



MELCOR – DAKOTA coupling for uncertainty analyses in the SNAP environment/architecture

Fulvio Mascari^{a,*}, Andrea Bersano^{a,*}, Giuseppe Agnello^b, Michela Angelucci^c, Jesse Phillips^d, David Luxat^d

^a ENEA, Nuclear Safety, Security and Sustainability Division (FSN-SICNUC), Via Martiri di Monte Sole 4, 40129 Bologna, Italy

^b University of Palermo, Department of Engineering, Viale delle Scienze, Edificio 6, 90128 Palermo, Italy

^c University of Pisa, Department of Civil and Industrial Engineering (DICI), Largo Lucio Lazzarino, 56122 Pisa, Italy

^d Sandia National Laboratories, 1515 Eubank SE Albuquerque, 87123 New Mexico, USA

ARTICLE INFO

Keywords:

MELCOR

DAKOTA

SNAP

Uncertainty analysis

Severe Accident

ABSTRACT

Uncertainty estimation to assess figures of merit characterizing evolution of a severe accident transient is a topic of current investigation in development of best-estimate plus uncertainty methodology. The probabilistic method to propagate input uncertainty is one of the methodologies used to develop Uncertainty Analyses (UAs). Using this methodology, UAs are performed by sampling probability distributions that describe the range of possible values that computer simulation model inputs can have. For each sample (or realization) of a set of uncertain input parameters, a computer simulation is performed. From the range of code simulation results obtained for each input realization, a distribution of code results is obtained. In this process, the distribution of input uncertainties is propagated to obtain a distribution of possible code results (i.e., the code output uncertainty). This probabilistic methodology is facilitated using Uncertainty Tools (UTs), which can be coupled with the accident analysis computer code to perform an UA. One of the UTs currently available is DAKOTA, developed by Sandia National Laboratories. DAKOTA is also provided as a SNAP plug-in. SNAP is a graphical user interface designed to support the use of USNRC codes (MELCOR, TRACE, etc). This paper is entirely derived from the NUREG/IA-532 issued by USNRC in April 2023 (Mascari et al., 2023) and has as a major target to summarize the main needs of UA in severe accident, the main element of the probabilistic method to propagate input uncertainty, and the workflow within SNAP to assist other interested analysts with their applications given they are members of the USNRC Cooperative Severe Accident Research Program (CSARP).

1. Introduction

Considering the complexity of and the numerous interacting/inter-related phenomena/processes that occur during the course of a Severe Accident (SA) evolution, state-of-the-art SA integral codes (Mascari et al., 2019; Van Dorsselaere et al., 2018; de la Rosa Blul et al., 2018) have come to play an important role in understanding the overall plant behavior. Codes such as ASTEC (Chatelard et al., 2016), MAAP (Fauske

& Associates, LLC, 2013), and MELCOR (Sandia National Laboratories, 2015) are relied on in the U.S. and internationally to evaluate accident progression and determine characteristics of radiological release to the environment. These analytical capabilities enable the determination of the main Figure of Merit (FOMs) used in a range of safety case development and regulatory decision-making applications. These capabilities have also been used to inform and evaluate accident management strategies.

Abbreviations: ASTEC, Accident Source Term Evaluation Code; BDBA, Beyond Design Basis Accident; BEPU, Best Estimate Plus Uncertainty; CDF, Cumulative Distribution Function; CRP, Coordinate Research Project; CSARP, Cooperative Severe Accident Research Program; DAKOTA, Design Analysis Kit for Optimization and Terascale Application; DBA, Design Basis Accident; ENEA, Agenzia Nazionale per le Nuove Tecnologie L'ENERGIA E LO SVILUPPO ECONOMICO SOSTENIBILE; FOM, FIGURE of Merit; IAEA, International Atomic Energy Agency; LWR, Light Water Reactor; MAAP, Modular Accident Analysis Program; MUSA, Management and Uncertainties of Severe Accidents; PDF, Probability Density Function; PWR, Pressurized Water Reactor; RELAP, Reactor Excursion and Leak Analysis Program; SA, Severe Accident; SNAP, Symbolic Nuclear Analysis Package; SNL, Sandia National Laboratories; SOARCA, State-of-the-Art Reactor Consequence Analyses; TRACE, TRAC/RELAP Advanced Computational Engine; UA, Uncertainty Analysis; USNRC, U. S. Nuclear Regulatory Commission; UT, Uncertainty Tool.

* Corresponding authors.

E-mail addresses: fulvio.mascari@enea.it (F. Mascari), andrea.bersano@enea.it (A. Bersano).

<https://doi.org/10.1016/j.nucengdes.2024.113078>

Received 28 December 2023; Received in revised form 27 February 2024; Accepted 29 February 2024

0029-5493/© 2024 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

Several models/correlations have been implemented in the state-of-the-art SA codes and must be set by a code user during the input deck development. Models/correlations that are implemented in a SA code reflect the state-of-the-art knowledge of SA phenomena/processes. However, even though several experimental campaigns in the field of SA phenomena (OECD/NEA/CSNI, 1992; OECD/NEA/CSNI, 1999; OECD/NEA/CSNI, 2001; OECD/NEA/CSNI, 2009; OECD/NEA/CSNI, 2014) have been performed and provided a valuable “assessment database” (Mascari et al., 2015) to assess SA simulation tools, the analyses of the current state-of-the-art shows that there is the need to reduce some uncertainties¹ still present (Mascari et al., 2019; Van Dorsselaere et al., 2016). Thus, the investigation of phenomena/processes, to date not investigated in detail in geometric prototypical experimental facility with prototypical material, should be addressed. Therefore, discrepancies in some of the core degradation phenomena can be still observed when comparing the results predicted by different SA simulation tools considering the different core degradation models implemented in the codes (Mascari et al., 2019; Humphries, 2018).

Traditional safety analysis methods for evaluation of plant performance under normal and off-normal conditions, up to and including Design Basis Accidents (DBAs), are intended to demonstrate achievement of selected criteria. In the original licensing of many Light Water Reactor (LWR) power plants operating in the United States and internationally, conservatisms were incorporated in these analysis methods. These conservatisms ensured that, despite prevailing *uncertainties* in analytical characterization of plant performance under off-normal and accident conditions, sufficient safety margin would be incorporated into the design and robust plant performance could be achieved, even under the most severe conditions within the plant’s design envelope.

Over the past 20 years, the SA knowledge-base and integral plant response computer codes have achieved a significant level of maturity. While substantial uncertainties still remain in some areas, the understanding of and ability to quantitatively model the broad spectrum of SA phenomena has enabled the application of SA computer code analysis to resolution of a broad range of BDBA safety issues. This has been exemplified most recently by the application of codes such as MELCOR and MAAP to resolution of post-Fukushima safety issues. An important element of these issue resolutions was consideration of the impact of uncertainties on quantitative estimates (e.g., FOMs) utilized as part of the overall decision-making process. While still under active development, significant interest has developed in the area of characterizing uncertainties in SA analyses.

Therefore, the discussion and application of SA progression analyses with uncertainty estimation is currently a key topic in the Best Estimate Plus Uncertainty (BEPU) framework (D’Onorio et al., 2018; Mascari et al., 2019b). Today, in fact, for the evaluation of the safety margins, the use of BEPU approach by coupling selected calculated parameters with the related uncertainty range is of great interest for the international scientific community. Considering the key role of SA codes for deterministic safety analyses and source term evaluations, several research activities in national and international frameworks are in progress and are planned to reduce and/or estimate the uncertainty in SA phenomena prediction. In this framework two main activities are currently in progress or just concluded: the Management and Uncertainties Of Severe

¹ Uncertainty is used as a measure of the error made with the code in predicting the plant behavior. In general, the sources of uncertainty can be grouped in (IAEA, 2008): Code uncertainty (e.g. approximations in the conservation equation and in the closure models and correlations); Representation uncertainty (nodalization effect); Scaling issue (codes validated against scaled-down facilities); Plant uncertainty (e.g. initial and boundary conditions); User effect. - Code uncertainty (e.g. approximations in the conservation equation and in the closure models and correlations); - Representation uncertainty (nodalization effect); - Scaling issue (codes validated against scaled-down facilities); - Plant uncertainty (e.g. initial and boundary conditions).

Accidents (MUSA) project (Herranz et al., 2021), founded in Horizon 2020 EURATOM NFRP-2018-1- Safety assessments to improve accident management strategies for Generation II & III reactors, and the IAEA CRP on “Advancing the State-of-Practice in Uncertainty and Sensitivity Methodologies for Severe Accident Analysis in Water Cooled Reactors (I31033) (Mascari et al., 2022).

In this framework, as already underlined in (IAEA, 2019), from an operative point of view, one of the key needs is to have an automatic coupling between the Uncertainty Tool (UT) and the SA code. For example, by using the probabilistic method to propagate input uncertainty² (IAEA, 2008), the UT will perform several steps (e.g. creation of a certain number of code runs obtained combining the input uncertain parameters selected by the user through a random sampling, statistical and correlation analysis of the selected FOMs, etc.) and several parameters should be selected by the user (selection of PDF type and parameters, selection of the random sampling methods, selection of correlation coefficients for the analyses, etc.) for setting the UA. Therefore, the use of a user-friendly environment/architecture, which permits a direct coupling between the SA code and the UT to perform all the steps in the Uncertainty Analysis (UA), is one of key operative needs.

Within this regard, USNRC codes (e.g. MELCOR (Sandia National Laboratories, 2015), TRACE (U.S. Nuclear Regulatory Commission, 2013)) can be used together with a user-friendly front end, Symbolic Nuclear Analysis Package (SNAP) (Applied Programming Technology, Inc., 2021), able to support the user in the development and visualization of the nodalization, to show a direct visualization of selected calculated data, and to accept existing code input-decks. SNAP is a suite of integrated applications including the Model Editor, the Job Status, the Configuration Tool client applications and the Calculation Server. In particular, the Model Editor is used for the nodalization development and visualization and for the visualization of the selected calculated data by using its graphical and animation model capability (Mascari et al., 2012; Mascari et al., 2016).

One of the features of SNAP is the possibility of coupling USNRC codes with the Design Analysis Kit for Optimization and Terascale Applications (DAKOTA) toolkit (Applied Programming Technology, Inc., 2021; Applied Programming Technology, Inc., 2012). DAKOTA, in fact, is provided as a plug-in (Applied Programming Technology, Inc., 2012) in SNAP, and through SNAP it is possible to set up the DAKOTA UA (Applied Programming Technology, Inc., 2012; Bersano and Mascari, 2019; Mascari, 2019) and to perform automatically all the steps of the UA.

Considering that currently the international nuclear technical community is exploring the possibility of using SA codes in a BEPU framework, a report has been prepared to show the main details of the MELCOR/DAKOTA coupling in a SNAP environment/architecture, and the different steps necessary to set it up, including sample applications (Mascari et al., 2023).³ The aim of this paper is to provide an overview of the MELCOR/DAKOTA coupling in SNAP, summarizing the main steps. The study has been performed under the USNRC Cooperative Severe Accident Research Program (CSARP) framework (U.S. Nuclear Regulatory Commission, 2015); it also includes one output of the activities done in the framework of the Work Package 4 (Application of Uncertainty Quantification Methods against Integral Experiments – AUQMIE -) of the MUSA H2020 EURATOM project (Mascari et al., 2024). This paper could be generally beneficial to MELCOR users as well as current and future international initiatives.

The paper is organized as follows. Section 2 describes the DAKOTA

² Several methodologies have been developed in the past to perform UA, and they can be grouped in: - Methods to propagate input uncertainty: - Probabilistic (e.g. CSAU, GRS, IPSN, etc.); - Deterministic (e.g. AEAU, EDF-Framatome, etc.); - Method to extrapolate output uncertainty (e.g. UMAE) (IAEA, 2008).

³ MELCOR code is available through the USNRC Cooperative Severe Accident Research Program (CSARP).

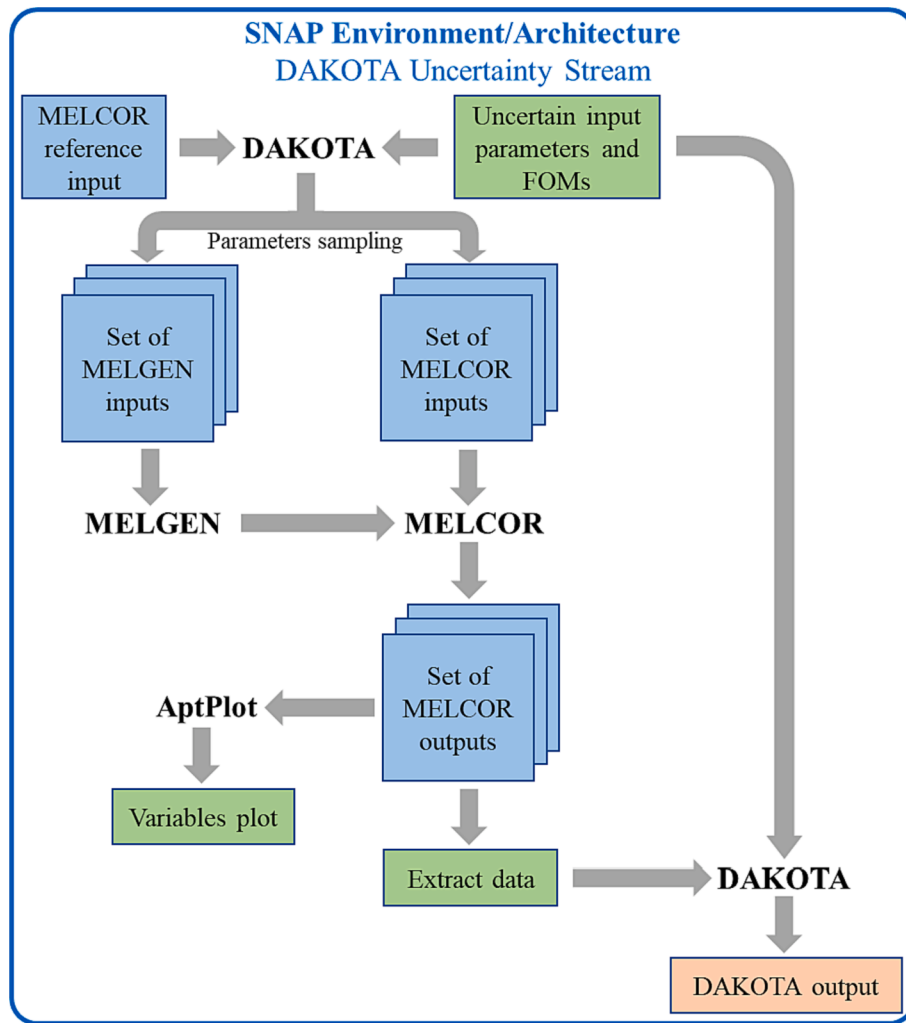


Fig. 1. DAKOTA UA workflow for MELCOR code in the SNAP environment/architecture.

UA workflow in the SNAP environment/architecture for MELCOR applications and the different steps necessary to create the MELCOR/DAKOTA analyses within the SNAP environment/architecture. Section 3 describes the report automatically generated by DAKOTA as result of the UA. Section 4 shows one exemplificative MELCOR/DAKOTA UA. Section 5 summarizes the conclusions. Finally in Appendix A the new Python directed job-stream capability in SNAP is presented.

2. Methods: MELCOR - DAKOTA coupling within snap

2.1. Propagation of input uncertainties

The probabilistic method to propagate input uncertainties (Glaeser, 2008) is a common approach to perform UA. It consists in the propagation of input parameter uncertainties through a computer code simulation. In general, the following steps are involved:

- Identification of the uncertain input parameters necessary to characterize the model used by the computer code for transient analysis;
- Definition of the variability of each input parameter by defining a Probability Density Function (PDF);
- Generation, for each input parameter, of N random realizations (i.e., input for code runs) by means of random sampling;
- Creation of code inputs for each realization of input parameters;
- Execution of all the N code inputs;

- Post-processing of code results generated from running the N input code realizations to:
 - Characterize the range of realized variation for selected FOMs;
 - Correlate realized variability in FOMs with the uncertain input parameters to assess the contribution of the different input uncertainties to code output uncertainty.

2.2. DAKOTA toolkit

DAKOTA (Sandia National Laboratories, 2017a; b; c) is an open source software written in C++ and developed by Sandia National Laboratories (SNL) to perform sensitivity analysis, optimization, parameter estimation, parametric and UA in a fast and automatic way.

The DAKOTA toolkit is also provided as a plug-in (Applied Programming Technology, Inc., 2012) for SNAP (Applied Programming Technology, Inc., 2021), which is a graphical user interface designed to support the use of USNRC computer codes (e.g., MELCOR, TRACE, etc.). Using SNAP, it is possible to build the input model in a graphical environment and to have a direct visualization of the computed data by using SNAP animation capabilities (Mascari et al., 2019; Mascari et al., 2012; Mascari et al., 2016; Mascari et al., 2011). Through SNAP, it is possible to set up a DAKOTA UA (Applied Programming Technology, Inc., 2012; Applied Programming Technology, Inc., 2021; Mascari et al., 2019) and to perform all analytical steps automatically.

Fig. 1 shows a schematic representation of a DAKOTA UA workflow within SNAP for MELCOR applications. As it can be seen, DAKOTA is

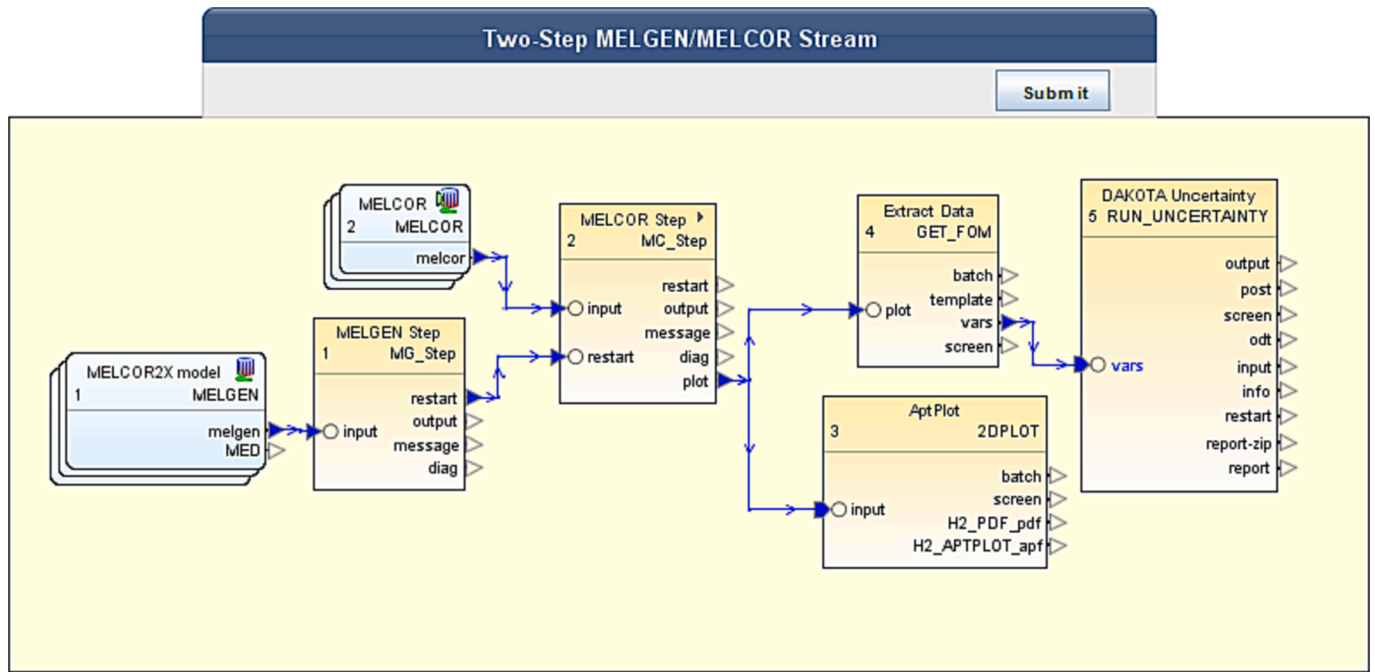


Fig. 2. MELCOR - DAKOTA Uncertainty Job Stream in SNAP.

first run to sample the uncertain input parameters values and to generate the set of code inputs. Then, after the solution of the set of code inputs and the extraction of the desired data, DAKOTA is run a second time to perform the actual UA and to characterize the relationship between input uncertain parameters and output parameters selected as FOMs.

The main steps necessary to define the UA in a MELCOR - DAKOTA application within SNAP are described in the following.

2.3. MAIN steps for an uncertainty analysis through a MELCOR - DAKOTA coupling within snap

To perform a complete UA through a MELCOR – DAKOTA coupling within SNAP, a *DAKOTA Uncertainty* job stream has to be set up and run. A comprehensive description of the process can be found in (Mascari et al., 2023). However, a concise exposition is reported hereafter.

A *DAKOTA Uncertainty* job stream requires the definition (and consequent configuration) of several steps:

- MELGEN step;
- MELCOR step;
- APTPLOT step;
- *Extract_Data* step;
- *DAKOTA (DAKOTA Uncertainty)* step.

First of all, the *DAKOTA Uncertainty* job stream has to be selected from the *Job Streams* list in the *SNAP Model Editor*. Secondly, within the *DAKOTA Uncertainty* job stream configuration, the *Three-Step MELGEN/MELCOR/APTPLOT Stream* has to be selected. In this way, a job stream with a basic configuration is automatically created without the need of setting up the three different steps (namely MELGEN, MELCOR and APTPLOT) separately.⁴ The three-step job stream is also visible in a dedicated view. At this point, with the definition of the steps related to the simulation execution (MELGEN and MELCOR) and to the results

plotting (APTPLOT), a step for the extraction of data needed for the subsequent UA is required. In this regard, the *Extract_Data* step has to be added to the *Stream Steps* related to the *DAKOTA Uncertainty* job stream under development. As last thing, the *DAKOTA (DAKOTA Uncertainty)* step has to be included in the *Stream Steps* list in order to obtain the UA outputs.

As it will be briefly discussed in the following, each step requires a thorough configuration, following which the *DAKOTA Uncertainty* job stream can be run. Once all the steps have been completed, the DAKOTA output file is created together with all plots requested in the AptPlot step.

Fig. 2 shows the MELCOR - DAKOTA Uncertainty Job Stream within SNAP.

2.3.1. DAKOTA uncertainty job stream configuration

From the *Properties* view of the *DAKOTA Uncertainty* job stream, various tabs of the *Parametric Properties* allow the user to characterize the Uncertainty Quantification (UQ) process by:

- Adding the FOMs selected for the analysis;
- Characterizing the sampling process defining the method (direct Monte Carlo sampling or Latin Hypercube stratified sampling) and the number of required samples;
- Defining the uncertain input parameters as well as their range and PDF;
- Setting the properties for the automatically generated final report containing the results of the UA.

2.3.1.1. Definition of DAKOTA Properties. By selecting the *Parametric Properties* button, in the first tab *Dakota Properties*, the user can specify one or more FOMs for the analysis. A *Time Dependent* option is also present and, as discussed in (Applied Programming Technology, Inc., 2012), it enables computation of response correlation coefficients at selected time instances of the transient.

In addition, the user is allowed to choose the *Sampling Method* (Monte-Carlo or Latin Hypercube methods available) and to impose the *Number of Samples*. As for the number of samples, a *Calculated Samples*

⁴ In older versions of the Model Editor, the *Three-Step MELGEN/MELCOR/APTPLOT stream* option is not available. In such a case, the *Two-Step MELGEN/MELCOR/APTPLOT stream* has to be selected and the APTPLOT step manually added and connected to the results of the MELCOR step.

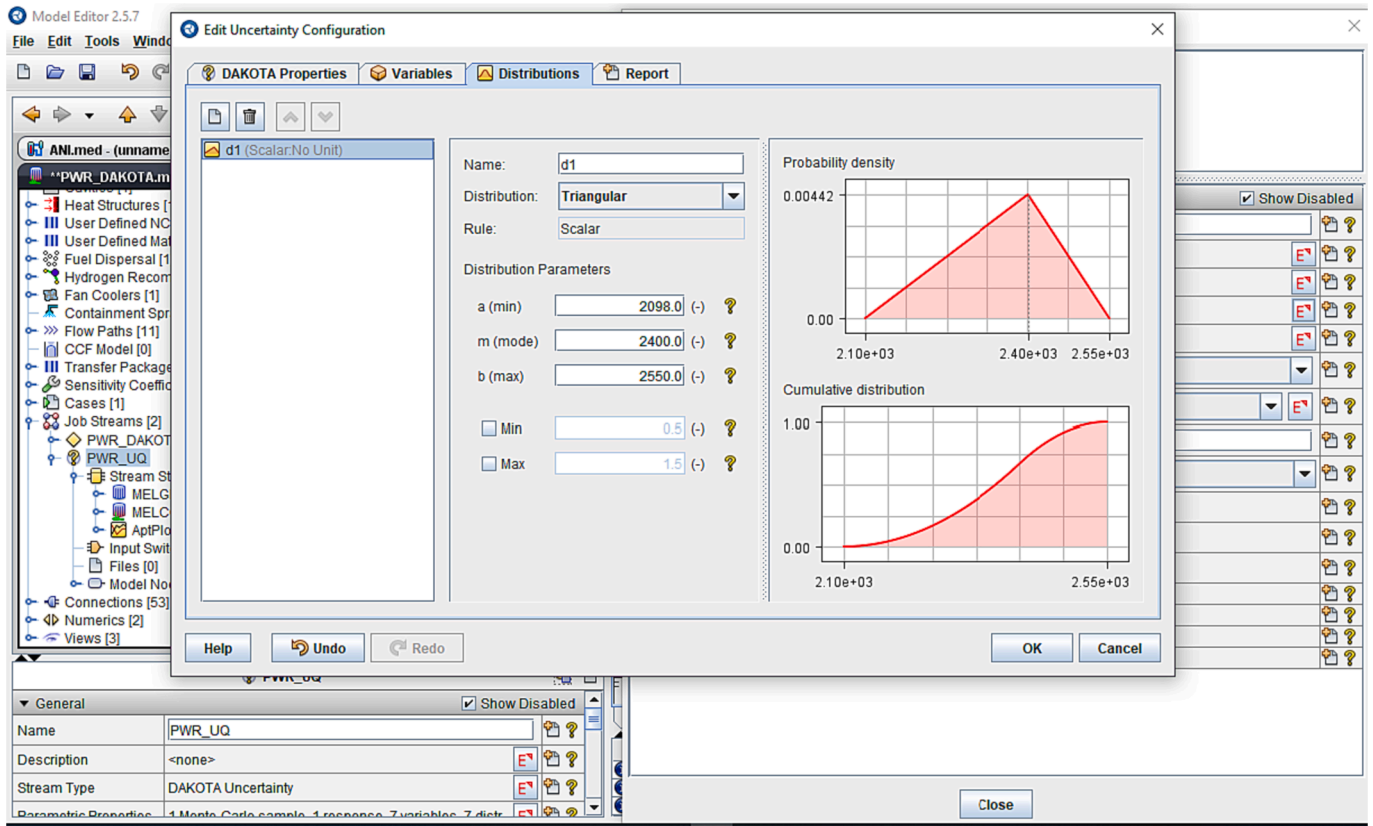


Fig. 3. Definition of the distribution properties of the uncertain input parameters.

button is available to evaluate the required number of samples on the bases of the Wilks method (Applied Programming Technology, Inc., 2012). In particular, the minimum number of samples (and, consequently, of code runs), N , is calculated taking into account the number of FOMs, plus three other parameters (*Order*, *Probability* – α , and *Confidence* – β) as defined by the user.

In the case in which only one FOM is investigated, for the one-sided tolerance interval, the required number of code runs can be found, based on the well-known Wilks formula (Wilks, 1941; Wilks, 1942), by solving the following equation with respect to N :

$$1 - \alpha^N = \beta \quad (1)$$

If more than one FOM is investigated, for the one-sided tolerance interval, the required number of code runs can be found by solving the following equation with respect to N (Bajorek and Gingrich, 2013):

$$\beta = \sum_{j=0}^{N-p} \frac{N!}{(N-j)!j!} \alpha^j (1 - \alpha)^{N-j} \quad (2)$$

where p is the number of FOMs.

More information on statistical aspects of traditional BEPU analysis can be found in (Guba et al., 2003).

2.3.1.2. Definition of uncertain parameters and their respective distributions. Still considering the *Parametric Properties* configuration, through the second and the third tabs, *Variables* and *Distributions* respectively, it is possible to:

- Add the selected uncertain input parameters;
- Define the characteristics of each input parameter distribution.

The input uncertain parameters can be selected among the ones

previously defined in the MELCOR input deck. They can be either a MELCOR sensitivity coefficient of a user-defined, real-valued variable. Regardless of the nature of the input parameter, a distribution has to be defined for each one of them. Depending on the selected distribution type (i.e., Normal, Lognormal, Uniform, Loguniform, Triangular, Exponential, Beta, Gamma, Gumbel, Frechet, Weibull, Histogram), a user may have to define also the associated distribution parameters (such as lower and upper bound).

Fig. 3 shows an example of distribution definition and characterization in the *Distributions* tab.

2.3.1.3. Specification for the automatic UA reporting. In the fourth tab, *Report*, it is possible to configure the final UA report, automatically generated by DAKOTA after the conclusion of job stream. Addition of desired plots to the report is also possible. Different options are present for *Plotted Values*:

- Selected input parameters as function of the iteration index;
- FOM as a function of a selected input parameter;
- FOM as function of the iteration index.

2.3.2. APTPLOT configuration

After the configuration of the *DAKOTA Uncertainty* job stream, the APTPLOT step can be set up. In particular, in the *Properties* view, by clicking on the *Plot* button, the user has the possibility to edit the properties of the plot, the graph, and the uncertain input parameters data sets. At the same time, it is possible to specify the plot variables, which could include FOMs as well as other parameters. It is worth underlining that, when first configuring this plotting step, only one set (corresponding to the input parameters related to only one code run) is displayed. In order to add all the sets, it is necessary to set the model nodes as parametric. To do so, in the *DAKOTA Uncertainty* job stream, under the *Model Nodes* view, the *Parametric* flag has to be set to True for

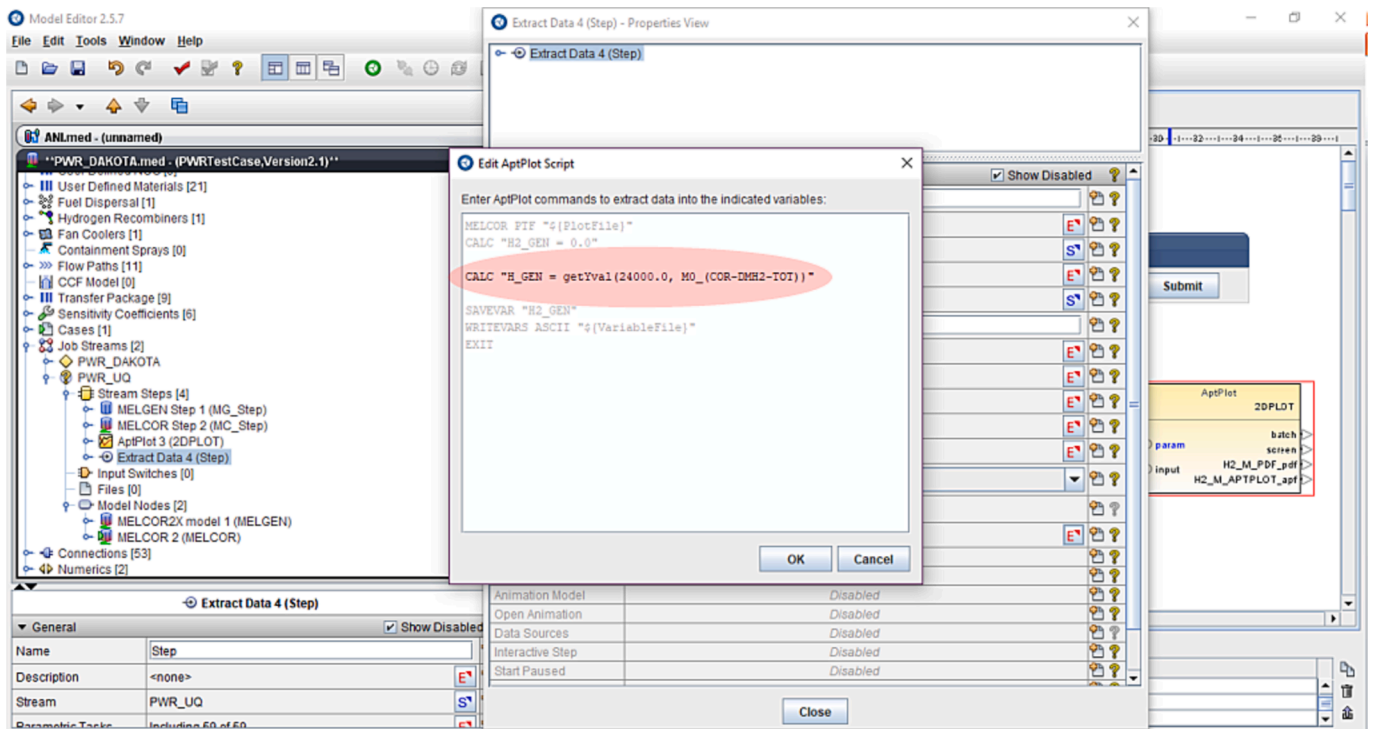


Fig. 4. The script for the extraction of the calculation results for the DAKOTA UA.

Table 1
Uncertainty Quantification Input Options - Model Variables and Distributions.

Distribution name	Distribution type	Application rule	Distribution parameters	Model variables	Nominal*
d1	Triangular	Scalar	a:2098.0, m:2400.0, b:2550.0	COR 1131-2	Replacement
d2	Triangular	Scalar	a:0.01, m:0.083, b:1.0	COR 1141-2	Replacement
d3	Triangular	Scalar	a:0.05, m:0.1, b:1.2	VFALL	Replacement
d4	Triangular	Scalar	a:50.0, m:1000.0, b:1100.0	HDBLH	Replacement
d5	Normal	Scalar	μ :2700.0, σ :120.0, $[-\infty, \infty]$	COR 1132-1	Replacement
d6	Normal	Scalar	μ :5.0, σ :1.0, $[-\infty, \infty]$	COR 1502-2	Replacement
d7	Normal	Scalar	μ :2800.0, σ :150.0, $[-\infty, \infty]$	COR 1250-1	Replacement

* Replacement indicates that the sampled value substitutes the nominal value of the input-deck.

both MELGEN and MELCOR.

2.3.3. Extract Data configuration

From the *Properties* view of the *Extract Data* step, the user is able to configure it by means of three main operations:

- Selection of the *Plot Type File*, namely MELCOR;
- Definition of the *InputFiles*. By opening the related selection window, it is possible to define the input *Source* by selecting the *MELCOR Step 2 (MC Step)* plot is selected;
- Setting of the variable to be extracted from the code runs' results for the subsequent uncertainty evaluation.

For the last operation, an AptPlot script is needed. The script can be entered by opening the *AptPlot Script* window under the *Properties* view of the *Extract Data* step. An AptPlot command should be entered by the user in the central part of the script window. Several extraction possibilities are available in AptPlot.

An example of *AptPlot Script* is presented in Fig. 4. In this case, the extracted FOM is "COR-DMH2-TOT", and it is extracted at a given time, i.e., 24000 s.

2.3.4. DAKOTA (DAKOTA Uncertainty) configuration

The final task to be completed for the UA setup is the *DAKOTA*

Uncertainty step configuration. After it is created as part of the *DAKOTA Uncertainty* job stream, the step should be connected to the output variables of the *Extract Data* step.

At this point, the UA setup is completed, and it is possible to run the Job Stream.

2.4. Replacement Samples option and Python Directed job stream

As from the previous discussion, the adoption of SNAP allows to simplify the MELCOR - DAKOTA coupling. However, a problem arises when one or more calculations fail during the MELGEN/MELCOR execution. Even the failure of only one run prevents the finalization of the UA (Mascari et al., 2021a; Mascari et al., 2021b). Two solutions can be used to overcome this issue:

- The *Replacement Samples* option can be selected to specify a given number of runs to be executed and replace the failed ones (Mascari et al., 2023). At the time of the development of the (Mascari et al., 2023), this option was not available when using the SNAP/GUI. Therefore, using the SNAP/GUI, if one calculation fails, it prevents UA finalization (Mascari et al., 2021a).
- A new job stream capability, called *Python Directed Stream* has been added to SNAP.

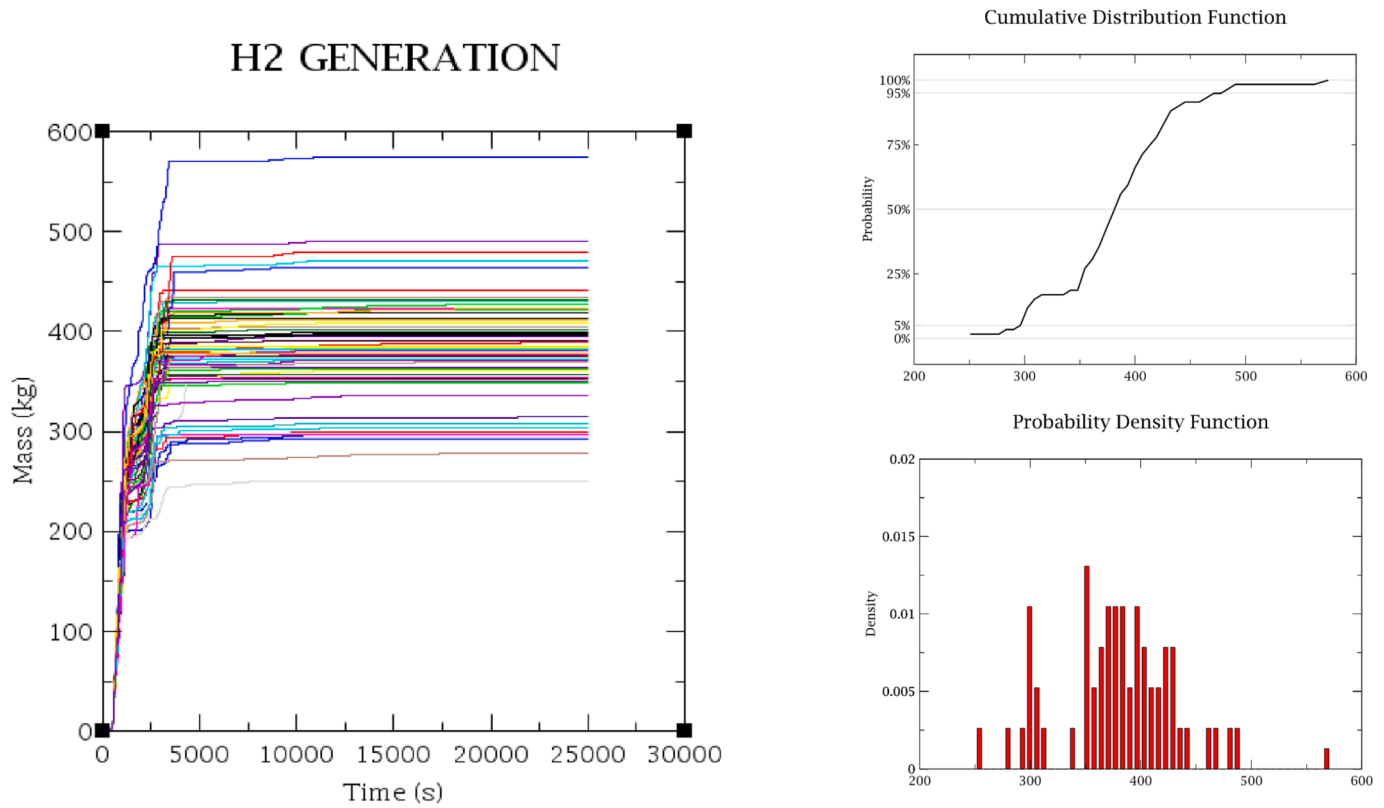


Fig. 5. Tot hydrogen mass generated dispersion, CDF, and PDF.

Table 2

Statistical results at the end of the calculation based on the 59 samples.

Summary	Value	Task #
Min Value	250.87505	54
Max Value	574.76666	51
Mean	381.43805	–
Median	382.44507	40
Standard Deviation	56.8461	–
Coefficient of Variance	0.38578	–

Table 3

Response correlations at the end of the calculation reported in the automatically generated DAKOTA report based on the 59 samples.

	Simple	Partial	Simple Rank	Partial Rank
d5	0.0979063	0.102384	0.135067	0.268539
d6	0.110172	0.146704	0.0338983	0.0599765
d7	0.098521	0.123725	0.126768	0.209831
d1	0.737284	0.791904	0.765926	0.855157
d2	−0.495872	−0.627241	−0.379486	−0.653184
d3	0.03035	0.00594472	0.0189947	0.0516358
d4	−0.0188159	0.0125986	−0.0151373	0.0108442

Considering the last approach, a MELCOR- DAKOTA coupling through *Python Directed Stream* in SNAP has been developed and tested, along the MUSA EURATOM project, by University of Pisa in collaboration with ENEA. Through this coupling approach, the management of failed code runs is possible with SNAP 3.1.6 (and newer). As implied by the name, the coupling in this case is managed through a Python script, which, in turn, is elaborated by SNAP.

The script, as it was developed in the MUSA project, allows to:

- Run the MELGEN/MELCOR runs with the different sets of input uncertain parameters created by the DAKOTA uncertainty plug-in;
- Calculate the FOM values for each run;
- Plot the dispersion of the FOMs through the module “PyPost”, developed by AptPlot;
- Generate the UQ final report.

The new *Python Directed Stream* capability in SNAP is presented in Appendix A, where a basic Python script is also reported.

3. DAKOTA UA report

As last step in the UA, DAKOTA automatically generates a report, which follows the structure presented hereafter:

- Table of contents;
- Introduction section;
- Uncertainty quantification input options section;
- Variate and response data section;
- DAKOTA Results;
- DAKOTA Input File.

As underlined before, the presence of eventual failed calculations may prevent the creation of the DAKOTA UA report. A solution to overcome this issue is the adoption of the *Python Directed Stream*, introduced in the previous section and better reported in Appendix A.

3.1. Uncertainty quantification input options section

The *uncertainty quantification input options* section includes:

- A tabulated summary of the main information characterizing the UA application;

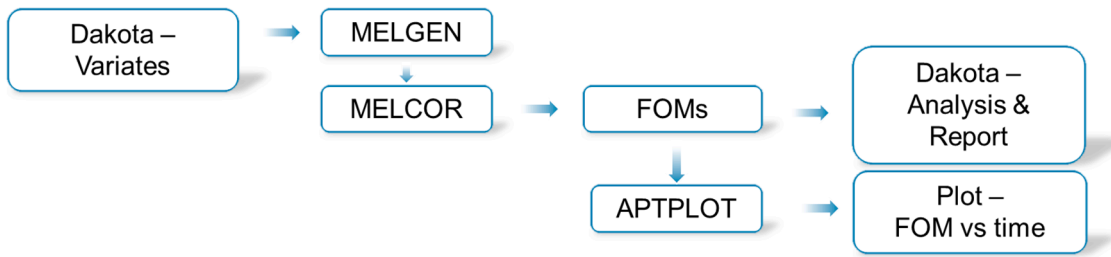


Figure A1. Python Directed Job Stream logic.

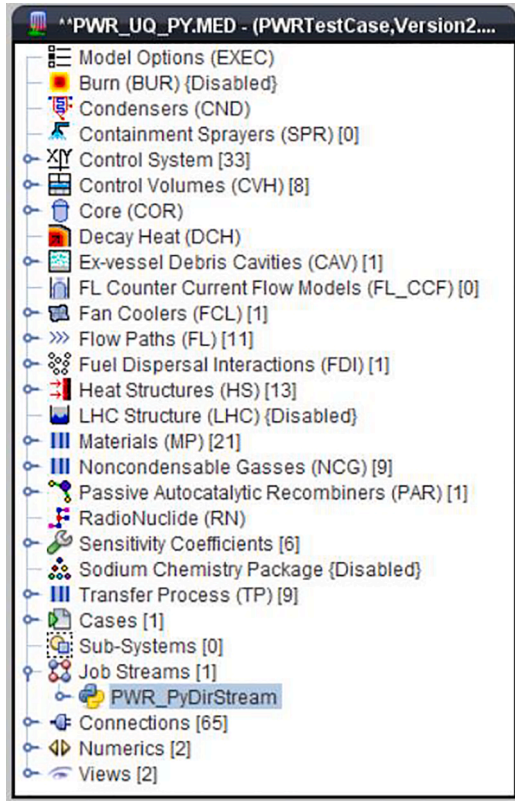


Figure A2. New Job Stream.

- A tabulated summary of distributions, their characteristic parameters, and the selected model variables;
- A table listing the application used in the job stream;
- The user-identified FOMs in the analysis.

3.2. Variate and response data section

In the *variate and response data* section, after a summary of the first DAKOTA run,⁵ the following information are present:

- Plot requested in the *Report* configuration;
- Variate data table;
- Response data table.

The plots present in the report automatically generated by DAKOTA are:

⁵ A first DAKOTA run is performed using the specified input uncertain parameters to generate a set of variates for each run. The individual runs are then performed, and the FOMs are extracted from the completed calculations.

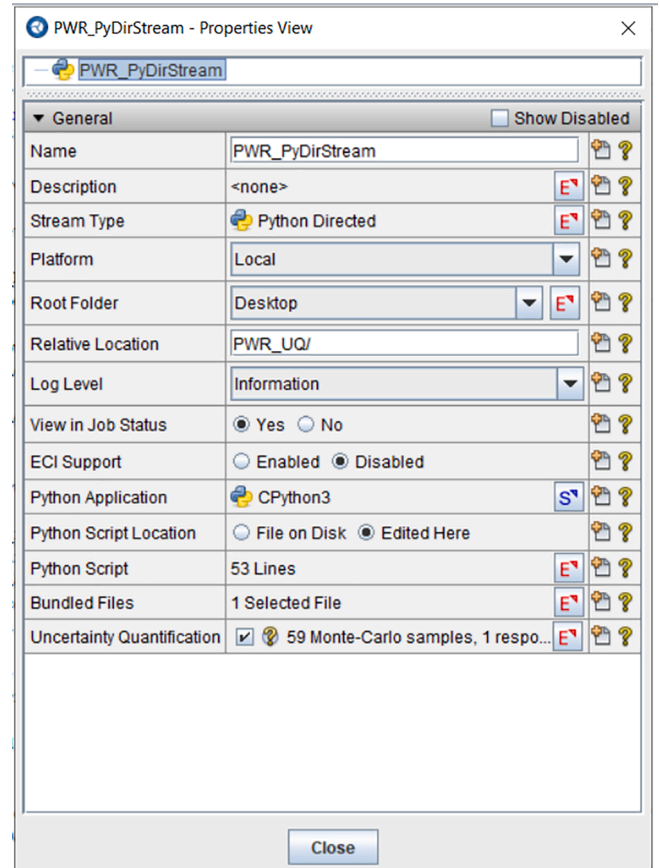


Figure A3. Job Stream Properties.

- Input uncertain parameter against iteration index;
- FOM vs input uncertain parameter;
- Input uncertain parameter vs another input uncertain parameter.

For each FOM:

- Response table reports response data for each run;
- Variate table presents the value of each uncertain parameter for each run.

3.3. DAKOTA results

In this section, the following information are reported for each FOM:

- Cumulative Distribution Function (CDF) and PDF plot;
- Statistical results;
- Response correlations.

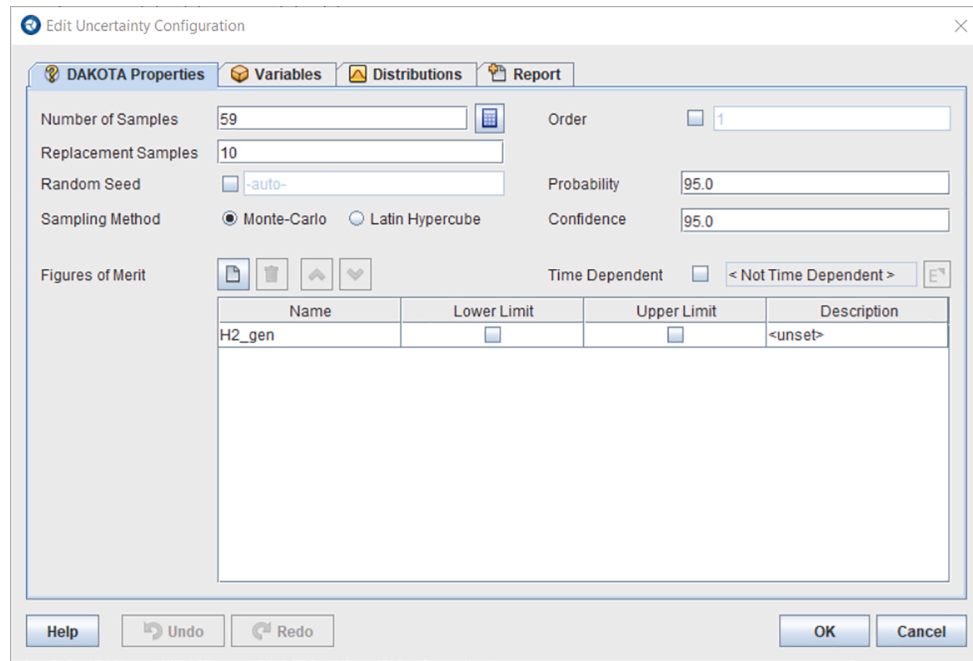


Figure A4. Replacement Samples.

As for the statistical results, the following data are reported in a tabular manner:

- Min value and the related run;
- Max value and the related run;
- Mean value;
- Median value;
- Standard deviation;
- Coefficient of variance.

As a result of the uncertainty analysis, currently DAKOTA computes four response correlation coefficients (Applied Programming Technology, Inc., 2012; Sandia National Laboratories, 2017a; Gamble and Swiler, 2020): simple, partial, simple rank and partial rank. The simple coefficient is related to the actual input and output data. The simple coefficient r between an input variable x and an output variable y , in n samples, is computed using the Pearson's correlation. It is a measure of the degree of linear correlation between the two variables, and its value is comprised between -1 and 1 :

- If $r < 0$, the correlation is negative (an increment of x leads to a reduction in y);
- If $r = 0$, there is no correlation between the two variables;
- if $r > 0$ the correlation is positive (an increment of x leads to an increment of y).

The partial correlation coefficient is computed similarly to the simple coefficient, but considering the effects of the other variables. This is useful, for example, if there is a strong correlation between two inputs. The correlation of the second input on the output may be adjusted after having considered the correlation between the first input and the output.

Rank correlation coefficients use the ranked data instead of the actual ones. Ranks are obtained by ordering the data in ascending order and are more convenient to be used when inputs and outputs are characterized by sensible difference in magnitude. It is possible to understand if the input sample with the lower rank is associated to the output with the lower rank and so on (Sandia National Laboratories, 2017a; Gamble and Swiler, 2020). To compute the rank correlation, DAKOTA uses the Spearman's rank correlation that is similar to Pearson's

correlation, but with the ranked data instead of the actual values. If two variables are monotonically related, without repetitions, the Spearman coefficient is -1 or $+1$ (depending upon whether the function is monotonically decreasing or increasing), since the ranked values are used. Moreover, Spearman's correlation is less sensitive to possible outlier values of the variables than Pearson's.

4. Practical example: Application for a PWR

In the following a MELCOR - DAKOTA uncertainty study is reported. It represents only a first exercise showing complete application of the coupling procedure of MELCOR and DAKOTA within the SNAP environment/architecture. The main purpose of this exercise is to show the feasibility of the MELCOR - DAKOTA coupling, to provide details on the capabilities associated to the followed methodology, and to show the great advantage to having a graphical user interface. Therefore, it should be underlined that the purpose of this work is not to represent a complete or representative analysis of the MELCOR code uncertainty. Also, the selected uncertain input parameters and related PDFs are only chosen for methodology demonstration purposes. In order to have indication about the range and PDF of selected uncertainty input parameters the SOARCA studies can be considered by the reader.⁶

4.1. MELCOR - DAKOTA coupling for a PWR

In the CSARP framework, a MELCOR - DAKOTA sample input model has been developed by ENEA using the `PWR_v2-0.inp` input available for MELCOR users.

For the presented analysis, the total cumulative hydrogen production in the core from all the oxidation processes (COR-DMH2-TOT) at the end

⁶ State-of-the-Art Reactor Consequence Analyses Project, Uncertainty Analysis of the Unmitigated Short-Term Station, Blackout of the Surry Power Station, Draft Report, <https://www.nrc.gov/docs/ML1522/ML15224A001.pdf>. State-of-the-Art Reactor Consequence Analyses Project Uncertainty Analysis of the Unmitigated Long-Term Station Blackout of the Peach Bottom Atomic Power Station, NUREG/CR-7155 SAND2012-10702P, Draft Report, <https://www.nrc.gov/docs/ML1318/ML13189A145.pdf>.

of the transient has been selected as the only FOM.

As for the uncertain input parameters, a short list of them has been chosen:

- Vfall: Velocity of falling debris,
- Hdblh: Heat transfer coefficient from debris to the lower head,
- SC1132(1): Core Component Failure Parameters – Temperature at which oxidized fuel rods can stand in the absence of unoxidized Zr in the cladding,
- SC1131(2): Zircaloy melt breakout temperature,
- SC1141(2): Core Melt Breakthrough Candling Parameters – Maximum melt flow rate per unit width after breakthrough,
- SC1502(2): Minimum Component Masses – Minimum total mass of component subject to the maximum temperature change criterion for timestep control, and
- SC1250(1): Conduction Enhancement for Molten Components – Temperature above which enhancement is employed.

Only to show the methodology, the type of PDF and the range value of the input uncertain parameter are based on a first analyses reported in (D'Onorio et al., 2018).⁷ Table 1 shows the model variables and distribution as summarized in the DAKOTA automatically generated report.

Fig. 5 shows the dispersion of the total hydrogen mass generated, CDF, and PDF. Table 2 shows the statistical results based on the 59 samples, and Table 3 shows the response correlation reported in the automatically generated DAKOTA report based on the 59 samples.

5. Conclusion

In the BEPU framework uncertainty quantification of SA codes results is being increasingly adopted. To conduct the UA using the probabilistic method to propagate input uncertainty, MELCOR code can be coupled with DAKOTA tool in the SNAP application.

The present paper summarizes a report (Mascari et al., 2023) where the main steps necessary to couple DAKOTA and MELCOR within SNAP are presented in detail: the creation of the Job Stream, the setup of the analysis through the different Uncertainty Job Stream Tab, the plot definition, the data extraction and the execution of the UA. Moreover, the structure and content of the report automatically generated by DAKOTA at the end of the UA is also presented. In addition, one explorative sample application is also presented to show the potential of the MELCOR/DAKOTA coupling and the possible information that can be derived from the UA.

Currently the MUSA project, founded in Horizon 2020 EURATOM NFRP-2018-1- Safety assessments to improve accident management strategies for Generation II & III reactors, just finished and the IAEA-CRP “Advancing the State-of-Practice in Uncertainty and Sensitivity Methodologies for Severe Accident Analysis in Water Cooled Reactors (I31033)” is in its final stage. In those frameworks representative uncertainty analyses have been performed for plant application or against experimental data. The guide described in this paper has been beneficial for these projects and can be useful for the general MELCOR users involved in other current and future activities.

CRediT authorship contribution statement

Fulvio Mascari: Writing – original draft, Visualization, Supervision, Resources, Methodology, Investigation, Formal analysis, Conceptualization. **Andrea Bersano:** Writing – original draft, Visualization, Methodology, Investigation, Formal analysis, Conceptualization. **Giuseppe Agnello:** Writing – review & editing, Investigation. **Michela Angelucci:**

Writing – review & editing, Software, Methodology, Investigation. **Jesse Phillips:** Writing – review & editing, Investigation. **David Luxat:** Writing – review & editing, Investigation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors are unable or have chosen not to specify which data has been used.

Acknowledgments

Special thanks to USNRC for their comments during the preparation of this document.

Part of the activity has been conducted in the framework of the MUSA project.

This project has received funding from the Euratom Research and Training Programme 2014-2018 under grant agreement No 847441.



This paper reflects only the author's view, and the European Commission is not responsible for any use that may be made of the information it contains.

ENEA carried out the research activity with MELCOR code and SNAP, obtained in the framework of the ENEA-ISIN agreement signed on March 23rd 2020 as part of the General Arrangement between the United States Nuclear Regulatory Commission (US-NRC) and the Italian National Inspectorate for Nuclear Safety and Radiation Protection (ISIN).

UNIPI carried out the research activity with MELCOR code and SNAP, obtained in the framework of the UNIPI-ISIN agreement signed on November 6th 2020 as part of the General Arrangement between the United States Nuclear Regulatory Commission (US-NRC) and the Italian National Inspectorate for Nuclear Safety and Radiation Protection (ISIN).

UNIPA carried out the research activity with MELCOR code and SNAP, obtained in the framework of the UNIPA-ISIN agreement signed on March 26th 2021 as part of the General Arrangement between the United States Nuclear Regulatory Commission (US-NRC) and the Italian National Inspectorate for Nuclear Safety and Radiation Protection (ISIN).

Appendix A. Python directed job stream

The set-up of the Python Directed Job Stream as an alternative way to perform an UA in SNAP is presented in this appendix. The methodology is based on the use of SNAP as single platform to carry out the entire analysis, and on a user-specified Python script to guide it. Contrary to the previously presented approach, replacement samples are exploited in the case of code failures, allowing to successfully undertake the uncertainty analysis, as will be better explained below.

It should be highlighted that the following is an output of the activities done in the framework of the EURATOM MUSA project (Herranz et al., 2021), coordinate by CIEMAT (Spain). In this regard, the Python script has been developed by University of Pisa, with a close interaction with ENEA, SNL and USNRC for the PHEBUS FPT1 (Jacquemain et al., 2000) application developed in the WP4 (Application of UQ Methods against Integral Experiments – AUQMIE -) (Mascari et al., 2024) of the MUSA project, coordinated by ENEA (Italy). In this report that script, have been updated applying to the sample 2.

⁷ The characterization of the COR 1131-2 parameter, not present in (D'Onorio et al., 2018), has been taken from (Sandia National Laboratories, 2016).

A.1. General approach

As the name suggests, the core of this approach is a user-specified Python script, which provides instructions to SNAP. The logic behind the script is shown in Fig. A1.

DAKOTA, through the Uncertainty plugin embedded in SNAP, is employed to perform the sampling on the base of the user-selected input parameters and their distributions. Once the variates are created, several input decks are set up. The Python Job Stream manager handle the MELGEN/MELCOR runs, checking for their status. Additional calculations are run when one or more calculations fail, making use of the replacement samples. Afterwards, the selected FOM values are extracted and used by DAKOTA for the UA and the relative report. Finally, an extra step can be added for AptPlot to generate the required plots.

A step-by-step tutorial will be proposed in the next section.

A.2 Step-by-step tutorial

To proceed with the set-up of the Python Directed Job Stream, it is essential to have installed:

- SNAP 3.1.6 (or later versions);
- Updated SNAP uncertainty plugin;
- PyPost (PyPost is a Python Postprocessor for accessing and extracting variables plot data from several engineering analysis codes, such as MELCOR, RELAP5, TRACE, etc.);
- MELGEN/MELCOR executables;
- Python 3 (3.6 or later) or Python 2.7 (optional).

The first step is the creation of a new Python Directed Job Stream (Fig. A2):

- In the SNAP Model Editor, right-click on Job Streams;
- Select New;
- From the Job Stream list, select Python Directed.

Once the Job Stream is created, the next step involves the configuration of the job properties (Fig. A3):

- Name: the name for the job stream is imposed here;
- Root folder/relative location: the location of the folder in which the stream will be executed is chosen;
- Python application: possibility is given to select the Jython interpreter included in SNAP or Python, if installed on the computer;
- Python script: it is the core of the approach. Instructions are given to SNAP through this user-specified Python script. A basic example will be presented in the next section, complemented by a thorough description of each action;
- Uncertainty Quantification: the uncertainty configuration follows the same steps proposed in section 3.1.2. The only exception is the possibility to impose a number of additional variates to be created (namely Replacement Samples) to compensate for possible code failures (Fig. A4).

Last step is the execution of the analysis:

- Right-click on the Python Directed Job Stream;
- Click on Check Stream;
- If no issues are found, right-click again on the Python Directed Stream;
- Click on Submit Stream to Local.

Once the steps have been completed, an “uncertainty report” folder is created, containing several files related to DAKOTA execution as well as the uncertainty report, as presented in Chapter 4.

A.2.1 Python script

A basic Python script will be illustrated hereafter.

Python module ‘parametric’ is needed to retrieve the uncertainty configuration:

```
import parametric
```

SNAP streams and model editor, MELGEN and MELCOR jobs, and PyPost, are essential for the process to work. Therefore, the related modules are imported:

```
from snap import streams
```

```
import snap.model_editor as model_editor
```

```
from snap.codes.melcor import MelgenActor, MelcorActor
```

```
from pypost.codes.melcor import MELCOR
```

Reference is made to the current running stream:

```
stream = streams.get_stream()
```

The uncertainty table is here defined:

```
uq_table = parametric.get_table()
```

The base-case, to which variates will be applied to, is opened:

```
pwr_med = stream.get_bundled_file('PWR_UQ_PY.MED')
```

```
pwr = model_editor.open_model(pwr_med)
```

A function is added to the script for the generation of MELGEN and MELCOR jobs. The MELGEN and MELCOR jobs are created for each row of the uncertainty table (so considering each single of variates). Afterwards, they are added to the stream and launched:

```
def submit_jobs(table_row):
```

```
    melgen_run = MelgenActor(row.new_task_name("Melgen_Job"),
    input = pwr).
```

```
    melcor_run = MelcorActor(row.new_task_name("Melcor_Job"),
```

```
    input = pwr.case('MELCOR'),
```

```
    restart_in = melgen_run.restart_out)
```

```
    stream.add([melgen_run, melcor_run])
```

A second function is added to the script to calculate the FOM (or FOMs) obtained in each run (i.e., for each UQ row). First, the plotfile associated with the MELCOR job is searched, with an error message shown in the case the plotfile does not exist. Afterwards, PyPost is employed to open the MELCOR plotfile and obtain the wanted data from it. And finally, the selected FOM value is stored in the UQ table:

```
def calculate_fom(table_row):
```

```
    plot_file = (row.search().label_eq("plot")
```

```
    .task_name_contains("Melcor_Job")
```

```
    .task_completed()
```

```
    .result())
```

```
    if plot_file is None:
```

```
        stream.logger.error("Row {} failed or did not produce a plot file."
```

```
        .format(table_row.row_index))
```

```
        table_row.failed = True
```

```
        return
```

```
    file_index = MELCOR.openPlotFile(plot_file.location)
```

```
    H2_gen = MELCOR.getData(file_index, 'COR-DMH2-TOT').maxYval
```

```
)
```

```
MELCOR.closeAll()
```

```
print(H2_gen)
```

```
row.set_fom_value('H2_gen', H2_gen)
```

A while-loop is employed to make use of the replacement samples in the case of runs' failures. New input decks are set up on the basis of the available additional variates, the new cases are launched and, once they all finished, the FOM is extracted and stored. Previously defined Python functions are employed:

```
while uq_table.check_available():
```

```
    for row in uq_table.available():
```

```
        row.apply_values(pwr)
```

```
        submit_jobs(row)
```

```
    stream.wait()
```

```
    for row in uq_table.applied():
```

```
        calculate_fom(row)
```

Once the number of completed runs matches the number of runs defined during the uncertainty configuration, a job is created for the

generation of the UQ report:

```
parametric.get_table().generate_report()
```

As aforementioned, the script here presented is the most basic one. Additional features can be added to extract different type of data, as briefly explained in the PyPost documentation ([Applied Programming Technology, Inc., 2019](#)). Moreover, instructions could be added in the script in order to obtain selected plots by means of AptPlot. The complete Python script is reported in ([Mascari et al., 2023](#)).

References

- Applied Programming Technology, Inc. Uncertainty analysis User manual, Symbolic Nuclear Analyses Package (SNAP), (2012).
- Applied Programming Technology, Inc. PyPost: A Python Postprocessor for Analysis of Code Results and Experimental Data (2019).
- Applied Programming Technology, Inc., Symbolic Nuclear Analysis Package (SNAP) Users Manual, 2021.
- S.M. Bajorek, C. Gingrich, Uncertainty Methods Framework Development for the TRACE Thermal-Hydraulics Code by the U.S.NRC, OECD/CSNI Workshop on Best Estimate Methods and Uncertainty Evaluations, NEA/CSNI/R(2013)8/PART2 (2013).
- Bersano, A., Mascari, F., 2019. Evaluation of a double-ended guillotine LBLOCA transient in a generic three-loops PWR-900 with TRACE code coupled with DAKOTA uncertainty analysis. *ATW Int. J. Nucl. Power* 64 (11/12), 526–532.
- Chatelard, P., Belon, S., Bosland, L., Carénini, L., Coindreau, O., Cousin, F., Marchetto, C., Nowack, H., Piar, L., Chailan, L., 2016. Main modelling features of ASTEC V2.1 major version. *Ann. Nucl. Energy* 93, 83–93.
- D'Onorio, M., Giannetti, F., Mascari, F., Caruso, G., 2018. "Uncertainty Analyses Using The RAVEN Software Tool Coupled With MELCOR Severe Accident Code", Proceedings of ANS Best Estimate Plus Uncertainty International Conference (BEPU 2018) Real Collegio, Lucca, Italy, May 13–19, BEPU2018-282.
- de la Rosa Blul, J.C., Brumm, S., Mascari, F., Lee, S.J., Carenini, L., 2018. ASTEC-MAAP comparison of a 2 inch cold leg LOCA until RPV failure. *Hindawi Sci. Technol. Nucl. Installat.* 9189010.
- Fauske & Associates, LLC "Transmittal Document for MAAP5 Code Revision MAAP5.02", FAI/13-0801, 2013.
- Gamble, K.A., Swiler, L.P. 2020. Uncertainty Quantification and Sensitivity Analysis Applications to Fuel Performance Modeling, SAND2016-4597C.
- Glaeser, H., 2008. GRS method for uncertainty and sensitivity evaluation of code results and applications. *Sci. Technol. Nucl. Installat.*
- Guba, A., Makai, M., Pál, L., 2003. Statistical aspects of best estimate method-I. *Reliab. Eng. Syst. Safety* 80, 217–232.
- Herranz, L.E., Beck, S., Sanchez-Espinoza, V.H., Mascari, F., Brumm, S., Coindreau, O., Paci, S., 2021. The EC MUSA project on management and uncertainty of severe accidents: Main pillars and status. *Energies* 14.
- L. Humphries, MELCOR Code Development Status EMUG 2018, 10th Meeting of the European MELCOR User Group, University of Zagreb, Zagreb, Croatia, Faculty of Electrical Engineering and Computing – FER, April 25–27, 2018.
- IAEA, Best Estimate Safety Analysis for Nuclear Power Plants: Uncertainty Evaluation, Safety Reports Series (2008).
- IAEA, Status and Evaluation of Severe Accident Simulation Codes for Water Cooled Reactors, IAEA-TECDOC-1872, IAEA, 2019.
- Jacquemain, D., Bourdon, S., de Braemaeker, A., Barrachin, M., 2000. "FPT1 Final Report (Final version)", IPSN/DRS/SEA/PEPF report SEA1/00, IP/00/479 (2000).
- Mascari, F., Bersano, A., Massone, M., Agnello, G., Coindreau, O., Beck, S., Tiborcz, L., Paci, S., Angelucci, M., Herranz, L.E., Bocanegra, R., Pontillon, Y., Berdai, M., Cherednichenko, O., Iskra, A., Nudi, M., Groudev, P., Petrova, P., Kretzschmar, F., Gabrielli, F., Kanglong, Z., Vileiniskis, V., Kaliatka, T., Kalilainen, J., Malicki, M., Gumenyuk, D., Vorobyov, Y., Kotsuba, O., Dejardin, P., Di Giulio, M., Thomas, R., Ivanov, I.V., Giannetti, F., D'Onorio, M., Caruso, G., Salay, M., Sevón, T., 2024. Main outcomes of the Phebus FPT1 uncertainty and sensitivity analysis in the EU-MUSA project. *Ann. Nucl. Energy* 196, 110205.
- Mascari, F., Vella, G., Woods, B.G., Welter, K., D'Auria, F., 2012. Analysis of Primary/Containment Coupling Phenomena Characterizing the MASLWR Design During a SBLOCA Scenario, Nuclear Power Plants, Intech.
- Mascari, F., Nakamura, H., Umminger, K., De Rosa, F., D'Auria, F., 2015. Scaling Issues For The Experimental Characterization Of Reactor Coolant System In Integral Test Facilities And Role Of System Code As Extrapolation Tool, Proceedings Of International Topical Meeting on Nuclear Reactor Thermal Hydraulics 2015, NURETH 2015, Volume 6, pp. 4921–4934 (2015).
- Mascari, F., De Rosa, F., Woods, B.G., Welter, K., Vella, G., D'Auria, F., 2016. Analysis of the OSU-MASLWR 001 and 002 Tests by Using the TRACE Code, NUREG/IA-0466, U.S. Nuclear Regulatory Commission.
- F. Mascari, ENEA activities in the field of severe accidents by using MELCOR code, 11th Meeting of the European MELCOR and MACCS User Group, Paul Scherrer Institute, Villigen, Switzerland, April 3–5, 2019.
- Mascari, F., De La Rosa Blul, J.C., Sangiorgi, M., Bandini, G., 2019. Analyses of an Unmitigated Station Blackout Transient in a Generic PWR-900 with ASTEC, MAAP and MELCOR Codes, NUREG/IA-0515.
- Mascari, F., Rocchi, F., Carny, P., Liptak, L., Adorni, M., Fontanet, J., Herranz, L.E., Shawkat, M., Raskob, W., Cousin, F., de la Rosa Blul, J.C., Urbonavicius, E., Di Dedda, F., Augustsson, T., Constantin, M., Arbaev, G., Isaksson, P., Kubicek, J., 2019. FASTNET SCENARIOS DATABASE DEVELOPMENT AND STATUS, The 9TH European Review Meeting on Severe Accident Research (ERMSAR2019), Prague, Czech Republic, March 18–20, 2019b.
- Mascari, F., Bersano, A., Agnello, G., Angelucci, M., 2021a, MELCOR/DAKOTA coupling in the SNAP environment/architecture, 12th Meeting of the European MELCOR and MACCS User Group, Organized by NUBIKI-Nuclear Safety Research Institute, Hungary, virtual meeting, 12–16 April 2021.
- Mascari, F., Bersano, A., Agnello, G., Angelucci, M., 2021b, MELCOR/DAKOTA coupling in the SNAP environment/architecture, CSARP meeting 2021.
- Mascari, F., Bersano, A., Gabrielli, F., Luxat, D., Albright, L., Jevremovic, T., Ur Rehman, H., Ortiz-Villafuerte, J., Ahn, K.L., Ryzhov, N., Garcia, J.M., Petoukhov, S.M., 2022, Overview of IAEA CRP I31033 "Advancing the state-of-practice in uncertainty and sensitivity methodologies for severe accident analysis in water cooled reactors", The 10th European Review Meeting on Severe Accident Research (ERMSAR2022), Akademiehôtel, Karlsruhe, Germany, May 16–19, 2022, Log Number: 314.
- Mascari, F., Bersano, A., Agnello, G., Angelucci, M., Phillips, J., Luxat, D., 2023. MELCOR – DAKOTA Coupling for Uncertainty Analyses, in a SNAP Environment/Architecture, NUREG/IA-0532, 2023.
- Mascari, F., Vella, G., Woods, B.G., Welter, K., Pottorf, J., Young, E., Adorni, M., D'Auria, F., 2011. Sensitivity analysis of the MASLWR helical coil steam generator using TRACE. *Nucl. Eng. Des.* 241, 1137–1144.
- Mascari, F., 2019. ENEA activities in the field of severe accidents by using MELCOR code, 11th Meeting of the European MELCOR and MACCS User Group, Paul Scherrer Institute, Villigen, Switzerland, April 3–5, 2019.
- OECD/NEA/CSNI, 1992, Source term uncertainty, recent development in understanding fission product behaviour, NEA/CSNI/R(92)2.
- OECD/NEA/CSNI, 1999, SOAR on Containment Thermalhydraulics and Hydrogen Distribution, NEA/CSNI/R(1999)16.
- OECD/NEA/CSNI, 2001, In-Vessel Core Degradation Code Validation Matrix, Update 1996–1999, NEA/CSNI/R(2000)21, France (2001).
- OECD/NEA/CSNI, 2009, State-Of-The-Art Report On Nuclear Aerosols, NEA/CSNI/R(2009)5.
- OECD/NEA/CSNI, 2014, Containment code validation matrix, NEA/CSNI/R(2014)3.
- Sandia National Laboratories, MELCOR Computer Code Manuals, Vol.1: Primer and Users' Guide, SAND2015-6691 R; Vol. 2: Reference Manual, SAND 2015-6692 R; Vol. 3: MELCOR Assessment Problems, SAND 2015-6693 R, 2015.
- Sandia National Laboratories, State-of-the-Art Reactor Consequence Analyses Project Uncertainty Analysis of the Unmitigated Long-Term Station Blackout of the Peach Bottom Atomic Power Station, NUREG/CR-7155 SAND2012-10702P, 2016.
- Sandia National Laboratories, Dakota, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 6.6 User's Manual, SAND2014-4633, 2017a.
- Sandia National Laboratories, Dakota, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 6.6 Reference Manual, SAND2014-5015, 2017b.
- Sandia National Laboratories, Dakota, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 6.6 Theory Manual, SAND2014-4253, 2017c.
- U.S. Nuclear Regulatory Commission, TRACE v5.840 Theory Manual (2013).
- U.S. Nuclear Regulatory Commission, Cooperative Severe Accident Research Program (CSARP), NUREG/BR-0524, 2015.
- Van Dorsselaere, J.P., Mustoe, J., Power, S., Adorni, M., Schaffrath, A., Nieminen, A., 2016. ETSON views on R&D priorities for implementation of the 2014 Euratom directive on safety of nuclear installations. *Kerntechnik* 81 (5), 527–534.
- Van Dorsselaere, J.P., Barrachin, M., Power, S., Adorni, M., Hrehor, M., Mascari, F., Schaffrath, A., Tiselj, I., Uspuras, E., Yamamoto, Y., Gumenyuk, D., Fedotova, N., Cronvall, O., Liska, P., 2018. The summary of ETSON strategic orientations on research activities, ETSON research group activity. *ATW Int. J. Nucl. Power* 63 (1 January).
- Wilks, S.S., 1941. Determination of sample sizes for setting tolerance limits. *Ann. Math. Stat.* 12 (1), 91–96.
- Wilks, S.S., 1942. Statistical prediction with special reference to the problem of tolerance limits. *Ann. Math. Stat.* 13 (4), 400–409.