

MELCOR-To-MELCOR Coupling Method in Severe Accident Analysis Involving Core and Spent Fuel Pool

Hector Lopez, Alessandro Petruzzi, Walter Giannotti, Domenico De Luca

Summary—A lot of effort has been spent to prevent the occurrence of SA in nuclear plant and to develop Severe Accidents (SA) Management to mitigate the consequences of a SA. Those consequences are mainly related to limit the release of fission product to the environment. The core in the vessel is not the only source of fission products as the Spent Fuel Pool (SFP) hosting the fuel removed by the core is, in some NPP, inside the containment and SA conditions can also occur. This is especially important in reactors having proximity between the RPV and SFP such as the VVER-1200. This close proximity implies that any SA occurring in the SFP potentially affects the RPV and vice-versa. This potential combination might cause unexpected evolution in the SA progression to whom the safety systems are not able to contain. MELCOR code is a widely used, flexible powerful SA code but it is incapable (due to the uniqueness of the COR package use inside the same input) to reproduce a situation in which both the fuel in vessel core and the fuel in the SFP, inside the same containment, are going to experience a severe accident scenario. The current study presents a MELCOR-to-MELCOR coupling method to simulate simultaneously scenarios with both, core and SFP, as sources capable of H₂ generation, fuel damage and FP release in a VVER-1200 NPP. The coupling is performed by running two simulations in parallel and with the data exchange supervised and managed by a dedicated Python coupling supervising script developed at NINE.

Keywords—MELCOR, SA, severe accident, spent fuel pool, coupling, VVER-1200

I. INTRODUCTION

In Severe Accidents (SA) Management, the main target is to avoid the release of fission products to the environment and to limit the dose to the population that is the general regulatory safety requirement. A lot of effort has been spent to manage the consequences of the core damage constituting the begin of the SA phase in the accident progression. However, the core in the vessel is not the only source of fission products in the containment as the Spent Fuel Pool (SFP) hosting the fuel removed by the core is, in some NPP, inside the containment and SA conditions can occur in the pool.

An effect of the occurrence of a SA in the SFP is the intensive

generation of H₂ due to the heat up and oxidation of the rod cladding if the coolant inventory is lost or if system providing cooling to the pool fails. The impact of the H₂ is enhanced by the fact that the spent fuel pool releases H₂ directly in the containment leading to scenarios with large amounts of H₂ in the containment in early stages of the accident. An SA occurring in the SFP could potentially lead to the melting of the contained fuel rods leading to the damage of the structures in the containment. This is especially important in reactors having proximity between the RPV and SFP such as the VVER-1200. This close proximity implies that any SA occurring in the SFP potentially affects the RPV and vice-versa. This potential combination might cause unexpected evolution in the SA progression to whom the safety systems are not able to contain.

MELCOR is a fully integrated, engineering-level computer code developed at Sandia National Laboratories for the U.S. nuclear Regulatory Commission whose primary purpose is to model the progression of accidents till the SA phase in LWR NPPs. MELCOR code is capable to analyze a large spectrum of severe accident phenomena in both BWR and PWR. Although MELCOR is a widely used, flexible powerful SA code, it is incapable to reproduce a situation in which both the fuel in vessel core and the fuel in the SFP, inside the same containment, are going to experience a severe accident scenario due to the uniqueness of the COR package use inside the same input [1][2].

Initial attempts to circumvent this MELCOR limitation have been performed by using separate integral calculations with subsequent combination of individual core and SFP SA effects but this approach was not suitable to fully capture the interactions between the systems. Thus, a MELCOR-to-MELCOR coupling approach have been widely assumed to perform such calculations.

Previous studies on the implementation of MELCOR-to-MELCOR coupling method present documented examples of the use of the Parallel Virtual Machine (PVM) Message Exchange [3]. This virtual machine created by the executive program after reading the input monitors the information to be exchanged and coordinates the advancement through time.

Despite presenting promising results in the aforementioned studies [3], the main limitation with the use of the PVM is the requirement of the user having access to the MELCOR source code in order to link the PVM libraries with the code. This requirement makes the PVM unsuitable for the present study. Thus, the development of a new coupling method that does not require access to the source code was done in N.I.N.E..

(Corresponding author: Hector Lopez)

Hector Lopez, Alessandro Petruzzi, Walter Giannotti, and Domenico De Luca are with the Nuclear and Industrial Engineering (NINE) Lucca, Italy (e-mails: h.lopez@nineeng.com, a.petruzzi@nineeng.com, w.giannotti@nineeng.com, d.deluca@nineeng.com)

The paper is organized as follows. The description of the coupling method developed by Nuclear Industrial Engineering (N.I.N.E.) is presented in Section 2. Section 3 show the proof of concept scenario designed to test the coupling and, the results of the aforementioned scenario are shown in Section 4. Lastly, the future work is described in Section 5.

II. N.I.N.E. MELCOR-TO-MELCOR COUPLING

The basic concept of the coupling method consists into run two simulation models in parallel, each one representing a portion of the overall domain, with constant data exchange between them through the interface. Based on the previous definition two major elements are defined; the simulation models domains and the data exchange management. Both elements are described in this Section.

In case of a MELCOR-to-MELCOR coupling, each domain is defined by the aforementioned limitation of the code; the presence of a COR package. For the scenario of interest, the domains consist on the CORE domain which includes the RPV, cooling circuits and most of the containment system and the SFP domain which includes the fuel pool itself and the remaining part of the containment system. Each domain is assigned to a separate MELCOR calculation with dedicated input and different inputs COR package (the CORE and the SFP respectively). Figure 1 depicts the domains used for a VVER-1200 simulation model.

As mentioned above, the coupling method requires a constant data exchange between both domains in order to provide accurate results. Figure 2 shows the basic chart flow of the coupling method function. As the figures suggests through the red lines, on each time step, the data needed is extracted from the output of one of the files and sent to the input of the other file, which is modified accordingly with the new information received. After the data is exchanged between both domains, both calculations are repeated with the updated values until a convergence check is passed.

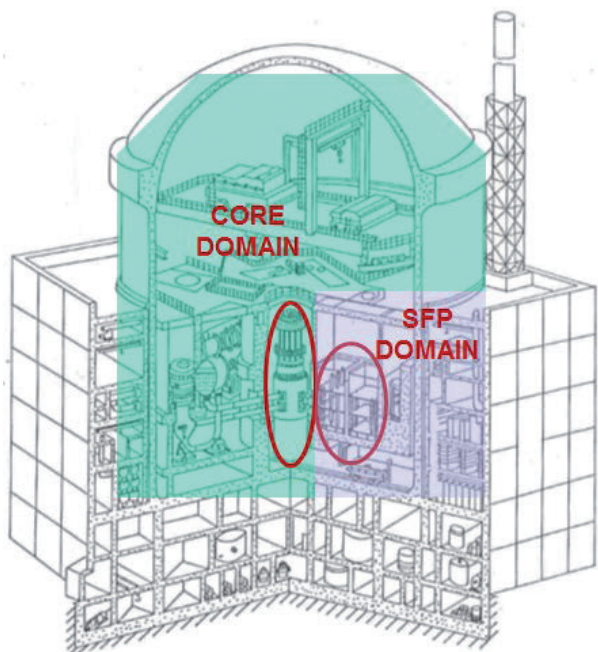


Fig. 1: VVER-1200 MELCOR-to-MELCOR Coupling Domains.

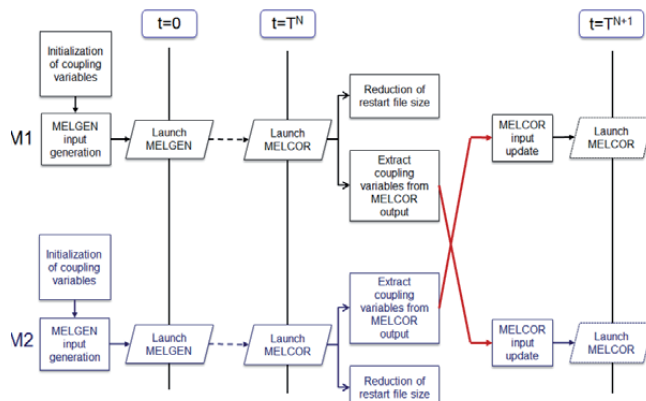


Fig. 2: N.I.N.E. MELCOR-to-MELCOR Coupling Flow Chart.

Thus, how to manage the data exchange is a key element when developing the coupling. Weaver et. al. [4] describe the different coupling schemes used by RELAP and other codes. The most common coupling methods use either explicit or semi-implicit coupling schemes.

The first of the coupling schemes is the explicit scheme. In this methodology, although the data exchange occurs at the beginning of the time step, T^n , in both directions, pressures in the boundary volumes are held constant throughout the time step and only updated at the end. This condition causes the systems to be limited by the sonic Courant condition. Scenarios with larger Courant numbers will turn out in undesired oscillations in the results of the calculations. In addition, there is no control and correction of truncation error during the data exchange. Previous studies have proved this restriction (Aumiller et. al.) [5]. This restriction makes fully explicit coupling impractical. Figure 3 and Figure 4 show the data exchange of the explicit scheme diagram at a given time step.

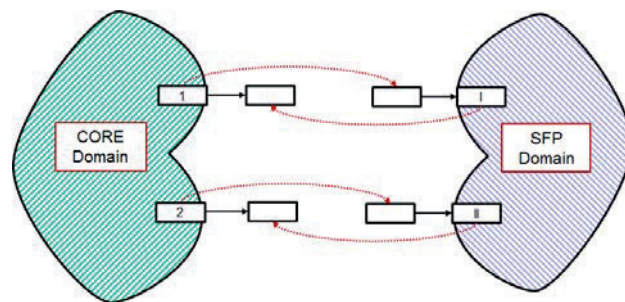


Fig. 3: Schematic of Explicit Coupling Methodology.

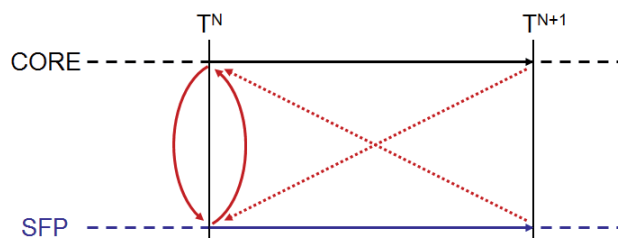


Fig. 4: Explicit Coupling Methodology Time Step Advancement.

The semi-implicit coupling scheme, on the other hand, uses a Master-Slave scheme. To use this scheme, exchange volumes are created in both inputs and act as interface between both domains as depicted in Figure 5. Through these volumes, as shown in Figure 6, the Master sends variables to the Slave at the beginning of the time step, then the Slave advances of time step, from T^n to T^{n+1} . After

advancing, the Slave sends variables back to the Master following the equation (1).

$$X = aX^n + bX^{n+1} \tag{1}$$

Where:

X is the variable exchanged;

“a” and “b” are constants and equal to 0.5 (Crank-Nicholson).

After receiving the variables, the Master advances of time step, from T^n to T^{n+1} . This process is repeated with the updated values until the convergence check is passed. Using this methodology allows for a consistent energy and mass flow rates exchange between the domains that results in pressure calculation carried out independently on each domains. Due to these facts, the semi-implicit methodology results in a more stable and self-correcting coupling scheme and in consequence, it is used for the development of a coupling methodology.

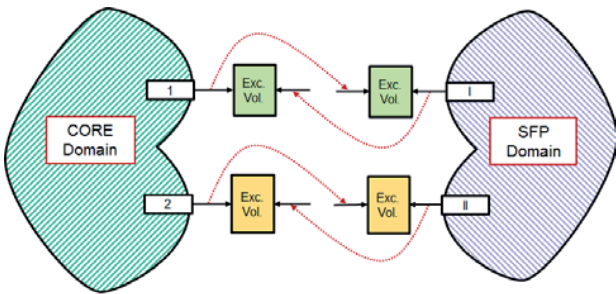


Fig. 5: Schematic of Semi-Implicit Coupling Methodology.

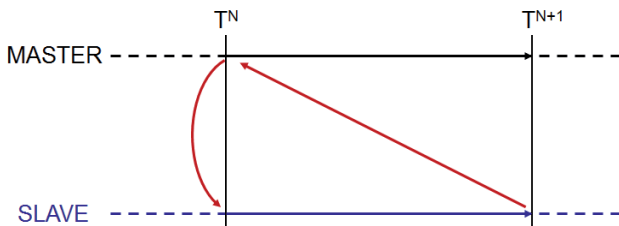


Fig. 6: Semi-Implicit Coupling Methodology Time Step Advancement.

After selecting a semi-implicit approach, a coupling method has been developed at N.IN.E. for MELCOR 1.8.6. In order to create this new methodology, a dedicated Python coupling supervising script has been designed with the purpose of managing the MELCOR-to-MELCOR data exchange. MELCOR 1.8.6 has been selected initially because its numerical card identification and syntax are easier to implement in the Python script than the alphanumeric approach used in newer versions of the code.

The Python script design is based on the generation and update of restart data at each time step. As depicted in Figure 2, after the generation of the restart file at a given time step, T^n , the data required is extracted from the output and sent to the corresponding input through dedicated control functions. At the same time, the size of the restart file is restrained by the use of MELCOR options in the EXEC package to avoid large size files. At each time step, after receiving the corresponding data through the control functions, a new input file for each domain is generated with the updated CF values. Once the semi-implicit convergence check is met, the code advances the simulations to the next time step, T^{n+1} .

The script includes the possibility to select the coupling exchange variables, “a” and “b” from equation (1). In addition, a time step handler is also implemented in case that the convergence is not reached. This design allows for a synchronous coupling (data

exchange on every time step) meaning that the coupling time step is equal to the calculation time step at any given moment.

As a semi-implicit methodology, the Python scrip requires the presence of interface elements between the domains to allow the data exchange. The script is designed for two different types of interface elements; exchange volumes and exchange heat structures.

- Exchange volumes are used for material and enthalpy exchange between the two domains. The energy and mass flow rates are exchanged through the coupling algorithm while the pressure is computed independently at exchange volume and used to perform the convergence check;
- Exchange heat structures for heat exchange between the two domains. The boundary temperatures are exchanged through the coupling algorithm while the heat flux is computed independently at the exchange heat structure and used to perform the converge check.

Fig. 7 depicts the exchange elements between the CORE and SFP domains in a VVER-1200.

At this stage of development, the information exchange related to the RN package is not considered, but it will be implemented in later stages.

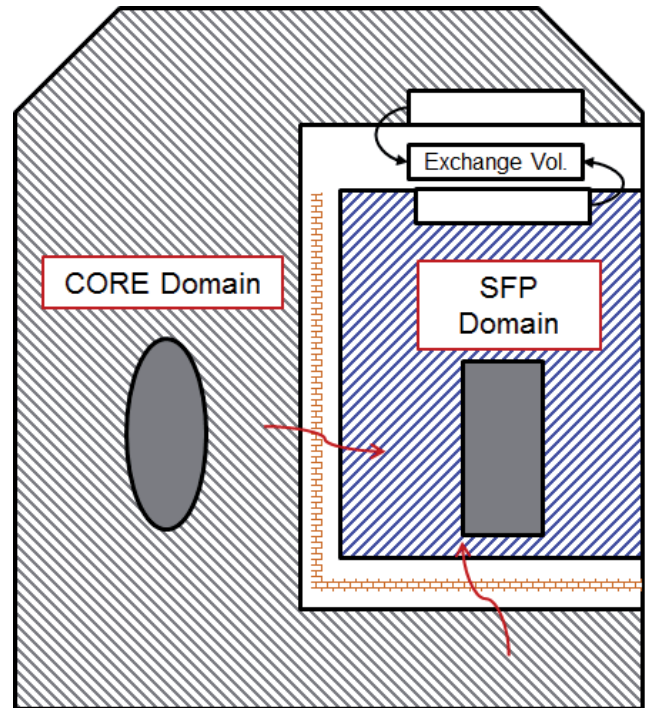


Fig. 7: VVER-1200 Exchange Elements.

III. PROOF OF CONCEPT SCENARIO – DESCRIPTION

The main objective of the development of the coupling mechanism is to analyze the effect of the SFP in accidents in VVER-1200 by using a MELCOR-to-MELCOR coupling to circumvent the uniqueness of the COR package and having, effectively, two active cores.

Prior to implement the coupling methodology into a fully scaled VVER-1200, a proof of concept was redeemed necessary to validate the developed Python supervising script. A simplified nodalization, depicted in Figure 8, has been designed to represent both, CORE and SFP domains with the containment dome acting

as exchange volume. The nodalization uses the operational and geometrical parameters of a prototypical VVER-1200 for the RPV and SFP but with a simplified PS/SS and containment.

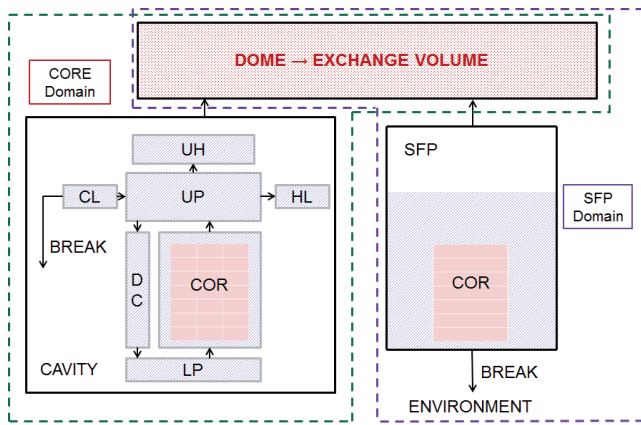


Fig. 8: Proof of Concept Nodalization.

As mentioned above, the nodalization depicted in Figure 8 describes both CORE and SFP domains as follows.

- CORE domain, in green, includes
 - RPV formed by 5 CVs; LP, core, DC, UP and UH filled with water;
 - COR package of 3 rings and 7 axial levels. This COR package reproduces the LP and active core region cells from EL 16.93 m to EL 22.38 m. The first three levels represent the LP internals while levels 4-7 represent the active core;
 - Main loops formed by 2 CVs; HL and CL filled with water;
 - RPV Cavity of 294 m³;
 - Containment dome of 500 m³ acting as the exchange volume;
 - Break that connects the CL with the RPV cavity (0.01 m², 2% of the CL area).
- SFP domain, in purple, includes
 - SFP of 200 m³ and 7.4 m water level;
 - COR package of 1 ring and 7 axial levels to reproduce the spent fuel. This COR package reproduces the spent fuel assembly cells from EL 20.7 m to EL 24.8884 m. Level 3-6 represent the fuel region of the spent fuel assembly. Levels 1, 2 and 7 represent the region of the assembly below and above the fuel.;
 - Containment dome of 500 m³ acting as the exchange volume;
 - The environment;
 - Break that connects the SFP with the environment (0.01 m²).

Table 1 shows the initial conditions applied to the nodalization as well as the main events of the transient. The core COR package generates a power equal to 3GWt of operating power decay heat while the SFP COR package power has been assumed constant through all the transient.

Initially, all the cells of both COR packages are well submerged in water while both domains are stable on normal operation conditions as shown in Table 1. At the start of the transient, both breaks

occur simultaneously at the CL and at bottom of the SFP.

The breaks used in the scenario have been arbitrary sized; the objective is to set up a small break on each location in order to have slower mass depletion and observe the transient clearly

TABLE 1: INITIAL CONDITIONS AND MAIN EVENTS.

INITIAL CONDITIONS	
Primary Side	
Property	Value
Pressure [MPa]	15.7
Temperature [K]/[°C]	570/297
Core power [MW]	3000
Containment	
Pressure [MPa]	0.1
Temperature [K]/[°C]	330/57
SF decay power [MW]	5
Environment	
Pressure [MPa]	0.1
Temperature [K]/[°C]	293/20
MAIN EVENTS	
CL break opening time [s]	0.0
SFP loss of coolant time [s]	0.0

IV. PROOF OF CONCEPT SCENARIO — RESULTS

Once the nodalization has been defined, the transient is simulated by running both inputs (one dedicated to each domain) and the Python script. For this scenario, the CORE input has been established as the Master while the SFP has been established as the Slave for the semi-implicit methodology purposes. The transient described in Section 3 is simulated for 1000 s. The objective is to prove that the Python script is able to manage the data exchange providing reliable results while simulating the evolution of two COR packages under different conditions. In the next figures, the main results are discussed.

Figure 9 shows the pressure in all the containment-related CVs; the cavity, the SFP and the containment dome. All three compartments present identical behavior despite being in different inputs (cavity is in the CORE domain while SFP is in the SFP domain and the containment dome is in both of them) implying that the script is able to keep the convergence of the pressure at each time step of the transient.

The pressure suffers an initial increment until the water inventory of the primary system is released into the cavity at which point starts to stabilize until the complete depletion of the SFP at around 280 s due to the release of SFP water into the environment. At this point, the containment suffers a depressurization.

The situation originated in the primary system due to the break in the CL at 0.0 s causes a sudden depressurization of the system and a massive relocation of the RPV water into the cavity compartment. Figure 10 shows how the core is completely uncovered in 60 s after the break. On the other hand, the SFP suffers a slower depletion due to the mass relocation into the environment and its COR package is not fully uncovered until 280 s.

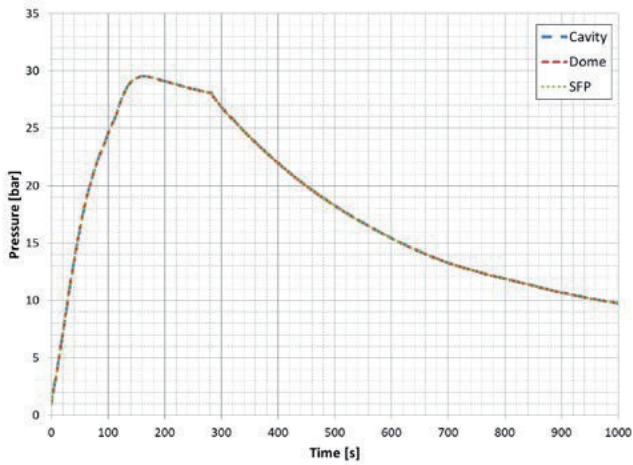


Fig. 9. Containment Pressure.

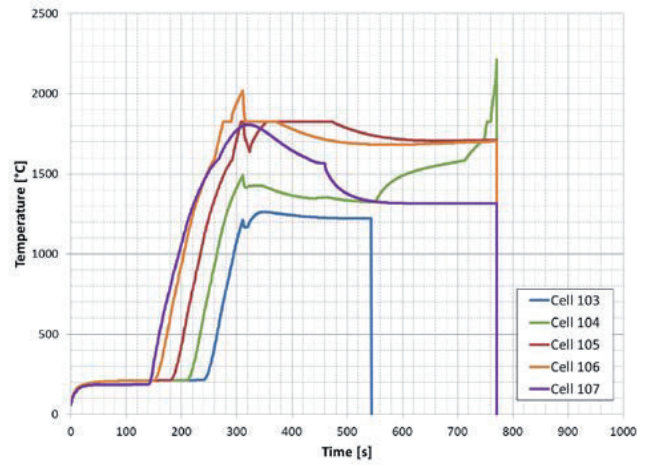


Fig. 12. SFP Domain Cladding Temperature.

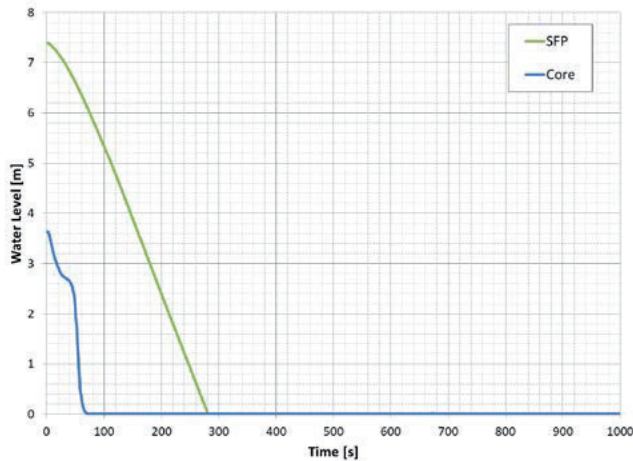


Fig. 10. Core Region and SFP Water Level.

Besides the convergence of the pressure between both domains, the most important parameters to evaluate are those related to the COR package. Figure 11 and Figure 12 depict the cladding temperatures of the CORE domain COR package (channel 1) and the SFP domain COR package respectively. The temperatures depicted correspond to the axial levels with active fuel (4-7 for CORE and 3-7 for SFP).

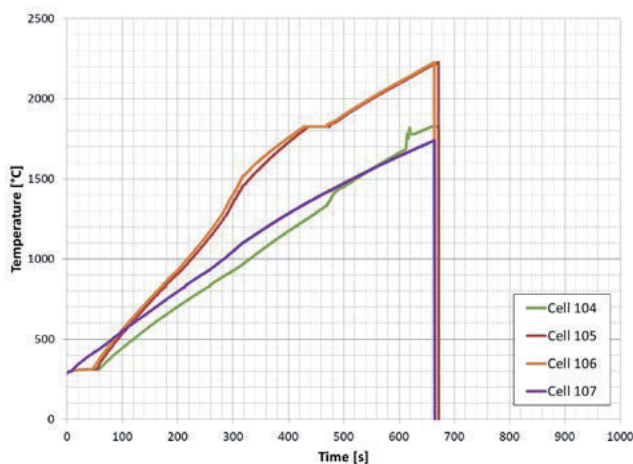


Fig. 11. CORE Domain Cladding Temperature.

As expected, despite being connected, each COR package predicted a different evolution based on the transient conditions.

First, the CORE domain temperatures started to increase right after the core became uncovered at 50 s due to the decay heat of the fuel. The temperature kept rising on a steady paste due the lack of any recovery mechanism with the oxidation and H₂ release once the temperature surpass the 1100 °C at around 220 s (Figure 15). The code predicted a complete collapse of the channel 1 of the core around 650 s after the break. This relocation into the LP caused the eventual failure of the lower head.

The temperatures of the SPF, on the other hand, are kept under control until the COR cells start to be uncovered. In this case, the onset of the temperature rise for each cell is clearly differentiated since the SFP level decreases much more slowly that in the RPV (Figure 10). Finally, the code predicts a complete collapse around 780 s.

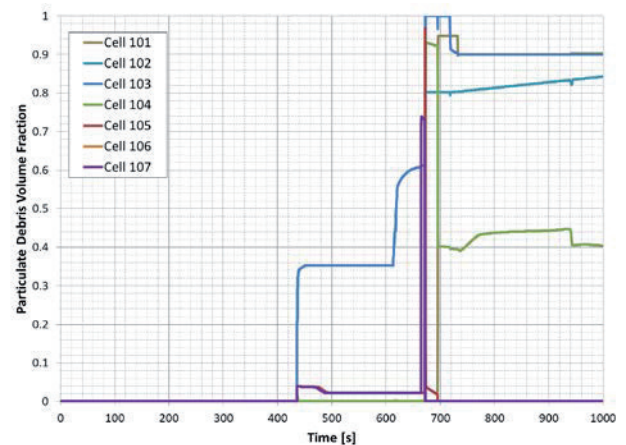


Fig. 13. CORE Domain Particulate Debris Volume Fraction

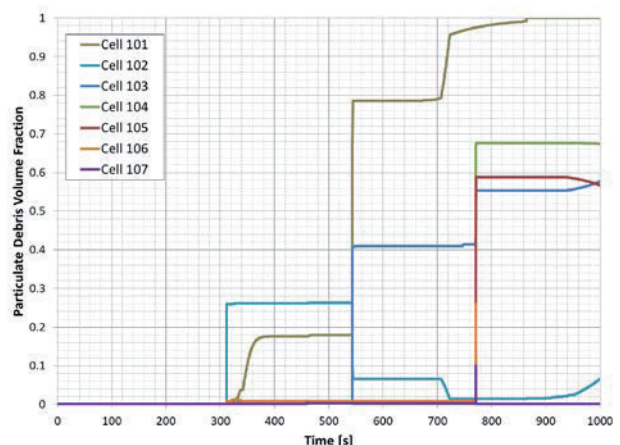


Fig. 14. SFP Domain Particulate Debris Volume Fraction.

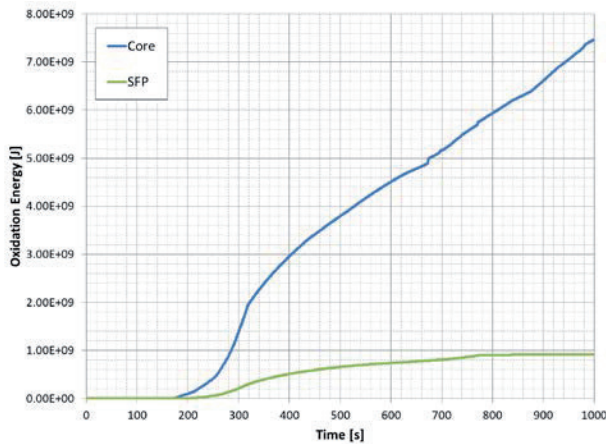


Fig. 15. Oxidation Energy.

Analogously to the cladding temperatures depicted in previous figures, Figure 13, Figure 14 and Figure 15 present other COR-related parameters; the particulate debris and the oxidation energy. Analogously to the cladding temperatures, each domain presented an independent core degradation.

Lastly, it is worth saying that the additional time added by the coupling into the simulation time has been negligible but this is mainly due to the simplicity of the simulation model used for the proof of concept. Further analysis on full NPP simulation models have to be performed in order to address the real impact on the simulation time.

The results presented in this Section, although being from a simple proof of concept, are encouraging results as they demonstrate that the coupling mechanism developed at N.IN.E. is capable to capture both core degradation independently while continuously manage and supervise the data exchange so any phenomena occurring in one domain can affect the other.

V. FUTURE WORK

With the final goal in mind, the next step will be the implementation of the interface heat structures to validate if the heat transfer and boundary temperatures between the domains are correctly predicted. Once that is validated, the implementation of the coupling mechanism into a fully scaled VVER NPP, as the one depicted in Figure 16 and Figure 17, will be performed.

With the final implementation, the analysis of selected scenarios involving the RPV and the SFP will be performed.

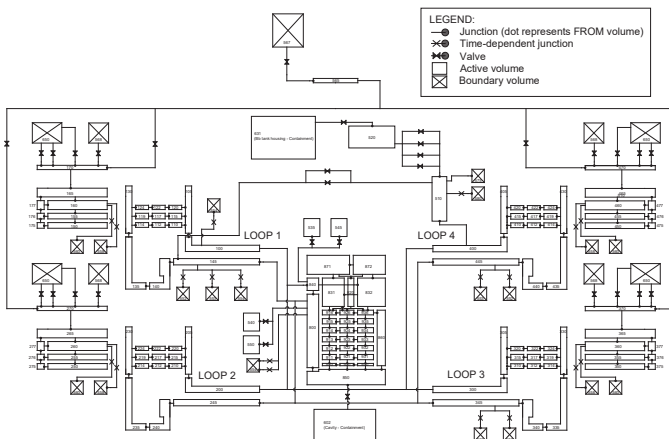


Fig. 16: VVER NPP Primary/Secondary System Nodalization.

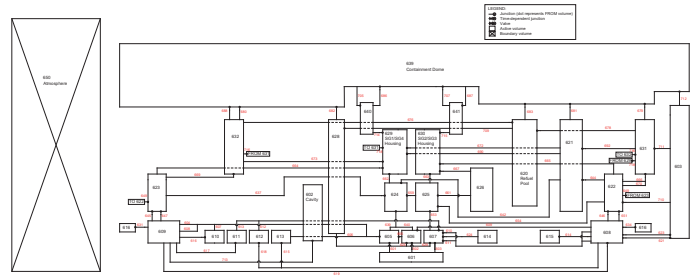


Fig. 17. VVER NPP Containment Nodalization.

VI. CONCLUSIONS

As stated at the beginning of the paper, the RPV is not the only source of fission products in the containment as the SFP hosting the depleted fuel is, in some NPP, inside the containment and SA conditions can occur in the pool. The impact of the SFP H₂ directly released into the containment can alter the expected evolution of a given transient. This becomes even more relevant in reactors having proximity between the RPV and SFP such as the VVER-1200. This close proximity implies that any SA occurring in the SFP potentially affects the RPV and vice-versa.

As such, the main objective of the present paper was to establish a coupling mechanism for MELCOR that allowed to circumvent the code limitation of one unique COR package in order to analyze accidents that may involve both, the RPV and the SFP. One of the requirements was that this mechanism should be used even if access to the source code was not possible. This was performed by N.IN.E. by developing a coupling mechanism using a dedicated Python supervising script. This script manages the data exchange between the different domains (inputs) by a semi-implicit scheme and updates the corresponding input at each time step with dedicated control functions.

The results from the proof of concept presented in this paper proved to be encouraging that demonstrated the capabilities of the aforementioned coupling, being able to capture both core degradation independently while continuously managing the data exchange. In addition, the tax observed on the simulation time due to the coupling was negligible most certainly due to the simplicity of the simulation model used. Although further analysis are required in more complex nodalization in order to assess the real impact.

The future steps involve further validation of the capabilities of the mechanism in order to perform its final implementation in a real VVER reactor nodalization and the analysis of the subsequent accident scenarios involving the SFP as a secondary core.

REFERENCES

- [1] NRC, "MELCOR Computer Code Manuals Vol. 1: Primer and Users' Guide Version 1.8.6", NURE/CR-6119, Vol.1, Rev. 3, SAND 2005-5713, Washington, September 2005.
- [2] NRC, "MELCOR Computer Code Manuals Vol. 2: Reference Manuals Version 1.8.6", NURE/CR-6119, Vol.2, Rev. 3, SAND 2005-5713, Washington, September 2005.
- [3] R. K. Cole, Jr. "Coupling of MELCOR to Other Codes under an Executive Program using PVM Message Exchange", 2002 RELAP5 Users Seminar, Park City, Idaho, September 4-6 2002.
- [4] W. L. Weaver, E. T. Tomlinson, and D. L. Aumiller, "An Executive Program for use with RELAP5-3D", B-T-3394, 2001 RELAP5 Users Seminar, Sun Valley, Idaho, September 5-8 2001.
- [5] Aumiller, D. L., Tomlinson, E. T., Bauer, R. C., 2001a, "A Coupled RELAP5-3D/CFD Methodology with Proof-of-Principle Calculation" Nuclear Engineering and Design, Vol. 205, pp 83-90.