Operator GENE_FONC_ALEA

1 Drank

To generate a trajectory of a multivariate process stochastic monodimensional (i.e with several components and indexed on only one variable), steady, of average null from his matrix of spectral concentration (interspectral matrix). The trajectories of the stochastic process are temporal functions which make it possible to carry out a transient dynamic computation then.

The trajectories obtained have an interspectral matrix which converges on average towards the interspectral matrix targets and are the achievements of a process asymptotically gaussian (i.e when the number of pullings tends towards the infinite one). The algorithm used is an algorithm of simulation per trigonometrical series with phase random and transformed of opposite fast Fourier. Product

a concept of the type counts `_fonction. Syntax`

2 vf [array

```
    _fonction] = GENE_FONC_ALEA (♦INTE
                        _SPEC=intf    [        interspectrum          ]
◊NUMÉRIQUE
                        _VITE_FLUI=nk  [I]       #                Case

                              with authorized interpolation: ◊/◊
                                 INTERPOL=' OUI'     ,   [DEFAULT        ]
_TIRAGE=durée
                              ◊DUREE        [R] ◊FREQ
                              _INIT=fi   [R]          ◊FREQ
                              _FIN=ff    [R]          #            Case

                              with interpolation not - authorized: /◊INTERPOL
                                 = ' NON'     ,   ◊NB

                        _POIN=nb_poin              [I] ◊NB

                        _TIRAGE=/nt   [I       ]      /1           [DEFAULT
                                                              ] ◊INIT

                        _ALEA=ni   [I]             ◊INFO

                        =/1  [DEFAULT                             ] /2 ◊
                                                  TITER

                        =titer               [  l_Kn]           ); Operands
```

3 Operand

3.1 INTE_SPEC ♦INTE

_SPEC = intf [interspectrum] Name of

the interspectral matrix towards which the interspectral matrix of the generated signals must tend. The interspectral matrix

is a complex matrix, whose each term is written where $S_{XY}(f)=\int_{-\infty}^{+\infty} E[X(t)Y(t-\tau)]e^{-2i\pi f\tau} d\tau$ is $E[\cdot]$ the expectation, and where and are X Y two steady random processes (for example two components of a loading in two points distinct from a mesh). Note:

To be physical, the interspectral matrix must be a definite-positive hermitian matrix. This

interspectral matrix is generated mainly by the operators: DEFI_
INTE_SPEC, LIRE_INTE_SPEC and CALC _ INTE_SPEC. For

further information on the meaning of the parameters, the reader is invited to consult of the command documentation DEFI_ INTE_SPEC [U4.36.02]. Operand

3.2 NUME_VITE_FLUI ◊NUMÉRIQUE

_VITE_FLUI = nk [I] This
sequence number

corresponds at a rate of flow if the interspectrums model, via operators CALC_ FLUI_STRU and DEFI _ SPEC_TURB, a turbulent excitation induced by a fluid flow. Operand

3.3 INTERPOL ◊INTERPOL

=/"OUI'

[DEFAULT] One
authorizes

to interpolate the functions in frequency constituting the terms of the interspectral matrix. In particular, the new discretization will depend on the period of the signal to generate (key word DUREE_TIRAGE) and amongst points of the FFT (key word NB_POIN) . /"NON

" the values

of the interspectrum used are only the existing values (not interpolation used). Caution:

If INTERPOL

= ' NON", it is necessary that: the various

- *terms of the interspectral matrix have exactly the same discretization and with a constant step in frequency. If the number of points of discretization of the interspectrum is not a*

power of 2, and/or with the interspectrum one is not compatible or of alarms are emitted and a power of 2 ad hoc is chosen, the interspectrums
- *are sufficiently finely discretized to allow a generation of temporal with sufficient times. _TIRAGE=durée*

3.4 operand DUREE_TIRAGE

◊DUREE [R] Lasted

of the signal to generate, for each pulling (the total period will thus be ntdurée $\times$). If

key word DUREE_TIRAGE is not present, the period of the signal to be generated is calculated by where durée$=\frac{1}{\Delta f}$ is Δf the step in frequency of the interspectrum (not minimum of origin of the interspectrum, or not calculated starting from key keys FREQ_INIT , FREQ_FIN and NB_POIN). Note:

The generated

signals start at time 0. with a value zero. Note:

To obtain the desired period, the algorithm of generation adjusts the number of points used in the FFT (cf [§3.6] Operand NB_POIN ◊ NB_POIN prolongs if need the interspectrum beyond the frequencies min and max by zero values. Operands

3.5 FREQ_INIT /FREQ_FIN ◊FREQ

_INIT = fi [R] ◊FREQ
_FIN = FF [R] First

and last values of frequency for which the interspectrum will be taken into account. The presence

of these key words causes to truncate the interspectrum. If the key words are not present, in fact the values of minimal and maximum frequency of the interspectrum are used. Operand

3.6 NB_POIN ◊ NB_POIN

= nb_poin [I] Number of points

of discretization of the interspectrum to be used in the algorithm of generation. This number

must be a power of 2 because it corresponds to the number of points of the fast transform of Fourier reverses used by the algorithm of generation. If such is not the case, it is the power of 2 immediately above than nb_poin which is retained. If

key word NB_POIN is not present, the number of points is calculated so that the theorem of Shannon is respected, i.e. that, where $\frac{1}{\Delta t}>2\cdot f_{max}$is Δt the step in time of the signal to generate (which depends on NB_POIN and DUREE_TIRAGE) and where is f_{max} the maximum frequency selected of the interspectrum. Suggestion

: It

is advised not to specify the number of points, the coded algorithm choosing in this case automatically the optimal value. In particular, if key word DUREE_TIRAGE `and NB_POIN are not` *present, then one is assured that the generated signal is coherent at the same time with the step of discretization of the interspectrum and with the maximum frequency. However, if the user wants to specify NB_POIN , the two remarks which follow must help there. Note:*

If

key word DUREE_TIRAGE `and NB_POIN are` *present then one a: In this case,* $\Delta t=\frac{1}{2}\times\frac{\text{durée}}{\text{nb\_poin}}$ *if nb_poin* `is too` *small compared to period* `so that` *the theorem of Shannon is respected; then an alarm is emitted and it is the minimum number of points allowing the respect of the theorem of Shannon which is retained. The value specified by NB_POIN is thus taken into account only when it is higher than the value minimum. By imposing a number of points raised, one can force the signal to have a step of discretization in time smaller than that by default. It is necessary then to be conscious that the discretization of temporal is finer than the maximum frequency of the interspectrum allows it theoretically. Note:*

If INTERPOL

`= ' NON' and if` *nb_poin* `is not` *compatible with the interspectrum, an alarm is emitted and a power of 2 ad hoc is selected. Note:*

The number

of points constituting the generated signals is equal to twice the number of points of discretization of the interspectrum, and thus to twice nb_poin `when` *the key word is indicated. This number of points is useful to know for a posterior use in CALC_* `INTE_SPEC [U4.36` *.03] (cf [§ 3.9] 3.9 INFO ◊INFO [§ 6] " 6 # This*

3.7 NB_TIRAGE ◊ NB_TIRAGE

`= NT [I] Many`

pullings which must contain the generated temporal signals. The signals results will contain NT end to end put statistically independent pullings. One can
then post-treat the results got from these signals generated with operator CALC_ `INTE_SPEC [U4.36` .03], cf [§6] "Examples6# This

If

key word DUREE_TIRAGE `is present` *, the total period of signal will be NT lasted* $\times$ *.* `However`*, it is not of course equivalent to generate 1 pulling of lasted period NT* $\times$ `_tirage and NT` *pullings of lasted period* `end to end put _tirage` *. There is statistical independence between the various sections of period D in the second case and not in the first. Operand*

3.8 INIT_ALEA ◊INIT

`_ALEA=ni [I] Causes`

the initialization in its nor-ième term of the continuation of pseudo-random numbers employed for the generation of the signals. If

key word 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

: A less

than one particular use, it is advised not to inform key word INIT_ALEA in *the following operators: 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 key word INIT_ALEA* to each *call of these operators in the command file guarantees the statistical independence of the pseudo-random numbers used. 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 non regression of the results statistical ones not converged). If one wishes to generate results statistically independent from one execution to another, then it is necessary to use key word 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 modulus "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 modulus 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. Operand

3.9 INFO ◊INFO

= /1: not

d'impression. /2: printing

of time step, initial time, the final time of the generated signals, and amongst points used in the transform of fast Fourier opposite. Note:

INFO=

2 makes it possible to know the number of points constituting the generated signals (it is twice the number of points used in the fast transform of Fourier reverses.) It is to better use this number of points then in operator CALC_ INTE_SPEC key word NB_POIN (cf [§ 6]" 6 # This

3.10 TITER ◊TITER

= title title

is the title of computation to be printed at the top of the results [U4.03.01]. Phase

4 of checking Various

checks are carried out to ensure that the data are coherent (nb_poin sufficiently large compared to the maximum frequency and period to generate, lasted sufficiently great compared to the discretization of the interspectrum, etc…). Alarms are emitted if necessary. If key word NB_POIN is present , then it is checked that the value given is a power of 2. In the contrary case, an alarm is emitted and the value is modified. If INTERPOL = ' NON', one checks that the various terms of the interspectral matrix have the same discretization with a constant step in frequency. Count

5 produced the parameters

of the produced array are following PARAMETRETYPEDESCRIPTIONNUME

		_ORDREInuméros
		of ordreFONCTIONK24nom
		generated functions Examples

6 # This

```
example is only giving an idea of the syntax and # of
the useful associated operators (for the recovery of generated functions #
and their possible checking). # the interspectrum
used does not have meaning. There is not # of use
of the generated functions (with a dyna_tran_modal by # example
). #
the cases test zzzz180a and sdll107a provide more complete examples #
Definition

of the interspectrum spect
11 = DEFI_FONCTION (NOM_PARA
                      = "FREQ", VALE_C
                      = (0. , 10
                                      . , 0     .   , 50.0,10
                                      . , 0  .   , 150     . ,
                                      0.1,0  .   ,)     ) spect



12 = DEFI_FONCTION (NOM_PARA
                      = "FREQ", VALE_C
                      = (0. , 0.5
                                      , 0.8         ,    150   . ,
                                      0.5,0.8       ,   )   ) spect



22 = DEFI_FONCTION (NOM_PARA
                      = "FREQ" VALE_C
                      = (0. , 1 .
                                      , 0.      ,    150      . ,
                                      1. , 0 .   ,)     ) mat

                      _int

= DEFI_INTE_SPEC (DIMENSION
                      = 2, PAR_FONCTION
                      = (_F (         NUME_ORDRE_I = 1, NUME_ORDRE_J
                                            = 1, FONCTION
                                            = sp11), _F (NUME_ORDRE_I
                                            = 1, NUME_ORDRE_J
```

```
                                                  = 2, FONCTION
                                                  = sp12), _F (NUME_ORDRE_I
                                                  = 2, NUME_ORDRE_J
                                                  = 2, FONCTION
                                                  = sp22))) #
                                           Generation


of the two temporal functions vect
= GENE_FONC_ALEA (INTE_SPEC=mat_int     , DUREE
                       _TIRAGE=5.,   NB_TIRAGE
                       =10,   )    # Recovery


of the two functions for example for an IMPR_FONCTION FONC1
= RECU_FONCTION (TABLE=vect   , FILTRE
                       =_F (NOM_PARA = "NUME_ORDRE", VALE_I
                                 = 1), NOM_PARA
                       _TABL=' FONCTION',) FONC2

= RECU_FONCTION (TABLE=VECT1  , FILTRE
                       =_F (NOM_PARA = "NUME_ORDRE", VALE_I
                                 = 2), NOM_PARA
                       _TABL=' FONCTION',) # Checking

: Computation of the interspectrum of generated functions # Attention
: the value given to NB_POIN is important. It is # desirable
to take it equal to the constituent number of points #
the functions (2*nb_fft if coming from GENE_FONC_ALEA). INTERS
=CALC_INTE_SPEC (INST_INIT=0  ., INST_
                       FIN=50., DUREE
                       _ANALYZE=5., DUREE
                       _DECALAGE=5., NB_POIN
                       =2048, FONCTION
                       = (FONC1, FONC2  ,)) # Recovery

of the auto--spectrum of FONC1 for comparison with spetc11 F11
=RECU  _FONCTION (INTE_SPEC=INTERS, NUME_
                    ORDRE_I=1,)
```