A new interfacing approach between level 1 and level 2 PSA

Nicolas Duflot[a], Nadia Rahni[a], Thomas Durin[a], Yves Guigueno[a] and Emmanuel Raimond[a]
[a] IRSN, Fontenay aux Roses, France

Abstract: IRSN (TSO of the French Nuclear Safety Authority) has been developing L2 PSAs for many years, using its own probabilistic tool, KANT (probabilistic event trees software) associated to a very fast-running source term code (MER). Since the IRSN L1PSAs event trees are developed with one other dedicated software, the L1-L2 PSA interface methodology is a key and difficult point of the IRSN PSA methodology.

In the previous versions of the IRSN PSAs, L1-L2 PSA interface was a mostly manual process, resulting in significant resources allocation. To cope with such a difficulty, a new interfacing approach, allowing computerized generation of plant damage states (PDSs), has been developed. This approach is based on the introduction of flag events (basic events with a probability of one) into the L1PSA minimal cut sets (MCSs) in order to transfer information related to front lines systems (needed for accident management) status and operators actions. Afterwards, the MCSs are filtered to identify automatically the different PDSs of the L1-L2 PSA interface using a new dedicated tool.

The automatic PDS generation allows implementing a very detailed L1-L2PSA interface easy to update. Since this new IRSN interfacing approach is based on fault trees only, it can be implemented with most of the level 1 PSA tools.

Keywords: Level 2 PSA, Interface, Plant Damage States (PDS), bridge trees.

1. INTRODUCTION

The purpose of this paper is to present the new interfacing approach used at IRSN (the French TSO) to interface the level 1 PSA (L1PSA), which assesses the probability of core damage, with the level 2 PSA (L2PSA), which aims to assess the risk of radiological release of a Nuclear Power Plant. This paper presents the motivation for the development of a new automatized interfacing method, the principles which sustain this method illustrated by examples and some results.

1.1. Context

As presented in the reference [1], in a L1PSA, *"the design and operation of the plant are analysed in order to identify the sequences of events that can lead to core damage and the core damage frequency is estimated. Level 1 PSA provides insights into the strengths and weaknesses of the safety related systems and procedures in place or envisaged as preventing core damage.*

In Level 2 PSA, the chronological progression of core damage sequences identified in Level 1 PSA are evaluated, including a quantitative assessment of phenomena arising from severe damage to reactor fuel. Level 2 PSA identifies ways in which associated releases of radioactive material from fuel can result in releases to the environment. It also estimates the frequency, magnitude, and other relevant characteristics of the release of radioactive material to the environment. This analysis provides additional insights into the relative importance of accident prevention, mitigation measures, and the physical barriers to the release of radioactive material to the environment (e.g. a containment building)".

Since the L2PSA is the prolongation of the L1PSA's sequences after the beginning of core degradation, an interface between the L1PSA and the L2PSA is required to transfer, from L1PSA to L2PSA, all information needed for L2PSA. As presented in [2], the interface is defined by plant damage states (PDS). The PDSs are the equivalent of initiating events in the level 1 event trees, i.e. they are the initiating events in the level 2 event trees. Accident sequences from L1PSA are grouped together into PDSs in such a manner that all accidents within a given PDS can be treated in the same way. Each PDS represents a group of level 1 accident sequences that have similar characteristics for severe accident scenarios, e.g. accident timelines, potential for generation of loads on the containment or systems availability, thereby resulting in similar severe accident progression and radiological source terms.

The plant damage states have to characterize the parameters that are needed to describe the sequences in the L2PSA analysis and those that influence the accident progression and source term. The PDSs definition is based on the definition of interface variables which can have different values. Each PDS corresponds to a specific combination of values of the different interface variables (an example of interface variables and their values is given in section 2.1).

As presented in [2], there are two approaches in developing the probabilistic model of a L2PSA:

- the integrated models,
- the separated models.

With the integrated model approach, the same computer tool is applied for L1 and L2PSA and the model database contains all level 1 and level 2 information. The advantage of such an approach consists in the possibility to use, in the level 2 event trees, the same fault trees as the ones used in the level 1. Consequently, the interface between level 1 and level 2 event trees is simplified since it is not required to codify in the PDSs the status of each system at the core damage time to make available this information in the level 2. On the other hand, the L2PSA has to be developed with the same event tree formalism as the L1PSA (i.e. event tree and fault trees, with Boolean fusion algorithms), which has limitation regarding L2PSA needs (especially for the modeling of severe accident phenomenology, dependencies between events and uncertainties). In addition, the quantification of such integrated models, if detailed, may be very time consuming since L1PSA and L2PSA have to be quantified simultaneously.

With the separated models approach, two different probabilistic event tree softwares can be used: one appropriate to the L1PSA and another one appropriate to L2PSA. It is thus required to implement a very detailed interface since the only link between L1PSA and L2PSA is the PDSs list. The list of interface variables has to be developed to inform about:

- the initiating events that start the accidental sequences leading to core damage,
- the status of the front lines systems[a] (e.g. fully available / partly available / not operable),
- the status of the key operator actions (initiated or not) like the initiation of the Reactor Coolant System (RCS) feed and bleed on LWR.

Furthermore, if some systems are modeled in L1PSA and some other in L2PSA, it may be complicated to guaranty the effective correlation between these systems failure (due to support systems, shared components…).

This disadvantage, intrinsic to the separated models, is counterbalanced by the possibility to use a L2PSA dedicated tool specifically designed to take care of the modeling of the level 2 part (accident progression after core damage). For example, IRSN, the French safety authority's TSO, has developed its own L2PSA tool package (see ref. [3]) which consists in KANT (probabilistic event trees software), MER (for source term calculations) and MERCOR (for standardized radiological consequences assessment). Since these codes are specifically designed for L2PSA, KANT allows, for example, the development of simplified (fast running) physical models to simulate each phenomenon during accident progression and allows also the transmission of physical values (time, pressures, temperatures…), allowing a precise description of the NPP status, through the Accident Progression Event Tree (APET). In the same manner, MER allows detail releases assessment for each severe accident sequence generated by the APET quantification.

At IRSN, the L1PSA is developed with a commercial tool: RiskSpectrum. The L2PSA is developed with the tool package KANT / MER / MERCOR. These codes, which deal with probabilistic

---

[a] A front line system is a system whose operating influences directly the accident progress. For example, the Safety Injection System (SIS) is a front line system whereas the Component Cooling Water System (CCWS) - even if it is a key system regarding safety - is not a front line system, but a support system.

quantification of severe accident phenomena, systems and human failures, allow fast-running consequences/frequencies severe accident calculations for a very large number of accidental sequences (thousands of release categories). The severe accident phenomena analyses are mostly performed with the reference code ASTEC[a] (Accident Source Term Evaluation Code). Each IRSN L2PSA (one for each French type of NPP) is supported by large set of ASTEC accident scenario calculations (between 100 and 200) to consider in detail and in a "best estimate" manner (few conservative assumptions) the different possible accidental sequences on a reactor

1.2. IRSN motivation for a new interface development

Thanks to recent developments (see ref. [4]), the IRSN tools package allows consequences/frequencies calculations for a large amount of severe accidental sequences in a detailed and best estimate (plus uncertainties) manner. To take advantage of these possibilities, a detailed and precise interface has to be developed.

The generation of such an interface has historically been made by adding bridge trees in the L1PSA model. These bridge trees are event trees connected to L1PSA core damage sequences and are used to specify the state of systems that are not considered in L1PSA sequences (for additional information regarding bridge trees construction, see section 2.2.4 from ref. [2]).

Even if this approach allows precise definition of PDS directly from an "extended" L1PSA, it requires a manual development of a large amount of sequences and the attribution of the PDSs to these sequences is also manual (at least in RiskSpectrum, the L1PSA software tool used by IRSN). In addition, the creation or the deletion of an interface variable involves a manual modification of all these sequences and consequences.

Thus, the objective of the new IRSN interface method development was:

- to offer the same level of precision as the bridge trees approach,
- to avoid deep modification of L1PSA event trees,
- to avoid any manual post processing of the L1PSA results,
- to guaranty a reasonable computation time when updating the PDSs frequencies.

2. IMPLEMENTATION OF A NEW INTERFACE APPROACH

2.1. Preamble

The construction of the interface between L1PSA and L2PSA requires, first, that all the information from L1PSA that should be used in the L2PSA, is identified. Then, this information has to be attributed to the different interface variables, which have to be precisely defined for this purpose.

For example, on a PWR, the "*manual start of the high pressure safety injection*" is identified as one key operator action that has to be considered in the L2PSA. In parallel, the *High Pressure Safety Injection System* (HPSI) is identified as a front line system whose availability has to be known in the interface. It is, then, possible to combine both pieces of information in a variable as shown in Table 1.

Table 1: Example of interface variable attributes (HPSI)

Values for HP variable	Description
1	The HPSI is available and started by the operators
2	The HPSI is started by the operators and available until the switch in recirculation mode
3	The HPSI is available but the operators have failed to start it (human error)
4	The HPSI is available until the switch in recirculation mode but the operators have failed to start it (human error)
5	The HPSI is not available

---

[a] ASTEC is an integral code which is able to simulate the plant behaviour from the initiating event to the possible release of radioactive products (the 'source term') outside the containment. It covers all the severe accident phenomena except steam explosion and containment mechanical integrity.

As an example, the list of interface variable considered in the on-going update of IRSN's 900MWe PWR L2PSA is given in the Table 2.

Table 2: Example of interface variable list (900 MWe PWR)

Variable name	**Description**
AC	Status of the safety injection accumulators
AE	Status of the Emergency Steam Generators Feed Water tank makeup water means
AG	Status of gravitational reactor pool makeup
AP	Status of the ultimate makeup from the neighbor plant
AP-RC	Status of the automatic makeup when operating at mid-loop level
AR	Status of normal steam generator feedwater system
AS	Status of the containment spray system
AT	Status of the diversified RCS pumps seals water injection pump
BP	Status of the low pressure safety injection system (LPSI)
CR	Status of the reactor criticality after initiating event occurrence (i.e. ATWS)
DL	Presence of a coolant boron dilution
EF	Status of the ultimate containment venting system
EG	Possible presence of a boron plugging in the core (i.e. failure of simultaneous injection in hot leg and cold leg if required)
ET	Initial plant state (at the time of the initiating event)
GM	Status of the reactor coolant pumps (operating or not)
GV	Status of the secondary cooling (merging the Emergency Feed Water System (EFWS) and the steam discharge (in atmosphere) system)
HP	Status of the High Pressure Safety Injection System (HPSI)
IE	Status of the containment isolation system
IG	Capability to perform Steam Generators isolation
LC	Status of the relaying 48V DC power
LH	Staus of the 6,6 kV AC emergency supplied distribution system
PL	RCS break location (if any)
PT	RCS break size (if any)
RA	Status of the residual heat removal system (CHRS)
RC	Status of the chemical and volume control system
RT	Type of Steam Generator Tube Rupture (if any)
SE	Type of steam line break (if any)
SO	Status of the pressurizer safety valves
SR	Status of the residual heat removal system's safety valves
VL	Type of interfacing LOCA (V-LOCA), if any
IC	Availability of the main control room
KT	Status of the automatic information transmission system from the main control room (MCR) to the national emergency teams
MP	Status of the containment pressure measurement (for severe accident)
MG	Status of the steam generators water level measurement
DD	Status of the in containment dose measurement
TR	Status of the core outlet temperature measurement

2.2. Principle of the method

The three steps of this new interface method consist in:

- firstly, expressing the different values of each interface variable thanks to binary questions (i.e. with answer yes / no). Each value of a given interface variable can be expressed as a combination of the answers to these questions,
- secondly, adding, in the Minimal Cut Set (MCS) from the L1PSA, flag events (i.e. basic events with a probability of one) which indicate the answer to these binary questions,
- finally, identifying automatically the PDS corresponding to each MCS based on a flag events filtering thanks to a dedicated tool named OIPK.

In the following paragraphs, these three steps are presented in more details and illustrated by an example issued from table 1.

First step:
This step consists in expressing the different values of a given interface variable thanks to binary questions. These questions are formulated in such a manner that the answer *yes* corresponds to the normal situation when systems are operational (and *no* corresponds to their unavailability). If the example of table 1 is used, the *HP* variable can be expressed based on the following questions:

a) Have the operators started manually the HPSI? (yes / no)
b) Is the HPSI available for direct injection mode (before switching in recirculation mode) ? (yes / no)
c) Is the HPSI available after the switch in recirculation mode? (yes / no)

As a result, the value 2 (*The HPSI is started by the operators and available until the switch in recirculation mode*) of variable *HP* corresponds to the following combination of binary questions:

question a = *yes* * question b = *yes* * question c = *no*

This step has to be done for all interface variables to be able to express all their values in term of combination of binary questions.

Second step:
This step consists in adding, if needed, flag events in L1PSA MCSs. A flag event is a basic event with a probability equal to one (i.e. when a flag event is added in a MCS, the MCS frequency is not modified). There is one flag event for each binary questions from step one. For a given binary question, there are two possibilities when a L1PSA is considered:

- either the MCS involves an unique answer to the binary question,
- or the MCS does not allow to identify an unambiguous answer to the binary question.

If there is a unique answer and if the answer to the binary question is *yes*, the flag event is added in the MCS. It is not added if the answer is *no*. No additional MCS is created.
If the answer to the binary question is unknown, both cases are considered. In a first case, the answer is supposed to be *yes* and the flag event is added in the MCS. In a second time, the answer *no* is supposed, the flag event is not added and one or several MCSs are created to consider the initial MCS plus the adverse event corresponding to the answer *no* (in our example, the MCS is combined with the failure of HPSI in injection mode).

For example, let us consider that the MCS from the L1PSA corresponds to a large break LOCA with a common cause failure of all emergency busbars:

*LB-LOCA * CCF_busbars*

It is then obvious that the answer to the question *Is the HPSI available for direct injection?* is *no* since the HPSI pumps are not powered. Consequently, no flag event is added, the MCS is not modified.
Let us consider, now, a MCS which corresponds to a large break LOCA with a total failure of the containment spray system:

*LB-LOCA * CCF_Cont.spray*

It is not possible, based on this MCS, to know if the HPSI is available for direct injection or not. Consequently, both cases are considered:

$$\text{initial MCS plus HPSI success} \rightarrow LB - LOCA * CCF\_Cont.spray * \underbrace{\S\ HP\_INJ\_OK}_{\text{Flag event which indicates that direct injection is available}}$$

$$\text{initial MCS plus HPSI failure} \rightarrow LB - LOCA * CCF\_Cont.spray * \underbrace{(\text{Failure of HPSI in direct injection})}_{\text{All the MCS corresponding to HPSI failure in injection mode}}$$

As a result,

- the initial MCS is preserved with the additional information direct injection available. Its frequency is unchanged (flag event has a probability of one) and
- a sub set of MCSs is created to consider the occurrence of the initial MCS and the failure of the direct injection. The frequency of this sub set of modified MCS is consistent with the frequency of the initial MCS considered in conjunction with the HPSI direct injection failure.

This second step **has to be performed for all the binary questions of all the interface variables**. As a result each initial MCS, corresponding to core damage, may contain many flag events and/or may have been modified to consider additional failures not considered in L1PSA. The new MCSs set obtained does not correspond anymore to the core damage but to the core damage and to the success or failure of all systems relevant for the PDSs construction. This new MCSs set is named extended MCSs set. Each MCS corresponding to core damage is extended to consider relevant information for PDSs construction.

The effective implementation of this second step in a fault trees / event trees context is presented in section 2.3.

Third step:

This third step consists in identifying, for each extended MCS, the corresponding PDS. This step is supported by a dedicated tool developed by IRSN and named OIPK. This tool has, as an input, a file containing all the extended MCSs. Before identifying automatically the existing PDSs and defining their frequencies, the user has to define, in OIPK, the interface variables and their values. Then, each value of each interface variable has to be expressed as a combination of flag events. The combination can used AND, OR, NOT and brackets.

Let us consider that

- the flag event which indicates that the HPSI direct injection is available is §HP_INJ_OK,
- the flag event which indicates that the HPSI recirculation is available is §HP_RECR_OK,
- all the basic events name, corresponding to the operator failure to start HPSI, start with H_SIS

It is then possible to define, in OIPK, the interface variable *HP* which contains five values. Based on the flag events name given above, the different values of HP are expressed as:

Table 1: Example of filters defined in OIPK for the variable HP (HPSI)

Num.	**Values of HP variable**	**Filter**
1	The HPSI is available and started by the operators	§HP_INJ_OK **and** §HP_RECR_OK **and not** H_SIS*****
2	The HPSI is started by the operators and available until the switch in recirculation mode	§HP_INJ_OK **and not** §HP_RECR_OK **and not** H_SIS*****
3	The HPSI is available but the operators have failed to start it (human error)	§HP_INJ_OK **and** §HP_RECR_OK **and** H_SIS*****
4	The HPSI is available until the switch in recirculation mode but the operators have failed to start it (human error)	§HP_INJ_OK **and not** §HP_RECR_OK **and** H_SIS*****
5	The HPSI is not available	**not** §HP_INJ_OK

Once the interface variables and their values are defined and expressed in terms of combinations of flag events, OIPK is able to

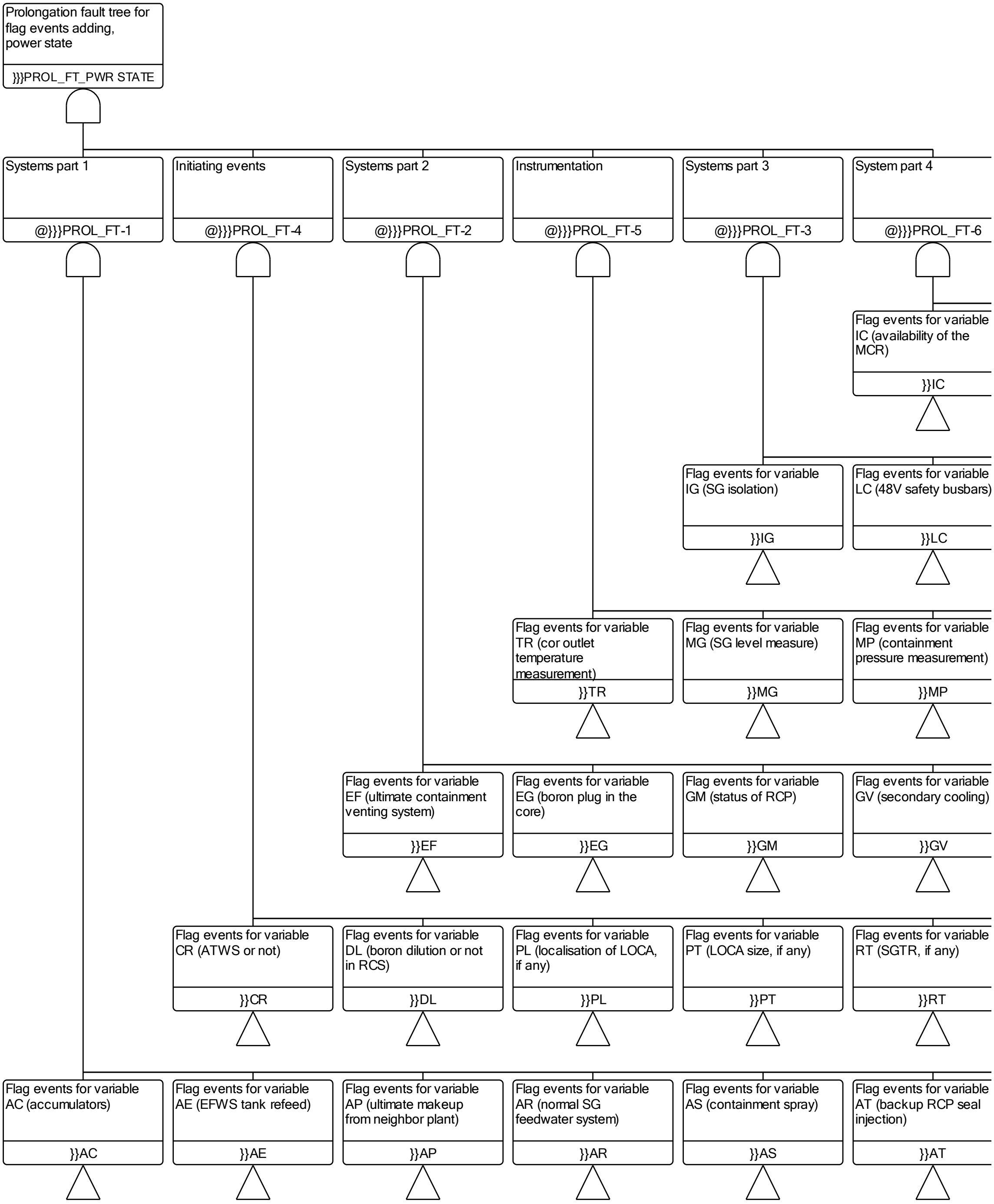
- identify automatically the existing PDS (no need to define the potentially existing PDS manually),
- built the sub MCSs set corresponding to each PDS and,
- based on this sub MCSs set, define the frequency of each PDS.

These inputs (list of PDS and their frequencies) are then transmitted to the L2PSA software (the software in charge of the Accident Progression Event Tree (APET) modeling).
Further information about OIPK software is presented in section 2.4.

2.3. Implementation of this method in L1PSA

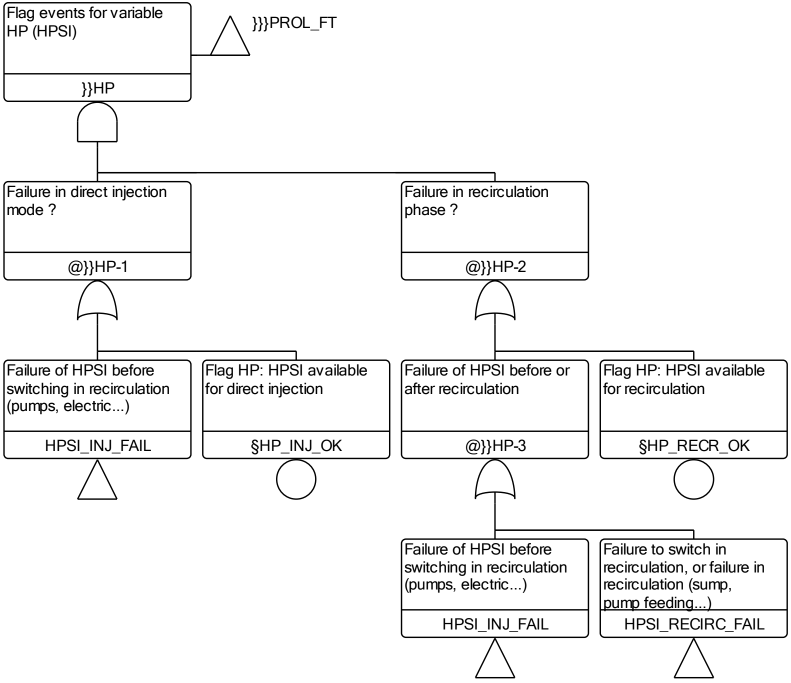
Step one: construction of the prolongation fault tree
The construction of the extended MCSs set is obtained with a unique fault tree in charge of adding flag events. This fault tree contains, under an AND gate, one sub fault tree for each interface variable. This unique fault tree is named the "prolongation fault tree". The sub fault trees implement the flag events corresponding to each interface variable. The figure 1 gives an example of a prolongation fault tree.

Figure 1: Example of a prolongation fault tree



Each sub fault tree, corresponding to an interface variable, is made of one top AND gate and sub OR gates as shown in figure 2.

Figure 2: Example of a sub fault tree for flag event adding



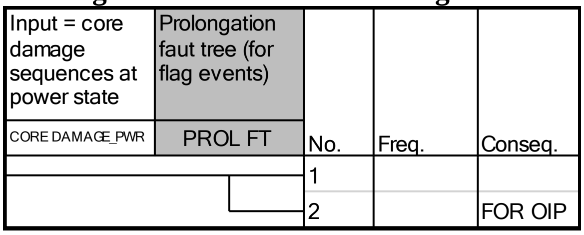
As shown on the above figure, if the failure of HPSI in direct injection mode is given in the MCS from L1PSA (due to CCF on the safety busbars, for example), the top gate of the fault tree HPSI_INJ_FAIL is true and the flag events are not added into the extended MCS.

In these sub fault trees, the same systems modeling as the one used in L1PSA fault trees is used to reduce the work load and to ensure consistency[a]. For example, the system fault tree HPSI_INJ_FAIL is also used in L1PSA modeling (for example, it is included in the function event LHSI of figure 4).

Step two: integration of the prolongation fault tree in the event trees

The "failure" coming from the prolongation fault tree has to be considered in all core damage sequences of the L1PSA event trees. To do so, it is possible, either to put the prolongation fault tree in a unique bridge tree connected to all the L1PSA sequences leading to core damage (see figure 3) or, if the PSA software does not allow event trees linking, to add it in all core damage sequences (see figure 4).

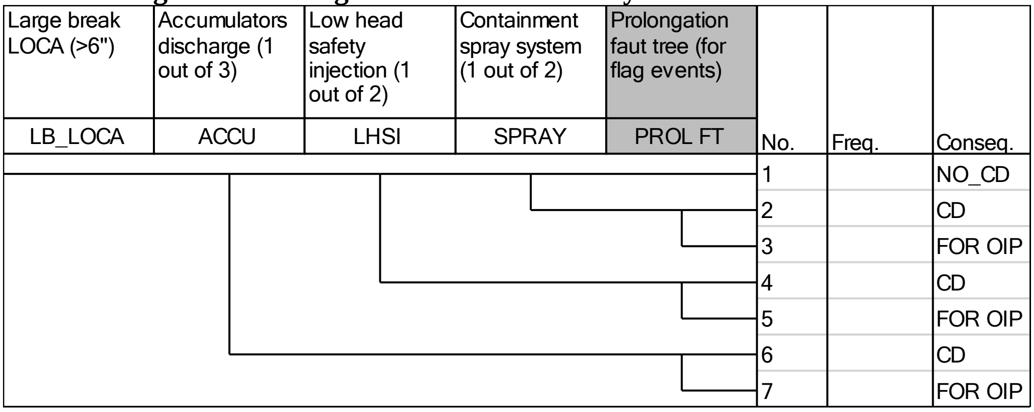
Figure 3: Utilization of a bridge tree



---

[a] A careful attention has to be paid on the consistency between the initial L1PSA system fault trees and the added flag events systems fault trees: the failure of a system in the L1PSA must not be missed in the interface.

Figure 4: Prolongation fault tree directly in L1PSA event trees



When quantified, the consequence “FOR OIPK” produces the extended MCSs set which constitutes the input for the OIPK software.

2.4. Automation of the PDSs list definition and PDSs frequency updating with OIPK

To identified, from the extended MCSs set, the PDSs and to calculate their total frequencies, an interface tool, named OIPK, has been developed by IRSN.
This tool uses, as inputs, the extended MCSs set in a text format (required) and the list of basic events of the extended PSA (optional).
As an output, this tool can produce:

- the list of PDSs and their frequency in a text format adapted to KANT,
- the list of PDS and their frequency in an Excel spreadsheet for results analysis,
- the MCSs set corresponding, for example, to a given PDS or to all the PDS sharing a given value for a given interface variable.

To produce such results, as presented in the step 3 of section 2.2, the OIPK user needs to define the interface variables (name + description). Then, for each interface variable, its possible values have to be defined and expressed in terms of combination of flag events through a graphical interface. As a result, tables similar to Table 1 are produced in OIPK.
One of the advantages of OIPK is that the user does not have to define, a priori, the PDSs that have to be quantified. Based on the interface variables definition made in this software, the existing PDSs are automatically identified and quantified from L1PSA results. The number of PDSs generated depends:

- on the number of interface variables and the number of their values and
- on the cut-off frequency applied when the extended MCSs set has been generated with L1PSA tool.

The computation time to produce PDSs set, once the extended MCSs set has been generated, is limited (about half an hour in the worst case).

3. OTHER IRSN’S SPECIFICITIES REGARDING THE INTERFACE APPROACH

In this section, some other elements relative to the IRSN’s interfacing approach are presented.

3.1. Modeling of systems dedicated to severe accident management

To guaranty a good consistency between the systems modeled in the L1PSA and some active systems useful for severe accident management (containment isolation system, severe accident instrumentation in MCR, reactor building venting …) IRSN has chosen to model these systems in the extended L1PSA and to transfer their status through dedicated interface variables. For example, the filtered containment venting system (FCVS) is modeled in the extended L1PSA (i.e. in the prolongation fault tree) to consider, consistently with other systems, the availability of its electric heaters.

As a result, all the active systems are modeled in the extended L1PSA[a]. Their state is defined for L2PSA event trees, thanks to the interface, in the PDSs. However, a system which is considered as available in the interface may be considered later unavailable in the Accident Progression Event Tree (APET) if its environmental conditions are too degraded (temperature, pressure, radiations…) or in case of energetic phenomena during the severe accident (like steam explosion). These induced failures are easy to consider in KANT since these physical values (temperature, pressure, time…) are transmitted and possibly modified from one node of the APET to another.

3.2. Definition of ASTEC accident simulations to support L2PSA development

As introduced in section 1.1, each L2PSA developed by IRSN is supported by a large set of ASTEC accident simulations (typically between 100 and 200 scenarios of accidents are calculated). These ASTEC simulations are defined to fit with the PDSs. However, the PDSs attributes contain information not useful to define an ASTEC simulation (for example, the status of the containment isolation system, which is only considered in KANT and MER). Thus, different PDS may lead to the same ASTEC run.

To facilitate the definition of needed ASTEC accident simulations, a graphical interface has been implemented in OIPK to allow an additional PDS merging by a “merging” tree formalism. The “head events” considered in these trees are the interface variables. Each sequence corresponds to the definition of an ASTEC run. In the nodes of a merging tree, several values for a given interface variable can be grouped together (if their frequency is to low or if the two values lead to similar plant behavior in the given context). OIPK automatically updates the frequency of each node. It is then easy to identify the branches which are not significant and to adapt the grouping accordingly.

In the figure 6, a (simplified) example is given to illustrate the “merging tree” used to define the ASTEC simulations in case of large break LOCA. In this definition, the interface variable SO (pressurizer safety valves), GV (secondary system availability: EFWS, MSB) and AE (EFWS tank water makeup) are not considered to differentiate the ASTEC simulations. Indeed, in this simplified example, it is assumed that, in case of large break LOCA, neither the pressurizer safety valves opening nor the secondary system cooling will modify significantly the accident progression. To neglect these interface variables in a given context (i.e. large break LOCA context), they can be either not mentioned (like SO) or not used to split sequences (like GV and AE).

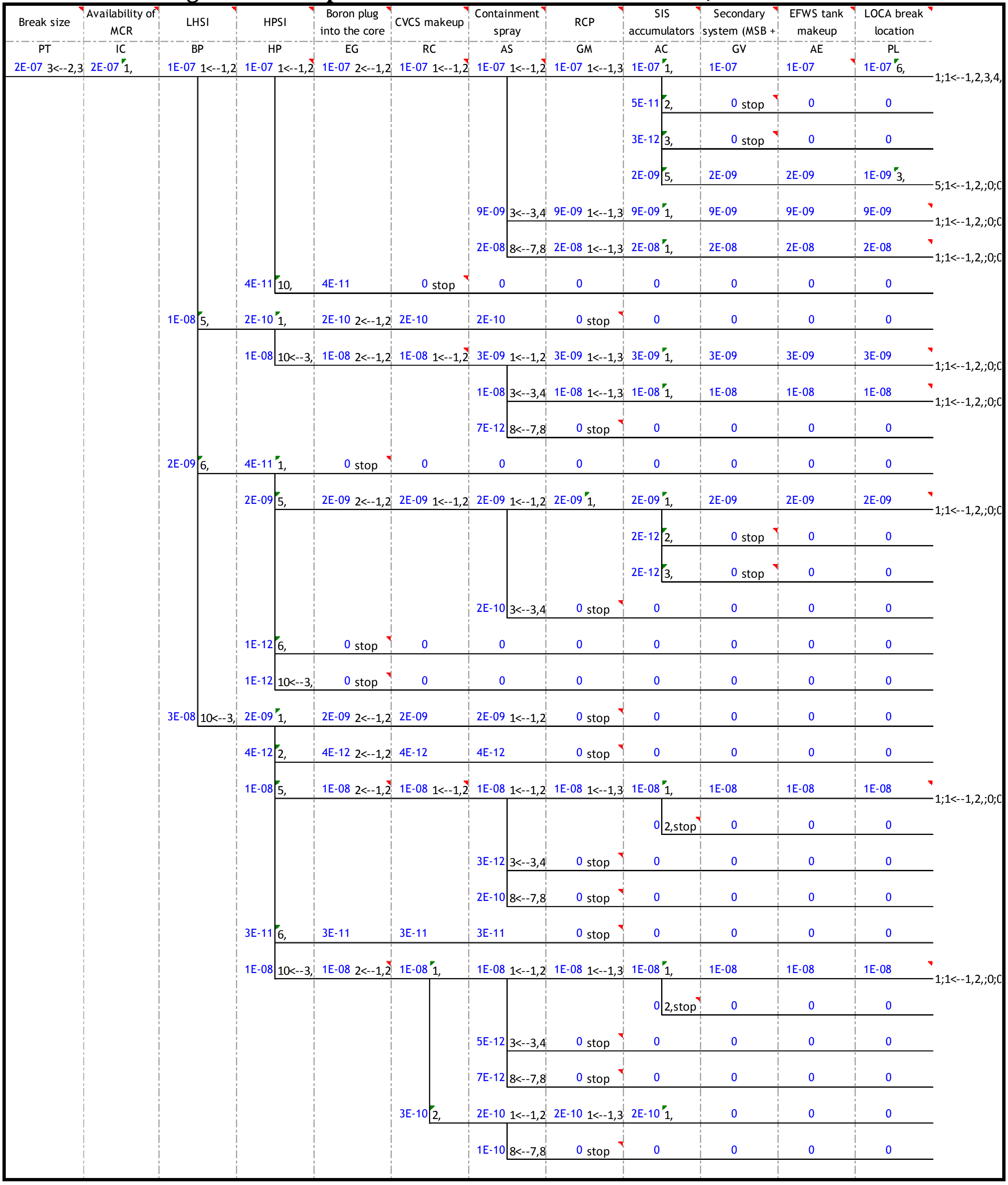
As seen in this figure, some branches are stopped due to their low frequency whereas some others are grouped.

This “merging” tree functionality is a flexible tool to define and document the ASTEC runs definition based on PDS. It can also be used to reproduce some L1PSA event trees and, then, verify that the frequencies are consistent with the ones of the L1PSA sequences.

---

[a] Some passive systems, like the autocatalytic H2 recombineurs, are not modeled in the extended L1PSA since they have no link with the other systems (i.e. recombineurs do not require cooling or electric supply, they neither include shared components).

Figure 6: Example of an ASTEC runs definition tree, from OIPK



4. CONCLUSION

Results obtained:

Due to its automation and its flexibility, this new interfacing approach, developed by IRSN, allows a **transmission** of information between L1PSA and L2PSA **with the same precision as obtained with an integrated model** with an affordable cost. But, thanks to the use of separated and appropriate dedicated tools for L1PSA and L2PSA, parallel work on L1PSA and L2PSA is facilitated. L2PSA can be quantified separately from L1PSA. The L1PSA is not calculated again for the L2PSA quantifications.

After a L1PSA modification, the interface update is a **“button click” update** and the **development of a new interface is strongly facilitated**.

In the same manner, the **adjunction of additional information in the interface is very simple and efficient**. For example, adding a new interface variable is done by simply introducing the corresponding flag events in the prolongation fault tree and rerunning the extended L1PSA[a]. Eventually, a new variable in OIPK has to be declared and OIPK rerun.

If the L1PSA software does not allow event trees linking, this interfacing approach allows to preserve the L1PSA event trees readability[b] (only one additional function event is added in the event trees). In addition, since the extended L1PSA model remains almost as compact as before its extension, **the CPU time to run it is not strongly impacted** (at least with RiskSpectrum tool).

Regarding the PDS validation, **the results produced by OIPK can be easily verified** through the MCSs set presented for each PDS.

The interfacing approach presented in this paper can be **declined for any L1PSA software** using fault trees and event trees. This approach is especially interesting if the L1PSA software does not allow event tree linking since it is an alternative to the bridge trees construction.

This approach is particularly suitable **to implement internal and external hazards** in L1-L2PSA interface. Indeed, since the sub fault trees used in the prolongation fault tree are mainly based on a reuse of L1PSA fault trees, the interface is “automatically” updated when the L1PSA fault trees are adapted to the internal and external hazards modeling. In addition, due to the facility to add an interface variable and to modify the existing ones, it is simple and fast to add hazard specific variables in the interface (for internal fire localization by example).

Perspectives:

Coupled KANT-MER-MERCoR calculations provide a large amount of data, which allows various ways to analyze and present the L2PSA results. In particular, it is possible to assess contribution of each PDS regarding containment failure modes, radioactive release and radiological consequences.

With OIPK, it could be possible to identify the contribution of a given basic event of the L1PSA to the different PDSs. Consequently, it could be feasible to compute automatically the importance measures of the extended L1PSA’s basic events, in regards of the dose consequences of their failure (and not only in regards of core damage). Such importance measures would allow building a better hierarchy of components importance, especially for the components dedicated to severe accident management and to those contributing to the core damage prevention and to the releases mitigation.

References

[1] AIEA, “*Development and Application of Level 2 Probabilistic Safety Assessment for Nuclear Power Plants”*, Specific Safety Guide N° SSG-4.

[2] “*ASAMPSA2 best-practices guidelines for L2PSA development and applications - Volume 2 - Best practices for the Gen II PWR, Gen II BWR L2PSAs. Extension to Gen III reactors”*, 2013; collaboration between IRSN, GRS, NUBIKI, TRACTEBEL, IBERINCO, UJV, VTT, ERSE, AREVA NP GmbH, AMEC NNC, CEA, FKA, CCA, ENEA, NRG, VGB, PSI, FORTUM, STUK, AREVA NP SAS, SCANDPOWER,
available on internet at http://www.asampsa.eu or http://www.asampsa2.eu

[3] N. Rahni et al., “*A methodology for the characterization of the severe accident consequences and the results presentation in level 2 probabilistic safety assessment”*, conference PSA 2011, Wilmington

[4] T. Durin et al., “*Very fast running codes for the characterization of severe accident radiological consequences and the results presentation in level 2 PSA*”, conference PSAM 11, Helsinki

---

[a] That is the L1PSA model with the prolongation fault tree added.

[b] This was not an issue for IRSN since RiskSpectrum allows event trees linking.