Opérateur DYNA_NON_LINE

1 But

Calculer l'évolution dynamique d'une structure dont le matériau ou la géométrie ont un comportement non linéaire. Il peut s'agir par exemple de non linéarités de matériau (plasticité ou de géométrie (grands déplacements)) [R5.05.05]. La syntaxe de cette commande est très semblable à celle de l'opérateur STAT_NON_LINE [U4.51.03].

L'évolution dynamique est étudiée à partir d'un état initial, configuration de référence, qui peut être produit par une analyse quasi-statique (opérateur STAT_NON_LINE [U4.51.03]) ou dynamique antérieure (opérateur DYNA_NON_LINE).

L'évolution dynamique peut être étudiée en plusieurs travaux successifs, par une poursuite à partir d'un instant déjà calculé, si une base de données a été définie dans le profil d'étude de l'utilisateur.

Produit un concept de type evol_noli.

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

Table des Matières

1 But....1
2 Syntaxe....4
3 Opérandes....7
3.1 Mot clé RESULTAT....7
3.2 Opérandes MODELE/CHAM_MATER/CARA_ELEM....7
3.3 Opérandes MODE_STAT/MASS_DIAG....7
3.4 Mot clé EXCIT....7
3.4.1 Opérandes CHARGE/FONC_MULT....7
3.4.2 Opérande TYPE_CHARGE....7
3.4.3 Opérandes MULT_APPUI/ACCE/VITE/DEPL/DIRECTION/GROUP_NO....8
3.5 Mot clé CONTACT....8
3.6 Mot clé SOUS_STRUC....8
3.7 Mot-clé COMPORTEMENT....9
3.8 Mot clé ETAT_INIT....9
3.9 Mot clé INCREMENT....9
3.10 Mot clé NEWTON....9
3.11 Mot clé RECH_LINEAIRE....9
3.12 Mot clé SOLVEUR....9
3.13 Mot clé CONVERGENCE....10
3.14 Mot clé ARCHIVAGE....10
3.14.1 Opérande CHAM_EXCLU....10
3.15 Mot clé AMOR_RAYL_RIGI....10
3.16 Mot clé AMOR_MODAL....10
3.16.1 Opérandes MODE_MECA / AMOR_REDUIT / LIST_AMOR / NB_MODE....10
3.16.2 Opérande REAC_VITE....11
3.17 Description du schéma d'intégration en temps [bib2] [R5.05.05]....11
3.17.1 Cas SCHEMA = 'NEWMARK'....12
3.17.2 Cas SCHEMA = 'HHT'....12
3.17.3 Cas SCHEMA = 'DIFF_CENT'....12
3.17.4 Cas SCHEMA = 'TCHAMWA'....12
3.17.5 Opérande COEF_MASS_SHIFT....13
3.18 Mot-clé CRIT_STAB....13
3.19 Mot-clé MODE_VIBR....13
3.20 Opérande ENERGIE....14
3.21 Opérande PROJ_MODAL....14
3.21.1 Opérandes MODE_MECA, NB_MODE....15
3.21.2 Opérandes MASS_GENE , RIGI_GENE , AMOR_GENE....15
3.21.3 Opérandes DEPL_INIT_GENE , VITE_INIT_GENE , ACCE_INIT_GENE....15

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

3.22 Mot clé EXCIT_GENE....15
3.23 Opérande INFO....16
3.24 Opérande TITRE....16
4 Contenu de la table_container ANALYSE_MODALE....16
5 Exemple : mouvement d'un pendule de grande amplitude....17

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

2 Syntaxe

```
dynanl [evol_noli] = DYNA_NON_LINE
(  ◊  RESULTAT         =  dynanl,                        [evol_noli]
   ♦  MODELE           =  mo,                            [modele]
   ♦  CHAM_MATER       =  chmat,                         [cham_mater]
   ◊  MODE_STAT        =  modestat,                      [mode_meca]
   ◊  CARA_ELEM        =  carac,                         [cara_elem]
   ◊  MASS_DIAG        =  /'OUI',
                          /'NON',

   ◊  EXCIT            =_F(
                  ◊  TYPE_CHARGE  =  /'FIXE_CSTE',     [DEFAUT]
                                     /'SUIV',
                                     /'DIDI',
                  ◊  CHARGE       =  chi,              [char_meca]
                                                       [char_cine_meca]
                  ◊  /  FONC_MULT =  fi ,              [fonction]
                     /  DEPL      =  depl,             [fonction]
                        VITE      =  vite,             [fonction]
                        ACCE      =  acce,             [fonction]
                  ◊  MULT_APPUI   =  /'NON',           [DEFAUT]
                                     /'OUI',
                  ◊  DIRECTION    =  (d1, d2, d3),     [l_R]
                  ◊  GROUP_NO     =  lgrno,            [l_gr_noeud]
               ),

   ◊  SOUS_STRUC       =_F( voir le document [U4.51.03] ),

   ◊  AMOR_RAYL_RIGI   =  /'TANGENTE',                  [DEFAUT]
                          /'ELASTIQUE',

   ◊  AMOR_MODAL       =_F(
                  ♦  MODE_MECA    =  mode,             [mode_meca]
                  ♦  /AMOR_REDUIT =  l_amor,           [l_R]
                     /LIST_AMOR   =  lisamor           [listr8]
                  ◊  NB_MODE      =  nbmode,           [I]
                  ◊  REAC_VITE    =  /'OUI',           [DEFAUT]
                                     /'NON',
               ),

   ♦    |  COMPORTEMENT    =_F( voir le document [U4.51.11] ),


   ◊  ETAT_INIT        =_F( voir le document [U4.51.03]
                     |  VITE         =  vite,          [cham_no_DEPL_R]
                     |  ACCE         =  acce,          [cham_no_DEPL_R]
               ),

   ♦  INCREMENT        =_F( voir le document [U4.51.03])
   ◊  EXCIT_GENE       =_F(
                  ◊  FONC_MULT    =  fomult,      [fonction_sdaster]
                  ♦  VECT_GENE    =  vecgen,         [vect_asse_gene]
               ),

   ◊  NEWTON           =_F( voir le document [U4.51.03] ),
```


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

```
        ◊  RECH_LINEAIRE    =_F ( voir le document [U4.51.03] ),

        ◊  SOLVEUR      =  _F  ( voir le document [U4.50.01] ),

        ◊  CONVERGENCE  =  _F  ( voir le document [U4.51.03] ),
        ◊  MODE_VIBR =_F(
            ◊  OPTION          =  /' PLUS_PETITE ',              [DEFAUT]
                                  /'BANDE',
                                  /'CALIBRATION',
            Si OPTION='PLUS_PETITE'
            {
              ◊  NMAX_FREQ        =  /3,                        [DEFAUT]
                                     /nbfreq,                   [I]
            }
            Si OPTION='BANDE'
            {
              ◊  FREQ             =  intba  ,                   [listr8]
            }
            Si OPTION='CALIBRATION'
            {
              ◊  FREQ             =  intba  ,                   [listr8]
            }

           ◊  MATR_RIGI        =  /'ELASTIQUE',                [DEFAUT]
                                  /'TANGENTE',
                                  /'SECANTE',
           ◊  /LIST_INST       =  list_r8,                     [listr8]
              /INST            =  l_r8,                        [R]
              /PAS_CALC        =  npas,                        [I]
           ◊  PRECISION        =  /1.E-6,                      [DEFAUT]
                                  /prec,                       [R]
           ◊  CRITERE          =  /'RELATIF',                  [DEFAUT]
                                  /'ABSOLU' ,
           ),

        ◊  CRIT_STAB    = _F ( voir le document [U4.51.03] ),
        ◊  ENERGIE      = _F()

        ◊  ARCHIVAGE    = _F ( voir le document [U4.51.03]),

        ♦  SCHEMA_TEMPS =_F (
                        ♦ SCHEMA =  /   'NEWMARK',
                                    /   'HHT',
                                    /   'DIFF_CENT',
                                    /   'TCHAMWA',
                         ◊     COEF_MASS_SHIFT =  /   0.,       [DEFAUT]
                                                  /   coef,     [R]
    { Si SCHEMA      =   'NEWMARK'
                         ◊     BETA   =  /   0.25,              [DEFAUT]
                                         /   beta,              [R]
                         ◊     GAMMA  =  /   0.5,               [DEFAUT]
                                         /   gamm,              [R]
    }
    { Si SCHEMA      =   'HHT'
                         ◊     ALPHA  =  /   -0.3,              [DEFAUT]
                                         /   alph,              [R]
                         ◊     MODI_EQUI =     /   'OUI',
                                               /   'NON',       [DEFAUT]
```


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

```
      }
      { Si SCHEMA      =  'TCHAMWA'
                             ◊      PHI    =  /  1.05                [DEFAUT]
                                               /  phi,                [R]
      }
      { Si SCHEMA      =  /  'NEWMARK',
                          /  'HHT',
                          ♦  FORMULATION  =  /  'DEPLACEMENT',
                                             /  'VITESSE',
                                             /  'ACCELERATION',
      }
      { Si SCHEMA      =  /  'DIFF_CENT',
                          /  'TCHAMWA',
                             ♦  FORMULATION  =     'ACCELERATION',
                             ◊   STOP_CFL    =  /  'OUI',        [DEFAUT]
                                                /  'NON',
      }, ),

            ◊  OBSERVATION  = _F   ( voir le document [U4.51.03]),
            ◊  AFFICHAGE    = _F   ( voir le document [U4.51.03]),
            ◊  SCHEMA_THM   = _F   ( voir le document [U4.51.03]),

            ◊  PROJ_MODAL =_F   (
                       ♦  MODE_MECA =  mode,                [mode_meca]
                       ◊  NB_MODE      =  nbmode,           [I]
                       ◊  /MASS_GENE   =  massgen           [matr_asse_gene_R]
                           RIGI_GENE   =  rigigen           [matr_asse_gene_R]
                          /  AMOR_GENE =  amorgen           [matr_asse_gene_R]
                        ◊     DEPL_INIT_GENE = deplgen,     [ vect_asse_gene ]
                        ◊     VITE_INIT_GENE = vitegen,     [ vect_asse_gene ]
                        ◊     ACCE_INIT_GENE = accegen,     [ vect_asse_gene ]
                         ) ,

            ◊  INFO      =     / 1,                         [DEFAUT]
                             / 2,

            ◊  TITRE = tx   ,                               [Kn]   )
```


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

3 Opérandes

3.1 Mot clé RESULTAT

Dans le cas où l'on souhaite poursuivre un calcul déjà exécuté, RESULTAT indique quel objet est enrichi (voir aussi ETAT_INIT).

3.2 Opérandes MODELE/CHAM_MATER/CARA_ELEM

Ces opérandes ont la même signification que dans le document [U4.51.03].

3.3 Opérandes MODE_STAT/MASS_DIAG

◊ MODE_STAT = modestat

Nom du mode statique nécessaire dans le cas d'un calcul sismique avec excitations multi-appuis [R4.05.01].

◊ MASS_DIAG = / 'OUI',
/ 'NON',

Option à utiliser avec un schéma en temps explicite [bib2] et qui permet de résoudre avec une matrice de masse lumpée (diagonalisée). Cette option n'est pas disponible pour tous les types d'éléments, en particulier les discrets (dans ce cas, il faut résoudre avec la matrice de masse consistante). En implicite il n'est pas possible d'utiliser une matrice lumpée (émission d'une erreur fatale si l'utilisateur est dans ce cas)

3.4 Mot clé EXCIT

◊ EXCIT =_F

Ce mot clé facteur permet de décrire à chaque occurrence une charge (sollicitations et conditions aux limites), et éventuellement un coefficient multiplicateur et/ou un type de charge.

Les opérandes ont la même signification que dans le document [U4.51.03] mais il y a quelques spécificités liées à la dynamique.

Remarque importante pour les schémas en temps:

> *Si l'on impose des conditions aux limites en déplacement qui évoluent au cours du temps, il faut tenir compte de l'inconnue primale du schéma utilisé. Ces conditions sont en fait imposées en accélération en explicite (car c'est l'inconnue primale). Cela signifie que l'on doit entrer dans DYNA_NON_LINE la dérivée seconde du signal en déplacement que l'on veut imposer. Cette évolution du déplacement imposé doit donc être dérivable au moins deux fois en temps. De même pour le théta-schéma en vitesse, l'inconnue primale est la vitesse et l'on doit entrer dans DYNA_NON_LINE la dérivée première du signal en déplacement que l'on veut imposer.*

3.4.1 Opérandes CHARGE/FONC_MULT

♦ CHARGE = ch_i
◊ FONC_MULT = f_i

Les opérandes ont la même signification que dans le document [U4.51.03].

3.4.2 Opérande TYPE_CHARGE

◊ TYPE_CHARGE = / 'FIXE_CSTE' , [DEFAUT]
/ 'SUIV',

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

```
                    / 'DIDI',
```

L'opérande a la même signification que dans le document [U4.51.03], sauf qu'un chargement ne peut pas être piloté en dynamique, et donc tchi ne peut pas être FIXE_PILO.

3.4.3 Opérandes MULT_APPUI/ACCE/VITE/DEPL/DIRECTION/GROUP_NO

```
◊  MULT_APPUI=  /'NON',                    [DEFAUT]
                /'OUI',
♦  ACCE      =  ac,                        [fonction]
♦  VITE      =  vi,                        [fonction]
♦  DEPL      =  dp,                        [fonction]
◊  DIRECTION =  (dx,dy,dz,drx,dry,drz),    [l_R]
◊  GROUP_NO  =  lgrno,                     [l_groupe_no]
```

On précise que le calcul multi-appui repose fondamentalement sur la superposition additive des mouvements, cf. [R5.05.05, § 2.6], donc si les modes statiques, élastiques, sont des solutions engendrant des déformations dans la structure, alors pour que la superposition soit licite, il faut que le comportement des matériaux constitutifs soit linéaire, et que la cinématique le soit aussi.

Les mouvements d'entraînement fonctions du temps produisent donc des forces d'inertie qui se cumulent aux autres charges et la résolution produit une réponse dynamique en hypothèse appuis ou supports bloqués. Ces forces d'inertie d'entraînement sont construites avec l'opérateur CALC_CHAR_SEISME [U4.63.01] à partir des modes statiques et de la matrice de masse. Seule l'accélération d'entraînement en chaque appui ou support sert donc à la résolution, et les vitesses et déplacements en chaque appui ou support servent seulement à reconstruire par superposition additive la solution complète en vitesses et déplacements. On note que les forces d'amortissement ne proviennent que des vitesses résolues en hypothèse appuis ou supports bloqués, et non des vitesses complètes reconstruites par superposition avec les vitesses en chaque appui ou support.

Dans le cas d'une excitation multi-appuis (MULT_APPUI= 'OUI'), les autres opérandes ont exactement la même signification que dans le mot clé facteur EXCIT de l'opérateur DYNA_TRAN_MODAL [U4.53.21]. Dans ce cas, les champs 'DEPL', 'VITE', 'ACCE' correspondent respectivement aux déplacements, vitesses et accélérations, fonctions du temps, du mouvement d'entraînement sur les appuis ou supports spécifiés dans 'GROUP_NO' et pour la direction 'DIRECTION'. Les nouveaux champs 'DEPL_ABSOLU', 'VITE_ABSOLU', 'ACCE_ABSOLU' sont alors créés dans le concept de type evol_noli produit par l'opérateur DYNA_NON_LINE et correspondent respectivement aux déplacements, vitesses et accélérations du mouvement absolu, sommes du mouvement d'entraînement sur les appuis ou supports et du mouvement relatif par rapport à ce mouvement d'entraînement multi-appuis.

Les solutions élastiques linéaires associées aux mouvements d'entraînement unitaires sur les appuis ou supports, cf. [R5.05.05, § 2.6], sont décrites par les modes statiques MODE_STAT renseignés au mot-clé correspondant, voir § 3.3. Si cela n'est pas le cas, il faut alors décrire les excitations par des déplacements imposés fonctions du temps sur tous les appuis ou supports, et la résolution se fera en référentiel absolu.

3.5 Mot clé CONTACT

♦ CONTACT = contact

Ce mot clé simple permet d'activer la résolution du contact-frottement ou la prise en compte d'une liaison unilatérale. contact est un concept issu de l'opérateur DEFI_CONTACT [U4.44.11].
L'opérande a la même signification que dans le document [U4.51.03].

3.6 Mot clé SOUS_STRUC

◊ SOUS_STRUC=_F

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

Ce mot clé facteur permet de préciser quels sont les chargements à utiliser pour les sous-structures statiques qui font alors obligatoirement partie du modèle.
L'opérande a la même signification que dans le document [U4.51.03].

3.7 Mot-clé COMPORTEMENT

La syntaxe de ces mots-clés communs à plusieurs commandes est décrite dans le document [U4.51.11]. Toutes les relations de comportement supportées par STAT_NON_LINE sont disponibles également dans DYNA_NON_LINE, à condition que le calcul de la matrice de masse des éléments concernés soit prévu.

3.8 Mot clé ETAT_INIT

◊ ETAT_INIT =_F

Sous ce mot clé sont définies les conditions initiales du problème. Les opérandes du mot clé ETAT_INIT ont la même signification que dans le document [U4.51.03].

Remarque :

Dans le cas où l'utilisateur a spécifié que le concept résultat est réentrant, le mot-clé ETAT_INIT est obligatoire. Le concept enrichi est fourni par le mot-clé RESULTAT.

En dynamique, on peut définir en plus les champs de vitesse et d'accélération initiales.

♦ / |VITE = vite
/ | ACCE = acce

Si les mots clés EVOL_NOLI, DEPL, et VITE sont absents, on suppose que l'état initial est à déplacements, vitesses et contraintes nuls, et on calcule les accélérations correspondant au chargement à l'instant instini défini par l'opérande INST.

3.9 Mot clé INCREMENT

♦ INCREMENT =_F

Définit la liste des instants de calcul. Les opérandes du mot clé INCREMENT ont la même signification que dans le document [U4.51.03].

3.10 Mot clé NEWTON

◊ NEWTON =_F

Précise les caractéristiques de la méthode de résolution du problème incrémental non linéaire (méthode de Newton-Raphson). Les opérandes du mot clé NEWTON ont la même signification que dans le document [U4.51.03].

3.11 Mot clé RECH_LINEAIRE

◊ RECH_LINEAIRE =_F(
◊ METHODE = /'CORDE' [DEFAUT]
/'MIXTE')

Permet d'activer la recherche linéaire. Les opérandes du mot clé RECH_LINEAIRE ont la même signification que dans le document [U4.51.03], sauf que la méthode PILOTAGE n'existe pas.

3.12 Mot clé SOLVEUR

◊ SOLVEUR =_F
La syntaxe de ce mot clé commun à plusieurs commandes est décrite dans le document [U4.50.01].

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

3.13 Mot clé CONVERGENCE

◊ CONVERGENCE =_F

Ce mot clé décrit les paramètres permettant d'apprécier la convergence de la méthode de NEWTON utilisée pour résoudre le problème mécanique non linéaire. Les opérandes du mot clé CONVERGENCE ont la même signification que dans le document [U4.51.03].

3.14 Mot clé ARCHIVAGE

◊ ARCHIVAGE =_F

Permet d'archiver des ou certains résultats à tous ou certains instants du calcul.
En l'absence de ce mot clé tous les pas de temps sont archivés, y compris les instants de calculs nouvellement créés par redécoupage automatique du pas de temps. Les opérandes du mot clé ARCHIVAGE ont la même signification que dans le document [U4.51.03], sauf pour le mot-clef CHAM_EXCLU.

3.14.1 Opérande CHAM_EXCLU

```
◊ CHAM_EXCLU      =  |'DEPL'
                     |'VITE'
                     |'ACCE'
                     |'SIEF_ELGA'
                     |'VARI_ELGA'
```

Permet de préciser les champs qui ne seront pas archivés, excepté au dernier pas de temps.

3.15 Mot clé AMOR_RAYL_RIGI

```
◊ AMOR_RAYL_RIGI  =  /'TANGENTE',                [DEFAUT]
                     /'ELASTIQUE'
```

Ce mot clé permet de spécifier la matrice de raideur $\boldsymbol{K}$ qui sera utilisée pour construire l'amortissement de Rayleigh $\boldsymbol{C}=\alpha.\boldsymbol{K}+\beta.\boldsymbol{M}$.
Avec la valeur par défaut ('TANGENTE'), la matrice $\boldsymbol{K}$ sera la même que celle qui est utilisée pour le calcul des efforts internes. En choisissant la valeur 'ELASTIQUE', on force le calcul de l'amortissement de Rayleigh avec la matrice de raideur élastique.
Pour les lois adoucissantes ou de type GLRC on conseille d'utiliser la matrice élastique.

3.16 Mot clé AMOR_MODAL

◊ AMOR_MODAL =_F

Ce mot clé permet de prendre en compte un amortissement équivalent à de l'amortissement modal décomposé sur une base de modes pré-calculée sous forme de concept de type mode_meca. Cet amortissement est globalement pris en compte dans l'équation d'équilibre dynamique comme une force correctrice au second membre $-\boldsymbol{C}.\dot{\boldsymbol{X}}$.

3.16.1 Opérandes MODE_MECA / AMOR_REDUIT / LIST_AMOR / NB_MODE

```
♦ MODE_MECA       =  mode
♦ /  AMOR_REDUIT  =  l_amor,
  /  LIST_AMOR    =  lisamor
◊ NB_MODE         =  nbmode
```

Le concept mode de type mode_meca (entré par l'opérande MODE_MECA) représente la base de modes pré-calculée sur laquelle on décompose l'amortissement modal. Cette base doit impérativement avoir le même profil de numérotation que celui du système dynamique défini par les

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

paramètres du mot clé SOLVEUR [§3.12]. Il est possible de tronquer la base modale à un nombre de modes défini par NB_MODE. A défaut, on prend tous les modes de la base modale.

Les amortissements modaux sous forme réduite sont donnés par une liste de réels dont le nombre de termes est inférieur ou égal au nombre de modes pris en compte. Si le nombre de termes de la liste est strictement inférieur, on étend cette liste avec la valeur de son dernier terme jusqu'à ce que sa taille atteigne le nombre de modes calculés.

3.16.2 Opérande REAC_VITE

```
◊  REAC_VITE=    /'OUI',                    [DEFAUT]
                 /'NON',
```

Si sa valeur est 'OUI', on modifie la force correctrice d'amortissement modal à chaque itération interne de Newton définie dans le mot clé NEWTON [§9].
Si sa valeur est 'NON', on ne remet à jour ce terme qu'au début de chaque pas de temps.

3.17 Description du schéma d'intégration en temps [bib2] [R5.05.05]

```
♦  SCHEMA_TEMPS =_F()
   ◊  STOP_CFL  =  /'OUI',                 [DEFAUT]
                   /'NON',
   ◊  FORMULATION= /'DEPLACEMENT',
                   /'VITESSE',
                   /'ACCELERATION',
```

On peut utiliser les méthodes implicites de NEWMARK (mot clé SCHEMA='NEWMARK' ou accélération moyenne modifiée : SCHEMA='HHT' avec MODI_EQUI = 'NON'), ou de HILBER-HUGHES-TAYLOR (SCHEMA='HHT' avec MODI_EQUI = 'OUI').

Avec un schéma implicite, les résolutions en déplacement, vitesse ou accélération sont actuellement disponibles (mot clé FORMULATION = 'DEPLACEMENT', 'VITESSE' ou 'ACCELERATION').

Inversement, on peut choisir une méthode explicite de type différences centrées (mot clé SCHEMA='DIFF_CENT') ou un schéma dissipatif de type TCHAMWA (mot clé SCHEMA='TCHAMWA'). Avec un schéma explicite, on ne peut résoudre qu'en accélération (mot clé FORMULATION = 'ACCELERATION').

Les schémas explicites étant conditionnellement stables, il peut être utile de vérifier si le pas de temps donné en entrée du calcul respecte bien la condition de stabilité (condition CFL). Si STOP_CFL = 'OUI' (défaut), alors si la liste d'instants fournie par l'utilisateur comporte un ou plusieurs pas de temps supérieurs à la condition de stabilité, le calcul s'arrête en erreur fatale. Si STOP_CFL = 'NON', on émet une alarme et continue le calcul.
Dans tous les cas, le pas de temps critique est donné dans le fichier de messages pour information.
Le calcul de la CFL n'est pas programmé pour tous les éléments (en particulier les éléments discrets sont ignorés.); la CFL estimée par *Code_Aster* peut donc être plus grande (moins pénalisante) que la CFL réelle, avec les risques de divergence brutale qui en découlent.
En explicite, il est aussi recommandé d'utiliser une matrice de masse lumpée (diagonalisée) : ce que l'on peut obtenir avec le mot clé MASS_DIAG = 'OUI' [§7].

Remarques :

- Le choix *MASS_DIAG='NON' est déconseillé avec les coques DKT.*
- *Avec les éléments DKT/DKTG il est nécessaire de préciser dans AFFE_CARA_ELEM, sous le mot clé facteur COQUE, le mot clé simple INER_ROTA = 'OUI'. Sinon la matrice masse est singulière et le schéma explicite est inutilisable.*
- *En explicite, on déconseille l'usage des éléments finis quadratiques (qui peuvent générer des oscillations parasites sur les champs solutions).*

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

- *Pour les schémas multi-pas (Newmark avec `MODI_EQUI='OUI'` et théta-schéma), il est nécessaire de recalculer les efforts intérieurs à l'instant précédent. Ce mode de fonctionnement est problématique au premier instant d'une reprise de calcul. En effet, si on a des variables de commandes (qui dépendent donc de l'instant de calcul), il est nécessaire de récupérer la valeur de l'instant précédent. Ce n'est pas toujours possible, en particulier si on n'est pas en `reuse` et si `ETAT_INIT` se fait à partir de champs individuels (pas d'utilisation du mot-clef `ETAT_INIT/EVOL_NOLI`). Dans ce cas, on ne calcule pas cette contribution au pas précédent, ce qui peut éventuellement modifier légèrement la convergence de l'algorithme.*

3.17.1 Cas SCHEMA = 'NEWMARK'

```
◊  BETA      =  /0.25,                    [DEFAUT]
                /beta,                    [R]
◊  GAMMA     =  /0.5 ,                    [DEFAUT]
                /gamm,                    [R]
```

La méthode d'intégration en temps est celle de NEWMARK, avec les valeurs données des paramètres beta et gamm.
Quand on ne précise ni beta, ni gamm, on a la méthode dite « règle du trapèze » ($beta=0.25$; $gamm=0.5$) qui, en linéaire, est inconditionnellement stable et n'apporte aucune dissipation parasite (*i.e.* amortissement numérique), mais qui, en non linéaire, peut être instable [bib1][bib2].

3.17.2 Cas SCHEMA = 'HHT'

```
◊  ALPHA     =  /-0.3,                    [DEFAUT]
                /alph,                    [R]
◊  MODI_EQUI =  /'OUI',
                /'NON',                   [DEFAUT]
```

Pour MODI_EQUI = 'NON' (valeur par défaut), la méthode d'intégration en temps (schéma d'intégration implicite) est celle de l'accélération moyenne modifiée (de la famille de Newmark) ([bib1], [bib2]), avec la valeur négative de alph donnée. Plus |alph| est grand, plus l'amortissement numérique apporté par le calcul est important. Mais cette dissipation est parfois nécessaire, en non linéaire, pour assurer la stabilité (à moins d'affecter un amortissement par matériau à la structure).
Pour MODI_EQUI = 'OUI', la méthode d'intégration en temps (schéma d'intégration implicite) est celle de HILBER-HUGHES-TAYLOR (HHT ou $\alpha-\text{méthode}$) [bib2], avec la valeur négative de alph donnée. Plus |alph| est grand, plus l'amortissement numérique apporté par le calcul est important. Par rapport au schéma précédent (MODI_EQUI = 'NON') d'accélération moyenne modifiée, l'amortissement numérique induit est plus « sélectif » : il est plus faible aux basses et moyennes fréquences (asymptotiquement nul à fréquence nulle) et il va croître plus vite quand la fréquence devient grande.
Ce deuxième schéma se base sur le premier avec, en plus, une modification de l'équation d'équilibre (on décale dans le temps les efforts intérieurs et extérieurs) [bib2].

3.17.3 Cas SCHEMA = 'DIFF_CENT'

Le schéma des différences centrées est un schéma explicite d'ordre deux de la famille de Newmark, de paramètres BETA = 0 et GAMMA = 0.5. Il s'agit d'un schéma à un pas qui ne présente pas de dissipation numérique.

3.17.4 Cas SCHEMA = 'TCHAMWA'

```
◊  PHI       =  /1.05,                    [DEFAUT]
                /phi,                     [R]
```

Une alternative aux schémas des différences centrées est le schéma développé par Bertrand Tchamwa et Christian Wielgosz.

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

Ce schéma explicite a plusieurs particularités intéressantes. Ce n'est pas un dérivé de Newmark, et la variation de son paramètre PHI permet une dissipation numérique contrôlable des hautes fréquences. Lorsqu'il vaut 1, la dissipation est nulle. Pour ne pas trop dégrader la condition de Courant et conserver des propriétés de stabilité comparables au schéma des différences centrées, il est recommandé de ne pas choisir un PHI supérieur à 1.10. 1.05 est la valeur choisie par défaut.

3.17.5 Opérande COEF_MASS_SHIFT

```
◊   COEF_MASS_SHIFT =  /0.                    [DEFAUT]
                       /coef
```

La donnée du coefficient coef permet de réaliser un shift de la matrice de masse M qui devient:
$M' = M + coef\ K$
La valeur de ce coefficient, par défaut nulle, doit être non nulle pour pouvoir inverser en dynamique avec schéma explicite la matrice de masse quand celle-ci a des termes nuls pour certains degrés de liberté spécifiques, par exemple la pression pour les éléments de modélisation HM.
L'entrée de ce coefficient permet également d'améliorer fortement la convergence en dynamique avec schéma implicite dans ce même type de modélisation en imposant une fréquence de coupure inversement proportionnelle à la valeur de coef (au prix d'une légère distorsion de l'ensemble des fréquences propres du système)

3.18 Mot-clé CRIT_STAB

```
◊   CRIT_STAB =_F
```

Ce mot-clé permet de déclencher le calcul, à la fin de chaque incrément de temps, d'un critère de stabilité, identique à ce qui est proposé dans STAT_NON_LINE . Ce critère est utile pour déceler, au cours du chargement, le point à partir duquel on perd la stabilité (par flambage par exemple). Il ne s'agit pas d'un critère de stabilité au sens dynamique (amortissement négatif).Les opérandes du mot clé CRIT_STAB ont la même signification que dans le document [U4.51.03].
Il convient cependant de signaler que l'utilisation de CRIT_STAB sur des modèles fluide-structure couplés (formulation (u, p, ϕ), *cf.* documentation [R4.02.02], qui sont disponibles avec DYNA_NON_LINE mais pas STAT_NON_LINE) oblige à exclure les degrés de liberté fluide car la matrice de rigidité assemblée globale est singulière pour ces degrés de liberté. Pour cela, il faut combiner les mot-clés DDL_EXCLUS (en excluant tous les degrés de liberté spécifiques aux éléments fluides, comme 'PRES' , 'PHI' et, dans les cas avec surface libre, 'DH') et MODI_RIGI = 'OUI' . Plus de détails sont donnés dans les documentations [U4.51.03], [U2.06.11] et [U2.08.04].

L'ensemble des résultats est stocké dans une structure de données de type table_container de nom 'ANALYSE_MODALE' attachée à la structure de données evol_noli . Les modes de flambement (TYPE='FLAMBEMENT') sont stockés comme TYPE_MODE=' MODE_FLAMB ' dans la table tandis que les modes de stabilité (TYPE=' STABILITE ') y sont stockés comme TYPE_MODE=' MODE_STAB ' (voir § 4).

3.19 Mot-clé MODE_VIBR

```
◊  MODE_VIBR     =_F (
          ◊  OPTION          =  /' PLUS_PETITE ',          [DEFAUT]
                                /'BANDE',
                                /'CALIBRATION',
          Si OPTION='PLUS_PETITE'
          {
            ◊  NMAX_FREQ        =  /3,                   [DEFAUT]
                                   /nbfreq,              [I]
          }
          Si OPTION='BANDE'
          {
            ◊  FREQ             =  intba  ,              [listr8]
```


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

```
 }
 Si OPTION='CALIBRATION'
 {
   ◊  FREQ            =  intba  ,            [listr8]
 }
◊   MATR_RIGI       =  /'ELASTIQUE',         [DEFAUT]
                        /'TANGENTE',
                        /'SECANTE',
◊   /LIST_INST      =  list_r8,              [listr8]
   /INST            =  l_r8,                 [R]
   /PAS_CALC        =  npas,                 [I]
◊  PRECISION        =  /1.e-6                [DEFAUT]
                        /prec
◊  CRITERE          =  /'RELATIF',           [DEFAUT]
                         /'ABSOLU',
),
```

Ce mot-clé permet de déclencher le calcul, à la fin de chaque incrément de temps, d'une recherche de modes propres vibratoires.
Ce critère est utile pour suivre, au cours du calcul transitoire, l'évolution de la réponse vibratoire de la structure non linéaire.

Ce critère est calculé de la façon suivante : à la fin d'un pas de temps, on résout $det\left(\boldsymbol{K}-\omega^2.\boldsymbol{M}\right)=0$. K peut soit être la matrice de raideur élastique, soit la matrice tangente cohérente à l'instant courant, soit la matrice sécante. M est la matrice de masse. Cette analyse modale n'est autorisée que pour les matrices symétriques (masse et raideur).

Le mot-clé OPTION permet de choisir le type de recherche :

- OPTION='PLUS_PETITE' : les NMAX_FREQ vecteurs propres et valeurs propres correspondant aux plus petites valeurs propres ;
- OPTION='BANDE' : les vecteurs propres et valeurs propres pour les fréquences comprises dans la bande FREQ ;
- OPTION='CALIBRATION' dont juste le nombre de valeurs propres pour les fréquences comprises dans la bande FREQ.

L'ensemble des résultats est stocké dans une structure de données de type table_container de nom 'ANALYSE_MODALE' attachée à la structure de données evol_noli . Les modes vibratoires sont stockés comme TYPE_MODE='DEPL_VIBR' dans la table (voir § 4).

Les instants pour lesquels on veut faire un calcul de mode vibratoire sont donnés par une liste d'instants LIST_INST ou INST (list_r8 ou l_r8) ou par une fréquence PAS_CALC (tous les $npas$ de temps).

En l'absence de ces mots clés l'analyse modale vibratoire est réalisée à tous les pas de temps.
Les mots-clefs PRECISION et CRITERE permettent de sélectionner les instants, *cf.* [U4.71.00].

3.20 Opérande ENERGIE

◊ ENERGIE = _F()

Ce mot-clé permet d'activer le calcul du bilan d'énergie, son affichage en cours de calcul et son stockage dans la table de nom PARA_CALC. Le bilan d'énergie peut être extrait de cette table à l'aide de la commande RECU_TABLE [U4.71.02].

3.21 Opérande PROJ_MODAL

◊ PROJ_MODAL =_F

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

Ce mot clé permet de faire le calcul sur une base modale (ou de Ritz) préalablement calculée. Il est à utiliser avec un schéma d'intégration en temps explicite.

3.21.1 Opérandes MODE_MECA, NB_MODE

♦ MODE_MECA = mode, [mode_meca]

◊ NB_MODE = nbmode, [I]

On spécifie la base à utiliser (MODE_MECA) et le nombre de modes (NB_MODE).

Remarque importante :

La base modale doit s'appuyer sur une numérotation cohérente avec celle de l'évolution calculée (cf. [§10]) : même profil de numérotation.

3.21.2 Opérandes MASS_GENE , RIGI_GENE , AMOR_GENE

```
◊  /  MASS_GENE = massgen,      [ matr_asse_gene_R ]
      RIGI_GENE = rigigen,      [ matr_asse_gene_R ]
      AMOR_GENE = amorgen,      [ matr_asse_gene_R ]
```

Ces opérandes sont utilisés ensemble dans le cas où l'on veut condenser dynamiquement une partie du modèle au comportement linéaire, en ne calculant strictement par DYNA_NON_LINE que des domaines au comportement non-inéaire. Ceci, afin de réduire la taille du modèle de calcul. Dans ce cas, il est nécessaire de calculer une base modale de Ritz sur l'ensemble des domaines : le domaine non linéaire modélisé pour le calcul faisant appel à DYNA_NON_LINE et les autres domaines linéaires condensés dynamiquement. Cette base doit être orthogonalisée par rapport à la masse et à une rigidité linéaire de l'ensemble des domaines. Elle doit simplement être représentative des mouvements activant l'ensemble des domaines. Par contre, on ne renseignera derrière MODE_MECA que les modes obtenus par réduction de la base de Ritz au modèle de calcul traité par DYNA_NON_LINE. Un exemple de calcul est fourni par le cas test SDNV107A [V5.03.107].

L'opérande MASS_GENE permet d'entrer la projection de la matrice masse de l'ensemble des domaines sur la base de Ritz avec un stockage diagonal. L'opérande RIGI_GENE permet d'entrer la projection de la matrice rigidité des domaines linéaires condensés seuls sur la base de Ritz avec un stockage plein. L'opérande AMOR_GENE permet d'entrer éventuellement la projection d'une matrice d'amortissement (si elle existe) des domaines linéaires condensés seuls sur la base de Ritz avec un stockage plein.

3.21.3 Opérandes DEPL_INIT_GENE , VITE_INIT_GENE , ACCE_INIT_GENE

```
◊     DEPL_INIT_GENE = deplgen,     [ vect_asse_gene ]
◊     VITE_INIT_GENE = vitegen,     [ vect_asse_gene ]
◊     ACCE_INIT_GENE = accegen,     [ vect_asse_gene ]
```

Ces opérandes sont associés à l'utilisation des opérandes MASS_GENE, RIGI_GENE et éventuellement AMOR_GENE dans le mot clé PROJ_MODAL. Il servent à introduire un vecteur généralisé issu de la projection par **PROJ_VECT_BASE** (TYPE='DEPL') d'un champ déplacement ou vitesse ou accélération du modèle complet (y compris le domaine au comportement linéaire) sur la base modale de ce modèle complet. Ce vecteur généralisé fera office de condition initiale de l'évolution des coordonnées généralisées du calcul sur le modèle réduit au domaine non linéaire. Il faudra alors également renseigner les opérandes du mot-clé ETAT_INIT avec des champs correspondants déplacement, vitesse, accélération, contrainte, variable interne du modèle réduit. Un exemple de calcul est fourni par le cas test SDNV107C [V5.03.107].

3.22 Mot clé EXCIT_GENE

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

```
◊  EXCIT_GENE =_F(
   ◊   FONC_MULT =  fomult,       [fonction_sdaster]
   ♦    VECT_GENE =  vecgen,      [vect_asse_gene]
   )
```

Ce mot clé répétable est associé à l'utilisation des opérandes MASS_GENE, RIGI_GENE et éventuellement AMOR_GENE dans le mot clé PROJ_MODAL. Il sert à introduire les forces appliquées sur des domaines de comportement linéaire condensés dynamiquement et non modélisés dans le calcul faisant appel à un schéma d'intégration en temps explicite. Ces forces sont projetées sur la base de Ritz calculée sur l'ensemble des domaines.

VECT_GENE sert à renseigner les vecteurs de force projetés sur la base de Ritz. FONC_MULT sert à renseigner la fonction multiplicatrice dépendant du temps associée à chaque vecteur au sein d'une occurrence du mot clé EXCIT_GENE.

3.23 Opérande INFO

◊ INFO = inf

Permet d'effectuer dans le fichier message diverses impressions intermédiaires.
D'autres impressions sont faites systématiquement lors du calcul non linéaire, indépendamment de la valeur affectée au mot-clé INFO : ce sont les impressions des résidus et des incréments relatifs de déplacement au cours des itérations de Newton.
Attention, les fichiers .mess peuvent devenir très importants avec INFO = 2.

3.24 Opérande TITRE

◊ TITRE = tx

tx est le titre du calcul. Il sera imprimé en tête des résultats. Voir [U4.03.01].

4 Contenu de la table_container ANALYSE_MODALE

Cette table contien t les résultats de l'analyse modale réalisée par les mot_clefs facteurs MODE_VIBR et CRIT_STAB . Elle contient les colonnes suivantes :

NUME_INST	I	Numéro de l'instant de calcul
INST	R	Valeur de l'instant de calcul
NB_MODE	I	Nombre de modes calculés
NUME_MODE	I	Numéro d'ordre du mode calculé
TYPE_MODE	K16	Type de mode : DEPL_VIBR, MODE_FLAMB ou MODE_STAB
FREQ	R	Valeur propre pour TYPE_MODE='DEPL_VIBR'
CHAR_CRIT	R	Valeur propre pour TYPE_MODE='MODE_FLAMB'
CHAR_STAB	R	Valeur propre pour TYPE_MODE='MODE_STAB'
TYPE_OBJET	K16	La table ne contient que des cham_no
NOM_SD	K24	Nom de la structure de données JEVEUX correspondant au cham_no

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

5 Exemple : mouvement d'un pendule de grande amplitude

```
#   TITRE Pendule simple en grande oscillation
#
#   PENDULE CONSTITUE D'UN ELEMENT DE CABLE (test SDNL100A).
#
RESU=DYNA_NON_LINE( MODELE=MO, CHAM_MATER=CHMAT, CARA_ELEM=CARA,
                    EXCIT=( _F(  CHARGE = CHA1),
                            _F(  CHARGE = CHA2)),
                    INCREMENT=_F( INST_INIT = 0., LIST_INST = L_INST1),
                    ARCHIVAGE=_F( LIST_INST = L_INST2),
                    SCHEMA_TEMPS=_F( SCHEMA='NEWMARK',
                                     FORMULATION='DEPLACEMENT'),
                    COMPORTEMENT=_F( RELATION = 'CABLE',
                                  DEFORMATION = 'GREEN'),
                    CONVERGENCE=_F( RESI_GLOB_RELA = 1.E-6,
                                    ITER_GLOB_MAXI = 100),
                    NEWTON=_F( REAC_ITER = 1)
                      )
FIN()
```

- la charge `cha1` impose au `noeud 1` de rester fixe et au `noeud 2` de se déplacer dans le plan vertical `XZ`,
- la charge `cha2` est la pesanteur,
- la commande `DYNA_NON_LINE` spécifie que :
  - la méthode d'intégration du temps sera celle de `'NEWMARK'`, "règle du trapèze" (appelée aussi accélération moyenne), car il n'y a aucun argument sous `'NEWMARK'`,
  - l'état initial, à l'instant 0, est à déplacement nul, c'est-à-dire que les déplacements seront évalués à partir de la position initiale, et à vitesse nulle,
  - le calcul itératif se poursuivra tant que le résidu relatif sera $>10^{-2}$, mais le nombre des itérations sera limité à 100,
  - enfin la matrice tangente du système linéaire à résoudre sera réévaluée à chaque itération (par défaut puisque le mot clé `NEWTON` est absent).

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