Opérateur GENE_FONC_ALEA

1 But

Générer une trajectoire d'un processus stochastique multi-varié mono-dimensionnel (i.e. à plusieurs composantes et indexé sur une seule variable), stationnaire, de moyenne nulle à partir de sa matrice de densité spectrale (matrice interspectrale). Les trajectoires du processus stochastique sont des fonctions temporelles qui permettent d'effectuer ensuite un calcul dynamique transitoire.

Les trajectoires obtenues ont une matrice interspectrale qui converge en moyenne vers la matrice interspectrale cible et sont les réalisations d'un processus asymptotiquement gaussien (i.e. quand le nombre de tirages tend vers l'infini). L'algorithme utilisé est un algorithme de simulation par séries trigonométriques à phase aléatoire et transformée de Fourier rapide inverse.

Les fonctions temporelles produites sont déposées dans une structure de données `interspectre` . Chaque fonction est repérée par son numéro d'ordre. Elle peut être extraite en utilisant l'opérateur `RECU_FONCTION` [U4.32.03].

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

2 Syntaxe

```
vf [interspectre] = GENE_FONC_ALEA

                ( ♦ INTE_SPEC        =  intf          [interspectre]

                  ◊ NUME_VITE_FLUI   =  nk              [I]

                      # Cas avec interpolation autorisée :
                  ◊ /  ◊ INTERPOL     =  'OUI' ,        [DEFAUT]
                       ◊ DUREE_TIRAGE =  durée          [R]
                       ◊ FREQ_INIT    =  fi             [R]
                       ◊ FREQ_FIN     =  ff             [R]

                      # Cas avec interpolation non autorisée :
                    /  ◊ INTERPOL     =  'NON' ,

                  ◊ NB_POIN          =  nb_poin        [I]

                  ◊ NB_TIRAGE        =  /  nt          [I]
                                        /  1           [DEFAUT]

                  ◊ INIT_ALEA        =  ni             [I]

                  ◊ INFO             =  /  1           [DEFAUT]
                                        /  2

                  ◊ TITRE            =  titre          [l_Kn]

    )  ;
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

3 Opérandes

3.1 Opérande INTE_SPEC

♦ INTE_SPEC = intf [interspectre]

Nom de la matrice interspectrale vers laquelle la matrice interspectrale des signaux générés doit tendre.

La matrice interspectrale est une matrice complexe, dont chaque terme s'écrit $S_{XY}(f)=\int_{-\infty}^{+\infty} E\left[X(t)\,Y(t-\tau)\right] e^{-2i\pi f\tau}\,d\tau$ où $E[\cdot]$ est l'espérance mathématique, et où X et Y sont deux processus aléatoires stationnaires (par exemple deux composantes d'un chargement en deux points distincts d'un maillage).

Remarque :

Pour être physique, la matrice interspectrale doit être une matrice hermitienne définie-positive.

Cette matrice interspectrale est générée principalement par les opérateurs :

DEFI_INTE_SPEC, LIRE_INTE_SPEC et CALC_INTE_SPEC.

Pour de plus amples informations sur le sens des paramètres, le lecteur est invité à consulter la documentation de la commande DEFI_INTE_SPEC [U4.36.02].

3.2 Opérande NUME_VITE_FLUI

◊ NUME_VITE_FLUI = nk [I]

Ce numéro d'ordre correspond à une vitesse d'écoulement si les interspectres modélisent, via les opérateurs CALC_FLUI_STRU et DEFI_SPEC_TURB, une excitation turbulente induite par un écoulement fluide.

3.3 Opérande INTERPOL

◊ INTERPOL =

/ 'OUI' [DEFAUT]

On autorise à interpoler les fonctions en fréquence constituant les termes de la matrice interspectrale. En particulier, la nouvelle discrétisation dépendra de la durée du signal à générer (mot-clé DUREE_TIRAGE) et du nombre de points de la FFT (mot-clé NB_POIN).

/ 'NON'

Les valeurs de l'interspectre utilisées sont uniquement les valeurs existantes (pas d'interpolation utilisée).

Attention :

Si INTERPOL='NON', il faut que :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

- *les différents termes de la matrice interspectrale aient exactement la même discrétisation et avec un pas en fréquence constant. Si le nombre de points de discrétisation de l'interspectre n'est pas une puissance de 2, et/ou n'est pas compatible avec l'interspectre une ou des alarmes sont émises et une puissance de 2 ad hoc est choisie,*
- *les interspectres soient suffisamment finement discrétisés pour permettre une génération de temporels avec suffisamment d'instants.*

3.4 Opérande DUREE_TIRAGE

◊ DUREE_TIRAGE = durée [R]

Durée du signal à générer, pour chaque tirage (la durée totale sera donc nt × durée).

Si le mot-clé DUREE_TIRAGE n'est pas présent, la durée du signal à générer est calculée par $\text{durée} = \frac{1}{\Delta f}$ où Δf est le pas en fréquence de l'interspectre (pas minimum d'origine de l'interspectre, ou pas calculé à partir des mots clés FREQ_INIT, FREQ_FIN et NB_POIN).

Remarque :

Les signaux générés commencent à l'instant 0. avec une valeur nulle.

Remarque :

Pour obtenir la durée souhaitée, l'algorithme de génération ajuste le nombre de points utilisés dans la FFT (cf. [§3.6] Opérande NB_POIN) et prolonge si besoin l'interspectre au delà des fréquences min et max par des valeurs nulles.

3.5 Opérandes FREQ_INIT / FREQ_FIN

◊ FREQ_INIT = fi [R]
◊ FREQ_FIN = ff [R]

Première et dernière valeurs de fréquence pour lesquelles l'interspectre sera pris en compte.

La présence de ces mots-clés a pour effet de tronquer l'interspectre. Si les mots-clés ne sont pas présents, ce sont les valeurs de fréquence minimale et maximale de l'interspectre qui sont utilisées.

3.6 Opérande NB_POIN

◊ NB_POIN = nb_poin [I]

Nombre de points de discrétisation de l'interspectre à utiliser dans l'algorithme de génération.

Ce nombre doit être une puissance de 2 car il correspond au nombre de points de la transformée de Fourier rapide inverse utilisée par l'algorithme de génération. Si tel n'est pas le cas, c'est la puissance de 2 immédiatement supérieure à nb_poin qui est retenue.

Si le mot-clé NB_POIN n'est pas présent, le nombre de points est calculé afin que le théorème de Shannon soit respecté, c'est à dire que $\frac{1}{\Delta t} > 2 \cdot f_{max}$, où Δt est le pas en temps du signal à générer (qui dépend de NB_POIN et de DUREE_TIRAGE) et où f_{max} est la fréquence maximale retenue de l'interspectre.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Suggestion :

Il est conseillé de ne pas spécifier le nombre de points, l'algorithme codé choisissant dans ce cas automatiquement la valeur optimale. Notamment, si les mot-clés DUREE_TIRAGE et NB_POIN ne sont pas présents, alors on est assuré que le signal généré est cohérent à la fois avec le pas de discrétisation de l'interspectre et avec la fréquence maximale. Toutefois, si l'utilisateur veut spécifier NB_POIN, les deux remarques qui suivent doivent l'y aider.

Remarque :

Si les mot-clés DUREE_TIRAGE et NB_POIN sont présents alors on a : $\Delta t = \frac{1}{2} \times \frac{\text{durée}}{\text{nb_poin}}$

Dans ce cas, si nb_poin est trop petit par rapport à durée pour que le théorème de Shannon soit respecté; alors une alarme est émise et c'est le nombre de points minimum permettant le respect du théorème de Shannon qui est retenu. La valeur spécifiée par NB_POIN n'est donc prise en compte que lorsqu'elle est supérieure à la valeur minimum. En imposant un nombre de points élevé, on peut forcer le signal à avoir un pas de discrétisation en temps plus petit que celui par défaut. Il faut alors être conscient que la discrétisation du temporel est plus fine que le permet théoriquement la fréquence maximale de l'interspectre.

Remarque :

Si INTERPOL='NON' et si nb_poin n'est pas compatible avec l'interspectre, une alarme est émise et une puissance de 2 ad hoc est choisie.

Remarque :

Le nombre de points constituant les signaux générés est égal à deux fois le nombre de points de discrétisation de l'interspectre, et donc à deux fois nb_poin lorsque le mot-clé est renseigné. Ce nombre de points est utile à connaître pour une utilisation postérieure dans CALC_INTE_SPEC [U4.36.03] (cf [§ 3.9] " Opérande INFO " et [§ 6] " Exemples ").

3.7 Opérande NB_TIRAGE

◊ NB_TIRAGE = nt [I]

Nombre de tirages que doit contenir les signaux temporels générés. Les signaux résultats contiendront nt tirages statistiquement indépendants mis bout à bout.
On peut ensuite post-traiter les résultats obtenus à partir de ces signaux générés avec l'opérateur CALC_INTE_SPEC [U4.36.03], cf. [§6] "Exemples".

Attention :

Si le mot-clé DUREE_TIRAGE est présent, la durée totale de signal sera nt × durée. Cependant, il n'est bien sûr pas équivalent de générer 1 tirage de durée nt × durée_tirage et nt tirages de durée durée_tirage mis bout à bout. Il y a indépendance statistique entre les différents tronçons de durée d dans le second cas et pas dans le premier.

3.8 Opérande INIT_ALEA

◊ INIT_ALEA = ni [I]

Provoque l'initialisation à son ni-ième terme de la suite de nombres pseudo-aléatoires employée pour la génération des signaux.

Si le mot-clé INIT_ALEA est absent, les termes utilisés de la suite sont ceux immédiatement consécutifs à ceux déjà utilisés. Si aucun terme n'a encore été utilisé, la suite est initialisée à son premier terme.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Suggestion :

A moins d'un usage particulier, il est conseillé de ne pas renseigner le mot-clé INIT_ALEA *dans les opérateurs suivants :* GENE_FONC_ALEA, GENE_VARI_ALEA *et* GENE_MATR_ALEA*. Dans ce cas, au premier appel à l'un de ces opérateurs, la suite de nombres pseudo-aléatoires est initialisée à son premier terme. L'omission du mot-clé* INIT_ALEA *à chacun des appels de ces opérateurs dans le fichier de commande garantit l'indépendance statistique des nombres pseudo-aléatoires utilisés.*

Remarque :

Le germe de la suite reste identique d'une exécution à l'autre de Code_Aster; les résultats restent donc rigoureusement identiques (on peut ainsi tester la non régression de résultats statistiques non convergés). Si l'on souhaite générer des résultats statistiquement indépendants d'une exécution à l'autre, alors il faut utiliser le mot-clé INIT_ALEA *avec des valeurs majorant le nombre de termes utilisés dans les exécutions antérieures.*

Attention :

Le générateur de variable aléatoire utilisé est celui du module "random" de Python. Il dépend de la version de Python exploitée par Code_Aster. Des résultats non convergés statistiquement peuvent donc varier d'une version à l'autre de Code_Aster ou d'une plate-forme à l'autre, si la version de Python n'est pas la même et qu'entre les deux versions le module random a évolué (cas entre Python 2.1 et 2.3).

Remarque :

En version Python 2.3, la période du générateur est de 2**19937-1.

3.9 Opérande INFO

◊ INFO =

/ 1 : pas d'impression.

/ 2 : impression du pas de temps, du temps initial, du temps final des signaux générés, et du nombre de points utilisés dans la transformée de Fourier rapide inverse.

Remarque :

INFO=2 permet de connaître le nombre de points constituant les signaux générés (c'est deux fois le nombre de points utilisés dans la transformée de Fourier rapide inverse.) Il vaut mieux utiliser ce nombre de points ensuite dans l'opérateur CALC_INTE_SPEC *mot-clé* NB_POIN *(cf. [§ 6] " Exemples ").*

3.10 Opérande TITRE

◊ TITRE = titre

titre est le titre du calcul à imprimer en tête des résultats [U4.03.01].

4 Phase de vérification

Diverses vérifications sont effectuées pour assurer que les données sont cohérentes (nb_poin suffisamment grand par rapport à la fréquence maximale et la durée à générer, durée suffisamment grande par rapport à la discrétisation de l'interspectre, etc…). Des alarmes sont émises le cas échéant.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Si le mot-clé NB_POIN est présent, alors on vérifie que la valeur donnée est une puissance de 2. Dans le cas contraire, une alarme est émise et la valeur est modifiée.
Si INTERPOL='NON', on vérifie que les différents termes de la matrice interspectrale ont la même discrétisation avec un pas en fréquence constant.

5 Concept produit

Les fonctions temporelles produites sont déposées dans une structure de données interspectre. Chaque fonction est repérée par son numéro d'ordre. Elle peut être extraite en utilisant l'opérateur RECU_FONCTION [U4.32.03].

6 Exemples

```
# Cet exemple n'a pour but que de donner une idée de la syntaxe et
# des opérateurs associés utiles (pour la récupération des fonctions
# générées et leur éventuelle vérification).
# L'interspectre utilisée n'a pas de signification. Il n'y a pas
# d'utilisation des fonctions générées (avec un dyna_tran_modal par
# exemple).
# Les cas test zzzz180a et sdll107a fournissent des exemples plus complets

# Définition de l'interspectre
spect11 = DEFI_FONCTION(
                        NOM_PARA = 'FREQ',
                        VALE_C =  (
                                        0.     ,  10.,   0.,
                                        50.0   ,  10.,   0.,
                                        150.   ,  0.1,   0.,
                                  )
                        )

spect12 = DEFI_FONCTION(
                        NOM_PARA = 'FREQ',
                        VALE_C =  (
                                        0.     ,  0.5,   0.8,
                                        150.   ,  0.5,   0.8,
                                  )
                        )

spect22 = 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,
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
                                                NUME_ORDRE_J = 2,
                                                FONCTION = sp22)
                                        )
                        )

# Génération des deux fonctions temporelles via la structure de données
# interspectre
vect = GENE_FONC_ALEA(INTE_SPEC  =  mat_int,
                      DUREE_TIRAGE =  5.,
                      NB_TIRAGE =  10,
                      )

# Récupération des deux fonctions par exemple pour un IMPR_FONCTION
FONC1 = RECU_FONCTION( INTE_SPEC = vect,
                      NUME_ORDRE = 1)

FONC2 = RECU_FONCTION( INTE_SPEC = VECT1,
                      NUME_ORDRE = 2)

# Vérification : Calcul de l'interspectre des fonctions générées
# Attention : la valeur donnée à NB_POIN est importante. Il est
# souhaitable de la prendre égale au nombre de points constituant
# les fonctions (2*nb_fft si provenant de GENE_FONC_ALEA).
INTERS=CALC_INTE_SPEC( INST_INIT=0.,
                      INST_FIN=50.,
                      DUREE_ANALYSE=5.,
                      DUREE_DECALAGE=5.,
                      NB_POIN=2048,
                      FONCTION=( FONC1,  FONC2, ))

# Récupération de l'auto-spectre de FONC1 pour comparaison à spetc11
F11  =RECU_FONCTION(INTE_SPEC=INTERS,
                    NUME_ORDRE_I=1,
                    )
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)