Integrated Uncertainty Analysis (IUA) package in

RELAP/SCDAPSIM/MOD4.x and /MOD3.x code versions



Prepared by Marina Perez-Ferragut

Rev0 : May 2015

Rev1 : December 2015

Rev2 : March 2017

Table of Contents

[RELAP/SCDAPSIM Integrated Uncertainty Analysis Package 3](#_Toc439266430)

[Setup phase 5](#_Toc439266431)

[Sampling 7](#_Toc439266432)

[Polynomial (or trapezoidal) distribution sampled with the inverse method 7](#_Toc439266433)

[Normal-based distributions sampled with the Box-Mueller method 8](#_Toc439266434)

[Simulation phase 9](#_Toc439266435)

[Post-processing phase 9](#_Toc439266436)

[Cards 291DDWWW – weight information 11](#_Toc439266437)

[Examples 12](#_Toc439266438)

[Example 1 – General case I 12](#_Toc439266439)

[Example 2 – General case II 13](#_Toc439266440)

[Example 3 – Normalization of a range of uncertainty values 14](#_Toc439266441)

[Example 4 – Limitation of the uncertainty value to a maximum value of 1. 15](#_Toc439266442)

# RELAP/SCDAPSIM Integrated Uncertainty Analysis Package

The Integrated Uncertainty Analysis package is available in RELAP/SCDAPSIM code versions MOD4.x and MOD3.x

The uncertainty evaluation capability is implemented as an alternative run mode, the "uncertainty" mode, which allows the automatic execution of an uncertainty analysis based on the probabilistic approach. A complete uncertainty analysis using RELAP/SCDAPSIM/MOD4.0 code requires the execution of three related phases, namely the "setup" phase, the "simulation" phase consisting of several executions, and the "post-processing" phase. Because of the large number of files involved in the uncertainty analysis, specific suffixes on file names are required.

The "setup" phase generates the total number of sampled values, also called "weights", and information needed to build the tolerance bounds during the "post-processing" phase. The weights are used to associate uncertainty to code parameters by applying them as multipliers to the base values. During this phase the code also computes the required number of code runs by using the Wilks' formula, or simply uses the value supplied by the user. The setup information is written on disk files, one for each simulation run, and another one for the post processing. The command line for executing the "setup" phase is

*relap5o.exe -i lbloca.is -uncrun setup (MOD4.x)*

*relap5o.exe -i lbloca.is -U setup (MOD3.x)*

The "uncrun" (MOD4.x code versions) or “U” (MOD3.x code versions) field activates the uncertainty run mode and the "setup" specifies the first phase of the package. The input file name, "lbloca" in the example above, has to be the same name in all phases, and is also the given name for the output files (output print, restart plot, and weight files), the only difference being the file extension. The command line will specify the input file name but the other file names will be generated internally and are not entered on the command line. The required data in the setup input file is the information to compute the number of needed code runs, the uncertainty information to perform the sampling, and the base case input deck for checking process. The suffix "is" is the required suffix for the "setup" input file. The output file will be simul.os (-o), the restart plot file will be simul.rs (-r), and the weight files will be simulNNNN.w (-s). The number of weight files Nw, will be equal the number of uncertainty runs entered or determined during this phase and NNNN, written as a four-digit number with leading zeros, will range from 0001 through Nw.

The "simulation" phase consists of the base case run in which the simulation is done as if there were no uncertainty option available, and the set of uncertainty runs which have input and source modifications. To eliminate modification of the input file used during the simulation runs command line option is used to indicate the uncertainty phase and the run number during the simulation phase runs. Except for the base case run, each run of the simulation phase reads its corresponding weight file generated by the "setup" phase for that run. All simulation runs write information to be used in the "postprocessing" phase into the plot records of the restart-plot file. The input data used for the simulation phase should be the same for the base run and each simulation run and is also the same used during the setup run. The command line for executing the "simulation" phase is

*relap5o.exe -i lbloca.i -uncrun n (MOD4.x)*

*relap5o.exe -i lbloca.i -U n (MOD3.X)*

The suffix "i" is the required suffix for the input file. The number n is 0 for the base case and is the run number for the uncertainty runs, and this number written as four digits with leading zeros will be appended to the name of the output and restart plot files, lblocaNNNN.o lblocaNNNN.r. The weight file to be used in the uncertainty runs will be lblocaNNNN.w. To allow the simulation runs to be restarted from a restart-plot file containing stead-state results, the -r option can be entered and its file name and suffix could be lbloca.r but any other name and suffix could be used. The code does not modify the indicated restart-plot file but only copies it to lblocaNNNN.r. The command line for executing a restart run from a unique restart-plot file is:

*relap5o.exe -i lbloca.i -r stst.r -uncrun n (MOD4.x)*

*relap5o.exe -i lbloca.i -r stst.r -U n (MOD3.x)*

From the above command, the code will generate lblocaNNNN.o and lblocaNNNN.r

The "post-processing" phase reads the restart-plot files written during the base case and the uncertainty runs and generates the rank matrices for the output quantities defined in the "post-processing" input file. The rank matrices contain the values for the output parameters sorted according to its rank and are used to determine the tolerance intervals. The information required in the "post-processing" input file also includes the simulation runs to be used in the generation of the tolerance intervals. The command line for executing the post processing phase is

*relap5o.exe -i lbloca.ip -uncrun postpr (MOD4.x)*

*relap5o.exe -i lbloca.ip -U postpr (MOD3.x)*

The "postpr" field specifies the last phase of the uncertainty package. The "postprocessing" phase will generate rank- and run- based matrices files for each requested output parameter. Each rank-based matrix is written to disk with the file name lblocaALFNUM.m and each run-based matris is written to disk with the file name lblocaALFNUM.mruns, where ALF is the variable code (alphanumeric) and NUM is the parameter (numeric) of requested quantities. A graph containing the time history of the base result, the upper and lower bounds and the span between them will also be generated for each requested parameter. As for the other phases, the output file name is defined from the input file name and thus, for the example will be lbloca.op. Figure 1sketches the input and output files required and generated during a full application of the uncertainty package.



Figure 1 File chart

# Setup phase

The "setup" phase generates the required number of weight files containing the multipliers used for the uncertainty association (up to now, uncertainty is associated only by means of multipliers). The execution of the setup phase requires an input deck file containing information related to:

* Number of uncertainty code runs needed.
* Uncertainty data for "input treatable" parameters.
* Uncertainty data for "source correlation" coefficients.
* Base input deck for checking.

The code uses the Wilks' formula to compute the required number of uncertainty runs given the:

* Percentile (<1) or coverage of the tolerance limit.
* Confidence level (<1).
* Order for Wilks' formula application (1, 2 . . . ).

The number of code runs can either be computed by the code using the above information or be fixed by the user.

The percentile and confidence level are the characteristic parameters of the uncertainty bands to be obtained: the percentile is the amount of the population contained below the uncertainty bound with a certain confidence level. The user should specify the percentile and the confidence level for an upper bound and a lower bound will be also obtained for the corresponding (1-percentile) and the same confidence level. The two bounds, obtained at the end of the process after going through the three phases, are called Unilateral Tolerance Levels and have to be regarded separately.

The form of the Wilks' formula is the incomplete beta function. The code computes the number of needed runs by an iterative process that tests that the confidence level supplied in the "setup" input deck is greater or equal to the confidence level computed by the code using the incomplete beta function. The order of the Wilks' formula application increases the number of required code runs but will also obtain a more accurate estimation of the uncertainty bounds.

An additional feature is the implementation of extra fields to input start and end code run numbers, which allows the continuation of previous work or the correctionof any failure without repeating the whole process. The package also includes the possibility of adding extra runs (which might be useful to account for possible code failures), setting a maximum or minimum number of runs (when the user is not sure of the required number of runs computed from Wilks' formula) and the introduction of the "seed" number to start the random generating process. The next seed to be used is also written in the output print file generated in this phase and might be useful to continue previous work, i.e. increasing the number of runs. These extra fields allow the user to adapt to the specific methodology.

The available code correlations for uncertainty association are:

* Interfacial heat transfer coefficients.
* Heat transfer coefficients.
* Critical Heat Flux.
* Gap thermal conductivity from the gap conductance model. In this correlation the user may apply different multipliers to different ranges of temperature, or a single multiplier for the whole temperature range.
* Viscosity.
* Thermal conductivity.
* Surface tension.

The structure of the coding is prepared to easily prepare other correlations for uncertainty association.

The uncertainty data can be supplied for the two types of parameters, the "input treatable" and the "source correlation" quantities. The required information is the probability distribution function and its characteristic parameters.

Four distribution types are available:

* Uniform distribution: the required parameters are the minimum and the maximum values.
* Normal distribution: the required parameters are the mean and the standard deviation.
* Log-normal distribution: the required parameters are the mean and the standard deviation.
* Polynomial (or trapezoidal) distribution: the required parameters are P1, P2, P3 and P4 defining the minimum and the maximum values with zero and maximum probability. This distribution may be used when the (uniformly distributed) uncertainty available seems to be rather narrow and the range is enlarged but without keeping the restrictive uniform shape. An example of a polynomial distribution with characteristic parameters P1=0.4, P2=0.79, P3=0.99 and P4=1.12 is depicted in Figure 2.



Figure 2 Example of trapezoidal (polynomial) distribution

The introduction of a discrete distribution is within the future plans. A discrete distribution would allow, for different uncertain runs, switching between different correlations available to simulate a same phenomenon.

The user can choose whether the weight is computed on an entered distribution data basis or, instead, specify a value (a bias) which will be used as the uncertainty multiplier in all uncertainty runs during the "simulation" phase.

For the "input treatable" parameters type, a second set of information is required to specify the input data card and the word number within that data card to be modified by the uncertainty multiplier. These fields in the setup input file allow the following options:

* Different or equal weights can be used for quantities with the same distribution function, that is, a sampling may be performed for each parameter or the first generated value is used for all specified parameters;
* Use entered bias instead of the computed weight;
* Normalization to the base case value: when this ag is activated the code computes the sum of the base case values for the marked parameters and renormalizes the modified values, i.e. values obtained after multiplying the base case values by the corresponding weights, so that their sum is equal to the base case values' sum;
* Maximum of 1 for the modified value: by activating this ag the modified value is limited to 1. This feature might be useful when associating uncertainty to decay power tables built as multipliers to the nominal power.

The "setup" input file has to contain the base case input deck to check the consistency of the uncertainty data entered for the "input treatable" quantities with the base case input deck.

As a result of the setup phase three different types of files will be generated:

* Output print file: readable format file that contains Wilks' formula related data, the uncertainty data entered in the setup input file, and the list of the sampled (or specified) uncertainty multipliers for each uncertainty run calculation.
* Restart-plot file: binary format file that contains Wilks' related information needed in the "post-processing phase" to derive the uncertainty bounds.
* Weight files: binary format files that contain the multipliers to be used in the uncertainty runs. The number of weight files may be equal to the computed number by Wilks' formula, to the user-supplied value, or limited to a specified minimum and maximum run number.

## Sampling

The sampling technique implemented in the code is the Simple Random Sampling (SRS). The process generates sampled values from distribution functions defining the state of knowledge of each input parameter with uncertainty associated.

As mentioned above, four types of distribution functions are available for the uncertainty package.

The sampled values from the uniform function are directly obtained by using firstly the Fortran random generator and secondly scaling the generated [0 ; 1 ] value to the requested minimum and maximum values. The methods used to generate the sampled values for the polynomial and normal-based laws are detailed in next sections. To be more precise, the sampling from the polynomial distribution function uses the inverse method, and the normal based functions (normal and log-normal distributions) are sampled using the Box-Müeller method.

### Polynomial (or trapezoidal) distribution sampled with the inverse method

The inverse method can be used to sample values from a distribution p(x ) when the inverse of the cumulative function P(x ) of that distribution p(x ) can be solved analytically.

The cumulative function

is a monotone non-decreasing function that takes values in the interval . A uniformly distributed variable in the interval and the cumulative function are used to sample from the trapezoidal distribution .

The sampling process can be summarized into two steps:

* Generate a uniformly distributed value in the interval .
* Identify with the cumulative function to obtain the value of the distribution .

To apply the inverse method, the distribution and cumulative functions have to be analytically defined. The trapezoidal function is defined by four parameters , , and and 3 regions (see Figure 3):

* Region 1 is de\_ned by:
* Region 2 is de\_ned by:
* Region 3 is de\_ned by:

There are five unknowns a, b, c, d and N, and five equations:

* Left limit in Region 1:
* Right limit in Region 1:
* Left limit in Region 3:
* Right limit in Region 3:

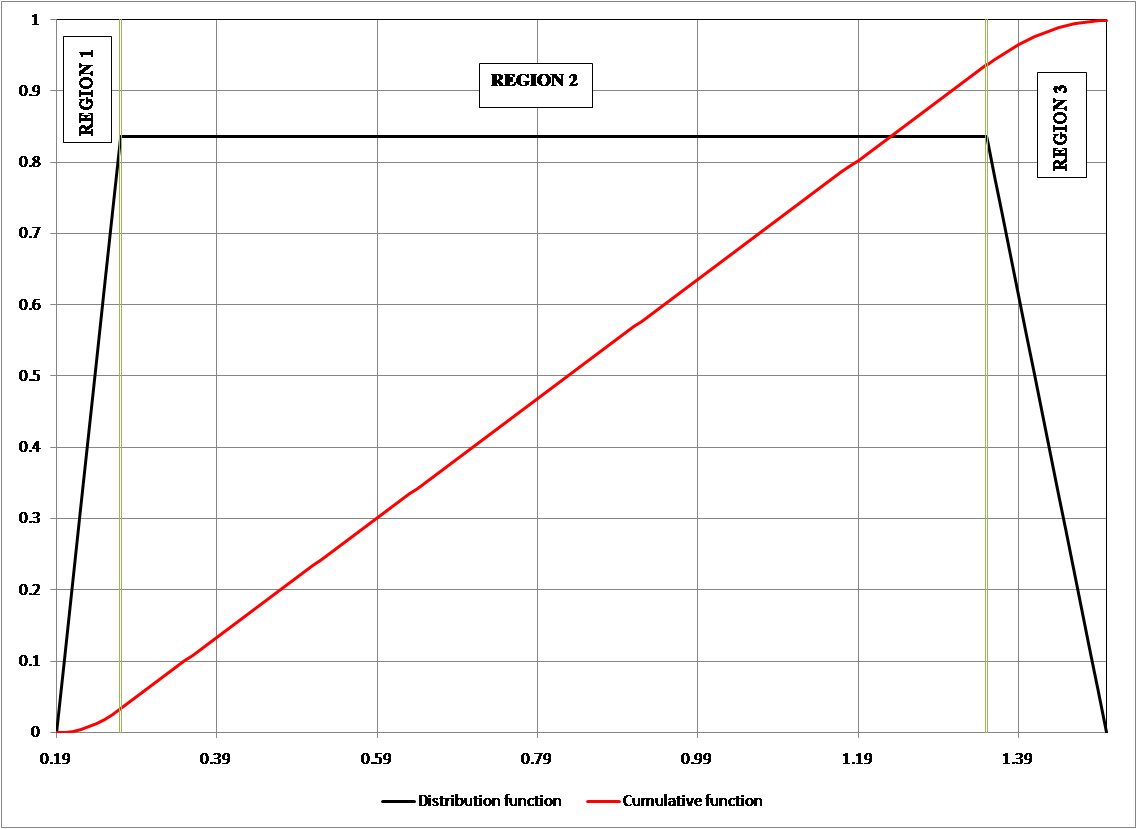


Figure 3 Distribution and Cumulative functions

Normalization of the density function:

Solving the system of equations:

Once the equations and the parameters are defined, a random value ξ uniformly distributed in is obtained by using the internal FORTRAN function "random number".

Then the region where this value falls is identified (the regions are now defined by the cumulative function) and the inverse function of the cumulative distribution is used to obtain the sampled value . For the general trapezoidal distribution defined above:

* Region 1 corresponds to sampled values in , where is the cumulative value at .

The cumulative function for this region is a quadratic function of :

, with

After some work on the quadratic formula the solution can be written as:

Only the positive value is valid as the distribution function is defined in the

interval .

* Region 2 corresponds to sampled values in ,.

The cumulative function for this region is:

, with

Then

* Region 3 corresponds to sampled values within , where . The

cumulative function for this region is a quadratic function of :

, with

After some work on the quadratic formula, the solution can be written as:

Or, using the expressions for , and the expression used in the coding can

be obtained:

Only the negative value is taken to stay within the defined range.

### Normal-based distributions sampled with the Box-Mueller method

The cumulative distribution function of the normal distribution cannot be inverted analytically and therefore the inverse method is not applicable.

The Box-Müeller method generates values from the normal distribution by generating two independent random variables at a time.

This method considers two independent and normally distributed random variables and with a distribution function , hxi i = 0 and 1 for (the subscript refers to normal distribution). By then using the polar coordinates the resulting cumulative distribution can be analytically solved and the inverse method can be applied. The formulas for the two independent random variables are:

Where and are two independent random numbers.

To generate a random variable distributed normally with a mean equal to and a standard deviation , only a linear transformation has to be applied so that:

Only the first formula using is used in the coding.

For the log normal distribution, the same procedure is followed and the exponential of the normally distributed value is the log-normally distributed value.

# Simulation phase

During the "simulation" phase uncertainty runs plus the base case are executed.

The needed files to execute this phase are the base case input deck and the uncertainty weight files generated in the "setup" phase. The code reads the sampled values for the "input treatable" quantities and substitutes them in the adequate input data cards from the base case input deck file, and applies the sampled values for the "source correlation" quantities to the appropriate correlations coefficients.

The execution of an uncertainty run generated the output print and restart plot files. The given names of the generated files are the same as the base case input deck plus an appended number identifying the uncertainty simulation, written in a four-digit format.

# Post-processing phase

The last phase of the package generates the uncertainty bounds for the desired code output quantities.

The required data in the "post-processing" input file are the output quantities for which the tolerance intervals are to be generated; and the uncertainty run numbers to be used for that purpose.

By specifying the run numbers, the user can avoid the use of a failed calculation, specify a different set of calculations, and evaluate the sample effect and order effect.

The output quantities are specified using the available "minor edits", in the same way they are specified in a strip process. The output quantities are the time-dependent variables extracted from each uncertainty restart-plot file.

The Wilks' formula related data is read from the restart-plot file generated in the "setup" phase (suffix \*.rs).

The tolerance intervals are generated by statistical treatment of the output data.

The output values generated during the "simulation" phase are ordered according to its rank from the lower value (order 1) to the maximum value (order , being the number of calculations) at each time step (minor edit frequency). The number of timecurves obtained after the ordering process is the same as the number of the calculations included to derive the uncertainty bounds. Depending upon the user-specified order, the code will use the first and last orders (1st order), second and order (2nd order), and so on.

After the execution of this phase, the code will produce graphs containing the time history for the:

* Upper and lower uncertainty bound, i.e. unilateral tolerance bounds of the specified percentile and (1-percentile), respectively, confidence level, and order of Wilks' application.
* Base case time trend.
* Span between the upper and lower bounds at each time step.

The "post-processing" phase also generates EXCEL compatible files with the sorted values for each requested output quantity.

# Cards 291DDWWW – weight information

|  |  |  |  |
| --- | --- | --- | --- |
| W1  (R) | Weight related number.  Note that the options in W2 affect the usage of W1. | -1. | a weight will be computed according to W2 - W6 entered information, i.e. the weight will be computed using the specified distribution type and its characteristic parameters. |
| 0. | the previously computed or entered weight will be used. |
| >0. | weight to be used for all uncertainty code runs (bias): instead of simple random sampling the same weight is applied for all uncertainty code runs |
| W2  (I) | This number consists of two digits and a sign: **-wn**.   * The minus sign, if entered, will reduce the weight such that the product of the base value and the weight is not greater than 1. * The digit w controls both the computation and the application of the weights. * The digit n allows normalization over a selected range of weights. The normalization consists of applying an additional factor such that the sum of the resulting weights times the base values is equal to the sum of the base values, i.e. when the normalization capability is applied:   sum base values = sum uncertainty run values | w=0 | the weight is the quantity entered in W1 and thus W1 must be positive non-zero. |
| w=1 | the previously computed or entered weight will be used and thus W1 must be 0.0. The previously used weight is from a previous set on this card or previous cards. |
| w=2 | a weight is computed and used for all the words within this set |
| w=3 | a different weight is computed for each word within this set.  If the user desires a different user specified weight, i.e. W1>0. (bias) for each word, a set must be entered for each word. |
| n=0 | the state of the normalization (either done or not done) is continued over the words in this set |
| n=1 | indicates the start of a normalization section |
| n=2 | the normalization is terminated with the last word specified in the set |
| n=3 | the normalization section begins with the first word of the set and ends with the last word of the set. |
| W3 | Card number containing quantities to be modified. | | |
| W4 | Word number of the first quantity in the card to be modified. | | |
| W5 | Word number of the last quantity in the card to be modified. | | |
| w6 | Skip factor to move from W4 to W5. This quantity must be entered even if it’s equal to one. | | |

# Examples

## Example 1 – General case I

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| RS base case input deck | 20100301 | | | 32. | | | 2.9267e-3 | | | | 392. | | 2.9267e-3 | | | |
| 20100302 | | | 752. | | | 1.2478e-3 | | | |  | |  | | | |
| 20100303 | | | 1112. | | | 4.7297e-3 | | | |  | |  | | | |
| 20100304 | | | 1472. | | | 5.0508e-3 | | | |  | |  | | | |
| Uncertainty data | 29101000 | ND | 1. | | 0.001 | 0. | | 0. |  |  | |  |  |  |  |  |
| 29101001 | -1. | 20 | | 20100301 | 2 | | 4 | 2 | 0. | | 10 | 20100302 | 2 | 2 | 1 |
| 29102000 | UD | 0.85 | | 1.08 | 0. | | 0. |  |  | |  |  |  |  |  |
| 29102001 | -1. | 20 | | 20100303 | 2 | | 2 | 1 | 0. | | 10 | 20100304 | 2 | 2 | 1 |

Normal distribution (mean =1., sd = 0.001) for thermal conductivity values corresponding to temperatures range [32.,800.] F:

Thermal conductivity values corresponding to cards 20100301 and 20100302 have the same weight, thus:

* 29101001 W1=-1. indicates that a value is to be computed, W7=0. and W9=10 indicate that the previous computed value is to be used.
* 29101001 W2=20 indicates that the computed weight will be used for all the words within this set.
* 29101001 W4=2, W5=4, W6=2 indicates that the weight is to be applied to W2 and W4 of card 20100301 (and thus the skip factor=2=W6)

Uniform distribution (between [0.85,1.08]) for thermal conductivity values corresponding to temperature range [800.,1500.]:

Thermal conductivity values corresponding to cards 20100303 and 20100304 have the same weight, thus:

* 29102001 W1=-1. indicates that a value is to be computed, W7=0. and W9=10 indicate that the previous computed value is to be used.
* There is only one pair of data per input card, therefore 29102001 W4=2, W5=2, W6=0, i.e. word number of the first quantity = word number of the last quantity in the card to be modified and skip factor is zero.

## Example 2 – General case II

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| RS base case input deck | 1001201 | | | 3 | | | 2245.1 | | | | 589.25 | | 0. | 0 | | 0 | 1 |
| 501 | | | p | | | 150010000 | | | | le | | null | 0 | | 1860. | l |
| Uncertainty data | 29101000 | ND | 1. | | 0.02 | 0. | | 0. |  |  | |  |  |  |  |  | |
| 29101001 | -1. | 30 | | 1001201 | 2 | | 3 | 1 | -1. | | 30 | 501 | 6 | 6 | 1 | |

Normal distribution (mean =1., sd = 0.02) for initical condition values (pressure, temperature) and for set point pressure in trip 501. Each quantity requries a different weight:

* All three quantities share the same distribution information, i.e. card 29101000
* 29101001 W2=30 1001202 2 3 1: in this set two uncertainty weights are requested (from W2 to W3 with a skip factor of 1), but different weights are applied to each word (w=3), i.e. two values distributed according to the same type of distribution.

## Example 3 – Normalization of a range of uncertainty values

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| RS base case input deck | 13360701 | | | 1000 | | | 0.141211 | | | | 0. | | 0.003769 | 1 | | |
| 13360702 | | | 1000 | | | 0.204599 | | | | 0. | | 0.005461 | 2 | | |
| 13360703 | | | 1000 | | | 0.208339 | | | | 0. | | 0.005561 | 3 | | |
| 13360704 | | | 1000 | | | 0.206780 | | | | 0. | | 0.005520 | 4 | | |
| 13360705 | | | 1000 | | | 0.174414 | | | | 0. | | 0.004656 | 5 | | |
| 13360706 | | | 1000 | | | 0.038668 | | | | 0. | | 0.001032 | 6 | | |
| Uncertainty data | 29101000 | ND | 1. | | 0.015 | 0. | | 0. |  |  | |  |  |  |  |  |
| 29101001 | 1. | 01 | | 13360701 | 2 | | 2 | 1 | 0. | | 10 | 13360702 | 2 | 2 | 1 |
| 29101002 | -1. | 20 | | 13360703 | 2 | | 2 | 1 | 0. | | 10 | 13360704 | 2 | 2 | 1 |
| 29101003 | 1. | 00 | | 13360705 | 2 | | 2 | 1 | 0. | | 12 | 13360706 | 2 | 2 | 1 |

Normal distribution (mean =1., sd = 0.015) is applied to the 2 maximum values of the axial power distribution

* The multiplier is to be applied to W2 in cards 13360703 and 13360704, therefore:
  + 29101002 W1=-1. to request a random sampled value, W2=20, where w=2 indicates that the same sampled value will be used for all values in this set (in this set there is only one word); n=0 indicates to keep the normalization state, i.e. keep it on.
* Stricly speaking no multpilier is to be applied to the other axial levels, however this HS corresponds to the fuel and therefore the six values (W2 in cards 13360701 to 13360706) have to add up to the original sum:
  + Initialization of the normalization is indicated in 29101001 W2, n=1
  + Termination of the normalization is indicated in 29101002 W8, n=2
  + Note that in this case a certain fraction of the power corresponds to direct moderator heating multiplier (W4 in cards 1CCCG07XX), therefore the normalization request during the generation of the uncertainty values will keep the base case sum of the axial multipliers specified in W2 in cards 1CCCG07XX.

## Example 4 – Limitation of the uncertainty value to a maximum value of 1.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| RS base case input deck | 20200200 | | | | power | | | | 501 | | | 1.0 | | 3600. | | |
| 20200201 | | | | -1.0 | | | | 1. | | |  | |  | | |
| 20200202 | | | | 0.0 | | | | 1. | | |  | |  | | |
| 20200203 | | | | 0.2 | | | | 1. | | |  | |  | | |
| 20200204 | | | | 0.3 | | | | 0.988 | | |  | |  | | |
| 20200205 | | | | 0.7 | | | | 0.72 | | |  | |  | | |
| 20200206 | | | | 0.9 | | | | 0.49 | | |  | |  | | |
| 20200207 | | | | 1.0 | | | | 0.34 | | |  | |  | | |
| 20200208 | | | | 2.0 | | | | 0.119 | | |  | |  | | |
| 20200209 | | | | 3.0 | | | | 0.107 | | |  | |  | | |
| 20200210 | | | | 4.0 | | | | 0.0986 | | |  | |  | | |
| 20200211 | | | | 10.0 | | | | 0.0739 | | |  | |  | | |
| 20200212 | | | | 1.0+5 | | | | 0.00146 | | |  | |  | | |
| Uncertainty data | 29101000 | ND | 1. | 0.01 | | 0. | 0. |  | |  |  |  |  | |  |  |
| 29101001 | -1.0 | -20 | 20200204 | | 2 | 2 | 1 | | 0.0 | -10 | 20200205 | 2 | | 2 | 1 |
| 29101002 | 0.0 | -10 | 20200206 | | 2 | 2 | 1 | | 0.0 | -10 | 20200207 | 2 | | 2 | 1 |
| 29101003 | 0.0 | -10 | 20200208 | | 2 | 2 | 1 | | 0.0 | -10 | 20200209 | 2 | | 2 | 1 |
| 29101004 | 0.0 | -10 | 20200210 | | 2 | 2 | 1 | | 0.0 | -10 | 20200211 | 2 | | 2 | 1 |
| 29101005 | 0.0 | -10 | 20200212 | | 2 | 2 | 1 | |  |  |  |  | |  |  |

Normal distribution (mean =1., sd = 0.01) is applied to the 2 maximum values of the axial power distribution

* The multiplier is to be applied for time ≥ 0.3s and the resulting multiplier cannot be greater than 1. as we do not expect the power to increrase after the scram.
* 29101002 W1=-1. to request a random sampled value, W2=20, where w=2 indicates that the same sampled value will be used for all values in this set (in this set there is only one word); n=0 indicates to keep the normalization state, i.e. keep it on