quantification involves a family of CDFs - one for each sample. The user can opt to have the raw data post-processed and output the median failure function with its confidence limits defined by the confidence level.

3.3. Importance Measures

Importance measures are key ingredients of PRA used to rank the relative contributions to risk between end states or components in a system. A wide range of time-dependent importance measures have been included in ADS-IDAC: conditional, marginal, improvement potential, criticality, diagnostic, risk achievement worth, and risk reduction worth.

The conditional importance measure of a component is based on the time-dependent conditional probability of system failure given that component has already failed:

$$\boxed{I^{conditional}(i|t) = P_S(F_S|F_i, t)}$$

The marginal or Birnbaum's importance measure quantifies the rate of change of the system reliability because of changes to the reliability of a single component. Its mathematical expression is:

$$\boxed{I^{marginal}(i|t) = \frac{\partial R_S(t)}{\partial R_i(t)}}$$

If the importance measure is large for a component i, then a small change in the reliability of component i results in a large change in the system reliability at time t. The marginal importance measure can be interpreted to be the probability that at time t component i is critical for the system.

Also note that the marginal importance measure of component i is independent of the actual reliability of component i. In other words, it only depends on the structure of the system and the reliabilities of the other components.

In practice, components with a very low value of the marginal importance measure have a small effect on the system reliability, thus requirements for finding highly reliable components to perform their functions may be relaxed. On the other hand, components with a very high value of the marginal importance measure are critical for the system at that time t; therefore, a lot more effort should be put into finding components with higher reliability, finding higher quality reliability data, or even changing the structure of the system.

The improvement potential importance measure is the difference between the system reliability with an ideal component i (that is, its reliability is equal to 1), and the system reliability with the actual component i. Mathematically, this is defined as:

$$\boxed{I^{potential}(i|t) = R(t|1_i) - R(t)}$$

The improvement potential importance measure, as its name is suggesting, indicates how much it is possible to improve the current system reliability by replacing component i with an ideal component.

The criticality importance measure is defined as the probability that component i is critical for the system and is failed at time t, given that the system is failed at time t. It can be obtained from the marginal importance measure in the following way:

$$\boxed{I^{criticality}(i|t) = \frac{I^{marginal}(i|t) \cdot p_i(t)}{p_s(t)}}$$

In other words, the criticality importance measure gives a measure of the probability that a component i causes the system to fail. Therefore, if the component i is repaired, the system is expected to function again. The prioritizing of maintenance or repair actions in complex systems can be accomplished with the criticality importance measure.

The diagnostic or Fussell-Vesely importance measure is the probability that at least one minimal cut set that contains component i is failed at time t, given that the system is failed at time t. It is expressed as:

$$\boxed{I^{diagnostic}(i|t) = \frac{\sum_{j=1}^{mcs_i} p_j^i(t)}{p_s(t)}}$$

In practice, it should give similar results as the criticality importance measure, thus they can be used for the same scope.

The risk achievement worth (RAW) importance measure quantifies the relative increase in the system failure given that component i is in a failed state:

$$\boxed{I^{RAW}(i|t) = \frac{F_s(t|1_i)}{F_s(t)}}$$

In practice, the RAW importance measure is used to find the risk significance of components that are removed from the system. If the importance measure is close to 1, then its improvement has negligible effect on the system.

The risk reduction worth (RRW) importance measure quantifies the relative reduction in the system failure given that component i is functioning:

$$\boxed{I^{RRW}(i|t) = \frac{F_s(t)}{F_s(t|0_i)}}$$

The basic event i may sometimes represent an operator action instead of a component failure. For such cases, it may be useful to analyze the effect of operator inaction on the mission success. It is similar to the critically importance measure.

4. UNCERTAINTY QUANTIFICATION OF A PWR STEAM GENERATOR TUBE RUPTURE EVENT

A test case capturing the system and crew behavior given a concurrent steam generator tube rupture (SGTR) and a main steam line break (MSLB) designed for the International HRA Empirical Study [12] was simulated to showcase the new models implemented into ADS-IDAC. The scope of the empirical study was to perform experiments at the Halden Reactor Project’s HAMMLAB (HAIden huMan-Machine LABoratory) research simulator where real crews were asked to respond to a series of carefully designed accident conditions to build an empirically based understanding of the performance, strengths, and weaknesses of the most used conventional HRA methods.

The HAMMLAB research simulator is a three-loop Westinghouse PWR. Also, The HAMMLAB's EOPs were loosely based on the emergency response guidelines (ERGs) developed by the Westinghouse Owners Group. The EOPs used in the complex SGTR scenario are: E-0 – "Reactor Trip or Safety Injection" and E-3 "Tube rupture in one or several steam generators". E-0 is the safety systems verification and diagnosis procedure used by the crew when the reactor has tripped, when safety injection has started, or when there is a need for either of them. E-3 is the procedure to which the crew are typically expected to transfer from E-0 when a diagnosis of SGTR is declared. E-3 contains the recovery instructions following a SGTR.

The complex SGTR scenario starts with a concomitant SGTR and a main steam line break (MSLB) in normal operation at 100% that immediately activates automatic SCRAM and the expectation that the crew would open the E-0 procedure for verification of safety systems and diagnosing the NPP condition. Two more complications were introduced to create a more complex scenario: the main steam isolation valves (MSIVs) close automatically in response to the MSLB, and the failure of any remaining secondary radiation indications. The scenario was designed so that as the MSLB drives the NPP response early in the scenario, where the initial symptoms of the NPP resemble a severe MSLB with the quick closure of the MSIVs, and all the secondary radiation indications fail. These conditions are expected to mask the occurrence of the SGTR and make its diagnosis a lot more challenging than the typical SGTR alone where the crews are expected to transfer to E-3 at step 19. This challenges the crew's procedure-following strategy, and a correct diagnosis of the NPP conditions would heavily rely on their knowledge-based reasoning or otherwise delay the transfer until they reach step 21.

The HFE selected to be quantified in ADS-IDAC is HFE 1B: failure of the crew to isolate the faulted steam generator. The sequence of interest for the selected test case is replicated only by crew M that diagnose the SGTR based on the steam generator level at step 21 by following the procedures. The top events of interest leading to HFE 1B are: successful reactor trip, feedwater available, and high-pressure injection started. Note that ADS-IDAC generates multiple sequences and covers most of the crew variability observed in the International HRA Empirical Study. Also, the algorithms and methodology behind the point-estimate quantification has already been published in [13].

For uncertainty quantification in ADS-IDAC, a probability distribution with its variance was needed for the HFEs given that the means are obtained in the point estimate stage of the dynamic simulation. In this respect, the results of the empirical study provide a good basis. A lognormal distribution has been selected for all the HFEs. From the empirical study estimation using various HRA methods, the variance in probabilities of all the HFEs is around 1.5E-3, therefore this value was adopted in this analysis as well.

In order to obtain the probability of success for HFE 1B (that is, the crew succeeds to isolate the faulted steam generator) all the decision-maker's and action-taker's conditional event probabilities from the time step when the high-pressure injection starts until the crew has isolated steam generator A have been multiplied to obtain: 87.82% with the 5th and 95th percentile bounds 82.69% and, respectively, 89.55%. The confidence bounds have been obtained by running a Monte Carlo simulation with 100,000 samples. Therefore, the probability of HFE 1B (failure of the crew to isolate the faulted steam generator) is 12.18% with the 5th and 95th percentile bounds 10.45% and, respectively, 17.31%. This falls in the band for HFE 1B, which is in agreement with the empirical study results.

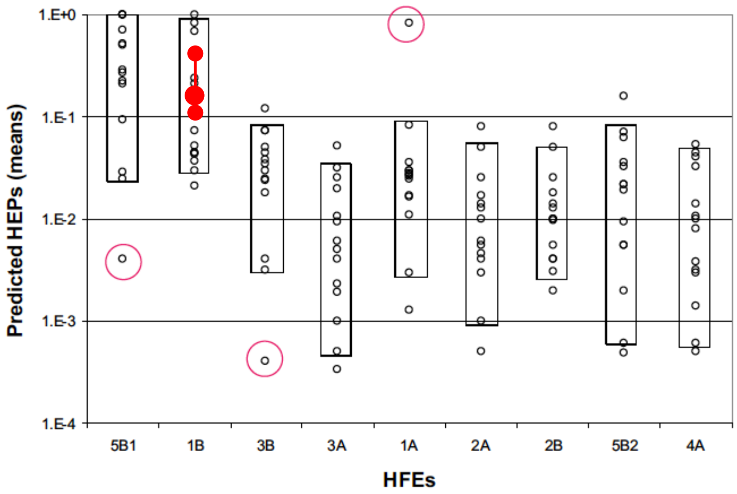

Figure 4 HFE 1B prediction of ADS-IDAC

5. CONCLUSION

In this paper the new quantification features of ADS-IDAC necessary to perform uncertainty quantification have been described. Also, an example of a fully quantified with uncertainties human failure event in a scenario with a concurrent steam generator tube rupture and a main steam line break is described and compared with available empirical data obtained from the International HRA Empirical Study. Good agreement between the simulation and empirical data is shown, nevertheless the qualitative predictive power of ADS-IDAC is as important as the quantification itself.

Acknowledgements

This work was funded through a Research Grant (NRC Grant HQ-60-14-G-0013) by the U.S. Nuclear Regulatory Commission (USNRC). Also, the authors acknowledge the contribution of Nilesh Gupta to the developmental work in HCL's expression parsing module.

References

[1] C. G. Acosta, N. O. Siu. Dynamic Event Tree Analysis Method (DETAM) for Accident Sequence Analysis. Massachusetts Institute of Technology, Nuclear Engineering Department, MITNE-295, http://hdl.handle.net/1721.1/89676, (1991).
[2] P. C. Cacciabue. Cognitive modeling: a fundamental issue for human reliability assessment methodology? Reliability Engineering & System Safety, vol. 38, pp. 91-97, (1992).
[3] K. S. Hsueh, A. Mosleh. The development and application of the accident dynamic simulator for dynamic probabilistic risk assessment of nuclear power plants, Reliability Engineering & System Safety, vol. 52, pp. 297-314, (1996).
[4] C. Smidts, S. H. Shen, and A. Mosleh. The IDA cognitive model for the analysis of nuclear power plant operator response under accident conditions. Part I: Problem solving and decision making model," Reliability Engineering & System Safety, vol. 55, pp. 51-71, (1997)
[5] Y.-H. Chang. Cognitive Modeling and Dynamic Probabilistic Simulation of Operating Crew Response to Complex System Accidents (ADS-IDA Crew). University of Maryland, (2007).
[6] K. A. Coyne. A Predictive Model of Nuclear Power Plant Crew Decision-Making And Performance In A Dynamic Simulation Environment. University of Maryland, (2009).
[7] Y. Li. Modeling and Simulation of Operator Knowledge-Based Behavior. University of Maryland, (2013).
[8] M. A. Diaconeasa, A. Mosleh. The ADS-IDAC Dynamic PSA Platform with Dynamically Linked System Fault Trees. International Topical Meeting on Probabilistic Safety Assessment and Analysis (PSA), Pittsburg, PA, (2017).

[9] E. Groen. http://evelynegroen.github.io/Code/Uncertaintypropagation.html Accessed on August 2017.
[10] M. Matsumoto, and T. Nishimura. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. ACM Trans. Model. Comput. Simul. TOMACS, vol. 8, no. 1, pp. 3–30, (1998).
[11] B. Beachkofski, and R. Grandhi. Improved Distributed Hypercube Sampling, in 43rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, American Institute of Aeronautics and Astronautics, (2002).
[12] E. Lois, V. N. Dang, J. Forester, H. Broberg, S. Massaiu, M. Hildebrandt, P.Ø. Braarud, G. W. Parry, J. Julius, R. Boring, I. Männistö, A. Bye. International HRA empirical study—description of overall approach and first pilot results from comparing HRA methods to simulator data (HWR-844/NUREG/IA-0216 vol 1). OECD Halden Reactor Project/US Nuclear Regulatory Commission, Halden, Norway/Washington, DC, (2009).
[13] M. A. Diaconeasa, A. Mosleh. Branching rules and quantification based on human behavior in the ADS-IDAC dynamic PRA platform. European Safety and Reliability Conference (ESREL), Trondheim, Norway, (2018).