Operator GENE_VARI_ALEA

1 Goal

To generate a realization of a real random variable of law of probability given (laws gamma or exponential exits of the application of the maximum of entropy, [R4.03.05]).

At exit, a reality is obtained.

2 Syntax

```
[reality]    =  GENE_VARI_ALEA   (

    ◊  / TYPE   =   ' GAMMA'                                [DEFECT]
          ◊  VALE_MOY  =  /  vale_moy                       [R]
                           / 1.0                            [DEFECT]
          ◊  BORNE_INF =  /  has                               [R]
                          /  0.                             [DEFECT]
           ◊  COEF_VAR =  /  delta                           [R]
                          /  0.1                             [DEFECT]

       /  TYPE   =  'EXPONENTIAL'
          ◊  VALE_MOY  =  /  vale_moy                       [R]
                          /  0.                             [DEFECT]
          ◊  BORNE_INF =  /  has                               [R]
                          /  -1.0                           [DEFECT]

       /  TYPE   =  'EXP_TRONQUEE'
          ◊  VALE_MOY  =  /  vale_moy                       [R]
                          /  0.                             [DEFECT]
          ◊  BORNE_INF =  /  has                               [R]
                          /  -1.0                           [DEFECT]
          ◊  BORNE_SUP =  /  B                               [R]
                          /  1.0                            [DEFECT]
    ◊  INIT_ALEA =  nor                                     [I]
    );
```

3 Operands

3.1 Keyword TYPE

According to information usable on the random variable to simulate, three types of law of probability are available. If information available is a support not limited $[a,+\infty[$, an average $\underline{w}$, and a scatter coefficient δ, the law is gamma. If information available is a support not limited $[a,+\infty[$ and an average $\underline{w}$, the law is exponential. If information available is a compact support $[a,b]$ and an average $\underline{w}$, the law is exponential truncated.

/ TYPE = ' GAMMA' [DEFECT]

The random variable follows a law of probability of type "gamma" of which probability distribution $P_W(dw)$ is defined by :

$$P_W(dw)=I_{[a,+\infty[}(w)\frac{\left(\underline{w}\delta^2-a\delta^2\right)^{-\frac{1}{\delta^2}}}{\Gamma\left(\frac{1}{\delta^2}\right)}(w-a)^{\frac{1-\delta^2}{\delta^2}}\exp\{-\frac{w-a}{(\underline{w}-a)\delta^2}\}dw$$

with $\Gamma(z)=\int_0^{+\infty}t^{z-1}e^{-t}\,dt$

/ TYPE = 'EXPONENTIAL'

The random variable follows a law of "exponential" probability of the type of which probability distribution $P_W(dw)$ is defined by :

$$P_W(dw)=I_{[a,+\infty[}(w)\frac{1}{\underline{w}-a}\exp\{-\frac{w-a}{\underline{w}-a}\}dw$$

with $I_{[a,+\infty[}(w)=1$ si $w\in[a,+\infty[$ et $I_{[a,+\infty[}(w)=0$ si $w\notin[a,+\infty[$

/ TYPE = 'EXP_TRONQUEE'

The random variable follows a law of exponential probability of the type "truncated" of which probability distribution $P_W(dw)$ is defined by:

$$P_W(dw)=I_{[a,b]}(w)\frac{k}{\alpha(k)}e^{-kw}\,dw$$

with $I_{[a,b]}(w)=1$ si $w\in[a,b]$ et $I_{[a,b]}(w)=0$ si $w\notin[a,b]$ and where k is such as $(\underline{w}k-1)\alpha(k)-k\beta(k)=0$, with $\alpha(k)=e^{-ak}-e^{bk}$ and $\beta(k)=ae^{-ak}-be^{-bk}$.

3.2 Keyword VALE_MOY

```
◊    VALE_MOY  =  /  w                    [R]
                  /  0. or 1.0            [DEFECT]
```

Indicate the median value of the random variable to simulate.

3.3 Keyword COEF_VAR

◊	COEF_VAR =	/	δ	[R]
		/	0.1	[DEFECT]

This keyword informs the scatter coefficient (standard deviation report on absolute value of the average). The value taken by default is 0.1.

3.4 Keywords BORNE_INF AND BORNE_SUP

◊	BORNE_INF =	/	a	[R]
		/	-1.0 or 0.	[DEFECT]
◊	BORNE_SUP =	/	b	[R]
		/	+1.0	[DEFECT]

These keywords inform the lower limit and the upper limit (when they exist) support $[a,b]$ or $[a,+\infty[$ laws.

3.5 Operand INIT_ALEA

◊	INIT_ALEA =	/	nor	[I]

Cause initialization with sound `nor`ième term of the continuation of pseudo-random numbers used for the generation of the variables.

If the keyword `INIT_ALEA` is absent, the terms used of the continuation are those immediately consecutive with those already used. If no term were still used, the continuation is initialized in its first term.

Suggestion:

With less than one particular use, it is advised not to inform the keyword `INIT_ALEA` in the operators according to: `GENE_FONC_ALEA`, `GENE_VARI_ALEA` and `GENE_MATR_ALEA`. In this case, with the first call to the one of these operators, the continuation of pseudo-random numbers is initialized in its first term. The omission of the keyword `INIT_ALEA` to each call of these operators in the command file the statistical independence of the pseudo-random numbers used guarantees.

Note:

The germ of the continuation remains identical of one execution to the other of Code_Aster; the results thus remain rigorously identical (one can thus test to it not regression of not converged statistical results). If one wishes to generate results statistically independent from one execution to another, then the keyword should be used `INIT_ALEA` with values raising the number of terms used in the former executions.

Caution:

The generator of random variable used is that of the module "random" of Python. It depends on the version of Python exploited by Code_Aster. Not statistically converged results can thus vary from one version to another of Code_Aster or platform to another, if the version of Python is not the same one and that between the two versions the module random evolved (case between Python 2.1 and 2.3).

Note:

In version Python 2.3, the period of the generator is of 2 ** 19937-1.

4 Example

By call, the order generates only one realization of the random variable to simulate. To generate several achievements of the same random variable, it is necessary to repeat the order without changing its parameters or placing the order in a loop of the process control language of *Code_Aster* - the language Python. Each realization is statistically independent of the other achievements.

In the following example, one generates ns achievements of a random variable gamma of median value 25000, support and scatter coefficient 0.1 positive realities. These achievements are then used as values of stiffness of shock.

```
ns=100

for K in arranges (1, ns+1):

# Generation
    KN=GENE_VARI_ALEA ( TYPE      =  'GAMMA',
                        BORNE_INF =  0. ,
                        VALE_MOY  =  25000. ,
                        COEF_VAR  =  0.1,
                        )

    DYN=DYNA_TRAN_MODAL (
                        …
                        CHOC=_F (
                                  …
                                  RIGI_NOR  =  KN,
                                  …
                                  )
                        )

# Here for example, statistical processing of DYN

    TO DESTROY (CONCEPT=_F (NOM= (DYN, KN)))

# End of the loop (indentation)
```

For more complete examples, one can consult documentation "Digital simulation of Monte Carlo" [U2.08.05] or the case test SDNS01 [V5.06.001].