Integration of Fault-Trees and Event-Trees into Dynamic PRA

D. Mandelli,* D. Maljovec, A. Alfonsi, C. Wang, Z. Ma, C. Parisi, C. Smith

Idaho National Laboratory (INL), 2525 Fremont Ave, 83402 Idaho Falls (ID), USA

*diego.mandelli@inl.gov

INTRODUCTION

The scope of this paper is to present a series of methods designed to incorporate classical Probabilistic Risk Assessment (PRA) [1] models such as Event-Trees (ETs) and Fault-Trees (FTs) into Dynamic PRA [2]. Dynamic PRA couples stochastic methods (i.e., stochastic sampling methods) with safety analysis codes to determine the risk associated to complex systems such as nuclear power plants. Compared to classical PRA methods, they can evaluate with higher resolution the safety impacts of timing and sequencing of events on the accident progression.

As part of a Dynamic PRA analysis, it is not uncommon that some components of the system under consideration might not require a complex simulation model (which would be computationally expensive) due to its intrinsic characteristics, but such components could be actually modeled by a classical PRA model (either ET or FT).

In this paper, we investigate how we can integrate classical PRA models (either ET or FT) into a dynamic PRA: an “hybrid” PRA. We show how this integration has been performed within the RAVEN statistical framework [3]. Lastly, the proposed PRA incorporation method can be employed to compare the analyses generated by Classical and Dynamic PRA methods and determine where the differences between the two methods lie.

CLASSICAL PRA

Classical PRA methods employ static Boolean structures to model system reliability. These structures are typically Event-Trees (ETs) and Fault-Trees (FTs) [1]. ETs inductively model the accident progression using a tree structure (i.e., with multiple branches) with the goal of obtaining all possible accident sequences. On the other hand, FTs deductively model system logical dependencies through a series of logical gates (e.g., AND, OR, NOT gates).

Typically, classical PRA methods employ both methods: ET to model accident progression (with ET branching dictated by system status) while FTs are used to determine branching probabilities. Note that in the ET the sequencing of events (and accompanying branchings) are already pre-fixed in the system logic designed by the analyst.

DYNAMIC PRA

Dynamic PRA methods [2] integrally employs both deterministic and stochastic methods in a single analysis framework. Compared to classical PRA, they heavily employ simulation tools in order to model: thermal-hydraulic behavior of the plant, external events such as flooding and operator responses to the accident scenario. Typically, this performed by [4]:

1. Associating a probabilistic distribution function (pdf) to the set of stochastic parameters $\boldsymbol{s}$ (e.g., timing of events)
2. Performing stochastic sampling of the pdfs defined in Step 1
3. Performing a simulation run given $\boldsymbol{s}$ sampled in Step 2
4. Repeating Steps 2 and 3 M times and evaluating user defined stochastic parameters such as CD probability (P_{CD}).

INTEGRATION OF CLASSICAL INTO DYNAMIC PRA

In a Dynamic PRA analysis, it is not unusual that sub-systems of the considered overall system do not require heavy computational tools. Instead, a static Boolean logic might be judged as “sufficient” to mimic the sub-system response. In such case, the ideal situation would be to link the available Classical PRA model of the sub-system to the Dynamic PRA model.

This integration has been accomplished in the RAVEN statistical framework [3] by introducing two additional model classes. These classes are designed to read from file the structure of the FT or the ET in the OpenPSA format [5]. The structure of the ET and the FT is described in a hierarchical fashion using `xml` markup language where each node characterizes either and ET branch or a FT gate.

The resulting object in RAVEN is a model characterized by a set of input and output variables that can be employed by RAVEN like any other model (e.g., codes like RELAP5-3D [6], surrogate models).

FT Integration

For a FT model, the set of input variables is the set of basic events while the output variable is the logical status of the top-event and, in general, any gate of the FT (see Fig.1). An example of FT in OpenPSA format is provided in Scheme 1. Such FT implements the following Boolean logic:

TopEvent = G1 or G2
= (BE1 and BE2) or (BE3 and BE4)

The resulting RAVEN FT model accepts in input a Boolean value (i.e., either 0 or 1, for example generated by a Binomial distribution) for each of the four input basic events (BE1, BE2, BE3 and BE4) and would generate a Boolean value for the output variable TOP.

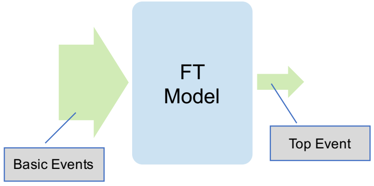

Fig. 1. Logical scheme for a RAVEN FT model

```
<define-fault-tree name="FT">
  <define-gate name="TOP">
    <or>
      <gate name="G1"/>
      <gate name="G2"/>
    </or>
  </define-gate>
    <define-gate name="G1">
      <and>
        <basic-event name="BE1"/>
        <basic-event name="BE2"/>
      </and>
    </define-gate>
    <define-gate name="G2">
      <and>
        <basic-event name="BE3"/>
        <basic-event name="BE4"/>
      </and>
    </define-gate>
    <define-basic-event name="BE1">
      <float value="1.2e-3"/>
    </define-basic-event>
      <define-basic-event name="BE2">
        <float value="2.4e-3"/>
      </define-basic-event>
      <define-basic-event name="BE3">
        <float value="5.2e-3"/>
      </define-basic-event>
      <define-basic-event name="BE4">
        <float value="3.1e-3"/>
      </define-basic-event>
<define-fault-tree>
```

Scheme 1. Example of FT in OpenPSA format

ET Integration

The integration of ETs follows path similar to the one presented for the FTs. For an ET model, the set of input variables is the set of branching conditions while the output variables are the sequence number and the predicted outcome of such sequence (see Fig. 2). As an example, the ET shown in Fig. 3 can be described in the OpenPSA format as shown in Scheme 2. The branching conditions are here dictated by the status of the systems ACC, LPI and LPR (i.e., functional events).

CLASSICAL AND DYNAMIC PRA COMPARISON

As an example of application of the methods presented in the previous section, we present a comparison of the accident sequences generated by SAPHIRE [7] in an ET form and the sequences generated by RAVEN/RELAP5-3D in the form of simulated transients. The objective is to compare Dynamic PRA data with the corresponding ET structure.

This has been performed by developing a data post-processor in RAVEN which associates each transient simulated by RAVEN/RELAP5-3D to a single specific branch of the corresponding ET by:

- identifying the ET branching conditions in the simulated transient (e.g., successful activation of the accumulator system)
- Determining the successful/unsuccessful outcome of each branching condition
- Identifying the ET branch that matches the set of branching condition outcomes; if no match is found then the ET requires a review (e.g., additional branches/branching conditions)

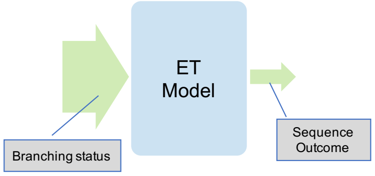

Fig. 2. Logical scheme for a RAVEN ET model

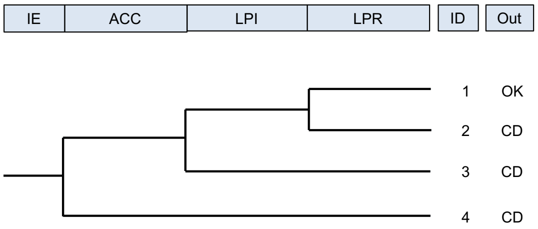

Fig. 3. Large LOCA ET model

```
<define-event-tree name="LOCAeventTree">
  <define-functional-event name="ACC"/>
  <define-functional-event name="LPI"/>
  <define-functional-event name="LPR"/>
  <define-sequence name="0"/>
  <define-sequence name="1"/>
  <define-sequence name="2"/>
  <define-sequence name="3"/>
  <initial-state>
    <fork functional-event="ACC">
      <path state="0">
        <fork functional-event="LPI">
          <path state="0">
            <fork functional-event="LPR">
              <path state="0">
                <sequence name="0"/>
              </path>
              <path state="+1">
                <sequence name="1"/>
              </path>
            </fork>
          </path>
          <path state="+1">
            <sequence name="2"/>
          </path>
        </fork>
      </path>
      <path state="+1">
        <sequence name="3"/>
      </path>
    </fork>
  </initial-state>
</define-event-tree>
```

Scheme 2. Example of ET in OpenPSA format

TEST CASE

We have developed this comparison between the Dynamic PRA approach (using RAVEN/RELAP5-3D) and the classical approach (using SAPHIRE) for an industry relevant test case: a PWR Large Break LOCA (LB-LOCA).

The system considered is a 3-loop PWR system which undergoes a double guillotine break of one of the three hot-legs. In this scenario, depressurization of the primary vessel occurs very quickly and large amount of water inventory is lost due to the break. In order to compensate loss of water inventory and provide cooling to the core in order to avoid core damage, several systems are employed (see Fig. 3):

- Accumulator system (ACC) which consists of water tanks that are employed right at the beginning of the transient in order to flood the RPV
- Low Pressure Injection (LPI) system which is an injection system that transfers cold water from the RWST tank to the RPV
- Low Pressure Recirculation (LPR) system which is employed once the RWST tank is empty; this system is composed of the same components as the LPI system but the water source is now the water collected inside the containment through the containment sump.

ANALYSIS SCHEME

An overview of the performed analysis is structured as follows:

1. Perform the calculation of the LB-LOCA ET-FT model using SAPHIRE: determine CD probability, and probability of each ET branch
2. Import the LB-LOCA ET into RAVEN
3. Import the FT models for the ACC, LPI and LPR systems in RAVEN and connect them to the RELAP5-3D model for LB-LOCA (see Fig. 4)
4. Perform a dynamic analysis using RAVEN/RELAP5-3D for the system constructed in Step 3
5. Classify the simulated transients obtained in Step 4 by matching them into the ET structure imported in Step 2
6. Compare the results obtained in Steps 1 and 5.

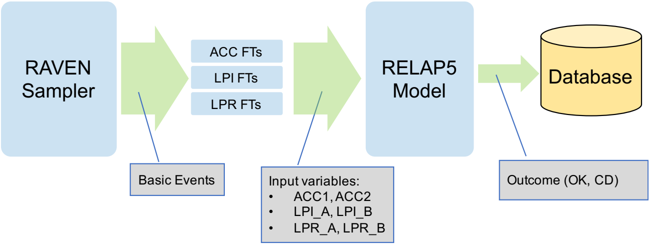

Fig. 4. LB-LOCA analysis performed in RAVEN by coupling FTs with RELAP5 model

RESULTS

At a first glance, the two analyses provided almost identical results in terms of core damage probability: 8.13 E-3 from SAPHIRE calculation and 8.24 E-3 from RAVEN/RELAP5-3D. However, the first differences between the two methods arise when we evaluate the probability associated to each branch of the ET as indicated in Fig. 5. The first three branches have characterized with almost identical probabilities values; on the other side, probability of branch 4 had a much lower probability value when RAVEN/RELAP5-3D was employed: 5.76 E- 10. The cause of this difference is the success criteria that in SAPHIRE requires 2 ACCs out of 2 while one ACC system (out of 2) is sufficient to avoid CD in at the beginning of the transient.

ID	Out	Branch Probability	
		SAPHIRE	RAVEN
1	OK	0.99187	0.99176
2	CD	7.27 E-3	7.365 E-3
3	CD	8.12 E-4	8.744 E-4
4	CD	4.80 E-5	5.76 E-10

Fig. 5. Comparison of branch probabilities calculated using SAPHIRE and RAVEN/RELAP5

CONCLUSIONS

In this paper we have briefly summarized a few methods that can be employed to incorporate classical PRA models into simulation based (i.e., dynamic) PRA analyses. The objective is to model parts of the system with components that do not require advanced simulation tools but instead can be modeled using static Boolean structure such as FTs or ETs.

REFERENCES

1. U. S. NRC, "NUREC/CR-1150: Severe Accident Risks: An Assessment For Five U.S. Nuclear Power Plants Final Summary Report", Technical Report, U.S. Nuclear Regulatory Commission, Washington DC (2005).
2. J. DEVOOGHT, C. SMIDTS, "Probabilistic Dynamics as a tool for Dynamic PSA," *Reliability Engineering & System Safety*, **52**, 185–196 (1996).
3. C. RABITI, A. ALFONSI, J. COGLIATI, D. MANDELLI, R. KINOSHITA, "RAVEN, A New Software For Dynamic Risk Analysis," *in Proceedings of the Probabilistic Safety Assessment and Management PSAM* 12 (2014).
4. D. MANDELLI, C. SMITH, T. RILEY, J. NIELSEN, A. ALFONSI, J. COGLIATI, C. RABITI, J. SCHROEDER, "BWR station blackout: A RISMC analysis using RAVEN and RELAP5-3D," *Nuclear Technology,* **193**, 161–174 (2016).
5. Open-PSA Model Exchange Format GITHUB repository: https://github.com/open-psa
6. RELAP5-3D Development Team, *RELAP5-3D Code Manual*, Technical Report, Idaho National Laboratory Technical Report (2005).
7. C. SMITH, J. KNUDSEN, D. O NEAL, T. PETERSON, K. VEDROS, *SAPHIRE 8 New Features and Capabilities*, Technical Report, Idaho National Laboratory Technical Report: INL/EXT-08-14841 (2008).