Bridging the gap between Static and Dynamic Probabilistic Assessment: a step forward

Claudia Picoco[a*], Valentin Rychkov[a]
[a] EDF R&D, Palaiseau, France

Abstract: Dynamic Probabilistic Risk Assessment (DPRA) is acknowledged to be an important tool complementary to static Boolean approach in those cases where physics or timing of events may play an important role into the system evolution and, thus, into the risk estimate.

Although, the methods and the tools classified as Dynamic PRA (DPRA) are becoming more and more known and common in the last few years, some key challenges remain and should be addressed to make DPRA industrially applicable.

The results presentation and post-processing is one of these challenges that may impact the potential connection with PRA world. A DPRA model returns thousands or even tens of thousands of dynamic sequences that are inaccessible to be analyzed one by one and require therefore some post-processing to obtain insights and quantify figures of merit. Furthermore, since Monte Carlo is often used for model simulation, sequences binning is necessary to obtain probabilistic quantification. PRA instead returns a rank of minimal cut sets (MCS) which are the minimal static combinations of basic failures leading to the undesired events.

The challenge in DPRA is twofold. On one hand, it is necessary to condensate the information coming from all the sequences in a form such that results are still accessible and dynamic aspects can still be appreciated. On the other hand, in order to be used in connection with traditional PRA, the post-processing should get to a result that is similar to a minimal cut sets.

It is a trade-off problem where a level of detail of post processed sequences too high (limit having all sequences separately) does not allow for getting insightful quantitative information while a level of detail too coarse (limit to the level of MCS) risk losing the “dynamic“ information coming with the sequences.

In this paper, we will present a post processing framework that allows DPRA resulted sequences to be analyzed and, at the same time, “compared” to a static PRA approach. This post-processing, although it may be considered “basic” in terms of approaches, has been developed for a real case study.

Keywords: PRA, Dynamic PSA, Post-Processing

1. INTRODUCTION

Dynamic methods allow for more realistic modeling of accidental scenarios. They have demonstrated to be useful in some particular cases such as those cases where the event timing and/or the physics plays a key role, reparations, passive systems, etc. [1][2][3]

Different dynamic methods and tools exist, however, in general a dynamic analysis is often characterized by thousands of simulated sequences, using Monte Carlo simulations. When having to deal with such a great number of simulated sequences, post-processing is necessary to extract the most of information and quantify the expected figures of merit. The analysis of the sequences one-by-one is impossible.

Although, several possible post processing approaches has been proposed in literature (see §2), the question remains of how to deal with results coming from a dynamic analysis order to make them accessible by Probabilistic Risk Assessment (PRA) practitioners that are used to Minimal Cut Sets (MCS) ?

In this paper, we propose a framework to post process results coming from a dynamic analysis that does not simulate physics but only time of event occurrence. In a context where the static PRA is the standard practice, the reason behind the use of dynamic methods for an industrial case was to challenge the static PRA modeling. On one hand, the goal is to find coherent results with PRA in terms of sequences/cutset to comfort us in the modeling choice. On the other hand, all differences in the results and all the insights could point us towards scenarios missing in our static model.

This post processing approach has been elaborated in collaboration with PRA practitioners of our engineering division for a real industrial application. The approach can be generalized for any dynamic methods simulating sequences. The challenge is twofold: on one hand, we want to highlight most of the dynamic information as this is the added value with respect to a traditional PRA approach, on the other hand the goal of getting closer to PRA form of results (e.g., minimal cut sets) in order to make the results accessible to PRA analysts, goes in the direction of masking the dynamic information.

Since the approach has been applied to an industrial case [1], all the results presented on this paper are fictious for purpose of examples.

2. LITERATURE REVIEW

As we mentioned in the introduction, several dynamic tools exist such EMRALD [4], RAVEN [5], ADAPT [6], GRIF [7], DICE [8], BDMP [9], etc. adopting different dynamic approaches. These tools have been developed for reliability and PRA applications, therefore, they mostly provide results in some post-processed form (e.g., BDMP provides the most probable sequences [10]) or, in some cases, they provide post-processing capabilities (e.g., clustering) for the user to apply (e.g., RAVEN).

Some works can be found in literature proposing some post-processing techniques that can be useful to interpret results and simulations coming from dynamic analysis. Zio et al, for example, propose clustering technique to identify Safe, Near Missed and Failed simulations [11][12].

In this work, we faced two challenges for post-processing:

1. Our dynamic analysis does not consider physics, our sequences therefore consist of discrete sequences of timestamp and event. In other words, we did not have continuous variables temporal series simulated.
2. The modeling approach we used [1] is not specific to reliability and PRA applications. This gave us the opportunity to construct post-processing from scratch.

3. POST-PROCESSING FRAMEWORK

The starting point is the results of the dynamic analysis, namely thousand (even hundreds of thousands) of generated sequences. In our case they are of type (timestamp, event):

Sequence 1 … Sequence M
T_1, Event at T_1 … T_1, Event at T^*_1
T_2, Event at T_2 … T_2, Event at T^*_2
… …
T_N, Event at T_N T_P, Event at T^*_P

Sequences terminate because of mission time or because they met a given undesired condition (e.g., core damage). They contain all type of events (failures, human actions, systems starting, alarms, etc.).
Here below a sample of sequence is shown as an example (time indications and events are not real):

```
…
03:15:00, Appearance of a break: LOCA accident
04:25:00, I&C signal to automatically start HPI failed
04:30:00, Human action to start HPI
04:30:40, Start of HPI
07:15:32, Failure in operation of HPI
…
09:23:41, Core Damage
```

From this point on, we adopted a step-by-step approach.

3.1. Estimation of the figure of merits

This first step is the easiest one as it does not really required manipulation of the generated sequences. According to the Monte Carlo simulation, the probability of a given condition is equal to the number of sequences ending with that condition divided by the total number of simulated sequences.
Let’s take the example of the probability of core damage, P_{CD}:

$$P_{CD} = \frac{Number\ of\ sequences\ ending\ with\ core\ damage}{Number\ of\ simulated\ sequences} = \frac{N_{CD}}{N} \tag{1}$$

The given level of precision that depends on the number of simulation N and on the estimator $\hat{p}$ of P_{CD} can be estimated using the formula below for a confidence level of 95% [13]:

$$\hat{p} \pm \frac{1{,}96\sqrt{\hat{p}(1-\hat{p})}}{\sqrt{N}} \tag{2}$$

This precision can be improved by increasing the number of simulations.

3.2. First event filter on sequences

The sequences issued from the simulation contain all the events occurring. We decided therefore to apply a first filter that removes all those events that are considered redundant, useless, etc. This way we are left with sequences that “tell” us the history of the simulated events: I&C, human actions, failures, alarm trigger, etc.

The representation of sequences in this form is necessary within the verification process: a sample of sequences are analyzed in detail to make sure that the way the system evolves and responds to events is coherent with reality.

As we construct our model using a hierarchical approach like statechart [1], this step is necessary in our case to “clean up” the sequence and improve readability. We would indeed have condition like:

```
00:00:00, state AFW entered

00:00:00, state AFW_ON entered
```

That could be simplified directly as:

```
00:00:00, AFW ON
```

3.3. Second event filter on sequences

As the MCS contain only failure event, the second filter we apply consist of keeping only unavailability events removing all the other events such human action, triggers, successful I&C, that do not imply any failure/maintenance events. This way we get to a form of sequence that are similar to MCS.

If we consider the first example of §3.1, the extract will simply become:

```
…
03:15:00, Appearance of a break: LOCA accident
04:25:00, I&C signal to automatically start HPI failed
07:15:32, Failure in operation of HPI
09:23:41, Core Damage
```

3.4. Binning sequences keeping the order of occurrence of events

Up to the previous steps, we still analyze via a parsing script one sequence at a time. Since we deal with Monte Carlo simulations, as long as we consider one sequence at a time, each sequence has a likelihood of 1/N to occur. Therefore, sequences binning is necessary to get numerical insights.

The question of how to group sequences is to be posed. At this step, we have plenty of sequences containing failure events only, however, since we deal with a dynamic analysis, two elements are worth noticing:

- Order of events appearance in these sequences is important.
- Each event has a timing associated.

We will perform the first binning by keeping the dynamic information concerning the order of event appearance. Moreover, since timing is still an available information, it is possible to calculate the minimum, maximum and mean time of occurrence of these events.

Results will appear in the form of sequences like:

```
[minT₁, meanT₁, maxT₁], Event₁
…
[minTN, meanTN, maxTN], EventN
```

For the extract example:

```
[03:15:00,04:23:32,06:58:00],Appearance of a break: LOCA accident
[03:24:56,05:56:21,07:02:14],I&C signal to automatically start HPI failed
[04:28:51,07:43:12,09:47:09],Failure in operation of HPI
[06:04:25,09:53:02,11:45:56],Core Damage
```

An html file with hyperlinks keep track of all the simulation included in the bin and allows to go back and analyze them one by one in the readable form as in §3.2. These groups sequences have a probability associated and can be ranked in the decreasing order of probability.

3.5. Binning sequences without keeping the order of occurrence of events

According to the grouping at the §3.4, the two following sequences are considered as different since the order of appearance of the two events EDG_A failure and EDG_B failure is different:

LOOP, EDG_A failed in operation, EDG_B failed in operation, CORE DAMAGE

LOOP, EDG_B failed in operation, EDG_A failed in operation, CORE DAMAGE

However, if the goal is to compare the results with some static MCS then these two should be summed up together. So, the next binning step consists in grouping together sequences containing same failure events independently from their order of appearance. When we sum up the probability of these sequences, then the explicit information concerning the dynamic aspects is lost but we get to something that can be in a comparable form to static MCS.

According to our experience, at this point we would find sequences that looks like not minimal one with respect to the other. Could we reduce them as in Boolean logic ? The answer is no. The reader should remember that into a dynamic analysis, the dynamic information is still there even if it does not appear explicitly.

Let’s consider the following example:

SEQUENCE 1: LOOP, EDG_A, EDG_B, CORE DAMAGE

SEQUENCE 2: LOOP, EDG_A, EDG_B, FAILURE OF THE TEST PUMP[1], CORE DAMAGE

In a static model, Sequence 2 would not appear in the results as non-minimal. However, this is not the case in a dynamic analysis.

In Sequence 1, the timing of appearance of failures of the two EDGs is such that when the test pump needs to be activated to prevent Seal LOCA, batteries are already drained. This results in the unavailability of the I&C to start the test pump, thus of the pump itself, and therefore in a Seal LOCA, leading to the core damage.

In Sequence 2, the timing of failure of the two EDGs are such that, by the time the test pump needs to be started, battery (and thus) I&C are still available. The I&C command is correctly launched but the test pump failed on demand leading to the Seal LOCA and therefore to the core damage.

In this second sequence, if the pump would have worked correctly then no core damage would have occurred. This explains why the two sequences are not minimal one with respect to the other.

3.6. Contribution of loss of safety function

The statechart approach gives us a great flexibility in terms of modeling, since it allows to track all the states entered within one sequence. To provide a different perspective and insights, we decided to associate the core damage to the loss of one among the modeled safety functions.

[1] Test pump ensures the seal injection in Station Black Out sequences.

As presented in [1], the final model consists of several components, safety and support systems. Furthermore, the safety functions fulfilled by the different safety systems and the core damage state are also modeled. This means that we are able to track the path towards the core damage. This tracking allows us to associate each core damage to the failure of one of three safety functions.

In order to quantify the contribution of the loss of each safety function, some assumptions were made. For a given sequence:

1. The loss of a single safety function is considered to automatically lead to the core damage
2. The earliest safety function lost will be “attributed” to the considered core damage sequence and will terminate the sequence.

This second assumption made possible to quantify how much the loss of each safety function contributes to the overall core damage frequency as :

$$contribution\ of\ safety\ function_{\ i} = \frac{N_{sequences\ ending\ with\ core\ damage\ due\ to\ the\ loss\ of\ safety\ function\ i}}{N_{sequences\ ending\ with\ core\ damage}}$$

At the end of the simulation, all core damage sequences have been “attributed” to the failure a given safety function. These contributions for the different safety functions can be graphically represented by a pie chart as the one shown in Figure 1.

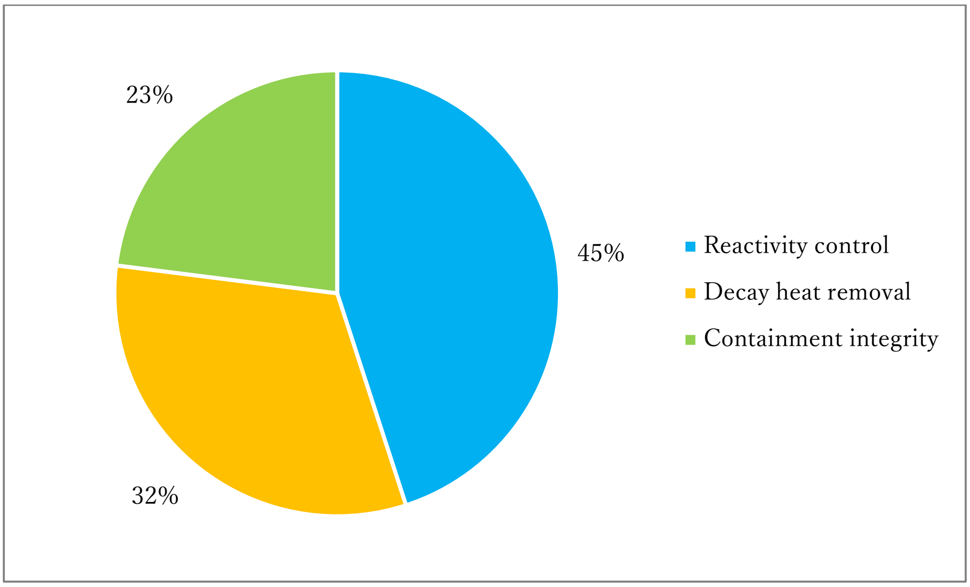

Figure 1. Contribution of loss of each safety function to the global risk

This representation is closer to a certain extent to the modelling that we find in event trees.

3.7. Sensitivities

In a dynamic analysis, all the variables (physics delay, reliability data, thresholds, operator numbers, etc.) are parameters that can be changed. This implies that sensitivity analysis of the dynamic model behavior can be easily performed. For example, we can perform a sensitivity analysis on the number of fieldworkers available for local actions and defines what is the optimal number with respect to the global risk for the given scenario.

The post-processing would have to be rerun but, for example, insights could come by the observation of how the failure of the safety functions changes based on some given parameters’ values.

3.8. Summary

Figure 2 summarizes the post-processing framework. In yellow are shown the steps where the sequences are analyzed one-by-one while in green are shown the steps where sequences that met a given criteria are binned together.

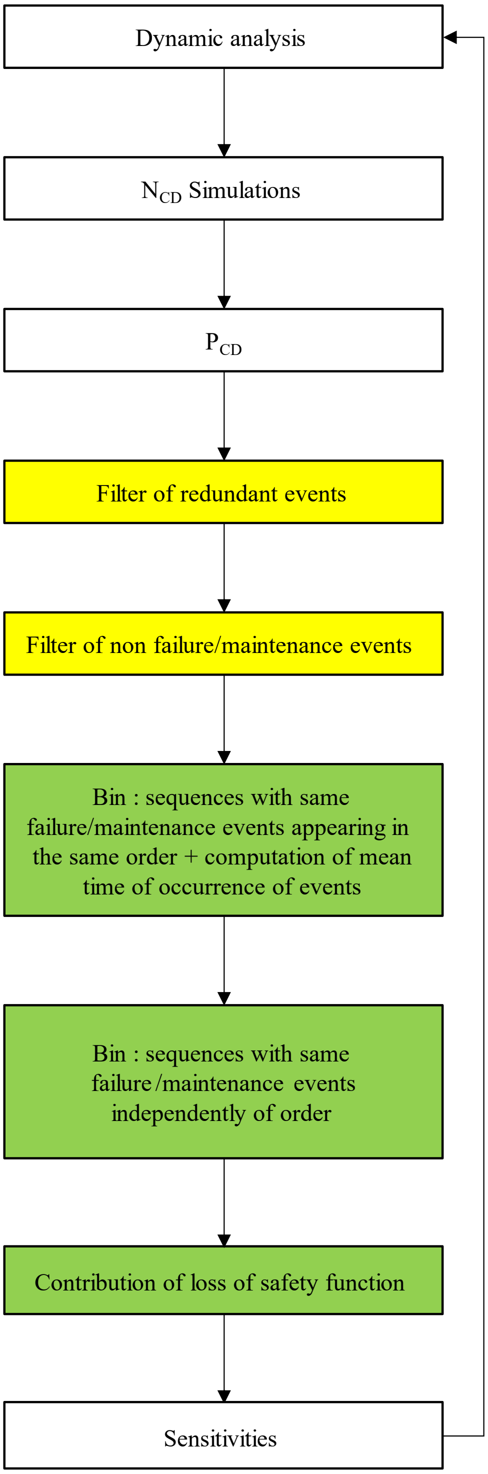
Dynamic analysis
N_{CD} Simulations
P_{CD}
Filter of redundant events
Filter of non failure/maintenance events
Bin : sequences with same failure/maintenance events appearing in the same order + computation of mean time of occurrence of events
Bin : sequences with same failure /maintenance events independently of order
Contribution of loss of safety function
Sensitivities

Figure 2. Summary of post-processing framework

4. CONCLUSION

In this paper we present a framework to post-process results from a dynamic analysis. The post-process has been applied to a real-size industrial case and has been elaborated in collaboration with PRA practitioners from the engineering division. The post processing of sequences generated by a dynamic analysis is a necessary step to get qualitative (e.g., not previously identified scenarios) and quantitative insights and to make accessible to PRA analysts.

In the next step we will investigate possible methods to associate and connect the outcomes of dynamic analysis with the PRA model.

ACKNOWLEDGEMENT

The authors would like to thank Romain Roy et Caroline Massoulier from EDVANCE engineering division for the discussions and the contributions to the development of the post-processing here presented.

REFERENCES

[1] Massoulier C., Roy R., Rychkov V., Picoco C., Statecharts as a Dynamic Method for Risk Assessment. (2023), ANS PSA 2023, Knoxville (US) 10.13182/PSA23-41115

[2] Mi Y., Tokuhiro A., Dynamic PRA based on system codes coupling for passive safety system in integral pressurized water reactor (2020) International Conference on Nuclear Engineering, Proceedings, ICONE, 3

[3] Picoco C., Rychkov V., Aldemir T., Integration of recoveries into dynamic event trees: A case study (2019) PSA 2019 - International Topical Meeting on Probabilistic Safety Assessment and Analysis, pp. 494 - 503

[4] https://emrald.inl.gov/SitePages/Overview.aspx Accessed 05.2024

[5] https://raven.inl.gov/SitePages/Overview.aspx Accessed 05.2024

[6] https://www.sandia.gov/app/uploads/sites/85/2021/06/document.pdf Accessed 05.2024

[7] https://grif.totalenergies.com/fr Accessed 05.2024

[8] Lee S. W., Baek, S. J., Heo G., Young K., Wan T., and Kim J. H. (2018). "Development of DICE (Dynamic Integrated Consequence Evaluation) for Procedure Coverability Studies: Conceptual Design Phase," in Proceedings of the KNS 2018 Fall Meeting (Korea: Republic of: KNS)

[9] Bouissou M., Bon J.-L., A new formalism that combines advantages of fault-trees and Markov models: Boolean logic driven Markov processes, Reliability Engineering & System Safety, Volume 82, Issue 2, 2003, Pages 149-163, ISSN 0951-8320

[10] Bouissou M., A Benchmark on Reliability of Complex Discrete Systems: Emergency Power Supply of a Nuclear Power Plant (2017). Electronic Proceedings in Theoretical Computer Science. 244. 200-216. 10.4204/EPTCS.244.8

[11] Di Maio F., Rossetti R., Zio E., Postprocessing of Accidental Scenarios by Semi-Supervised Self-Organizing Maps (2017) Science and Technology of Nuclear Installations, 2017, art. no. 2709109

[12] Di Maio F., Vagnoli M., Zio E., Risk-based clustering for near misses identification in integrated deterministic and probabilistic safety analysis (2015) Science and Technology of Nuclear Installations, 2015, art. no. 693891

[13] https://www.math.arizona.edu/~tgk/mc/book_chap2.pdf Accessed 05.2024