

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`.

Table des matières

1But.....	1
2Syntaxe.....	4
3Opérandes.....	10
3.1Opérandes MODELE / CHAM_MATER / CARA_ELEM / MODE_STAT / MASS_DIAG.....	10
3.2Mot clé EXCIT.....	10
3.2.1Opérandes CHARGE / FONC_MULT.....	11
3.2.2Opérande TYPE_CHARGE.....	11
3.2.3Opérandes MULT_APPUI /ACCE /VITE /DEPL /DIRECTION /NOEUD /GROUP_NO.....	11
3.3Mot clé SOUS_STRUC.....	12
3.3.1Opérande CAS_CHARGE.....	12
3.3.2Opérandes TOUT / SUPER_MAILLE.....	12
3.3.3Opérande FONC_MULT.....	12
3.4Mot clé COMP_INCR.....	12
3.5Mot clé COMP_ELAS.....	12
3.6Mot clé ETAT_INIT.....	13
3.7Mot clé INCREMENT.....	13
3.8Mot clé NEWTON.....	13
3.9Mot clé SOLVEUR.....	13
3.10Mot clé CONVERGENCE.....	13
3.11Mot clé ARCHIVAGE.....	13
3.12Mot clé AMOR_MODAL.....	13
3.12.1Opérandes MODE_MECA / AMOR_REDUIT / LIST_AMOR / NB_MODE.....	14
3.12.2Opérande REAC_VITE.....	14
3.13Mot clé OBSERVATION.....	14
3.13.1Opérande SUIVI_DDL.....	14
3.13.2Opérandes LIST_ARCH / LIST_INST / INST / PAS_OBSE.....	14
3.13.3Opérandes NOM_CHAM / NOM_CMP.....	14
3.13.4Opérandes NOEUD / GROUP_NO.....	14
3.13.5Opérandes MAILLE / POINT.....	14
3.13.6Opérande NUME_SUIVI.....	15
3.13.7Opérandes VALE_MIN / VALE_MAX.....	15
3.13.8Opérandes PRECISION / CRITERE.....	15
3.14Description du schéma d'intégration en temps [bib2] [R5.05.05].....	15
3.14.1Mot clé NEWMARK.....	16
3.14.2Mot clé HHT.....	16
3.14.3Mot clé THETA_METHODE.....	16
3.14.4Mot clé DIFF_CENT.....	17
3.14.5Mot clé TCHAMWA.....	17
3.15Mot clé AFFICHAGE.....	17

3.15.1Opérande UNITE.....	17
3.15.2Opérande NOM_COLONNE.....	18
3.15.3Opérande INFO_RESIDU.....	19
3.15.4Opérandes LONG_R, PREC_R et LONG_I.....	19
3.16Mot-clé CRIT_FLAMB.....	20
3.17Mot-clé MODE_VIBR.....	21
3.18Opérandes SENSIBILITE.....	21
3.19Opérande PROJ_MODAL.....	21
3.19.1Opérandes MODE_MECA, NB_MODE.....	22
3.19.2Opérandes MASS_GENE , RIGI_GENE , AMOR_GENE	22
3.20Mot clé EXCIT_GENE.....	22
3.21Opérande INFO.....	23
3.22Opérande TITRE.....	23
4Exemple : mouvement d'un pendule de grande amplitude.....	24
5Bibliographie.....	25

2 Syntaxe

```
dynanl [evol_noli] = DYNA_NON_LINE

(  ◇ reuse           = dynanl,                [evol_noli]
    ◆ MODELE         = mo,                    [modele]
    ◆ CHAM_MATER     = chmat,                 [cham_mater]
    ◇ MODE_STAT      = modestat,              [mode_stat_depl]
                                           [mode_stat_acce]
                                           [mode_stat_forc]
    ◇ CARA_ELEM      = carac,                [cara_elem]
    ◇ MASS_DIAG      = / 'OUI',
                                           / 'NON',

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

    ◇ SOUS_STRUC =_F (
        ◆ CAS_CHARGE          = nocas,            [K8]
        ◆ / TOUT               = 'OUI',
        / SUPER_MAILLE = lmail,                    [l_maille]
    ◇ FONC_MULT              = fi,                [fonction]
    ),

    ◇ AMOR_MODAL =_F (
        ◆ MODE_MECA = mode,                        [mode_meca]
        ◆ /AMOR_REDUIT = l_amor,                    [l_R]
        /LIST_AMOR     = lisamor                    [listr8]
    ◇ NB_MODE         = / nbmode,                    [I]
                                           / 9999,    [DEFAULT]
    ◇ REAC_VITE       = / 'OUI',                    [DEFAULT]
                                           / 'NON',

    ),

    ◆ | COMP_INCR =_F ( voir le document [U4.51.11] ),
      | COMP_ELAS =_F ( voir le document [U4.51.11] ),
```

```

    ◇ ETAT_INIT = _F (
        ◆ / | SIGM = sig, [cham_elem_SIEF_R]
                                [carte_SIEF_R]
        | VARI = vain, [cham_elem_VARI_R]
        | DEPL = depl, [cham_no_DEPL_R]
        | VITE = vite, [cham_no_DEPL_R]
        / EVOL_NOLI = evol, [evol_noli]
        ◇ / NUME_ORDRE = nuini, [I]
        / INST = instini, [R]
        ◇ CRITERE = / 'RELATIF', [DEFAULT]
        / 'ABSOLU' ,
{ b_prec_rela CRITERE = 'RELATIF'
        ◇ PRECISION = / 1.0E-6, [DEFAULT]
        / prec, [R]
}
{ b_prec_abso CRITERE = 'ABSOLU'
        ◆ PRECISION = prec, [R]
},
        ◇ NUME_DIDI = nudidi, [I]
    ◇ INST_ETAT_INIT = istetaini, [R]
    ),

    ◆ INCREMENT = _F (
        ◆ LIST_INST = litps, [listr8]
        ◇ / NUME_INST_INIT = nuini, [I]
        / INST_INIT = instini, [R]
        ◇ / NUME_INST_FIN = nufin, [I]
        / INST_FIN = instfin, [R]
        ◇ PRECISION = / 1.0E-6, [DEFAULT]
        / prec, [R]
        ◇ SUBD_METHODE = / 'UNIFORME', [DEFAULT]
        / 'AUCUNE',
        / 'EXTRAPOLE',
{ b_subd_unif SUBD_METHODE = 'UNIFORME'
        ◇ SUBD_PAS = / 4, [DEFAULT]
        / subpas, [I]
        ◇ SUBD_COEF_PAS_1 = / 1., [DEFAULT]
        / coefsub, [R]
        ◇ / SUBD_PAS_MINI = submini, [R]
        / SUBD_NIVEAU = 3, [DEFAULT]
        subniv, [I]
}
{ b_subd_extr SUBD_METHODE = 'EXTRAPOLE'
        ◇ SUBD_OPTION = / 'IGNORE_PREMIERES', [DEFAULT]
        / 'GARDE_DERNIERES',
        ◇ SUBD_ITER_IGNO = / 3, [DEFAULT]
        / subigno, [I]
        ◇ SUBD_ITER_FIN = / 8, [DEFAULT]
        / subfin, [I]
        ◇ SUBD_PAS = / 4, [DEFAULT]
        / subpas, [I]
        ◇ / SUBD_PAS_MINI = submini, [R]
        / SUBD_NIVEAU = subniv, [I]
        ◇ SUBD_ITER_PLUS = / 50, [DEFAULT]
        / subplus, [I]
},),

```

```

    ◇ EXCIT_GENE =_F (
        ◇ FONC_MULT = fomult, [fonction_sdaster]
        ◆ VECT_GENE = vecgen, [vect_asse_gene]
    ),

    ◇ NEWTON =_F (
        ◇ PREDICTION = / 'TANGENTE' , [DEFAULT]
                        / 'ELASTIQUE',
        ◇ MATRICE = / 'TANGENTE', [DEFAULT]
                  / 'ELASTIQUE',
        ◇ REAC_INCR = / 1, [DEFAULT]
                  / mf, [I]
        ◇ REAC_ITER = / 0, [DEFAULT]
                  / it, [I]
        ◇ REAC_ITER_ELAS = / 0, [DEFAULT]
                  / it, [I]
        ◇ PAS_MINI_ELAS= pasmini, [I]
    ),

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

    ◇ CONVERGENCE =_F (
        ◇ / RESI_GLOB_RELA = 1.E-6 , [DEFAULT]
          / | RESI_GLOB_MAXI = resmax , [R]
            | RESI_GLOB_RELA = resrel , [R]
            | RESI_REFE_RELA = resref, [R]
        ◇ SIGM_REFE = sigref, [R]
        ◇ EPSI_REFE = sigref, [R]
        ◇ FLUX_THER_REFE = sigref, [R]
        ◇ FLUX_HYD1_REFE = sigref, [R]
        ◇ FLUX_HYD2_REFE = sigref, [R]
        ◇ ITER_GLOB_ELAS = / 25, [DEFAULT]
                  / maxelas, [I]
        ◇ ITER_GLOB_MAXI = / 10, [DEFAULT]
                  / maglob, [I]
        ◇ ARRET = / 'OUI', [DEFAULT]
                  / 'NON',
        ◇ TYPE = / 'PIC', [DEFAULT]
                  / 'PLATEAU',
    { b_plateau TYPE = 'PLATEAU'
        ◇ PLATEAU_ITER = / 3, [DEFAULT]
                  / platit, [I]
        ◇ PLATEAU_RELA = / 1,E-6, [DEFAULT]
                  / platrel, [R]
    }, ),

    ◇ CRIT_FLAMB =_F (
        ◇ NB_FREQ = / 3, [DEFAULT]
                  / nbfreq, [I]
        ◇ CHAR_CRIT = / (-10,10), [DEFAULT]
                  / intcc,
        ◇ INST_CALCUL = / 'LISTE_ARCHIVAGE', [DEFAULT]
                  / 'TOUT_PAS',
    ),
```

```

◇  MODE_VIBR =_F (
  ◇  NB_FREQ      = / 3,                                [DEFAULT]
                                     / nbfreq,            [I]
  ◇  MATR_RIGI     = / 'ELASTIQUE',                      [DEFAULT]
                                     / 'TANGENTE',
                                     / 'SECANTE',
  ◇  BANDE         = intba ,                               [listr8]
  ◇  INST_CALCUL   = / 'LISTE_ARCHIVAGE',                [DEFAULT]
                                     / 'TOUT_PAS',
),

◇  SENSIBILITE ( voir le document [U4.50.02] ),

◇  ARCHIVAGE =_F (
  ◇  / LIST_INST = list_r8,                                [listr8]
  ◇  / INST      = l_r8,                                    [R]
  ◇  / PAS_ARCH  = npas,                                    [I]
  ◇  PRECISION   = / 1.E-6,                                [DEFAULT]
                                     / prec ,               [R]
  ◇  / ARCH_ETAT_INIT = 'OUI',
  ◇  / NUME_INIT    = nuinit,                              [I]
  ◇  DETR_NUME_SUIV = 'OUI',
  ◇  CHAM_EXCLU    = | 'DEPL',
                     | 'VITE',
                     | 'ACCE',
                     | 'SIEF_ELGA',
                     | 'VARI_ELGA',
                     | 'LANL_ELGA',
),

◆  SCHEMA_TEMPS =_F (
  ◆  SCHEMA = / 'NEWMARK',
               / 'HHT',
               / 'THETA_METHODE',
               / 'DIFF_CENT',
               / 'TCHAMWA',

{ b_newmark          SCHEMA = 'NEWMARK'
  ◇  BETA = / 0.25,                                [DEFAULT]
               / beta,                                [R]
  ◇  GAMMA = / 0.5,                                [DEFAULT]
               / gamm,                                [R]
}

{ b_hht              SCHEMA = 'HHT'
  ◇  ALPHA = / -0.3,                                [DEFAULT]
               / alph,                                [R]
  ◇  MODI_EQUI = / 'OUI',
               / 'NON',                                [DEFAULT]
}

{ b_theta            SCHEMA = 'THETA_METHODE'
  ◇  THETA = / 1.,                                [DEFAULT]
               / theta,                                [R]
}

{ b_tchamwa          SCHEMA = 'TCHAMWA'
  ◇  PHI = / 1.05                                [DEFAULT]
               / phi,                                [R]
}

```

```
{ b_implicite          SCHEMA      = / 'NEWMARK',
                        / 'HHT',
                        / 'THETA_METHODE',
                        ♦ FORMULATION = / 'DEPLACEMENT',
                        / 'VITESSE',
                        / 'ACCELERATION',
}

{ b_explicite          SCHEMA      = / 'DIFF_CENT',
                        / 'TCHAMWA',
                        ♦ FORMULATION = 'ACCELERATION',
                        ◇ STOP_CFL   = / 'OUI',          [DEFAULT]
                        / 'NON',
}, ),

    ◇ OBSERVATION =_F (
        ♦ NOM_CMP   = lnocmp ,          [l_Kn]
        ◇ SUIVI_DDL = / 'OUI',
        / 'NON',          [DEFAULT]

{ b_suiwi              SUIVI_DDL   = 'OUI'
        ♦ / | NOEUD      = lno ,          [l_noeud]
        | GROUP_NO      = lgmo ,          [l_gr_noeud]
        / MAILLE = lma ,          [l_maille]
        POINT          = lpoint ,        [l_I]
        ♦ NOM_CHAM      = | 'DEPL',
        | 'VITE',
        | 'ACCE',
        | 'DEPL_ABSOLU',
        | 'VITE_ABSOLU',
        | 'ACCE_ABSOLU',
        | 'SIEF_ELGA',
        | 'VARI_ELGA',
        | 'FORC_NODA',
        ◇ GROUP_MA      = lgrma ,          [l_gr_maille]
        ♦ NUME_SUIVI    = nsui          [I]
        ◇ VALE_MAX      = 'OUI',
        ◇ VALE_MIN      = 'OUI',
}

{ b_non_suiwi          SUIVI_DDL   = 'NON'
        ♦ / | NOEUD      = lno ,          [l_noeud]
        | GROUP_NO      = lgmo ,          [l_gr_noeud]
        / MAILLE = lma ,          [l_maille]
        POINT          = lpoint ,        [l_I]
        ♦ NOM_CHAM      = | 'DEPL',
        | 'VITE',
        | 'ACCE',
        | 'DEPL_ABSOLU',
        | 'VITE_ABSOLU',
        | 'ACCE_ABSOLU',
        | 'SIEF_ELGA',
        | 'VARI_ELGA',
        ♦ / LIST_ARCH   = larch ,          [listis]
        / LIST_INST     = linst ,          [listr8]
        / INST          = linst ,          [l_R]
        / PAS_OBSE      = pas ,          [I]
        ◇ CRITERE       = / 'RELATIF', [DEFAULT]
        / 'ABSOLU' ,

{ b_prec_rela          CRITERE     = 'RELATIF'
        ◇ PRECISION    = / 1.0E-6,      [DEFAULT]
        / prec ,          [R]
```



```

}
{ b_prec_abso          CRITERE = 'ABSOLU'
  ♦ PRECISION =          prec,          [R]
},),

♦ AFFICHAGE =_F (
    ♦ / LIST_INST      = list_r8,          [listr8]
      / INST           = l_r8,            [R]
      / PAS_ARCH      = npas,            [I]

    ♦ UNITE            = unite            [I]
    ♦ LONG_R           = / 12              [DEFAULT]
                          / long_r        [I]
    ♦ PREC_R           = / 5               [DEFAULT]
                          / prec_r        [I]
    ♦ LONG_I           = / 6               [DEFAULT]
                          / long_i        [I]
    ♦ NOM_COLONNE      = | 'STANDARD',    [DEFAULT]
                          | 'MINIMUM',
                          | 'ITER_NEWT',
                          | 'INCR_TPS',
                          | 'RESI_RELA',
                          | 'RELA_NOEU',
                          | 'RESI_MAXI',
                          | 'MAXI_NOEU',
                          | 'RESI_REFE',
                          | 'REFE_NOEU',
                          | 'RELI_ITER',
                          | 'RELI_COEF',
                          | 'PILO_PARA',
                          | 'MATR_ASSE',
                          | 'ITER_DEBO',
                          | 'CTCD_ITER',
                          | 'CTCD_GEOM',
                          | 'CTCD_NOEU',
                          | 'CTCC_CONT',
                          | 'CTCC_FROT',
                          | 'CTCC_GEOM',
                          | 'ITER_FETI',
                          | 'SUIV_1',
                          | 'SUIV_2',
                          | 'SUIV_3',
                          | 'SUIV_4'

    ♦ INFO_RESIDU      = 'NON',            [DEFAULT]
                          'OUI'

),

♦ PROJ_MODAL =_F (
    ♦ MODE_MECA = mode,                    [mode_meca]
                                          [base_modale]
    ♦ NB_MODE      = / nbmode,            [I]
                          / 9999,          [DEFAULT]
    ♦ / MASS_GENE = massgen                [ matr_asse_gene_R

]
    RIGI_GENE      = rigigen                [
matr_asse_gene_R ]
    / AMOR_GENE     = amorgen                [
matr_asse_gene_R ]
),

```

Titre : Opérateur DYNA_NON_LINE
Responsable :

Date : 20/04/2009 Page : 10/24
Clé : U4.53.01 Révision : 828

◇ INFO = / 1 , [DEFAULT]
/ 2 ,
◇ TITRE = tx , [Kn])

3 Opérandes

3.1 Opérandes `MODELE/CHAM_MATER/CARA_ELEM/MODE_STAT/` `MASS_DIAG`

◆ `MODELE = mo`

Nom du modèle dont les éléments font l'objet du calcul mécanique.

◆ `CHAM_MATER = chmat`

Nom du champ de matériau affecté sur le modèle `mo`.

◇ `CARA_ELEM = carac`

Nom des caractéristiques des éléments de coque, poutre, barre, câble, et éléments discrets affectés sur le modèle `mo`, si nécessaire.

◇ `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 est fortement recommandé d'utiliser la matrice de masse consistante (l'absence du mot clé `MASS_DIAG` correspond aussi à ce cas).

3.2 Mot clé `EXCIT`

◆ `EXCIT = _F`

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

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.

Attention :

Dans un calcul thermo-mécanique, si la température initiale est différente de la température de référence (donnée dans l'opérateur `AFFE_MATERIAU`), le champ de déformation associé à l'instant initial peut être incompatible et donc conduire à un état de contraintes et de variables internes associé non nul. Si l'on utilise une relation de comportement incrémentale (mot clé facteur `COMP_INCR`) et si on ne définit pas explicitement un état de contraintes et de variables internes initial (associé à un champ de température initiale différente de la température de référence), le champ de contraintes et de variables internes calculé au premier incrément ne tiendra compte que de la seule variation de température entre l'instant initial et le premier instant, et non des éventuelles contraintes de compatibilité associées à la température initiale. Pour prendre cet état initial en compte, il faut le donner explicitement, par exemple grâce aux mots clés `SIGM`, `DEPL`, `VARI` et `VARI_NON_LOCAL` dans `ETAT_INIT`.

Pour éviter de telles situations qui peuvent conduire à des erreurs de calculs, il vaut mieux commencer un calcul en considérant qu'il faut partir d'un état vierge.

Si on réalise un calcul en axisymétrie et que l'on impose des forces nodales, ces efforts doivent être divisés par $2.\pi$ (on travaille sur un secteur de 1 radian) par rapport aux chargements réels. De même, si l'on souhaite calculer la résultante des efforts, le résultat est à multiplier par $2.\pi$ pour avoir la résultante totale sur la structure complète. De même en contraintes planes ou en déformation plane, on travaille sur une épaisseur unité : les efforts (sur l'épaisseur) appliqués doivent être divisés par l'épaisseur, les efforts réels sont obtenus en multipliant par l'épaisseur les efforts « du calcul ».

3.2.1 Opérandes CHARGE / FONC_MULT

♦ CHARGE = ch_i

ch_i est le chargement mécanique (comportant éventuellement l'évolution d'un champ de température) précisé à la $i^{\text{ème}}$ occurrence de EXCIT.

◇ FONC_MULT : f_i

f_i est la fonction du temps multiplicatrice du chargement précisé à la $i^{\text{ème}}$ occurrence de EXCIT. Le chargement et les conditions aux limites pour n occurrences du mot clé facteur EXCIT sont :

$$ch = \sum_{i=1}^n f_i \cdot ch_i$$

Pour les conditions de Dirichlet, bien entendu, seule la valeur imposée est multipliée par f_i .

Par défaut : $f_i=1$.

Le champ de température n'est pas multiplié par f_i .

3.2.2 Opérande TYPE_CHARGE

◇ TYPE_CHARGE = $tchi$

Par défaut, $tchi$ vaut 'FIXE_CSTE' : cela correspond à un chargement appliqué sur la géométrie initiale et non piloté. Il peut cependant être une fonction, et en particulier dépendre du temps.

Si $tchi$ vaut 'SUIV', le chargement est dit "suiveur", c'est-à-dire qu'il dépend de la valeur des inconnues : par exemple, la pression, étant un chargement s'appliquant dans la direction normale à une structure, dépend de la géométrie actualisée de celle-ci, et donc des déplacements. Un chargement suiveur est réévalué à chaque itération de l'algorithme de résolution. Un chargement fixe n'est réévalué qu'à chaque nouvel instant, et seulement si ch_i dépend du temps (défini dans AFFE_CHAR_MECA_F et paramétré par l'instant).

Actuellement les chargements qui peuvent être qualifiés de 'SUIV' sont le chargement de pesanteur pour l'élément de CABLE_POULIE, la pression pour les modélisations 3D, 3D_SI, D_PLAN, D_PLAN_SI, AXIS, AXIS_SI, C_PLAN, C_PLAN_SI et pour toutes les modélisations THM (3D_HHM, 3D_HM, 3D_JOINT_CT, 3D_THH, 3D_THHM, 3D_THM, AXIS_HHM, AXIS_HM, AXIS_THH, AXIS_THHM, AXIS_THM, D_PLAN_HHM, D_PLAN_HM, D_PLAN_THH, D_PLAN_THHM, D_PLAN_THM) et la force centrifuge en grands déplacements (mot clé ROTATION dans AFFE_CHAR_MECA).

Si $tchi$ vaut 'DIDI' alors les conditions de DIRICHLET (déplacements imposés, conditions linéaires) s'appliqueront sur l'incrément de déplacement à partir de l'instant donné sous ETAT_INIT/NUME_DIDI (par défaut l'instant de reprise du calcul) et non sur le déplacement total. Par exemple pour un déplacement imposé (mot clé DDL_IMPO de AFFE_CHAR_MECA) la condition sera de la forme : $u - u_0 = d$ où u_0 est le déplacement défini par NUME_DIDI et non $u = d$.

3.2.3 Opérandes MULT_APPUI/ACCE/VITE/DEPL/DIRECTION/NOEUD/GROUP_NO

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 du mouvement relatif par rapport au mouvement d'entraînement multi-appuis. Les nouveaux champs 'DEPL_ABSOLU', 'VITE_ABSOLU', 'ACCE_ABSOLU' sont alors créés et correspondent respectivement aux déplacements, vitesses et accélérations du mouvement absolu, somme du mouvement d'entraînement multi-appuis et du mouvement relatif par rapport à ce mouvement d'entraînement multi-appuis.

3.3 Mot clé SOUS_STRUC

◇ SOUS_STRUC

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. En son absence, les chargements sur les sous structures sont nuls.

Ces chargements s'ajoutent aux chargements « éléments finis » qui peuvent être appliqués sur le reste du modèle.

3.3.1 Opérande CAS_CHARGE

◆ CAS_CHARGE = *nocas*

nocas est le nom du cas de charge à utiliser. Voir opérateur MACR_ELEM_STAT [U4.62.01].

3.3.2 Opérandes TOUT / SUPER_MAILLE

◆ / TOUT = 'OUI'

Ce mot clé permet d'affecter le chargement *nocas* à toutes les sous structures du modèle.

/ SUPER_MAILLE = *l_mail*

Ce mot clé facteur permet de n'affecter le chargement *nocas* qu'à certaines sous-structures.

3.3.3 Opérande FONC_MULT

◇ FONC_MULT = *fi*

f_i est la fonction du temps multiplicatrice du chargement précisé à la *i*^{ème} occurrence de SOUS_STRUCT.

Le comportement de ce mot clé est le même que pour son occurrence dans EXCIT.

3.4 Mots-clés COMP_INCR et COMP_ELAS

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.

3.5 Mot clé ETAT_INIT

◇ ETAT_INIT =_F

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

3.6 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].

3.7 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].

3.8 Mot clé SOLVEUR

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

3.9 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].

3.10 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].

3.11 Mot clé AMOR_MODAL

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 $-C \cdot \dot{X}$.

3.11.1 Opérands **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 paramètres du mot clé `SOLVEUR` [§3.9]. 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.

3.11.2 Opérande **REAC_VITE**

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` [§15].

Si sa valeur est 'NON', on ne remet à jour ce terme qu'au début de chaque pas de temps.

3.12 Mot clé **OBSERVATION**

Ce mot clé permet de post-traiter certains champs aux nœuds ou aux éléments sur des parties de modèle à des instants d'une liste (dite d'observation) généralement plus raffinée que la liste des instants archivés définie dans le mot clé `ARCHIVAGE` [§3.11] (où on stocke tous les champs sur tout le modèle). Il sert essentiellement à des économies de stockage.

Ce mot clé est répétable et permet la création d'une table d'observation de même nom que le concept résultat de `DYNA_NON_LINE`.

Il permet aussi de suivre l'évolution d'un ou plusieurs ddl, au travers du mot clé `SUIVI_DDL`.

3.12.1 Opérande **SUIVI_DDL**

Cet opérande permet de choisir entre le mode d'observation classique ou de faire du suivi de ddl.

3.12.2 Opérands **LIST_ARCH** / **LIST_INST** / **INST** / **PAS_OBSE**

Ces opérandes permettent de définir aux choix une liste d'instants d'observation. Ils ont la même signification que les opérandes de même nom servant à définir une liste d'archivage. `PAS_OBSE` jouant le même rôle que `PAS` dans `ARCHIVAGE` [§3.11].

3.12.3 Opérands **NOM_CHAM** / **NOM_CMP**

Ces opérandes permettent de définir les champs à post-traiter ainsi que leurs composantes données par leur nom (par `NOM_CMP`).

3.12.4 Opérands **NOEUD** / **GROUP_NO**

Ces opérandes permettent de définir les nœuds de post-traitement pour des champs aux nœuds ('DEPL', 'VITE', 'ACCE', 'DEPL_ABSOLU', 'VITE_ABSOLU', 'ACCE_ABSOLU').

3.12.5 Opérands **MAILLE** / **POINT**

Ces opérandes qui vont de pair permettent de définir les mailles de post-traitement et leurs points d'extraction pour des champs aux éléments ('SIEF_ELGA' ou 'VARI_ELGA').

3.12.6 Opérande NUME_SUIVI

En mode SUIVI_DDL, on peut choisir de suivre et d'afficher entre un et quatre ddl.

3.12.7 Opérandes VALE_MIN / VALE_MAX

En mode SUIVI_DDL, au lieu de suivre un ddl particulier, on peut demander à afficher la valeur minimale ou maximale d'une composante d'un champ donné.

3.12.8 Opérandes PRECISION / CRITERE

Ces paramètres permettent de gérer la précision de la sélection des instants pour l'observation. Pour plus de précisions, cf. document [U4.71.00].

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

On peut soit utiliser une méthode implicite de NEWMARK (mot clé NEWMARK ou accélération moyenne modifiée : HHT avec MODI_EQUI = 'NON'), de HILBER-HUGHES-TAYLOR (HHT avec MODI_EQUI = 'OUI') ou bien une THETA_METHODE. 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 DIFF_CENT) ou un schéma dissipatif de type TCHAMWA (mot clé 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' [§11].

Nota bene

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.

3.13.1 Mot clé NEWMARK

```
♦ / NEWMARK=_F(  
  ◇ BETA =/ 0.25 [DEFAULT]  
    / beta  
  ◇ GAMMA = / 0.5  
    / gamm [DEFAULT]  
)
```

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].

3.13.2 Mot clé HHT

```
/  HHT=_F(  
  
◇  ALPHA = / -0.3 [DEFAULT]  
           / alph [R]  
◇  MODI_EQUI = / 'OUI',  
               / 'NON', [DEFAULT]  
  
)
```

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 α -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].

3.13.3 Mot clé THETA_METHODE

```
/  THETA_METHODE =_F(  
  
◇  THETA = / 1., [DEFAULT]  
           / theta, [R]  
  
)
```

Le schéma d'intégration en temps est un θ -schéma implicite d'ordre 1, en vitesse. Dans le cas d'utilisation avec des charges de contact, on doit aussi faire appel à la méthode `CONTINUE` (`AFFE_CHAR_MECA / CONTACT / METHODE = 'CONTINUE'`) et la formulation en vitesse (`FORMULATION = 'VITESSE'`).

`theta` doit être compris entre 0,5 et 1 : 0,5 correspond à un minimum de dissipation numérique, 1 correspond à un maximum de dissipation numérique. `theta = 1` permet de retrouver le schéma d'Euler. Ce schéma est aussi utilisable avec une formulation en déplacement.

3.13.4 Mot clé DIFF_CENT

Le schéma des différences centrées est un schéma explicite d'ordre 2 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.

3.13.5 Mot clé TCHAMWA

```
/ THETA_METHODE = _F (  
  ◇ PHI = / 1.05 [DEFAULT]  
          / phi, [R]  
)
```

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

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.

3.14 Mot clé AFFICHAGE

Ce mot-clef facteur permet de personnaliser l'affichage du tableau de convergence dans `STAT_NON_LINE` ou `DYNA_NON_LINE`. Les opérandes du mot clé `AFFICHAGE` ont la même signification que dans le document [U4.51.03].

3.15 Mot-clé CRIT_FLAMB

```
◇ CRIT_FLAMB = _F (  
  ◇ NB_FREQ = / 3, [DEFAULT]  
              / nbfreq, [I]  
  ◇ CHAR_CRIT = / (-10,10), [DEFAULT]  
                / intcc,  
  ◇ INST_CALCUL = / 'LISTE_ARCHIVAGE', [DEFAULT]  
                  / 'TOUT_PAS',  
) ,
```

Ce mot-clé permet de déclencher le calcul, à la fin de chaque incrément de temps, d'un critère de stabilité. Ce critère est utile pour déceler, au cours du chargement, le point à partir duquel on perd la stabilité (par flambage par exemple).

Ce critère est calculé de la façon suivante : à la fin d'un pas de temps, en petites perturbations, on résout $\det(\mathbf{K}^T - \lambda \cdot \mathbf{K}^g) = 0$. \mathbf{K}^T est la matrice tangente cohérente à cet instant.

\mathbf{K}^g est la matrice de rigidité géométrique, calculée à partir du champ de contraintes à cet instant.

En pratique, le chargement est instable si $|\lambda| < 1$ (en fait $-1 < \lambda < 0$). On calcule les valeurs propres par la méthode de Sorensen (cf `MODE_ITER_SIMULT` [U4.52.03]). Ceci peut être assez coûteux pour les problèmes de grande taille.

Le mot-clé `CHAR_CRIT` permet de gagner du temps en ne faisant qu'un test de Sturm dans la bande de fréquence fournie. Si on trouve au moins une fréquence, alors on calcule réellement les valeurs des charges critiques dans cet intervalle.

Pour les grands déplacements et les grandes déformations `GREEN(_GR)` ou `SIMO_MIEHE`, on résout $\det(\mathbf{K}^T - \lambda \cdot \mathbf{I}) = 0$ car \mathbf{K}^T contient alors \mathbf{K}^g .

Le critère est alors un critère d'instabilité : quand λ change de signe (donc passe par 0) le chargement est instable.

Le mot-clé `NB_FREQ` (3 par défaut) désigne le nombre de charges critiques à calculer. En fait seule la première suffit mais il peut y avoir des modes multiples

On stocke le mode propre correspondant à la plus petite charge critique (en valeur absolue) dans la S.D. `RESULTAT`, sous le nom `MODE_FLAMB`. Ce mode propre peut être extrait et visualisé (comme un champ de déplacements ou un mode propre classique). Il est normalisé à 1 sur la plus grande composante de déplacement.

Le mot-clé `INST_CALCUL` précise à quelle occasion est calculé ce critère. Par défaut (`INST_CALCUL='LISTE_ARCHIVAGE'`), on ne le calcule qu'aux instants d'archivage (voir §15). Si on précise `INST_CALCUL='TOUT_PAS'`, on calculera le critère à tous les pas de temps, même ceux qui ne sont pas archivés. Dans le cas où l'instant de calcul n'est pas archivé, les valeurs (charge critique et mode de flambement) ne seront pas stockées dans la sd `evol_noli` mais les charges critiques seront imprimées dans le fichier `.mess`.

3.16 Mot-clé `MODE_VIBR`

```
◇  MODE_VIBR =_F (
    ◇  NB_FREQ          = / 3, [DEFAULT]
                        / nbfreq, [I]
    ◇  MATR_RIGI         = / 'ELASTIQUE', [DEFAULT]
                        / 'TANGENTE',
                        / 'SECANTE',
    ◇  BANDE             = intba , [listr8]
    ◇  INST_CALCUL       = / 'LISTE_ARCHIVAGE', [DEFAULT]
                        / 'TOUT_PAS',
    ),
```

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 $\det(\mathbf{K} - \omega^2 \cdot \mathbf{M}) = 0$. \mathbf{K} peut soit être la matrice de raideur élastique, soit la matrice tangente cohérente à l'instant courant, soit la matrice sécante. \mathbf{M} est la matrice de masse.

Le mot-clé `NB_FREQ` (3 par défaut) désigne le nombre de fréquences propres à calculer.

On stocke le mode propre correspondant à la plus petite fréquence propre dans la S.D. `RESULTAT`, sous le nom `MODE_MECA`. Ce mode propre peut être extrait et visualisé (comme un champ de déplacements ou un mode propre classique). Il est normalisé à 1 sur la plus grande composante de déplacement.

Le mot-clé `BANDE` permet de spécifier sur quelle bande de fréquence on veut faire la recherche de fréquences propres.

Le mot-clé `INST_CALCUL` a le même rôle que pour `CRIT_FLAMB`. Il permet de spécifier à quels instants on va mener l'analyse modale vibratoire. Par défaut, on ne va effectuer ce calcul qu'aux instants d'archivage. Si l'on veut faire ce calcul à tous les pas, il faut spécifier la valeur `'TOUT_PAS'`. Rappelons que l'on ne stocke dans l'objet résultat que les données aux instants d'archivage (si un archivage est spécifié, dans le cas contraire on stocke les données à tous les pas). Donc si l'on utilise l'option `'TOUT_PAS'` avec un archivage, les données calculées par l'analyse modale vibratoire qui correspondent à des pas non archivés ne seront pas stockées. La seule trace restante sera l'affichage des fréquences propres dans le fichier de messages `.mess`.

3.17 Opérandes `SENSIBILITE`

```
◇  SENSIBILITE =liste de paramètres sensibles [l_para_sensi]
```

Active le calcul de la dérivée des champs de déplacement, vitesse et accélération par rapport à un paramètre sensible du problème.

Le document [U4.50.01] précise le fonctionnement du mot clé.

3.18 Opérande PROJ_MODAL

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.

3.18.1 Opérandes MODE_MECA, NB_MODE

```
♦  MODE_MECA  = mode,          [mode_meca]
                               [base_modale]
◇  NB_MODE    =  /  nbmode,     [I]
                  /  9999,      [DEFAULT]
```

On spécifie la base à utiliser (MODE_MECA ou BASE_MODALE) 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. [§16]) : même profil de numérotation.

3.18.2 Opérandes MASS_GENE , RIGI_GENE , AMOR_GENE

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-linéaire. 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]. La syntaxe de ces opérandes se présente ainsi :

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

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.

3.19 Mot clé EXCIT_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.

```
◇  FONC_MULT = fomult,          [fonction_sdaster]
♦  VECT_GENE = vecgen,          [vect_asse_gene]
```

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.

3.20 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.

3.21 Opérande TITRE

◇ TITRE = tx

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

4 Exemple : mouvement d'un pendule de grande amplitude

```
# TITRE Pendule simple en grande oscillation
#
# PENDULE CONSTITUE D'UN ELEMENT DE CABLE (test SDNL100A).
#
DEBUT ();
#
MA=LIRE_MALLAGE( )
MA=DEFI_GROUP( reuse=MA, MALLAGE=MA,
               CREA_GROUP_MA=_F( NOM = 'TOUT', TOUT = 'OUI'))
MO=AFFE_MODELE( MALLAGE = MA, AFFE=_F(GROUP_MA = 'CABLE',
    PHENOMENE = 'MECANIQUE', MODELISATION = 'CABLE') )
MAT=DEFI_MATERIAU( ELAS=_F( E = 1.E8, NU=0., RHO = 1.),
                  CABLE=_F(EC_SUR_E = 1.E0),)
CHMAT=AFFE_MATERIAU( MALLAGE=MA, AFFE=_F( TOUT = 'OUI', MATER = MAT),)
CHA1=AFFE_CHAR_MECA( MODELE=MO,DDL_IMPO=(
    _F( NOEUD = 'N1', DX = 0., DY = 0., DZ = 0.),
    _F( NOEUD = 'N2', DY = 0.)) )
CHA2=AFFE_CHAR_MECA( MODELE=MO, PESANTEUR=(9.81,0.,0.,-1.,) )
CARA=AFFE_CARA_ELEM( MODELE=MO, CABLE=_F( GROUP_MA = 'TOUT',SECTION = 1.))
L_ARCHI=DEFI_LIST_ENTI( DEBUT=1,INTERVALLE=(
    _F( JUSQU_A = 11, PAS = 1),
    _F( JUSQU_A = 21, PAS = 2),
    _F( JUSQU_A = 41, PAS = 5)))
L_INST1=DEFI_LIST_REEL( DEBUT=0.,
    INTERVALLE=_F( JUSQU_A = 1.6744, NOMBRE = 40))
L_INST2=DEFI_LIST_REEL( DEBUT=0.,INTERVALLE=(
    _F( JUSQU_A = 0.4186, NOMBRE = 10),
    _F( JUSQU_A = 0.8372, NOMBRE = 5),
    _F( JUSQU_A = 1.6744, NOMBRE = 4)))
#
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'),
    COMP_ELAS=_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).

5 Bibliographie

- M. AUFAURE : Méthodes directes d'analyse dynamique des structures en non-linéaire. Note HI-70/93/124.
- N. GREFFET : « Vers de nouvelles méthodes numériques pour l'intégration temporelle dans le *Code_Aster* »
Note HT-62/04/016.